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

Vasilis Gkolemis[1,2] **Loukas Kavouras**[2] **Dimitrios Kyriakopoulos**[2,6]
Konstantinos Tsopelas[2] **Dimitrios Rontogiannis**[4] **Giuseppe Casalicchio**[5]
Theodore Dalamagas[2] **Christos Diou**[1,3]

[1]Department of Informatics and Telematics, Harokopio University of Athens
[2]IMSI, ATHENA RC [3]Archimedes AI, ATHENA RC [4]Max Planck ISS
[5]LMU Munich [6]National Observatory of Athens

Abstract

Generalized additive models (GAMs) offer high 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. We propose *Conditionally Additive Local Models* (CALMs), a new model class that narrows this trade-off: it stays close to GAMs in interpretability and close to GA^{2}Ms in accuracy. 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 never underperform GAMs, often beat them and achieve accuracy broadly comparable to GA^{2}Ms, while preserving the univariate auditability that GA^{2}Ms forfeit.

1 Introduction

In high-stakes decision making, models must provide not only reliable predictions but also human-understandable explanations [93, 30]. This has motivated *interpretable-by-design* models, whose structure permits direct inspection of their behavior [101, 88]. Interpretability, however, is not a yes-or-no property but a continuum: interpretable-by-design models range from more to less interpretable, and gains in predictive accuracy often come at the cost of greater structural complexity [82, 49].

This trade-off is evident in GAMs [83] and their extension, GA^{2}Ms [84, 21]. GAMs model the prediction as a sum of univariate effects, which makes them highly interpretable but blind to feature interactions which limits their accuracy. GA^{2}Ms gain accuracy by adding selected pairwise interactions, but the interaction terms cost them two interpretability properties [98, 116]: (i) a prediction no longer splits into one contribution per feature, because an interaction term belongs to two features jointly, and (ii) global auditing is hard to do visually: a feature's effect is spread over its univariate and bivariate terms, so no single plot shows how the feature affects the prediction (see Section 3.2).

We introduce *Conditionally Additive Local Models (CALMs)*, a model class that narrows this trade-off: it stays close to GAMs in interpretability and close to GA^{2}Ms in accuracy. The core idea is to learn *conditional feature effects* [52, 8]: several univariate effects per feature, each active in a region of the input space where the feature is nearly additive [64], i.e., barely interacts with other features. The regions are defined by threshold conditions on the features it interacts with. Figure 1 shows a CALM fitted on the Bike Sharing dataset [34] and Figure 2 compares it with a GAM and a GA^{2}M.

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

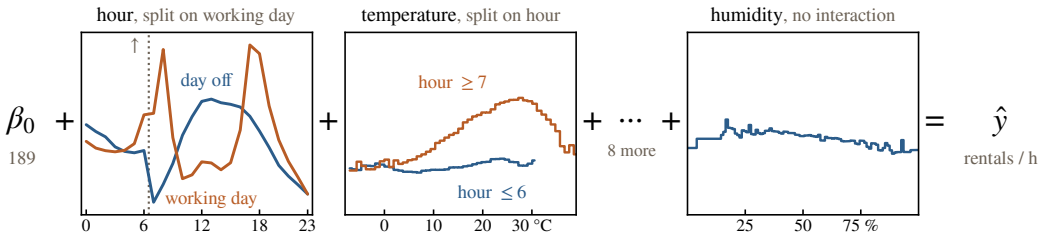
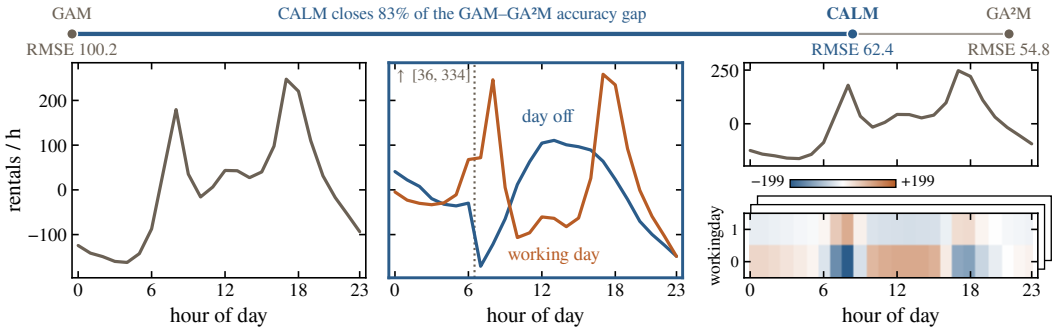


Figure 1: A CALM fitted on the Bike Sharing dataset: the prediction sums one plot per feature, each plot holding one curve per region. The hour splits on the working day; the temperature splits on the hour, mattering by day and barely at night; the humidity interacts with no feature and keeps one curve. The dotted line on the hour's plot marks where the temperature's and the year's regions switch and its arrow gives the sign of the hidden jump; Section 3.2 explains it in depth.



(a) GAM (1 curve) **(b) CALM** (2 curves) **(c) GA²M** (1 curve + 7 surfaces)

Figure 2: The effect of the hour on bike rentals (Bike Sharing dataset). Top: test RMSE (Table 5); CALM closes 83% of the gap from GAM to GA^{2}M. (a) GAM: one curve. The interaction with the working day is averaged away, which limits accuracy. (b) CALM: one curve for working days and one for days off (at $d_{\max} = 1$). The dotted line mark a hidden jump (see Section 3.2 for more details). (c) GA^{2}M: a curve and seven 2D surfaces (only one shown here) for decoding the effect of the hour, which limits its interpretability.

In this example, the feature `hour` interacts with `workingday`, so a CALM draws its effect as two curves, one for working and one for non-working days. On working days rentals peak at the morning and evening commute, whereas on non-working days they rise to a single broad peak around midday. The CALM thus captures the interaction while keeping (i) one contribution per feature and (ii) one plot per feature, unlike a GAM, which averages the two curves into one and loses accuracy, and a GA^{2}M, which expresses the interaction as a pairwise surface and loses both interpretability properties.

We fit CALMs with a three-step distillation pipeline: (1) train a black-box reference model that captures the interactions in the data; (2) for each feature, grow a CART-style tree [13, 56] that splits the input space into regions of reduced interaction, then greedily keep the trees that matter most for reproducing the reference model's predictions; (3) fit one curve per region with a region-aware backfitting [17]. Two parameters bound what a reader must inspect: the tree depth $d_{\max}$ caps how many curves each feature has and K caps how many interactions the model captures in total.

Contributions. (i) CALMs, an interpretable-by-design model class built on conditional feature effects (Section 3); (ii) a distillation-based training algorithm (Section 3.1); (iii) a formal analysis showing that CALMs keep the two interpretability properties that GA^{2}Ms lose (Section 3.2); (iv) an evaluation on 24 real-world regression and classification datasets, in which CALMs never lose to a GAM baseline and match or beat EB^{2}M on more than half (Section 4.2); (v) evidence that CALMs stay small and stable: about eight curves beyond a GAM, growing with tree depth as accuracy rises (Section 4.2), and the same structure for most features across resamples (Section 4.3); (vi) a within-subjects user study ($N = 25$), in which participants answered more accurately from CALM plots than from GA^{2}M heatmaps (Section 4.4). Code: `https://github.com/givasile/calm-paper`.

2 Background and Related Work

Decision trees [13], rule lists [80, 4] and prototype-based classifiers [25, 10] are typical interpretable-by-design models; among them, GAMs [51, 115] are popular for their simple global interpretability.

GAMs model the output as $g(\mathbb{E}[y|\mathbf{x}]) = \beta_0 + \sum_i f_i(x_i)$, with univariate shape functions f_i, intercept β_0 and link function g. They differ mainly in how f_i are learned: penalized splines [115] give smooth curves, boosted bagged shallow trees [83, 94] give step functions that follow sharp changes [24], and neural variants learn one network per feature [3], a set of basis functions shared by all features [98], or a boosted ensemble of extreme learning machines [75].

GA^{2}Ms add pairwise terms, $g(\mathbb{E}[y|\mathbf{x}]) = \beta_0 + \sum_i f_i(x_i) + \sum_{(i,j)\in\mathcal{I}} f_{ij}(x_i, x_j)$, over a set $\mathcal{I}$ of selected pairs, and differ in how they choose $\mathcal{I}$ and fit the terms [112, 66]. GA^{2}M [84] fits boosted trees and adds pairs greedily, ranked by how much each reduces the residual loss; GAMI-Net [117] fits one network per term and considers only pairs with at least one already-selected main effect; GRAND-SLAMIN [70] fits soft trees and selects pairs end to end through differentiable gates, optionally under hierarchy constraints; and NODE-GA^{2}M [23] fits differentiable oblivious trees, each restricted to at most two features.

CALM draws inspiration from *regional effect methods*, post-hoc explainability methods that extend *global effect plots*. Global effect plots, such as Partial Dependence Plots (PDP) [37], Accumulated Local Effects (ALE) [6] and SHAP Dependence Plots (SHAP-DP) [85], explain a black-box model f by decomposing it into d one-dimensional plots, one per feature x_i. When f contains strong interactions, these plots average heterogeneous local effects into one curve and can mislead, a phenomenon known as aggregation bias [87, 56, 7, 41, 45, 91].

Regional effect methods [43] address this by splitting the input space with a tree into subregions where interactions are weak [104], and drawing each feature's plot separately in every subregion. REPID [56] grows one tree per feature, while GADGET [59], FD-Tree [77] and GRANITE [58] grow one tree shared by several features. CALM adopts GADGET's heterogeneity criterion and REPID's per-feature trees to find its regions (Step 1(b), Section 3.1). Per-feature trees let each feature split only on the features it interacts with [8, 18], whereas a shared tree imposes the same splits on every feature [118]. In Figure 1, a shared tree would split the humidity, which interacts with no feature, on the working day only because the hour needs it. Finally, we clarify that GADGET, REPID and all regional effect methods are not competitors of CALM: they explain a fitted model post hoc, whereas a CALM is itself a predictor.

CALM also relates to model distillation [61, 108] and surrogate models [48], which train an interpretable student to mimic a black-box teacher, locally [99] or globally [39, 48]. However, instead of explaining the teacher, a CALM uses it only to find where interactions are weak, and then fits its curves to the data. In a rough analogy, CALM serves as the interpretable-by-design student that replaces the black-box teacher as the final predictor [55, 47]. SLIM [68] and other model-based trees [57] fit a simple model in each leaf of a single tree, mainly as surrogates that explain a black box, so all features share one partition. Regionally additive models [44] also build an interpretable model from regional effects, but do not characterize its interpretability or evaluate it 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.

Model formulation. Let $\mathbf{X} = (X_1, \ldots, X_d)$ be a random vector with distribution $P_{\mathbf{X}}$ on $\mathcal{X} = \mathcal{X}_1 \times \cdots \times \mathcal{X}_d \subseteq \mathbb{R}^d$, and $\mathbf{x} = (x_1, \ldots, x_d)$ a point of $\mathcal{X}$. We use the subscript $-i$ to denote quantities excluding the i-th feature (e.g., $\mathbf{X}_{-i}$ defined on $\mathcal{X}_{-i}$). 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 $r_i : \mathcal{X}_{-i} \to \{1, \ldots, R_i\}$ chooses the specific shape function for x_i based on the context of the 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. We denote the prediction of a CALM by $\hat{y}(\mathbf{x}) = f_{\texttt{CALM}}(\mathbf{x}) = g^{-1}\big(\beta_0 + \sum_{i=1}^{d} f_i^{(r_i(\mathbf{x}_{-i}))}(x_i)\big)$.

Partition trees. Each selection function r_i partitions $\mathcal{X}_{-i}$, i.e., the input space excluding x_i, into regions $\mathcal{P}_i := \{\mathcal{R}_i^{(r)}\}_{r=1}^{R_i}$, independently for each feature 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. Each partition is represented by its own 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:

$$\mathcal{R}_i^{(r)} = \left\{ \mathbf{x}_{-i} \;\middle|\; \bigwedge_{k=1}^{m_i^{(r)}} (x_{j_k} \ \mathrm{op}_k \ \tau_k) \right\} \tag{2}$$

where $j_k \in \{1, \ldots, d\} \setminus \{i\}$ indexes an interacting feature, $\mathrm{op}_k \in \{<, \geq, =, \neq\}$ is a comparison operator, and τ_k is a threshold or a category for categorical features.

Centering. The shape functions are centered at feature level. In a GAM, each feature has one curve with zero mean over the data, and the intercept β_0 is the average prediction (on the scale of g). In a CALM, a feature has several curves, so we require them to jointly average to zero:

$$\mathbb{E}\big[f_i^{(r_i(\mathbf{X}_{-i}))}(X_i)\big] = \sum_{r=1}^{R_i} \mathbb{P}\big(\mathbf{X}_{-i} \in \mathcal{R}_i^{(r)}\big)\, \mathbb{E}\big[f_i^{(r)}(X_i) \,\big|\, \mathbf{X}_{-i} \in \mathcal{R}_i^{(r)}\big] = 0 \quad \text{for all } i. \tag{3}$$

We deliberately avoid centering each curve separately, as this would erase meaningful differences, e.g., the gap between the working-day and day-off curves of the hour in Figure 1.

High-order interactions. As each region conditions on up to $d_{\max}$ features, CALM captures interactions of up to $d_{\max} + 1$ features. For example, at $d_{\max} = 2$ the hour of the Bike Sharing data has one curve for working days of the second year (`workingday` $= 1 \wedge$ `yr` $= 1$, Figure 7). This three-feature interaction is modeled by a univariate curve, whereas a GA^{2}M cannot express it at all, and a model with three-way terms would need a 3-D surface.

Hyperparameters. Increasing $d_{\max}$ typically improves accuracy (Section 4.2), but at the cost of interpretability. To balance this trade-off, CALM exposes two hyperparameters: $d_{\max}$ (maximum tree depth), which bounds the number of shape functions per feature, $R_i \leq 2^{d_{\max}}$, and K, which bounds the total number of *conditional interactions* across all trees. For instance, the three-feature interaction of the Bike Sharing example above counts as two conditional interactions: (hour, working day) and (hour, year). Two thresholds, ϵ and γ, control how readily a split and a tree are accepted (Step 1 of Algorithm 1). In all experiments, we set $d_{\max} = 2$, $\epsilon = 0.05$, $\gamma = 0.01$, and leave K unconstrained.

Expressivity. CALM learns at most $2^{d_{\max}}$ curves per feature, yet, since every instance selects one region per feature, these curves combine into up to $\prod_i R_i \leq 2^{d \cdot d_{\max}}$ distinct additive models. Fitting these models one by one would split the data into up to $\prod_i R_i$ cells, leaving most of them with too few instances. CALM instead shares each curve across all the models whose region it covers, so every curve is still fitted on enough instances: about a $1/R_i$ share of the data for balanced regions. By contrast, a single tree shared by all features would yield at most $2^{d_{\max}}$ additive models.

3.1 Training algorithm for fitting CALM

Given a dataset $\mathcal{D} = \{(\mathbf{x}^{(k)}, y^{(k)})\}_{k=1}^{N}$, we fit a CALM by minimizing the empirical risk $\mathbb{E}_{(\mathbf{X},Y)\sim\hat{P}_{\mathcal{D}}}[\mathcal{L}(Y, \hat{y}(\mathbf{X}))]$, where $\mathcal{L}$ is a loss function and $\hat{P}_{\mathcal{D}}$ the empirical distribution over $\mathcal{D}$. This requires estimating (i) the region selection functions $r_i\ \forall i$, i.e., a partition tree T_i for each feature, and (ii) one shape function per region: $\{f_i^{(r)}\}_{r=1}^{R_i}$ and the intercept β_0. We solve this problem in two sequential steps summarized in Algorithm 1.

Step 1: Find the regions. Step 1 searches for a partition tree T_i per feature such that, within each region, the effect of x_i is well captured by a single univariate shape function, i.e., interactions are

Algorithm 1 Training a CALM model

Require: Training data $\mathcal{D} = \{(\mathbf{x}^{(k)}, y^{(k)})\}_{k=1}^{N}$
Output: CALM predictor $f_{\texttt{CALM}}(\mathbf{x})$
1: **Step 1:** Find the regions.
2: **1(a):** Train a reference model $f_{\texttt{ref}}$ on $\mathcal{D}$.
3: **1(b):** Propose a tree T_i for each feature with H_i at or above the median. {Eq. (5), Alg. 2}
4: **1(c):** Select the trees $\mathcal{S}$ greedily on the surrogate R^2. {Eq. (6), Alg. 3}
5: **Step 2:** Fit the shape functions $\{\{f_i^{(r)}\}_{r=1}^{R_i}\}_{i=1}^{d}$ and β_0. {Alg. 5}

locally weak. It has three substeps: Step 1(a) trains a reference model, Step 1(b) *proposes* a tree per feature, and Step 1(c) *selects* which trees enter the model.

Step 1(a) Train a reference model. We train a high-capacity black-box predictor $f_{\texttt{ref}} : \mathcal{X} \to \mathbb{R}$ on $\mathcal{D}$, as a proxy for $\mathbb{E}[Y|\mathbf{X}]$. Any accurate predictor can be used, such as gradient-boosted trees [26] (our default), neural networks, random forests [14] or foundation models like TabPFN [63].

Step 1(b) Propose a tree per feature. To find such regions, we use the interaction-related heterogeneity of GADGET [59]. Feature effect methods such as PDP or ALE explain a model like $f_{\texttt{ref}}$ through local effects $h(x_i, \mathbf{x}_{-i}^{(k)})$, the contribution of x_i to instance $\mathbf{x}^{(k)}$, and average them into a global effect $f_i(x_i)$. For PDP, our default, the local effect is the centered ICE curve [45], $h(x_i, \mathbf{x}_{-i}^{(k)}) = f_{\texttt{ref}}(x_i, \mathbf{x}_{-i}^{(k)}) - \mathbb{E}_{X_i}\left[f_{\texttt{ref}}(X_i, \mathbf{x}_{-i}^{(k)})\right]$. Heterogeneity is the spread of the local effects around their average: high where x_i interacts with other features, low in near-additive regions [54]. In a region $\mathcal{R} \subseteq \mathcal{X}_{-i}$ (e.g., $\mathcal{R}_i^{(r)}$), the pointwise heterogeneity of x_i is

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

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

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

The square root places $H_i^{\mathcal{R}}$ in the units of the output: it is the typical deviation of a local effect from the average effect of x_i within $\mathcal{R}$. ALE- and SHAP-DP-based variants are valid alternatives [56, 59], and RHALE [41] is preferable under strong feature correlations (Appendix A.2.1). We grow each T_i greedily, in a CART-style fashion, with $H_i^{\mathcal{R}}$ as the criterion. A tree is proposed only for features whose global heterogeneity $H_i^{\mathcal{X}_{-i}}$ is at or above the median; the others keep a single shape function. At each node, the split over the conditioning features x_j, $j \neq i$, that most reduces heterogeneity is accepted if the relative reduction exceeds ϵ; otherwise the node becomes a leaf (Appendix A.2.4). CART is one choice of partitioner and any other rule learners such as BRL [80] could replace it.

Theoretical justification. We show that, under specific assumptions, the error of CALM in approximating $f_{\texttt{ref}}$ is at most d times the sum of the expected squared feature heterogeneities (Proposition A.1). Step 1(b) lowers a computable proxy of this bound, the PDP heterogeneity of $f_{\texttt{ref}}$; whether it also lowers the predictive error is an empirical question, addressed in Section 4.2.

Step 1(c) Select the trees to keep. The trees of Step 1(b) are proposed one feature at a time and are often redundant: an interaction between x_i and x_j raises the heterogeneity of both features, so both trees split on each other, although one suffices. Keeping every candidate would double-count such interactions and increase the number of curves. We therefore keep the trees that most improve an additive surrogate of $f_{\texttt{ref}}$. For a set $\mathcal{S}$ of accepted trees, let $\mathcal{R}_i(\mathbf{x}_{-i})$ be the region of T_i that $\mathbf{x}_{-i}$ falls in if $i \in \mathcal{S}$, and $\mathcal{X}_{-i}$ otherwise. The surrogate sums the average local effects $\mu_i^{\mathcal{R}}$ of Eq. (4) on these regions, and its fidelity to $f_{\texttt{ref}}$ is

$$g_{\mathcal{S}}(\mathbf{x}) = c_{\mathcal{S}}(\mathbf{x}) + \sum_{i=1}^{d} \mu_i^{\mathcal{R}_i(\mathbf{x}_{-i})}(x_i), \qquad R^2(\mathcal{S}) = 1 - \frac{\mathbb{E}\left[\left(f_{\texttt{ref}}(\mathbf{X}) - g_{\mathcal{S}}(\mathbf{X})\right)^2\right]}{\mathrm{Var}\left[f_{\texttt{ref}}(\mathbf{X})\right]}, \tag{6}$$

where each $\mu_i^{\mathcal{R}}$ is centered over the instances of $\mathcal{R}$, $c_{\mathcal{S}}$ is a piecewise-constant offset over the regions of the trees in $\mathcal{S}$, fitted by least squares, and the expectations are over the training data. The curves are those already computed in Step 1(b); only $c_{\mathcal{S}}$ is fitted.

Starting from the GAM surrogate ($\mathcal{S} = \emptyset$), each round accepts the candidate with the largest gain in $R^2(\mathcal{S})$ on top of the trees already accepted, provided the gain is at least γ; selection stops when no candidate clears γ or when the next tree would exceed K conditional interactions (Appendix A.3, Algorithm 3). A redundant tree adds almost nothing on top of its partner and is discarded; this is what keeps CALMs small (Section 4.2), and K only truncates the same sequence. We stress that $R^2(\mathcal{S})$ measures in-sample fidelity to $f_{\texttt{ref}}$, not predictive accuracy, which is assessed on held-out data in Section 4.2. By construction, CALM cannot discover interactions that $f_{\texttt{ref}}$ has not captured; where $f_{\texttt{ref}}$ is near-additive, CALM remains a GAM.

Step 2: Fit the shape functions. Given the trees, we estimate the region-specific shape functions $\{f_i^{(r)}\}$ and β_0 by minimizing the empirical risk. We use a modified gradient boosting [16] in which each iteration updates a single shape function $f_i^{(r)}$, using only the instances whose $\mathbf{x}_{-i}$ falls in $\mathcal{R}_i^{(r)}$ (Algorithm 5).

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 characterizes its population target and the convergence of its idealized exact version, which gradient boosting approximates in practice; the proof is in Appendix A.4.1.

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 1) and $\mathbb{E}[Y^2] < \infty$. Let $\mathcal{H}(\{T_i\})$ denote the fixed-tree CALM class (Appendix A.4.1), and assume $\mathcal{H}(\{T_i\}) \subset L_2(P_{\mathbf{X}})$ is nonempty, closed, and convex. Then:*

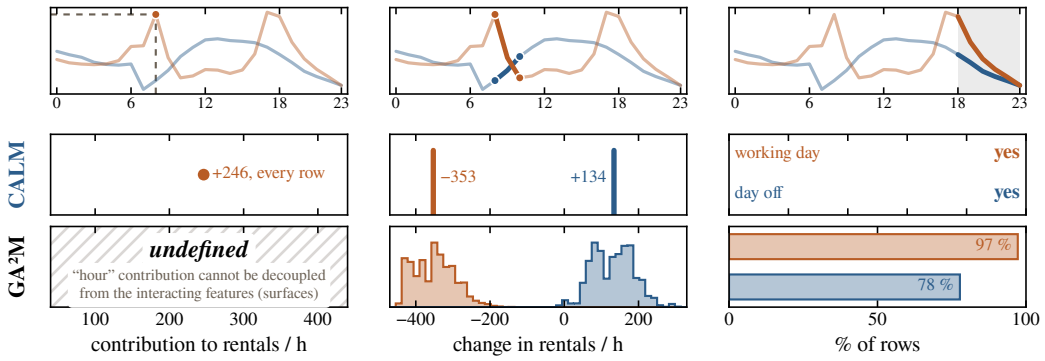
1. Optimality*: Any $s^\star \in \arg\min_{s \in \mathcal{H}(\{T_i\})} \mathbb{E}[(Y - s(\mathbf{X}))^2]$ is the $L_2(P_{\mathbf{X}})$-best approximation of m within $\mathcal{H}(\{T_i\})$.*
2. Convergence*: The idealized exact cyclic regional backfitting converges to an empirical risk minimizer over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

3.2 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 the* `hour` *on* `rentals` *follows the working-day curve of Figure 2 on a working day and the day-off curve on a day off.*

Interpreting a CALM plot. In Figure 2b each curve gives the contribution of the `hour` to `rentals` in a specific region; the orange curve on a working day and the blue curve on a day off. For example, at 8 am the contribution is $+246$ (working day) or -122 (day off). The plots also illustrate how altering x_i to $x_i \to x_i + \Delta x$ impacts the prediction. Vertical dotted lines mark points of a hidden discontinuity which is due to x_i participating as an interaction term for another feature. As shown in Figure 1, the effect of the `temperature` is conditioned by `hour` ≤ 6 and `hour` ≥ 7 (as is the year's), therefore in Figure 2b we observe a vertical line at 6.5. If a change in x_i does not cross a vertical line, the change in the output ($\Delta\hat{y}$) equals the curve difference (Δf_i). Crossing a line signifies a hidden jump, in the range $[\alpha, \beta]$, so $\Delta f_i + \alpha \leq \Delta\hat{y} \leq \Delta f_i + \beta$. Arrows provide a fast understanding of the jump: $\uparrow$ means $\Delta\hat{y} > \Delta f_i$, $\downarrow$ means $\Delta\hat{y} < \Delta f_i$, $\updownarrow$ means it depends. The jump at 6.5 is $[36, 334]$, always $\uparrow$. Below, we outline three crucial interpretability properties, along with discussion about their satisfiability by GAM, GA^{2}M, and CALM; Figure 3 answers each on the model of Figure 2. Complete proofs are provided in Appendix B.

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,



(a) P1: "hour" contribution at 8 am **(b) P2:** 8 → 10 am **(c) P3:** falls 6–11 pm?

Figure 3: The three properties on the model of Figure 2, each asked on every row of the data. Top: CALM's curves, the question marked. CALM's answers are read by eye; the GA^{2}M's are computed, each of the 17,379 rows through its eight terms with the hour. (a) The hour's contribution at 8 am on a working day: CALM reads $+246$ off the working-day curve, the same on every row; the GA^{2}M's is undefined, since the hour's contribution cannot be decoupled from the interacting features (its seven surfaces). (b) $\hat{y}(10\,\text{am}) - \hat{y}(8\,\text{am})$: CALM -353 on working days and $+134$ on days off, exact on every row since no dotted line is crossed; the GA^{2}M spreads from -430 to -257 and from $+54$ to $+215$ (5th–95th percentile). (c) Does the prediction fall every hour from 6 pm to 11 pm? CALM: yes on working days and days off, since both curves fall and cross no dotted line; the GA^{2}M: on 97% of working days and 78% of days off.

the contribution is neither explicit nor unique: the pairwise surfaces must be split post hoc, e.g. via SHAP [86] or LIME [99] (Figure 3a). In CALM, the contribution is $f_i^{(r_i(\mathbf{x}_{-i}))}(x_i)$.

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 over $[x_i, x_i + \Delta x]$. In GAM, the change is $\Delta\hat{y} = f_i(x_i + \Delta x) - f_i(x_i)$. In GA^{2}M, the change depends on every feature that interacts with x_i, one surface each (Figure 3b). 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 $\Delta\hat{y} := \{\Delta f_i^{(r)}\}_{r=1}^{R_i}$, one for each curve in the plot: $\Delta f_i^{(r)} = f_i^{(r)}(x_i + \Delta x) - f_i^{(r)}(x_i)$. 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, every row of every surface f_{ij} must be checked against the curve f_i (Figure 3c), an inspection infeasible for a human. In CALM, if no vertical lines are present, simply 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.

4 Empirical Evaluation

We evaluate CALM on three synthetic regression datasets and 24 public real-world tabular datasets, 10 for classification and 14 for regression. Every experiment runs CALM with its default configuration: XGBoost as the reference model in Step 1(a), PDP-based heterogeneity to propose the trees in Step 1(b), EBM to fit the curves in Step 2, and the default hyperparameters of Section 3 for Steps 1(b)–(c). No hyperparameter is tuned per dataset, for both CALM and the baselines. Appendix C provides full implementation details. Results are reported as the mean $\pm$ standard deviation over 5-fold cross-validation.

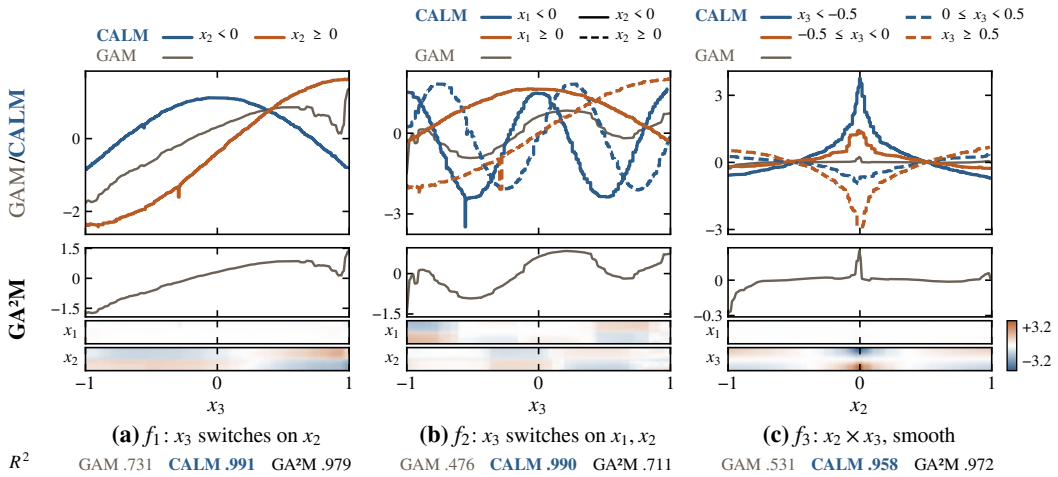


Figure 4: Cases 1 to 3 (one column each) of Section 4.1. Top: CALM's curves, one per region, over the GAM's single curve (gray). Bottom: the GA^{2}M's main curve alongside its two heatmaps. Under each column: the mean R^2 over 5 folds (Table 3), with CALM at depth 2; f_1's panel is drawn at depth 1.

4.1 Synthetic example

On three synthetic regression datasets, we compare CALM against EBM, as the GAM, and EB^{2}M (EBM with pairwise interactions), as the GA^{2}M. In all three, the dataset consists of 1,000 draws of three independent features $x_i \sim \mathcal{U}(-1, 1)$; only the target $y = f_k(\mathbf{x})$ varies and each f_k contains a different interaction structure: in Case 1, a conditional interaction of two features (one curve of x_3 per sign of x_2), which both CALM and a GA^{2}M can express; in Case 2, a conditional interaction of three features (one curve of x_3 per signs of x_1 and x_2), which only CALM can express; in Case 3, a general interaction of two features (a full x_2–x_3 term), which a GA^{2}M can express and CALM can only approximate by regions. The targets are

$$f_1(\mathbf{x}) = x_1^2 + \log|x_2| + 2\sin(\tfrac{\pi}{2}x_3)\,\mathbf{1}_{x_2\geq 0} + 2\cos(\tfrac{\pi}{2}x_3)\,\mathbf{1}_{x_2<0},$$
$$f_2(\mathbf{x}) = x_1^2 + \log|x_2| + 2\sin(\omega x_3)\,\mathbf{1}_{x_2\geq 0} + 2\cos(\omega x_3)\,\mathbf{1}_{x_2<0}, \quad \omega = \tfrac{\pi}{2}\,\mathbf{1}_{x_1\geq 0} + 2\pi\,\mathbf{1}_{x_1<0},$$
$$f_3(\mathbf{x}) = x_1^2 + \log|x_2|\sin(\tfrac{\pi}{2}x_3).$$

Figure 4 gives the R^2 of each model and Table 3 (Appendix C.2) its spread over the folds.

Case 1. The GAM averages the sine and cosine modes of x_3 into one curve (Fig. 4a, gray) and reaches $R^2 = 0.731$. Both the GA^{2}M and CALM capture the switch on the sign of x_2, with near-perfect accuracy (0.979 and 0.991).

Case 2. The GAM merges the four modes of x_3 into one curve, limiting its accuracy to $R^2 = 0.476$. The GA^{2}M can only approximate the three-feature switch with its x_1–x_3 and x_2–x_3 surfaces and reaches 0.711, while CALM conditions x_3 on both signs and reaches 0.990.

Case 3. Averaged over x_3, the effect of x_2 cancels, so the GAM's curve of x_2 is nearly flat and its accuracy low ($R^2 = 0.531$). The GA^{2}M captures the product in its x_2–x_3 surface (0.972). CALM approximates it by splitting x_3 into four ranges, each with its own curve of x_2, and still reaches 0.958, slightly below the GA^{2}M.

Interpretability. In every case, the GA^{2}M spreads the effect of one feature over three views, its main curve and two heatmaps, which the reader must combine to draw conclusions (Fig. 4, bottom). CALM shows the same effect as a few 1D curves, each labelled by the condition that selects it (Fig. 4, top), so the questions P1–P3 of Section 3.2 are answered at a glance.

4.2 Real-world datasets

On the real-world datasets, we compare CALM with the XGBoost black box [26], two GAMs (NAM and EBM) and two GA^{2}Ms: EB^{2}M (EBM with pairwise interactions) and NODE-GA^{2}M. We report

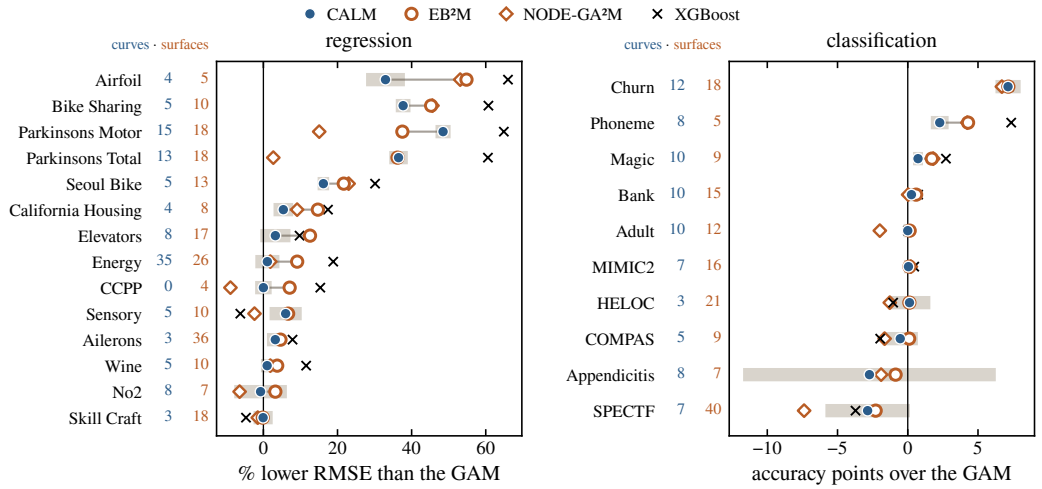


Figure 5: What each model gains over the GAM on each dataset: the black box XGBoost, the GA^{2}Ms NODE-GA^{2}M and EB^{2}M, and CALM (means over 5 folds, higher is better). The gray band behind each dot is $\pm \max(\text{std}_{\text{CALM}}, \text{std}_{\text{EB}^2\text{M}})$ over the folds. A ring inside it is a draw, outside a win or a loss. CALM ahead/level/behind: against the GAM, 8/6/0 in regression and 3/7/0 in classification; against EB^{2}M, 1/5/8 and 0/7/3 (see Tables 4 and 5 for full results). Left of each row: the curves that CALM adds on top of the GAM, and the pairwise surfaces that EB^{2}M fits.

accuracy for classification and RMSE for regression, and count a win or a loss only when two means differ by more than the larger of their standard deviations over the folds, a draw otherwise. Figure 5 shows what each model gains over the GAM on every dataset, and Tables 4 and 5 (Appendix C.3) give every baseline with its spread, including GAMI-Net.

Classification results. On the 10 classification datasets (Fig. 5, right; Table 4), CALM (i) matches or beats each GAM on all datasets, tying with NAM on 5 and EBM on 7; (ii) matches or beats each GA^{2}M on most datasets, losing to EB^{2}M on 3 and to NODE-GA^{2}M on 2; and (iii) matches or beats XGBoost on 6, losing on 4.

Regression results. On the 14 regression datasets (Fig. 5, left; Table 5), CALM (i) matches or beats each GAM on all datasets, tying with NAM on 4 and EBM on 6; (ii) matches or beats NODE-GA^{2}M on 10, but loses to EB^{2}M on 8; and (iii) matches or beats XGBoost on 3, losing on 11.

Summary. In both tasks, CALM improves over GAMs, closing the GAM-to-XGBoost gap by a median of 37 % in regression and of 31 % in classification. Against the GA^{2}Ms, it is competitive on most datasets, but it still trails: in regression, EB^{2}M is more accurate on 8 of the 14 datasets, on 7 of them by more than 5 points in Figure 5. We attribute this gap, like the one to XGBoost, to CALM's stricter constraint: one-dimensional curves instead of two-dimensional surfaces.

Complexity. We measure the complexity [89] of a CALM in two quantities: (i) the conditional interactions of Section 3, the features x_j that are part of the partition tree T_i of x_i (Appendix C.4) and (ii) the number of curves it adds to the GAM, which we compare with a GA^{2}M's pairs and surfaces. For (i), CALM uses 6.6 conditional interactions on average in regression and 6.8 in classification (Tables 6 and 7); for (ii), it adds 8.2 and 7.9 curves (Fig. 5, left of each row). EB^{2}M fits 14.3 and 15.2 pairs, one surface each. These numbers come from CALM's default configuration, which puts no cap on the number of interactions. CALM still finds a sparse structure, because a tree grows only when a split reduces heterogeneity enough (Step 1(b)), and the selection keeps only the trees that add fidelity to the reference model, i.e., that increase $R^2(\mathcal{S})$ (Step 1(c)). This is a key advantage of CALM: it adds an interaction only when the data asks for it.

Sensitivity to hyperparameters. We vary $d_{\max}$ and γ and find that only the depth affects accuracy and complexity substantially. Over the datasets where XGBoost beats the GAM, i.e., where there is room for improvement (12 of 14 in regression, 5 of 10 in classification), the median share of that gap that CALM closes grows from 17 % at $d_{\max} = 1$ to 37 % at $d_{\max} = 2$ and 49 % at $d_{\max} = 3$ in regression (17, 31 and 41 % in classification), while the median number of conditional interactions

roughly doubles per level (1.9, 4.5 and 9.6 in regression). On Bike Sharing, deeper trees keep paying: $d_{\max} = 6$ closes 85 % of the gap, against 62 % at the default depth, with 9.8 conditional interactions instead of 4.6 (Appendix C.5). In general, depth is a knob to trade interpretability for accuracy, and the default $d_{\max} = 2$ is a good compromise. Halving or doubling γ changes the accuracy beyond one standard deviation on at most two of the 24 datasets, so its default is a safe choice.

Runtime. On a laptop CPU (10 threads), CALM fits 19 of the 24 datasets in under a minute (median 21 s, at most 2 min, on Energy), against a median of 4 s for EBM on the same runs. Most of this time is spent proposing the trees (Step 1(b), $\mathcal{O}(d_{\max}\, d^2 N \log N)$) and fitting the shape functions (Step 2). NODE-GA^{2}M, timed earlier on other hardware, takes 1 to 2 minutes per fit (Tables 11–12).

4.3 Stability

We assess the stability of CALM: does refitting it return similar trees? This matters when CALM is used not only to predict with an interpretable-by-design model, but also to learn about the process that generated the data [96, 36]. We therefore refit CALM on the 5 cross-validation folds (each on 80 % of the rows) and on 20 bootstrap resamples, and ask three questions of increasing strictness: (i) is the feature split at all (the *interaction verdict*), (ii) on which feature is its root split (the *root conditioner*), and (iii) is its whole tree the same (the root and the depth-2 conditioners, the *full structure*); "not split" counts as an answer. Each question is asked separately of each of the 376 features of the 24 datasets. Table 1 gives the share of features on which all 5 refits agree, or at least 4; the bootstrap shares are averaged over 4 disjoint groups of 5 resamples. On whether a feature is split, 77 % of the features agree across all five folds and 90 % across four; agreement falls only a few points as the question becomes stricter, to 71 % and 81 % for the full structure, and the bootstrap, which perturbs the data more, lowers each share by one to three points. Where a rule recurs, its threshold barely moves: over the 88 rules found in at least 3 of the 5 folds, the median spread of the threshold is 0.08 standard deviations of the feature, and 73 % stay within 0.2.

	folds		bootstrap	
agreement on the	5/5	≥4/5	5/5	≥4/5
interaction verdict	77 %	90 %	74 %	89 %
root conditioner	74 %	87 %	71 %	86 %
full structure	71 %	81 %	68 %	80 %

Table 1: Share of the 376 features on which the refits of CALM agree.

4.4 User study

In an online study, 25 participants, from novices to experts in machine learning (13 self-reported advanced or expert), answered questions on the effect of the hour in a Bike Sharing model, once from CALM's conditional curves and once from a GA^{2}M's main effect and heatmaps [53]: how the prediction changes when the hour moves [50]. Participants answered 54 % of the questions correctly with CALM and 34 % with the GA^{2}M ($p < 0.05$), and the gap is widest on the question of how long the hour takes to lower the prediction (40 % vs. 12 %). The advantage holds among the 13 advanced or expert participants, for whom heatmaps are standard tooling (65.4 % vs. 30.8 %, Wilcoxon $p = 0.031$), and accuracy with the GA^{2}M does not rise with expertise (Spearman $\rho = -0.29$, $p = 0.14$); Appendix D gives the design, the questions and the grading. Order and dataset were not counterbalanced.

5 Conclusion

We introduced CALM, an interpretable-by-design model that narrows the gap between the interpretability of GAMs and the accuracy of GA^{2}Ms: it models interactions as *conditional feature effects*, one univariate curve per feature and region of its interacting features. On 24 datasets, CALM never loses to a GAM, adds only about eight curves to it and closes about a third of the gap to XGBoost; in a user study, it is read more accurately than a GA^{2}M. Its main limitation is that interpretability may degrade as interactions grow: although CALM uses only 6.6 conditional interactions on average in regression and 6.8 in classification, a feature that depends on many others gets many curves and vertical dotted lines, which make it harder to read [2]. Future work includes replacing the distillation pipeline with end-to-end training, e.g., a neural network under the same structural constraints, and studying the identifiability of CALM and its connection to the functional ANOVA decomposition [65, 79, 60]. Python package: `https://pypi.org/project/calm-additive`.

Acknowledgments and Disclosure of Funding

The research leading to this work has received funding from the European Union's Horizon Europe research and innovation program under Grant Agreement No. 101135826 (aidapt.eu).

The work leading to these results has been funded by the European Union under Grant Agreement No. 101289094 (project AIRIS). Views and opinions expressed are however those of the authors and do not necessarily reflect those of the European Union or the granting authority (HaDEA). Neither the European Union nor the granting authority can be held responsible for them.

The authors declare no competing interests.

References

[1] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from `https://www.tensorflow.org/`, 2015.

[2] Ashraf Abdul, Christian von der Weth, Mohan Kankanhalli, and Brian Y. Lim. COGAM: Measuring and moderating cognitive load in machine learning model explanations. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2020.

[3] 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.

[4] Elaine Angelino, Nicholas Larus-Stone, Daniel Alabi, Margo Seltzer, and Cynthia Rudin. Learning certifiably optimal rule lists for categorical data. *Journal of Machine Learning Research*, 18(234):1–78, 2018.

[5] Craig F. Ansley and Robert Kohn. Convergence of the backfitting algorithm for additive models. *Journal of the Australian Mathematical Society (Series A)*, 57(3):316–329, 1994.

[6] Daniel W Apley and Jingyu Zhu. Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 82(4):1059–1086, 2020.

[7] Hubert Baniecki, Giuseppe Casalicchio, Bernd Bischl, and Przemyslaw Biecek. On the robustness of global feature effect explanations. In *Machine Learning and Knowledge Discovery in Databases. Research Track (ECML PKDD 2024)*, volume 14942 of *Lecture Notes in Computer Science*, pages 125–142. Springer, 2024.

[8] Moritz Berger, Gerhard Tutz, and Matthias Schmid. Tree-structured modelling of varying coefficients. *Statistics and Computing*, 29(2):217–229, 2019.

[9] Przemysław Biecek, Hubert Baniecki, Mateusz Krzyziński, and Dianne Cook. Performance is not enough: The story told by a Rashomon quartet. *Journal of Computational and Graphical Statistics*, 33(3):1118–1121, 2024.

[10] Jacob Bien and Robert Tibshirani. Prototype selection for interpretable classification. *The Annals of Applied Statistics*, 5(4):2403–2424, 2011.

[11] R. K. Bock, A. Chilingarian, M. Gaug, F. Hakl, T. Hengstebeck, M. Jiřina, J. Klaschka, E. Kotrč, P. Savický, S. Towers, A. Vaiciulis, and W. Wittek. Methods for multidimensional event classification: a case study using images from a Cherenkov gamma-ray telescope. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 516(2–3):511–528, 2004.

[12] Sebastian Bordt and Ulrike von Luxburg. From Shapley values to generalized additive models and back. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 206 of *Proceedings of Machine Learning Research*, pages 709–745. PMLR, 2023.

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

[14] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[15] Thomas F. Brooks, D. Stuart Pope, and Michael A. Marcolini. Airfoil self-noise and prediction. NASA Reference Publication RP-1218, NASA, 1989.

[16] Peter Bühlmann and Torsten Hothorn. Boosting algorithms: Regularization, prediction and model fitting. *Statistical Science*, 22(4):477–505, 2007.

[17] Andreas Buja, Trevor Hastie, and Robert Tibshirani. Linear smoothers and additive models. *The Annals of Statistics*, 17(2):453–510, 1989.

[18] Reto Bürgin and Gilbert Ritschard. Coefficient-wise tree-based varying coefficient regression with vcrpart. *Journal of Statistical Software*, 80(6):1–33, 2017.

[19] Luis M. Candanedo. Appliances energy prediction. UCI Machine Learning Repository, 2017.

[20] 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.

[21] 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 21st ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1721–1730, 2015.

[22] Mustafa Cavus, Jan N. van Rijn, and Przemysław Biecek. Beyond the single-best model: Rashomon partial dependence profile for trustworthy explanations in AutoML. In *Discovery Science: 28th International Conference, DS 2025*, volume 16090 of *Lecture Notes in Computer Science*, pages 445–459. Springer, 2025.

[23] 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.

[24] Chun-Hao Chang, Sarah Tan, Ben Lengerich, Anna Goldenberg, and Rich Caruana. How interpretable and trustworthy are GAMs? In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 95–105, 2021.

[25] 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 *Advances in Neural Information Processing Systems*, volume 32, 2019.

[26] Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 785–794, 2016.

[27] Lorenzo Ciannelli, Kung-Sik Chan, Kevin M Bailey, and Nils Chr Stenseth. Nonadditive effects of the environment on the survival of a large marine fish population. *Ecology*, 85(12):3418–3427, 2004.

[28] Paulo Cortez, António Cerdeira, Fernando Almeida, Telmo Matos, and José Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4):547–553, 2009.

[29] Brent A Coull, David Ruppert, and M P Wand. Simple incorporation of interactions into additive models. *Biometrics*, 57(2):539–545, 2001.

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

[31] Abhimanyu Dubey, Filip Radenovic, and Dhruv Mahajan. Scalable interpretability via polynomials. In *Advances in Neural Information Processing Systems*, volume 35, pages 36748–36761, 2022.

[32] James Enouen and Yan Liu. Sparse interaction additive networks via feature interaction detection and sparse selection. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

[33] Fiona Katharina Ewald, Martin Binder, Matthias Feurer, Bernd Bischl, and Giuseppe Casalicchio. CASHomon sets: Efficient Rashomon sets across multiple model classes and their hyperparameters. *arXiv preprint arXiv:2603.15321*, 2026.

[34] Hadi Fanaee-T and João Gama. Event labeling combining ensemble detectors and background knowledge. *Progress in Artificial Intelligence*, 2(2–3):113–127, 2014.

[35] FICO. Explainable Machine Learning Challenge: Home Equity Line of Credit (HELOC) dataset. `https://community.fico.com/s/explainable-machine-learning-challenge`, 2018.

[36] Timo Freiesleben, Gunnar König, Christoph Molnar, and Álvaro Tejero-Cantero. Scientific inference with interpretable machine learning: Analyzing models to learn about real-world phenomena. *Minds and Machines*, 34(3):32, 2024.

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

[38] Jerome H. Friedman and Bogdan E. Popescu. Predictive learning via rule ensembles. *The Annals of Applied Statistics*, 2(3):916–954, 2008.

[39] Nicholas Frosst and Geoffrey Hinton. Distilling a neural network into a soft decision tree. *arXiv preprint arXiv:1711.09784*, 2017.

[40] Fabian Fumagalli, Maximilian Muschalik, Eyke Hüllermeier, Barbara Hammer, and Julia Herbinger. Unifying feature-based explanations with functional ANOVA and cooperative game theory. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 258 of *Proceedings of Machine Learning Research*, pages 5140–5148. PMLR, 2025.

[41] Vasilis Gkolemis, Theodore Dalamagas, Eirini Ntoutsi, and Christos Diou. RHALE: Robust and heterogeneity-aware accumulated local effects. In *ECAI 2023*, volume 372 of *Frontiers in Artificial Intelligence and Applications*, pages 859–866. IOS Press, 2023.

[42] Vasilis Gkolemis, Christos Diou, Dimitris Kyriakopoulos, Konstantinos Tsopelas, Julia Herbinger, Hubert Baniecki, Dimitrios Rontogiannis, Loukas Kavouras, Maximilian Muschalik, Theodore Dalamagas, Eirini Ntoutsi, Bernd Bischl, and Giuseppe Casalicchio. Effector: A Python package for global and regional feature effects. *arXiv preprint arXiv:2404.02629*, 2024.

[43] Vasilis Gkolemis, Christos Diou, Eirini Ntoutsi, and Theodore Dalamagas. Fast and accurate regional effect plots for automated tabular data analysis. In *Proceedings of Workshops at the 50th International Conference on Very Large Data Bases (VLDB 2024), Workshop on Tabular Data Analysis (TaDA)*, 2024.

[44] Vasilis Gkolemis, Anargiros Tzerefos, Theodore Dalamagas, Eirini Ntoutsi, and Christos Diou. Regionally additive models: Explainable-by-design models minimizing feature interactions. In *Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD 2023 International Workshops)*, volume 2134 of *Communications in Computer and Information Science*, pages 433–447. Springer, 2025.

[45] Alex Goldstein, Adam Kapelner, Justin Bleich, and Emil Pitkin. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, 24(1):44–65, 2015.

[46] Alicja Gosiewska and Przemysław Biecek. Do not trust additive explanations. *arXiv preprint arXiv:1903.11420*, 2019.

[47] Alicja Gosiewska, Anna Kozak, and Przemysław Biecek. Simpler is better: Lifting interpretability-performance trade-off via automated feature engineering. *Decision Support Systems*, 150:113556, 2021.

[48] 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.

[49] Dominik Hammann and Marc Wouters. Explainability versus accuracy of machine learning models: The role of task uncertainty and need for interaction with the machine learning model. *European Accounting Review*, 35(2):343–375, 2026.

[50] Peter Hase and Mohit Bansal. Evaluating explainable AI: Which algorithmic explanations help users predict model behavior? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5540–5552, 2020.

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

[52] Trevor Hastie and Robert Tibshirani. Varying-coefficient models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 55(4):757–779, 1993.

[53] Stefan Hegselmann, Thomas Volkert, Hendrik Ohlenburg, Antje Gottschalk, Martin Dugas, and Christian Ertmer. An evaluation of the doctor-interpretability of generalized additive models with interactions. In *Proceedings of the 5th Machine Learning for Healthcare Conference*, volume 126 of *Proceedings of Machine Learning Research*, pages 46–79, 2020.

[54] Timo Heiß, Julia Herbinger, Bernd Bischl, and Giuseppe Casalicchio. SAILS: Surrogate-based analysis of interactions via local effect smooths. *arXiv preprint arXiv:2606.09404*, 2026.

[55] Roel Henckaerts, Katrien Antonio, and Marie-Pier Côté. When stakes are high: Balancing accuracy and transparency with model-agnostic interpretable data-driven surrogates. *Expert Systems with Applications*, 202:117230, 2022.

[56] 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, 2022.

[57] Julia Herbinger, Susanne Dandl, Fiona K Ewald, Sofia Loibl, and Giuseppe Casalicchio. Leveraging model-based trees as interpretable surrogate models for model distillation. In *Artificial Intelligence. ECAI 2023 International Workshops*, volume 1947 of *Communications in Computer and Information Science*, pages 232–249. Springer, 2024.

[58] Julia Herbinger, Gabriel Laberge, Maximilian Muschalik, Yann Pequignot, Marvin N. Wright, and Fabian Fumagalli. GRANITE: A generalized regional framework for identifying agreement in feature-based explanations. In *Proceedings of The 29th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 300 of *Proceedings of Machine Learning Research*, pages 4888–4896. PMLR, 2026.

[59] 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.

[60] Munir Hiabu, Joseph T. Meyer, and Marvin N. Wright. Unifying local and global model explanations by functional decomposition of low dimensional structures. In *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 206 of *Proceedings of Machine Learning Research*, pages 7040–7060. PMLR, 2023.

[61] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[62] Anders Hjort, Ida Scheel, Dag Einar Sommervoll, and Johan Pensar. Locally interpretable tree boosting: An application to house price prediction. *Decision Support Systems*, 178:114106, 2024.

[63] Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. TabPFN: A transformer that solves small tabular classification problems in a second. In *International Conference on Learning Representations*, 2023.

[64] Giles Hooker. Discovering additive structure in black box functions. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 575–580, 2004.

[65] Giles Hooker. Generalized functional ANOVA diagnostics for high-dimensional functions of dependent variables. *Journal of Computational and Graphical Statistics*, 16(3):709–732, 2007.

[66] Linwei Hu, Jie Chen, Arpita Mukherjee, and Vijayan N Nair. Using model-based trees with boosting to fit low-order functional ANOVA models. *arXiv preprint arXiv:2207.06950*, 2022.

[67] Linwei Hu, Jie Chen, Vijayan N Nair, and Agus Sudjianto. Locally interpretable models and effects based on supervised partitioning (LIME-SUP). *arXiv preprint arXiv:1806.00663*, 2018.

[68] 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.

[69] Xin Huang, Jia Li, and Jun Yu. Cluster-based generalized additive models informed by random Fourier features. *arXiv preprint arXiv:2512.19373*, 2025.

[70] 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, 2023.

[71] Jingyun Jia, Chandan Singh, Rich Caruana, and Ben Lengerich. Selecting feature interactions for generalized additive models by distilling foundation models. *arXiv preprint arXiv:2604.13332*, 2026.

[72] Harmanpreet Kaur, Harsha Nori, Samuel Jenkins, Rich Caruana, Hanna Wallach, and Jennifer Wortman Vaughan. Interpreting interpretability: Understanding data scientists' use of interpretability tools for machine learning. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2020.

[73] Markelle Kelly, Rachel Longjohn, and Kolby Nottingham. The UCI machine learning repository. `https://archive.ics.uci.edu`, 2025.

[74] Ron Kohavi. Scaling up the accuracy of Naive-Bayes classifiers: A decision-tree hybrid. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, pages 202–207. AAAI Press, 1996.

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

[76] Sven Kruschel, Lasse Bohlen, Julian Rosenberger, Patrick Zschech, and Mathias Kraus. Quantifying visual properties of GAM shape plots: Impact on perceived cognitive load and interpretability. In *Proceedings of the 58th Hawaii International Conference on System Sciences*, 2025.

[77] Gabriel Laberge, Yann Pequignot, Mario Marchand, and Foutse Khomh. Tackling the XAI disagreement problem with regional explanations. In *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 238 of *Proceedings of Machine Learning Research*, pages 2017–2025. PMLR, 2024.

[78] Isaac Lage, Emily Chen, Jeffrey He, Menaka Narayanan, Been Kim, Samuel J. Gershman, and Finale Doshi-Velez. Human evaluation of models built for interpretability. In *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing*, volume 7, pages 59–67, 2019.

[79] Benjamin Lengerich, Sarah Tan, Chun-Hao Chang, Giles Hooker, and Rich Caruana. Purifying interaction effects with the functional ANOVA: An efficient algorithm for recovering identifiable additive models. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 108 of *Proceedings of Machine Learning Research*, pages 2402–2412. PMLR, 2020.

[80] 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):1350–1371, 2015.

[81] Sichao Li, Rong Wang, Quanling Deng, and Amanda S. Barnard. Exploring the cloud of feature interaction scores in a Rashomon set. In *International Conference on Learning Representations*, 2024.

[82] 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.

[83] 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.

[84] 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.

[85] Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1):56–67, 2020.

[86] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

[87] Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM Computing Surveys*, 54(6):1–35, 2021.

[88] Christoph Molnar. Interpretable Machine Learning: A guide for making black box models explainable. `https://christophm.github.io/interpretable-ml-book`, 2025. 3rd edition, ISBN 978-3-911578-03-5.

[89] Christoph Molnar, Giuseppe Casalicchio, and Bernd Bischl. Quantifying model complexity via functional decomposition for better post-hoc interpretability. In *Machine Learning and Knowledge Discovery in Databases: International Workshops of ECML PKDD 2019*, volume 1167 of *Communications in Computer and Information Science*, pages 193–204. Springer, 2020.

[90] Christoph Molnar, Gunnar König, Bernd Bischl, and Giuseppe Casalicchio. Model-agnostic feature importance and effects with dependent features: a conditional subgroup approach. *Data Mining and Knowledge Discovery*, 38(5):2903–2941, 2024.

[91] Christoph Molnar, Gunnar König, Julia Herbinger, Timo Freiesleben, Susanne Dandl, Christian A. Scholbeck, Giuseppe Casalicchio, Moritz Grosse-Wentrup, and Bernd Bischl. General pitfalls of model-agnostic interpretation methods for machine learning models. In *xxAI – Beyond Explainable AI*, volume 13200 of *Lecture Notes in Computer Science*, pages 39–68. Springer, 2022.

[92] Sérgio Moro, Paulo Cortez, and Paulo Rita. A data-driven approach to predict the success of bank telemarketing. *Decision Support Systems*, 62:22–31, 2014.

[93] W James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. Definitions, methods, and applications in interpretable machine learning. *Proceedings of the National Academy of Sciences*, 116(44):22071–22080, 2019.

[94] Harsha Nori, Samuel Jenkins, Paul Koch, and Rich Caruana. InterpretML: A unified framework for machine learning interpretability. *arXiv preprint arXiv:1909.09223*, 2019.

[95] R. Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics & Probability Letters*, 33(3):291–297, 1997.

[96] Michel Philipp, Thomas Rusch, Kurt Hornik, and Carolin Strobl. Measuring the stability of results from supervised statistical learning. *Journal of Computational and Graphical Statistics*, 27(4):685–700, 2018.

[97] ProPublica. COMPAS data and analysis for 'Machine Bias'. `https://github.com/propublica/compas-analysis`, 2016.

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

[99] 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.

[100] 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. *Bioinformatics*, 38(3):878–880, 2022.

[101] 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.

[102] 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: A public-access intensive care unit database. *Critical Care Medicine*, 39(5):952–960, 2011.

[103] Sathishkumar V E, Jangwoo Park, and Yongyun Cho. Using data mining techniques for bike sharing demand prediction in metropolitan city. *Computer Communications*, 153:353–366, 2020.

[104] Christian A. Scholbeck, Giuseppe Casalicchio, Christoph Molnar, Bernd Bischl, and Christian Heumann. Marginal effects for non-linear prediction functions. *Data Mining and Knowledge Discovery*, 38(5):2997–3042, 2024.

[105] Rickmer Schulte and David Rügamer. Additive model boosting: New insights and path(ologie)s. In *Proceedings of The 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pages 1–9, 2025.

[106] Heidi Seibold, Torsten Hothorn, and Achim Zeileis. Generalised linear model trees with global additive effects. *Advances in Data Analysis and Classification*, 13(3):703–725, 2019.

[107] Gero Szepannek and Karsten Lübke. How much do we see? On the explainability of partial dependence plots for credit risk scoring. *Argumenta Oeconomica*, 2023(1):137–150, 2023.

[108] 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.

[109] Sarah Tan, Giles Hooker, Paul Koch, Albert Gordo, and Rich Caruana. Considerations when learning additive explanations for black-box models. *Machine Learning*, 112(9):3333–3359, 2023.

[110] Joseph J. Thompson, Mark R. Blair, Lihan Chen, and Andrew J. Henrey. Video game telemetry as a critical tool in the study of complex skill learning. *PLoS ONE*, 8(9):e75129, 2013.

[111] A. Tsanas, M. A. Little, P. E. McSharry, and L. O. Ramig. Accurate telemonitoring of Parkinson's disease progression by noninvasive speech tests. *IEEE Transactions on Biomedical Engineering*, 57(4):884–893, 2010.

[112] Michael Tsang, Dehua Cheng, and Yan Liu. Detecting statistical interactions from neural network weights. In *International Conference on Learning Representations (ICLR)*, 2018.

[113] Pınar Tüfekci. Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods. *International Journal of Electrical Power & Energy Systems*, 60:126–140, 2014.

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

[115] Simon N Wood. *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC, 2nd edition, 2017.

[116] Guangzhi Xiong, Sanchit Sinha, and Aidong Zhang. Neural additive experts: Context-gated experts for controllable model additivity. In *Proceedings of the 29th International Conference on Artificial Intelligence and Statistics*, volume 300 of *Proceedings of Machine Learning Research*, pages 775–783, 2026.

[117] Zebin Yang, Aijun Zhang, and Agus Sudjianto. GAMI-Net: An explainable neural network based on generalized additive models with structured interactions. *Pattern Recognition*, 120:108192, 2021.

[118] Achim Zeileis, Torsten Hothorn, and Kurt Hornik. Model-based recursive partitioning. *Journal of Computational and Graphical Statistics*, 17(2):492–514, 2008.

A Conditionally Additive Local Models (CALMs)

Algorithm 1 fits a CALM (Eq. (1)) to a dataset $\mathcal{D} = \{\mathbf{x}^{(k)}, y^{(k)}\}_{k=1}^{N}$ in two steps. Step 1 finds the regions: (a) train a reference model $f_{\texttt{ref}}$ on $\mathcal{D}$, (b) propose a partition tree T_i per feature from a heterogeneity measure H_i, and (c) select the trees that enter the model. Step 2 fits the shape functions $\{f_i^{(r)}\}_{r=1}^{R_i}$ of each feature. We detail each step below.

A.1 Step 1(a): Train a reference model $f_{\texttt{ref}}$

Any sufficiently accurate model can serve as $f_{\texttt{ref}} : \mathbb{R}^d \to \mathbb{R}$, e.g., gradient-boosted trees (XGBoost [26], our default), neural networks, random forests [14], or TabPFN [63].

A.2 Step 1(b): Propose a partition tree T_i per feature

We first define the heterogeneity measure H_i (Section A.2.1) and relate it to CALM's error in approximating $f_{\texttt{ref}}$. Motivated by this result, we fit a candidate partition tree T_i for each feature i whose heterogeneity is at or above the median over all features (Section A.2.4).

A.2.1 Heterogeneity H_i

Interaction-related heterogeneity H_i [59] quantifies the extent to which a feature x_i interacts with all other features x_j, where $j \in \{1, \dots, d\} \setminus \{i\}$, like the H-statistic [38]. The computation of H_i is based on the variance of local effects: $h(x_i, \mathbf{x}_{-i}^{(j)})$.

Local effect. The local effect $h(x_i, \mathbf{x}_{-i}^{(j)})$ is the change in the output of $f_{\texttt{ref}}$ when the i-th feature of the j-th background observation is set to x_i [45]. Its form depends on the feature-effect method, e.g., Partial Dependence Plot (PDP), SHAP-Dependence Plot (SHAP-DP) or Robust and Heterogeneity-aware ALE (RHALE). We use PDP by default:

$$h\big(x_i,\, \mathbf{x}_{-i}^{(j)}\big) = f_{\texttt{ref}}\big(x_i,\, \mathbf{x}_{-i}^{(j)}\big) - c(\mathbf{x}_{-i}^{(j)}) \tag{7}$$

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 [59]. For details on alternative approaches, we refer the reader to [59, 56].

Correlated features. While PDP can suffer from extrapolation bias under feature correlations [91, 90], 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 [41] is a drop-in alternative. 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 [56].

Point-wise heterogeneity $H_i(x_i)$. The pointwise heterogeneity (heterogeneity at a specific point x_i) is then defined as

$$H_i(x_i) = \mathbb{E}_{\mathbf{X}_{-i}}\left[\big(h(x_i, \mathbf{X}_{-i}) - \mu_i(x_i)\big)^2\right] = \text{Var}_{\mathbf{X}_{-i}}\left[h(x_i, \mathbf{X}_{-i})\right] \tag{8}$$

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 H_i. Feature-level heterogeneity (heterogeneity of the feature) shows the overall strength of interactions of the feature with all other features, and is defined as the root of the expected pointwise heterogeneity, i.e., the root energy of the 1d curve of pointwise heterogeneity:

$$H_i = \sqrt{\mathbb{E}_{X_i}\left[H_i(X_i)\right]} = \sqrt{\mathbb{E}_{X_i}\left[\text{Var}_{\mathbf{X}_{-i}}[h(X_i, \mathbf{X}_{-i})]\right]} \tag{9}$$

Inside a region this definition takes the form of Eq. (5). The square root places H_i in the units of the output. In the proposition and proof below, the natural object is the squared quantity H_i^2, the expected variance of the local effects, and we write it explicitly.

A.2.2 Heterogeneity and CALM approximation error

Inside a region, the best single curve for x_i is the average of its local effects, and what this curve misses is exactly the heterogeneity left in the region. The proposition below sums these misses over regions and features into a bound on CALM's error in approximating $f_{\texttt{ref}}$, so a split that lowers the heterogeneity lowers the bound, i.e., raises a floor on CALM's R^2 with respect to $f_{\texttt{ref}}$; Step 1(b) lowers a computable proxy of this heterogeneity, the PDP heterogeneity of $f_{\texttt{ref}}$. For example, in Bike Sharing the effect of the hour differs between working days and days off, so a single curve for the hour misses much of it and its heterogeneity is high; split on working day, each curve fits its own rows and the heterogeneity drops.

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*

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

such that each $\psi_i^{r_i(\mathbf{x}_{-i})}$ captures the contribution of the i-th feature to the function's output (including interactions). Also, assume that $X_i \perp \mathbf{X}_{-i}$ within each region (Remark A.2) and 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 (10) *holds up to the residual approximation variance. Then,*

$$\min_{f_{CALM}} \mathcal{E} \leq d \sum_{i=1}^{d} \left(\sum_{r=1}^{R_i} P_{\mathcal{R}_i^{(r)}} \left(H_i^{\mathcal{R}_i^{(r)}} \right)^2 \right) \tag{10}$$

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 $\psi_i^{\mathcal{R}_i^{(r)}}(\mathbf{x})$ as in Eq. (5)*, with ψ_i in place of the local effect h.*

With zero heterogeneity in every region the bound is zero, the per-feature counterpart of GADGET's result that regions without interaction-related heterogeneity make the regional effects additive [59, Thm. 3 and Eq. 8].

Proof. For the approximation error we have

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

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

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

$$\mathcal{E} \leq d \sum_{i=1}^{d} \mathcal{E}_i^{r_i(\mathbf{x}_{-i})} \tag{11}$$

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

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

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

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

The cross-term equals zero because:

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

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

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

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:

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

Taking the expectation over X_i:

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

which gives

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

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

Since the squared heterogeneity is the minimum error for each region, we can estimate the feature-level error across regions as

$$\min\mathcal{E}_i=\sum_{r=1}^{R_i}P_{\mathcal{R}_i^{(r)}}\left(H_i^{\mathcal{R}_i^{(r)}}\right)^2$$

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

Using this result with Eq. (11), we have

$$\min_{f_{\texttt{CALM}}}\mathcal{E}\leq d\sum_{i=1}^{d}\left(\sum_{r=1}^{R_i}P_{\mathcal{R}_i^{(r)}}\left(H_i^{\mathcal{R}_i^{(r)}}\right)^2\right)$$

□

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.

The bound links heterogeneity to the error of a single curve inside a region, which motivates splitting on it. The additive approximation of $f_{\texttt{ref}}$ is the usual assumption of feature-effect methods, and the local effects and their heterogeneity are used only to find the regions, not to estimate $f_{\texttt{ref}}$.

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

$$\hat{H}_i^{\mathcal{I}}(x_i) = \frac{1}{|\mathcal{I}|} \sum_{j \in \mathcal{I}} \Big(h(x_i, \mathbf{x}_{-i}^{(j)}) - \mu(x_i) \Big)^2, \text{ where } \mu(x_i) = \frac{1}{|\mathcal{I}|} \sum_{j \in \mathcal{I}} h(x_i, \mathbf{x}_{-i}^{(j)}) \tag{12}$$

Eq.(12) 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:

$$\hat{H}_i^{\mathcal{I}} = \sqrt{\frac{1}{M} \sum_{m=1}^{M} \hat{H}_i^{\mathcal{I}}(\tilde{x}_i^{(m)})}. \tag{13}$$

The square root returns the heterogeneity to the units of the output, so that $\hat{H}_i^{\mathcal{I}}$ is the typical deviation of a local effect from the average effect over $\mathcal{I}$; the relative drop of Eq. (14) below is computed on this scale.

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

For each feature x_i whose heterogeneity $\hat{H}_i$ over the full training data is at or above the median over all features, we construct a binary tree T_i of maximum depth $d_{\max}$; the remaining features keep a single global shape function. At each internal node of the tree, we evaluate candidate binary splits on every other feature x_j, $j \neq i$, as in CART [13]. The best split is the one with the largest heterogeneity drop: a candidate split partitions the current instances $\mathcal{I}$ into $\mathcal{I}_L$ and $\mathcal{I}_R$, and

$$\Delta H_i = \frac{H_i^{\mathcal{I}} - \left(\frac{|\mathcal{I}_L|}{|\mathcal{I}|} H_i^{\mathcal{I}_L} + \frac{|\mathcal{I}_R|}{|\mathcal{I}|} H_i^{\mathcal{I}_R} \right)}{H_i^{\mathcal{I}}} \tag{14}$$

where $H_i^{\mathcal{I}_L}$ and $H_i^{\mathcal{I}_R}$ are the heterogeneities of x_i on the two children, weighted by their share of the instances. The drop is relative, so the threshold ϵ reads as a fraction: we use $\epsilon = 0.05$, i.e., a split must reduce the heterogeneity of the node by at least 5% on the standard-deviation scale of Eq. (13). A split is also rejected if either child would contain fewer than 10 training instances. For numerical conditioning features we evaluate $M = 20$ equally spaced candidate thresholds; for categorical conditioning features we evaluate one-vs-rest splits over the observed levels.

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

A.3 Step 1(c): Select the trees to keep

The trees of Section A.2.4 are proposed one feature at a time and are therefore candidates: an interaction between x_i and x_j raises the heterogeneity of both features [59, Thm. 2], so both trees typically split on each other, although one of them suffices to express the interaction. Applying every candidate double-counts such interactions in the regional curves; the resulting model is larger and can be *less* faithful to $f_{\texttt{ref}}$ than a model that keeps only one of the two trees. We therefore select the subset $\mathcal{S}$ of trees that enter the model by greedy forward selection on an additive surrogate of $f_{\texttt{ref}}$.

The surrogate. For a candidate set $\mathcal{S}$, every feature contributes the average local effect of $f_{\texttt{ref}}$ computed in Section A.2.1: the global curve $\mu_i(x_i)$ if $i \notin \mathcal{S}$, and the regional curve $\mu_i^{\mathcal{R}_i^{(r)}}(x_i)$ of the region $\mathcal{R}_i^{(r)}$ that $\mathbf{x}_{-i}$ falls in if $i \in \mathcal{S}$. Each curve is centered over the instances it applies to, so that the curves carry only shape. What the curves leave over is absorbed by a piecewise-constant offset $c_{\mathcal{S}}(\mathbf{x})$: a global intercept plus one coefficient per region of every tree in $\mathcal{S}$ (one region per tree is dropped to keep the design full rank), fitted jointly by least squares to the residual of the curves on the training data. With a single tree the offsets reduce to the per-region means of the residual; with several trees the joint fit assigns a shared mean shift once, rather than to each overlapping

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.(13)
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**
      $\mathcal{I}_L \leftarrow \{k \in \mathcal{I} : x_j^{(k)} < \tau\}, \mathcal{I}_R \leftarrow \mathcal{I} \setminus \mathcal{I}_L$
    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. (14)
    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

region. The surrogate is $g_{\mathcal{S}}(\mathbf{x}) = c_{\mathcal{S}}(\mathbf{x}) + \sum_{i=1}^{d} \mu_i^{\mathcal{R}_i(\mathbf{x}_{-i})}(x_i)$, in the notation of Eq. (6), and its fidelity [109, 107] is

$$R^2(\mathcal{S}) = 1 - \frac{\frac{1}{N}\sum_{k=1}^{N}\left(f_{\texttt{ref}}(\mathbf{x}^{(k)}) - g_{\mathcal{S}}(\mathbf{x}^{(k)})\right)^2}{\mathrm{Var}_k\left[f_{\texttt{ref}}(\mathbf{x}^{(k)})\right]}, \tag{15}$$

which may be negative and is not clamped. $R^2(\emptyset)$ is the fidelity of the GAM surrogate, and $R^2(\mathcal{S}) - R^2(\emptyset)$ is what the trees in $\mathcal{S}$ add. Scoring a candidate set requires no further evaluations of $f_{\texttt{ref}}$: the curves are read from the local effects already computed for the heterogeneity.

Greedy forward selection. Algorithm 3 starts from $\mathcal{S} = \emptyset$ and, in each round, scores every remaining candidate on top of the trees already accepted. The candidate with the largest $R^2(\mathcal{S} \cup \{i\})$ is accepted if its gain over $R^2(\mathcal{S})$ is at least γ; otherwise the procedure stops. It also stops when the next acceptance would exceed the cap K on the number of conditional interactions, counted as one per (feature, conditioning feature) pair appearing in the accepted trees; a capped selection is therefore a prefix of the uncapped one. Because each gain is measured on top of the current set, the gains of the accepted trees sum exactly to $R^2(\mathcal{S}) - R^2(\emptyset)$, and a candidate whose interaction is already expressed by an accepted tree shows a gain close to zero. Rejected candidates are recorded with the reason: *redundant* if their gain is close to zero, *below threshold* if it is real but smaller than γ, or the cap that stopped the sequence. In all experiments, $\gamma = 0.01$, i.e., a tree must explain at least one additional percentage point of the variance of $f_{\texttt{ref}}$, and K is unconstrained.

The output of the selection is a sequence of models, from the GAM surrogate to the final CALM, with the gain of every accepted tree and the reason for every rejected one. We report it alongside each

Algorithm 3 Selecting the trees to keep

Input: candidate trees $\{T_i\}_{i \in \mathcal{C}}$, local effects of $f_{\texttt{ref}}$ on $\mathcal{D}$; **Parameters:** gain threshold γ, cap K (optional)
Output: the set $\mathcal{S} \subseteq \mathcal{C}$ of trees that enter the model
$\mathcal{S} \leftarrow \emptyset$, $\rho \leftarrow R^2(\emptyset)$ {the GAM surrogate, Eq. (15)}
while $\mathcal{C} \setminus \mathcal{S} \neq \emptyset$ **do**
  $i^\star \leftarrow \arg\max_{i \in \mathcal{C} \setminus \mathcal{S}} R^2(\mathcal{S} \cup \{i\})$ {score every candidate on top of $\mathcal{S}$}
  if $R^2(\mathcal{S} \cup \{i^\star\}) - \rho < \gamma$ **then**
    break {no candidate clears the gain threshold}
  end if
  if K is set and the conditional interactions of $\mathcal{S} \cup \{i^\star\}$ exceed K **then**
    break {the cap stops the sequence}
  end if
  $\mathcal{S} \leftarrow \mathcal{S} \cup \{i^\star\}$, $\rho \leftarrow R^2(\mathcal{S})$
end while
return $\mathcal{S}$

Algorithm 4 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 residuals}
    Learn shape 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

fitted model, since it states in one table which interactions the model carries and what each of them contributes to the fidelity to $f_{\texttt{ref}}$.

A.4 Step 2: Fit the shape functions

A GAM is fitted by round-robin gradient boosting [37]: each iteration updates one shape function f_i on the current residuals of the whole dataset (Algorithm 4, as in [83]).

For a CALM, each update of $f_i^{(r)}$ uses only the instances in its region $\mathcal{R}_i^{(r)}$, so that it captures only the behaviour of x_i there (Algorithm 5).

Algorithm 5 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 shape 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, let $c_i \leftarrow \frac{1}{N}\sum_{j=1}^N f_i^{(r_i(\mathbf{x}_{-i}^{(j)}))}(x_i^{(j)})$; set $f_i^{(r)} \leftarrow f_i^{(r)} - c_i$ for all r and $\beta_0 \leftarrow \beta_0 + c_i$ {Eq. (17); fitted values unchanged}
end for

A.4.1 Theoretical analysis of Step 2

This section formalizes the claims made in Step 2 of the main text. We show that, under squared loss and fixed region-selection trees, (i) the population target of Step 2 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

$$s(\mathbf{x}) = \beta_0 + \sum_{i=1}^{d} f_i^{(r_i(\mathbf{x}_{-i}))}(x_i), \tag{16}$$

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 $\mathbb{E}[Y^2] < \infty$, hence $m \in L_2(P_{\mathbf{X}})$. We restrict attention to scores $s \in L_2(P_{\mathbf{X}})$.

The representation (16) is not unique: every $\mathbf{x}$ activates exactly one curve per feature, so adding a constant to all curves of feature i and subtracting it from β_0 leaves s unchanged. As in GAMs, we fix this constant by the centering condition

$$\mathbb{E}\Big[f_i^{(r_i(\mathbf{X}_{-i}))}(X_i)\Big] = 0 \quad \text{for all } i, \tag{17}$$

which moves the mean contribution of each feature into β_0 and leaves the level of every region inside its curve. The condition is imposed per feature, not per region: centering each $f_i^{(r)}$ within its own region would also remove the region levels, which a single intercept cannot absorb, and would therefore shrink the class. Constants can still move between the curves of x_i and the terms of another feature whenever the indicator of a region of x_i can also be expressed by those terms (e.g., a region defined by a single split on x_j, which f_j can absorb as a step). Such moves change neither s nor any prediction difference; the results below concern s and do not require the components to be unique.

Let $\mathcal{H}(\{T_i\})$ denote the class of all CALM score functions of the form (16). With the trees fixed, it is a linear space, hence convex; we assume $\mathcal{H}(\{T_i\}) \subset L_2(P_{\mathbf{X}})$ is closed. Condition (17) selects a representation of each $s \in \mathcal{H}(\{T_i\})$ and does not restrict the class.

A.4.2 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. Moreover, $s^\star$ is unique as an element of $L_2(P_{\mathbf{X}})$, being the projection of m onto a closed convex set; its components are unique only up to the moves described above. □

Interpretation. Proposition A.3 shows that, with fixed regions, Step 2 computes an $L_2(P_{\mathbf{X}})$-best approximation of the true regression function m within the CALM function class. In the population, any remaining error is therefore due to the structural limitations of the chosen regions and univariate components, not to the optimization procedure.

A.4.3 Empirical convergence of exact Step 2

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 2 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 the empirical version of* (17)*: the sample mean $\frac{1}{N}\sum_{k=1}^{N} f_i^{(r_i(\mathbf{x}_{-i}^{(k)}))}(x_i^{(k)})$ of feature i's contribution is subtracted from every curve of x_i and added to 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 backfitting with projection smoothers [17, 5] 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 step changes no fitted value; it only fixes the constant of each feature. □

Interpretation. This result shows that Step 2 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 [17]. Under exact updates, this procedure is guaranteed to converge to the best CALM fit for the given data and fixed regions; with base learners linear in their parameters, boosting provably converges to it as well [105].

A.5 Further related work

Several models give a feature more than one univariate curve. Neural Additive Experts [116] mix several expert curves per feature through a soft gate learned end to end over all features, so which curve applies cannot be read as a rule. Locally interpretable tree boosting [62] fits one GAM per level of a given grouping variable, and cluster-based GAMs [69] fit one GAM per latent regime found from a fitted random-feature model; in both, all features share the same regions. Threshold GAMs [27] switch the curve of a chosen feature at a threshold on other variables, and factor-by-curve terms [29] give one curve per level of a given factor, whereas CALM learns which features condition each curve. Closest in structure are tree-based variants of varying-coefficient models [52]: TSVC [8] and vcrpart [18] grow one tree per feature over other variables, but their leaves hold a slope rather than a curve, and the trees are grown by likelihood-based splits on the data rather than on the local effects of a reference model. Model-based recursive partitioning [118, 106] and LIME-SUP [67] instead grow a single tree shared by all varying parameters. Higher-order additive models go beyond pairs: SIAN [32] adds the feature subsets found by interaction detection as terms, and SPAM [31] models all interactions up to a chosen degree with low-rank polynomials; both represent an interaction with a term over several features, whereas CALM represents it with univariate curves. Like CALM, maidrr [55], SAFE [47] and TabDistill [71] use a black box only to decide the structure of an interpretable model that is then fitted to the data, but none of them has regions. Prior user studies found that clinicians and data scientists can misread GAM and GA^{2}M plots [53, 72], and that reading effort grows with the number of curves and kinks [2, 76]. Finally, equally accurate models can differ in their effects, interactions and importances [9, 81, 22, 33], which is why we report the stability of CALM's trees (Section 4.3).

B Formal Proofs for Interpretability Properties

This appendix provides formal characterizations and proofs for the interpretability properties discussed in Section 3.2. 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 $\eta(\mathbf{x}) = \beta_0 + \sum_i f_i^{(r_i(\mathbf{x}_{-i}))}(x_i)$. With a non-identity link g (Eq. (1)), they hold on η, the scale on which the model is additive; monotonicity (P3) also carries over to $\hat{y} = g^{-1}(\eta)$ for any increasing link g (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, unless every f_{jk} is additively separable, there is no decomposition

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

with each $\phi_i(\mathbf{x})$ depending only on x_i; decompositions whose ϕ_i also depend on other features exist (e.g., assign each f_{jk} to one of its two features) but are not unique [60, 40, 12].

(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 read off the model 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 [46].

(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:

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

□

As in any additive model, the components, and hence the ϕ_i, are defined only up to moves of constants that leave $\hat{y}$ unchanged, which the centering of Eq. (17) fixes up to the moves described in Appendix A.4.1; the point of P1 is that in GAM and CALM the contributions are read off the fitted model, whereas in GA^{2}M they must be constructed post hoc.

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 the features x_j that interact with x_i, and thus, in general, 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 no region transitions are triggered in other features. Otherwise, the change depends in general on $\mathbf{x}_{-i}$.*

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:

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

and before perturbation:

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

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 x_j for $j \in \mathcal{J}$, it cannot, in general, 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. It holds for every $\mathbf{x}_{-i}$ when no tree T_j, $j \neq i$, splits on x_i at a threshold in $[x_i, x_i + \varepsilon)$, i.e., when the perturbation crosses no vertical line of the plot.

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, in general, 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 values of the features other than x_i, 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; under condition 2, each of them is non-negative.

Therefore, if both conditions are satisfied, every term is non-negative, implying $\Delta\hat{y} \geq 0$. The conditions are sufficient but not necessary: a downward jump can be offset by an increase of $f_i^{(r)}$ at the same point.

□

C Detailed Experiments

C.1 Experimental Setup Details

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

Implementations. We used open-source implementations for all baselines: XGBoost [26], tensorflow [1] (NAM [3]), interpretml [94] (EBM, EB^{2}M), and the official repositories for NODE-GA^{2}M [23] and GAMI-Net [117].

Table 2: The 24 real-world datasets: number of rows, number of features and source.

Classification (10)				**Regression (14)**			
Dataset	Rows	Features	Source	Dataset	Rows	Features	Source
Adult	45 222	13	UCI [73, 74]	Ailerons	13 750	39	OpenML [114]
Appendicitis	106	7	PMLB [100]	Airfoil	1 503	5	OpenML [114, 15]
Bank	45 211	16	UCI [73, 92]	Bike Sharing	17 379	11	UCI [73, 34]
Churn	5 000	19	PMLB [100]	California Housing	20 640	8	[95]
COMPAS	6 167	9	[97]	CCPP	9 568	4	UCI [73, 113]
HELOC	10 459	23	[35]	Elevators	16 599	18	OpenML [114]
Magic	19 020	10	UCI [73, 11]	Energy	19 735	28	[19, 20]
MIMIC2	24 508	17	[102]	No2	500	7	OpenML [114]
Phoneme	5 404	5	PMLB [100]	Parkinsons Motor	5 875	19	UCI [73, 111]
SPECTF	349	44	PMLB [100]	Parkinsons Total	5 875	19	UCI [73, 111]
				Sensory	576	11	OpenML [114]
				Seoul Bike	8 465	14	UCI [73, 103]
				Skill Craft	3 338	19	UCI [73, 110]
				Wine	6 497	11	UCI [73, 28]

Dataset details. Table 2 lists the 10 classification and 14 regression datasets with their size and source.

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.

The black box is XGBoost with 300 boosting rounds, a learning rate of 0.1 and the log-loss evaluation metric for classification; CALM uses the same model as its reference model.

The GAMs are NAM and EBM without interactions. 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. 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 with $\lceil 0.9\, d \rceil$ pairwise interactions, i.e., `interactions=0.9` in interpret; NODE-GA^{2}M activates its interaction mode 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; GAMI-Net is run with the default hyper-parameters.

For CALM, we use trees of maximum depth $d_{\max} = 2$, a heterogeneity drop threshold of $\epsilon = 0.05$ on the standard-deviation scale of Eq. (13), a selection threshold of $\gamma = 0.01$, and no cap K on the number of conditional interactions (Section A.3). Heterogeneity is measured with PDP, as implemented in Effector [42]; the detector evaluates 20 candidate split points per numerical conditioning feature and one-vs-rest splits for categorical ones, with at least 10 instances per leaf. The shape functions of Step 2 are fitted with an EBM without interactions on a masked design: one column per feature and region, missing outside its region. Each curve is its column's shape function

plus the missing-value scores of the feature's other columns, so the gaps between curves survive and the centering stays per feature.

Computer resources. XGBoost, EBM, EB^{2}M and CALM were run on a laptop (Intel Core i7-10750H, 10 CPU threads, 16 GB of RAM); NAM, NODE-GA^{2}M and GAMI-Net are reported from an earlier run on a server (Intel Core i9-10900X, 128 GB of RAM); EB^{2}M's runtimes in Tables 11 and 12 are from the earlier run too. No GPU was used.

C.2 Synthetic Examples

Table 3 gives the accuracy that Figure 4 prints under each case, with its spread over the folds.

Table 3: Synthetic examples: R^2 comparison (mean $\pm$ std. dev. over 5-fold CV).

Dataset	GAM	GA^{2}M	CALM
Case 1	$0.731_{\pm 0.027}$	$0.979_{\pm 0.010}$	$0.991_{\pm 0.006}$
Case 2	$0.476_{\pm 0.061}$	$0.711_{\pm 0.039}$	$0.990_{\pm 0.008}$
Case 3	$0.531_{\pm 0.023}$	$0.972_{\pm 0.006}$	$0.958_{\pm 0.012}$

C.3 Main Results, per Dataset

Tables 4 and 5 give the comparison of Figure 5 for every baseline. In regression the GA^{2}M's extra gain over CALM is beyond the band on eight datasets and exceeds five points on seven of them (Airfoil, Bike Sharing, Seoul Bike, California Housing, Elevators, Energy, CCPP); on Parkinsons Motor CALM is ahead, and on the remaining five the two are within a fold's spread. In classification every difference but Phoneme, Magic and Bank is inside the band.

Table 4: Classification accuracy vs. baselines (mean $\pm$ std. dev. over 5-fold CV). ✓/✗: higher/lower than CALM by more than one std. dev. (the larger of the two); =: within one.

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	$0.664_{\pm 0.009}$ ✗	$0.682_{\pm 0.016}$ =	$0.683_{\pm 0.011}$ =	$0.678_{\pm 0.013}$	$0.684_{\pm 0.012}$ =	$0.667_{\pm 0.013}$ =	$0.686_{\pm 0.013}$ =
HELOC	$0.717_{\pm 0.013}$ =	$0.723_{\pm 0.011}$ =	$0.728_{\pm 0.013}$ =	$0.729_{\pm 0.015}$	$0.729_{\pm 0.011}$ =	$0.715_{\pm 0.008}$ =	$0.725_{\pm 0.012}$ =
MIMIC2	$0.890_{\pm 0.003}$ ✓	$0.886_{\pm 0.001}$ =	$0.886_{\pm 0.003}$ =	$0.886_{\pm 0.003}$	$0.887_{\pm 0.002}$ =	$0.888_{\pm 0.002}$ =	$0.884_{\pm 0.003}$ =
Appendicitis	$0.859_{\pm 0.073}$ =	$0.848_{\pm 0.056}$ =	$0.868_{\pm 0.046}$ =	$0.841_{\pm 0.090}$	$0.859_{\pm 0.077}$ =	$0.849_{\pm 0.016}$ =	$0.764_{\pm 0.074}$ =
Phoneme	$0.900_{\pm 0.004}$ ✓	$0.808_{\pm 0.002}$ ✗	$0.827_{\pm 0.007}$ ✗	$0.849_{\pm 0.005}$	$0.870_{\pm 0.006}$ ✓	$0.869_{\pm 0.003}$ ✓	$0.876_{\pm 0.009}$ ✓
SPECTF	$0.857_{\pm 0.028}$ =	$0.839_{\pm 0.030}$ =	$0.894_{\pm 0.037}$ =	$0.865_{\pm 0.030}$	$0.871_{\pm 0.029}$ =	$0.820_{\pm 0.067}$ =	$0.874_{\pm 0.016}$ =
Magic	$0.885_{\pm 0.004}$ ✓	$0.850_{\pm 0.006}$ ✗	$0.858_{\pm 0.004}$ ✗	$0.865_{\pm 0.002}$	$0.875_{\pm 0.004}$ ✓	$0.876_{\pm 0.004}$ ✓	$0.874_{\pm 0.003}$ ✓
Bank	$0.910_{\pm 0.004}$ ✓	$0.901_{\pm 0.003}$ ✗	$0.903_{\pm 0.002}$ =	$0.905_{\pm 0.002}$	$0.909_{\pm 0.001}$ ✓	$0.903_{\pm 0.001}$ =	$0.908_{\pm 0.003}$ =
Churn	$0.959_{\pm 0.005}$ =	$0.885_{\pm 0.009}$ ✗	$0.887_{\pm 0.005}$ ✗	$0.958_{\pm 0.005}$	$0.959_{\pm 0.009}$ =	$0.954_{\pm 0.009}$ =	$0.952_{\pm 0.007}$ =
W/D/L	**4/5/1**	**0/5/5**	**0/7/3**		**3/7/0**	**2/7/1**	**2/7/1**

C.4 Complexity

Figure 6 plots the counts of Figure 5 against the gains: what each model asks a reader to inspect beyond the GAM, against what it gains over it. Tables 6 and 7 give the counts per dataset. For CALM, the conditional interactions are $\sum_i |S_i|$, where S_i is the set of features that split the tree T_i, and the added curves are those beyond the GAM's one per feature, one per split. On average, CALM uses 6.8 conditional interactions in classification and 6.6 in regression and adds 7.9 and 8.2 curves, against 15.2 and 14.3 pairs for EB^{2}M; NODE-GA^{2}M uses 97.7 and 88.7 pairs, GAMI-Net its default cap of 20.

C.5 Sensitivity to hyperparameters

We vary one hyperparameter at a time, the others at their defaults, and compare each variant with the default CALM by the rule of Section 4.

Table 5: Regression RMSE vs. baselines (mean ± std. dev. over 5-fold CV). ✓/✗: lower/higher than CALM by more than one std. dev. (the larger of the two); =: within one. Elevators and Ailerons are scaled; NAM, NODE-GA^{2}M and GAMI-Net there are from an earlier run that kept one significant digit, too few to rank them against CALM, so they count as draws.

Dataset	BB	GAM		CALM	GA^{2}M		
	XGB	NAM	EBM		EB^{2}M	NODE	GAMI
Bike Sharing	39.35±1.38 ✓	101.97±1.31 ✗	100.17±0.99 ✗	62.40±1.99	54.83±1.14 ✓	54.47±1.58 ✓	53.44±1.90 ✓
California Housing	0.45±0.01 ✓	0.61±0.01 ✗	0.55±0.01 ✗	0.52±0.01	0.47±0.01 ✓	0.50±0.01 ✓	0.51±0.04 =
Parkinsons Motor	1.44±0.09 ✓	6.11±0.16 ✗	4.11±0.04 ✗	2.12±0.08	2.57±0.06 ✗	3.49±0.31 ✗	2.76±0.31 ✗
Parkinsons Total	1.86±0.08 ✓	7.90±0.10 ✗	4.73±0.06 ✗	3.00±0.12	3.02±0.07 =	4.60±0.53 ✗	3.81±0.66 ✗
Seoul Bike	209.6±3.47 ✓	320.2±4.38 ✗	300.0±3.68 ✗	251.4±4.65	234.9±2.58 ✓	231.0±4.23 ✓	245.82±8.45 =
Wine	0.62±0.01 ✓	0.72±0.01 ✗	0.70±0.01 =	0.70±0.01	0.68±0.01 ✓	0.69±0.01 =	0.71±0.00 =
Energy	67.97±2.58 ✓	89.80±2.67 ✗	83.75±2.61 =	82.85±2.26	76.11±2.72 ✓	82.18±2.53 =	180.1±84.09 ✗
CCPP	3.09±0.09 ✓	4.21±0.05 ✗	3.65±0.07 =	3.65±0.07	3.39±0.08 ✓	3.97±0.07 ✗	3.92±0.07 ✗
Elevators ($\times 10^{-3}$)	2.15±0.05 ✓	2.0±0 =	2.38±0.08 =	2.30±0.10	2.08±0.06 ✓	2.0±0 =	2.0±0 =
No2	0.47±0.03 =	0.50±0.02 =	0.49±0.03 =	0.49±0.03	0.47±0.03 =	0.52±0.02 =	0.50±0.03 =
Sensory	0.51±0.03 ✗	0.48±0.01 ✗	0.48±0.01 ✗	0.45±0.02	0.45±0.02 =	0.49±0.01 ✗	0.46±0.03 =
Airfoil	1.54±0.10 ✓	4.80±0.20 ✗	4.54±0.14 ✗	3.05±0.24	2.05±0.10 ✓	2.13±0.11 ✓	4.79±0.21 ✗
Skill Craft	0.94±0.02 ✗	0.93±0.03 =	0.90±0.02 =	0.90±0.02	0.90±0.02 =	0.91±0.02 =	1.38±0.80 =
Ailerons ($\times 10^{-4}$)	1.58±0.05 ✓	2.0±0 =	1.71±0.04 ✗	1.66±0.04	1.63±0.03 =	2.0±0 =	2.0±0 =
W/D/L	**11/1/2**	**0/4/10**	**0/6/8**		**8/5/1**	**4/6/4**	**1/8/5**

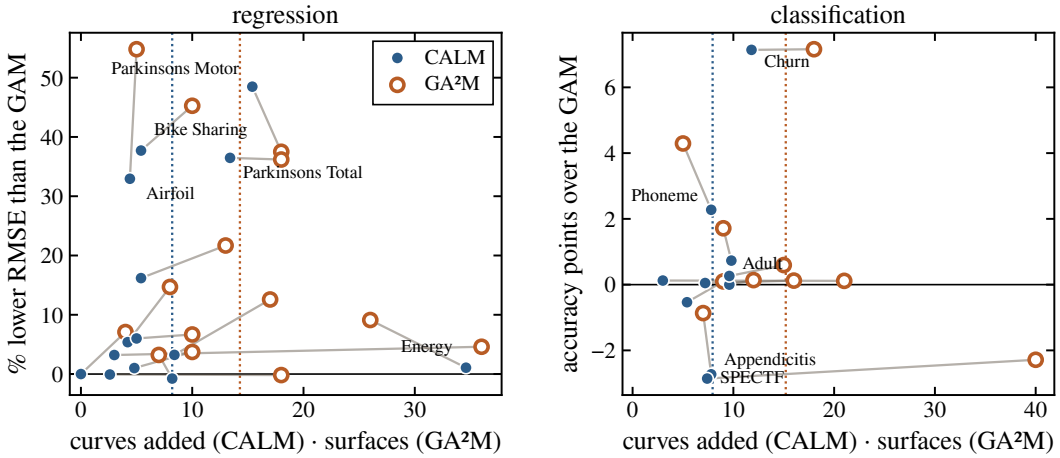


Figure 6: What each model asks a reader to inspect beyond the GAM's one curve per feature, against what it gains over the GAM (24 datasets; means over 5 folds). For CALM the count is its splits, each of which adds one curve; for the GA^{2}M it is its pairwise surfaces. One dot (CALM) and one ring (GA^{2}M) per dataset, joined; dotted lines mark the means. CALM inspects 8.2 added curves on average against 14.3 surfaces in regression and 7.9 against 15.2 in classification, and recovers about 60 % of the GA^{2}M's gain where that gain exceeds one point. Significance is marked in Tables 4 and 5.

Selection threshold. Halving γ changes no result and adds about two conditional interactions; doubling it loses on Bike Sharing and Seoul Bike and removes about 1.5 (Table 8).

Depth. Table 9 gives the share of the GAM-to-XGBoost gap that CALM closes at $d_{\max} = 1, 2, 3$, over the datasets where XGBoost beats the GAM by more than half a fold's standard deviation; Table 10 continues to $d_{\max} = 6$ on four datasets. Figure 7 shows the same feature of the same dataset at three partition depths. Every level closes more of the gap to the black box and doubles the curves a reader has to hold: two curves answer the questions of §3.2 at a glance, eight no longer do. The default $d_{\max} = 2$ is where most of the gain has arrived and the plot is still read as a few named regions.

Table 6: Complexity per dataset, classification: CALM's conditional interactions and added curves, and the pairs of each GA^{2}M.

	CALM				
Dataset	**interactions**	**curves**	**EB^{2}M**	**NODE-GA^{2}M**	**GAMI-Net**
Adult	9.2±1.5	9.6±1.2	12.0±0.0	72.6±3.2	20.0±0.0
COMPAS	4.4±2.1	5.4±2.2	9.0±0.0	36.0±0.0	20.0±0.0
HELOC	3.0±1.1	3.0±1.1	21.0±0.0	163.8±4.4	20.0±0.0
MIMIC2	6.4±0.8	7.2±1.0	16.0±0.0	115.0±3.3	20.0±0.0
Appendicitis	6.6±2.7	7.8±1.9	7.0±0.0	21.0±0.0	6.2±8.1
Phoneme	5.4±0.8	7.8±1.5	5.0±0.0	10.0±0.0	10.0±0.0
SPECTF	7.2±2.1	7.4±2.4	40.0±0.0	273.0±14.4	20.0±0.0
Magic	7.2±2.2	9.8±2.1	9.0±0.0	44.0±0.6	20.0±0.0
Bank	7.6±1.5	9.6±1.6	15.0±0.0	104.2±3.2	20.0±0.0
Churn	10.8±0.4	11.8±0.4	18.0±0.0	137.4±2.9	20.0±0.0
Avg.	6.8	7.9	15.2	97.7	17.6

Table 7: Complexity per dataset, regression: CALM's conditional interactions and added curves, and the pairs of each GA^{2}M.

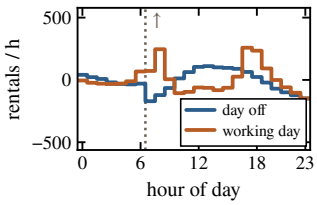
	CALM				
Dataset	**interactions**	**curves**	**EB^{2}M**	**NODE-GA^{2}M**	**GAMI-Net**
Bike Sharing	4.6±0.8	5.4±1.2	10.0±0.0	53.6±1.4	20.0±0.0
California Housing	2.0±0.9	4.2±1.5	8.0±0.0	28.0±0.0	20.0±0.0
Parkinsons Motor	10.6±2.4	15.4±3.0	18.0±0.0	126.2±1.7	20.0±0.0
Parkinsons Total	9.8±2.7	13.4±2.7	18.0±0.0	129.2±3.9	20.0±0.0
Seoul Bike	4.6±0.8	5.4±1.2	13.0±0.0	83.2±0.7	20.0±0.0
Wine	4.2±0.7	4.8±0.7	10.0±0.0	54.6±0.5	20.0±0.0
Energy	31.0±2.5	34.6±1.6	26.0±0.0	192.0±4.5	4.0±8.0
CCPP	0.0±0.0	0.0±0.0	4.0±0.0	6.0±0.0	5.8±0.4
Elevators	5.8±1.0	8.4±1.2	17.0±0.0	115.4±2.1	20.0±0.0
No2	7.4±2.3	8.2±2.2	7.0±0.0	21.0±0.0	20.0±0.0
Sensory	4.4±1.0	5.0±1.3	10.0±0.0	54.0±0.9	16.0±4.9
Airfoil	3.0±0.9	4.4±1.2	5.0±0.0	10.0±0.0	10.0±0.0
Skill Craft	2.0±0.6	2.6±0.5	18.0±0.0	135.6±2.4	16.0±8.0
Ailerons	2.6±0.5	3.0±0.0	36.0±0.0	232.4±5.1	20.0±0.0
Avg.	6.6	8.2	14.3	88.7	16.6

Table 8: Halving and doubling γ against the default $\gamma = 0.01$: wins, draws and losses of the variant, and the mean change in conditional interactions.

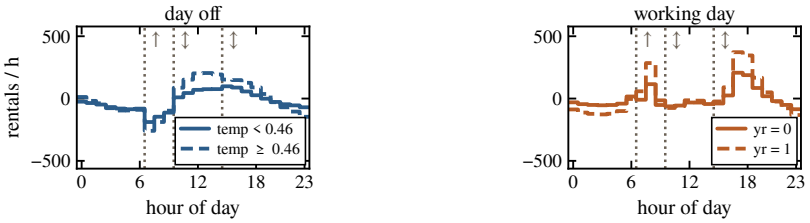
γ	task	W / D / L	Δ interactions
0.005	classification	0 / 10 / 0	+1.7
	regression	0 / 14 / 0	+2.1
0.02	classification	0 / 10 / 0	−1.8
	regression	0 / 12 / 2	−1.5

Table 9: Share of the GAM-to-XGBoost gap closed by depth (median and quartiles over the datasets with a gap), and the median number of conditional interactions over all datasets.

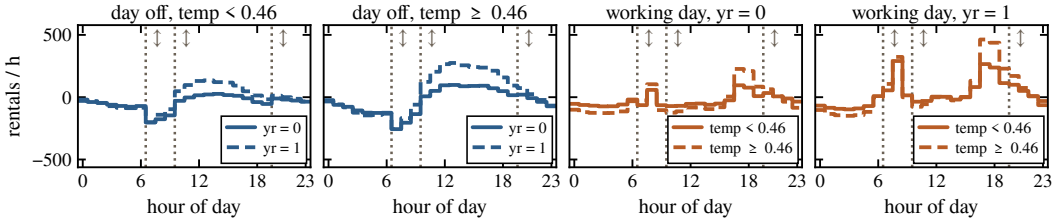
	classification (5 of 10)		regression (12 of 14)	
$d_{\max}$	gap closed	interactions	gap closed	interactions
1	17 % (10–17)	2.5	17 % (2–42)	1.9
2	31 % (27–36)	6.9	37 % (8–55)	4.5
3	41 % (35–44)	10.8	49 % (13–64)	9.6



(a) $d_{\max} = 1$: RMSE 73.7, 43% of the gap closed



(b) $d_{\max} = 2$: RMSE 62.4, 62% of the gap closed



(c) $d_{\max} = 3$: RMSE 56.7, 71% of the gap closed

Figure 7: What depth buys: the hour of the Bike Sharing data under CALM at $d_{\max} = 1$, 2 and 3 (fits on all rows; depth 1 is the model of Figure 2, depth 2 the default of Table 5). Each level splits every region once more, giving 2, 4 and 8 curves: (a) shows both, (b) one panel per day type and (c) one panel per region of the first two splits, each panel above the two it splits into; the hue is the day type, the line style the last split; dotted lines mark where another feature's region switches on the hour, each with the sign of its hidden jump (§3.2): $\uparrow [36, 334]$ at depth 1; $\uparrow [8, 267]$, $\updownarrow [-159, 63]$ and $\updownarrow [-48, 70]$ at depth 2; all three $\updownarrow$ at depth 3. Under each depth, the held-out RMSE of that depth and the share of the GAM $\rightarrow$ XGBoost gap it closes, over 5 folds (GAM 100.2, XGBoost 39.4): 43 %, 62 % and 71 %.

Table 10: Depth ladders: score (mean $\pm$ std. dev. over 5-fold CV; RMSE for Bike Sharing, accuracy otherwise) and conditional interactions (#) at $d_{\max} = 1, \ldots, 6$. Gap closed at $d_{\max} = 1, \ldots, 6$: Bike Sharing 43, 62, 71, 78, 83, 85 %; Bank 17, 36, 44, 55, 53, 64 %; Phoneme 10, 31, 41, 43, 45, 59 %; Adult has no gap to close.

	Bike Sharing		**Adult**		**Bank**		**Phoneme**	
$d_{\max}$	RMSE	#	accuracy	#	accuracy	#	accuracy	#
GAM	100.17±0.99		0.8699±0.0016		0.9027±0.0020		0.8266±0.0071	
1	73.74±0.96	1.8	0.8702±0.0018	4.0	0.9040±0.0028	3.2	0.8338±0.0108	2.4
2	62.40±1.99	4.6	0.8699±0.0015	9.2	0.9054±0.0023	7.6	0.8494±0.0055	5.4
3	56.71±1.31	3.2	0.8699±0.0026	11.8	0.9060±0.0028	12.2	0.8568±0.0091	6.2
4	52.54±1.53	5.0	0.8691±0.0030	12.6	0.9068±0.0020	21.0	0.8583±0.0121	7.8
5	49.68±1.31	8.0	0.8693±0.0020	16.2	0.9067±0.0015	24.8	0.8595±0.0107	8.0
6	48.67±1.60	9.8	0.8679±0.0014	20.2	0.9075±0.0017	31.0	0.8703±0.0050	11.2
XGB	39.35±1.38		0.8700±0.0025		0.9101±0.0037		0.9003±0.0037	

C.6 Runtime

Tables 11 and 12 give the fit time of each model. CALM's time is end to end, from training the reference model to fitting the shape functions of Step 2. The columns taken from the earlier run were measured on other hardware (Section C.1) and compare only with each other.

Table 11: Runtime (seconds), classification. XGB, EBM, CALM: one fit (fold 0) on the laptop; NAM, EB^{2}M, NODE-GA^{2}M, GAMI-Net: from an earlier run, mean $\pm$ std. dev. over 5 folds on the server.

Dataset	**BlackBox**	**GAM**		**CALM**	**GA^{2}M**		
	XGB	**NAM**	**EBM**		**EB^{2}M**	**NODE-GA^{2}M**	**GAMI-Net**
Adult	0.3	38±2	87	92	13±2	97±9	718±122
COMPAS	0.1	10±0.1	0.6	2.7	1±0.04	51±1	129±5
HELOC	0.3	31±6	2.1	25	2±0.2	57±0.1	249±60
MIMIC2	0.3	31±0.3	3.0	24	5±1	69±9	359±32
Appendicitis	0.3	5±0.1	2.5	1.2	1±0.5	24±0.1	6±5
Phoneme	0.5	6±1	6.5	5.7	4±1	61±4	166±38
SPECTF	0.1	32±0.3	1.8	4.2	4±5	25±0.2	180±13
Magic	0.3	28±18	8.6	23	7±4	98±7	489±63
Bank	0.5	44±1	13	49	13±1	98±17	994±265
Churn	0.3	19±3	1.8	11	3±2	56±2	257±32
Avg.	0.3	24	13	24	5.3	64	355

Table 12: Runtime (seconds), regression. Sources as in Table 11.

Dataset	**BlackBox**	**GAM**		**CALM**	**GA^{2}M**		
	XGB	**NAM**	**EBM**		**EB^{2}M**	**NODE-GA^{2}M**	**GAMI-Net**
Bike Sharing	0.3	18±1	6.2	20	19±3	187±31	299±33
California Housing	0.3	15±0.1	16	21	16±0.5	200±46	677±147
Parkinsons Motor	0.5	21±6	45	89	44±4	193±34	536±88
Parkinsons Total	0.5	18±0.2	70	82	40±5	227±76	543±79
Seoul Bike	0.3	15±0.2	4.4	12	12±4	127±18	268±53
Wine	0.3	12±1	1.5	6.5	3±0.4	62±10	187±20
Energy	0.5	43±0.4	54	118	97±15	206±57	253±214
CCPP	0.2	6±0.1	42	42	14±2	133±18	88±8
Elevators	0.3	25±0.4	5.8	35	43±8	181±38	773±269
No2	0.2	5±0.04	0.7	1.9	2±0.1	26±0.1	78±1
Sensory	0.1	10±1	1.0	2.8	2±0.4	27±0.5	74±16
Airfoil	0.4	4±0.03	2.8	3.7	7±1	51±12	76±34
Skill Craft	0.5	17±0.2	2.1	7.9	1±0.1	44±0.1	141±52
Ailerons	0.4	53±1	4.0	69	3±0.1	126±36	619±134
Avg.	0.3	19	18	37	22	128	329

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 analysis. The survey was anonymous, collected no personal data, asked for informed consent and was not paid; under our institution's policy, such a study is exempt from ethics-committee review. The CALM questions came first, with a warm-up on a synthetic model, and the GA^{2}M appeared only on Bike Sharing; presentation order and dataset complexity were not counterbalanced, so the comparison should be read with this caveat.

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

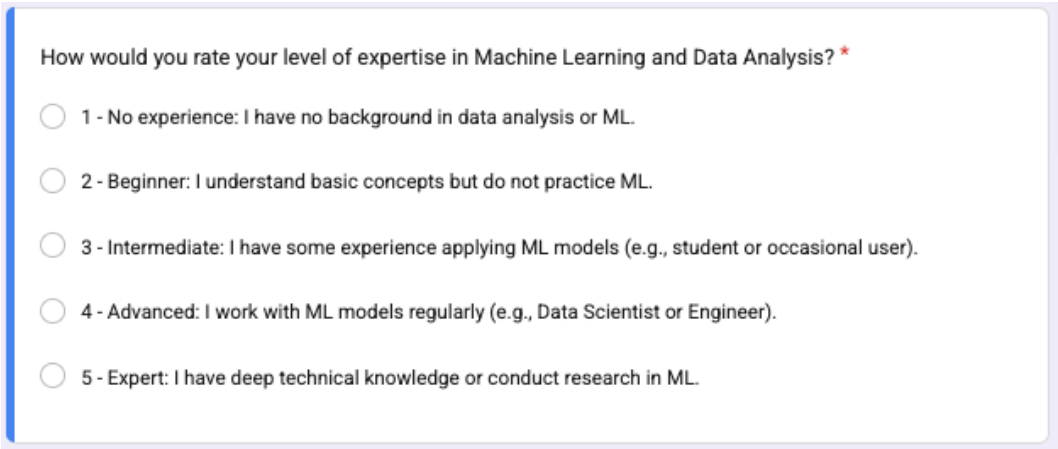


Figure 8: Self-reported expertise question used to assess participant background.

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 [78, 50].

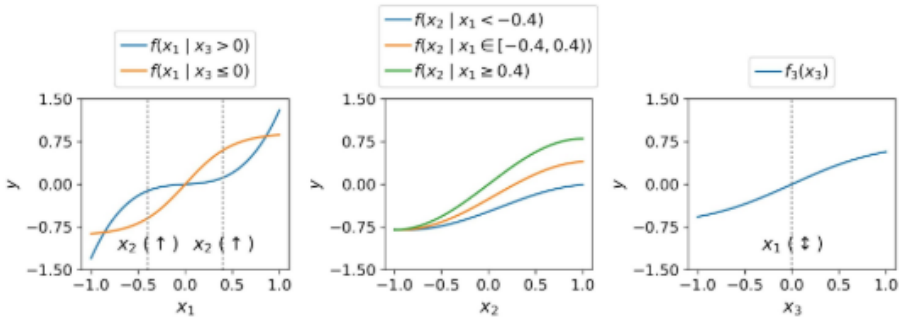
Example questions. Figure 10 shows a representative example of the questions used in the study. The full questionnaire is available at `https://forms.gle/tTippi5eJ6cZHrfy9`.

Evaluation metrics. Every question is multiple choice with one correct option, so grading is automatic: an answer is correct if and only if it selects that option. 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. On the five warm-up questions on the synthetic model, participants answered 83.2% correctly with CALM. On real-world tasks, CALM significantly outperformed heatmap-based representations in overall task performance (54% vs. 34%, $p < 0.05$), driven by the question of how long the hour takes to lower the prediction by 50 rentals (40% vs. 12%, $p < 0.05$; 68% vs. 56% on the other). The advantage holds among the 13 participants who rated themselves advanced or expert (65.4% vs. 30.8%, Wilcoxon $p = 0.031$), and accuracy with the GA^{2}M does not rise with self-reported expertise (Spearman $\rho = -0.29$, $p = 0.14$).

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.

How to Read the Conditional Plot



Plot for x_1 - Interpretation

Select the Curve: Use the **Blue** curve if $x_3 > 0$ and the **Orange** curve if $x_3 \leq 0$.

Read Contribution: Read the y-value directly from the selected curve.

Calculate Changes:

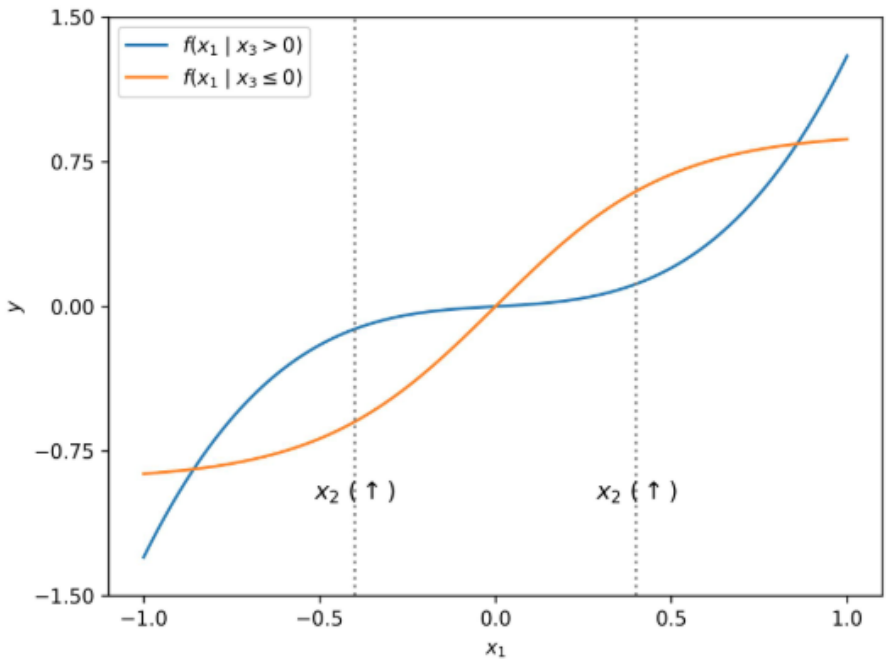
- **Standard:** If the change does **not** cross a dotted line, read the difference directly from the curve.
- **Discontinuity:** If the change **crosses a vertical dotted line**, the actual output jumps. Check the arrow:
  - ↑ **(Up):** Actual change is **larger** than the curve shows.
  - ↓ **(Down):** Actual change is **smaller** than the curve shows.
  - ↕ **(Both):** Actual change is **uncertain**.

Plot for x_2 and x_3 - Interpretation

- **Feature x_2:** Has **3 curves** (depends on x_1). No vertical dotted lines = **No hidden jumps**.
- **Feature x_3:** Has **1 curve**. The dotted line at $x_3 = 0$ with a double arrow (↕) = **Uncertain jump**.

Figure 9: Instruction panel provided to participants for interpreting conditional feature plots.

Q1. Given $x_1 = 0.5$, $x_3 = 0.2$, what is the approximate contribution of feature x_1 to the prediction? *



- -0.7
- +0.7
- +0.2

Figure 10: Example question requiring estimation of feature contribution from a conditional plot.

E Broader Impact

CALM aims at models that are transparent and easy to audit: each feature keeps one plot, and each curve states the region where it applies, which supports decision making in high-stakes domains such as credit scoring and healthcare. The same plots should be used with care: they show what *the model* does, not how *the data* were generated, and conclusions about the process that generated the data need further assumptions [36].

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 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; the code that reproduces every experiment is public at `https://github.com/givasile/calm-paper`.

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 code and the instructions to reproduce every experiment are public at `https://github.com/givasile/calm-paper`; the method is released as the `calm-additive` package on PyPI.

   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: Appendix E discusses both: models that are transparent and easy to audit, and the risk of drawing conclusions about the process that generated the data from their plots without further assumptions [36].

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 method is released as the documented `calm-additive` package (`https://pypi.org/project/calm-additive`), and the experiment code at `https://github.com/givasile/calm-paper`.

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: [Yes]

    Justification: Appendix D: the survey was anonymous, collected no personal data, asked for informed consent and was not paid; under our institution's policy, such a study is exempt from ethics-committee review.

    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.