Interpretability-by-Design with Accurate Locally Additive Models and Conditional Feature Effects

Anonymous Author(s)
Affiliation
Address
email

Abstract

Generalized additive models (GAMs) offer interpretability through independent univariate feature effects but underfit when interactions are present in data. GA^{2}Ms add selected pairwise interactions which improves accuracy, but sacrifices interpretability and limits model auditing. We propose *Conditionally Additive Local Models* (CALMs), a new model class, that balances the interpretability of GAMs with the accuracy of GA^{2}Ms. CALMs allow multiple univariate shape functions per feature, each active in different regions of the input space. These regions are defined independently for each feature as simple logical conditions (thresholds) on the features it interacts with. As a result, effects remain locally additive while varying across subregions to capture interactions. We further propose a principled distillation-based training pipeline that identifies homogeneous regions with limited interactions and fits interpretable shape functions via region-aware backfitting. Experiments on diverse classification and regression tasks show that CALMs consistently outperform GAMs and achieve accuracy broadly comparable to GA^{2}Ms, while preserving the univariate auditability that GA^{2}Ms forfeit. Overall, CALMs offer a favorable trade-off between predictive accuracy and interpretability.

1 Introduction

In high-stakes decision making, machine learning models must provide not only reliable predictions but also human-understandable explanations [31, 10]. This requirement has motivated the development of *interpretable-by-design* models, whose structure permits direct inspection of their behavior [30]. However, interpretability is a continuum rather than a binary property: interpretable-by-design models span a spectrum where gains in predictive accuracy often entail greater structural complexity [38, 27].

This trade-off is particularly evident in GAMs [28] and their extension, GA^{2}Ms [7]. GAMs achieve high interpretability by modeling predictions as a sum of independent univariate effects, but this strict additivity prevents them from capturing feature interactions, limiting their predictive accuracy. GA^{2}Ms add selected pairwise interactions, improving performance at the cost of crucial interpretability properties [35]. Specifically, interaction terms (i) obscure the unique attribution of the prediction to individual features because the pairwise interaction terms are generally not additively separable (ii) complicate global auditing by requiring the simultaneous inspection of multiple (univariate and bivariate) effects. Section 3.3 analyzes these limitations in detail.

We introduce *Conditionally Additive Local Models (CALMs)*, a new model class that strikes a balance between the predictive accuracy of GA^{2}Ms and the interpretability of GAMs. The core idea is to learn *conditional feature effects*: multiple univariate effects per feature, each active within a distinct region of the input space where the feature exhibits nearly additive behavior—that is, where it minimally interacts with others. These regions are defined through threshold-based conditions on interacting

Submitted to 40th Conference on Neural Information Processing Systems (NeurIPS 2026). Do not distribute.

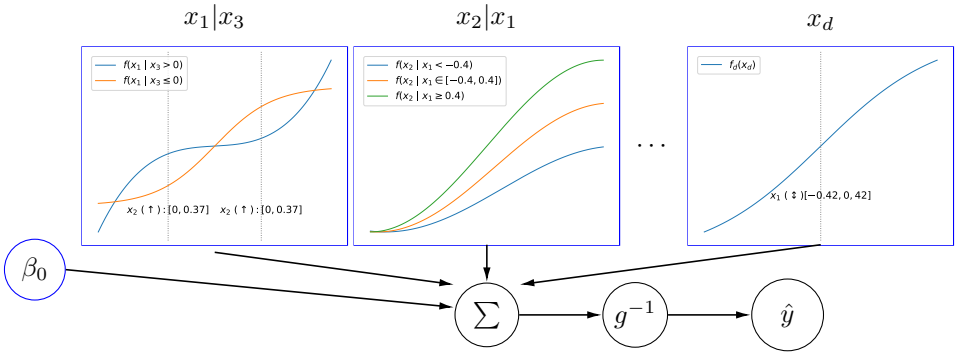


Figure 1: CALMs use *conditional feature effects*: every feature effect is expressed by a collection of 1D functions, each associated with a different region of the input space. The regions are defined conditioning on the interacting features. In the example, the effect of x_1 conditions on x_3, the effect of x_2 on x_1, while x_d does not interact with any other feature and thus has a single plot. CALMs are *interpretable-by-design*—summarized in d figures of 1D plots—and *accurate*, as they can model feature interactions.

features. For example, if x_i interacts with x_j, a CALM may learn separate effects for x_i; one when $x_j < \tau$ and another when $x_j \geq \tau$ for some threshold τ. Figure 1 illustrates this representation. Conditional effects thus enable interaction-aware modeling while maintaining (i) transparent feature contributions and (ii) straightforward global model auditing.

To fit CALMs to data, we propose a three-step distillation-based pipeline: (1) Train a black-box reference model to capture complex interactions present in the data; (2) Partition the input space independently for each feature using CART-based splitters [17] optimized to minimize heterogeneity—a criterion that, when minimized, provably reduces feature interactions within the resulting regions [19]; (3) Fit region-specific shape functions using a region-aware extension of the standard backfitting algorithm. This procedure efficiently identifies interacting features (e.g., x_i interacts with x_j), optimal split points (e.g., τ such that separate effects are learned for $x_j < \tau$ and $x_j \geq \tau$), and requires minimal tuning—primarily the maximum allowed number of interactions per feature. Extensive evaluation on diverse regression and classification benchmarks demonstrates that CALMs consistently outperform GAMs in predictive accuracy, and often match GA^{2}Ms. Furthermore, we formally show that CALMs satisfy key interpretability properties unsupported by GA^{2}Ms. Code for reproducing all experiments is provided in the supplementary material.

Contributions. (i) We introduce *Conditionally Additive Local Models* (CALMs), a novel interpretable-by-design model class that balances GAM's interpretability and GA^{2}Ms accuracy via conditional feature effects. (ii) We develop a robust training algorithm based on distillation, heterogeneity-minimization and region-aware backfitting. (iii) We provide a formal analysis of the interpretability properties inherent in CALMs. (iv) We present extensive empirical evidence demonstrating improved accuracy–interpretability trade-offs over GAMs and GA^{2}Ms. (v) A within-subjects user study (N=25) suggests that non-expert users interpret CALM plots more accurately than heatmap-based bivariate-interaction visualizations (Appendix D).

2 Background and Related Work

Interpretable-by-design models enforce transparency in their decision-making process. Examples include decision trees [4], rule lists [26, 2], and prototype-based classifiers [9, 3]. Among them, GAMs [16, 42] stand as a particularly popular candidate, mainly due to their simple global interpretability.

GAMs model the output as $g(\mathbb{E}[y|\mathbf{x}]) = \beta_0 + \sum_i f_i(x_i)$, where each f_i is a univariate shape function, β_0 is the global intercept, and g is a link function. The key benefit of GAMs is that feature effects can be *independently* visualized through a simple 1D plot. Methods for learning f_i for all i include spline-based approaches [28, 42], gradient boosting [11], and neural-based variants [1, 25, 35]. However, standard GAMs assume that features contribute independently to the output, which limits

their accuracy when feature interactions are present in data. To capture these interactions, GA^{2}Ms add pairwise terms: $g(\mathbb{E}[y|\mathbf{x}]) = \beta_0 + \sum_i f_i(x_i) + \sum_{i<j} f_{ij}(x_i, x_j)$ [29]. GA^{2}Ms instances vary in (i) how they select the top-K interactions to maintain sparsity and (ii) how they learn f_i and f_{ij}. For example, [29] greedily selects interactions based on loss reduction, while, neural-based approaches [43, 23, 8] integrate pairwise interactions into standard neural-based architectures.

While GA^{2}Ms improve accuracy, they obscure individual feature contributions and complicate model auditing (see Section 3.3). A key open question remains: *Can we approach GA^{2}M-level accuracy while maintaining GAM-level interpretability?* To this end, we draw inspiration from *regional effect methods*, which handle interactions by partitioning the feature space into subregions where interactions are weak. To understand regional effects, consider first how standard global effects work. Global effect plots, such as PDP, ALE or SHAP-DP, explain a black box model by decomposing its complex d-dimensional function $f(\mathbf{x}) \to y$ into d one-dimensional plots $x_i \to y$. However, when strong interactions exist between x_i and other features, these plots can yield misleading explanations [12, 13]. Regional feature effect plots address that by partitioning the feature space into subregions where interactions are minimal [17, 19]. Within each subregion, simple univariate plots accurately represent feature effects without being confounded by interactions with other features. We adopt this strategy to identify low-interaction subregions and then we fit shape functions within each subregion.

Our approach relates to model distillation [20, 40] and surrogate modeling [15]. These approaches train an interpretable "student" model to mimic a complex "teacher", either locally [36] or globally [15]. However, instead of explaining the teacher, we use it to identify subregions with minimal feature interactions, on which we then fit shape functions. In a rough analogy, CALM serves as the interpretable-by-design "student" that replaces the black-box "teacher" as the final predictor. Other works that share the above idea are: SLIM [22] and related analyses [18] propose tree-based models with simple predictors in each region, mainly in the context of model distillation and explanation. In contrast, our goal is not to approximate a black-box model, but to construct an interpretable-by-design predictor with GAM-like structure. Related ideas also appear in [14], which leverages regional feature effects to build interpretable models; however, that work does not explicitly characterize the interpretability guarantees of the learned model or evaluate them systematically.

3 CALM: Conditionally Additive Local Model

CALM captures feature interactions via *conditional feature effects*, a set of univariate shape functions per feature, each active in a different region of the input space.

3.1 Model Formulation

Let $\mathbf{X} = (X_1, \ldots, X_d) \in \mathcal{X} \subseteq \mathbb{R}^d$ be a random vector with joint distribution $P_{\mathbf{X}}$ over the input data, $\mathbf{x} = (x_1, \ldots, x_d)$ where $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_d$. We use the subscript $-i$ to denote quantities excluding the i-th feature (e.g., $\mathbf{X}_{-i}$ defined on space $\mathcal{X}_{-i}$). Let also Y be the output random variable, with $Y \in \mathbb{R}$ for regression and $Y \in \{0, 1\}$ for binary classification. CALM is then defined as

$$g\left(\mathbb{E}[Y \mid \mathbf{X} = \mathbf{x}]\right) = \beta_0 + \sum_{i=1}^{d} f_i^{(r_i(\mathbf{x}_{-i}))}(x_i) \tag{1}$$

where $\beta_0 \in \mathbb{R}$ is an intercept and g is a link function (identity for regression and logit for classification). For each feature i, CALM learns a set of univariate shape functions $\{f_i^{(r)}\}_{r=1}^{R_i}$. At prediction time, a region selection function $r_i(\mathbf{x}_{-i}) : \mathbb{R}^{d-1} \to \{1, \ldots, R_i\}$ chooses the specific shape function for x_i based on the context of other features $\mathbf{x}_{-i}$. This allows the effect of x_i to vary across regions, effectively modeling feature interactions, while maintaining univariate interpretability within each region. The final prediction is given by:

$$f_{\texttt{CALM}}(\mathbf{x}) = g^{-1}\left(\beta_0 + \sum_{i=1}^{d} f_i^{(r_i(\mathbf{x}_{-i}))}(x_i)\right) \tag{2}$$

For brevity, we use $\hat{y}(\mathbf{x})$ to denote the prediction $f_{\texttt{CALM}}(\mathbf{x})$. As in standard GAMs, identifiability requires region-wise centering of each shape function; we adopt the centering constraint stated in Appendix A.3.1 (Eq. 15). The selection function $r_i(\mathbf{x}_{-i})$ partitions the input space (excluding x_i)

Algorithm 1 Training a CALM model

Require: Training data $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$
Output: CALM predictor $f_{\texttt{CALM}}(\mathbf{x})$
1: **Step 1:** Train reference model $f_{\texttt{ref}}$ on $\mathcal{D}$.
2: **Step 2:** Learn feature-specific trees T_i, $i = 1, \ldots, d$.
3: **Step 3:** Estimate shape functions $\{\{f_i^{(r)}\}_{r=1}^{R_i}\}_{i=1}^{d}$.

into $\mathcal{P}_i := \{\mathcal{R}_i^{(r)}\}_{r=1}^{R_i}$, independently for each feature i, where each $\mathcal{R}_i^{(r)} \subseteq \mathcal{X}_{-i}$. The goal of these partitions is to minimize interactions between x_i and other features within each region, ensuring the effect of x_i is accurately captured by a single univariate shape function. To maintain interpretability, each partition is represented by a binary decision tree T_i of maximum depth d_{max}, built over $\mathbf{x}_{-i}$. Internal nodes apply axis-aligned splits, i.e., inequality thresholds $x_j < \tau$ or $x_j \geq \tau$ for continuous features, and equality tests $x_j = \tau$ or $x_j \neq \tau$ for categorical ones. Each leaf of the tree defines a region $\mathcal{R}_i^{(r)}$. Formally, each region is defined by a conjunction of at most $m_i^{(r)} \leq d_{max}$ rules:

$$\mathcal{R}_i^{(r)} = \left\{ \mathbf{x}_{-i} \;\middle|\; \wedge_{k=1}^{m_i^{(r)}} (x_{j_k} \text{ op}_k \tau_k) \right\}. \tag{3}$$

where $j_k \in \{1, \ldots, d\} \setminus \{i\}$ indexes an interacting feature, $\text{op}_k \in \{<, \geq, =, \neq\}$ is a comparison operator, and $\tau_k \in \mathbb{R}$ is a threshold.

CALM is able to capture feature interactions involving up to $(d_{max} + 1)$ features, as each region conditions on up to d_{max} features. While increasing d_{max} can improve accuracy, it comes at the cost of interpretability. To balance this accuracy-interpretability tradeoff, CALM exposes two hyperparameters: d_{max} (tree depth) bounds the number of shape functions per feature to $R_i \leq 2^{d_{max}}$, while K limits the total number of shape functions across all features via $\sum_i R_i \leq K$. In our experiments, we set $d_{max} = 2$ (yielding $R_i = 4$ regions per feature) and leave K unconstrained.

Notably, CALM learns, at most, $d \cdot 2^{d_{max}}$ univariate shape functions (up to $2^{d_{max}}$ per feature), yet these combine to express up to $2^{d \cdot d_{max}}$ distinct additive models–an exponential increase in expressive power.

3.2 Training algorithm for fitting CALM

Given a dataset $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$, we fit a CALM predictor by minimizing the empirical risk $\mathbb{E}_{(\mathbf{X},Y)\sim\hat{P}_{\mathcal{D}}}[\mathcal{L}(Y, \hat{y}(\mathbf{X}))]$, where $\hat{y}(\mathbf{X})$ denotes a CALM model, $\mathcal{L}$ a loss function and $\hat{P}_{\mathcal{D}}$ the empirical distribution over $\mathcal{D}$. Fitting a CALM requires estimating: (a) a set of d partitions $\mathcal{P}_i$, $i \in \{1, \ldots, d\}$, represented by binary trees T_i (one per feature); (b) region-specific shape functions $\{f_i^{(r)}\}_{r=1}^{R_i}$ and a global intercept β_0. We propose a three-step distillation-based pipeline summarized in Algorithm 1.

Step 1: Train a Reference Model. We train a high-capacity black-box predictor $f_{\texttt{ref}} : \mathcal{X} \to \mathbb{R}$ on $\mathcal{D}$. This model serves as functional proxy for $\mathbb{E}[Y|\mathbf{X}]$ and is used exclusively to detect interactions. The reference model is discarded after training. The choice of $f_{\texttt{ref}}$ is independent of later stages; any accurate predictor can be used, such as gradient-boosted trees (our default), neural networks, random forests or foundation models, like TabPFN [21].

Step 2: Learn feature-specific partitioning trees. This step learns d independent, feature-specific partitions $\mathcal{P}_i := \{\mathcal{R}_i^{(r)}\}_{r=1}^{R_i}$, each represented by a binary tree T_i defined over $\mathbf{x}_{-i}$. The objective is to identify near-additive regions $\mathcal{R}_i^{(r)}$, in which the effect of x_i on the output is well approximated by a univariate function $f_i^{(r)}$, due to locally weak higher-order interactions.

To identify such regions, we use the interaction-related heterogeneity measure from [19]. Feature effect methods (e.g., PDP or ALE) explain a black-box model (like $f_{\texttt{ref}}(\mathbf{x})$) by decomposing it into univariate effects $f_i(x_i)$ for all i. To do so, they first define the local effects $h(x_i, \mathbf{x}_{-i}^{(j)})$ that quantify the contribution of feature x_i on a specific instance $\mathbf{x}^{(j)}$. Then they compute $f_i(x_i)$ by averaging the local effects. Heterogeneity is the variability of these local effects around their average, directly measuring how much x_i's effect depends on other features. High heterogeneity signals strong

interactions, while low heterogeneity identifies near-additive, interaction-free regions—making it an effective criterion for our purpose.

Following the above, we define the *pointwise* heterogeneity of x_i in a region $\mathcal{R}$ (e.g., $\mathcal{R}_i^{(r)}$) as:

$$H_i^{\mathcal{R}}(x_i) = \mathbb{E}_{\mathbf{X}_{-i}|\mathbf{X}_{-i}\in\mathcal{R}}\left[\left(h(x_i, \mathbf{X}_{-i}) - \mu_i^{\mathcal{R}}(x_i)\right)^2\right] \tag{4}$$

where $\mu_i^{\mathcal{R}}(x_i) = \mathbb{E}_{\mathbf{X}_{-i}|\mathbf{X}_{-i}\in\mathcal{R}}\left[h(x_i, \mathbf{X}_{-i})\right]$. The corresponding *feature-level heterogeneity* is:

$$H_i^{\mathcal{R}} = \mathbb{E}_{X_i|\mathbf{X}_{-i}\in\mathcal{R}}\left[H_i^{\mathcal{R}}(X_i)\right] \tag{5}$$

In our experiments, heterogeneity is computed using PDP-based formulas; ALE- and SHAP-DP variants are valid alternatives [17, 19]. While PDP can suffer from extrapolation bias under feature correlations, here it is used only for region detection, not for prediction: even biased local effects yield elevated heterogeneity when genuine interactions are present. When correlations are strong, RHALE [13] is a drop-in alternative, with comparable results in Appendix C. A related caveat is that an interaction with one feature can also raise heterogeneity along its correlated proxies, so CALM may split on either; both choices recover the interaction structure, though the conditioning labels may not be uniquely identifiable under strong collinearity, a limitation shared by partition-based interaction detectors such as REPID.

As additional theoretical justification for using heterogeneity as the splitting criterion, we show that, under specific assumptions, the approximation error of CALM is bounded by the sum of the expected feature heterogeneities. Consequently, reducing heterogeneity of each feature through partitioning yields a CALM model $f_{\texttt{CALM}}$ with lower mean squared error, assuming perfect fit in Step 3 of Algorithm 1. The proof is provided in Appendix A (Proposition A.1).

Therefore, we use $H_i^{\mathcal{R}}$ as the splitting criterion and learn a binary decision tree T_i for each feature x_i using a greedy CART-style algorithm. At each node, we consider splits over all conditioning features $x_j \neq x_i$ and select the split that maximizes the reduction in heterogeneity. For numerical features, splits take the form $x_j\{\leq, >\}\tau$, while for categorical features we use $x_j\{=, \neq\}\tau$. A split is accepted only if the relative reduction in $H_i^{\mathcal{R}}$ exceeds a threshold ϵ; otherwise, the node becomes a leaf. This procedure yields a partition $\mathcal{P}_i = \{\mathcal{R}_i^{(r)}\}_{r=1}^{R_i}$ where heterogeneity is significantly reduced, ensuring the effect of x_i is less distorted by interactions than in the global space. Additional implementation details are provided in Appendix A. In all experiments, we use PDP-based heterogeneity, set the maximum tree depth to $d_{\max} = 2$, and fix $\epsilon = 0.2$. It is important to emphasize that this CART-style algorithm is only used to determine the subregions for each feature and is not used for the final prediction.

Step 3: Estimate Shape Functions. Given the partitions $\{\mathcal{P}_i\}_{i=1}^d$, we estimate the region-specific shape functions $\{f_i^{(r)}\}$ by minimizing the empirical loss of the CALM predictor. We use a modified gradient boosting procedure in which, at each iteration, a single shape function $f_i^{(r)}$ is updated using only the observations whose $\mathbf{x}_{-i}$ fall into the corresponding region $\mathcal{R}_i^{(r)}$ (see Appendix A for details).

Although each update is restricted to a single region, the resulting optimization problem is inherently *coupled*. The regions are defined separately for each feature, so the subsets of samples used to update different shape functions generally overlap. Consequently, updating one $f_i^{(r)}$ changes the residuals seen by all other shape functions. Therefore it can be understood as a coordinated optimization of a single additive predictor with region-gated components.

The following proposition formalizes this intuition by characterizing the target of Step 3 at the population level and its convergence behavior under idealized updates. The proof is in Appendix A. In practice, Step 3 is implemented using gradient boosting to approximate the exact regional updates.

Proposition 3.1 (Optimality and convergence)**.** *Let $m(\mathbf{x}) = \mathbb{E}[Y \mid \mathbf{X} = \mathbf{x}]$ denote the true regression function. Assume regression with squared loss, fixed partition trees $\{T_i\}_{i=1}^d$ (as learned by Step 2) and $\mathbb{E}[Y^2] < \infty$. Let $\mathcal{H}(\{T_i\})$ denote the fixed-tree CALM class (Appendix A.3.1), and assume $\mathcal{H}(\{T_i\}) \subset L_2(P_X)$ is nonempty, closed, and convex. Then:*

1. Optimality*: Any $s^\star \in \arg\min_{s\in\mathcal{H}(\{T_i\})} \mathbb{E}[(Y - s(\mathbf{X}))^2]$ is the $L_2(P_{\mathbf{X}})$-best approximation of m within $\mathcal{H}(\{T_i\})$.*
2. *Convergence: The idealized exact cyclic regional backfitting converges to an empirical risk minimizer over $\mathcal{H}(\{T_i\})$.*

3.3 Interpretability of a CALM

CALMs offer interpretability similar to GAMs, as both require inspecting d univariate plots; one per feature. CALMs are slightly more complex: unlike GAMs, each plot can contain up to $2^{d_{\max}}$ curves corresponding to a different region of the input space and with interaction-induced discontinuities marked by vertical lines. However, because regions are interpretable (specified by simple threshold conditions), these region-specific curves provide explanations which are suitable for model auditing and decision support. For example: *The effect of* `age` *on* `mortality_rate` *follows this curve when the patient is* `male` *and has a* `BMI` *above 30.*

Interpreting a CALM plot.

In Figure 2 each curve gives the contribution of x_1 to y in a specific region; the blue curve when $x_3 > 0$ and the orange curve when $x_3 \leq 0$. For example, at $x_1 = -0.5$, the contribution is approximately -0.2 (blue) or -0.75 (orange), depending on x_3. The plots also illustrate how altering x_1 to $x_1 \rightarrow x_1 + \Delta x$ impacts the prediction. Vertical dotted lines mark points of a hidden discontinuity which is due to x_1 participating as an interaction term for feature x_2. As shown in Figure 1, the effect of x_2 is conditioned by $x_1 \leq -0.4$, $-0.4 < x_1 \leq 0.4$ and $x_1 > 0.4$, therefore in Figure 2 we observe vertical lines in $x_1 \pm 0.4$. If a change in x_1 does not cross a vertical line, the change in the output (Δy) equals the curve difference (Δf_i). Crossing a line signifies a hidden jump, in the range $[\alpha, \beta]$, so $\Delta f_i + \alpha \leq \Delta y \leq \Delta f_i + \beta$. Arrows provide a fast understanding of the jump: $\uparrow$ means $\Delta y > \Delta f_i$, $\downarrow$ means $\Delta y < \Delta f_i$, $\updownarrow$ means it depends. Below, we outline three crucial interpretability properties, along with discussion about their satisfiability by GAM , GA^{2}M, and CALM. Complete proofs are provided in Appendix B.

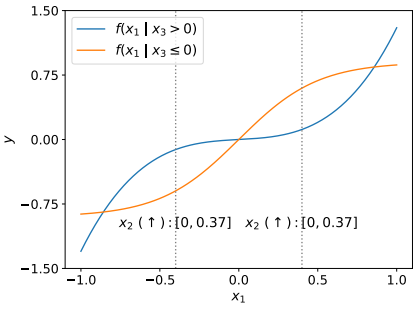
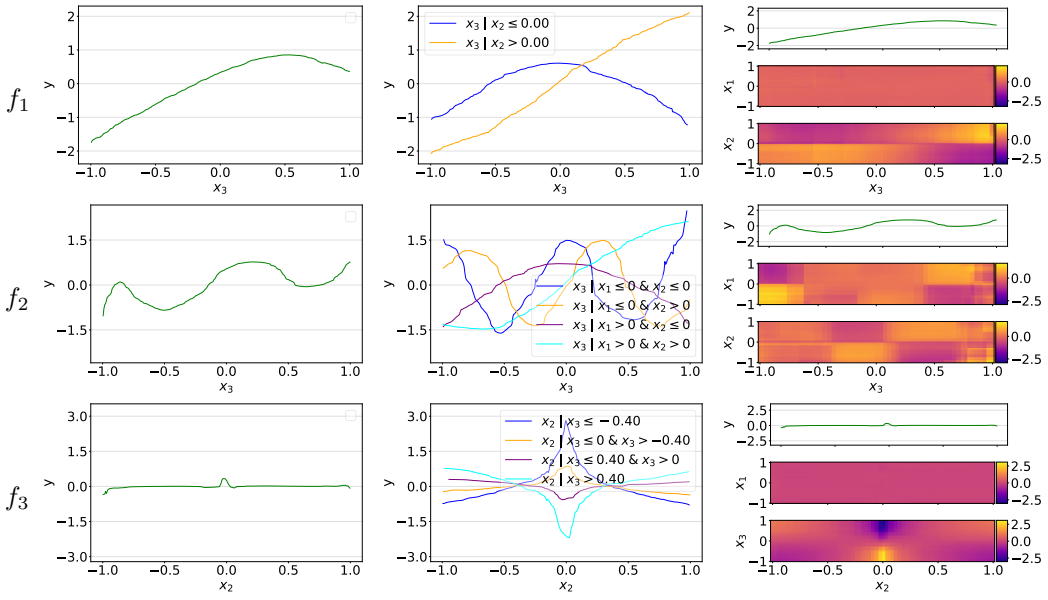


Figure 2: CALM plot for x_1, from a synthetic function where x_1 interacts with x_2 and x_3. Each curve shows the contribution of x_1 to y (P1) in a different region of the input space defined by $x_3 \lessgtr 0$. Vertical lines indicate interaction-induced discontinuities: at $x_1 \approx \pm 0.4$, changing x_1 causes x_2's region to switch, inducing a hidden jump of $[0, 0.37]$ that must be considered when assessing regional sensitivity (P2) or global properties (P3).

P1. Local Feature Contribution: What is the contribution of each feature to the prediction?
Formally: Given an input $\mathbf{x}$, how much does each x_i contribute to $\hat{y}(\mathbf{x})$?
The explanation is *local*—it concerns a specific input $\mathbf{x}$. In GAM , the contribution is $f_i(x_i)$. In GA^{2}M, the contribution is neither explicit nor unique; it must be inferred post-hoc (e.g., via SHAP or LIME), with each method relying on different assumptions and yielding different results. In CALM, the contribution is $f_i^{(r(\mathbf{x}_{-i}))}(x_i)$.

P2. Regional Feature Sensitivity: How does changing x_i change the prediction? *Formally:* Given x_i and $\Delta x > 0$, what is $\Delta\hat{y} = \hat{y}(\mathbf{x} + \mathbf{e}_i \Delta x) - \hat{y}(\mathbf{x})$, assuming only x_i is perturbed and $\mathbf{x}_{-i}$ is fixed? The explanation is *regional*—it characterizes the effect of x_i on a specific region $([x_i, x_i + \Delta x])$ *independently* of the values of other features. In GAM , the change is $\Delta\hat{y} = f_i(x_i + \Delta x) - f_i(x_i)$. In GA^{2}M, the change cannot be determined without knowing the values of all features that interact with x_i. In CALM, an exact answer is possible only when the perturbation does not cross a vertical line. In this case, the resulting change is a set of values $\Delta y := \{\Delta f_i^{(r)}\}_{r=1}^{R_i}$, one for each curve in the plot: $\Delta f^{(r)} = f_i^{(r)}(x_i + \Delta x) - f_i^{(r)}(x_i)$. If a crossing occurs, an exact answer is not attainable, however, for less precise questions, such as whether the Δx change in x_i will have a positive impact on y, an answer is still feasible (see **P3.** below).

P3. Global Feature Property: Is the model globally monotonic increasing with respect to x_i?
Formally: For all $\mathbf{x}$ and $\Delta x > 0$, is $\Delta\hat{y} = \hat{y}(\mathbf{x} + \mathbf{e}_i \Delta x) - \hat{y}(\mathbf{x}) > 0$?
The explanation is *global*—it assesses the monotonicity of x_i across the entire input space. In GAM , monotonicity is easily determined by the shape of $f_i(x_i)$. In GA^{2}M, verifying monotonicity requires a concurrent examination of the 1D shape of $f_i(x_i)$ along with all pairwise interactions f_{ij} along all j, an inspection which is infeasible for a human. In CALM, if no vertical lines are present, simply



(a) **GAM** (b) **CALM** (c) **GA^{2}M**

Figure 3: Explanatory plots for the three synthetic regression tasks: f_1 shows a two-region interaction; f_2 shows a four-region interaction; f_3 shows general interactions.

check whether all curves $f_i^{(r)}(x_i)$ for $r = 1, \ldots, R_i$ are monotonically increasing. If vertical lines exist, simply verify that all arrows are positive.

3.4 Efficiency

CALM's runtime is dominated by Step 2: fitting d binary trees costs $\mathcal{O}(d_{\max}\, d^2 N \log N)$, which reduces to $\mathcal{O}(d^2 N \log N)$ for the shallow trees we use—acceptable since d is on the order of tens for tabular data. Crucially, the local effects needed to score candidate splits are precomputed once and reused via indexed lookups, so each split evaluation avoids repeated black-box queries. Steps 1 and 3 are standard model fits (XGBoost and gradient boosting in our default setup) and are fast in practice. Overall, CALM fits most tabular datasets within a few seconds; full runtimes are reported in Appendix C.

4 Empirical Evaluation

The empirical evaluation of CALM includes three synthetic regression datasets and 25 public real-world tabular datasets—10 for classification and 15 for regression. Across all experiments, CALM is applied with its default configuration: Step 1 uses XGBoost as the black-box model. Step 2 uses PDP-based heterogeneity with $d_{\max} = 2$, $\epsilon = 0.2$, and K remains unconstrained. Step 3 uses standard gradient boosting. All predictive results are reported as mean $\pm$ standard deviation over standard 5-fold cross-validation. Full experimental details are in Appendix C.

Implementation details. We used well-known open-source implementations for all baseline models: scikit-learn (random forests), XGBoost, tensorflow (neural networks and NAM), interpretml (EBM, EB^{2}M), pygam (SPLINE), and official repositories for NODE-GA^{2}M[8] and GAMI-Net[43].

4.1 Synthetic example

We compare CALM against EBM (as the GAM baseline) and EB^{2}M (as the GA^{2}M baseline) on three synthetic datasets. Both EBM and EB^{2}M are used with their default parameters. Evaluation is based on R^2 performance. Each dataset contains 1000 samples with features drawn from $x_i \sim \mathcal{U}(-1, 1)$. Table 1 summarizes the accuracy of each approach.

Table 2: Classification accuracy vs. baselines (mean ± std. dev. over 5-fold CV). ✓/✗ denotes strictly higher/lower accuracy than CALM; = denotes exactly equal accuracy to CALM.

Dataset	BB	GAM		CALM	GA^{2}M		
	XGB	NAM	EBM		EB^{2}M	NODE	GAMI
Adult	$0.870_{\pm 0.002}$ =	$0.851_{\pm 0.002}$ ✗	$0.870_{\pm 0.002}$ =	$0.870_{\pm 0.002}$	$0.871_{\pm 0.003}$ ✓	$0.850_{\pm 0.002}$ ✗	$0.857_{\pm 0.003}$ ✗
COMPAS	$0.661_{\pm 0.007}$ ✗	$0.682_{\pm 0.016}$ ✗	$0.681_{\pm 0.012}$ ✗	$0.684_{\pm 0.013}$	$0.684_{\pm 0.011}$ =	$0.667_{\pm 0.013}$ ✗	$0.686_{\pm 0.013}$ ✓
HELOC	$0.717_{\pm 0.013}$ ✗	$0.723_{\pm 0.011}$ ✗	$0.728_{\pm 0.014}$ =	$0.728_{\pm 0.012}$	$0.730_{\pm 0.012}$ ✓	$0.715_{\pm 0.008}$ ✗	$0.725_{\pm 0.012}$ ✗
MIMIC2	$0.890_{\pm 0.001}$ ✓	$0.886_{\pm 0.001}$ =	$0.886_{\pm 0.003}$ =	$0.886_{\pm 0.003}$	$0.886_{\pm 0.003}$ =	$0.888_{\pm 0.002}$ ✓	$0.884_{\pm 0.003}$ ✗
Appendicitis	$0.868_{\pm 0.069}$ ✗	$0.848_{\pm 0.056}$ ✗	$0.877_{\pm 0.077}$ ✗	$0.878_{\pm 0.063}$	$0.869_{\pm 0.079}$ ✗	$0.849_{\pm 0.016}$ ✗	$0.764_{\pm 0.074}$ ✗
Phoneme	$0.898_{\pm 0.006}$ ✓	$0.808_{\pm 0.002}$ ✗	$0.821_{\pm 0.007}$ ✗	$0.861_{\pm 0.011}$	$0.863_{\pm 0.007}$ ✓	$0.869_{\pm 0.003}$ ✓	$0.876_{\pm 0.009}$ ✓
SPECTF	$0.862_{\pm 0.029}$ ✗	$0.839_{\pm 0.030}$ ✗	$0.894_{\pm 0.015}$ =	$0.894_{\pm 0.015}$	$0.882_{\pm 0.017}$ ✗	$0.820_{\pm 0.067}$ ✗	$0.874_{\pm 0.016}$ ✗
Magic	$0.885_{\pm 0.004}$ ✓	$0.850_{\pm 0.006}$ ✗	$0.857_{\pm 0.005}$ ✗	$0.864_{\pm 0.004}$	$0.872_{\pm 0.003}$ ✓	$0.876_{\pm 0.004}$ ✓	$0.874_{\pm 0.003}$ ✓
Bank	$0.908_{\pm 0.003}$ ✓	$0.901_{\pm 0.003}$ ✗	$0.902_{\pm 0.002}$ ✗	$0.905_{\pm 0.002}$	$0.909_{\pm 0.002}$ ✓	$0.903_{\pm 0.001}$ ✗	$0.908_{\pm 0.003}$ ✓
Churn	$0.958_{\pm 0.004}$ ✓	$0.885_{\pm 0.009}$ ✗	$0.886_{\pm 0.005}$ ✗	$0.946_{\pm 0.003}$	$0.957_{\pm 0.009}$ ✓	$0.954_{\pm 0.009}$ ✓	$0.952_{\pm 0.007}$ ✓
W/D/L	**5/1/4**	**0/1/9**	**0/4/6**		**6/2/2**	**4/0/6**	**5/0/5**

Case 1. We define f_1 as $y = x_1^2 + \log(|x_2|) + 2\sin\left(\frac{\pi}{2}x_3\right)\mathbf{1}_{x_2\geq 0} + 2\cos\left(\frac{\pi}{2}x_3\right)\mathbf{1}_{x_2<0}$. Since GAM cannot model the $x_2 - x_3$ interaction, it simplifies the $x_3 \to y$ effect by averaging the sine and cosine modes into one curve (Fig. 3a, f_1), resulting in $R^2 = 0.737$. Both GA^{2}M and CALM capture the interaction correctly, achieving near-perfect accuracy (0.974 and 0.995). However, their interpretability differs. GA^{2}M requires three views to interpret x_3's effect: (i) the main effect averaging the sine and cosine cases, (ii) a near-zero x_1–x_3 heatmap, and (iii) an x_2–x_3 heatmap capturing the sign-dependent interaction (Fig. 3c, f_1). In contrast, CALM simplifies the interpretation with two 1D plots and a clear separation of the regimes by conditioning on x_2's sign (Fig. 3b, f_1).

Case 2. We define f_2 as: $y = x_1^2 + \log(|x_2|) + 2\big[\sin\left(\frac{\pi}{2}x_3\right)\mathbf{1}_{x_1\geq 0,\, x_2\geq 0} + \cos\left(\frac{\pi}{2}x_3\right)\mathbf{1}_{x_1\geq 0,\, x_2<0} + \sin(2\pi x_3)\mathbf{1}_{x_1<0,\, x_2\geq 0} + \cos(2\pi x_3)\mathbf{1}_{x_1<0,\, x_2<0}\big]$. GAM (Fig. 3a, f_2) merges all four modes into a single curve, limiting its accuracy to $R^2 = 0.479$. GA^{2}M (Fig. 3c, f_2) captures partial structure through two 2D interactions (x_1–x_3 and x_2–x_3) but misses the full 3-way interaction, reaching $R^2 = 0.712$. Furthermore, the model's behavior is hard to grasp, as it requires integrating information from three distinct plots: the main effect and two interaction heatmaps. CALM (Fig. 3b, f_2) separates the four regimes into distinct 1D curves, achieving both high accuracy ($R^2 = 0.949$) and clear interpretability.

Table 1: Synthetic examples: R^2 comparison (mean ± std. dev. over 5-fold CV).

Dataset	GAM	GA^{2}M	CALM
Case 1	$0.737_{\pm 0.028}$	$0.974_{\pm 0.011}$	$0.995_{\pm 0.002}$
Case 2	$0.479_{\pm 0.052}$	$0.712_{\pm 0.036}$	$0.949_{\pm 0.024}$
Case 3	$0.527_{\pm 0.023}$	$0.961_{\pm 0.009}$	$0.975_{\pm 0.006}$

Case 3. We define f_3 as $y = x_1^2 + \log(|x_2|)\sin(\frac{\pi}{2}x_3)$, where the logarithmic effect of x_2, $\log|x_2|$, is modulated by the sinusoidal term $\sin(\frac{\pi}{2}x_3)$, resulting in a general interaction structure. GAM (Fig. 3a, f_3) fails to capture the interaction and instead fits a blurred sinusoidal curve, leading to low accuracy ($R^2 = 0.527$). GA^{2}M (Fig. 3c, f_3) achieves high accuracy ($R^2 = 0.961$) by capturing the interaction in the x_2–x_3 heatmap. Still, interpretation remains difficult, as understanding the $x_2 \to y$ effect requires integrating three views: the main effect plot and two interaction heatmaps. CALM (Fig. 3b, f_3) cannot fully express the continuous x_2–x_3 interaction, but approximates it by partitioning x_3 and assigning each segment an average logarithmic response. While simplified, this yields a high accuracy ($R^2 = 0.975$) and a concise, transparent explanation that remains *close* to the underlying behavior.

4.2 Evaluation on Real Datasets

On each real dataset, we evaluate CALM against an XGBoost baseline (with neural networks and random forests included in the appendix), two GAM models (NAM and EBM, as well as SPLINE included in the appendix), and three GA^{2}M models: EB^{2}M (EBM with pairwise interactions enabled), NODE-GA^{2}M, and GAMI-Net. Evaluation is based on accuracy for classification tasks and RMSE for regression datasets. In the main tables, CALM uses XGBoost as the Step-1 reference model and EBM as the regional GAM; full results across all teacher × regional-GAM combinations are in Appendix C.

Classification Results. On classification datasets (Table 2), CALM: (i) outperforms both GAM baselines in almost all cases (9/10 over NAM; 6/10 over EBM with 4 draws) (ii) is competitive

with the three GA²M models: CALM matches or beats EB²M on 4/10, NODE-GA²M on 6/10, and GAMI-Net on 5/10 datasets, and (iii) matches or beats the XGBoost black-box on 5/10 datasets.

Table 3: Regression RMSE vs. baselines (mean ± std. dev. over 5-fold CV). ✓/✗ denotes strictly lower/higher RMSE than CALM; = denotes exactly equal RMSE to CALM.

Dataset	BB	GAM		CALM	GA^{2}M		
	XGB	NAM	EBM		EB^{2}M	NODE	GAMI
Bike Sharing	$39.35_{\pm 1.38}$ ✓	$101.97_{\pm 1.31}$ ✗	$100.21_{\pm 1.01}$ ✗	$55.67_{\pm 1.22}$	$54.80_{\pm 0.96}$ ✓	$54.47_{\pm 1.58}$ ✓	$53.44_{\pm 1.90}$ ✓
California Housing	$0.45_{\pm 0.01}$ ✓	$0.61_{\pm 0.01}$ ✗	$0.55_{\pm 0.01}$ ✗	$0.51_{\pm 0.01}$	$0.49_{\pm 0.01}$ ✓	$0.50_{\pm 0.01}$ ✓	$0.51_{\pm 0.04}$ =
Parkinsons Motor	$1.44_{\pm 0.09}$ ✓	$6.11_{\pm 0.16}$ ✗	$4.20_{\pm 0.09}$ ✗	$2.24_{\pm 0.13}$	$2.35_{\pm 0.06}$ ✗	$3.49_{\pm 0.31}$ ✗	$2.76_{\pm 0.31}$ ✗
Parkinsons Total	$1.86_{\pm 0.08}$ ✓	$7.90_{\pm 0.10}$ ✗	$4.85_{\pm 0.11}$ ✗	$2.97_{\pm 0.09}$	$2.77_{\pm 0.06}$ ✓	$4.60_{\pm 0.53}$ ✗	$3.81_{\pm 0.66}$ ✗
Seoul Bike	$209.6_{\pm 3.47}$ ✓	$320.2_{\pm 4.38}$ ✗	$303.7_{\pm 3.86}$ ✗	$238.9_{\pm 1.64}$	$235.2_{\pm 1.58}$ ✓	$231.0_{\pm 4.23}$ ✓	$245.82_{\pm 8.45}$ ✗
Wine	$0.62_{\pm 0.01}$ ✓	$0.72_{\pm 0.01}$ ✗	$0.70_{\pm 0.01}$ ✗	$0.69_{\pm 0.02}$	$0.68_{\pm 0.01}$ ✓	$0.69_{\pm 0.01}$ =	$0.71_{\pm 0.00}$ ✗
Energy	$67.97_{\pm 2.58}$ ✓	$89.80_{\pm 2.67}$ ✗	$85.23_{\pm 2.49}$ ✗	$83.09_{\pm 1.97}$	$77.33_{\pm 2.46}$ ✓	$82.18_{\pm 2.53}$ ✓	$180.1_{\pm 84.09}$ ✗
CCPP	$3.09_{\pm 0.09}$ ✓	$4.21_{\pm 0.05}$ ✗	$3.44_{\pm 0.08}$ ✗	$3.42_{\pm 0.07}$	$3.28_{\pm 0.07}$ ✓	$3.97_{\pm 0.07}$ ✗	$3.92_{\pm 0.07}$ ✗
Electrical	$0.04_{\pm 0.02}$ ✗	$0.04_{\pm 0.01}$ ✗	$0.02_{\pm 0.01}$ =	$0.02_{\pm 0.01}$	$0.02_{\pm 0.01}$ =	$0.01_{\pm 0.01}$ ✓	$0.02_{\pm 0.01}$ =
Elevators	$2.0\times10^{-3}{\pm 0}$ =	$2.0\times10^{-3}{\pm 0}$ =	$2.0\times10^{-3}{\pm 0}$ =	$2.0\times10^{-3}{\pm 0}$	$2.0\times10^{-3}{\pm 0}$ =	$2.0\times10^{-3}{\pm 0}$ =	$2.0\times10^{-3}{\pm 0}$ =
No2	$0.47_{\pm 0.03}$ ✓	$0.50_{\pm 0.02}$ ✗	$0.49_{\pm 0.03}$ =	$0.49_{\pm 0.04}$	$0.47_{\pm 0.03}$ ✓	$0.52_{\pm 0.02}$ ✗	$0.50_{\pm 0.03}$ ✗
Sensory	$0.51_{\pm 0.03}$ ✗	$0.48_{\pm 0.01}$ ✗	$0.48_{\pm 0.01}$ ✗	$0.45_{\pm 0.02}$	$0.45_{\pm 0.02}$ =	$0.49_{\pm 0.01}$ ✗	$0.46_{\pm 0.03}$ ✗
Airfoil	$1.54_{\pm 0.10}$ ✓	$4.80_{\pm 0.20}$ ✗	$4.57_{\pm 0.15}$ ✗	$2.50_{\pm 0.15}$	$2.17_{\pm 0.09}$ ✓	$2.13_{\pm 0.11}$ ✓	$4.79_{\pm 0.21}$ ✗
Skill Craft	$0.94_{\pm 0.02}$ ✗	$0.93_{\pm 0.03}$ ✗	$0.90_{\pm 0.02}$ =	$0.90_{\pm 0.02}$	$0.90_{\pm 0.03}$ =	$0.91_{\pm 0.02}$ ✗	$1.38_{\pm 0.80}$ ✗
Ailerons	$2.0\times10^{-4}{\pm 0}$ =	$2.0\times10^{-4}{\pm 0}$ =	$2.0\times10^{-4}{\pm 0}$ =	$2.0\times10^{-4}{\pm 0}$	$2.0\times10^{-4}{\pm 0}$ =	$2.0\times10^{-4}{\pm 0}$ =	$2.0\times10^{-4}{\pm 0}$ =
W/D/L	**10/2/3**	**0/2/13**	**0/5/10**		**9/5/1**	**6/3/6**	**1/4/10**

Regression Results. On regression datasets (Table 3), CALM: (i) outperforms both GAM baselines in the large majority of cases (13/15 over NAM; 10/15 over EBM); (ii) matches NODE-GA^{2}M (6W/3D/6L) and outperforms GAMI-Net (10W/4D/1L), while EB^{2}M achieves lower RMSE in 9/15 cases; (iii) matches or beats XGBoost on 5/15 datasets. CALM thus achieves clear improvements over GAMs while maintaining 1D shape functions per region; the gap relative to EB^{2}M and XGBoost reflects the expected cost of stronger interpretability constraints.

Model Complexity. In addition to its strong predictive performance, CALM achieves its results using remarkably few pairwise feature interactions, 6.2 on average for classification and 15.1 for regression. On classification tasks, this is fewer than EB^{2}M (15.2), GAMI-Net (17.6), and NODE-GA^{2}M (97.7). On regression tasks, CALM remains sparser than GAMI-Net (16.3) and NODE-GA^{2}M (87.3), and close to EB^{2}M (14.1), as shown in Tables 6 and 7 of Appendix C. While most GA^{2}M methods allow explicit control over the number of interactions, we apply them using their default configurations. In contrast, CALM does not impose an interaction cap, but still discovers a sparse structure through region-based conditioning.

Runtime. On classification tasks, CALM is slightly slower than EBM and EB^{2}M, but noticeably faster than the other two GA^{2}M models (GAMI-Net and NODE-GA^{2}M), and even faster than NAM, despite the fact that NAM does not model interactions. In regression tasks, CALM exhibits a runtime similar to NAM and EB^{2}M, and remains significantly faster than the remaining GA^{2}M baselines. Average runtimes across datasets are reported in Tables 8 and 9 (Appendix C).

Human Interpretability. A within-subjects user study ($N = 25$) showed that CALM explanations are significantly easier to use than heatmap-based GA^{2}M representations: participants achieved 54% vs. 34% accuracy overall ($p < 0.05$), with the largest gain on prediction-change tasks (40% vs. 12%). Full details are in Appendix D.

5 Conclusion

We introduced CALM, a novel class of interpretable-by-design models that bridge the gap between the interpretability of GAMs and the accuracy of GA^{2}Ms. CALMs model feature interactions using *conditional feature effects*, a set of univariate shape functions per feature, conditioned on its interacting features. Extensive evaluation shows that conditional effects often match the accuracy of GA^{2}Ms, without resorting to 3D plots or heatmaps. CALM's main limitation is that interpretability may harden as the number of interactions increases. While each feature is associated with up to up to $2^{d_{\max}} = 4$ shape functions and, in practice, CALM activates an average of only 6.2 interactions in total (Table 6), extensive dependencies with other features can result in plots with numerous vertical dashed lines, making interpretation more difficult.

References

[1] Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey E Hinton. Neural additive models: Interpretable machine learning with neural nets. In *Advances in Neural Information Processing Systems*, volume 34, pages 4699–4711, 2021.

[2] Elaine Angelino, Nicholas Larus-Stone, Daniel Alabi, Margo Seltzer, and Cynthia Rudin. Learning certifiably optimal rule lists for categorical data, 2018.

[3] Jacob Bien and Robert Tibshirani. Prototype selection for interpretable classification. *The Annals of Applied Statistics*, 5(4), 2011.

[4] L. Breiman, J. Friedman, R.A. Olshen, and C.J. Stone. *Classification and Regression Trees*. Chapman and Hall/CRC, 1984.

[5] Luis M. Candanedo. Appliances energy prediction dataset, 2017.

[6] Luis M. Candanedo, Véronique Feldheim, and Dominique Deramaix. Data driven prediction models of energy use of appliances in a low-energy house. *Energy and Buildings*, 140:81–97, 2017.

[7] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1721–1730, 2015.

[8] Chun-Hao Chang, Rich Caruana, and Anna Goldenberg. Node-gam: Neural generalized additive model for interpretable deep learning. In *International Conference on Learning Representations*, 2022.

[9] Chaofan Chen, Oscar Li, Chaofan Tao, Alina Jade Barnett, Jonathan Su, and Cynthia Rudin. This looks like that: Deep learning for interpretable image recognition. In *NeurIPS*, 2019.

[10] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*, 2017.

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

[12] Vasilis Gkolemis, Theodore Dalamagas, and Christos Diou. Dale: Differential accumulated local effects for efficient and accurate global explanations. In *Asian Conference on Machine Learning*, pages 375–390. PMLR, 2023.

[13] Vasilis Gkolemis, Theodore Dalamagas, Eirini Ntoutsi, and Christos Diou. Rhale: robust and heterogeneity-aware accumulated local effects. In *ECAI 2023*, pages 859–866. IOS Press, 2023.

[14] Vasilis Gkolemis, Anargiros Tzerefos, Theodore Dalamagas, Eirini Ntoutsi, and Christos Diou. Regionally additive models: Explainable-by-design models minimizing feature interactions. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 433–447. Springer, 2023.

[15] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A survey of methods for explaining black box models. *ACM Computing Surveys*, 51(5):1–42, 2018.

[16] Trevor Hastie and Robert Tibshirani. *Generalized Additive Models*. Chapman and Hall/CRC, 1990.

[17] Julia Herbinger, Bernd Bischl, and Giuseppe Casalicchio. Repid: Regional effect plots with implicit interaction detection. In Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera, editors, *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pages 10209–10233. PMLR, March 2022.

[18] Julia Herbinger, Susanne Dandl, Fiona K Ewald, Sofia Loibl, and Giuseppe Casalicchio. Leveraging model-based trees as interpretable surrogate models for model distillation. In *European Conference on Artificial Intelligence*, pages 232–249. Springer, 2023.

[19] Julia Herbinger, Marvin N Wright, Thomas Nagler, Bernd Bischl, and Giuseppe Casalicchio. Decomposing global feature effects based on feature interactions. *Journal of Machine Learning Research*, 25(381):1–65, 2024.

[20] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. In *NeurIPS Deep Learning and Representation Learning Workshop*, 2015.

[21] Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848*, 2022.

[22] Linwei Hu, Jie Chen, Vijayan N Nair, and Agus Sudjianto. Surrogate locally-interpretable models with supervised machine learning algorithms. *arXiv preprint arXiv:2007.14528*, 2020.

[23] Shibal Ibrahim, Gabriel Afriat, Kayhan Behdin, and Rahul Mazumder. GRAND-SLAMIN' Interpretable Additive Modeling with Structural Constraints. *Advances in Neural Information Processing Systems*, 36:61158–61186, December 2023.

[24] Markelle Kelly, Rachel Longjohn, and Kolby Nottingham. The uci machine learning repository, 2025.

[25] Mathias Kraus, Daniel Tschernutter, Sven Weinzierl, and Patrick Zschech. Interpretable generalized additive neural networks. *European Journal of Operational Research*, 317(2):303–316, 2024.

[26] Benjamin Letham, Cynthia Rudin, Tyler H. McCormick, and David Madigan. Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. *The Annals of Applied Statistics*, 9(3), 2015.

[27] Zachary C Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

[28] Yin Lou, Rich Caruana, and Johannes Gehrke. Intelligible models for classification and regression. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 150–158, 2012.

[29] Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. Accurate intelligible models with pairwise interactions. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 623–631, 2013.

[30] Christoph Molnar. *Interpretable Machine Learning*. 2020.

[31] W James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. Interpretable machine learning: definitions, methods, and applications. *arXiv preprint arXiv:1901.04592*, 2019.

[32] Averkiy Oliabev. Home equity line of credit (HELOC), 2018.

[33] R. Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics and Probability Letters*, 33:291–297, 1997.

[34] ProPublica. Compas data and analysis for 'machine bias', 2016.

[35] Filip Radenovic, Abhimanyu Dubey, and Dhruv Mahajan. Neural basis models for interpretability. *Advances in Neural Information Processing Systems*, 35:8414–8426, 2022.

[36] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1135–1144. ACM, 2016.

[37] Joseph D Romano, Trang T Le, William La Cava, John T Gregg, Daniel J Goldberg, Praneel Chakraborty, Natasha L Ray, Daniel Himmelstein, Weixuan Fu, and Jason H Moore. Pmlb v1.0: an open source dataset collection for benchmarking machine learning methods. *arXiv preprint arXiv:2012.00058v2*, 2021.

[38] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.

[39] Mohammed Saeed, Mauricio Villarroel, Andrew T. Reisner, Gari Clifford, Li-Wei Lehman, George Moody, Thomas Heldt, Tin H. Kyaw, Benjamin Moody, and Roger G. Mark. Multiparameter intelligent monitoring in intensive care ii (mimic-ii): A public-access intensive care unit database. *Critical Care Medicine*, 39(5):952–960, 2011.

[40] Sarah Tan, Rich Caruana, Giles Hooker, and Yin Lou. Distill-and-compare: Auditing black-box models using transparent model distillation. In *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society*, pages 303–310, 2018.

[41] Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luis Torgo. Openml: Networked science in machine learning. *SIGKDD Explorations*, 15(2):49–60, 2013.

[42] Simon N Wood. *Generalized additive models: an introduction with R*. CRC press, 2017.

[43] Zebin Yang, Aijun Zhang, and Agus Sudjianto. Gami-net: An explainable neural network based on generalized additive models with structured interactions. *arXiv preprint arXiv:2003.07132*, 2020.

A Conditionally Additive Local Models (CALMs)

In the main paper (Algorithm 1), we summarize the procedure of fitting a Conditionally Additive Local Model (CALM):

$$f_{\texttt{CALM}}(\mathbf{x}) = g^{-1}\left(\beta_0 + \sum_{i=1}^{d} f_i^{r_i(\mathbf{x}_{-i})}(x_i)\right),$$

as defined in Eq.(1), to a dataset $\mathcal{D} = \{\mathbf{x}^{(i)}, y^{(i)}\}_{i=1}^{N}$ in three main steps: (i) fit a high-capacity reference model $f_{\texttt{ref}}$ on $\mathcal{D}$, (ii) fit a partition tree T_i using an interaction-related heterogeneity measure H_i for each feature $i = 1, \ldots, d$, and (iii) estimate region-specific effects $\{f_i^{(r)}\}_{r=1}^{R_i}$ for each feature $i = 1, \ldots, d$. We provide additional details for each step below.

A.1 Step 1: Fit a high-capacity reference model $f_{\texttt{ref}}$ on $\mathcal{D}$

The initial stage involves fitting a high-capacity predictive model, denoted as $f_{\texttt{ref}} : \mathbb{R}^d \to \mathbb{R}$ to dataset $\mathcal{D} = \{(\mathbf{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$. The reference model serves as an accurate functional approximation of the underlying relationship between the input features $\mathbf{x}$ and the response variable y. The choice of $f_{\texttt{ref}}$ is flexible; any sufficiently expressive and accurate model can be employed. Commonly used options include gradient-boosted decision trees (e.g., XGBoost, which we adopt by default), deep neural networks, random forests, or more recent architectures such as TabPFN [21].

A.2 Step 2: Fit a partition tree T_i using heterogeneity H_i for each feature

We first define a suitable heterogeneity measure H_i (see Section A.2.1), and then explore the relationship between heterogeneity and CALM's error in approximating $f_{\texttt{ref}}$. Motivated by these results, we fit a partition tree T_i for each feature i (see Section A.2.4). Finally, if the user specifies a constraint K on the maximum number of interactions, we apply a pruning step to retain the K most significant interactions (see Section A.2.5).

A.2.1 Heterogeneity Measures H_i

Interaction-related heterogeneity H_i quantifies the extent to which a feature x_i interacts with all other features x_j, where $j \in \{1, \ldots, d\} \setminus \{i\}$. The computation of H_i is based on the variance of local effects: $h(x_i, \mathbf{x}_{-i}^{(j)})$.

Local Effect. We define $h(x_i, \mathbf{x}_{-i}^{(j)})$ as the local effect of feature x_i when applied to the j-th background observation $\mathbf{x}_{-i}^{(j)}$. That is, if we take the j-th observation and substitute the i-th feature with the value x_i, the resulting change (local effect) in the model output is captured by $h(x_i, \mathbf{x}_{-i}^{(j)})$. The computation of $h(\cdot)$ relies on two components: (i) a reference teacher model $f_{\texttt{ref}}$ and (ii) a background dataset $\{\mathbf{x}^{(i)}\}_{i=1}^{N}$ containing input features only. The specific form of $h(\cdot)$ depends on the chosen feature effect method. Several methods can be used, including Partial Dependence Plot (PDP), SHAP-Dependence Plot (SHAP-DP) and Robust and Heterogeneity-aware ALE (RHALE).

Below, we provide the definition of h in the case of PDP, which we adopt as the default method in our experiments. Let $f_{\texttt{ref}}$ be the reference model and $\{\mathbf{x}^{(j)}\}_{j=1}^{N}$ the background dataset. The PDP-based heterogeneity is defined as:

$$h\big(x_i,\, \mathbf{x}_{-i}^{(j)}\big) = f_{\texttt{ref}}\big(x_i,\, \mathbf{x}_{-i}^{(j)}\big) - c(\mathbf{x}_{-i}) \tag{6}$$

where $c(\mathbf{x}_{-i}^{(j)}) = \mathbb{E}_{X_i}\left[f_{\texttt{ref}}\big(X_i,\, \mathbf{x}_{-i}^{(j)}\big)\right]$ is a centering constant. For details on alternative approaches, we refer the reader to [19, 17].

Point-wise heterogeneity $H_i(x_i)$ and Heterogeneity H_i. The pointwise heterogeneity is then defined as

$$H_i(x_i) = \mathbb{E}_{\mathbf{X}_{-i}}\left[\left(h(x_i, \mathbf{X}_{-i}) - \mu_i(x_i)\right)^2\right] \tag{7}$$

where $\mu_i(x_i) = \mathbb{E}_{\mathbf{X}_{-i}}[h(x_i, \mathbf{X}_{-i})]$ is the mean local effect. Inside a region, this heterogeneity takes the form of Eq. (4). Feature-level heterogeneity is

$$H_i = \mathbb{E}_{X_i}\left[H_i(X_i)\right] = \mathbb{E}_{X_i}\left[\mathrm{Var}_{\mathbf{X}_{-i}}[h(X_i, \mathbf{X}_{-i})]\right] \tag{8}$$

Inside a region this definition takes the form of Eq. (5).

A more formal motivation for the use of heterogeneity in CALM comes from the proposition of the following section.

A.2.2 Heterogeneity and CALM approximation error

Proposition A.1. *Let $\mathcal{E} = \mathbb{E}_{\mathbf{X}}\left[\left(f_{ref}(\mathbf{X}) - f_{CALM}(\mathbf{X})\right)^2\right]$ be the mean squared error of the CALM approximation of f_{ref}. Assume that $f_{ref} : \mathcal{X} \to \mathbb{R}$ admits a multivariate regionally additive approximation*

$$f_{ref}(\mathbf{x}) \approx \beta_0 + \sum_{i=1}^{d} h_i^{r_i(\mathbf{x}_{-i})}(\mathbf{x})$$

such that each $h_i^{r_i(\mathbf{x}_{-i})}$ captures the contribution of the i-th feature to the function's output (including interactions). Also, assume that there is no error in fitting of the univariate CALM shape functions. For clarity of exposition, the proof treats this approximation as an exact equality; the bound (9) *holds up to the residual approximation variance. Then,*

$$\min_{f_{CALM}} \mathcal{E} \le d \sum_{i=1}^{d} \left(\sum_{r=1}^{R_i} P_{\mathcal{R}_i^{(r)}} H_i^{\mathcal{R}_i^{(r)}} \right) \tag{9}$$

where region $\mathcal{R}_i^{(r)}$ is the r-th region of the i-th feature, $P_{\mathcal{R}_i^{(r)}}$ is the probability mass of the region in the data and $H_i^{\mathcal{R}_i^{(r)}}$ is the heterogeneity of $h_i^{\mathcal{R}_i^{(r)}}(\mathbf{x})$.

Proof. For the approximation error we have

$$\mathcal{E} = \mathbb{E}_{\mathbf{X}}\left[\left(f_{\texttt{ref}}(\mathbf{X}) - f_{\texttt{CALM}}(\mathbf{X})\right)^2\right] = \mathbb{E}_{\mathbf{X}}\left[\left(\left(\beta_0 + \sum_{i=1}^{d} h_i^{r_i(\mathbf{x}_{-i})}(\mathbf{x})\right) - \left(\beta_0 + \sum_{i=1}^{d} f_i^{r_i(\mathbf{x}_{-i})}(x_i)\right)\right)^2\right]$$

$$= \mathbb{E}_{\mathbf{X}}\left[\left(\sum_{i=1}^{d}\left(h_i^{r_i(\mathbf{x}_{-i})}(\mathbf{x}) - f_i^{r_i(\mathbf{x}_{-i})}(x_i)\right)\right)^2\right] \le d \sum_{i=1}^{d} \mathbb{E}_{\mathbf{X}}\left[\left(h_i^{r_i(\mathbf{x}_{-i})}(\mathbf{x}) - f_i^{r_i(\mathbf{x}_{-i})}(x_i)\right)^2\right]$$

where the last step results from the Jensen's inequality. We therefore have

$$\mathcal{E} \leq d \sum_{i=1}^{d} \mathcal{E}_i^{r_i(\mathbf{x}_{-i})} \tag{10}$$

Next, we show that the minimum error achieved for each region is equal to the heterogeneity of the local effects. For any measurable function $g_i(x_i)$ and region $\mathcal{R}$, decompose, using $\mu_i^{\mathcal{R}}(x_i) = \mathbb{E}_{\mathbf{X}_{-i}|\mathbf{X}_{-i}\in\mathcal{R}}\left[h_i^{\mathcal{R}}(x_i, \mathbf{X}_{-i})\right]$:

$$h_i(X_i, \mathbf{X}_{-i}) - g_i(X_i) = \left(h_i(X_i, \mathbf{X}_{-i}) - \mu_i^{\mathcal{R}}(X_i)\right) + \left(\mu_i^{\mathcal{R}}(X_i) - g_i(X_i)\right)$$

Taking expectations inside $\mathcal{R}$ and squaring:

$$\begin{aligned}
&\mathbb{E}_{X_i,\mathbf{X}_{-i}|\mathbf{X}_{-i}\in\mathcal{R}}\left[\left(h_i^{\mathcal{R}}(X_i, \mathbf{X}_{-i}) - g_i(X_i)\right)^2\right] \\
&= \mathbb{E}_{X_i,\mathbf{X}_{-i}|\mathbf{X}_{-i}\in\mathcal{R}}\left[\left(h_i^{\mathcal{R}}(X_i, \mathbf{X}_{-i}) - \mu_i^{\mathcal{R}}(X_i)\right)^2\right] \\
&\quad + \mathbb{E}_{X_i}\left[\left(\mu_i^{\mathcal{R}}(X_i) - g_i(X_i)\right)^2\right] \\
&\quad + 2\mathbb{E}_{X_i}\left[\mathbb{E}_{\mathbf{X}_{-i}|\mathbf{X}_{-i}\in\mathcal{R}}\left[h_i^{\mathcal{R}}(X_i, \mathbf{X}_{-i}) - \mu_i^{\mathcal{R}}(X_i)\right]\left(\mu_i^{\mathcal{R}}(X_i) - g_i(X_i)\right)\right]
\end{aligned}$$

The cross-term equals zero because:

$$\mathbb{E}_{\mathbf{X}_{-i}|\mathbf{X}_{-i}\in\mathcal{R}}\left[h_i^{\mathcal{R}}(x_i, \mathbf{X}_{-i}) - \mu_i^{\mathcal{R}}(x_i)\right] = \mu_i^{\mathcal{R}}(x_i) - \mu_i^{\mathcal{R}}(x_i) = 0$$

by definition of $\mu_i^{\mathcal{R}}$. Therefore:

$$\mathbb{E}_{X_i,\mathbf{X}_{-i}|\mathbf{X}_{-i}\in\mathcal{R}}\left[\left(h_i^{\mathcal{R}}(X_i, \mathbf{X}_{-i}) - g_i(X_i)\right)^2\right] = \mathcal{E}_i^{\mathcal{R}} + \mathbb{E}_{X_i}\left[\left(\mu_i^{\mathcal{R}}(X_i) - g_i(X_i)\right)^2\right] \geq \mathcal{E}_i^{\mathcal{R}}$$

with equality if and only if $g_i = \mu_i^{\mathcal{R}}$ almost surely. This indicates that within each region, $\mu_i^{\mathcal{R}}$ minimizes the approximation error.

Moreover, from the definition of pointwise heterogeneity:

$$H_i^{\mathcal{R}}(x_i) := \mathbb{E}_{\mathbf{X}_{-i}|\mathbf{X}_{-i}\in\mathcal{R}}\left[\left(h_i^{\mathcal{R}}(x_i, \mathbf{X}_{-i}) - \mu_i^{\mathcal{R}}(x_i)\right)^2\right]$$

Taking the expectation over X_i:

$$H_i^{\mathcal{R}} = \mathbb{E}_{X_i}\left[H_i^{\mathcal{R}}(X_i)\right] = \mathbb{E}_{X_i}\left[\mathbb{E}_{\mathbf{X}_{-i}|\mathbf{X}_{-i}\in\mathcal{R}}\left[\left(h_i^{\mathcal{R}}(X_i, \mathbf{X}_{-i}) - \mu_i^{\mathcal{R}}(X_i)\right)^2\right]\right]$$

which gives

$$H_i^{\mathcal{R}} = \mathbb{E}_{X_i,\mathbf{X}_{-i}|\mathbf{X}_{-i}\in\mathcal{R}}\left[\left(h_i^{\mathcal{R}}(X_i, \mathbf{X}_{-i}) - \mu_i^{\mathcal{R}}(X_i)\right)^2\right] = \min \mathcal{E}_i^{\mathcal{R}}$$

i.e., heterogeneity is the minimum approximation error.

Since heterogeneity minimizes the error for each region we can estimate the feature-level error across regions as

$$\min \mathcal{E}_i = \sum_{r=1}^{R_i} P_{\mathcal{R}_i^{(r)}} H_i^{\mathcal{R}_i^{(r)}}$$

where $P_{\mathcal{R}_i^{(r)}}$ is the probability mass of region $\mathcal{R}_i^{(r)}$.

Using this result with Eq. (10), we have

$$\min_{f_{\texttt{CALM}}} \mathcal{E} \leq d \sum_{i=1}^{d} \left(\sum_{r=1}^{R_i} P_{\mathcal{R}_i^{(r)}} H_i^{\mathcal{R}_i^{(r)}} \right)$$

□

Remark A.2. The switch from the conditional expectation $\mathbb{E}_{X_i|\mathbf{X}_{-i}\in\mathcal{R}}$ to the marginal $\mathbb{E}_{X_i}$ in the second term of the decomposition above implicitly assumes $X_i \perp \mathbf{X}_{-i} \mid \mathbf{X}_{-i} \in \mathcal{R}$ within each region. This is the standard assumption underlying PDP-style decompositions and is shared by all feature-effect methods that use the marginal distribution of X_i.

This result links heterogeneity with the estimation error of the local effect inside a region and motivates the CALM approach for region splitting. The initial assumption about the additive approximation of $f_{\texttt{ref}}$ is a common approach followed by feature effect methods (where the effect of each feature is computed independently). Moreover, the selected local effect function and heterogeneity are solely used for identifying the splitting regions and not as estimators of $f_{\texttt{ref}}$. Using this result, and especially Eq. (9), the following two sections present the CALM approach for using heterogeneity to define a partition for each feature.

A.2.3 Heterogeneity estimation

For heterogeneity estimation, denote with $\mathcal{I} \subseteq \{1, \ldots, N\}$ the index set of active background instances considered in the heterogeneity calculation i.e., those that reside in region $\mathcal{R}_i^{(r_i(\mathbf{x}_{-i}))}$ of the i-th instance. Then, the point-wise heterogeneity at feature value x_i over $\mathcal{I}$ can be estimated by

$$\hat{H}_i^{\mathcal{I}}(x_i) = \frac{1}{|\mathcal{I}|} \sum_{j \in \mathcal{I}} \Big(h(x_i, \mathbf{x}_{-i}^{(j)}) - \mu(x_i) \Big)^2, \text{ where } \mu(x_i) = \frac{1}{|\mathcal{I}|} \sum_{j \in \mathcal{I}} h(x_i, \mathbf{x}_{-i}^{(j)}) \tag{11}$$

Eq.(11) quantifies the strength of the interactions of the i-th feature, at position x_i, as the variance of the local effects at x_i. To obtain a global measure of these interactions, we average the pointwise heterogeneity over a grid of M values $\{\tilde{x}_i^{(m)}\}_{m=1}^M$ sampled from the domain of x_i:

$$\hat{H}_i^{\mathcal{I}} = \frac{1}{M} \sum_{m=1}^{M} H_i^{\mathcal{I}}(\tilde{x}_i^{(m)}). \tag{12}$$

A.2.4 Computing the partition tree T_i for each feature.

For each feature x_i, we construct a binary tree T_i of maximum depth $d_{\max}$. At each internal node of the tree, we evaluate candidate binary splits based on all features x_j for $j \in \{1, \ldots, d\} \setminus \{i\}$. For each candidate splitting feature x_j, we consider a fixed number T of candidate threshold values τ. These candidate thresholds are selected as T equally spaced values over the range of x_j in the training data, i.e., $[\min_k x_j^{(k)}, \max_k x_j^{(k)}]$, where $x_j^{(k)}$ denotes the value of feature x_j for the k-th training instance. Alternatively, users may choose to evaluate all unique observed values of x_j, i.e., $\{x_j^{(k)}\}_{k=1}^N$.

To determine the best split at each node, we compute the heterogeneity drop, which quantifies the decrease in interaction-related heterogeneity of the target feature x_i after performing the split. Specifically, for a candidate split, we partition the current set of instances $\mathcal{I}$ into left and right subsets $\mathcal{I}_L$ and $\mathcal{I}_R$, based on whether the splitting feature x_j falls below or above the threshold τ. The heterogeneity drop is defined as:

$$\Delta H_i^{=} \frac{H_i^{\mathcal{I}} - \left(\frac{|\mathcal{I}_L|}{|\mathcal{I}|} H_i^{\mathcal{I}_L} + \frac{|\mathcal{I}_R|}{|\mathcal{I}|} H_i^{\mathcal{I}_R} \right)}{H_i^{\mathcal{I}}} \tag{13}$$

where $H_i^{\mathcal{I}}$ denotes the interaction-related heterogeneity of feature x_i over the current set of instances $\mathcal{I}$; $\mathcal{I}_L$ and $\mathcal{I}_R$ are the subsets of $\mathcal{I}$ resulting from the candidate split; $H_i^{\mathcal{I}_L}$ and $H_i^{\mathcal{I}_R}$ are the heterogeneities of x_i computed on the left and right subsets, respectively; and $|\mathcal{I}_L|/|\mathcal{I}|$ and $|\mathcal{I}_R|/|\mathcal{I}|$ are the proportions of instances in each subset, which serve as weights in the weighted average.

This normalized metric enhances interpretability by allowing users to define a meaningful threshold, ϵ, for split acceptance. For instance, a user may require a candidate split to achieve a minimum heterogeneity reduction of $\epsilon = 0.2$ (representing a 20% drop) to be considered significant. In our experimental setup, we utilize a default threshold of $\epsilon = 0.2$.

The procedure for fitting the tree T_i is given in Algorithm 2.

Algorithm 2 Fitting a Partition Tree T_i for Feature x_i

Input: f_{ref}, $\mathcal{D} = \{(\mathbf{x}^{(k)}, y^{(k)})\}_{k=1}^{N}$; **Parameters:** max depth $d_{\max}$, threshold ϵ, grid size $M = 20$
Output: Binary tree T_i

function BUILDTREE(node ν, depth ℓ, active indices $\mathcal{I}$)
if $\ell = d_{\max}$ **then**
  return ν {Stop splitting}
end if
Compute $H_i^{\mathcal{I}}$ using Eq.(12)
Initialize: $\Delta H_{\max} \leftarrow 0$
for each feature $j \neq i$ **do**
  Determine candidate thresholds $\mathcal{T}_j$ {Default: Unique values if x_j categorical, else M-grid (default: $M = 20$)}
  for each $\tau \in \mathcal{T}_j$ **do**
    if x_j is numerical **then**
      $\mathcal{I}_L \leftarrow \{k \in \mathcal{I} : x_j^{(k)} \leq \tau\}, \mathcal{I}_R \leftarrow \mathcal{I} \setminus \mathcal{I}_L$
    else {x_j is categorical}
      $\mathcal{I}_L \leftarrow \{k \in \mathcal{I} : x_j^{(k)} = \tau\}, \mathcal{I}_R \leftarrow \mathcal{I} \setminus \mathcal{I}_L$
    end if
    Compute ΔH_i using Eq. (13)
    if $\Delta H_i > \Delta H_{\max}$ **then**
      $\Delta H_{\max} \leftarrow \Delta H_i$, Store (j, τ) as optimal split
    end if
  end for
end for
if $\Delta H_{\max} > \epsilon$ **then**
  Split node ν into ν_L, ν_R using optimal (j, τ)
  $\nu_L \leftarrow$ BUILDTREE($\nu_L, \ell + 1, \mathcal{I}_L$)
  $\nu_R \leftarrow$ BUILDTREE($\nu_R, \ell + 1, \mathcal{I}_R$)
end if
return ν
end function

A.2.5 Prune trees to keep top K interactions

If the user sets an upper threshold K on the number of interactions, we want to select the K most important splits. Given the trees T_i for $i = 1, \ldots, d$, we denote by ΔH_i^{ν} the heterogeneity reduction associated with node ν in tree T_i. We collect all such splits ΔH_i^{ν} across all features i and nodes ν, and then sort them in descending order. From this ordered list, we select the top K nodes that correspond to the largest heterogeneity decreases, while ensuring that no child node is retained without its parent node also being included. Based on this selection, we prune each tree T_i by removing nodes outside the retained set, thereby preserving only the most significant interactions.

A.3 Step 3: Estimating the region-specific effects $\{f_i(r)\}_{r=1}^{R_i}$ for each feature.

The original gradient boosting procedure for fitting a standard, global GAM, $f_{\texttt{GAM}}(\mathbf{x}) = g^{-1}\left(\beta_0 + \sum_i f_i(x_i)\right)$, follows the round-robin approach of [11]. At each boosting iteration, one univariate shape function f_i is updated by fitting to the current residuals over the entire dataset, thereby greedily reducing the overall loss. The procedure as describe by [28] is summarized in Algorithm 3.

To fit a CALM: $f_{\texttt{CALM}}(\mathbf{x}) = g^{-1}\left(\beta_0 + \sum_{i=1}^{d} f_i^{r_i(\mathbf{x}_{-i})}(x_i)\right)$ we customize the standard gradient boosting to accommodate for region-specific effects, i.e., we adapt this scheme so that each shape function $f_i^{(r)}$ is trained only on the subset of instances that lie within its assigned subregion $\mathcal{R}_i^{(r)}$.

Algorithm 3 Gradient Boosting for GAM

Initialize $f_i \leftarrow 0$ for all $i = 1, \ldots, d$
for $m = 1$ **to** M **do**
  for $i = 1$ **to** d **do**
    $\mathcal{E} \leftarrow \{(x_i^{(j)}, y^{(j)} - f_{\text{GAM}}(\mathbf{x}^{(j)})\}_{j=1}^{N}$ {Compute partial residuals}
    Learn shaping function $S : x_i \to \mathbb{R}$ using $\mathcal{E}$ {Fit shape function}
    $f_i \leftarrow f_i + \eta S$ {Update with learning rate η}
  end for
end for

Algorithm 4 Gradient Boosting for CALM

Initialize $\beta_0 \leftarrow \frac{1}{N}\sum_{j=1}^{N} y^{(j)}$; $f_i^{(r)} \leftarrow 0$ for all $i = 1, \ldots, d$ and $r = 1, \ldots, R_i$
for $m = 1$ **to** M **do**
  for $i = 1$ **to** d **do**
    for $r = 1$ **to** R_i **do**
      $\mathcal{E} \leftarrow \{(x_i^{(j)}, y^{(j)} - f_{\text{CALM}}(\mathbf{x}^{(j)})) : \mathbf{x}_{-i}^{(j)} \in \mathcal{R}_i^{(r)}\}$ {Filter by region $\mathcal{R}_i^{(r)}$}
      Learn shaping function $S : x_i \to \mathbb{R}$ using $\mathcal{E}$
      $f_i^{(r)} \leftarrow f_i^{(r)} + \eta S$ {Update regional shape function}
    end for
  end for
  Center: for all i, r, let $c_i^{(r)} \leftarrow \frac{1}{|\mathcal{R}_i^{(r)}|}\sum_{\mathbf{x}^{(j)} \in \mathcal{R}_i^{(r)}} f_i^{(r)}(x_i^{(j)})$; set $f_i^{(r)} \leftarrow f_i^{(r)} - c_i^{(r)}$ and $\beta_0 \leftarrow \beta_0 + \sum_i \sum_r \mathbb{P}(\mathcal{R}_i^{(r)})\, c_i^{(r)}$ {Enforce identifiability}
end for

This preserves the additive, boosting framework while enforcing that each $f_i^{(r)}$ captures only the behavior of x_i within its corresponding region. The procedure is summarized in Algorithm 4.

A.3.1 Theoretical analysis of Step 3

This section formalizes the claims made in Step 3 of the main text. We show that, under squared loss and fixed region-selection trees, (i) the population target of Step 3 is an $L_2(P_{\mathbf{X}})$-best approximation of the true regression function onto the CALM function class, and (ii) an idealized exact version of the algorithm converges to the corresponding empirical risk minimizer.

Consider the CALM score function

$$s(\mathbf{x}) = \beta_0 + \sum_{i=1}^{d} f_i^{(r_i(\mathbf{X}_{-i}))}(x_i), \tag{14}$$

where the region-selection functions $\{r_i\}$ (or equivalently the trees $\{T_i\}$) are treated as fixed. Let

$$m(\mathbf{x}) = \mathbb{E}[Y \mid \mathbf{X} = \mathbf{x}]$$

denote the true regression function. We assume regression with identity link and squared loss, and $\mathbb{E}[Y^2] < \infty$. We restrict attention to scores $s \in L_2(P_{\mathbf{X}})$ (hence $m \in L_2(P_{\mathbf{X}})$).

To ensure identifiability, we impose the centering condition

$$\mathbb{E}\Big[f_i^{(r)}(X_i) \,\Big|\, r_i(\mathbf{X}_{-i}) = r\Big] = 0 \quad \text{for all } i, r \text{ with } \mathbb{P}(r_i(\mathbf{X}_{-i}) = r) > 0. \tag{15}$$

This convention assigns all region-wise constants to the intercept β_0 and ensures uniqueness of the decomposition.

Let $\mathcal{H}(\{T_i\})$ denote the class of all CALM score functions of the form (14) satisfying (15), and assume $\mathcal{H}(\{T_i\}) \subset L_2(P_{\mathbf{X}})$ is a nonempty closed *convex* set.

A.3.2 Population target

Proposition A.3 (Population optimality)**.** *There exists at least one minimizer*

$$s^\star \in \arg\min_{s \in \mathcal{H}(\{T_i\})} \mathbb{E}[(Y - s(\mathbf{X}))^2],$$

and it satisfies

$$s^\star \in \arg\min_{s \in \mathcal{H}(\{T_i\})} \mathbb{E}[(m(\mathbf{X}) - s(\mathbf{X}))^2].$$

In particular,

$$\mathbb{E}[(Y - s^\star(\mathbf{X}))^2] - \mathbb{E}[(Y - m(\mathbf{X}))^2] = \mathbb{E}[(m(\mathbf{X}) - s^\star(\mathbf{X}))^2].$$

Proof. For any measurable function s,

$$Y - s(\mathbf{X}) = (Y - m(\mathbf{X})) + (m(\mathbf{X}) - s(\mathbf{X})).$$

Taking squares and expectations yields

$$\mathbb{E}[(Y - s(\mathbf{X}))^2] = \mathbb{E}[(Y - m(\mathbf{X}))^2] + \mathbb{E}[(m(\mathbf{X}) - s(\mathbf{X}))^2],$$

since $\mathbb{E}[Y - m(\mathbf{X}) \mid \mathbf{X}] = 0$ eliminates the cross-term. The first term does not depend on s, so minimizing prediction error over $\mathcal{H}(\{T_i\})$ is equivalent to minimizing $\mathbb{E}[(m(\mathbf{X}) - s(\mathbf{X}))^2]$. Existence of $s^\star$ follows since $\mathcal{H}(\{T_i\}) \subset L_2(P_\mathbf{X})$ is nonempty, closed, and convex. Condition (15) ensures uniqueness of the representation $(\beta_0, \{f_i^{(r)}\})$ for any given $s \in \mathcal{H}(\{T_i\})$, assuming no exact functional dependence (concurvity) among features within each region. □

Interpretation. Proposition A.3 shows that, with fixed regions, Step 3 computes an $L_2(P_\mathbf{X})$-best approximation of the true regression function m within the CALM function class. Any remaining error is therefore purely due to the structural limitations of the chosen regions and univariate components, not to the optimization procedure.

A.3.3 Empirical convergence of exact Step 3

Given i.i.d. data $\{(x^{(k)}, y^{(k)})\}_{k=1}^N$, define the empirical risk

$$\widehat{\mathcal{R}}_N(s) = \frac{1}{N} \sum_{k=1}^{N} (y^{(k)} - s(x^{(k)}))^2.$$

Proposition A.4 (Convergence of exact cyclic regional backfitting)**.** *Assume fixed trees and squared loss. Consider an idealized version of Step 3 that cyclically updates each $f_i^{(r)}$ by exactly minimizing $\widehat{\mathcal{R}}_N$ over that function using only samples with $r_i(\mathbf{x}_{-i}^{(k)}) = r$, followed by empirical centering implemented by shifting the region-wise sample mean from $f_i^{(r)}$ into the intercept β_0, so that fitted values (and hence $\widehat{\mathcal{R}}_N$) are unchanged. Then $\widehat{\mathcal{R}}_N$ is non-increasing along the iterates, and the procedure converges (in empirical L_2) to a limit $\hat{s} \in \arg\min_{s \in \mathcal{H}(\{T_i\})} \widehat{\mathcal{R}}_N(s)$. Moreover, the fitted-value vector $(\hat{s}(\mathbf{x}^{(1)}), \ldots, \hat{s}(\mathbf{x}^{(N)}))$ is unique.*

Proof. With fixed trees, the CALM score is linear in the collection of shape functions $\{f_i^{(r)}\}$ evaluated on the data, and $\widehat{\mathcal{R}}_N$ is a convex quadratic function of these parameters. On the finite sample, $\widehat{\mathcal{R}}_N$ depends on each $f_i^{(r)}$ only through the finite vector $\{f_i^{(r)}(x_i^{(k)}) : r_i(\mathbf{x}_{-i}^{(k)}) = r\}$, so the problem can be viewed as a finite-dimensional least-squares problem. Each regional update solves the exact least-squares problem for one block of parameters while holding the others fixed, and therefore cannot increase $\widehat{\mathcal{R}}_N$. Although regions overlap across features, all updates jointly optimize a single global objective. Standard results for cyclic block coordinate descent on convex quadratics imply convergence to a global minimizer, and the minimizing fitted values are unique by strict convexity of the quadratic loss in the fitted-value vector. The centering condition (15) ensures a unique decomposition into components. □

Interpretation. This result shows that Step 3 is not a collection of independent local fits. Instead, it performs a coupled optimization of a single additive predictor with region-gated components, analogous to classical backfitting for GAMs. Under exact updates, this procedure is guaranteed to converge to the best CALM fit for the given data and fixed regions.

B Formal Proofs for Interpretability Properties

This appendix provides formal characterizations and proofs for the interpretability properties discussed in Section 3.3. For each property, we define it precisely and analyze whether it can be answered exactly under GAM , GA^{2}M, and CALM.

Note on link functions. Properties P1–P3 are stated and proved for the additive linear predictor $\eta(\mathbf{x}) = \beta_0 + \sum_i f_i^{(r_i(\mathbf{x}_{-i}))}(x_i)$. When a non-identity link function g is used (Eq. 1), these properties apply to η and translate to the final output $\hat{y} = g^{-1}(\eta)$ for any monotone link (e.g., logit, log).

Proposition B.1 (Local Feature Contribution)**.** *Let $\hat{y} : \mathbb{R}^d \to \mathbb{R}$ be a prediction function, and fix an input $\mathbf{x} = (x_1, \ldots, x_d) \in \mathbb{R}^d$. For each index $i \in \{1, \ldots, d\}$, we define the local contribution $\phi_i(\mathbf{x}) \in \mathbb{R}$ as a function intended to represent the contribution of feature x_i to the output $\hat{y}(\mathbf{x})$. We analyze whether such a decomposition*

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{d} \phi_i(\mathbf{x})$$

is uniquely determined by the structure of the model.

(A) In GAM . *Assume the model has the additive form*

$$\hat{y}(\mathbf{x}) = \sum_{j=1}^{d} f_j(x_j),$$

with each $f_j : \mathbb{R} \to \mathbb{R}$. Then for all $i \in \{1, \ldots, d\}$,

$$\phi_i(\mathbf{x}) := f_i(x_i)$$

is well-defined, and

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{d} \phi_i(\mathbf{x}).$$

(B) In GA^{2}M. *Assume the model includes pairwise interactions:*

$$\hat{y}(\mathbf{x}) = \sum_{j=1}^{d} f_j(x_j) + \sum_{j<k} f_{jk}(x_j, x_k),$$

where each $f_j : \mathbb{R} \to \mathbb{R}$ and $f_{jk} : \mathbb{R}^2 \to \mathbb{R}$. Then in general, there does not exist a unique additive decomposition

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{d} \phi_i(\mathbf{x})$$

with each $\phi_i(\mathbf{x})$ depending only on x_i.

(C) In CALM. *Assume the model has the form*

$$\hat{y}(\mathbf{x}) = \sum_{j=1}^{d} f_j^{(r_j(\mathbf{x}_{-j}))}(x_j),$$

where for each j, $r_j : \mathbb{R}^{d-1} \to \{1, \ldots, R_j\}$ is a region selector and $f_j^{(r)} : \mathbb{R} \to \mathbb{R}$ is a univariate shape function for region r. Here $\mathbf{x}_{-j} := \mathbf{x} \setminus x_j \in \mathbb{R}^{d-1}$.

Then for all $i \in \{1, \ldots, d\}$,

$$\phi_i(\mathbf{x}) := f_i^{(r_i(\mathbf{x}_{-i}))}(x_i)$$

is well-defined, and

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{d} \phi_i(\mathbf{x}).$$

Proof. We examine each case separately.

(A) In GAM . By direct substitution, the prediction is an additive sum of univariate functions, each depending only on a single feature x_j. Thus the contribution of feature x_i is uniquely defined as $f_i(x_i)$, and the decomposition is exact.

(B) In GA^{2}M. Suppose, for contradiction, that there exists a decomposition

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{d} \phi_i(\mathbf{x}),$$

where each $\phi_i : \mathbb{R}^d \to \mathbb{R}$ represents the contribution of feature x_i and depends only on x_i, i.e., $\phi_i(\mathbf{x}) = \phi_i(x_i)$.

Then each interaction term $f_{jk}(x_j, x_k)$ must be split between ϕ_j and ϕ_k in such a way that the total sum remains correct and additive over individual features. This is only possible if f_{jk} is additively separable, i.e., if there exist functions $g_j, g_k : \mathbb{R} \to \mathbb{R}$ such that

$$f_{jk}(x_j, x_k) = g_j(x_j) + g_k(x_k).$$

However, the model does not constrain f_{jk} to be separable. In the general case, f_{jk} is non-separable and depends jointly on x_j and x_k.

Therefore, no decomposition $\sum_i \phi_i(x_i)$ can reproduce the prediction $\hat{y}(\mathbf{x})$ using only univariate terms. The contributions $\phi_i(\mathbf{x})$ are not uniquely determined by the model, and must rely on external assumptions or post-hoc attribution methods.

(C) In CALM. For a fixed input $\mathbf{x}$, each region function $r_j(\mathbf{x}_{-j})$ returns a unique region index in $\{1, \ldots, R_j\}$. The corresponding function $f_j^{(r_j(\mathbf{x}_{-j}))}$ is applied to x_j, yielding a scalar. Thus, the contribution of feature x_i is precisely defined as:

$$\phi_i(\mathbf{x}) := f_i^{(r_i(\mathbf{x}_{-i}))}(x_i),$$

and the model prediction is recovered by summing over all features:

$$\hat{y}(\mathbf{x}) = \sum_{i=1}^{d} f_i^{(r_i(\mathbf{x}_{-i}))}(x_i) = \sum_{i=1}^{d} \phi_i(\mathbf{x}).$$

□

The second property asks: *How does the prediction change if we perturb a single feature x_i while keeping all others fixed?* The answer differs across GAM , GA^{2}M, and CALM, as shown below.

Proposition B.2 (Regional Feature Sensitivity)**.** *Let $\hat{y} : \mathbb{R}^d \to \mathbb{R}$ be a prediction function. Fix an input $\mathbf{x} \in \mathbb{R}^d$, an index $i \in \{1, \ldots, d\}$, and a perturbation $\varepsilon > 0$. Define the prediction change under perturbation of x_i as*

$$\Delta\hat{y} := \hat{y}(\mathbf{x} + \varepsilon \mathbf{e}_i) - \hat{y}(\mathbf{x}).$$

We analyze how $\Delta\hat{y}$ can be computed in GAM , GA^{2}M, and CALM.

(A) In GAM . *Assume the model has the additive form*

$$\hat{y}(\mathbf{x}) = \sum_{j=1}^{d} f_j(x_j),$$

with each $f_j : \mathbb{R} \to \mathbb{R}$ univariate. Then

$$\Delta\hat{y} = f_i(x_i + \varepsilon) - f_i(x_i).$$

(B) In GA^{2}M. *We show that in GA^{2}M, the change $\Delta\hat{y}$ caused by perturbing x_i depends on the values of all interacting features x_j for $j \neq i$, and thus cannot be computed from x_i alone.*

(C) In CALM: *We show that the change in prediction $\Delta\hat{y} := \hat{y}(\mathbf{x} + \varepsilon \mathbf{e}_i) - \hat{y}(\mathbf{x})$ is computable from x_i alone if and only if no region transitions are triggered in other features. Otherwise, the change depends on $\mathbf{x}_{-i}$, and regional sensitivity is not determined solely by x_i.*

Proof. Note: The proposition is stated for a fixed input $\mathbf{x}$ (single-valued); the set-valued definition of P2 in the main text arises by ranging over all possible values of $\mathbf{x}_{-i}$, each of which selects a region $r_i(\mathbf{x}_{-i}) \in \{1, \ldots, R_i\}$ and thus one element of the set $\{\Delta f_i^{(r)}\}_{r=1}^{R_i}$.

(A) In GAM . Since all other features remain unchanged, and their contributions are independent of x_i, we have

$$\hat{y}(\mathbf{x} + \varepsilon \mathbf{e}_i) = \sum_{j \neq i} f_j(x_j) + f_i(x_i + \varepsilon),$$
$$\hat{y}(\mathbf{x}) = \sum_{j \neq i} f_j(x_j) + f_i(x_i),$$

and therefore

$$\Delta \hat{y} = f_i(x_i + \varepsilon) - f_i(x_i).$$

(B) In GA^{2}M. Let $\mathcal{J} := \{j \neq i \mid f_{ij}(x_i, x_j) \text{ is present in the model}\}$, i.e., the set of features that interact with x_i. Since only x_i is perturbed, all other features remain unchanged. The only affected terms are the univariate function $f_i(x_i)$, and the interaction terms $f_{ij}(x_i, x_j)$ for each $j \in \mathcal{J}$. All other terms in the model remain constant.

The prediction after perturbation is:

$$\hat{y}(\mathbf{x} + \varepsilon \mathbf{e}_i) = f_i(x_i + \varepsilon) + \sum_{j \in \mathcal{J}} f_j(x_j) + \sum_{j \in \mathcal{J}} f_{ij}(x_i + \varepsilon, x_j) + C,$$

and before perturbation:

$$\hat{y}(\mathbf{x}) = f_i(x_i) + \sum_{j \in \mathcal{J}} f_j(x_j) + \sum_{j \in \mathcal{J}} f_{ij}(x_i, x_j) + C,$$

where $C := \sum_{j<k,\, j,k \neq i} f_{jk}(x_j, x_k)$ denotes the interaction terms not involving x_i, which are unaffected.

Subtracting, we obtain:

$$\Delta \hat{y} = f_i(x_i + \varepsilon) - f_i(x_i) + \sum_{j \in \mathcal{J}} \left[f_{ij}(x_i + \varepsilon, x_j) - f_{ij}(x_i, x_j) \right].$$

Since this expression depends on interacting x_j for $j \neq i$, it cannot be computed from x_i alone.

(C) In CALM. Assume the model has the form

$$\hat{y}(\mathbf{x}) = \sum_{j=1}^{d} f_j^{(r_j(\mathbf{x}_{-j}))}(x_j),$$

where $r_j : \mathbb{R}^{d-1} \to \{1, \ldots, R_j\}$ assigns a region index to each feature j based on the context $\mathbf{x}_{-j} := \mathbf{x} \setminus x_j$, and $f_j^{(r)} : \mathbb{R} \to \mathbb{R}$ is the shape function for region r.

Let $\mathbf{x}_{-i} = \mathbf{x} \setminus x_i$, and define the perturbation $x_i \mapsto x_i + \varepsilon$. Then

$$\Delta \hat{y} = f_i^{(r_i(\mathbf{x}_{-i}))}(x_i + \varepsilon) - f_i^{(r_i(\mathbf{x}_{-i}))}(x_i) + \sum_{j \neq i} \left[f_j^{(r_j(\mathbf{x}_{-j}^{+\varepsilon}))}(x_j) - f_j^{(r_j(\mathbf{x}_{-j}))}(x_j) \right],$$

where $\mathbf{x}_{-j}^{+\varepsilon} := \mathbf{x}_{-j}$ with $x_i \mapsto x_i + \varepsilon$ (since $x_i \in \mathbf{x}_{-j}$ for all $j \neq i$).

Case 1: No region transitions in other features. Assume that for all $j \neq i$, we have $r_j(\mathbf{x}_{-j}^{+\varepsilon}) = r_j(\mathbf{x}_{-j})$. Then:

$$\Delta \hat{y} = f_i^{(r_i(\mathbf{x}_{-i}))}(x_i + \varepsilon) - f_i^{(r_i(\mathbf{x}_{-i}))}(x_i).$$

This case is fully interpretable from the function $f_i^{(r)}$ alone.

Case 2: Region transitions occur in other features. If for some $j \neq i$, the region function changes: $r_j(\mathbf{x}_{-j}^{+\varepsilon}) \neq r_j(\mathbf{x}_{-j})$, then the additional terms contribute:

$$\Delta f_j := f_j^{(r_j(\mathbf{x}_{-j}^{+\varepsilon}))}(x_j) - f_j^{(r_j(\mathbf{x}_{-j}))}(x_j).$$

Hence, the total change includes not only the direct shift in f_i, but also discrete region-based changes in other features and the value of $\Delta \hat{y}$ cannot be recovered from x_i alone. □

The third property asks whether increasing a feature always increases the model's prediction, regardless of the values of other features. In other words, does the model treat the feature as globally monotonic?

Proposition B.3 (Global Feature Monotonicity)**.** *Let $\hat{y} : \mathbb{R}^d \to \mathbb{R}$ be a prediction function, and fix a feature index $i \in \{1, \ldots, d\}$. Define the global monotonicity condition for feature x_i as follows: the model is said to be* globally increasing in x_i *if for all $\mathbf{x} \in \mathbb{R}^d$ and all $\delta > 0$,*

$$\hat{y}(\mathbf{x} + \delta \mathbf{e}_i) \geq \hat{y}(\mathbf{x}),$$

and strictly increasing *if the inequality is strict.*

We analyze whether this property can be verified from the structure of the model.

(A) In GAM . *Assume the model has the form:*

$$\hat{y}(\mathbf{x}) = \sum_{j=1}^{d} f_j(x_j),$$

with each $f_j : \mathbb{R} \to \mathbb{R}$.

Then the model is globally increasing in x_i if and only if f_i is monotonically increasing.

(B) In GA^{2}M. *Assume the model has the form:*

$$\hat{y}(\mathbf{x}) = \sum_{j=1}^{d} f_j(x_j) + \sum_{j<k} f_{jk}(x_j, x_k).$$

Then global monotonicity in x_i cannot be determined from f_i alone.

(C) In CALM. *Assume the model has the form:*

$$\hat{y}(\mathbf{x}) = \sum_{j=1}^{d} f_j^{(r_j(\mathbf{x}_{-j}))}(x_j),$$

where $r_j : \mathbb{R}^{d-1} \to \{1, \ldots, R_j\}$ is a region selector, and $\mathbf{x}_{-j} := \mathbf{x} \setminus x_j$.

Then global monotonicity in x_i holds if the following two conditions are satisfied:

1. *For all regions $r \in \{1, \ldots, R_i\}$, the function $f_i^{(r)} : \mathbb{R} \to \mathbb{R}$ is increasing.*
2. *For all $j \neq i$, and all $x_j \in \mathbb{R}$, the function $x_i \mapsto f_j^{(r_j(\mathbf{x}_{-j}))}(x_j)$ is non-decreasing; i.e., region transitions caused by changing x_i do not decrease f_j's contribution.*

Proof. **(A) In GAM .** Since the model is additive and each term depends only on a single variable, we have:

$$\hat{y}(\mathbf{x} + \delta \mathbf{e}_i) - \hat{y}(\mathbf{x}) = f_i(x_i + \delta) - f_i(x_i).$$

Thus, $\hat{y}(\mathbf{x} + \delta \mathbf{e}_i) \geq \hat{y}(\mathbf{x})$ if and only if f_i is increasing.

(B) In GA^{2}M. As in case (A), $f_i(x_i + \delta) - f_i(x_i)$ gives the direct contribution. However, for each $j \neq i$, the term $f_{ij}(x_i, x_j)$ also contributes. The total change is:

$$\hat{y}(\mathbf{x} + \delta \mathbf{e}_i) - \hat{y}(\mathbf{x}) = f_i(x_i + \delta) - f_i(x_i) + \sum_{j \neq i} \left[f_{ij}(x_i + \delta, x_j) - f_{ij}(x_i, x_j) \right].$$

The sign of this expression depends on the values of x_j, and therefore cannot be determined from f_i alone. Consequently, verifying global monotonicity in x_i requires knowing the full interaction structure and input values.

(C) In CALM Let $\mathbf{x} \in \mathbb{R}^d$ and $\delta > 0$. Define:

$$\Delta \hat{y} := \hat{y}(\mathbf{x} + \delta \mathbf{e}_i) - \hat{y}(\mathbf{x}).$$

Then:

$$\Delta\hat{y} = f_i^{(r_i(\mathbf{x}_{-i}))}(x_i + \delta) - f_i^{(r_i(\mathbf{x}_{-i}))}(x_i) + \sum_{j \neq i} \left[f_j^{(r_j(\mathbf{x}_{-j}^{+\delta}))}(x_j) - f_j^{(r_j(\mathbf{x}_{-j}))}(x_j) \right],$$

where $\mathbf{x}_{-j}^{+\delta}$ is obtained by replacing $x_i \mapsto x_i + \delta$ in $\mathbf{x}_{-j}$.

The first term is non-negative if $f_i^{(r)}$ is increasing for all regions r, and the region $r_i(\mathbf{x}_{-i})$ is fixed.

The second term is a sum over changes in other features' contributions due to possible changes in their regions r_j. For the model to be globally increasing in x_i, all such contributions must be non-negative. This requires that increasing x_i does not cause a decrease in the contribution of any other feature x_j, i.e., region transitions must preserve or increase each f_j.

Therefore, global monotonicity in x_i requires both conditions above. Conversely, if both conditions are satisfied, then each term in the sum is non-negative, implying $\Delta\hat{y} \geq 0$.

□

C Detailed Experiments

C.1 Experimental Setup Details

Reporting convention. As in Section 4, we report mean $\pm$ standard deviation over 5-fold cross-validation.

Table 4: Classification Datasets

Dataset	Size	Attributes
Adult	45222	13
COMPAS	6167	9
HELOC	10459	23
MIMIC2	24508	17
Appendicitis	106	7
Phoneme	5404	5
SPECTF	349	44
Magic	19020	10
Bank	45211	16
Churn	5000	19

Table 5: Regression Datasets

Dataset	Size	Attributes
Bike Sharing	17379	11
California Housing	20640	8
Parkinsons Motor	5875	19
Parkinsons Total	5875	19
Seoul Bike	8465	14
Wine	6497	11
Energy	19735	28
CCPP	9568	4
Electrical	10000	13
Elevators	16599	18
No2	500	7
Sensory	576	11
Airfoil	1503	5
Skill Craft	3338	19
Ailerons	13750	39

Datasets details. Tables 4 and 5 summarize the 10 classification and 15 regression datasets used in our experiments. These datasets are sourced from various publicly available repositories. The UCI Machine Learning Repository [24] provides datasets including Adult, Magic, Bank, Bike Sharing, Parkinson's Motor, Parkinson's Total, Seoul Bike, Wine, CCPP and Skill Craft. OpenML [41] offers datasets such as Electrical, Elevators, No2, Sensory, Airfoil, and Ailerons. The Penn Machine Learning Benchmarks (PMLB) [37] includes datasets like Appendicitis, Phoneme, SPECTF, and Churn. Additional datasets used in our experiments include California Housing [33], MIMIC-II [39], COMPAS [34], HELOC [32], and Energy [5, 6].

Data Preprocessing. All input features are standardized using z-score normalization. For regression tasks, the target variable y is also standard scaled. The only exception is the GAMI-Net model, which requires input features to be scaled using min-max normalization. For RMSE, predictions and targets are inverse-transformed to the original scale prior to evaluation.

Model Configurations. Our black box models comprise a fully connected neural network, a random forest and an XGBoost ensemble. The deep network contains two hidden layers with 50 units each; ReLU activations are used for regression whereas a final sigmoid unit closes the classification variant. Training is carried out for 200 epochs with the Adam optimiser (learning rate=0.001), a batch size of 200 and either mean-squared error or binary cross-entropy loss, depending on the task. The random-forest baseline consists of 500 trees grown to a maximum depth of 25 with a minimum of three samples required at each leaf. For boosted trees we employ XGBoost with 300 boosting rounds, a learning rate of 0.1 and the log-loss evaluation metric for classification

For GAM models we consider NAM and EBM without interactions and Spline. The Neural Additive Model (NAM) builds one independent multilayer perceptron per feature; each sub-network has three hidden layers of 100,100 and 10 ReLU units followed by a linear output, and the additive sum is passed through a sigmoid for binary classification. Training uses Adam (learning rate 0.001), for 10 epochs and a mini-batch size of 32. Spline-GAMs are implemented with PyGAM's LinearGAM or LogisticGAM, allocating a single spline term to every feature while relying on PyGAM's automatic smoothing. Finally, the Explainable Boosting Machine (EBM) is used without interaction terms by explicitly setting $n\_interactions = 0$.

To capture pairwise interactions we experiment with three GA^{2}M variants. EB^{2}M extends the EBM by enabling interactions with a default strength of 0.9; NODE-GA^{2}M activates its interaction mode

Table 6: Number of Interactions per Model in Classification Datasets

Dataset	CALM	EB^{2}M	NodeGA^{2}M	GAMINet
Adult	4.2±1.4	12.0±0.0	72.6±3.2	20.0±0.0
COMPAS	1.4±1.7	9.0±0.0	36.0±0.0	20.0±0.0
HELOC	1.1±1.6	21.0±0.0	163.8±4.4	20.0±0.0
MIMIC2	11.5±2.1	16.0±0.0	115.0±3.3	20.0±0.0
Appendicitis	4.0±1.7	7.0±0.0	21.0±0.0	6.2±8.1
Phoneme	8.6±1.6	5.0±0.0	10.0±0.0	10.0±0.0
SPECTF	0.0±0.0	40.0±0.0	273.0±14.4	20.0±0.0
Magic	8.5±2.4	9.0±0.0	44.0±0.6	20.0±0.0
Bank	9.2±2.0	15.0±0.0	104.2±3.2	20.0±0.0
Churn	14.0±0.0	18.0±0.0	137.4±2.9	20.0±0.0
Avg.	6.2	15.2	97.7	17.6

via the ga2m=1 flag and restricts training to a five-minute time limit to ensure parity with the other models; this cap was chosen to match the practical training budget of competing methods, and we observed that NODE-GA^{2}M's training loss had largely plateaued within this window on most datasets where the cap was reached; GAMI-Net is run with the default hyper-parameters.

For CALM , we configure the heterogeneity drop threshold to 0.2, determining whether a split is considered statistically significant. The region detector, which measures heterogeneity , relies on Partial Dependence Plots (PDP) for RF and XGBoost black-box models, while for DNNs we consider both PDP and RHALE to measure heterogeneity. The detector evaluates 20 candidate split points per feature. The GAM used within regions is an EBM without interactions and the CALM is applied on top of all three black-box models (DNN, RF, XGBoost).

Compatibility. The GAMI-Net baseline depends on a native binary (lib_ebmcore_mac_x64.dylib) compiled for x86_64 architecture. As a result, it is not compatible with Apple Silicon Macs, and running it on such systems will lead to an architecture mismatch error. This issue is limited to this external model and does not affect any of the proposed methods or other baselines. We provide instructions in the README file to guide such users on running all other methods except GAMI-Net.

Computer Resources. All experiments were conducted on an in-house server with cloud infrastructure equipped with an Intel(R) Core(TM) i9-10900X CPU @ 3.70GHz, 128 GB of RAM. No GPU acceleration was utilized during these experiments.

C.2 Number of Interactions

Number of Interactions (Classification). For CALM, the number of interactions is computed as $\sum_i |S_i|$, where S_i is the set of conditioning features appearing in the region tree T_i for feature x_i (i.e., the set of features used as split variables in T_i). Table 6 reports the number of feature interactions selected or used by each model across classification datasets. CALM consistently uses significantly fewer interactions than GA^{2}M-style baselines, often by a wide margin. In most datasets, CALM activates less than a third of the interactions compared to NodeGA^{2}M, and even fewer than GAMINet's default of 20. The numbers shown for CALM reflect actual detected regions with interaction-specific behavior, confirming that its compact regionalization mechanism results in sparse and interpretable models.

Number of Interactions (Regression). As shown in Table 7, CALM activates fewer feature interactions than NodeGA^{2}M and GAMI-Net in regression tasks, and is comparable to EB^{2}M (15.1 vs. 14.1 on average). While GAMI-Net and EB^{2}M use a fixed or near-fixed interaction set, and NodeGA^{2}M often selects over 100 interactions, CALM remains sparse across datasets.

Table 7: Number of Interactions per Model in Regression Datasets

Dataset	CALM	EB^{2}M	NodeGA^{2}M	GAMINet
Bike Sharing	19.3±2.7	10.0±0.0	53.6±1.4	20.0±0.0
California Housing	10.7±2.0	8.0±0.0	28.0±0.0	20.0±0.0
Parkinsons Motor	13.8±4.4	18.0±0.0	126.2±1.7	20.0±0.0
Parkinsons Total	14.4±5.8	18.0±0.0	129.2±3.9	20.0±0.0
Seoul Bike	20.4±1.0	13.0±0.0	83.2±0.7	20.0±0.0
Wine	13.9±3.3	10.0±0.0	54.6±0.5	20.0±0.0
Energy	49.2±4.4	26.0±0.0	192.0±4.5	4.0±8.0
CCPP	1.9±2.2	4.0±0.0	6.0±0.0	5.8±0.4
Electrical	0.0±0.0	12.0±0.0	68.6±3.0	12.0±0.0
Elevators	11.7±3.0	17.0±0.0	115.4±2.1	20.0±0.0
No2	12.8±1.9	7.0±0.0	21.0±0.0	20.0±0.0
Sensory	6.9±4.2	10.0±0.0	54.0±0.9	16.0±4.9
Airfoil	9.3±0.5	5.0±0.0	10.0±0.0	10.0±0.0
Skill Craft	0.0±0.0	18.0±0.0	135.6±2.4	16.0±8.0
Ailerons	41.7±2.6	36.0±0.0	232.4±5.1	20.0±0.0
Avg.	15.1	14.1	87.3	16.3

Table 8: Runtime (Seconds) for Classification Datasets

Dataset	BlackBox	GAM		CALM	GA^{2}M		
	XGB	NAM	EBM		EB^{2}M	NodeGA^{2}M	GAMINet
Adult	0.2±0.01	38±2	10±1	36±1	13±2	97±9	718±122
COMPAS	0.1±0.004	10±0.1	2±2	2±0.1	1±0.04	51±1	129±5
HELOC	0.2±0.01	31±6	2±2	17±1	2±0.2	57±0.1	249±60
MIMIC2	0.2±0.01	31±0.3	4±3	26±3	5±1	69±9	359±32
Appendicitis	0.1±0.001	5±0.1	2±4	2±0.1	1±0.5	24±0.1	6±5
Phoneme	0.1±0.01	6±1	3±4	3±3	4±1	61±4	166±38
SPECTF	0.1±0.01	32±0.3	3±4	3±3	4±5	25±0.2	180±13
Magic	0.2±0.003	28±18	5±5	17±3	7±4	98±7	489±63
Bank	0.2±0.04	44±1	5±1	42±3	13±1	98±17	994±265
Churn	0.2±0.01	19±3	2±1	10±4	3±2	56±2	257±32
Avg.	0.2	24	4	16	5	64	355

C.3 Practical Runtime

We report the total runtime of each method across all datasets in Tables 8 and 9. For CALM, the reported time includes not only the regions detection fitting steps, but also the training time of the underlying black-box model and the GAM component used within regions. While this inclusion gives a complete view of end-to-end cost, it somewhat disadvantages CALM in comparison to other models, whose runtimes reflect only their own training processes. In practice, the black-box or GAM components can be selected to be lightweight and the blackbox model can be reused or pretrained independently, making CALM's region discovery step lightweight and modular.

Despite this conservative accounting, CALM's runtime remains competitive. For example, it often trains faster than or comparably to GA^{2}M methods such as NODE-GA^{2}M and GAMI-Net, which tend to have high computational overhead. On small and medium-sized datasets like COMPAS, HELOC, California Housing, and Wine, CALM runs in under a minute, showing that its interpretability gains come with reasonable computational cost. On larger datasets such as Bank or Ailerons, CALM's runtime scales moderately but remains practical.

Table 9: Runtime (Seconds) for Regression Datasets

Dataset	BlackBox	GAM		CALM	GA^2M		
	XGB	NAM	EBM		EB^2M	$NodeGA^2M$	GAMINet
Bike Sharing	0.1±0.01	18±1	2±1	15±1	19±3	187±31	299±33
California Housing	0.2±0.01	15±0.1	4±2	13±1	16±0.5	200±46	677±147
Parkinsons Motor	0.3±0.003	21±6	5±2	17±2	44±4	193±34	536±88
Parkinsons Total	0.3±0.01	18±0.2	5±3	16±2	40±5	227±76	543±79
Seoul Bike	0.2±0.01	15±0.2	3±3	13±3	12±4	127±18	268±53
Wine	0.2±0.01	12±1	2±3	9±3	3±0.4	62±10	187±20
Energy	0.3±0.01	43±0.4	8±3	82±5	97±15	206±57	253±214
CCPP	0.2±0.004	6±0.1	5±4	8±5	14±2	133±18	88±8
Electrical	0.1±0.002	25±20	9±4	11±5	8±3	112±28	279±43
Elevators	0.3±0.1	25±0.4	7±4	35±5	43±8	181±38	773±269
No2	0.1±0.003	5±0.04	2±0.1	2±0.1	2±0.1	26±0.1	78±1
Sensory	0.1±0.001	10±1	4±6	7±5	2±0.4	27±0.5	74±16
Airfoil	0.1±0.001	4±0.03	3±6	5±6	7±1	51±12	76±34
Skill Craft	0.3±0.003	17±0.2	4±6	9±6	1±0.1	44±0.1	141±52
Ailerons	0.3±0.02	53±1	5±7	70±8	3±0.1	126±36	619±134
Avg.	0.2	19	5	21	21	127	326

C.4 Performance Metrics for all Experiments

Performance Summary. We evaluated CALM on 25 datasets using three model classes: DNN, XGB, and RF. For each model and dataset, we compare CALM to both its corresponding GAM baseline and the original black-box model.

In comparisons with GAMs, we match the model structure: for example, CALM-NAM is compared directly to NAM, so the performance difference reflects only the benefit of introducing regional modeling. In comparisons with the black-box, we report the accuracy of the *best* CALM-GAM. For XGB and RF, this is selected across multiple GAM types (NAM, EBM, Spline) using a fixed heterogeneity threshold of 0.2. For DNNs, we first identify the best heterogeneity modeling method (PDP or RHALE) for each GAM, and then select the best-performing GAM among all types. While the threshold is fixed in our experiments, we note that alternative thresholds could yield further improvements.

The results cover six tables: Tables 10–12 report performance on regression datasets using DNN, XGB, and RF respectively, while Tables 13–15 report on the same three model classes for classification datasets. Across these settings, CALM consistently outperforms the corresponding GAM baseline. It matches or exceeds the performance of its GAM counterpart in 97.8% and 100% of cases for DNNs (Tables 10, 13), 97.8% and 96.7% for XGB (Tables 11, 14), and 95.6% and 96.7% for RF (Tables 12, 15). These results reflect direct, structure-matched comparisons between CALM and GAMs of the same architecture.

When compared to the black-box models, CALM also achieves strong results. For regression, the best CALM-GAM variant strictly outperforms the black-box on 9/15 datasets with DNN (with 2 ties; Table 10), on 3/15 with XGB (2 ties; Table 11), and on 3/15 with RF (1 tie; Table 12). In classification, it beats the black-box in 90.0% of DNN cases (Table 13), 50.0% for XGB (Table 14), and 60.0% for RF (Table 15).

Overall, CALM outperforms or matches the corresponding GAM in 219 out of 225 cases (97.3%) and surpasses the black-box in 40 out of 75 cases (53.3%). The rare cases where CALM does not improve over the GAM occur in both regression and classification tasks and almost always correspond to scenarios where the GAM itself performs as well as or better than the black-box. These cases suggest that the underlying function is already well captured by global additive effects, leaving little room for further refinement through regional modeling.

Table 10: RMSE Score for Regression Datasets, DNN

Dataset	BlackBox	GAM			CALM - NAM		CALM - EBM		CALM - Spline	
	DNN	NAM	EBM	Spline	PDP	RHALE	PDP	RHALE	PDP	RHALE
Bike Sharing	42.952±1.460	101.968±1.309	100.211±1.010	100.876±0.868	63.034±0.883	64.941±2.566	58.444±1.637	60.966±2.983	59.868±1.564	62.108±2.989
California Housing	0.519±0.017	0.611±0.011	0.554±0.008	0.632±0.009	0.580±0.012	0.591±0.016	0.522±0.010	0.526±0.009	0.593±0.018	0.597±0.006
Parkinsons Motor	3.654±0.162	6.112±0.158	4.204±0.087	6.027±0.070	4.220±0.192	5.303±0.312	2.681±0.105	3.658±0.324	4.149±0.211	5.318±0.430
Parkinsons Total	5.139±0.102	7.897±0.102	4.847±0.107	7.519±0.072	5.456±0.182	6.776±0.381	3.587±0.030	4.319±0.222	5.408±0.365	6.827±0.209
Seoul Bike	244.967±5.081	320.225±4.377	303.685±3.855	302.020±4.701	269.277±6.414	273.241±8.379	249.457±2.407	245.880±5.153	246.855±3.936	243.926±4.433
Wine	0.700±0.015	0.719±0.009	0.703±0.012	0.736±0.039	0.712±0.005	0.711±0.012	0.695±0.011	0.695±0.012	0.729±0.027	0.734±0.035
Energy	77.139±2.916	89.799±2.665	85.225±2.489	86.498±2.384	89.304±2.480	88.600±2.179	84.347±2.358	83.972±2.557	84.929±1.868	83.526±2.653
CCPP	3.916±0.079	4.213±0.050	3.435±0.076	4.144±0.054	4.205±0.067	4.188±0.079	3.427±0.058	3.379±0.083	3.995±0.061	4.006±0.058
Electrical	0.051±0.009	0.040±0.005	0.018±0.011	0.095±0.005	0.040±0.005	0.040±0.005	0.018±0.011	0.012±0.007	0.094±0.005	0.088±0.003
Elevators	0.002±0.000	0.003±0.000	0.002±0.000	0.002±0.000	0.002±0.000	0.002±0.000	0.002±0.000	0.002±0.000	0.002±0.000	0.002±0.000
No2	0.522±0.012	0.503±0.021	0.492±0.034	0.480±0.032	0.507±0.016	0.501±0.025	0.492±0.030	0.498±0.034	0.485±0.028	0.492±0.033
Sensory	0.522±0.022	0.483±0.014	0.477±0.009	0.482±0.012	0.461±0.025	0.449±0.019	0.447±0.021	0.444±0.012	0.453±0.025	0.446±0.020
Airfoil	2.131±0.131	4.801±0.198	4.565±0.148	4.542±0.121	3.275±0.196	3.143±0.150	2.647±0.194	2.449±0.227	2.697±0.161	2.588±0.248
Skill Craft	1.231±0.195	0.925±0.025	0.901±0.021	2.515±3.092	0.923±0.028	0.918±0.023	0.901±0.021	0.905±0.018	1.652±1.323	2.345±2.792
Ailerons	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.001±0.001

Table 11: RMSE Score for Regression Datasets, XGB

Dataset	BlackBox	GAM			CALM		
	XGB	NAM	EBM	Spline	NAM	EBM	Spline
Bike Sharing	39.346±1.375	101.968±1.309	100.211±1.010	100.876±0.868	59.732±1.414	55.667±1.220	56.656±1.320
California Housing	0.454±0.010	0.611±0.011	0.554±0.008	0.632±0.009	0.566±0.010	0.511±0.010	0.594±0.014
Parkinsons Motor	1.443±0.094	6.112±0.158	4.204±0.087	6.027±0.070	4.574±0.137	2.243±0.126	4.432±0.185
Parkinsons Total	1.863±0.079	7.897±0.102	4.847±0.107	7.519±0.072	6.109±0.361	2.968±0.089	5.638±0.288
Seoul Bike	209.587±3.474	320.225±4.377	303.685±3.855	302.020±4.701	270.365±7.049	238.912±1.643	237.705±2.658
Wine	0.622±0.012	0.719±0.009	0.703±0.012	0.736±0.039	0.703±0.011	0.693±0.016	0.737±0.043
Energy	67.967±2.579	89.799±2.665	85.225±2.489	86.498±2.384	87.416±2.894	83.088±1.974	82.747±2.421
CCPP	3.086±0.090	4.213±0.050	3.435±0.076	4.144±0.054	4.171±0.042	3.419±0.066	4.039±0.036
Electrical	0.037±0.015	0.040±0.005	0.018±0.011	0.095±0.005	0.040±0.005	0.018±0.011	0.095±0.005
Elevators	0.002±0.000	0.003±0.000	0.002±0.000	0.002±0.000	0.002±0.000	0.002±0.000	0.002±0.000
No2	0.473±0.026	0.503±0.021	0.492±0.034	0.480±0.032	0.498±0.022	0.492±0.036	0.479±0.031
Sensory	0.508±0.028	0.483±0.014	0.477±0.009	0.482±0.012	0.462±0.016	0.450±0.020	0.451±0.029
Airfoil	1.544±0.100	4.801±0.198	4.565±0.148	4.542±0.121	3.146±0.137	2.503±0.147	2.582±0.130
Skill Craft	0.938±0.019	0.925±0.025	0.901±0.021	2.515±3.092	0.925±0.025	0.901±0.021	2.167±2.261
Ailerons	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000

Table 12: RMSE Score for Regression Datasets, RF

Dataset	BlackBox	GAM			CALM		
	RF	NAM	EBM	Spline	NAM	EBM	Spline
Bike Sharing	43.247±1.037	101.968±1.309	100.211±1.010	100.876±0.868	61.174±1.544	57.158±1.288	57.823±1.393
California Housing	0.502±0.012	0.611±0.011	0.554±0.008	0.632±0.009	0.570±0.015	0.515±0.012	0.611±0.025
Parkinsons Motor	1.477±0.155	6.112±0.158	4.204±0.087	6.027±0.070	4.515±0.094	2.230±0.081	4.349±0.061
Parkinsons Total	1.869±0.243	7.897±0.102	4.847±0.107	7.519±0.072	5.761±0.259	3.044±0.068	5.393±0.162
Seoul Bike	225.572±3.086	320.225±4.377	303.685±3.855	302.020±4.701	270.514±8.146	243.910±2.580	240.471±4.337
Wine	0.618±0.012	0.719±0.009	0.703±0.012	0.736±0.039	0.708±0.006	0.691±0.011	0.726±0.036
Energy	69.406±2.781	89.799±2.665	85.225±2.489	86.498±2.384	88.448±2.607	84.741±2.520	84.690±2.546
CCPP	3.378±0.078	4.213±0.050	3.435±0.076	4.144±0.054	4.129±0.057	3.422±0.112	4.029±0.052
Electrical	0.009±0.009	0.040±0.005	0.018±0.011	0.095±0.005	0.040±0.005	0.018±0.011	0.095±0.005
Elevators	0.003±0.000	0.003±0.000	0.002±0.000	0.002±0.000	0.002±0.000	0.002±0.000	0.002±0.000
No2	0.466±0.033	0.503±0.021	0.492±0.034	0.480±0.032	0.481±0.023	0.482±0.031	0.484±0.029
Sensory	0.446±0.007	0.483±0.014	0.477±0.009	0.482±0.012	0.451±0.022	0.439±0.020	0.442±0.025
Airfoil	2.196±0.170	4.801±0.198	4.565±0.148	4.542±0.121	3.073±0.143	2.533±0.137	2.627±0.108
Skill Craft	0.916±0.019	0.925±0.025	0.901±0.021	2.515±3.092	0.926±0.037	0.901±0.022	2.320±2.696
Ailerons	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000	0.0002±0.000

Table 13: Accuracy Score for Classification Datasets, DNN

Dataset	BlackBox	GAM			CALM - NAM		CALM - EBM		CALM - Spline	
	DNN	NAM	EBM	Spline	PDP	RHALE	PDP	RHALE	PDP	RHALE
Adult	0.849±0.002	0.851±0.002	0.870±0.002	0.856±0.001	0.853±0.001	0.854±0.003	0.870±0.003	0.871±0.003	0.857±0.003	0.859±0.003
COMPAS	0.682±0.011	0.682±0.016	0.681±0.012	0.685±0.012	0.682±0.007	0.681±0.014	0.683±0.011	0.679±0.012	0.683±0.015	0.683±0.011
HELOC	0.721±0.009	0.723±0.011	0.728±0.014	0.726±0.010	0.723±0.011	0.723±0.013	0.727±0.015	0.728±0.014	0.721±0.012	0.726±0.011
MIMIC2	0.885±0.003	0.886±0.001	0.886±0.003	0.886±0.003	0.887±0.001	0.886±0.001	0.886±0.003	0.886±0.002	0.886±0.003	0.886±0.003
Appendicitis	0.877±0.072	0.848±0.056	0.877±0.077	0.887±0.064	0.839±0.025	0.848±0.056	0.897±0.055	0.877±0.077	0.887±0.064	0.887±0.064
Phoneme	0.815±0.009	0.808±0.002	0.821±0.007	0.832±0.006	0.839±0.012	0.834±0.008	0.858±0.010	0.856±0.008	0.855±0.009	0.852±0.006
SPECTF	0.814±0.016	0.839±0.030	0.894±0.015	0.845±0.040	0.848±0.036	0.882±0.025	0.885±0.021	0.903±0.014	0.848±0.032	0.782±0.036
Magic	0.870±0.004	0.850±0.006	0.857±0.005	0.855±0.004	0.860±0.007	0.858±0.004	0.862±0.006	0.860±0.006	0.865±0.005	0.860±0.005
Bank	0.903±0.003	0.901±0.003	0.902±0.002	0.903±0.002	0.902±0.001	0.903±0.005	0.905±0.003	0.904±0.003	0.906±0.003	0.905±0.005
Churn	0.938±0.005	0.885±0.009	0.886±0.005	0.889±0.005	0.957±0.010	0.949±0.004	0.953±0.006	0.946±0.006	0.959±0.006	0.941±0.007

D User Study

Study Overview. We conducted a within-subjects user study ($N = 25$) to evaluate the practical interpretability of CALMs. Participants ranged from novices to experts in machine learning and data

Table 14: Accuracy Score for Classification Datasets, XGB

Dataset	BlackBox	GAM			CALM		
	XGB	NAM	EBM	Spline	NAM	EBM	Spline
Adult	0.870±0.002	0.851±0.002	0.870±0.002	0.856±0.001	0.853±0.002	0.870±0.002	0.858±0.001
COMPAS	0.661±0.007	0.682±0.016	0.681±0.012	0.685±0.012	0.686±0.014	0.684±0.013	0.686±0.016
HELOC	0.717±0.013	0.723±0.011	0.728±0.014	0.726±0.010	0.724±0.011	0.728±0.012	0.726±0.011
MIMIC2	0.890±0.001	0.886±0.001	0.886±0.003	0.886±0.003	0.886±0.001	0.886±0.003	0.886±0.003
Appendicitis	0.868±0.069	0.848±0.056	0.877±0.077	0.887±0.064	0.868±0.069	0.878±0.063	0.878±0.063
Phoneme	0.898±0.006	0.808±0.002	0.821±0.007	0.832±0.006	0.843±0.006	0.861±0.011	0.860±0.010
SPECTF	0.862±0.029	0.839±0.030	0.894±0.015	0.845±0.040	0.857±0.043	0.894±0.015	0.845±0.040
Magic	0.885±0.004	0.850±0.006	0.857±0.005	0.855±0.004	0.863±0.007	0.864±0.004	0.868±0.006
Bank	0.908±0.003	0.901±0.003	0.902±0.002	0.903±0.002	0.903±0.002	0.905±0.002	0.907±0.002
Churn	0.958±0.004	0.885±0.009	0.886±0.005	0.889±0.005	0.954±0.007	0.946±0.003	0.953±0.008

Table 15: Accuracy Score for Classification Datasets, RF

Dataset	BlackBox	GAM			CALM		
	RF	NAM	EBM	Spline	NAM	EBM	Spline
Adult	0.843±0.001	0.851±0.002	0.870±0.002	0.856±0.001	0.853±0.002	0.871±0.003	0.859±0.001
COMPAS	0.662±0.016	0.682±0.016	0.681±0.012	0.685±0.012	0.682±0.016	0.683±0.012	0.685±0.012
HELOC	0.724±0.014	0.723±0.011	0.728±0.014	0.726±0.010	0.723±0.014	0.728±0.014	0.726±0.010
MIMIC2	0.888±0.002	0.886±0.001	0.886±0.003	0.886±0.003	0.886±0.002	0.887±0.002	0.886±0.003
Appendicitis	0.859±0.064	0.848±0.056	0.877±0.077	0.887±0.064	0.848±0.056	0.877±0.077	0.887±0.064
Phoneme	0.898±0.005	0.808±0.002	0.821±0.007	0.832±0.006	0.838±0.005	0.859±0.005	0.859±0.009
SPECTF	0.877±0.027	0.839±0.030	0.894±0.015	0.845±0.040	0.848±0.026	0.894±0.018	0.842±0.048
Magic	0.878±0.004	0.850±0.006	0.857±0.005	0.855±0.004	0.861±0.004	0.864±0.004	0.866±0.005
Bank	0.901±0.004	0.901±0.003	0.902±0.002	0.903±0.002	0.905±0.003	0.907±0.002	0.907±0.002
Churn	0.957±0.005	0.885±0.009	0.886±0.005	0.889±0.005	0.952±0.005	0.951±0.007	0.955±0.006

analysis, and were asked to estimate feature contributions for specific inputs and reason about how predictions change under feature perturbations. Participants were not financially compensated.

Participant Expertise. Participants self-reported their level of expertise using a 5-point scale ranging from no experience to expert.

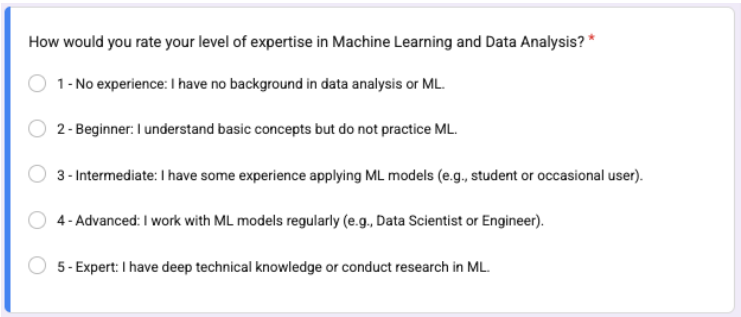


Figure 4: Self-reported expertise question used to assess participant background.

Interpretation Protocol. Before starting the tasks, participants were provided with standardized instructions on how to interpret conditional feature plots. These instructions explained how to: (i) select the appropriate curve based on feature conditions, (ii) read feature contributions directly from the plot, and (iii) account for interaction-induced discontinuities when estimating changes.

Task Design. Participants completed tasks targeting two aspects of interpretability: estimating feature contributions for specific inputs and reasoning about how predictions change under feature perturbations.

Example Questions. Figure 6 shows a representative example of the questions used in the study. The full questionnaire is available at `https://forms.gle/tTippi5eJ6cZHrfy9`.

Figure 5: Instruction panel provided to participants for interpreting conditional feature plots.

Evaluation Metrics. We evaluate performance using three metrics: task accuracy (percentage of correct answers), accuracy on prediction change tasks, and user experience, measured via self-reported ease of use and confidence.

Results. Participants achieved an average validation accuracy of 83.2% using CALM. On real-world tasks, CALM significantly outperformed heatmap-based representations in overall task performance (54% vs. 34%, $p < 0.05$), driven primarily by improvements in prediction change tasks (40% vs. 12%, $p < 0.05$).

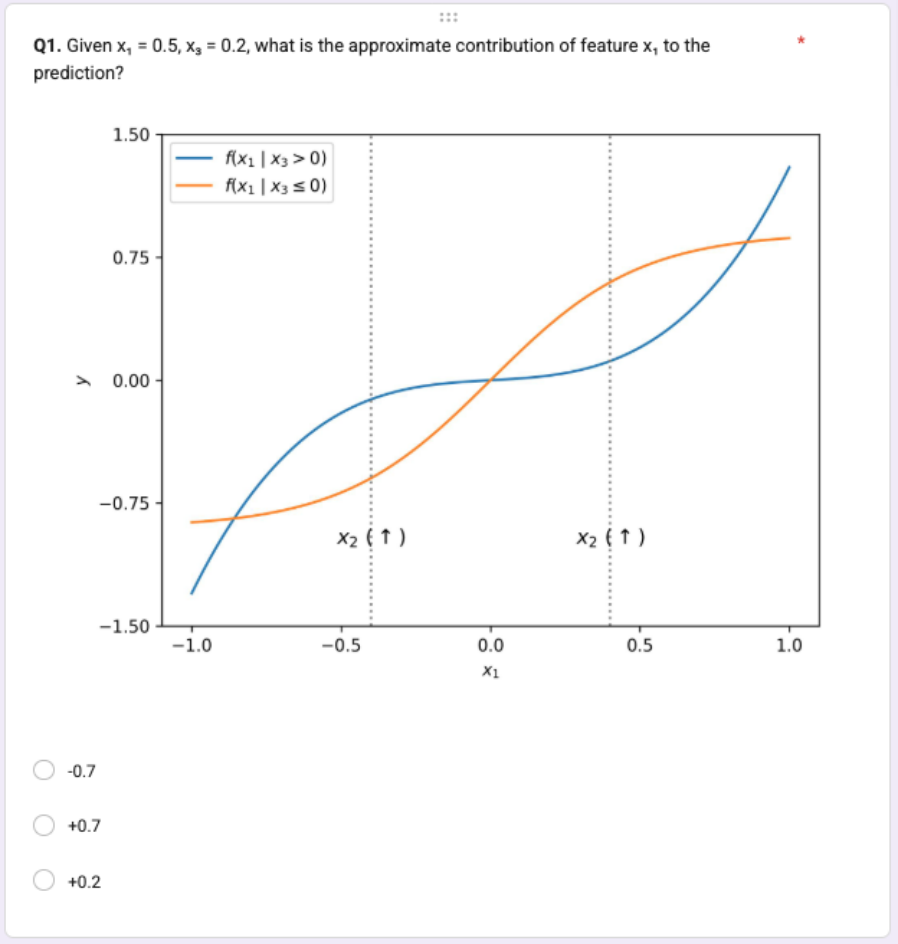


Figure 6: Example question requiring estimation of feature contribution from a conditional plot.

Participants also rated CALM as easier to use ($p < 0.01$) and more confidence-inducing ($p < 0.001$), indicating improved usability alongside stronger objective performance.

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 accurately summarize the paper's contributions, including the proposed CALM model, its training procedure, theoretical analysis, and empirical evaluation, while appropriately reflecting the scope of the results and supporting claims on interpretability with both quantitative benchmarks and a user study.

   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 paper explicitly discusses key limitations, including the dependence on the quality of the reference model, the potential reduction in interpretability with increasing interactions, and the trade-off between model complexity and clarity.

   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: The paper states the assumptions for its theoretical results and provides complete proofs in the appendix, with references from the main text.

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 provides detailed descriptions of the model, training pipeline, datasets, evaluation protocols, and hyperparameters, and includes code for reproducing all experiments in the supplementary material.

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 provides open access to code and includes instructions for reproducing the main experimental results, as described in the supplementary material.

   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: The paper specifies the experimental setup, including datasets, evaluation protocols, and key training details, with additional implementation and hyperparameter information provided in the appendix.

   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 results as mean $\pm$ standard deviation over 5-fold cross-validation, clearly indicating variability due to data splits; statistical significance is additionally reported for the user study.

   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: The paper reports the compute environment used for all experiments, including CPU type and memory, and clarifies that no GPU acceleration was used.

   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 research complies with the NeurIPS Code of Ethics, involving standard machine learning methodology, use of publicly available datasets, and an anonymized user study conducted with informed consent.

   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: [Yes]

Justification: The paper discusses the positive impact of improved interpretability and outlines technical limitations of the approach, though broader societal risks are not extensively explored.

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 involve the release of models or datasets that pose a high risk of misuse, as it focuses on a method for interpretable modeling on standard tabular data.

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: The paper uses standard publicly available datasets and implementations, which are properly cited; while licenses are not explicitly listed, all assets are used in accordance with their standard terms.

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: [Yes]

Justification: The paper introduces code for the proposed method, which is documented and provided in the supplementary material.

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:[Yes]

Justification: The paper includes detailed descriptions of the user study, with instructions, screenshots, and representative questions provided in the appendix, along with access to the full questionnaire and includes details about participant compensation.

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: [No]

Justification: The paper includes a user study but does not explicitly discuss IRB approval or risk disclosure procedures.

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: The research does not involve LLMs as part of the core methodology, experiments, or analysis.

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.