Regionally Additive Models: Explainable-by-design models minimizing feature interactions

Vasilis Gkolemis

June 15, 2023

Abstract

Generalized Additive Models (GAMs) are a popular class of explainable-by-design models that are widely used in practice. GAMs are based on the assumption that the effect of each feature on the target is independent of the values of the other features, however, in cases where this assumption is violated they may lead to poor performance. To address this limitation we propose Regionally Additive Models (RAMs), a novel class of explainable-by-design models, that fits multiple GAMs to subregions of the feature space where interactions are minimized. Our approach consists of two steps: first, we fit a black-box model and we identify the subregions where the black-box model is nearly locally additive, i.e., where the effect of each feature on the target is independent of the values of the other features. Secondly, we train a GAM specifically for each identified subregion.

We show that RAMs are more expressive than GAMs while they are still interpretable.

1 Introduction

Generalized Additive Models (GAMs) [?] are a popular class of explainable by design (x-by-design) models. Their popularity stems from their seamless interpretability; since they are a linear (additive) combination of univariate functions, $f(\mathbf{x}) = c + \sum_{s=1}^{D} f_s(x_s)$, each individual univariate function (component) can be readily visualized and comprehended in isolation. However, GAM's main limitation is that they cannot express interactions between features. To mitigation this limitation, some approaches [?] extend them enabling pairwise interactions, i.e.,

$f(\mathbf{x}) = c + \sum_{s=1}^{D} f_s(x_s) + \sum_{s=1}^{D} \sum_{c \neq s} f_{sc}(x_s, x_c)$. Pairwise interactions can also be visualized and understood in isolation, so these models also maintain their x-by-design nature. Unfortunately, this does not hold for any interaction that involves more than two features, thus, the expressiveness of GAMs is limited to capturing up to two-feature interactions.

To overcome this limitation, we propose Regionally Additive Models (RAMs), a novel class of x-by-design models, that fits multiple GAMs to subregions of the feature space where interactions are minimized. Our approach consists of a three-step pipeline. First, we fit a black-box model to capture all high-order interactions. Second, we identify the subregions where the black-box model is nearly locally additive. Finally, we train a GAM specifically for each identified subregion.

2 Background and motivation

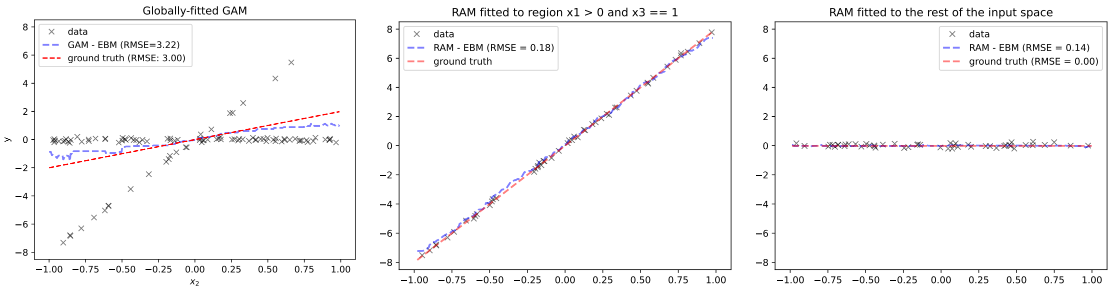
Consider the black-box function $f(\mathbf{x}) = 8x_2 \mathbb{1}_{x_1>0} \mathbb{1}_{x_3=0}$ with $x_1, x_2 \sim \mathcal{U}(-1,1)$ and $x_3 \sim \mathit{Bernoulli}(0,1)$. Although very simple, a GAM would fail to learn this mapping due to the existence of the three-features interaction term $8x_2 \mathbb{1}_{x_1>0} \mathbb{1}_{x_3=0}$. As we see in Figure 1a, a GAM misleadingly learns that $\hat{f}(\mathbf{x}) \approx 2x_2$, because in $\frac{1}{4}$ of the cases ($x_1 > 0$ and $x_3 = 0$) the impact of x_2 to the output is $8x_2$, and in the rest $\frac{3}{4}$ of the cases the impact of x_2 to the output is 0. However, if splitting the input space in two subregions we observe that f is additive in each one (regionally additive):

$$f(\mathbf{x}) = \begin{cases} 8x_2 & \text{if } x_1 > 0 \text{ and } x_3 = 1 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Therefore, if we knew the appropriate subregions, namely, $\mathcal{R}_{21} = \{x_1 > 0 \text{ and } x_3 = 0\}$ and $\mathcal{R}_{22} = \{x_1 \leq 0 \text{ or } x_3 = 1\}$, we could split the impact of x_2 appropriately and fit the following model to the data:

$$f^{\texttt{RAM}}(\mathbf{x}) = f_1(x_1) + f_{21}(x_2) \mathbb{1}_{(x_1,x_3) \in \mathcal{R}_{21}} + f_{22}(x_2) \mathbb{1}_{(x_1,x_3) \in \mathcal{R}_{22}} + f_3(x_3) \tag{2}$$

Equation (2) represents a Regionally Additive Model (RAM), which is simply a GAM fitted on each subregion of the feature space. Importantly, RAM's enhanced expressiveness does not come at the expense of interpretability. As we observe in Figures 1b and 1c, we can still visualize and comprehend each univariate function in isolation, exactly as we would do with a GAM, with the only difference being that we have to consider the subregions where each univariate function is active, The key challenge of RAMs is to appropriately identify the subregions where the black-box function is (close to) regionally additive. For this purpose, as we will see in Section 3.3, we propose a novel algorithm that is based on the idea of regional effect plots.



(a) $f_2(x_2)$ (b) $f_2(x_2)\mathbb{1}_{x_1>0 \text{ and } x_3=1}$ (c) $f_2(x_2)\mathbb{1}_{x_1\leq 0 \text{ or } x_3\neq 1}$

Figure 1: Caption for the entire figure

3 The RAM framework

3.1 Background

Let $\mathcal{X} \in \mathbb{R}^d$ be the d-dimensional feature space, $\mathcal{Y}$ the target space and $f(\cdot) : \mathcal{X} \rightarrow \mathcal{Y}$ the black-box function. We use index $s \in \{1, \ldots, d\}$ for the feature of interest and $c = \{1, \ldots, d\} - s$ for the rest. For convenience, we use $(x_s, \mathbf{x_{/s}})$ to refer to $(x_1, \cdots, x_s, \cdots, x_D)$ and, equivalently, (X_s, X_c) instead of $(X_1, \cdots, X_s, \cdots, X_D)$ when we refer to random variables. The training set $\mathcal{D} = \{(\mathbf{x}^i, y^i)\}_{i=1}^N$ is sampled i.i.d. from the distribution $\mathbb{P}_{X,Y}$. Finally, $f^{\texttt{<method>}}(x_s)$ denotes how $< \texttt{method} >$ defines the feature effect and $\hat{f}^{\texttt{<method>}}(x_s)$ how it estimates it from the training set.

3.2 Fit a black-box function

3.2.1 Objective

In the first step of the pipeline, a black-box function $f(\cdot)$ is fitted to the training set $\mathcal{D} = \{(\mathbf{x}^i, y^i)\}_{i=1}^N$. In principle, the black-box function can be any model that is expressive enough to accurately fit the data, meaning that it is able to learn the underlying mapping $\mathbb{P}_{Y|X}$.

3.2.2 Proposed Approach

For being able to use the DALE approximation that we propose in the next step, the black-box function should be differentiable. Recent developments have shown that differentiable Deep Learning models specifically designed for tabular data [**?**] can achieve state-of-the-art performance, make them a good candidate for this step. In the running example, we use a simple neural network with three hidden layers as the black-box function, which achieves a test `MSE` ≈ 0.01. The neural network is trained using the Adam optimizer (add citation) with a learning rate of 0.01. Based on the small test `MSE`, we can assume that the neural network is able to capture any interactions between the features.

3.3 Regions that minimize feature interactions

In this step, we use regional effect methods to identify the regions where the black-box function is nearly locally additive.

3.3.1 Objective

Regional effect methods yield for each individual feature s, a set of T_s non-overlapping regions, denoted as $\{\mathcal{R}_{st}\}_{t=1}^{T_s}$ where $\mathcal{R}_{st} \subseteq \mathcal{X}_{/s}$. Note that, first, we use a subscript, i.e. T_s, to denote that the number of non-overlapping regions can be different for each feature and, second, the regions $\{\mathcal{R}_{st}\}_{t=1}^{T_s}$ are disjoint and their union covers the entire feature space $\mathcal{X}_{/s}$. The primary objective is to identify regions in which the impact of the s-th feature on the output is *relatively independent* of the values of the other features $\mathbf{x}_{/\mathbf{s}}$. To better grasp this objective, if we decompose the impact of the s-th feature on the output y into two terms: $f_s(x_s, \mathbf{x}_{/\mathbf{s}}) = f_{s,ind}(x_s) + f_{s,int}(x_s, \mathbf{x}_{/\mathbf{s}})$, where $f_{s,ind}(\cdot)$ represents the independent effect and $f_{s,int}(\cdot)$ represents the interaction effect, the objective is to identify regions $\{\mathcal{R}_{st}\}_{t=1}^{T_s}$ such that the interaction effect is minimized. Regionally Additive Models (RAM) formulation is:

$$f^{\texttt{RAM}}(\mathbf{x}) = c + \sum_{s=1}^{D} \sum_{t=1}^{T_s} f_{st}(x_s) \mathbb{1}_{\mathbf{x}_{/\mathbf{s}} \in \mathcal{R}_{st}} \tag{3}$$

In the above formulation, $f_{st}(\cdot)$ is the component of the s-th feature which is active on the t-th region. RAM can be viewed as a GAM with T_s components per feature where each component is applied to a specific region $\mathcal{R}_{st}$. To facilitate this interpretation, we can define an enhanced feature space $\mathcal{X}^{\texttt{RAM}}$ defined as:

$$\mathcal{X}^{\texttt{RAM}} = \{x_{st} | s \in \{1, \ldots, D\}, t \in \{1, \ldots, T_s\}\}$$
$$x_{sk} = \begin{cases} x_s, & \text{if } \mathbf{x_{/s}} \in \mathcal{R}_{sk} \\ 0, & \text{otherwise} \end{cases} \tag{4}$$

and then define RAM as a typical GAM on the extended feature space $\mathcal{X}^{\texttt{RAM}}$:

$$f^{\texttt{RAM}}(\mathbf{x}) = c + \sum_{s,t} f_{st}(x_{st}) \tag{5}$$

In the running example, to minimize the effect of feature interactions, we have to split the effect of feature x_2 into two regions, $\mathcal{R}_{21} = \{x_1 > 0 \text{ and } x_3 = 1\}$ and $\mathcal{R}_{22} = \{x_1 \leq 0 \text{ or } x_3 = 0\}$. The RAM formulation is $f^{\texttt{RAM}}(\mathbf{x}) = f_1(x_1) + f_{21}(x_{21}) + f_{22}(x_{22}) + f_3(x_3)$.

3.3.2 Proposed Approach

To identify the regions of the input space where the impact of feature interactions is reduced, we have developed a regional effect method influenced by the research conducted by **?** and **?**. **?** introduced a versatile framework for detecting such regions, where one of the proposed methods is the Accumulated Local Effects [**?**]. We have adopted their approach with two notable modifications. First, instead of using the ALE plot, we employ the Differential ALE (DALE) method introduced by **?**, which provides considerable computational advantages when the underlying black-box function is differentiable. Second, we utilize variable-size bins, instead of the fixed-size ones in DALE, because the result in a more accurate approximation.

DALE and Feature Interaction DALE plot gets as input a black-box function $f(\cdot)$ and a dataset $\mathcal{D} = \{(\mathbf{x}^i, y^i)\}_{i=1}^N$, and decomposes the `effect` (impact) of a specific feature s on the output y. DALE's formulation is:

$$\hat{f}^{\texttt{DALE}}(x_s) = \Delta x \sum_{k=1}^{k_x} \underbrace{\frac{1}{|\mathcal{S}_k|} \sum_{i:\mathbf{x}^{(i)} \in \mathcal{S}_k} \frac{\partial f}{\partial x_s}(\mathbf{x}^i))}_{\hat{\mu}(z_{k-1}, z_k)} \tag{6}$$

where k_x the index of the bin such that $z_{k_x-1} \leq x_s < z_{k_x}$ and $\mathcal{S}_k$ is the set of the instances of the k-th bin, i.e. $\mathcal{S}_k = \{\mathbf{x}^i : z_{k-1} \leq x_s^{(i)} < z_k\}$. DALE computes the average effect (impact) of the feature x_s on the output of the black-box function $f(\cdot)$, first by dividing the feature space into bins and computing the average effect of the feature x_s in each bin (bin-effect) and then by aggregating the bin-level effects.

In cases where there are strong interactions between the features, the instance-level effects inside each bin start to deviate from the average bin-effect. We can measure such deviation using the standard deviation of the instance-level effects inside each bin (bin-deviation):

$$\hat{\sigma}^2(z_{k-1}, z_k) = \frac{1}{|\mathcal{S}_k| - 1} \sum_{i:\mathbf{x}^i \in \mathcal{S}_k} \left(\frac{\partial f}{\partial x_s}(\mathbf{x}^i) - \hat{\mu}(z_{k-1}, z_k) \right)^2 \tag{7}$$

and the global interaction between the feature x_s and the rest of the features with the aggregated bin-deviation

$$\mathcal{H}_s = \sqrt{\sum_{k=1}^{k_x} (z_k - z_{k-1})^2 \hat{\sigma}^2(z_{k-1}, z_k)} \tag{8}$$

Eq. (8) measures the interaction between the feature x_s and the rest of the features. It takes values in the range $[0, \infty)$ and it is zero when x_s does not interact with the rest of the features, i.e., when $f(\mathbf{x}) = f_s(x_s) + f_s(x_s)$ and it is higher when the interaction is stronger.

DALE for Regional Effect If instead of applying DALE globally, we restrict it on a subregion of the feature space $\mathcal{R}_{st} \subset \mathcal{X}$, we can measure the interaction between the feature x_s and the rest of the features inside the subregion $\mathcal{R}_{st}$. In compuatational terms, this means that instead of using the whole dataset $\mathcal{D}$ to compute the bin-effect and the bin-deviation, we use only the instances that belong to the subregion $\mathcal{R}_{st}$. In this way, we can define $\hat{\mu}_{\mathcal{R}_{st}}(z_{k-1}, z_k), \hat{\sigma}^2_{\mathcal{R}_{st}}(z_{k-1}, z_k)$ and $\mathcal{H}_{\mathcal{R}_{st}}$ exaclty as in Eq. (6), Eq. (7) and Eq. (8) respectively, but using only the instances that belong to the subregion $\mathcal{R}_{st}$, i.e. $\mathbf{x}^i : x_s^i \in \mathcal{S}_k \wedge x_s^i \in \mathcal{R}_{st}$. Analytical formulas in the Appendix 5.1.

Therefore, in order to minimize the interaction we search for a set of regions $\{\mathcal{R}_{st}\}_{t=1}^{T_s}$, that minimize:

$$\begin{aligned} \underset{\{\mathcal{R}_{st}\}_{t=1}^{T_s}}{\text{minimize}} \quad & \mathcal{L} = \sum_{s=1}^{S} \sum_{t=1}^{T} \mathcal{H}_{\mathcal{R}_{st}} \\ \text{subject to} \quad & \bigcup_{t=1}^{T} \mathcal{R}_{st} = \mathcal{X} \\ & \mathcal{R}_{st} \cap \mathcal{R}_{s\tau} = \emptyset, \quad \forall t \neq \tau \end{aligned} \tag{9}$$

For minimizing Eq. (12), we set up a CART-like algorithm, as proposed by [**?**]. The backbone of the algorithm is described in Algorithm 1. The basic idea is for each feature $s \in \{1, \dots, D\}$ to iterate over all the other features $c \in \{1, \dots, D\} \setminus s$, and for each feature c to find the optimal split point z_{k^*} that minimizes the interaction $\mathcal{H}_{\mathcal{R}_{st}}$.

Algorithm 1: DALE-based Region Detection

```
Input  : X, J, T
Output: optimal_splits
splits[s, c, t] ← None;
positions[s, c, t] ← None;
for $s = 1$ to $D$ do
    for $c \in \{1, \dots, D\} \setminus \{s\}$ do
        is_cat = True/False ;
        X_list ← [$X$];
        J_list ← [$J$];
        for $t = 1$ to $T$ do
            L, position, X_split, J_split ← BestSplit(X_list, J_list, s, is_cat);
            splits[s, c, t] ← L;
            positions[s, c, t] ← position;
        end
    end
end
```

3.4 Fitting the GAMs

3.4.1 Objective

3.4.2 Proposed Approach

3.5 Discussion

Recently, a number of methods have been proposed to extend traditional GAMs and make them more expressive. The majority of the ideas follow one of the following research directions; The first one targets on representing the main components of a GAM $\{f_i(x_i)\}$ with novel models. For example, **?** who used an end-to-end neural network for learning the main components. The second one focuses on extending the GAMs to model interactions between the features. For example, **?** proposed Explainable Boosting Machines (EBMs) which are a generalized additive model with

Algorithm 2: BestSplit

```
Input  : X_list, J_list, s, c, is_cat
Output: BestSplits
positions ← GetPositions(X_list, c, is_cat);
L ← [];
for p in positions do
    X_split, J_split ← SplitDataset(X_list, J_list, c, p, is_cat);
    L ← GetInteraction(X_split, J_split);
end
split_position = argmin(L);
position ← positions[split_position];
X_split, J_split ← SplitDataset(X_list, J_list, c, position, is_cat);
return L, position, X_split, J_split
```

Algorithm 3: SplitDataset

```
Input  : X_list, J_list, c, val, is_cat
Output: X_split, J_split
X_split ← [];
J_split ← [];
for i = 1 to len(X_list) do
    X= X_list[i];
    J= J_list[i];
    if is_cat then
        ind_1 ← X[:, c] = val;
        ind_2 ← X[:, c] $\neq$ val;
    else
        ind_1 ← X[:, c] $\leq$ val;
        ind_2 ← X[:, c] > val;
    end
    Append X[ind_1], X[ind_2] to X_split;
    Append J[ind_1], J[ind_2] to J_split;
end
return X_split, J_split
```

Algorithm 4: GetInteraction

```
Input   : $X\_list$, $J\_list$, $min\_points$
Output: weighted average of interaction levels
N ← total number of items in $X\_list$;
W ← [];
L ← [];
for $i$ in $len(X\_list)$ do
    X ← X_list[i];
    J ← J_list[i];
    if $|X|$ ¡ $min\_points$ then
        Append $\infty$ to L;
    else
        Append $\mathcal{L}(X, J)$ to $L$;
    end
    Append $\frac{|X|}{N}$ to $W$;
end
return sum($L \cdot W$);
```

pairwise interaction terms. It is worth noting that the proposed method is orthogonal to the aforementioned research directions. As we will show in the experiments, the proposed method can be used in conjunction with any of the aforementioned methods to improve the accuracy of the resulting model, while maintaining the interpretability of the model.

4 Experiments

4.1 Synthetic Examples

4.2 Real-World Datasets

5 Appendix

5.1 Regional DALE formulation

For this reason we define the bin-effect, the bin-deviation and the aggregated deviation on a subregion $\mathcal{R}_{st}$ as:

$$\hat{\mu}_{\mathcal{R}_{st}}(z_{k-1}, z_k) = \frac{1}{|\mathcal{S}_k \cap \mathcal{R}_{st}|} \sum_{i:\mathbf{x}^i \in \mathcal{S}_k \cap \mathcal{R}_{st}} \frac{\partial f}{\partial x_s}(\mathbf{x}^i) \tag{10}$$

$$\hat{\sigma}^2_{\mathcal{R}_{st}}(z_{k-1}, z_k) = \frac{1}{|\mathcal{S}_k \cap \mathcal{R}_{st}| - 1} \sum_{i:\mathbf{x}^i \in \mathcal{S}_k \cap \mathcal{R}_{st}} \left(\frac{\partial f}{\partial x_s}(\mathbf{x}^i) - \hat{\mu}(z_{k-1}, z_k) \right)^2 \tag{11}$$

$$\mathcal{L}_{\mathcal{R}_{sk}} = \sum_{k=1}^{K} (z_k - z_{k-1})^2 \hat{\sigma}^2_{\mathcal{R}_{st}}(z_{k-1}, z_k) \tag{12}$$

5.1.1 Algorithmic Details of Subregion Detection