Do Feature Interactions Break Explainability? Provable Explanations for Any-Order Neural Additive Models

Idan Refaeli[*]
The Hebrew University of Jerusalem

Yizhak Yisrael Elboher
The Hebrew University of Jerusalem
Masaryk University

Guy Katz
The Hebrew University of Jerusalem

Shahaf Bassan
The Hebrew University of Jerusalem

Abstract

Post-hoc explanation methods for neural networks are often heuristic and lack provable guarantees. A common alternative to such methods that was studied in recent years is to compute a *cardinally-minimal* subset of input features that is *provably sufficient* to determine the model's prediction. For general neural networks, however, finding such explanations is computationally intractable. In contrast, recent work showed that this problem becomes more tractable for a particular family of neural networks, namely neural additive models (NAMs), though standard NAMs are highly restrictive because they exclude feature interactions altogether. In this work, we study the computation of provably minimal and sufficient explanations for neural networks with restricted *feature interactions*, bridging the gap between fully general neural networks and purely additive models. We first prove that even with constant-order interactions, the problem is Σ_2^P-hard, implying that worst-case exponential complexity is unavoidable. We then identify two provably tractable settings: one based on sparsity in the interaction structure and another based on a stronger interaction-aware notion of sufficiency. Under either of these settings, we develop algorithms that compute provable explanations for neural networks with any-order interactions while retaining much of the computational efficiency of additive models. Empirically, we show that these explanations are significantly smaller and faster to obtain than those produced by standard algorithms, while being derived from models that achieve much higher accuracy than standard NAMs. Overall, our results help expand the class of neural networks that admit efficient provable explanations, while clarifying the fundamental role of feature interactions in shaping their complexity.

1 Introduction

Understanding why neural networks make particular predictions remains one of the central challenges in trustworthy ML. A significant amount of prior work has proposed *post-hoc explanation* methods, aiming to identify which input features are most responsible for a model's decision. Popular approaches such as LIME [91], SHAP [79], and Integrated Gradients [99] assign attribution scores to features, while subset-selection based methods such as Anchors [92] and SIS [36] seek small subsets of observed features that suffice to justify a prediction. These methods have become standard tools in practice, but they remain largely heuristic: their outputs typically rely on local approximations, sampling, or specific perturbation assumptions, and therefore do not provide provable guarantees.

[*]Corresponding author: idan.refaeli@mail.huji.ac.il.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

Recent work has increasingly focused on *provable explanations* [9, 19, 42, 61, 64, 74, 110]: explanations equipped with formal certificates guaranteeing that the model's prediction is preserved under all admissible completions of the unobserved features. Among these, *provably sufficient explanations* [19, 61, 64, 110] provide a particularly strong guarantee: fixing only the features in the explanation suffices to preserve the prediction regardless of how the remaining features vary. However, sufficiency alone is not enough: the full input trivially satisfies this property. For this reason, the gold-standard objective is to compute *cardinally-minimal sufficient explanations* [19, 61, 64, 110], which identify the smallest subset of features that provably preserves the prediction. In practice, most formal explainability work instead targets the weaker *subset-minimal* notion, since cardinal minimality is harder to compute; cardinal minimality still provides both formal correctness and conciseness, making it one of the strongest explanation guarantees studied in the literature.

Unfortunately, these guarantees come at a steep computational cost. For general neural networks, computing cardinally-minimal provably sufficient explanations is known to be Σ_2^P-hard [13], placing the problem at the second level of the polynomial hierarchy. Intuitively, this means that this problem is substantially harder than NP-Complete problems: it requires dispatching, in the worst-case, an *exponential* number of neural-network verification queries, each of which is itself NP-hard. Thus, while these provable explanations offer stronger guarantees than heuristic alternatives, they become computationally intractable precisely for the models where explanations are most needed.

A recent line of work provides an encouraging exception by focusing on a specific family of neural networks: *Neural Additive Models* (NAMs) [2]. These networks decompose predictions into sums of feature-specific components, and have received substantial attention in recent years [2, 24, 49, 72, 90, 118]. Recent work [21] showed that cardinally-minimal provably sufficient explanations can be computed in this setting using only a worst-case *linear* number of verification queries. This demonstrates that model structure can fundamentally change the computational landscape of computing provable explanations. However, standard NAMs are highly restrictive. By construction, they exclude feature interactions entirely, forcing each feature to contribute independently to the prediction. While this pure additivity enables tractability, it limits expressivity and can reduce predictive performance relative to models that explicitly represent interactions [37, 48, 78]. In many real-world settings, predictions depend critically on feature interactions, making this restriction a practical limitation.

Our contributions. In this work, we study the space between these two extremes. Purely additive NAMs correspond to first-order interaction structure, where provably minimal explanations are tractable. Fully general neural networks, by contrast, can encode arbitrary interactions among features, and in that setting the problem is Σ_2^P-hard. We study the intermediate case in which models contain explicit higher-order interactions, but those interactions remain structured rather than completely unrestricted. This perspective lets us isolate how the order and connectivity of feature interactions shape the complexity of computing provable minimal explanations for neural networks.

Our main contributions are as follows:

- **Hardness under limited interactions.** We prove that finding cardinally-minimal provably sufficient explanations for Any-Order NAMs is Σ_2^P-hard, even for neural networks over bounded *constant*-order interactions. Thus, unlike standard NAMs, even limited interactions make exact provable explanation search intractable.
- **Two tractable settings.** Interestingly, despite our negative complexity result, we recover tractability in two structural settings: (i) a stronger interaction-aware notion of sufficiency that decouples components; and (ii) sparse interaction hypergraphs with bounded overlap.
- **Certified algorithms for both settings.** We design neural network verification- and MIP-based algorithms for both settings: the verifier computes provable component bounds, and the MIP solver searches globally for an optimal minimum over feature subsets.
- **Minimality guarantees.** In the sparse setting, our method returns minimum-cardinality provably sufficient explanations for the original model. In the strongly-sufficient explanation setting, it returns minimum-cardinality explanations under the stronger notion, which still imply standard sufficiency while retaining significantly smaller explanations in practice.
- **Empirical gains.** We perform extensive experiments across 9 benchmarks, and show that our methods produce significantly smaller explanations than previously suggested algorithms,

often with better runtime, with models that are significantly more accurate than standard NAMs. They also provide guarantees unavailable to sampling-based interpretation methods.

Overall, our results significantly expand the class of neural architectures for which computing provably minimal and sufficient explanations are possible, while clarifying the precise role of feature interactions in determining their computational complexity.

Due to space constraints, we present the central ideas behind selected propositions in the main text and defer the full proofs to the appendix.

2 Preliminaries

Notation. Let $[n] := \{1, \dots, n\}$ denote the feature indices, and let $\mathcal{D} \subseteq \mathcal{X}_1 \times \cdots \times \mathcal{X}_n$ be the input domain. For ease of presentation, we assume binary classification, where a score function $F : \mathcal{D} \to \mathbb{R}$ induces the classifier $h_F(x) := \mathbb{I}[F(x) \geq 0]$; a description of the regression and multiclass settings is given in Appendix D. For $x \in \mathcal{D}$ and $S \subseteq [n]$, let x_S denote the restriction of x to S, let $\bar{S} := [n] \setminus S$, and define $\mathcal{C}(x, S) := \{z \in \mathcal{D} : z_S = x_S\}$ as the set of admissible completions consistent with x on S. For any $z \in \mathcal{D}$, we write $(x_S, z_{\bar{S}})$ for the hybrid input that agrees with x on S and with z on $\bar{S}$.

2.1 From First-Order to Any-Order NAMs

Standard (first-order) NAMs decompose the score into a sum of one-dimensional component functions. This first-order structure makes the contribution of each feature explicit and completely removes feature interactions. It is precisely this restriction that underlies the tractability guarantees known for standard NAM explanations [2, 20]. However, many realistic prediction tasks depend on interactions between multiple features. Any-Order NAMs extend this additive decomposition by allowing each component to depend on an arbitrary feature subset rather than on just one coordinate.

Definition 1 (Any-Order NAM)**.** *An* Any-Order NAM *has the form*

$$F(x) := b + \sum_{T \in \mathcal{H}} f_T(x_T),$$

where $\mathcal{H} \subseteq 2^{[n]} \setminus \{\emptyset\}$ is a family of interaction scopes and each f_T depends only on the coordinates in T [71]. The order of the model is $\max_{T \in \mathcal{H}} |T|$.

First-order NAMs are the special case $\max_{T \in \mathcal{H}} |T| = 1$, so they exclude feature interactions. Any-Order NAMs preserve additivity while allowing higher-order interactions, making them more expressive and closer to general neural networks (which allow arbitrary interactions). We ask how much of the tractability of first-order NAM explanations survives in this richer setting.

2.2 Neural Network Verification

Neural network verification asks whether a network satisfies a property over all inputs in a given set. Formally, given a neural network $F : \mathcal{D} \to \mathbb{R}$, an input set $\mathcal{X} \subseteq \mathcal{D}$, and a property P, the verification problem is to determine whether $\forall z \in \mathcal{X},\ P(F, z)$ holds. In classification, the property of interest is often label preservation. For fixed input x and completion set $\mathcal{C}(x, S)$, this asks whether

$$h_F(z) = h_F(x) \qquad \text{for all } z \in \mathcal{C}(x, S).$$

This problem is NP-hard in general [68, 93]. However, recent years have seen rapid advances in neural network verification, substantially improving the scalability and practical effectiveness of verifiers [40, 70, 109]. In particular, verifiers in the CROWN family, such as the state-of-the-art verifier α, β-CROWN [70, 73, 108, 117, 120], propagate provable bounds on the value of each neuron; these bounds can be extracted, and we will use them later.

3 Provable Explanations for Neural Networks and NAMs

In safety-critical settings, one often seeks explanations with formal guarantees rather than heuristic scores. For a fixed input, the key question is whether a subset of observed features preserves

the prediction under all admissible completions of the remaining features, often restricted to a local neighborhood. For $p \in [1, \infty]$ and $\epsilon > 0$, let $\mathcal{B}_{p,\epsilon}(x) := \{z \in \mathcal{D} : \|z - x\|_p \leq \epsilon\}$ and $\mathcal{C}_{p,\epsilon}(x, S) := \mathcal{C}(x, S) \cap \mathcal{B}_{p,\epsilon}(x)$.

Definition 2 (Sufficient Explanation)**.** *For a given neural network F, a fixed input $x \in \mathcal{D}$, a norm parameter $p \in [1, \infty]$, and a perturbation radius $\epsilon > 0$, a set $S \subseteq [n]$ is a* sufficient explanation *for the query $\langle F, x, p, \epsilon, S\rangle$ if*

$$\forall z \in \mathcal{C}_{p,\epsilon}(x, S), \qquad h_F(z) = h_F(x).$$

Sufficiency alone does not guarantee that an explanation is concise or informative. The full feature set $[n]$ is always a trivially provably sufficient explanation. This motivates the minimality notions below, which seek the smallest feature subset that still certifies the prediction.

Definition 3 (Minimal Sufficient Explanations)**.** *For a given neural network F, a fixed input $x \in \mathcal{D}$, a norm parameter $p \in [1, \infty]$, and a perturbation radius $\epsilon > 0$, a provably sufficient explanation $S^* \subseteq [n]$ for the query $\langle F, x, p, \epsilon, S^*\rangle$ may satisfy either of the following minimality notions:*

- *(i) it is* cardinally-minimal *if $|S^*| \leq |S|$ for every provably sufficient explanation $S \subseteq [n]$ for the query $\langle F, x, p, \epsilon, S\rangle$;*
- *(ii) it is* subset-minimal *if no proper subset $S \subsetneq S^*$ is itself a provably sufficient explanation for the query $\langle F, x, p, \epsilon, S\rangle$.*

These notions capture different goals: cardinal minimality optimizes size globally, whereas subset minimality only removes redundant revealed features. Most neural-network explanation methods target the weaker subset-minimal notion [21], typically via greedy remove-and-verify: start from the full feature set and delete a feature whenever a verifier still certifies sufficiency. The result is subset-minimal, but can depend strongly on the deletion order; sensitivity-based variants differ only in how they choose that order. We use this baseline later and give pseudocode in Appendix C as Algorithm 2.

For general neural networks, computing cardinally-minimal provably sufficient explanations is already Σ_2^P-hard, so exact provable minimal explanations are intractable in full generality; see Appendix C for brief background on this complexity class. NAMs offer an exception. In first-order NAMs, the absence of feature interactions enables linear-query algorithms for cardinally-minimal provable explanations [21]. This raises the key question of our work: what happens when interactions are reintroduced? Any-Order NAMs preserve additivity at the component level while allowing higher-order interactions, improving expressivity and often accuracy [71]. Section 4 shows that interactions restore hardness, while Sections 5 and 6 recover tractability under additional structure.

4 Hardness of Computing Minimal Explanations for Constant-Order NAMs

Following previous complexity-theoretic works on sufficient explanations [1, 6, 8, 13, 16], we study the decision variant of minimum-size sufficient explanation search: given a budget B, decide whether there exists a sufficient explanation of size at most B. Formally:

Minimum Sufficient Explanation for k-Order NAMs (MSE-k-NAM).
Given an Any-Order NAM F of order $k = \max_{T \in \mathcal{H}} |T|$, an input $x \in \mathcal{D}$, a norm parameter $p \in [1, \infty]$, a perturbation radius $\epsilon > 0$, and an integer budget $B \in \mathbb{N}$, decide whether there exists a sufficient explanation $S \subseteq [n]$ with respect to $\langle F, x, p, \epsilon, S\rangle$ such that $|S| \leq B$.

We show that this problem is already hard even for constant k:

Theorem 4. *There exists a fixed constant $k = \mathcal{O}(1)$ for which MSE-k-NAM is Σ_2^P-hard.*

In other words, Theorem 4 shows that allowing even a constant degree of feature interactions in a NAM raises the complexity of computing cardinally-minimal sufficient explanations to that of fully general neural networks [13]. In particular, the Σ_2^p-hardness result implies, under standard complexity assumptions, that this task cannot be solved using even a polynomial number of (NP-Complete) neural-network verification queries, and therefore requires *exponentially* many such queries in the worst case. This stands in sharp contrast to first-order neural networks, namely "classic" NAMs,

where cardinally-minimal sufficient explanations can be computed using a polynomial, and in fact *linear*, number of verification queries [21]. Thus, this theorem shows that feature interactions play a central role in the computational hardness of obtaining minimal sufficient explanations for neural networks, and this hardness already appears at very low, constant interaction order.

Interestingly, we show in Sections 5 and 6 that despite this significantly negative worst-case complexity result, tractability can nevertheless still be recovered in practice under one of two assumptions: (i) a stronger interaction-aware notion of sufficiency or (ii) a sparsity condition on the interaction graph.

5 A Tractable Relaxation via Strongly Sufficient Explanations

Section 4 shows that, in general, computing cardinally-minimal sufficient explanations for NAMs of a fixed order is Σ_2^P-hard. Thus, in the worst case, exact optimization requires an exponential number of neural-network verification queries. To overcome this barrier, we introduce a stronger interaction-aware notion of explanation sufficiency, which we call *strongly sufficient* explanations. This notion is conservative: a strongly sufficient explanation is always a sufficient explanation in the ordinary sense, but it need not be cardinally-minimal among all ordinary sufficient explanations.

The advantage is computational. Under strong sufficiency, the effect of each interaction component can be bounded independently, allowing us to compute provably *cardinally-minimal strongly sufficient explanations* using only a worst-case *linear* number of verification-query rounds. Although these explanations optimize a stronger notion than ordinary sufficiency, our experiments show that they are often much smaller than the explanations produced by standard greedy methods for sufficient explanations, suggesting that they are also practically closer to the true ordinary optimum.

Strongly sufficient explanations. Under sufficiency, all omitted features must be completed by a single globally consistent input. Strong sufficiency removes this coupling and lets each component choose its own local completion. For a component $T \in \mathcal{H}$ and a set of features $U \subseteq T$, let

$$\mathcal{C}^T_{p,\epsilon}(x, U) := \{z_T \in \mathcal{D}_T : z_{T\setminus U} = x_{T\setminus U},\ \|z_T - x_T\|_p \le \epsilon\}.$$

For a candidate explanation $S \subseteq [n]$, write $U_T(S) := T \setminus S$.

Definition 5 (Strongly Sufficient Explanations)**.** *We say that S is a* strongly sufficient explanation *for the query $\langle F, x, p, \epsilon, S\rangle$ if, for every choice of local completions $z_T \in \mathcal{C}^T_{p,\epsilon}(x, U_T(S))$ for all $T \in \mathcal{H}$,*

$$\mathbb{I}\left[b + \sum_{T\in\mathcal{H}} f_T(z_T) \ge 0\right] = h_F(x).$$

Proposition 6. *For a given Any-Order NAM F, a fixed input $x \in \mathcal{D}$, a norm parameter $p \in [1, \infty]$, a perturbation radius $\epsilon > 0$, and a candidate explanation $S \subseteq [n]$, if S is strongly sufficient for the query $\langle F, x, p, \epsilon, S\rangle$, then S is also a sufficient explanation for the same query.*

Example. Let $x_1, x_2, x_3 \in \{0, 1\}$, let F have two components $f_{12}(x_1, x_2)$ and $f_{23}(x_2, x_3)$ (so $b = 0$), and consider $x = (1, 0, 1)$ with candidate explanation $S = \{1, 3\}$ (freeing only x_2). Suppose $f_{12}(1, 0){=}{-}2$, $f_{12}(1, 1){=}3$, $f_{23}(0, 1){=}3$, $f_{23}(1, 1){=}{-}2$. Under ordinary sufficiency, x_2 takes one shared value: both completions ($x_2{=}0$: $-2+3=1$; $x_2{=}1$: $3-2=1$) are positive, so S is sufficient. Under strong sufficiency, each component gets its own private copy of x_2: taking $x_2{=}0$ in f_{12} and $x_2{=}1$ in f_{23} gives $-2-2=-4$, so S is *not* strongly sufficient. An ordinary completion is exactly the special case where the private copies agree, which is why Proposition 6 holds.

Thus, strong sufficiency is a conservative relaxation: it may exclude some ordinary sufficient explanations, but every strongly sufficient explanation is still certifiably sufficient. The central benefit is that, once S is fixed, each component can now be analyzed independently through the free set $U_T(S)$.

5.1 Finding Minimal Deviations in Each Interaction Component

To exploit this decoupling, we track how much each component can reduce the signed score when a subset of its features is left free. Let $\sigma(x) = 2h_F(x) - 1 \in \{-1, 1\}$. For a component $T \in \mathcal{H}$ and a set $U \subseteq T$, define the corresponding deviation $\Delta_T(x, U) := \sigma(x) f_T(x_T) - \min_{z_T \in \mathcal{C}^T_{p,\epsilon}(x,U)} \sigma(x) f_T(z_T)$. Thus $\Delta_T(x, U)$ is the worst signed margin loss contributed by component T when the features in U are allowed to vary. In particular, $\Delta_T(x, \emptyset) = 0$. Because strong

sufficiency lets the components choose their completions independently, the total worst-case margin loss is the sum of these deviations.

Proposition 7. *For a given Any-Order NAM F, a fixed input $x \in \mathcal{D}$, a norm parameter $p \in [1, \infty]$, a perturbation radius $\epsilon > 0$, and a candidate explanation $S \subseteq [n]$, the set S is strongly sufficient for the query $\langle F, x, p, \epsilon, S \rangle$ if and only if*

$$\sum_{T \in \mathcal{H}} \Delta_T\big(x, U_T(S)\big) \leq |F(x)|. \tag{1}$$

Proposition 7 is the key reduction: once the values $\Delta_T(x, U)$ are known, finding a strongly sufficient explanation becomes a finite combinatorial optimization problem.

5.2 Finding Cardinally-Minimal Strongly Sufficient Explanations Using MILP

Assume first that the exact deviation values $\Delta_T(x, U)$ are available for every component $T \in \mathcal{H}$ for every set $U \subseteq T$. We can then search for a smallest strongly sufficient explanation with one MILP. We use a binary variable b_i for each feature $i \in [n]$, where $b_i = 1$ means that feature i is included in the explanation. We also use a binary variable $c_{T,U}$ for each pair (T, U), where $c_{T,U} = 1$ means that U is the free set chosen inside component T. The exact MILP is

$$\text{Objective:} \quad \min \sum_{i=1}^{n} b_i \tag{2}$$

subject to

- $\sum_{U \subseteq T} c_{T,U} = 1$ $\qquad \forall T \in \mathcal{H},$
- $c_{T,U} \leq 1 - b_i$ $\qquad \forall T \in \mathcal{H},\ \forall U \subseteq T,\ \forall i \in U,$
- $c_{T,U} \leq b_i$ $\qquad \forall T \in \mathcal{H},\ \forall U \subseteq T,\ \forall i \in T \setminus U,$
- $\sum_{T \in \mathcal{H}} \sum_{U \subseteq T} \Delta_T(x, U) c_{T,U} \leq |F(x)|$
- $b_i \in \{0, 1\}$ $\qquad \forall i \in [n],$
- $c_{T,U} \in \{0, 1\}$ $\qquad \forall T \in \mathcal{H},\ \forall U \subseteq T.$

The first three constraint families ensure that $c_{T,U}$ encodes the free set induced by the chosen explanation, and the last constraint is exactly the strong-sufficiency condition from Proposition 7.

Proposition 8. *If* (2) *uses exact deviation values $\Delta_T(x, U)$, every optimal solution encodes a cardinally-minimal strongly sufficient explanation, and vice versa.*

The full algorithm. Computing every exact deviation $\Delta_T(x, U)$ is computationally infeasible. We therefore use DNN verification to compute bounds $\underline{\Delta}_T(x, U) \leq \Delta_T(x, U) \leq \overline{\Delta}_T(x, U)$, and obtain *optimistic* and *pessimistic* MILP programs by replacing $\Delta_T(x, U)$ in (2) with the lower and upper bounds, respectively. Appendix F.1.1 gives the details of the verifier encoding and this MILP construction. If the optimistic and pessimistic MILPs return different objective values, then some local deviation interval is still too wide. We then use bisection to narrow the intervals $[\underline{\Delta}_T(x, U), \overline{\Delta}_T(x, U)]$ for the unresolved pairs (T, U); Appendix F.1.1 gives the details.

All local pairs (T, U) can be processed independently, so initialization and refinement are fully parallelizable. Algorithm 1 summarizes the continuous-domain procedure. Appendix F.1.2 gives the mixed discrete/continuous variant, where finite completion sets are handled exactly by enumeration.

Algorithm 1 Strong-sufficiency algorithm for continuous domains

Input: Any-Order NAM F, input x, norm parameter $p \in [1, \infty]$, perturbation radius $\epsilon > 0$

for all $T \in \mathcal{H}$ and $U \subseteq T$ **in parallel do**
  Compute sound bounds $\underline{\Delta}_T(x, U) \leq \Delta_T(x, U) \leq \overline{\Delta}_T(x, U)$
while true **do**
  Solve the pessimistic MILP obtained from (2) using $\overline{\Delta}_T(x, U)$
  Let $S^{\uparrow} \leftarrow \{i \in [n] : b_i^{\uparrow} = 1\}$ and let $v^{\uparrow}$ be its objective value
  Solve the optimistic MILP obtained from (2) using $\underline{\Delta}_T(x, U)$
  Let $v^{\downarrow}$ be its objective value
  if $v^{\uparrow} = v^{\downarrow}$ **then**
    return $S^{\uparrow}$
  for all (T, U) with $\underline{\Delta}_T(x, U) < \overline{\Delta}_T(x, U)$ **in parallel do**
    Refine $\underline{\Delta}_T(x, U)$ and $\overline{\Delta}_T(x, U)$ by one bisection step

Theorem 9. *Assume that $\forall T \in \mathcal{H}, U \subseteq T$, the local optimization problem for $\Delta_T(x, U)$ is finitely encoded and verifier calls return rational bounds. Then Algorithm 1 terminates after linearly many outer refinement rounds in the encoding size of the Any-Order NAM, and returns a cardinally-minimal strongly sufficient explanation for $\langle F, x, p, \epsilon, S \rangle$. By Proposition 6, this set is also sufficient.*

Complexity. Each iteration of Algorithm 1 makes three calls to an NP oracle: one neural-network verification query and two MILP queries. By Theorem 9, the number of iterations is linear in the size of the model. Thus, the algorithm makes a *linear* number of NP-oracle calls overall. This stands in sharp contrast to the ordinary sufficiency setting, which is Σ_2^P-hard (Theorem 4) and therefore, under standard complexity assumptions, cannot be solved with polynomially many oracle calls.

6 A Tractable Relaxation for k-Sparse Any-Order NAMs

Section 4 shows that constant-order interactions are enough to make cardinally-minimal sufficient explanations Σ_2^P-hard. Section 5 recovered tractability by strengthening the notion of sufficiency, yielding explanations that are certified sufficient but minimal only under the stronger notion. Here, we show that when the interaction structure is sparse, no such relaxation is needed. By merging overlapping scopes into small disjoint components, ordinary sufficiency decomposes across components, allowing us to apply the same linear-query algorithm. Thus, for sparse Any-Order NAMs, we recover a complexity guarantee matching first-order NAMs: cardinally-minimal sufficient explanations for the original model can be computed using only a linear number of verification-query rounds.

6.1 Merging Interaction Components

Consider an Any-Order NAM $F(x) = b + \sum_{T \in \mathcal{H}} f_T(x_T)$. Define the feature-interaction graph $\mathcal{G}_{\mathcal{H}} = ([n], E)$ by placing an edge between two features $i, j \in [n]$ whenever there exists a scope $T \in \mathcal{H}$ with $i, j \in T$. Let $C_1, \ldots, C_m$ be the connected components of $\mathcal{G}_{\mathcal{H}}$, and call them the *merged components* of F. For each merged component C_ℓ, define the merged score $G_\ell(x_{C_\ell}) := \sum_{T \in \mathcal{H}: T \subseteq C_\ell} f_T(x_T)$. Then the model can be regrouped as $F(x) = b + \sum_{\ell=1}^{m} G_\ell(x_{C_\ell})$. This preprocessing only changes how the terms are grouped; it does not change the model.

Lemma 10. *Let $C_1, \ldots, C_m$ be the merged components of F. Then the sets $C_1, \ldots, C_m$ are pairwise disjoint, so each input feature belongs to at most one merged component. Moreover, every interaction scope $T \in \mathcal{H}$ is contained in a unique merged component C_ℓ.*

6.2 Cardinally-Minimal Provably Sufficient Explanations in k-Sparse Any-Order NAMs

Definition 11 (s-Sparse Any-Order NAM)**.** *Let $C_1, \ldots, C_m$ be the merged components of an Any-Order NAM F. We say that F is* s-sparse *if $\max_{\ell \in [m]} |C_\ell| \leq s$. (We use s rather than k for this bound to keep it distinct from the interaction order k of an Any-Order NAM.)*

Thus s bounds the largest merged component. When s is small, each score G_ℓ depends on only a few features. We therefore treat each C_ℓ as one component and run Algorithm 1 on the regrouped model $\widetilde{F}(x) = b + \sum_{\ell=1}^{m} G_\ell(x_{C_\ell})$. No new optimization machinery is needed: Algorithm 1 simply computes deviations for merged components rather than original scopes. The mixed discrete/continuous variant

is obtained analogously in Appendix F.1.2. By Lemma 10, different merged components do not interact. Hence $\widetilde{F}$ is exactly the disjoint-component model handled in Section 5, and strong sufficiency for $\widetilde{F}$ coincides with ordinary sufficiency for the original model.

Theorem 12. *Let F be an s-sparse Any-Order NAM with merged components $C_1, \ldots, C_m$ and merged scores $G_1, \ldots, G_m$. Fix a query $\langle F, x, p, \epsilon \rangle$. Assume that every local optimization problem for the merged components is finitely encoded and that each verifier call returns sound rational bounds. Then applying Algorithm 1 to the regrouped model $\widetilde{F}$ terminates after a linear number of outer refinement rounds and returns a cardinally-minimal sufficient explanation for the original model F.*

Complexity. Preprocessing requires only graph connectivity (e.g., union-find). Theorem 9 then applies to the regrouped model, with every NP oracle call to a neural network verifier or MILP verifier restricted to a merged component of size at most s. Hence, for constant s, the number of such calls is linear in the model size.

7 Empirical Evaluation

Experimental setup. We evaluate the strong-sufficiency algorithm (Section 5) and the preprocessing-based sparse reduction (Section 6) against two verifier-backed greedy baselines: canonical-order greedy and a sensitivity-guided variant. All methods use α-β-CROWN, the state-of-the-art in neural network verification, for the verification queries [117]; our methods additionally solve optimistic and pessimistic MILPs with Gurobi [55], the state-of-the-art MILP solver. Both baselines use greedy remove-and-verify from the full feature set and differ only in feature order, so they are guaranteed only to return subset-minimal explanations rather than globally cardinally-minimal ones. Since all methods share the same verifier backend, any gains reflect better use of model structure rather than a weaker certification standard. Exact sample counts, feature counts, train/test splits, and ablations appear in Appendix G and Appendix H.

7.1 Provably cardinally-minimal Strongly Sufficient Explanations for Any-Order NAMs

Table 1 reports the main strong-sufficiency results. For each dataset in this experiment, we train a second-order Any-Order NAM with components for all singleton features and all pairs of input features. It compares our strongly-sufficient algorithm against these two greedy subset-minimal baselines in terms of the mean explanation size they produce and the mean runtime required to compute them. Overall, our method achieves the best mean runtime on most benchmarks. Mean explanation size is more mixed: since strong sufficiency is a strictly more conservative notion than ordinary sufficiency (Proposition 6), it carries no guarantee of beating a subset-minimal baseline on size, and empirically our method attains the smallest mean size on a minority of benchmarks, remaining close to the best baseline on the rest. The sparse setting (Section 6), in contrast, does carry such a guarantee and wins on size across the board, as Table 2 below confirms.

Table 1: Strong-sufficiency results on nine order-2 tabular benchmarks at $\epsilon = 0.1$. Columns show test accuracy, mean explanation size $\pm$ standard deviation (% of features), and mean runtime $\pm$ standard deviation.

Dataset	Acc.	Size mean $\pm$ sd (%) (Ours / greedy / sens.)	Time mean $\pm$ sd (s) (Ours / greedy / sens.)
Adult	0.8635	**25.7**$\pm$13.2 / 45.2$\pm$21.7 / 52.8$\pm$29.1	**62.943**$\pm$364.322 / 2199.624$\pm$1060.915 / 2267.351$\pm$1467.501
Bike Sharing	0.9232	**33.5**$\pm$17.8 / 36.3$\pm$19.6 / 44.5$\pm$15.7	**61.279**$\pm$314.891 / 1137.386$\pm$1351.496 / 2666.257$\pm$1261.087
Breast Cancer	0.9825	4.0$\pm$16.3 / 3.5$\pm$15.9 / **2.7**$\pm$13.2	**126.792**$\pm$447.934 / 3082.496$\pm$402.879 / 3098.785$\pm$365.728
California Housing	0.8886	19.7$\pm$21.8 / 16.7$\pm$19.7 / **16.0**$\pm$18.8	71.860$\pm$412.863 / 18.732$\pm$6.640 / **17.525**$\pm$5.692
COMPAS	0.6742	50.6$\pm$23.7 / 50.3$\pm$10.1 / **39.9**$\pm$12.0	**26.411**$\pm$113.220 / 2805.141$\pm$1357.731 / 1806.214$\pm$1535.080
Diabetes	0.7273	10.2$\pm$16.6 / 11.1$\pm$19.4 / **9.8**$\pm$17.0	**38.469**$\pm$42.145 / 169.473$\pm$105.444 / 141.862$\pm$79.488
German Credit	0.7250	**64.9**$\pm$14.2 / 77.7$\pm$3.6 / 73.9$\pm$3.1	**136.689**$\pm$257.987 / 3550.136$\pm$419.894 / 3569.952$\pm$275.462
Heart Disease	0.8525	29.1$\pm$14.4 / **27.1**$\pm$10.3 / 34.2$\pm$7.5	**130.520**$\pm$54.842 / 2765.516$\pm$1121.928 / 3369.266$\pm$695.254
HELOC	0.7245	**38.4**$\pm$22.1 / 72.0$\pm$5.4 / 77.5$\pm$10.0	**910.066**$\pm$557.189 / 3596.063$\pm$251.951 / 3599.857$\pm$186.663

Figure 1 shows the COMPAS results; Appendix G.6 gives separate figures for the remaining benchmarks.

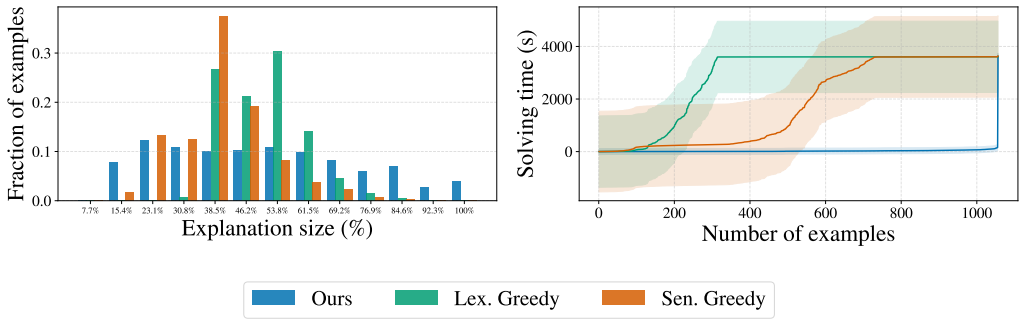

Figure 1: COMPAS under strong sufficiency. Combined explanation-size and runtime plot.

7.2 Provably cardinally-minimal Sufficient Explanations for Sparse Any-Order NAMs

Table 2 reports the sparse-setting results. As in Table 1, it compares our sparse algorithm against the same two greedy baselines, which compute subset-minimal explanations, in terms of the mean explanation size they find and the mean runtime required to compute them. In the sparse setting, our preprocessing-based method finds substantially smaller mean explanation sizes than either greedy baseline across all nine benchmarks, often by a wide margin, matching the guarantee of Theorem 12; it also achieves the best mean runtime on most benchmarks, losing only on California Housing, where the feature set is small enough that greedy's cheaper per-query cost outweighs our method's advantage in query count.

Table 2: Sparse-setting results on nine order-2 tabular benchmarks at $\epsilon = 0.1$. Columns show test accuracy, mean explanation size $\pm$ standard deviation (% of features), and mean runtime $\pm$ standard deviation.

Dataset	Acc.	Size mean $\pm$ sd (%) (Ours / greedy / sens.)	Time mean $\pm$ sd (s) (Ours / greedy / sens.)
Adult	0.8647	**19.5**$\pm$16.0 / 25.7$\pm$18.3 / 23.7$\pm$18.8	**866.652**$\pm$1397.304 / 1070.071$\pm$994.654 / 1090.491$\pm$951.636
Bike Sharing	0.9163	**23.5**$\pm$13.7 / 29.1$\pm$18.1 / 32.7$\pm$17.0	**242.984**$\pm$580.558 / 582.489$\pm$1059.078 / 2215.734$\pm$1224.846
Breast Cancer	0.9825	**1.6**$\pm$7.7 / 2.4$\pm$11.0 / 1.7$\pm$8.0	**46.244**$\pm$78.272 / 171.785$\pm$102.700 / 166.080$\pm$97.569
California Housing	0.8777	**15.7**$\pm$18.5 / 16.9$\pm$19.9 / 15.8$\pm$18.6	49.813$\pm$201.815 / **9.818**$\pm$2.837 / 9.802$\pm$2.841
COMPAS	0.6799	**19.7**$\pm$18.5 / 31.5$\pm$20.1 / 23.7$\pm$18.5	**40.424**$\pm$33.006 / 1334.135$\pm$1050.061 / 167.761$\pm$260.991
Diabetes	0.7273	**8.8**$\pm$14.2 / 11.6$\pm$18.7 / 9.7$\pm$15.3	68.908$\pm$63.193 / 78.631$\pm$58.277 / **59.267**$\pm$40.367
German Credit	0.7250	**31.3**$\pm$16.2 / 66.3$\pm$8.2 / 54.8$\pm$13.6	**215.777**$\pm$258.273 / 3063.644$\pm$822.614 / 1619.714$\pm$1295.282
Heart Disease	0.8525	**18.9**$\pm$10.2 / 24.5$\pm$11.8 / 24.9$\pm$12.7	**182.618**$\pm$453.838 / 920.890$\pm$762.224 / 1444.970$\pm$890.476
HELOC	0.7405	**21.1**$\pm$22.0 / 55.2$\pm$14.0 / 62.3$\pm$12.3	**710.099**$\pm$866.431 / 2991.469$\pm$1223.455 / 3216.143$\pm$1016.580

Appendix H additionally compares both settings against exact cardinally-minimal baselines (brute-force search and a minimum-hitting-set algorithm) and against VeriX+, a subset-minimal QuickXplain-based baseline, validates the sparsity assumption of Section 6 on interactions automatically detected from data via SIAN [48], and reports results under the full (unrestricted) admissible input space rather than an ϵ-ball.

7.3 Increasing the Interaction Order of the Any-Order NAM

We use the tabular Bike Sharing benchmark as a case study for demonstrating how our method scales with the interaction order of sparse Any-Order NAMs. Specifically, we train three sparse models of orders 2, 3, and 4, using sparsity parameters $s = 4, 5, 6$, respectively, and run the same experiment as in Subsection 7.2. Accuracy improves monotonically with order, from 0.9163 at order 2 to 0.9180 at order 3 and 0.9295 at order 4. Figure 2 shows the resulting distribution of certified explanation sizes at each order; our method solves every instance at orders 2 and 3, though convergence becomes substantially harder at order 4 as the largest merged components grow (Appendix H reports exact completion rates).

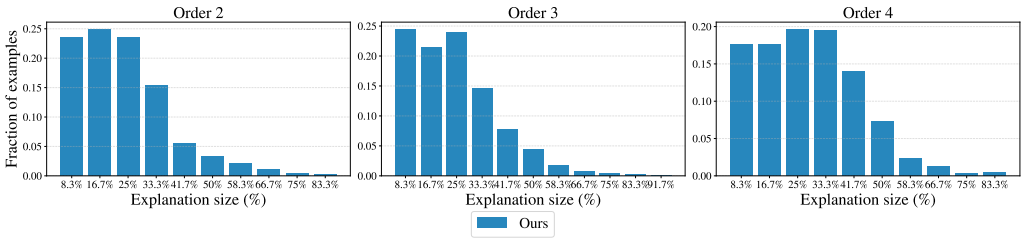

Figure 2: Higher-order results for sparse Any-Order NAMs for the Bike Sharing dataset. Explanation-size distributions for orders 2–4, comparing our method with lexicographic greedy and sensitivity-guided greedy search.

8 Related Work

Formal XAI. Our work relates to *formal XAI* [81], which seeks explanations with mathematical guarantees. Prior work on provably minimal sufficient explanations has mainly focused on restricted model classes, including decision trees [58], tree ensembles [10, 11, 30, 62, 63], linear models [13, 83, 98], and monotonic models [77, 83]. Closer to our setting are methods for neural networks [18, 31, 64, 74, 110, 111]; however, for general networks, computing cardinally-minimal explanations is computationally hard [13], so existing approaches typically rely on relaxed or suboptimal variants [18, 64, 110].

For additive models, [20] studied the complexity of several provable explanation notions, including sufficiency, and [21] proposed a verification-based method for computing cardinally-minimal sufficient explanations for NAMs. However, these approaches exclude feature interactions entirely. In contrast, we are the first to study provable explanations for additive neural models with interactions, where expressivity increases substantially and certification becomes significantly more challenging.

Neural Additive Models (NAMs) and feature interactions. NAMs extend Generalized Additive Models (GAMs) [57, 89], a classic interpretable yet expressive model family [28, 37, 39, 76, 119], by replacing each additive component with a neural network, thereby improving expressivity while preserving interpretability. This architecture was first proposed by [2], with many subsequent extensions in recent years [29, 65, 90, 101, 105, 112]. In particular, higher-order feature interactions are critical for NAM performance [38, 48, 90], and have generally received significant attention within the explainability community in recent years [48, 49, 52, 114].

9 Conclusion

We studied cardinally-minimal provably sufficient explanations for Any-Order NAMs, showing that feature interactions play a decisive role in determining when such explanations can be computed efficiently. On the negative side, we prove that even constant-order interactions recover the Σ_2^P-hardness of general neural networks, revealing a sharp computational gap between purely additive NAMs and models with interactions. On the positive side, we showed that this barrier can be overcome under additional structure: either by using a stronger interaction-aware notion of sufficiency, or by exploiting sparsity in the interaction graph. Together, these results significantly expand the class of neural networks for which certified explanations can be computed, while clarifying the structural conditions that make this possible. More broadly, our work highlights feature interactions as an important potential axis in the theory and practice of explainability with provable guarantees.

Acknowledgements and Disclosures

The work of Refaeli, Elboher, Katz and Bassan was supported by the European Union (ERC, VeriDeL, 101112713). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

References

[1] F. Adolfi, M. Vilas, and T. Wareham. The Computational Complexity of Circuit Discovery for Inner Interpretability. In *Proc. 13th Int. Conf. on Learning Representations (ICLR)*, 2025.

[2] R. Agarwal, L. Melnick, N. Frosst, X. Zhang, B. Lengerich, R. Caruana, and G. E. Hinton. Neural Additive Models: Interpretable Machine Learning with Neural Nets. In *Proc. 35th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 4699–4711, 2021.

[3] D. Alvarez-Melis and T. S. Jaakkola. Towards Robust Interpretability with Self-Explaining Neural Networks. In *Proc. 32nd Conf. on Neural Information Processing Systems (NeurIPS)*, pages 7775–7784, 2018.

[4] G. Amir, S. Bassan, and G. Katz. Hard to Explain: On the Computational Hardness of In-Distribution Model Interpretation. In *Proc. 27th European Conf. on Artificial Intelligence (ECAI)*, pages 818–825, 2024.

[5] A. Anani, T. Lorenz, M. Fritz, and B. Schiele. Pixel-Level Certified Explanations via Randomized Smoothing. In *Proc. 42nd Int. Conf. on Machine Learning (ICML)*, pages 1505–1533, 2025.

[6] M. Arenas, D. Baez, P. Barceló, J. Pérez, and B. Subercaseaux. Foundations of Symbolic Languages for Model Interpretability. *Proc. 35th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 11690–11701, 2021.

[7] M. Arenas, P. Barceló, M. Romero Orth, and B. Subercaseaux. On Computing Probabilistic Explanations for Decision Trees. In *Proc. 36th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 28695–28707, 2022.

[8] M. Arenas, P. Barceló, L. Bertossi, and M. Monet. On the Complexity of SHAP-Score-Based Explanations: Tractability via Knowledge Compilation and Non-Approximability Results. *Journal of Machine Learning Research*, pages 1–58, 2023.

[9] G. Audemard, S. Bellart, L. Bounia, F. Koriche, J. Lagniez, and P. Marquis. On the Computational Intelligibility of Boolean Classifiers. In *Proc. 18th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, pages 74–86, 2021.

[10] G. Audemard, S. Bellart, L. Bounia, F. Koriche, J.-M. Lagniez, and P. Marquis. On Preferred Abductive Explanations for Decision Trees and Random Forests. In *Proc. 31st Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 643–650, 2022.

[11] G. Audemard, J.-M. Lagniez, P. Marquis, and N. Szczepanski. Computing Abductive Explanations for Boosted Trees. In *Proc. 26th Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, pages 4699–4711, 2023.

[12] S. Azzolin, S. Malhotra, A. Passerini, and S. Teso. Beyond Topological Self-Explainable GNNs: A Formal Explainability Perspective. In *Proc. 42nd Int. Conf. on Machine Learning (ICML)*, pages 2045–2080, 2025.

[13] P. Barceló, M. Monet, J. Pérez, and B. Subercaseaux. Model Interpretability through the Lens of Computational Complexity. In *Proc. 34th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 15487–15498, 2020.

[14] S. Bassan and G. Katz. Towards Formal XAI: Formally Approximate Minimal Explanations of Neural Networks. In *Proc. 29th Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, pages 187–207, 2023.

[15] S. Bassan, G. Amir, D. Corsi, I. Refaeli, and G. Katz. Formally Explaining Neural Networks within Reactive Systems. In *Proc. 23rd Int. Conf. on Formal Methods in Computer-Aided Design (FMCAD)*, pages 10–22, 2023.

[16] S. Bassan, G. Amir, and G. Katz. Local vs. Global Interpretability: A Computational Complexity Perspective. In *Proc. 41st Int. Conf. on Machine Learning (ICML)*, pages 3133–3167, 2024.

[17] S. Bassan, G. Amir, M. Zehavi, and G. Katz. What Makes an Ensemble (Un) Interpretable? In *Proc. 42nd Int. Conf. on Machine Learning (ICML)*, pages 3142–3201, 2025.

[18] S. Bassan, Y. Y. Elboher, T. Ladner, M. Althoff, and G. Katz. Explaining, Fast and Slow: Abstraction and Refinement of Provable Explanations. In *Proc. 42nd Int. Conf. on Machine Learning (ICML)*, 2025.

[19] S. Bassan, R. Eliav, and S. Gur. Explain Yourself, Briefly! Self-Explaining Neural Networks with Concise Sufficient Reasons. In *Proc. 13th Int. Conf. on Learning Representations (ICLR)*, 2025.

[20] S. Bassan, M. Moshkovitz, and G. Katz. Additive Models Explained: A Computational Complexity Approach. In *Proc. 39th Conf. on Neural Information Processing Systems (NeurIPS)*, 2025.

[21] S. Bassan, Y. Y. Elboher, T. Ladner, V. Şahin, J. Kretinsky, M. Althoff, and G. Katz. Provably Explaining Neural Additive Models. In *Proc. 14th Int. Conf. on Learning Representations (ICLR)*, 2026.

[22] S. Bassan, S. Gur, S. Zeltyn, K. Mavrogiorgos, R. Eliav, and D. Kyriazis. Self-Explaining Neural Networks for Business Process Monitoring. In *Proc. 22nd Int. Conf. on Smart Business Technologies (ICSBT)*, Communications in Computer and Information Science, pages 72–89. Springer, 2026.

[23] S. Bassan, X. Huang, and G. Katz. Unifying Formal Explanations: A Complexity-Theoretic Perspective. In *Proc. 14th Int. Conf. on Learning Representations (ICLR)*, 2026.

[24] M. Bechler-Speicher, A. Globerson, and R. Gilad-Bachrach. The Intelligible and Effective Graph Neural Additive Network. *Proc. 38th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 90552–90578, 2024.

[25] G. Blanc, J. Lange, and L.-Y. Tan. Provably Efficient, Succinct, and Precise Explanations. In *Proc. 35th Conf. on Neural Information Processing Systems (NeurIPS)*, 2021.

[26] G. Blanc, C. Koch, J. Lange, and L.-Y. Tan. A Query-Optimal Algorithm for Finding Counterfactuals. In *Proc. 39th Int. Conf. on Machine Learning (ICML)*, pages 2075–2090, 2022.

[27] D. Boetius, S. Bassan, G. Katz, S. Leue, and T. Sutter. Verified SHAP: Provable Bounds for Exact Shapley Values of Neural Networks. In *Proc. 43rd Int. Conf. on Machine Learning (ICML)*, pages 8760–8793, 2026.

[28] S. Bordt and U. von Luxburg. From Shapley Values to Generalized Additive Models and Back. In *Proc. 26th Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, pages 709–745, 2023.

[29] K. Bouchiat, A. Immer, H. Yèche, G. Ratsch, and V. Fortuin. Improving Neural Additive Models with Bayesian Principles. In *Proc. 41st Int. Conf. on Machine Learning (ICML)*, pages 4416–4443, 2024.

[30] R. Boumazouza, F. Cheikh-Alili, B. Mazure, and K. Tabia. ASTERYX: A model-Agnostic SaT-basEd appRoach for sYmbolic and score-based eXplanations. In *Proc. 30th ACM Int. Conf. on Information & Knowledge Management (CIKM)*, pages 120–129, 2021.

[31] R. Boumazouza, R. Elsaleh, M. Ducoffe, S. Bassan, and G. Katz. FAME: Formal Abstract Minimal Explanation for Neural Networks. In *Proc. 14th Int. Conf. on Learning Representations (ICLR)*, 2026.

[32] L. Bounia. Enhancing the Intelligibility of Boolean Decision Trees with Concise and Reliable Probabilistic Explanations. In *Proc. 20th Int. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU)*, pages 205–218, 2024.

[33] L. Bounia. Using Submodular Optimization to Approximate Minimum-Size Abductive Path Explanations for Tree-Based Models. In *Proc. 41st Conf. on Uncertainty in Artificial Intelligence (UAI)*, pages 388–397, 2025.

[34] L. Bounia and F. Koriche. Approximating Probabilistic Explanations via Supermodular Minimization. In *Proc. 39th Conf. on Uncertainty in Artificial Intelligence (UAI)*, pages 216–225, 2023.

[35] C. Brix, S. Bak, T. T. Johnson, and H. Wu. The Fifth International Verification of Neural Networks Competition (VNN-COMP 2024): Summary and Results. *arXiv preprint arXiv:2412.19985*, 2024.

[36] B. Carter, J. Mueller, S. Jain, and D. Gifford. What Made You Do This? Understanding Black-Box Decisions with Sufficient Input Subsets. In *Proc. 22nd Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, pages 567–576, 2019.

[37] R. Caruana, Y. Lou, J. Gehrke, P. Koch, M. Sturm, and N. Elhadad. Intelligible Models for Healthcare: Predicting Pneumonia Risk and Hospital 30-Day Readmission. In *Proc. 21st ACM Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD)*, pages 1721–1730, 2015.

[38] C.-H. Chang, R. Caruana, and A. Goldenberg. NODE-GAM: Neural Generalized Additive Model for Interpretable Deep Learning. In *Proc. 10th Int. Conf. on Learning Representations (ICLR)*, 2022.

[39] H. Chen, Y. Wang, F. Zheng, C. Deng, and H. Huang. Sparse Modal Additive Model. *IEEE Transactions on Neural Networks and Learning Systems*, pages 2373–2387, 2021.

[40] H.-M. Chiu, H. Chen, H. Zhang, and R. Y. Zhang. SDP-CROWN: Efficient Bound Propagation for Neural Network Verification with Tightness of Semidefinite Programming. In *Proc. 42nd Int. Conf. on Machine Learning (ICML)*, 2025.

[41] A. Darwiche. Logic for Explainable AI. In *Proc. 38th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, pages 1–11, 2023.

[42] A. Darwiche and A. Hirth. On the reasons behind decisions. *arXiv preprint arXiv:2002.09284*, 2020.

[43] A. Darwiche and C. Ji. On the Computation of Necessary and Sufficient Explanations. In *Proc. 36th AAAI Conf. on Artificial Intelligence*, pages 5582–5591, 2022.

[44] A. De Palma, G. Dolcetti, and C. Urban. Faster Verified Explanations for Neural Networks. In *Proc. 40th European Conf. on Object-Oriented Programming (ECOOP)*, LIPIcs, pages 3:1–3:32, 2026.

[45] A. Dubey, F. Radenovic, and D. Mahajan. Scalable Interpretability via Polynomials. In *Proc. 36th Conf. on Neural Information Processing Systems (NeurIPS)*, 2022.

[46] R. Ehlers. Formal Verification of Piece-Wise Linear Feed-Forward Neural Networks. In *Proc. 15th Int. Symposium on Automated Technology for Verification and Analysis (ATVA)*, pages 269–286, 2017.

[47] O. El Harzli, B. Cuenca Grau, and I. Horrocks. Cardinality-Minimal Explanations for Monotonic Neural Networks. In *Proc. 32nd Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 3677–3685, 2023.

[48] J. Enouen and Y. Liu. Sparse Interaction Additive Networks via Feature Interaction Detection and Sparse Selection. In *Proc. 36th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 13908–13920, 2022.

[49] J. Enouen and Y. Liu. InstaSHAP: Interpretable Additive Models Explain Shapley Values Instantly. In *Proc. 13th Int. Conf. on Learning Representations (ICLR)*, 2025.

[50] T. Fel, M. Ducoffe, D. Vigouroux, R. Cadène, M. Capelle, C. Nicodème, and T. Serre. Don't Lie to Me! Robust and Efficient Explainability with Verified Perturbation Analysis. In *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 16153–16163, 2023.

[51] C. Ferrari, M. N. Müller, N. Jovanović, and M. Vechev. Complete Verification via Multi-Neuron Relaxation Guided Branch-and-Bound. In *Proc. 10th Int. Conf. on Learning Representations (ICLR)*, 2022.

[52] F. Fumagalli, M. Muschalik, P. Kolpaczki, E. Hüllermeier, and B. Hammer. SHAP-IQ: Unified Approximation of Any-Order Shapley Interactions. *Proc. 37th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 11515–11551, 2023.

[53] T. Gehr, M. Mirman, D. Drachsler-Cohen, P. Tsankov, S. Chaudhuri, and M. Vechev. AI2: Safety and Robustness Certification of Neural Networks with Abstract Interpretation. In *Proc. 39th IEEE Symposium on Security and Privacy (S&P)*, pages 3–18, 2018.

[54] M. Grabisch and M. Roubens. An Axiomatic Approach to the Concept of Interaction among Players in Cooperative Games. *International Journal of Game Theory*, pages 547–565, 1999.

[55] Gurobi Optimization, LLC. *Gurobi Optimizer Reference Manual*, 2026. URL `https://www.gurobi.com`.

[56] I. Hadad, G. Katz, and S. Bassan. Formal Mechanistic Interpretability: Automated Circuit Discovery with Provable Guarantees. In *Proc. 14th Int. Conf. on Learning Representations (ICLR)*, 2026.

[57] T. Hastie and R. Tibshirani. Generalized Additive Models. *Statistical Science*, pages 297–310, 1986.

[58] X. Huang, Y. Izza, A. Ignatiev, and J. Marques-Silva. On Efficiently Explaining Graph-Based Classifiers. In *Proc. 18th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, 2021.

[59] S. Ibrahim, G. Afriat, K. Behdin, and R. Mazumder. GRAND-SLAMIN' Interpretable Additive Modeling with Structural Constraints. In *Proc. 37th Conf. on Neural Information Processing Systems (NeurIPS)*, 2023.

[60] A. Ignatiev, A. Morgado, and J. Marques-Silva. RC2: An Efficient MaxSAT Solver. *Journal on Satisfiability, Boolean Modeling and Computation*, pages 53–64, 2019.

[61] A. Ignatiev, N. Narodytska, and J. Marques-Silva. Abduction-Based Explanations for Machine Learning Models. In *Proc. AAAI Conf. on Artificial Intelligence*, pages 1511–1519, 2019.

[62] A. Ignatiev, Y. Izza, P. J. Stuckey, and J. Marques-Silva. Using MaxSAT for Efficient Explanations of Tree Ensembles. In *Proc. 36th AAAI Conf. on Artificial Intelligence*, pages 3776–3785, 2022.

[63] Y. Izza and J. Marques-Silva. On explaining random forests with sat. *arXiv preprint arXiv:2105.10278*, 2021.

[64] Y. Izza, X. Huang, A. Morgado, J. Planes, A. Ignatiev, and J. Marques-Silva. Distance-Restricted Explanations: Theoretical Underpinnings & Efficient Implementation. In *Proc. 21st Int. Conf. on Principles of Knowledge Representation and Reasoning (KR)*, pages 475–486, 2024.

[65] Y. Jiao, C. J. Zdanski, J. S. Kimbell, A. Prince, C. Worden, S. Kirse, C. Rutter, B. Shields, W. Dunn, J. Mahmud, et al. NAISR: A 3D Neural Additive Model for Interpretable Shape Representation. In *Proc. 12th Int. Conf. on Learning Representations (ICLR)*, 2024.

[66] H. Jin, A. Xue, W. You, S. Goel, and E. Wong. Probabilistic Stability Guarantees for Feature Attributions. In *Proc. 39th Conf. on Neural Information Processing Systems (NeurIPS)*, 2025.

[67] U. Junker. QuickXplain: Preferred Explanations and Relaxations for Over-Constrained Problems. In *Proc. 19th National Conf. on Artificial Intelligence (AAAI)*, pages 167–172, 2004.

[68] G. Katz, C. Barrett, D. L. Dill, K. Julian, and M. J. Kochenderfer. Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks. In *Proc. 29th Int. Conf. on Computer Aided Verification (CAV)*, pages 97–117, 2017.

[69] G. Katz, D. A. Huang, D. Ibeling, K. Julian, C. Lazarus, R. Lim, P. Shah, S. Thakoor, H. Wu, A. Zeljić, D. L. Dill, M. J. Kochenderfer, and C. Barrett. The Marabou Framework for Verification and Analysis of Deep Neural Networks. In *Proc. 31st Int. Conf. on Computer Aided Verification (CAV)*, pages 443–452, 2019.

[70] K. Kaulen, T. Ladner, S. Bak, C. Brix, H. Duong, T. Flinkow, T. T. Johnson, L. Koller, E. Manino, T. H. Nguyen, and H. Wu. The 6th International Verification of Neural Networks Competition (VNN-COMP 2025): Summary and Results. *arXiv preprint arXiv:2512.19007*, 2025.

[71] M. Kim, H.-S. Choi, and J. Kim. Higher-Order Neural Additive Models: An Interpretable Machine Learning Model with Feature Interactions. In *Proc. 25th IEEE Int. Conf. on Data Mining (ICDM)*, pages 1310–1319, 2025.

[72] Y. K. Kim, J. M. Di Martino, and G. Sapiro. Generalizing Neural Additive Models via Statistical Multimodal Analysis. *Transactions on Machine Learning Research (TMLR)*, 2024.

[73] S. Kotha, C. Brix, J. Z. Kolter, K. Dvijotham, and H. Zhang. Provably Bounding Neural Network Preimages. In *Proc. 37th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 80270–80290, 2023.

[74] E. La Malfa, R. Michelmore, A. Zbrzezny, N. Paoletti, and M. Kwiatkowska. On Guaranteed Optimal Robust Explanations for NLP Models. In *Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 2658–2665, 2021.

[75] F. Labbaf, T. Kolárik, M. Blicha, G. Fedyukovich, M. Wand, and N. Sharygina. Space Explanations of Neural Network Classification. In *Proc. 37th Int. Conf. on Computer Aided Verification (CAV)*, pages 287–303, 2025.

[76] J. Liu, C. Zhong, M. Seltzer, and C. Rudin. Fast Sparse Classification for Generalized Linear and Additive Models. In *Proc. 25th Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, pages 9304–9333, 2022.

[77] X. Liu, X. Han, N. Zhang, and Q. Liu. Certified Monotonic Neural Networks. *Proc. 34th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 15427–15438, 2020.

[78] Y. Lou, R. Caruana, J. Gehrke, and G. Hooker. Accurate Intelligible Models with Pairwise Interactions. In *Proc. 19th ACM Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD)*, pages 623–631, 2013.

[79] S. M. Lundberg and S.-I. Lee. A Unified Approach to Interpreting Model Predictions. In *Proc. 31st Conf. on Neural Information Processing Systems (NeurIPS)*, 2017.

[80] J. Marques-Silva. Logic-Based Explainability in Machine Learning. In *Reasoning Web. Causality, Explanations and Declarative Knowledge: 18th Int. Summer School 2022, Tutorial Lectures*, Lecture Notes in Computer Science, pages 24–104. Springer, 2023.

[81] J. Marques-Silva and A. Ignatiev. Delivering Trustworthy AI through Formal XAI. In *Proc. 36th AAAI Conf. on Artificial Intelligence*, pages 12342–12350, 2022.

[82] J. Marques-Silva and A. Ignatiev. No Silver Bullet: Interpretable ML Models Must Be Explained. *Frontiers in Artificial Intelligence*, page 1128212, 2023.

[83] J. Marques-Silva, T. Gerspacher, M. Cooper, A. Ignatiev, and N. Narodytska. Explaining Naive Bayes and Other Linear Classifiers with Polynomial Time and Delay. In *Proc. 34th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 20590–20600, 2020.

[84] J. Marques-Silva, T. Gerspacher, M. Cooper, A. Ignatiev, and N. Narodytska. Explanations for Monotonic Classifiers. In *Proc. 38th Int. Conf. on Machine Learning (ICML)*, pages 7469–7479, 2021.

[85] R. Marzouk and C. de la Higuera. On the Tractability of SHAP Explanations under Markovian Distributions. In *Proc. 41st Int. Conf. on Machine Learning (ICML)*, pages 34961–34986, 2024.

[86] R. Marzouk, S. Bassan, and G. Katz. SHAP Meets Tensor Networks: Provably Tractable Explanations with Parallelism. In *Proc. 39th Conf. on Neural Information Processing Systems (NeurIPS)*, 2025.

[87] R. Marzouk, S. Bassan, G. Katz, and C. de la Higuera. On the Computational Tractability of the (Many) Shapley Values. In *Proc. 28th Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, pages 3691–3699, 2025.

[88] M. N. Müller, G. Makarchuk, G. Singh, M. Püschel, and M. Vechev. PRIMA: General and Precise Neural Network Certification via Scalable Convex Hull Approximations. *Proc. ACM Program. Lang. (POPL)*, pages 43:1–43:33, 2022.

[89] J. A. Nelder and R. W. Wedderburn. Generalized Linear Models. *Journal of the Royal Statistical Society Series A: Statistics in Society*, pages 370–384, 1972.

[90] F. Radenovic, A. Dubey, and D. Mahajan. Neural Basis Models for Interpretability. In *Proc. 36th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 8414–8426, 2022.

[91] M. T. Ribeiro, S. Singh, and C. Guestrin. "Why Should I Trust You?" Explaining the Predictions of Any Classifier. In *Proc. 22nd ACM Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD)*, pages 1135–1144, 2016.

[92] M. T. Ribeiro, S. Singh, and C. Guestrin. Anchors: High-Precision Model-Agnostic Explanations. In *Proc. 32nd AAAI Conf. on Artificial Intelligence*, 2018.

[93] M. Sälzer and M. Lange. Reachability Is NP-Complete Even for the Simplest Neural Networks. In *Proc. 15th Int. Conf. on Reachability Problems (RP)*, pages 149–164, 2021.

[94] Z. Shi, Q. Jin, Z. Kolter, S. Jana, C.-J. Hsieh, and H. Zhang. Neural Network Verification with Branch-and-Bound for General Nonlinearities. In *Proc. 31st Int. Conf. on Tools and Algorithms for the Construction and Analysis of Systems (TACAS)*, pages 315–335, 2025.

[95] A. Shih, A. Choi, and A. Darwiche. A Symbolic Approach to Explaining Bayesian Network Classifiers. In *Proc. 27th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 5103–5111, 2018.

[96] G. Singh, T. Gehr, M. Püschel, and M. Vechev. An Abstract Domain for Certifying Neural Networks. *Proc. ACM Program. Lang. (POPL)*, pages 41:1–41:30, 2019.

[97] J. Soria, Z. Chihani, J. Girard-Satabin, A. Grastien, R. Xu-Darme, and D. Cancila. Formal Abductive Latent Explanations for Prototype-Based Networks. In *Proc. 40th AAAI Conf. on Artificial Intelligence*, pages 25590–25598, 2026.

[98] B. Subercaseaux, M. Arenas, and K. Meel. Probabilistic Explanations for Linear Models. In *Proc. 39th AAAI Conference on Artificial Intelligence*, pages 20655–20662, 2025.

[99] M. Sundararajan, A. Taly, and Q. Yan. Axiomatic Attribution for Deep Networks. In *Proc. 34th Int. Conf. on Machine Learning (ICML)*, pages 3319–3328, 2017.

[100] M. Sundararajan, K. Dhamdhere, and A. Agarwal. The Shapley Taylor Interaction Index. In *Proc. 37th Int. Conf. on Machine Learning (ICML)*, pages 9259–9268, 2020.

[101] A. F. Thielmann, R.-M. Kruse, T. Kneib, and B. Säfken. Neural Additive Models for Location Scale and Shape: A Framework for Interpretable Neural Regression Beyond the Mean. In *Proc. 27th Int. Conf. on Artificial Intelligence and Statistics (AISTATS)*, pages 1783–1791, 2024.

[102] V. Tjeng, K. Y. Xiao, and R. Tedrake. Evaluating Robustness of Neural Networks with Mixed Integer Programming. In *Proc. 7th Int. Conf. on Learning Representations (ICLR)*, 2019.

[103] C.-P. Tsai, C.-K. Yeh, and P. Ravikumar. Faith-Shap: The Faithful Shapley Interaction Index. *Journal of Machine Learning Research*, pages 1–42, 2023.

[104] M. Tsang, D. Cheng, and Y. Liu. Detecting Statistical Interactions from Neural Network Weights. In *Proc. 6th Int. Conf. on Learning Representations (ICLR)*, 2018.

[105] N. Upadhya and E. Cohen. NeurCAM: Interpretable Neural Clustering via Additive Models. In *Proc. 27th European Conf. on Artificial Intelligence (ECAI)*, 2024.

[106] G. Van den Broeck, A. Lykov, M. Schleich, and D. Suciu. On the Tractability of SHAP Explanations. In *Proc. 35th AAAI Conf. on Artificial Intelligence*, pages 6505–6513, 2021.

[107] H. D. Wang, D. Manzanas Lopez, P. K. Robinette, I. Oguz, T. T. Johnson, and M. Ma. Towards Verified and Targeted Explanations through Formal Methods. *Journal of Artificial Intelligence Research*, 2026.

[108] S. Wang, H. Zhang, K. Xu, X. Lin, S. Jana, C.-J. Hsieh, and J. Z. Kolter. Beta-Crown: Efficient Bound Propagation with Per-Neuron Split Constraints for Neural Network Robustness Verification. In *Proc. 35th Conf. on Advances in Neural Information Processing Systems (NeurIPS)*, pages 29909–29921, 2021.

[109] H. Wu, O. Isac, A. Zeljić, T. Tagomori, M. Daggitt, W. Kokke, I. Refaeli, G. Amir, K. Julian, S. Bassan, et al. Marabou 2.0: A Versatile Formal Analyzer of Neural Networks. In *Proc. 36th Int. Conf. on Computer Aided Verification (CAV)*, pages 249–264, 2024.

[110] M. Wu, H. Wu, and C. Barrett. Verix: Towards Verified Explainability of Deep Neural Networks. *Proc. 37th Int. Conf. on Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

[111] M. Wu, X. Li, H. Wu, and C. Barrett. Efficiently Computing Compact Formal Explanations. In *Proc. 40th AAAI Conf. on Artificial Intelligence*, pages 35857–35866, 2026.

[112] S. Xu, Z. Bu, P. Chaudhari, and I. J. Barnett. Sparse Neural Additive Model: Interpretable Deep Learning with Feature Selection via Group Sparsity. In *Proc. 27th Joint European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD)*, pages 343–359, 2023.

[113] A. Xue, R. Alur, and E. Wong. Stability Guarantees for Feature Attributions with Multiplicative Smoothing. In *Proc. 37th Conf. on Neural Information Processing Systems (NeurIPS)*, 2023.

[114] Z. Yang, A. Zhang, and A. Sudjianto. GAMI-Net: An Explainable Neural Network Based on Generalized Additive Models with Structured Interactions. *Pattern Recognition*, page 108192, 2021.

[115] W. You, H. Qu, M. Gatti, B. Jain, and E. Wong. Sum-of-Parts: Self-Attributing Neural Networks with End-to-End Learning of Feature Groups. In *Proc. 42nd Int. Conf. on Machine Learning (ICML)*, pages 72747–72785, 2025.

[116] J. Yu, A. Ignatiev, P. J. Stuckey, N. Narodytska, and J. Marques-Silva. Eliminating the Impossible, Whatever Remains Must Be True: On Extracting and Applying Background Knowledge in the Context of Formal Explanations. In *Proc. 37th AAAI Conf. on Artificial Intelligence*, pages 4123–4131, 2023.

[117] H. Zhang, T.-W. Weng, P.-Y. Chen, C.-J. Hsieh, and L. Daniel. Efficient Neural Network Robustness Certification with General Activation Functions. In *Proc. 32nd Conf. on Neural Information Processing Systems (NeurIPS)*, 2018.

[118] W. Zhang, B. Barr, and J. Paisley. Gaussian Process Neural Additive Models. In *Proc. 38th AAAI Conf. on Artificial Intelligence*, pages 16865–16872, 2024.

[119] C. Zhong, Z. Chen, J. Liu, M. Seltzer, and C. Rudin. Exploring and Interacting with the Set of Good Sparse Generalized Additive Models. *Proc. 37th Conf. on Neural Information Processing Systems (NeurIPS)*, pages 56673–56699, 2023.

[120] D. Zhou, C. Brix, G. A. Hanasusanto, and H. Zhang. Scalable Neural Network Verification with Branch-and-Bound Inferred Cutting Planes. In *Proc. 38th Conf. on Neural Information Processing Systems (Neurips)*, 2024.

Appendix

The appendices are organized as follows:

Appendix A discusses the main limitations of our approach, including the dependence on verification or MILP subroutines and the remaining scalability gap for dense high-order interactions.
Appendix B discusses extended related work on neural network verification, formal XAI, and additive models with feature interactions.
Appendix C contains additional background, including the greedy-baseline template and polynomial-hierarchy terminology.
Appendix D extends our formulations to other prediction settings, including general classification and regression.
Appendix E contains the proof of Theorem 4.
Appendix F contains the proofs of all other theorems, propositions, and lemmas in the main text.
Appendix G contains additional experimental details, reproducibility information, compute requirements, and dataset summaries.
Appendix H contains ablation studies, including ϵ sensitivity and order-comparison sweeps.

A Limitations

Like all other methods for computing provably minimal sufficient explanations, our approach relies on neural-network verification queries. Although the scalability of neural-network verification has improved rapidly in recent years [40, 70, 70, 73, 108, 109, 117, 120], existing verifiers still do not scale to state-of-the-art neural networks. Thus, our results should be viewed as expanding the tractable frontier of certified explainability, rather than removing the underlying verification bottleneck. As neural network verification tools continue to improve, the scalability of our approach is expected to improve as well.

Our main contribution is therefore not to eliminate verification cost, but to show that, under suitable interaction structure, the number of required verification queries can be reduced from worst-case exponential to linear. This makes provable explanation search practical for richer models with feature interactions, which achieve substantially higher accuracy than standard NAMs and yield significantly smaller explanations than greedy verifier-backed baselines across standard benchmarks.

This reliance on verification and optimization solvers is not an implementation choice but a consequence of Theorem 4: since computing a provably cardinally-minimum sufficient explanation is already Σ_2^P-hard for constant-order Any-Order NAMs, obtaining such explanations necessarily involves solver calls (assuming $\mathrm{P} \neq \mathrm{NP}$). Two solvers play distinct roles. Provable sufficiency requires certifying that no admissible completion of the free features can flip the prediction; over a continuous region this requires a sound and complete neural-network verifier, whereas for discrete features with enumerable domains the maximal attainable deviation can instead be obtained directly by enumeration, with no verifier needed. Separately, even when every feature is discrete and all deviations are known exactly, a MILP solver (or an equivalent combinatorial search) is still required to extract the cardinally-minimal explanation itself; a heuristic such as our greedy baselines can substitute for either step, but only at the cost of the minimality guarantee.

Finally, our exact guarantees are specific to Any-Order NAMs rather than arbitrary (non-additive) neural architectures; for general DNNs, a minimum-cardinality sufficient explanation is Σ_2^P-hard to compute even to find [13], so an efficient exact extension to that setting is unlikely. One practical bridge is to first train a DNN, detect its significant interactions, fit a sparse Any-Order NAM to exactly those interactions, and then apply our method to the resulting model – the SIAN pipeline of Appendix H.7 – which in our experiments remains close to the original DNN's accuracy while admitting an efficient certified explanation. Training DNNs with objectives that explicitly favor low-order interaction structure, and using automatically discovered interactions as a heuristic for existing (uncertified) explanation methods for general DNNs, are both directions we leave to future work.

B Extended Related Work

Neural network verification. Neural network verification provides provable guarantees about the behavior of a network over a continuous input region, and is NP-complete even for simple ReLU networks [68, 93]. Classical SMT- and MILP-based approaches [46, 68, 69, 102, 109] encode the network exactly, yielding exact guarantees but scaling only to small-to-medium models, whereas abstract interpretation methods [53, 88, 96] propagate layer-wise over-approximations that are fast but incomplete. Linear bound propagation, notably CROWN [117], and the branch-and-bound frameworks built on it [40, 51, 73, 94, 108, 120] now achieve complete verification at much larger scales, as tracked by VNN-COMP [35, 70]. We use such verifiers only as subroutines that bound individual interaction components, combined through a MILP [55], so our approach directly benefits from future verifier improvements.

Formal XAI. Our work belongs to formal XAI [41, 80, 81], which studies explanations with provable guarantees, such as sufficient reasons, also known as abductive or prime-implicant explanations [9, 42, 43, 61, 95, 116], typically computed with tools such as QuickXplain and MaxSAT [60, 67]. Because such explanations are often computationally intractable [1, 4, 6, 7, 13, 16, 17, 23, 25, 26], a pattern that also holds for SHAP [8, 85–87, 106], much of the literature focuses on restricted model classes [82], including decision trees [32–34, 58], tree ensembles [10, 11, 30, 62, 63], linear models [83, 98], monotonic models [47, 77, 84], graph neural networks [12], and additive models [20, 21]. Closer to our setting are methods for general neural networks [14, 15, 61, 64, 74, 110, 111], including several recent works that improve scalability or broaden the form of the explanation [44, 56, 75, 97, 107]. Since computing cardinally-minimal explanations for general networks is Σ_2^P-hard [13], other approaches alleviate this cost through abstractions [18, 31], smoothing [5, 66, 113], or self-explaining architectures [3, 19, 22, 115]. Verification has also been used to equip feature attributions with guarantees, through verified perturbation analysis [50] and provable bounds on exact SHAP values [79] of neural networks [27].

Feature interactions and additive models. Feature interactions have been widely studied in explainability, through Shapley interaction indices [52, 54, 100, 103], their connections to GAMs [28, 49], and interaction detection in trained networks [104]. On the modeling side, GAMs [57] were extended with interactions in GA^{2}Ms [37, 78] and in many neural variants [38, 45, 48, 59, 71, 90, 114], which typically keep the set of interactions sparse or structured to retain interpretability. Our results give this design choice a complexity-theoretic justification: sparse, bounded-overlap interaction structure is one of the settings under which provably cardinally-minimal sufficient explanations become tractable.

C Additional Background

C.1 Greedy baseline template

For convenience, we record the generic verifier-backed greedy template used by the baselines in Section 7. The canonical and sensitivity-based variants differ only in the ordering routine π.

Algorithm 2 Generic greedy remove-and-verify baseline

Input: query $\langle F, x, p, \epsilon \rangle$, ordering routine π
Output: subset-minimal provably sufficient explanation S

$S \leftarrow [n]$
$(i_1, \ldots, i_n) \leftarrow \pi(F, x, p, \epsilon)$
for $t = 1, \ldots, n$ **do**
    $S' \leftarrow S \setminus \{i_t\}$
    if S' is verified sufficient for $\langle F, x, p, \epsilon, S' \rangle$ **then**
        $S \leftarrow S'$
return S

C.2 Polynomial-hierarchy terminology

Definition A.1 (Σ_2^P)**.** *A decision problem L belongs to Σ_2^P (also written Σ_2^P) if there exist a polynomial p and a polynomial-time computable predicate R such that for every instance w,*

$$w \in L \quad \Longleftrightarrow \quad \exists u \in \{0,1\}^{p(|w|)}\ \forall v \in \{0,1\}^{p(|w|)} : R(w,u,v) = 1.$$

Equivalently, $\Sigma_2^P = \mathrm{NP}^{\mathrm{NP}}$, namely the class of decision problems solvable by an NP machine with access to an NP oracle.

Definition A.2 (Σ_2^P-Hardness)**.** *A decision problem is Σ_2^P-hard if every problem in Σ_2^P can be reduced to it by a polynomial-time many-one reduction. Intuitively, such a hardness result means that the problem is at least as hard as the canonical quantified problems at the second level of the polynomial hierarchy. In particular, an exact polynomial-time algorithm for a Σ_2^P-hard problem would imply a collapse of the polynomial hierarchy.*

D Extensions to Additional Architectures

The main text focuses on binary classification, where a single score function F induces the prediction $h_F(x) = \mathbb{I}[F(x) \geq 0]$. As in the strategy extends to multi-class classification and regression after modifying the target margin condition.

D.1 General Classification

Suppose the model has class scores $F^{(1)}, \ldots, F^{(c)}$ and predicts the label

$$t = \arg\max_{j \in [c]} F^{(j)}(x).$$

To preserve this prediction, it suffices to enforce for every admissible completion $z \in \mathcal{C}(x, S)$ and every rival class $j \neq t$ that

$$F^{(t)}(z) - F^{(j)}(z) \geq 0.$$

Equivalently, one may work with pairwise winner-vs.-one margins

$$M_j(x) = F^{(t)}(x) - F^{(j)}(x), \qquad j \neq t,$$

or with the winner-vs.-all margin

$$M_{\text{all}}(x) = F^{(t)}(x) - \max_{j \neq t} F^{(j)}(x).$$

For Any-Order NAMs, each pairwise margin M_j is again an Any-Order NAM after merging the component families of the two logits, so the binary machinery from Sections 5 and 6 applies without conceptual changes. In the strong-sufficiency formulation, one simply replaces the single binary margin constraint by the family of constraints associated with all rivals $j \neq t$; in the sparse setting, the same regrouping argument is then applied to each pairwise margin model. Any subset that satisfies all of these pairwise constraints preserves the original winning class under every admissible completion.

D.2 Regression

For regression, a natural analogue is to fix a tolerance $\delta > 0$ and call S sufficient if every admissible completion keeps the output within δ of the original prediction:

$$|F(z) - F(x)| \leq \delta \qquad \text{for all } z \in \mathcal{C}(x, S).$$

This is equivalent to the pair of one-sided constraints

$$F(z) - F(x) \leq \delta \qquad \text{and} \qquad F(x) - F(z) \leq \delta.$$

Accordingly, in the strongly sufficient setting we compute, for each component $T \in \mathcal{H}$ and free set $U \subseteq T$, both the largest upward and downward deviations,

$$\Delta_T^+(x, U) := \max\{f_T(z_T) - f_T(x_T) : z_T \text{ is feasible and } z_U = x_U\},$$

and

$$\Delta_T^-(x, U) := \max\{f_T(x_T) - f_T(z_T) : z_T \text{ is feasible and } z_U = x_U\}.$$

The MILP then replaces the single classification-margin constraint with

$$\sum_{T\in\mathcal{H}} \Delta_T^+(x, U_T(S)) \le \delta \qquad \text{and} \qquad \sum_{T\in\mathcal{H}} \Delta_T^-(x, U_T(S)) \le \delta.$$

The sparse-setting method extends analogously after regrouping into merged components. Thus, once the desired regression guarantee is expressed through an output tolerance, the same verifier-and-MILP template yields certified minimum-cardinality explanations.

D.3 Contrastive Explanations

A contrastive explanation is a minimal set of features that, if changed, can flip the prediction; it is the dual notion to a sufficient explanation, since a set is contrastive exactly when its complement is *not* sufficient. We ask whether our two tractable settings dualize to yield cardinally-minimal contrastive explanations as well.

Sparse setting (yes). Let $C_1, \dots, C_m$ be the merged components of an s-sparse Any-Order NAM (Definition 11). By Proposition 7 applied to the regrouped model $\widetilde{F}$, a candidate $C \subseteq [n]$ is contrastive iff $\sum_{\ell=1}^m \Delta_{C_\ell}\big(x, C_\ell \cap C\big) > |F(x)|$, the complement of the sufficiency criterion. Since the merged components are disjoint (Lemma 10), each component's worst case is attained on its own coordinates, so the decoupled sum remains exact and any violation of the margin is witnessed by a single real perturbation (not merely a family of locally inconsistent ones). Swapping the free and fixed indicator constraints in the MILP of Section 6 and swapping the deviation bounds in Algorithm 1 yields an analog of Theorem 12 for cardinally-minimal contrastive explanations, in the same linear number of outer refinement rounds.

Strong sufficiency setting (no). The two notions relax in opposite directions. Strong sufficiency is conservative: decoupling the components' completions only enlarges the set of adversaries considered, so a strongly sufficient set remains genuinely sufficient (Proposition 6). Contrastiveness has no such safety margin in the other direction: a decoupled witness to the contrastive analog of (1) is a family of local completions, one per component, that need not agree on the value of any feature they share, and so need not correspond to any single admissible input. A set that is contrastive under this decoupled notion therefore need not be contrastive under the ordinary definition, which is why we do not dualize the strong-sufficiency algorithm of Section 5; checking a candidate there instead requires a direct query on the original (non-decoupled) model asking whether some admissible perturbation of it flips the prediction. A provably minimum-cardinality contrastive algorithm for the strong-sufficiency setting is left to future work.

E Proof of Theorem 4

In this appendix we prove Theorem 4: there exists a fixed constant k such that MSE-k-NAM , restricted to Boolean domains and Any-Order NAMs of order at most k, is Σ_2^P-hard. Thus we work throughout with Boolean domains, $\mathcal{D} = \{0,1\}^n$, and interaction scopes of size at most k. For simplicity, we also fix $p = \infty$ and $\epsilon = 1$; then every Boolean point lies in $\mathcal{B}_{p,\epsilon}(x)$, so $\mathcal{C}_{p,\epsilon}(x, S) = \mathcal{C}(x, S)$. A standard route for extending such Boolean hardness constructions to general continuous domains is described in [93]. Since this restricted family is a special case of Any-Order NAMs, the theorem from the main text follows immediately.

We begin with the logic-side problem from which the reduction is derived. A literal over a variable x_i is either x_i or $\neg x_i$. For an assignment $x \in \{0,1\}^n$, we write $x \models \ell$ when the literal ℓ evaluates to true under x. If $u = \ell_{i_1} \wedge \cdots \wedge \ell_{i_q}$ is a conjunction of literals, then $x \models u$ means that $x \models \ell_{i_j}$ for every $j \in [q]$. A DNF formula is a formula of the form

$$\psi = u_1 \vee \cdots \vee u_m,$$

where each u_j is a conjunction of literals. A set of literals C is an *implicant* of ψ if every total assignment satisfying all literals in C also satisfies ψ.

The reduction uses a restricted constant-width Shortest Implicant Core problem. Fix a constant $d \in \mathbb{N}$. An instance consists of a DNF formula

$$\psi = u_1 \vee \cdots \vee u_m \vee t^\star$$

over variables $x_1, \ldots, x_n$ together with an integer r, such that: (i) every term of ψ has width at most d, and (ii) every variable appearing in ψ also appears in the distinguished term $t^\star$. The question is whether there exists an implicant $C \subseteq t^\star$ of ψ such that $|C| \leq r$. By combining the constant-width SIC hardness route with the DNF-to-DNF purification theorem in [21], one obtains that this restricted problem is Σ_2^P-hard for some fixed constant d. We use that fact as an imported ingredient.

Lemma A.3. *Let $u = \ell_{i_1} \wedge \cdots \wedge \ell_{i_q}$ be a conjunction of literals of width q. Then there exists a component function $g_u : \{0,1\}^{\{i_1,\ldots,i_q\}} \to \{0,1\}$ such that for every $x \in \{0,1\}^n$,*

$$g_u(x_{\{i_1,\ldots,i_q\}}) = 1 \iff x \models u.$$

Proof. Define $g_u(y) = 1$ if and only if the partial assignment y satisfies every literal appearing in u, and set $g_u(y) = 0$ otherwise. By construction, g_u depends only on the variables occurring in u, and it evaluates to 1 exactly on the satisfying assignments of u. □

Lemma A.4. *Let*

$$\psi = u_1 \vee \cdots \vee u_m$$

be a DNF in which every term has width at most k. Then there exists an Any-Order NAM of order at most k with score function G_ψ such that for every $x \in \{0,1\}^n$,

$$h_{G_\psi}(x) = 1 \iff x \models \psi.$$

Proof. For each term u_j, let g_{u_j} be the component from Lemma A.3, and define

$$G_\psi(x) = -\frac{1}{2} + \sum_{j=1}^{m} g_{u_j}(x_{T_j}),$$

where T_j is the variable set appearing in u_j. Every component has arity at most k, so this is an Any-Order NAM of order at most k. If $x \models \psi$, then at least one term is satisfied and hence $\sum_{j=1}^m g_{u_j}(x_{T_j}) \geq 1$, implying $G_\psi(x) \geq \frac{1}{2}$ and therefore $h_{G_\psi}(x) = 1$. If $x \not\models \psi$, then every term is false, so each component contributes 0, yielding $G_\psi(x) = -\frac{1}{2}$ and therefore $h_{G_\psi}(x) = 0$. □

Lemma A.5. *Let*

$$\psi = u_1 \vee \cdots \vee u_m \vee t^\star$$

be a DNF over variables $x_1, \ldots, x_n$ such that every variable appearing in ψ also appears in $t^\star$. Let G_ψ be the score function from Lemma A.4, and let $x^\star \in \{0,1\}^n$ be the assignment satisfying all literals of $t^\star$. For any set of literals $C \subseteq t^\star$, define

$$S_C = \{i \in [n] : \textit{the literal of } t^\star \textit{ over } x_i \textit{ belongs to } C\}.$$

Then

$$C \textit{ is an implicant of } \psi \iff S_C \textit{ is a sufficient explanation for } x^\star \textit{ with respect to } G_\psi.$$

Proof. By Lemma A.4, the classifier h_{G_ψ} computes exactly the Boolean function represented by ψ.

Suppose first that C is an implicant of ψ. Let $z \in \mathcal{C}(x^\star, S_C)$ be any completion consistent with $x^\star$ on S_C. By definition of S_C, the point z satisfies all literals in C. Since C is an implicant, this implies $z \models \psi$, and hence $h_{G_\psi}(z) = 1$. Because $x^\star$ satisfies $t^\star$, it also satisfies ψ, so $h_{G_\psi}(x^\star) = 1$. Therefore $h_{G_\psi}(z) = h_{G_\psi}(x^\star)$ for every $z \in \mathcal{C}(x^\star, S_C)$, and S_C is a sufficient explanation.

Conversely, suppose that S_C is a sufficient explanation for $x^\star$, and let $z \in \{0,1\}^n$ satisfy all literals in C. Because every variable of ψ appears in $t^\star$, satisfying the literals in C is equivalent to agreeing with $x^\star$ on the coordinates in S_C. Thus $z \in \mathcal{C}(x^\star, S_C)$. Since S_C is sufficient, $h_{G_\psi}(z) = h_{G_\psi}(x^\star) = 1$. Using again that h_{G_ψ} computes ψ exactly, we conclude that $z \models \psi$. Thus every total assignment satisfying C also satisfies ψ, so C is an implicant of ψ. □

Proof of Theorem 4. Throughout this proof, we assume that all activations in the network are ReLUs. Fix a constant d for which the restricted SIC problem above is Σ_2^P-hard, and choose any fixed $k \geq d$. We fix the query parameters to $p = \infty$ and $\epsilon = 1$, and reduce from that restricted SIC problem to the restriction of MSE-k-NAM to Boolean-domain Any-Order NAMs of order at most k.

Consider an input instance (ψ, r), where

$$\psi = u_1 \vee \cdots \vee u_m \vee t^\star$$

and every term has width at most d. Construct the order-k Any-Order NAM with score function G_ψ from Lemma A.4, and let $x^\star \in \{0,1\}^n$ be the assignment satisfying all literals of the distinguished term $t^\star$. We claim that

$$\begin{aligned}&(\psi, r) \text{ is a yes-instance of restricted SIC}\\ \iff\ &(G_\psi, x^\star, p, \epsilon, r)\\ &\quad\text{is a yes-instance of MSE-}k\text{-NAM .}\end{aligned}$$

For the forward direction, suppose there exists an implicant $C \subseteq t^\star$ with $|C| \le r$. Let S_C be the corresponding feature set from Lemma A.5. That lemma implies that S_C is a sufficient explanation for $x^\star$ with respect to G_ψ, and clearly $|S_C| = |C| \le r$. Hence $(G_\psi, x^\star, p, \epsilon, r)$ is a yes-instance of MSE-k-NAM .

For the reverse direction, suppose $(G_\psi, x^\star, p, \epsilon, r)$ is a yes-instance of MSE-k-NAM . Then there exists a sufficient explanation $S \subseteq [n]$ with $|S| \le r$. Because every variable of ψ appears in $t^\star$, the set S determines a subset of literals

$$C_S = \{x_i : i \in S,\ x_i^\star = 1\} \cup \{\neg x_i : i \in S,\ x_i^\star = 0\},$$

which is contained in $t^\star$ and satisfies $|C_S| = |S| \le r$. Applying Lemma A.5 in the reverse direction shows that C_S is an implicant of ψ. Thus (ψ, r) is a yes-instance of the restricted SIC problem.

The reduction is polynomial-time: for each DNF term we create one NAM component, and each component has arity at most the fixed constant k. Therefore MSE-k-NAM is Σ_2^P-hard even for Boolean-domain Any-Order NAMs of order at most k, and hence also for unrestricted Any-Order NAMs. □

F Omitted Proofs

For brevity, throughout this appendix we use the signed-score formulation from Sections 5 and 6: the relevant certification condition is that the signed score remains nonnegative on the completion set under consideration. We also abbreviate $U_T(S) = T \cap \bar{S}$ for $T \in \mathcal{H}$ and $W_\ell(S) = C_\ell \cap \bar{S}$ for sparse blocks.

F.1 Proofs for Section 5

Proof of Proposition 6. Assume that $S \subseteq [n]$ is strongly sufficient for the query $\langle F, x, p, \epsilon, S\rangle$, and let $z \in \mathcal{C}_{p,\epsilon}(x, S)$ be any admissible completion consistent with x_S. For each component $T \in \mathcal{H}$, define $y^T = z_T$. Since $z_S = x_S$, we have

$$y^T_{S\cap T} = z_{S\cap T} = x_{S\cap T} \qquad \text{for every } T \in \mathcal{H}.$$

Moreover, because $\|z - x\|_p \le \epsilon$, every subvector also satisfies $\|y^T - x_T\|_p = \|z_T - x_T\|_p \le \epsilon$. Therefore $(y^T)_{T\in\mathcal{H}} \in \mathfrak{C}^{\mathrm{str}}(x, S)$. By strong sufficiency,

$$\mathbb{I}\left[b + \sum_{T\in\mathcal{H}} f_T(y^T) \ge 0\right] = h_F(x).$$

But

$$b + \sum_{T\in\mathcal{H}} f_T(y^T) = b + \sum_{T\in\mathcal{H}} f_T(z_T) = F(z),$$

so $h_F(z) = h_F(x)$. Since $z \in \mathcal{C}_{p,\epsilon}(x, S)$ was arbitrary, S is a sufficient explanation for the query $\langle F, x, p, \epsilon, S\rangle$. □

Proof of Proposition 7. Fix a candidate set $S \subseteq [n]$. For each component $T \in \mathcal{H}$, define

$$m_T(S) = \min\Big\{\sigma(x) f_T(z_T) : z_T \in \mathcal{D}_T,\ z_{T\setminus U_T(S)} = x_{T\setminus U_T(S)},\ \ \|z_T - x_T\|_p \le \epsilon\Big\}.$$

By definition of the component deviation Δ_T (Section 5),

$$\Delta_T\big(x, U_T(S)\big) = \sigma(x) f_T(x_T) - m_T(S).$$

Because strong sufficiency allows each component to choose its completion independently, the set $\mathfrak{C}^{\mathrm{str}}(x, S)$ is a Cartesian product over the components. Therefore, the minimum attainable signed score over all strong completions factorizes as

$$\min_{(y^T)_{T\in\mathcal{H}} \in \mathfrak{C}^{\mathrm{str}}(x,S)} \sigma(x)\Big(b + \sum_{T\in\mathcal{H}} f_T(y^T)\Big) = \sigma(x) b + \sum_{T\in\mathcal{H}} m_T(S).$$

Also,

$$|F(x)| = \sigma(x) b + \sum_{T\in\mathcal{H}} \sigma(x) f_T(x_T).$$

Subtracting the two identities yields

$$\begin{aligned}
|F(x)| - \sum_{T\in\mathcal{H}} \Delta_T\big(x, U_T(S)\big) &= \sigma(x) b + \sum_{T\in\mathcal{H}} \sigma(x) f_T(x_T) - \sum_{T\in\mathcal{H}} \big(\sigma(x) f_T(x_T) - m_T(S)\big) \\
&= \sigma(x) b + \sum_{T\in\mathcal{H}} m_T(S) \\
&= \min_{(y^T)\in\mathfrak{C}^{\mathrm{str}}(x,S)} \sigma(x)\Big(b + \sum_{T\in\mathcal{H}} f_T(y^T)\Big).
\end{aligned}$$

Hence S is strongly sufficient exactly when the minimum signed score over $\mathfrak{C}^{\mathrm{str}}(x, S)$ is nonnegative, which is equivalent to

$$|F(x)| - \sum_{T\in\mathcal{H}} \Delta_T\big(x, U_T(S)\big) \ge 0.$$

Rearranging gives (1). □

F.1.1 Bounding minimal deviations with DNN verification

Computing every exact value $\Delta_T(x, U)$ requires solving a nonlinear optimization problem over $\mathcal{C}^T_{p,\epsilon}(x, U)$. To obtain certified bounds instead, we encode the local component network f_T together with the constraints defining $\mathcal{C}^T_{p,\epsilon}(x, U)$ as a verifier query. Concretely, the coordinates in $T \setminus U$ are fixed to their values in x, the free coordinates in U are constrained by the local perturbation set, and the verifier is asked to certify the relevant extremal value of the component output.

For verifier queries, it is convenient to work with the extremal raw component value

$$e_T(x, U) := \begin{cases} \min_{z_T \in \mathcal{C}^T_{p,\epsilon}(x,U)} f_T(z_T), & \sigma(x) = 1, \\ \max_{z_T \in \mathcal{C}^T_{p,\epsilon}(x,U)} f_T(z_T), & \sigma(x) = -1. \end{cases}$$

Then $\Delta_T(x, U) = \sigma(x) f_T(x_T) - \sigma(x) e_T(x, U)$. Suppose the verifier returns certified bounds $\underline{e}_T(x, U) \le e_T(x, U) \le \overline{e}_T(x, U)$. These induce deviation bounds

$$\underline{\Delta}_T(x, U) = \begin{cases} \max\{0, f_T(x_T) - \overline{e}_T(x, U)\}, & \sigma(x) = 1, \\ \max\{0, \underline{e}_T(x, U) - f_T(x_T)\}, & \sigma(x) = -1, \end{cases}$$

and

$$\overline{\Delta}_T(x, U) = \begin{cases} \max\{0, f_T(x_T) - \underline{e}_T(x, U)\}, & \sigma(x) = 1, \\ \max\{0, \overline{e}_T(x, U) - f_T(x_T)\}, & \sigma(x) = -1. \end{cases}$$

These bounds satisfy

$$\underline{\Delta}_T(x, U) \le \Delta_T(x, U) \le \overline{\Delta}_T(x, U) \qquad \text{for every } T \in \mathcal{H},\ U \subseteq T.$$

Replacing $\Delta_T(x,U)$ in (2) by $\underline{\Delta}_T(x,U)$ gives the *optimistic* MILP. Replacing it with $\overline{\Delta}_T(x,U)$ gives the *pessimistic* MILP.

To tighten these bounds, we bisect the current interval for the extremal value $e_T(x,U)$. Given certified bounds $\underline{e}_T(x,U) \le e_T(x,U) \le \overline{e}_T(x,U)$, let $q_T(x,U) = (\underline{e}_T(x,U) + \overline{e}_T(x,U))/2$. If $\sigma(x) = 1$, we ask the verifier whether the minimum of f_T over $\mathcal{C}^T_{p,\epsilon}(x,U)$ is at least $q_T(x,U)$; if $\sigma(x) = -1$, we ask whether the maximum is at most $q_T(x,U)$. The answer identifies which half-interval still contains $e_T(x,U)$. Replacing $[\underline{e}_T(x,U), \overline{e}_T(x,U)]$ by that surviving half and then reapplying the formulas above yields refined bounds $\underline{\Delta}_T(x,U)$ and $\overline{\Delta}_T(x,U)$.

Lemma A.6. *Let $v^\downarrow$ and $v^\uparrow$ be the optimal values of the optimistic and pessimistic MILPs, respectively, and let v^* be the optimum of the exact MILP. Then $v^\downarrow \le v^* \le v^\uparrow$. Moreover, every optimal pessimistic solution is strongly sufficient, and if $v^\downarrow = v^\uparrow$ then the corresponding pessimistic solution is a cardinally-minimal strongly sufficient explanation.*

Proof of Lemma A.6. Let $\mathcal{F}(\widehat{\Delta})$ denote the feasible set of (2) when the last constraint uses the table $\widehat{\Delta}_T(x,U)$. Since $\underline{\Delta}_T(x,U) \le \Delta_T(x,U) \le \overline{\Delta}_T(x,U)$ for every $T \in \mathcal{H}$ and $U \subseteq T$, replacing the coefficients in the last constraint by larger values can only shrink the feasible region. Therefore,

$$\mathcal{F}(\overline{\Delta}) \subseteq \mathcal{F}(\Delta) \subseteq \mathcal{F}(\underline{\Delta}).$$

Taking the minimum of $\sum_i b_i$ over these nested feasible sets yields

$$v^\downarrow \le v^* \le v^\uparrow.$$

Now let $(b^\uparrow, c^\uparrow)$ be an optimal pessimistic solution and set $S^\uparrow = \{i \in [n] : b_i^\uparrow = 1\}$. Since $(b^\uparrow, c^\uparrow) \in \mathcal{F}(\overline{\Delta}) \subseteq \mathcal{F}(\Delta)$, it is feasible for the exact MILP. The first three constraint families in (2) force the unique active case variable in each component T to be $c_{T,U_T(S^\uparrow)} = 1$, so the last exact constraint becomes

$$\sum_{T \in \mathcal{H}} \Delta_T\big(x, U_T(S^\uparrow)\big) \le |F(x)|.$$

By Proposition 7, this is exactly the condition for $S^\uparrow$ to be strongly sufficient.

Finally, if $v^\downarrow = v^\uparrow$, then the inequality above forces $v^* = v^\uparrow$. Since $(b^\uparrow, c^\uparrow)$ is exact-feasible and has objective value $v^\uparrow$, it is also optimal for the exact MILP. Proposition 8 then implies that $S^\uparrow$ is a cardinally-minimal strongly sufficient explanation. □

F.1.2 Mixed discrete/continuous variant of Algorithm 1

The main text states Algorithm 1 only for continuous domains. When some perturbation-aware local completion sets $\mathcal{C}^T_{p,\epsilon}(x,U)$ are finite, the same optimistic/pessimistic MILP outer loop can exploit exact discrete subproblems by replacing the local bounding step with enumeration whenever $\mathcal{C}^T_{p,\epsilon}(x,U)$ is small enough. The resulting mixed discrete/continuous variant is shown in Algorithm 3.

Algorithm 3 Strong-sufficiency algorithm for mixed discrete/continuous domains

Input: Any-Order NAM F, input x, norm parameter $p \in [1, \infty]$, perturbation radius $\epsilon > 0$, signed label $\sigma(x) = 2h_F(x) - 1$, enumeration threshold B_{enum}

for all $T \in \mathcal{H}$ and $U \subseteq T$ **in parallel do**
  if $U = \emptyset$ **then**
    Set $\underline{\Delta}_T(x, U) \leftarrow 0$ and $\overline{\Delta}_T(x, U) \leftarrow 0$
  else if $\mathcal{C}^T_{p,\epsilon}(x, U)$ is finite and $|\mathcal{C}^T_{p,\epsilon}(x, U)| \leq B_{\text{enum}}$ **then**
    Compute $\delta_T(x, U) \leftarrow \sigma(x) f_T(x_T) - \min_{z_T \in \mathcal{C}^T_{p,\epsilon}(x,U)} \sigma(x) f_T(z_T)$ by enumeration
    Set $\underline{\Delta}_T(x, U) \leftarrow \overline{\Delta}_T(x, U) \leftarrow \delta_T(x, U)$
  else
    Compute sound bounds $\underline{e}_T(x, U) \leq e_T(x, U) \leq \overline{e}_T(x, U)$
    Convert $[\underline{e}_T(x, U), \overline{e}_T(x, U)]$ into $[\underline{\Delta}_T(x, U), \overline{\Delta}_T(x, U)]$
while true **do**
  Solve (2) with $\widehat{\Delta}_T(x, U) = \overline{\Delta}_T(x, U)$ for all (T, U)
  Let $S^{\uparrow} \leftarrow \{i \in [n] : b^{\uparrow}_i = 1\}$ be the pessimistic explanation
  Solve (2) with $\widehat{\Delta}_T(x, U) = \underline{\Delta}_T(x, U)$ for all (T, U)
  Let $S^{\downarrow} \leftarrow \{i \in [n] : b^{\downarrow}_i = 1\}$ be the optimistic explanation
  if $|S^{\uparrow}| = |S^{\downarrow}|$ **then**
    return $S^{\uparrow}$
  for all (T, U) with $\underline{\Delta}_T(x, U) < \overline{\Delta}_T(x, U)$ **in parallel do**
    Let $q_T(x, U) \leftarrow (\underline{e}_T(x, U) + \overline{e}_T(x, U))/2$
    Use one verifier query on $\mathcal{C}^T_{p,\epsilon}(x, U)$ to determine which half-interval still contains $e_T(x, U)$
    Replace $[\underline{e}_T(x, U), \overline{e}_T(x, U)]$ by that surviving half
    Update $[\underline{\Delta}_T(x, U), \overline{\Delta}_T(x, U)]$ accordingly

Whenever a local completion set is enumerated exactly, its deviation entry is fixed from the outset and never refined again. In the fully discrete case, all entries are exact after initialization, so the optimistic and pessimistic MILPs coincide immediately. The same feasible-set nesting argument used in the proof below yields correctness for this mixed-domain variant as well.

Proof of Theorem 9. Throughout this proof, we assume that all activations in the network are ReLUs. We begin by showing that the MILP template (2) exactly encodes strong sufficiency when supplied with the exact deviation table $\Delta_T(x, U)$. Let $\mathcal{F}(\widehat{\Delta})$ denote the feasible set of (2) when the last constraint uses the table $\widehat{\Delta}_T(x, U)$.

Take any feasible pair $(b, c) \in \mathcal{F}(\widehat{\Delta})$ and define $S = \{i \in [n] : b_i = 1\}$. For every component $T \in \mathcal{H}$, the constraint $\sum_{U \subseteq T} c_{T,U} = 1$ selects a unique subset $U^*_T \subseteq T$ with $c_{T,U^*_T} = 1$. The second family of constraints in (2) implies that $b_i = 0$ for every $i \in U^*_T$, while the third family implies that $b_i = 1$ for every $i \in T \setminus U^*_T$. Thus

$$U^*_T = \{i \in T : b_i = 0\} = T \cap \bar{S} = U_T(S).$$

Conversely, for any set $S \subseteq [n]$, setting $b_i = 1$ iff $i \in S$ and choosing exactly one active case variable per component via $c_{T,U_T(S)} = 1$ and $c_{T,U} = 0$ for all $U \neq U_T(S)$ satisfies the first three families of constraints. Therefore, when $\widehat{\Delta} = \Delta$, the last constraint in (2) is precisely (1), and the objective is exactly $|S|$. It follows that the exact MILP computes a minimum-cardinality strongly sufficient explanation.

Next, we use the interval bounds maintained by Algorithm 1. At initialization, and after every refinement step, the bounds satisfy

$$\underline{\Delta}_T(x, U) \leq \Delta_T(x, U) \leq \overline{\Delta}_T(x, U) \qquad \text{for every } T \in \mathcal{H},\ U \subseteq T,$$

because the local bounding routine always returns sound intervals and every refinement step only shrinks such an interval while preserving the true optimum inside it. Since larger coefficients in the last constraint of (2) only make feasibility harder, we obtain the nested feasible regions

$$\mathcal{F}(\overline{\Delta}) \subseteq \mathcal{F}(\Delta) \subseteq \mathcal{F}(\underline{\Delta}).$$

Let

$$v^{\uparrow} = \min\{\textstyle\sum_i b_i : (b, c) \in \mathcal{F}(\overline{\Delta})\}, \qquad v^{\downarrow} = \min\{\textstyle\sum_i b_i : (b, c) \in \mathcal{F}(\underline{\Delta})\},$$

and let v^* be the optimum of the exact MILP $\mathcal{F}(\Delta)$. By the set inclusion above,

$$v^{\downarrow} \leq v^* \leq v^{\uparrow}.$$

Let $(b^{\uparrow}, c^{\uparrow})$ be an optimal pessimistic solution and $S^{\uparrow} = \{i : b_i^{\uparrow} = 1\}$. Since $(b^{\uparrow}, c^{\uparrow}) \in \mathcal{F}(\overline{\Delta}) \subseteq \mathcal{F}(\Delta)$, the set $S^{\uparrow}$ is feasible for the exact MILP and hence satisfies (1). By Proposition 7, $S^{\uparrow}$ is strongly sufficient. Hence every pessimistic iterate is strongly sufficient. Proposition 6 then implies that every such iterate is also sufficient in the ordinary sense.

For optimality, suppose that the optimistic and pessimistic MILPs have the same optimal objective value, i.e., $v^{\downarrow} = v^{\uparrow}$. The inequality $v^{\downarrow} \leq v^* \leq v^{\uparrow}$ then forces $v^* = v^{\uparrow}$. Since $S^{\uparrow}$ is an exact feasible solution with objective value $v^{\uparrow}$, it attains the exact optimum and is therefore a cardinally-minimal strongly sufficient explanation.

For termination, we apply the same interval-halving idea used in the first-order NAM setting [21, App. D.1, Proposition 5], now to the finitely many local deviation subproblems (T, U). For each unresolved pair (T, U), the algorithm maintains an interval for the local extremal value $e_T(x, U)$. Because the subproblem is finitely encoded and the local bounding routine returns rational bounds, this interval has a finite binary precision budget. Each refinement step bisects the current interval and thus removes one bit of uncertainty from that local subproblem. Since Algorithm 1 refines all unresolved pairs in parallel, the number of outer refinement rounds is bounded by the largest precision budget of any one local subproblem, which is linear in the model encoding size. After that many rounds, every remaining interval is exact, so $\underline{\Delta} = \overline{\Delta} = \Delta$ and the two MILPs coincide with the exact MILP. Therefore, Algorithm 1 must terminate.

For the per-round complexity statement, let $r = \max_{T \in \mathcal{H}} |T|$. The algorithm must consider one deviation entry for every nonempty free subset $U \subseteq T$, namely $2^{|T|} - 1$ possibilities per component. Summing over all components gives at most

$$\sum_{T \in \mathcal{H}} (2^{|T|} - 1) \leq |\mathcal{H}|\, 2^r$$

local deviation entries in a batched refinement pass. The MILP contains n binary variables b_i and one binary case variable $c_{T,U}$ for each pair (T, U), for a total of $n + \sum_{T \in \mathcal{H}} 2^{|T|}$ variables. The first constraint family contributes $|\mathcal{H}|$ equalities; the second and third together contribute

$$\sum_{T \in \mathcal{H}} \sum_{U \subseteq T} |T| \leq r \sum_{T \in \mathcal{H}} 2^{|T|}$$

inequalities; and the deviation budget contributes one additional inequality. Hence, for fixed r, each outer round performs two MILP solves and one batched refinement pass, with polynomial-size MILPs and polynomially many local deviation subproblems. Combined with the linear outer-round bound above, this proves the claimed round complexity. □

F.2 Proofs for Section 6

Proof of Lemma 10. By definition, $C_1, \ldots, C_m$ are the connected components of the feature-interaction graph $\mathcal{G}_{\mathcal{H}}$, so they are pairwise disjoint. Hence each input feature belongs to exactly one connected component.

Now fix any interaction scope $T \in \mathcal{H}$. For every pair of features $i, j \in T$, the construction of $\mathcal{G}_{\mathcal{H}}$ places an edge between i and j. Thus the vertices of T form a clique in $\mathcal{G}_{\mathcal{H}}$, and in particular they all lie in the same connected component. Therefore T is contained in a unique merged component C_ℓ. □

Proof of Theorem 12. Let

$$\widetilde{F}(x) = b + \sum_{\ell=1}^{m} G_\ell(x_{C_\ell})$$

be the regrouped model from Section 6. By construction, $\widetilde{F}(x) = F(x)$ for every input x. Moreover, by Lemma 10, every original interaction scope is contained in a unique merged component, and distinct merged components share no interactions.

Fix any candidate explanation $S \subseteq [n]$. Because the merged components are disjoint, any choice of local completions for the components of $\widetilde{F}$ can be concatenated into a unique global completion for F that is consistent with the revealed coordinates x_S. Conversely, every global completion for F induces a compatible family of local completions for $\widetilde{F}$. Therefore S is sufficient for the original model F if and only if it is strongly sufficient for the regrouped model $\widetilde{F}$.

The assumptions of the theorem ensure that the local optimization problems for the merged components of $\widetilde{F}$ satisfy the hypotheses of Theorem 9. Hence Algorithm 1, when run on $\widetilde{F}$, terminates after a linear number of outer refinement rounds and returns a minimum-cardinality strongly sufficient explanation $S^\star$ for $\widetilde{F}$. By the equivalence above, $S^\star$ is a sufficient explanation for the original model F.

To prove cardinal minimality for F, suppose toward contradiction that there exists a sufficient explanation S for F with $|S| < |S^\star|$. By the same equivalence, S is strongly sufficient for $\widetilde{F}$. This contradicts the minimum-cardinality guarantee returned by Theorem 9. Therefore $S^\star$ is a cardinally-minimal sufficient explanation for F.

It remains to justify the complexity statement. The merged components can be computed by a union-find pass over the interaction scopes: initialize one set for each feature and, for every scope $T = \{i_1, \dots, i_q\}$, perform the unions $\mathrm{union}(i_1, i_j)$ for $j = 2, \dots, q$. Two features end up in the same union-find class exactly when they are connected by a chain of overlapping interaction scopes. This preprocessing step is linear up to inverse-Ackermann factors in the total hyperedge-incidence size.

After preprocessing, every verifier or MILP subproblem in Algorithm 1 is posed over a merged component C_ℓ of size at most $k = \max_{\ell \in [m]} |C_\ell|$. Consequently, the per-round and termination guarantees are exactly those of Theorem 9, with the worst-case exponential dependence governed by k rather than by the total number of features. □

G Additional Experimental Details

G.1 Setup

We evaluate both algorithmic settings introduced in the main text: the dual-MILP algorithm for strong sufficiency (Section 5) and its sparse reduction with preprocessing (Section 6). We compare both methods against two verifier-backed greedy baselines. The first is a plain greedy baseline that starts from the full set of input features, visits the features in a fixed canonical order, and after each tentative removal checks sufficiency through a CROWN-style verifier (α-CROWN in our implementation) [117]. The second is a sensitivity-greedy baseline that uses the same remove-and-verify loop, but orders features from least to most locally sensitive. The generic template appears in Appendix C as Algorithm 2; the two baselines differ only in the choice of the ordering routine.

All experiments are run on a SLURM cluster. We launch one job per explained test-set chunk, allocating a single GPU and 128 GB of RAM to each job. Unless noted otherwise, we use α-CROWN both to derive deviation bounds for our methods and to answer the baselines' sufficiency queries, while Gurobi [55] solves the optimistic and pessimistic MILP programs.

G.2 Training and Explanation Hyperparameters

Data are split once into stratified 60/20/20 train/validation/test partitions with seed 42. Continuous features are imputed with the median and standardized using statistics fit on the training split only. Categorical features are imputed with the most frequent value and ordinal-encoded, with unseen categories mapped to -1 at evaluation time.

The default K-Order NAM training configuration uses ReLU units, hidden layer widths $(64, 64, 32)$, dropout 0.1, feature dropout 0.05, learning rate 10^{-3}, weight decay 10^{-4}, batch size 256, a maximum of 200 epochs, early stopping with patience 20, and a `ReduceLROnPlateau` scheduler with patience 10 and multiplicative factor 0.5. The training script also records the exact model order, component counts, maximum merged-group size, validation/test metrics, and number of trained epochs in the saved metrics file for each run.

For the explanation experiments, the accompanying runner uses $\epsilon = 0.1$ and $p = \infty$ in the reported benchmark tables unless stated otherwise. The MILP pipeline uses at most 10 outer tightening iterations by default, while the greedy baselines use 20 α-CROWN refinement steps per verification query. The default discrete-domain threshold is 20 distinct values, the fallback enumeration threshold for greedy verification is 10^6, and the per-example timeout is disabled by default. The sparse-setting experiments are obtained by enabling the preprocessing flag in the same experiment runner.

G.3 Reported Statistics

The main-text tables report test accuracy, mean explanation size (as a percentage of the total number of features), and mean wall-clock runtime over the explained test set. To avoid reducing explanation-size behavior to a single aggregate number, we also report the full empirical explanation-size distribution through histograms/bar charts over explained examples. Runtime variability is summarized with standard deviation information in the corresponding plots, and the appendix ablation tables additionally report medians where that helps characterize skewed runtime distributions. Because the accompanying artifact stores the per-example outputs for every explained test point, all reported means, medians, standard deviations, and histograms can be recomputed directly from the saved result files.

G.4 Compute Resources and External Requirements

Training and explanation experiments are orchestrated through SLURM. The accompanying experiment engine processes the test set in contiguous chunks and is designed to launch one job per chunk; in our reported runs, each such job uses a single GPU worker and 64 GB of RAM. The training code detects a CUDA GPU when available and otherwise falls back to Apple MPS or CPU execution, while the explanation code records per-example wall-clock time in the saved results so that total benchmark runtime can be reconstructed exactly from the artifact.

The MILP-based experiments require a Gurobi installation and a valid Gurobi license. Academic users can request a free academic license. Beyond Gurobi, the remaining external assets used in the paper—including the public tabular datasets and the open-source software dependencies used by the training and verification pipelines—are available under their standard published terms and do not require an additional commercial license.

G.5 Dataset and Model Summary

We use nine public tabular benchmarks: Adult, Bike Sharing, Breast Cancer Wisconsin, California Housing, COMPAS, Diabetes, German Credit, Heart Disease, and HELOC. Adult, Bike Sharing, COMPAS, German Credit, and Heart Disease are mixed discrete–continuous tasks, while Breast Cancer, California Housing, Diabetes, and HELOC are fully continuous. The table below reports the exact split sizes and feature-type counts for all benchmarks used in the main-text experiments.

For every benchmark, we evaluate explanations on the full held-out test split, so the number of explained examples in Section 7 coincides with the test-set size listed below.

Dataset	Total	Train	Val	Test	Features	Continuous	Categorical
Adult	48,842	29,305	9,768	9,769	14	6	8
Bike Sharing	17,379	10,427	3,476	3,476	12	4	8
Breast Cancer	569	341	114	114	30	30	0
California Housing	20,640	12,384	4,128	4,128	8	8	0
COMPAS	5,278	3,166	1,056	1,056	13	2	11
Diabetes	768	460	154	154	8	8	0
German Credit	1,000	600	200	200	20	7	13
Heart Disease	303	181	61	61	13	5	8
HELOC	10,000	6,000	2,000	2,000	22	22	0

G.6 Additional Strong-Sufficiency Curves

The following figures show the strong-sufficiency plots omitted from the main text.

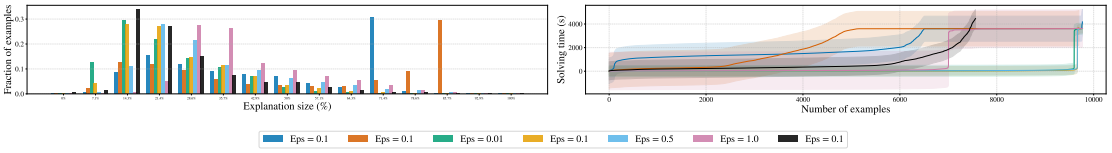

Figure 3: Adult: combined explanation-size and runtime plot under strong sufficiency.

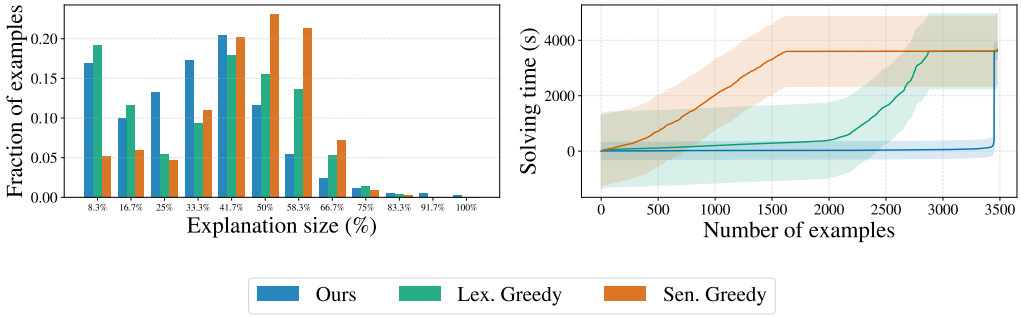

Figure 4: Bike Sharing: combined explanation-size and runtime plot under strong sufficiency.

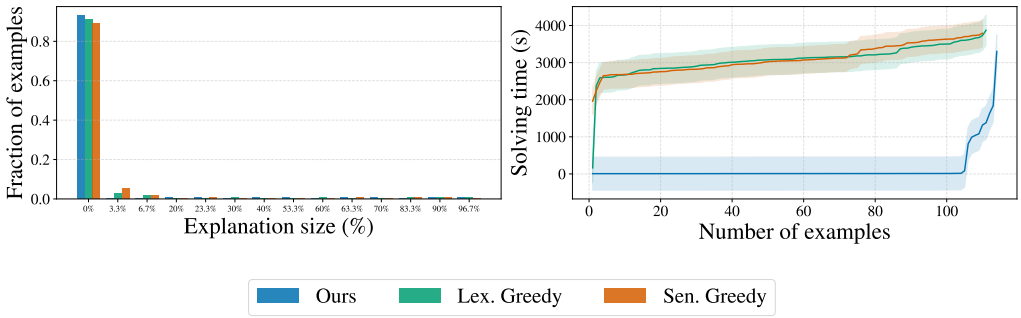

Figure 5: Breast Cancer: combined explanation-size and runtime plot under strong sufficiency.

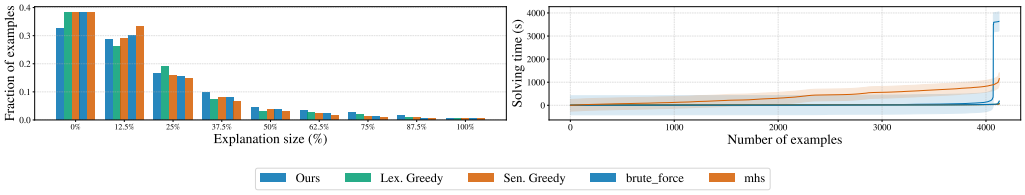

Figure 6: California Housing: combined explanation-size and runtime plot under strong sufficiency.

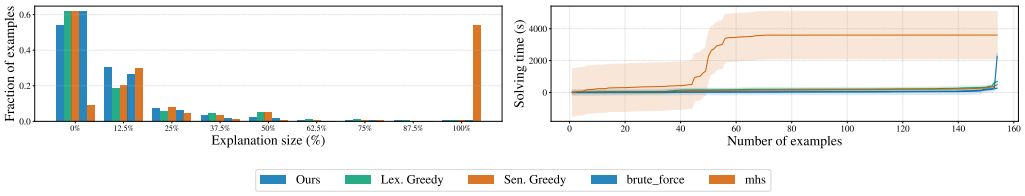

Figure 7: Diabetes: combined explanation-size and runtime plot under strong sufficiency.

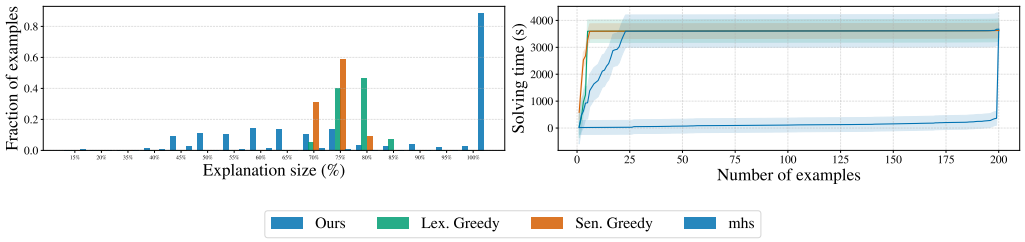

Figure 8: German Credit: combined explanation-size and runtime plot under strong sufficiency.

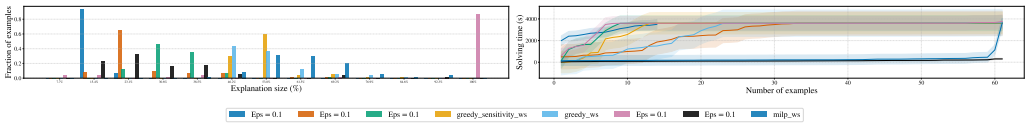

Figure 9: Heart Disease: combined explanation-size and runtime plot under strong sufficiency.

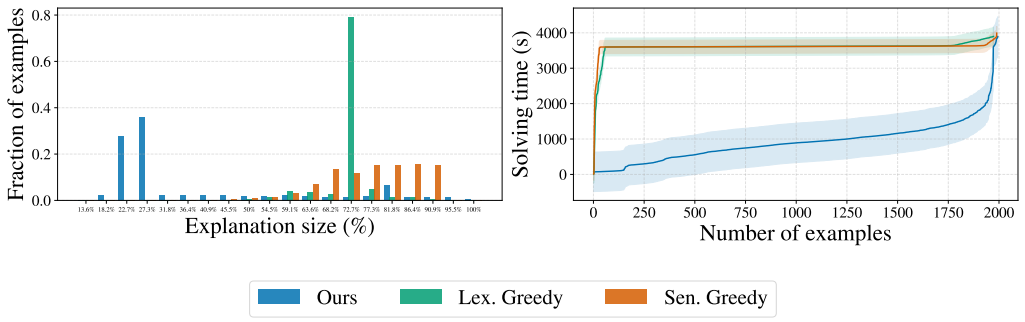

Figure 10: HELOC: combined explanation-size and runtime plot under strong sufficiency.

H Ablation Studies

H.1 Sensitivity to ϵ

As ϵ grows, certificates generally get larger, and runtime tails become heavier, though most inputs remain easy and a small set of hard examples drives the worst runtimes.

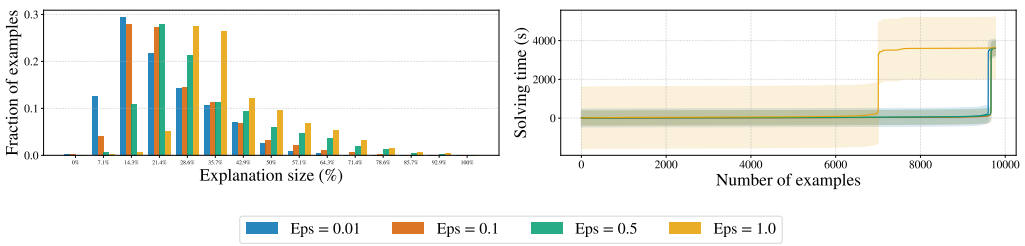

Figure 11: Adult under varying ϵ. Combined explanation-size and runtime plot.

H.2 Sensitivity to the Admissible Input Space

The ϵ-ball is a local special case of the more general distance-restricted sufficiency setting: on a bounded domain, a sufficiently large radius covers the entire admissible input space. To demonstrate that our method applies without modification in this unrestricted regime, we re-run the Heart Disease benchmark (strong-sufficiency setting) using the full training-data range $[\min_i x_i, \max_i x_i]$ for each continuous feature as its free region, instead of an ϵ-ball.

Table 3: Heart Disease under strong sufficiency: $\epsilon = 0.1$ vs. the full admissible input space. Each cell reports mean size $\pm$ sd (%) / mean runtime $\pm$ sd (s) / solved instances.

Setting	Ours	Greedy	Greedy Sensitivity
$\epsilon = 0.1$	29.1±14.4% / 130.5±54.8 / 61/61	27.1±10.3% / 2765.5±1121.9 / 48/61	34.2±7.5% / 3369.3±695.3 / 11/61
Whole admissible space	61.2±10.5% / 303.0±453.1 / 60/61	53.5±8.8% / 2820.5±1223.1 / 51/61	53.5±7.5% / 3196.4±996.5 / 13/61

As expected, every method's certified explanations grow once the perturbation space is unbounded, since more features must be fixed to guarantee the prediction cannot change anywhere in the admissible space. Our method continues to solve nearly every instance (60/61) and remains the fastest of the three; the sensitivity-guided greedy baseline is hit hardest by the harder setting, solving only 13/61 instances within budget. Removing the distance restriction makes the problem harder for every method, but our relative advantage persists.

H.3 First- vs. Second-Order NAMs

To isolate the effect of pairwise interactions, we compare three model classes on four representative datasets: a first-order NAM baseline, a full second-order NAM solved with the strongly-sufficient algorithm, and a sparse second-order NAM solved with the sparse algorithm. Adding second-order interactions improves predictive accuracy on three of the four datasets (Bike Sharing, California Housing, COMPAS); on Heart Disease, all three models tie exactly. The explanation-size picture now favors simpler models: the full second-order NAM yields the largest mean certificate on every dataset here, consistent with Section 7's observation that strong sufficiency carries no size guarantee relative to a simpler model. The sparse second-order variant recovers the smallest mean certificate on COMPAS and Heart Disease, while the first-order NAM is smallest on Bike Sharing and narrowly ahead of sparse on California Housing. Runtime still reflects the added expressive power: the first-order NAM is consistently fastest, while the sparse second-order pipeline incurs the largest overhead on Bike Sharing, COMPAS, and Heart Disease.

Table 4: First- vs. second-order NAM ablation on four datasets at $\epsilon = 0.1$ on the full test split.

Dataset	Model	Acc.	Size mean	Size med	Time mean (s)	Time med (s)
Bike Sharing	First-order	0.8800	1.74	1.0	1.651	1.082
Bike Sharing	Second-order	0.9232	4.02	4.0	61.279	26.505
Bike Sharing	Sparse second-order	0.9163	2.82	3.0	242.984	118.560
California Housing	First-order	0.8619	1.22	1.0	5.786	1.767
California Housing	Second-order	0.8886	1.57	1.0	71.860	7.564
California Housing	Sparse second-order	0.8777	1.26	1.0	49.813	18.113
COMPAS	First-order	0.6648	3.69	4.0	3.906	1.138
COMPAS	Second-order	0.6742	6.58	6.0	26.411	18.285
COMPAS	Sparse second-order	0.6799	2.56	2.0	40.424	30.043
Heart Disease	First-order	0.8525	2.85	3.0	1.069	1.035
Heart Disease	Second-order	0.8525	3.79	3.0	130.520	116.814
Heart Disease	Sparse second-order	0.8525	2.46	2.0	182.618	98.464

Figures 12–15 show the corresponding per-example distributions across the full test split of each dataset.

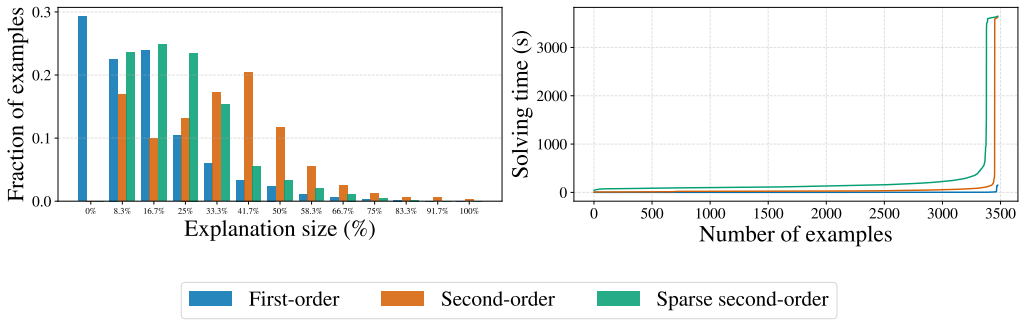

Figure 12: Bike Sharing: combined explanation-size and runtime plots for the first-vs-second-order ablation across explained test points.

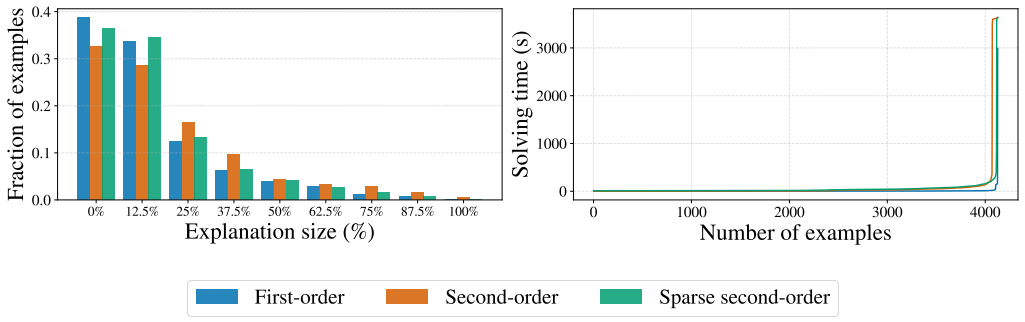

Figure 13: California Housing: combined explanation-size and runtime plots for the first-vs-second-order ablation across explained test points.

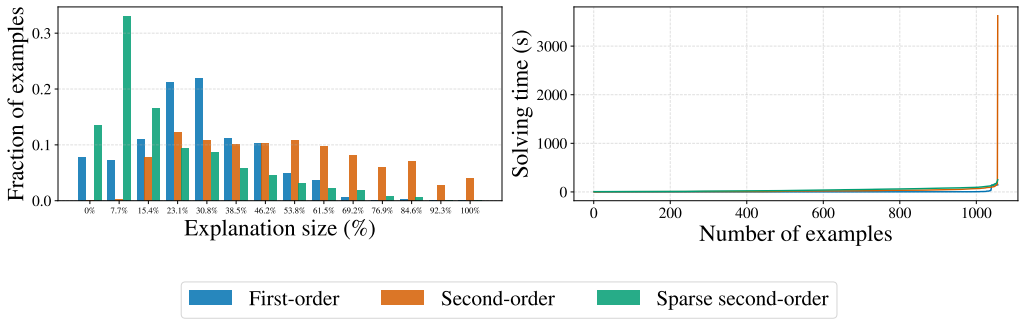

Figure 14: COMPAS: combined explanation-size and runtime plots for the first-vs-second-order ablation across explained test points.

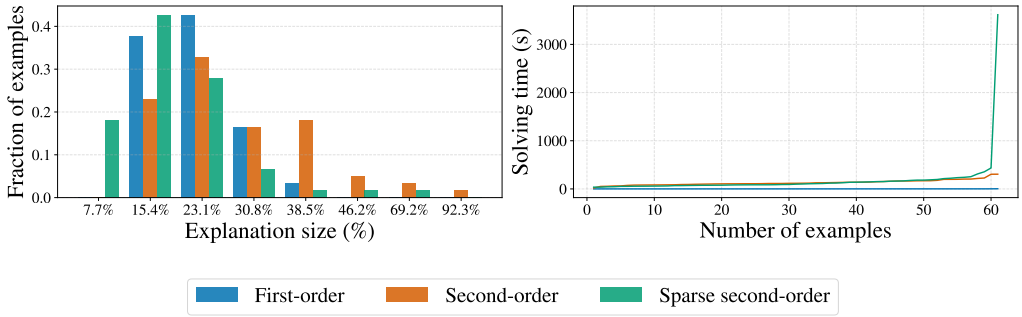

Figure 15: Heart Disease: combined explanation-size and runtime plots for the first-vs-second-order ablation across explained test points.

H.4 Preliminary Analysis: When Does the Gap Over Greedy Grow?

Section 7 observes that our method's advantage over the greedy baselines varies substantially across benchmarks. As a first step toward quantifying this, we measure, for each example solved by both our method and canonical-order greedy under strong sufficiency, the Pearson correlation between the gap (greedy's explanation size minus ours) and the margin $|F(x)|$ at that example, pooling and per-dataset.

Table 5: Correlation between the ours-vs-greedy size gap and $|F(x)|$, under strong sufficiency ($\epsilon = 0.1$).

Dataset	N	r(gap, $\lvert F(x)\rvert$)
Adult	6477	0.233
Bike Sharing	3017	0.057
Breast Cancer	105	−0.138
California Housing	4070	0.360
COMPAS	357	0.787
Diabetes	154	−0.055
Heart Disease	48	0.648
HELOC	70	−0.255
Pooled	14302	0.158

German Credit is omitted: canonical greedy times out on all but 4 of its 200 instances, too few for a meaningful correlation. Among the rest, the pooled correlation is positive and statistically significant given the sample size, but modest in magnitude and highly heterogeneous across datasets, strong on COMPAS and Heart Disease but flat or slightly negative on Breast Cancer, Diabetes, and HELOC. This is consistent with, though more mixed than, the intuition that our method's advantage grows with decision margin; it does not yet account for the complementary factor, the spread of explanation

sizes across alternative sufficient subsets (which requires repeated greedy runs under randomized feature orderings), left for future work.

H.5 Higher-Order NAMs

We also study how increasing interaction order affects the sparse family on Bike Sharing. In this controlled study, accuracy improves monotonically with order, from 0.9163 at order 2 to 0.9180 at order 3 and 0.9295 at order 4. The improved predictive performance comes at a cost: the mean explanation size increases from 2.82 at order 2 to 2.90 at order 3, and the mean solve time rises from 242.984s to 913.630s over the same range. At order 4, the largest merged components grow further and convergence becomes substantially harder: only 1096 of 3450 examples (31.8%) produce a usable explanation within the per-example time budget (3600s), compared to essentially full coverage at orders 2 and 3; the statistics below for order 4 are computed over that smaller, converged subset only; and are likely biased toward comparatively easier instances. Thus, higher-order interactions continue to improve accuracy, but certified explanations become both larger and substantially more expensive to compute, to the point of a practical tractability wall at order 4 on this benchmark.

Table 6: Bike Sharing higher-order ablation for sparse models at $\epsilon = 0.1$ (orders 2–4; full test split). Order 4's statistics are computed only over the 1096/3450 (31.8%) examples that produced a usable explanation within the 3600s per-example budget; see text.

Order	Acc.	Size mean	Size med	Time mean (s)	Time med (s)
2	0.9163	2.82	3.0	242.984	118.560
3	0.9180	2.90	3.0	913.630	634.051
4	0.9295	3.37	3.0	3500.879	3735.853

H.6 Comparison Against Exact and Subset-Minimal Baselines

The greedy baselines compared against in the main text are, by construction, only subset-minimal, which could in principle account for some of our method's advantage. We therefore compare against two further baselines spanning the subset-minimal and cardinally-minimal ends of the literature, under the strong-sufficiency setting (Section 5), holding the verifier fixed across all methods. Any Any-Order NAM can equivalently be written as a standard neural network with sparse, block-structured weight matrices, so general-purpose explanation algorithms apply to it directly without modification; both baselines here, and the two greedy baselines in the main text, treat the model exactly this way, so any gains we observe reflect how well a method exploits the additive interaction structure rather than a stronger verifier or a weaker certification standard.

VeriX+. VeriX [110] is, with the verifier held fixed, a sensitivity-guided sequential remove-and-verify procedure, coinciding with our sensitivity-guided greedy baseline. We instead compare against VeriX+ [111], which is algorithmically distinct: a QuickXplain-based divide-and-conquer search [67] that confirms a whole block of features as safe to free with a single query whenever possible, using far fewer oracle queries than one-at-a-time removal. On the full Adult test set, our method solves every instance (9665/9769, mean size 25.7$\pm$13.2%, median runtime 18.3s), whereas VeriX+ solves 7562/9769 within the same per-job time budget, with mean size 25.5$\pm$13.8% and median runtime 332.1s on the instances it solves. Explanation size is essentially tied between the two methods here; the runtime advantage of our method remains an order of magnitude, and it additionally solves every instance within budget, whereas VeriX+ does not.

Brute-force and MHS. We also compare against two baselines that target cardinal minimality directly: an exact brute-force search over subsets of increasing cardinality, and a minimum-hitting-set (MHS) enumeration algorithm exploiting the duality between contrastive and sufficient explanations [19, 61], solved with the MaxSAT solver RC2 [60]. Both are restricted to lower-dimensional benchmarks, since exact search is not tractable at higher dimensionality.

Table 7: Exact cardinally-minimal baselines vs. our method under strong sufficiency ($\epsilon = 0.1$). Each cell reports mean size $\pm$ sd / median runtime (s) / solved instances.

Dataset (# features)	Ours	Brute force	MHS
Diabetes (8)	0.82±1.33 / 29.0 / 154/154	0.62±1.13 / 41.0 / 154/154	4.81±3.52 / 3600.1 / 72/154
California Housing (8)	1.57±1.74 / 7.6 / 4070/4125	1.26±1.49 / 6.5 / 4128/4128	1.17±1.41 / 324.9 / 4128/4128
German Credit (20)	12.99±2.85 / 110.3 / 199/200	n/a / n/a / 0/200	18.85±3.40 / 3607.2 / 23/200
Heart Disease (13)	3.79±1.87 / 116.8 / 61/61	2.07±0.27 / 2783.6 / 14/61	11.77±3.32 / 3603.1 / 8/61

Brute-force only completes within budget on the two lowest-dimensional benchmarks (Diabetes, California Housing with 8 features each); on those, it is close to our method's size, as expected since it recovers the true cardinally-minimal ordinary-sufficient explanation. It times out entirely on German Credit's 20 features. MHS solves a minority of instances everywhere except California Housing, and takes orders of magnitude longer than our method where it does solve. Taken together, exact cardinally-minimal search and MHS do not scale to the dimensionalities in our main benchmarks, which is precisely why the main text compares against verifier-backed greedy baselines instead; our method remains close to the true optimum where one can be computed at all.

H.7 Validating the Sparsity Assumption via SIAN

The sparsity assumption of Section 6 is controlled by construction in our main experiments, which raises the question of whether small merged components (s small) also arise when interactions are discovered from data rather than imposed by us. We investigate this using Sparse Interaction Additive Networks (SIAN) [48]: a base DNN is trained, its most important interactions are detected via hierarchical Archipelago-based scoring, and a sparse Any-Order NAM is built from exactly those detected interactions. We apply this pipeline to the Diabetes and Heart Disease benchmarks and then run our exact sparse-setting algorithm on the resulting models.

Table 8: SIAN-detected interaction models: predictive accuracy and certified explanations under strong sufficiency (full test split, $\epsilon = 0.1$).

Dataset	DNN acc.	SIAN acc.	Order	s	Size mean $\pm$ sd (%) / Runtime mean $\pm$ sd (s)
Diabetes	0.7532	0.7273	4	4	8.4±15.0 / 66.1±63.7
Heart Disease	0.8689	0.8361	3	3	16.5±12.8 / 59.3±40.6

Both benchmarks yield data-discovered merged components of size 3–4 out of 8 and 13 features respectively, i.e., exactly the small-s regime our method targets, with no component size imposed by us. The SIAN models remain competitive with the underlying DNN in accuracy, and our exact method solves every instance (100% converged, 0 timeouts on both benchmarks), supporting the practical relevance of the sparsity assumption on models derived from standard DNNs rather than constructed by hand.

H.8 A Qualitative Case Study

To illustrate what a certified explanation looks like in practice, we walk through test patient 38 of the Heart Disease benchmark (strong-sufficiency model of Table 1). The model predicts no heart disease for this patient. Our method returns the five-feature certificate $\{\text{cp}, \text{restecg}, \text{exang}, \text{slope}, \text{thal}\}$ (chest-pain type, resting ECG, exercise-induced angina, ST-segment slope, and thalassemia status), and because our MILP converged, this additionally certifies that no explanation with four or fewer features exists. The sensitivity-guided greedy baseline instead returns an eight-feature set, $\{\text{sex}, \text{cp}, \text{restecg}, \text{thalach}, \text{exang}, \text{oldpeak}, \text{slope}, \text{ca}\}$, which is also a valid sufficient explanation but nearly twice as large, presenting a clinician with more factors than are actually needed to certify the prediction.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the hardness result, the two tractable regimes, and the empirical findings consistently with the technical sections and appendices.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The "Limitations and Future Work" section discusses the dependence of the approach on neural-network verification and MILP technology, noting that certification cost can grow quickly with network size, depth, and interaction structure.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: Assumptions are stated with the theoretical results, proof sketches appear in the main text, and the full omitted proofs are provided in Appendices B and C.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper is accompanied by the full codebase, experiment scripts, configurations, and result files needed to reproduce the main empirical results, together with the paper's description of the datasets, exact splits, and evaluation pipeline.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The paper is accompanied by an anonymized code artifact with the full training pipeline, implementations of all methods used in the experiments, the experiment engine and cluster-dispatch utilities, the released result files, and a detailed README that explains how to reproduce the paper's results.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Appendix G specifies the preprocessing pipeline, stratified data splits, model architecture defaults, optimization hyperparameters, early-stopping schedule, explanation-run settings, and the saved artifact structure used for reproduction.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: The paper reports the full empirical explanation-size distributions through histograms/bar charts, summarizes runtime variability with standard deviation information in the plots, and includes aggregate mean/median summaries in the accompanying tables.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix G states that the training and explanation runs are orchestrated through SLURM, that each reported explanation job uses a single GPU worker with 64 GB of RAM, and that per-example wall-clock times are stored in the released results.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The work is methodological and uses standard benchmark datasets without human-subject interaction or restricted-release components.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [N/A]

Justification: This paper is a foundational theoretical and algorithmic study of certified explanations, rather than a deployed socio-technical system or released high-risk application, so a separate broader-impacts discussion is not applicable here.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

Answer: [N/A]

Justification: The paper does not release high-risk models or scraped datasets, and the experiments are limited to standard tabular benchmarks.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Appendix G specifies that the MILP experiments require Gurobi and notes the availability of a free academic license, while the remaining datasets and software dependencies used in the paper are public or open-source assets used under their standard published terms and cited in the bibliography.

Guidelines:

- The answer [N/A] means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release a new dataset, model, or software artifact.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing experiments or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [N/A]

Justification: The paper does not involve human subjects, so IRB approval is not applicable.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: LLMs are not part of the core methodology, experiments, or claimed scientific contribution.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.