Statistical Inference for Explainable Boosting Machines

Haimo Fang[1] **Kevin Tan**[2] **Jonathan Pipping-Gamón**[2] **Giles Hooker**[2]
[1]School of Economics, Fudan University
[2]Department of Statistics and Data Science, The Wharton School, University of Pennsylvania

Abstract

Explainable boosting machines (EBMs) are popular "glass-box" models that learn a set of univariate functions using boosting trees. These achieve explainability through visualizations of each feature's effect. However, unlike linear model coefficients, uncertainty quantification for the learned univariate functions requires computationally intensive bootstrapping, making it hard to know which features truly matter. We provide an alternative using recent advances in statistical inference for gradient boosting, deriving methods for statistical inference as well as end-to-end theoretical guarantees. Using a moving average instead of a sum of trees (Boulevard regularization) allows the boosting process to converge to a feature-wise kernel ridge regression. This produces asymptotically normal predictions that achieve the minimax-optimal MSE for fitting Lipschitz GAMs with p features of $O(p\, n^{-2/3})$, successfully avoiding the curse of dimensionality. We then construct prediction intervals for the response and confidence intervals for each learned univariate function with a runtime independent of the number of datapoints, enabling further explainability within EBMs. Code is available at github.com/hetankevin/ebm-inference.

1 INTRODUCTION

Modern machine learning methods have proven to be highly successful, achieving impressively high performance even for complicated prediction tasks. Still, this comes at a cost. These methods have been critiqued for their lack of transparency, often described as "black-box" models. As such, much work has been done in an attempt to provide explanations for their behavior.[1] Another line of work attempts to convince practitioners to instead use inherently transparent models that are algebraically simple enough to be inspected by humans Rudin (2019). Neither is optimal. Explaining black-box models is a difficult problem Hooker et al. (2021); Fryer et al. (2021), and inherently interpretable models can sacrifice performance.

In light of this, Explainable Boosting Machines (EBM) Lou et al. (2012); Nori et al. (2019) have emerged as a popular "glass-box" model to strike a balance between both. An EBM learns a Generalized Additive Model (GAM) Hastie and Tibshirani (1986): $\mathbf{f}(\mathbf{x}) = \sum_{k=1}^{p} \mathbf{f}^{(k)}(\mathbf{x}^{(k)})$, where each $\mathbf{f}^{(k)}$ is assumed to be a univariate function of the k-th feature. However, instead of fitting splines, it instead fits a set of univariate gradient boosting regression trees. The tree-based boosting procedure attempts to recover the expressiveness of state-of-the-art algorithms Chen and Guestrin (2016); Ke et al. (2017), while the decomposition into additive components allows the measurement of the marginal contribution of each feature like linear models. While EBMs impose strong structural constraints on the resulting prediction, their ability to reproduce nonlinearities, along with appropriate choices of additive structure, provides comparable performance to SOTA boosting Nori et al. (2019).

While EBMs provide a clear graphical representation for the effects of each feature, these are still point estimates. The central question we ask for uncertainty quantification can be phrased as this: *Given a model design, if we collect a new dataset and retrain the model, how different would our new prediction be?* In response to this question, vanilla EBMs Lou et al. (2012) compute variance estimates via bootstrapping, though this is computationally intensive

Proceedings of the 29th International Conference on Artificial Intelligence and Statistics (AISTATS) 2026, Tangier, Morocco. PMLR: Volume 300.

[1]Popular methods include LIME Ribeiro et al. (2016), SHAP Lundberg and Lee (2017), ALE Apley and Zhu (2020) and partial dependence Friedman (2001) plots.

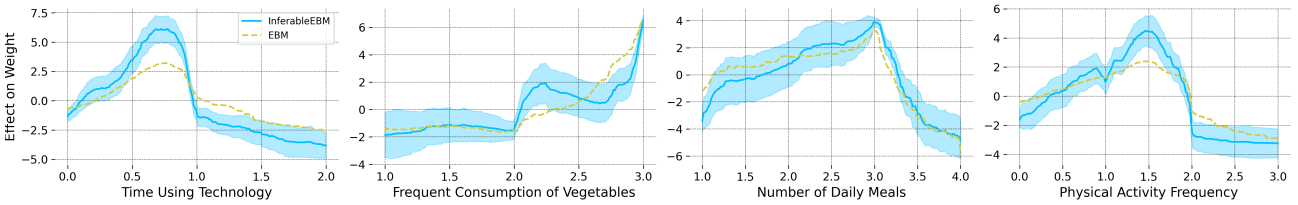

Figure 1: Example of the feature-specific confidence intervals generated by Algorithm 1, compared to the point estimates from EBM Nori et al. (2019). This visualizes centered feature effects on the UCI Machine Learning Reposity Obesity Palechor and la Hoz Manotas (2019) dataset, when predicting weight as the response.

and only heuristically justified. Our primary objective is hence *well-calibrated frequentist uncertainty quantification* for EBMs while retaining their interpretability and scalability.

A canonical alternative to bootstrapping is to construct confidence and prediction intervals using the asymptotic properties of the model. This is the case with GAMs fitted via the backfitting (but not boosting) algorithm, which enjoy a rich asymptotic theory that enables the construction of confidence and prediction intervals for them Wahba (1983). Another widely used alternative for prediction intervals is conformal prediction, which provides finite-sample marginal coverage with minimal assumptions. However, conformal methods do not directly yield feature-wise confidence intervals for the GAM components, which we aim to explore in this work.

Unfortunately, the asymptotic behavior of most gradient boosting algorithms (of which EBMs are one), is not known. Ustimenko et al. (2023) show that a randomized version of boosting converges to a Gaussian process. On the other hand, Zhou and Hooker (2019) show that taking an average instead of a sum of trees (in a process which they call Boulevard regularization) converges to a kernel ridge regression, yielding asymptotic normality. This was recently leveraged by Fang et al. (2025) to construct confidence intervals for the underlying nonparametric function $\mathbf{f}$ and prediction intervals for the labels, with extensions to random dropout and parallel boosting.

Our contributions. Unfortunately, the underlying nonparametric function itself can be highly complicated and uninterpretable in general. Restricting this to the class of GAMs, as with EBMs, yields interpretability. However, the existing Boulevard theory Zhou and Hooker (2019) does not directly apply to the class of algorithms we consider here. These results need to be extended to allow different structure matrices (resulting in different kernels) for each feature, along with enforcing the GAM identifiability conditions. While we sketch an algorithm that can be analyzed largely within the framework of Fang et al. (2025), further algorithmic variants that either employ backfitting directly or employ parallelization methods result in different algorithmic limits requiring separate analysis. Unlike prior Boulevard analyses, EBMs induce *feature-specific* kernels. This makes the theory substantially more delicate: one must control their interaction under centering and GAM identifiability, and establish asymptotic normality for each additive component, not just the overall predictor. To handle these additional challenges, we make the following contributions:

- **Asymptotic Theory.** We establish central limit theorems for three variations of EBMs (parallelized, backfitting-like parallelized, and sequential), showing that the Boulevard-regularized ensemble converges to a feature-wise kernel ridge regression. The resulting predictions are asymptotically normal and achieve the minimax-optimal MSE of $O(pn^{-2/3})$ for fitting Lipschitz p-dimensional GAMs Yuan and Zhou (2015). This avoids the curse of dimensionality, significantly improving upon the rates of Zhou and Hooker (2019); Fang et al. (2025).

- **Frequentist Inference.** By leveraging our proposed CLT for baseline vector $\boldsymbol{\beta}$ and feature-wise predictions $\widehat{\mathbf{f}}^{(k)}$, we extend the heuristic bootstrap to a set of inference toolkits. We provide confidence intervals for each underlying function $\mathbf{f}^{(k)}$ (see Figure 1 for an example), and prediction intervals for the response. This also allows a direct test of variable importance following the methods of Mentch and Hooker (2015). We round out those propositions by numerical experiments to back up the validity of our algorithms.

- **Computational Efficiency.** The derived procedures above rely on a kernel ridge regression, with a prohibitively expensive $O(n^3)$ runtime. Fang et al. (2025) employ the Nyström approximation to produce intervals in $O(np^2)$ kernel precomputation and $O(p^2)$ query time, but the recursive implementation of Musco and Musco (2017) can be slow in practice. Given that EBMs utilize histogram trees, we instead work out the computations in histogram bin-space. This bin-based

implementation runs in an additional lower-order $O(pm^2)$ time per training round, $O(pm^3)$ kernel precomputation time, and $O(pm^2)$ interval query time, where p is the number of features and m the maximum number of bins per-feature. As the number of bins is usually capped at 255 or 511, this is independent of the number of samples n!

2 SETUP AND NOTATION

Let $(\mathbf{X}_n, \mathbf{y}_n) = (\mathbf{x}_i, y_i)_{i=1}^n$ are i.i.d. samples with $\mathbf{x}_i = (\mathbf{x}_i^{(1)}, \ldots, \mathbf{x}_i^{(p)})^\top$. Let $\mathbf{J}_n = \mathbf{I}_n - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$ denote the centering projector. We adopt the standard GAM identifiability convention that each component is centered on the training set: $\sum_{i=1}^n \mathbf{f}^{(k)}(\mathbf{x}_i^{(k)}) = 0$ for all k, so the intercept equals the sample mean. Then:

Assumption 2.1 (Additive Lipschitz Nonparametric Regression)**.** Assume that the response satisfies

$$y = \boldsymbol{\beta} + \sum_{k=1}^{p} \mathbf{f}^{(k)}(\mathbf{x}^{(k)}) + \epsilon := \mathbf{f}(\mathbf{x}) + \epsilon,$$

where $y, \boldsymbol{\beta} \in \mathbb{R}$, $\mathbf{x} = (\mathbf{x}^{(1)}, \ldots, \mathbf{x}^{(p)})^\top \in \mathbb{R}^p$, and ε are i.i.d. sub–Gaussian errors with mean zero and variance $\sigma^2 < \infty$. Assume the covariates have a bounded density on their support: $0 < c_1 \le \mu(\mathbf{x}) \le c_2 < \infty$. Each component function $\mathbf{f}^{(k)} : \mathbb{R} \to \mathbb{R}$ is mean–zero with respect to the marginal of $\mathbf{x}^{(k)}$, and is L-Lipschitz. The intercept $\boldsymbol{\beta}$ is defined as the unconditional mean of the response: $\boldsymbol{\beta} := \mathbb{E}_{\mathbf{x},\epsilon}[y] = \int y \, d\mu_{\mathbf{x}}(x) \, d\mu_\epsilon(\epsilon)$.

Boulevard Regularization. Zhou and Hooker (2019) replace the usual additive update with a moving average. Instead of adding the next tree: $\widehat{\mathbf{f}}_b \leftarrow \mathbf{f}_{b-1} + \lambda \mathbf{t}_b$, they average over all previous trees: $\widehat{\mathbf{f}}_b \leftarrow \frac{b-1}{b}\mathbf{f}_{b-1} + \frac{\lambda}{b}\mathbf{t}_b$. This idea of shrinking the previous ensemble as also been adopted by Ustimenko and Prokhorenkova (2022); Ustimenko et al. (2023). This ensures convergence to a kernel ridge regression in the limit of infinite boosting rounds, and that the resulting predictions are asymptotically normal.[2]

3 ALGORITHMS

We adapt the above regularized boosting procedure to fitting EBMs below in Algorithm 1, in the hope that doing so allows us to show similar theoretical guarantees for the resulting procedure.

[2]A particular advantage of Boulevard regularization is that it is robust to overfitting Fang et al. (2025). Figure 4 shows this behavior for Algorithm 1. EBM achieves this here via tiny learning rates when cycling through features in sequence. We do this while fitting to features in parallel.

Algorithm 1 Inferable EBM

Input: $\mathbf{X} \in \mathbb{R}^{n\times p}$, $\mathbf{y} \in \mathbb{R}^n$, subsample rate ξ, learning rate λ, rounds B, truncation level $M > 0$.
Init: $\widehat{\boldsymbol{\beta}}_0 \leftarrow \frac{1}{n}\sum_{i=1}^n \mathbf{y}_i$; $\mathbf{f}_k^{(0)} \leftarrow 0$ for all k.
for $b = 1$ **to** B **do**
  for $k = 1$ **to** p **in parallel do**
    Sample $G_{b,k} \subset \{1, \ldots, n\}$ i.i.d. w.p. ξ.
    $r_{i,k} \leftarrow \mathbf{y}_i - (\widehat{\boldsymbol{\beta}}_{b-1} + \sum_\ell \mathbf{f}_{b-1}^{(\ell)}(\mathbf{x}_i^{(\ell)}))$.
    Fit tree $\mathbf{t}^{(b,k)}$ to $\{(\mathbf{x}_i^{(k)}, r_{i,k})\}_{i\in G_{b,k}}$.
    $\mu_{b,k} \leftarrow \frac{1}{n}\sum_{i=1}^n \mathbf{t}^{(b,k)}(\mathbf{x}_i^{(k)})$.
    $\widetilde{\mathbf{t}}_b^{(k)}(\mathbf{x}) \leftarrow t_b^{(k)}(\mathbf{x}) - \mu_{b,k}$.
    $\widetilde{\mathbf{t}}_b^{(k)}(x) \leftarrow \max\{-M, \min\{\widetilde{\mathbf{t}}_b^{(k)}(\mathbf{x}), M\}\}$.
    $\widehat{\boldsymbol{\beta}}_b \leftarrow \widehat{\boldsymbol{\beta}}_{b-1} + \mu_{b,k}$.
    $\mathbf{f}_b^{(k)} \leftarrow \frac{b-1}{b}\mathbf{f}_{b-1}^{(k)} + \frac{\lambda}{b}\widetilde{\mathbf{t}}_b^{(k)}$.
  end for
end for
Output: $\widehat{\mathbf{f}}(x) \leftarrow \widehat{\boldsymbol{\beta}}_B + \frac{1+\lambda}{\lambda}\sum_{k=1}^p \mathbf{f}_k^{(B)}(\mathbf{x}^{(k)})$.

Parallelization and inference. Algorithm 1 is a minimal parallelized variation on EBMs. At each round b, the new set of p univariate trees $(\mathbf{t}_b^{(k)})_{k=1}^p$ are fitted in parallel by regressing the current residual $r_{i,k}$ on each of the p features $\mathbf{x}_i^{(k)}$. The trees are then centered and clipped, before being introduced into the ensemble as part of the average: $\mathbf{f}_b^{(k)} \leftarrow \frac{b-1}{b}\mathbf{f}_{b-1}^{(k)} + \frac{1}{b}\widetilde{\mathbf{t}}_b^{(k)}$.

Averaging yields a kernel ridge regression limit and a CLT, but (as in Boulevard) the fitted signal is asymptotically shrunk, requiring rescaling by $\frac{1+\lambda}{\lambda}$. In Section E, we show that the prediction made by Algorithm 1 converge to a kernel-ridge-regression. Informally:

$$\widehat{\mathbf{f}}_b(\mathbf{x}) \xrightarrow[b\to\infty]{a.s.} \bar{\mathbf{y}} + \sum_{k=1}^{p} \mathbb{E}[\mathbf{s}^{(k)}(\mathbf{x})]^\top \mathbf{J}_n[\mathbf{I} + \mathbf{J}_n\mathbf{K}]^{-1}\mathbf{y}.$$

The down-scaling stems from the matrix $[\mathbf{I} + \mathbf{J}_n\mathbf{K}]^{-1}$: When the aggregated kernel $\mathbf{K} = \sum_{k=1}^p \mathbf{K}^{(k)}$ becomes nearly the identity (i.e. the aggregated decision boundaries are fine enough), we see a $\frac{\lambda}{1+\lambda}$ shrinkage.

Difficulty in parallelization. The parallelization in Algorithm 1 does not come for free. One has to either use a small learning rate of $1/p$, or make a rather strong assumption on the feature kernels (see Assumption 4.7) for Algorithm 1 to provably converge.

Though in practice this is not an issue, we explore alternatives below for the sake of completeness. One is to abandon parallelism, and randomly choose a single feature to fit a tree to during each boosting round. This is presented in Algorithm 3 (see Appendix C), which converges to the same fixed point as Algorithm 1, but without the computational gains.

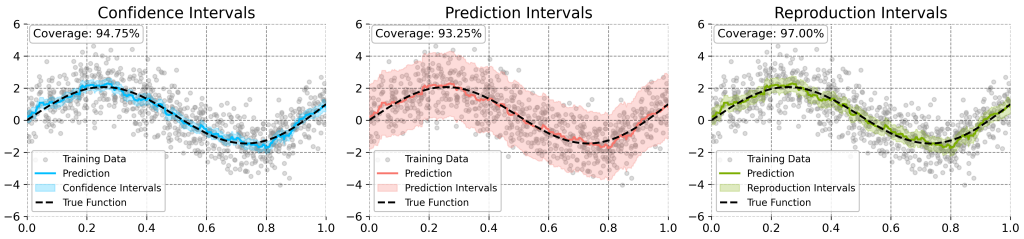

Figure 2: Example of confidence and prediction intervals on a simple 1D function $y = 2\sin(2\pi x) + x^2$.

Leave-one-out procedure. One problem remains. Algorithms 1 and 3 converges to $\frac{\lambda}{1+\lambda}\mathbf{f} \leq \mathbf{f}/2$, and so require rescaling by $\frac{1+\lambda}{\lambda}$. We provide an alternative inspired by the backfitting-like procedure of Fang et al. (2025) in Algorithm 2 that does not require rescaling.

This hybrid backfitting-boosting procedure amounts to not using predictions from trees trained on feature $\mathbf{x}^{(k)}$ when fitting a new tree $\mathbf{t}_b^{(k)}$ for it[3] – instead using residuals $\mathbf{y}_i - (\widehat{\boldsymbol{\beta}}_b + \sum_{\ell \neq k} \mathbf{f}_{b-1}^{(\ell)}(\mathbf{x}_i^{(\ell)}))$. Although this does not require rescaling, it produces complicated confidence intervals that are harder to compute efficiently. Informally, Algorithm 2 converges to:

$$\bar{\mathbf{y}} + \sum_{k=1}^{p} \mathbb{E}[\mathbf{s}^{(k)}(\mathbf{x})]^\top (\mathbf{I} - \mathbb{E}\mathbf{S}^{(k)})^{-1} \mathbf{J}_n \Big[\mathbf{I} - (\mathbf{I} + \mathcal{K})^{-1}\mathcal{K}\Big]\mathbf{y},$$

where $\mathcal{K} := \sum_{k=1}^{p} (\mathbf{I} - \mathbb{E}\mathbf{S}^{(k)})^{-1}\mathbf{J}_n \mathbb{E}\mathbf{S}^{(k)}$ is a somewhat more complicated combined kernel that unlike Algorithms 1 and 3, does not decompose into the sum of the individual feature kernels.

Comparison to vanilla EBMs. Our algorithms possess advantages over vanilla EBMs. The convergence to a kernel ridge regression allows us to show a central limit theorem for each function $\mathbf{f}^{(k)}$:

$$\|\mathbf{r}^{(k)}(\mathbf{x})\|_2^{-1} \left(c_E \cdot \widehat{\mathbf{f}}^{(k)}(\mathbf{x}) - \mathbf{f}^{(k)}(\mathbf{x})\right) \xrightarrow{d} \mathcal{N}(0, \sigma^2),$$

where c_E is a scaling factor that is $\frac{\lambda}{1+\lambda}$ for Algorithms 1 and 3, and 1 for Algorithm 2. $\mathbf{r}^{(k)}$ are the kernel weights for the kernel ridge regression predictions $\mathbf{r}^{(k)\top}\mathbf{y}$. For instance, Algorithms 1 and 3 converge to $\mathbf{r}^{(k)\top}\mathbf{y} := \mathbb{E}\left[\mathbf{s}^{(k)}(\mathbf{x})\right]^\top \mathbf{J}_n \left[\mathbf{I} + \mathbf{J}_n\mathbf{K}\right]^{-1}\mathbf{y}$.

This asymptotic normality yields confidence intervals for each $\mathbf{f}^{(k)}$ and prediction intervals for the response $\mathbf{y}$ – as long as we can compute $\|\mathbf{r}^{(k)}(\mathbf{x})\|_2$ efficiently. It also allows us to prove guarantees on the generalization error of each algorithm. We later show that $\|\mathbf{r}^{(k)}(\mathbf{x})\|_2 = \Theta(n^{-1/3})$, allowing us to show that our algorithms achieve the minimax-optimal MSE of $O(p\, n^{-2/3})$ for fitting Lipschitz GAMs. In comparison, vanilla EBMs hold *no such guarantees*.

[3]This can be interpreted as treating other functions $\mathbf{f}^{(\ell)} \neq \mathbf{f}^{(k)}$ as nuisances and learning them via boosting, while learning $\mathbf{f}^{(k)}$ as a random forest.

Comparison to Boulevard. The original gradient boosting algorithm with Boulevard regularization of Zhou and Hooker (2019) converges to a kernel ridge regression of the form $\mathbb{E}[\mathbf{s}(\mathbf{x})]^\top(\lambda^{-1}\mathbf{I} + \mathbb{E}[\mathbf{S}])^{-1}\mathbb{E}[\mathbf{S}]\mathbf{y}$. This is similar to our result for Algorithms 1 and 3, but we achieve convergence for each centered univariate function while they achieve convergence only for the overall function $\mathbf{f}$. As a consequence, their method has an MSE of $O(n^{-\frac{1}{d+1}})$ that grows exponentially in dimension. This is not even minimax-optimal for the nonparametric Lipschitz regression setting they are in.

The variants of Fang et al. (2025) incorporating dropout and parallel training converge to slightly different fixed points, but ultimately achieve the same asymptotic MSE rates exponential in dimension.

4 THEORY

We now work towards proving the central limit theorem outlined above. To begin, we introduce a few definitions. First, we use regression trees as base learners. These can be understood as a linear transformation from the observed signal to the predictions via a smoothing tree kernel, formally defined below:

Definition 4.1 (Regression trees)**.** A regression tree $\mathbf{t}_n$ trained on n datapoints segments the covariate space $[0,1]^d$ into a partition of hyper-rectangles $\{A_i\}_{i=1}^m$. When $A(\mathbf{x})$ is the rectangle containing $\mathbf{x}$, and $s_{n,j}(\mathbf{x})$ is the frequency at which training point $\mathbf{x}_j$ and test point $\mathbf{x}$ share a leaf, $s_{n,j}(\mathbf{x}) = \frac{\mathbb{1}(\mathbf{x}_j \in A(\mathbf{x}))}{\sum_{k=1}^n \mathbb{1}(\mathbf{x}_k \in A(\mathbf{x}))}$, the regression tree $\mathbf{t}_n$ predicts $\mathbf{t}_n(\mathbf{x}) = \sum_{j=1}^n s_{n,j}(\mathbf{x}) \cdot y_j$.

This suggests that $s_{n,j}(\mathbf{x})$, the frequency at which training point $\mathbf{x}_j$ and test point $\mathbf{x}$ share a leaf, defines a similarity measure. This observation is insightful. In fact, the frequency at which training points $\mathbf{x}_i, \mathbf{x}_j$ share a leaf, $s_{n,j}(\mathbf{x}_i)$, yields a kernel![4]

Definition 4.2 (Structure vectors and matrices)**.** Let $\mathbf{t}_n$ be a tree. $\mathbf{s}_n(\mathbf{x}) = (s_{n,1}(\mathbf{x}), ..., s_{n,n}(\mathbf{x}))^\top$ be the structure vector of $\mathbf{x}$. The matrix $\mathbf{S}_n = (s_{n,j}(\mathbf{x}_i))_{i,j=1}^n$ with rows $\mathbf{s}_n(\mathbf{x}_i)^\top$ is the structure matrix of $\mathbf{t}_n$.

[4]Theorem 3 of Zhou and Hooker (2019) shows that $\mathbb{E}\mathbf{S}_n$ is p.s.d. with spectral norm no more than 1.

Following Zhou and Hooker (2019); Fang et al. (2025), we require that the tree *structures* be separated from the leaf *values*, and stabilize after a burn-in period:

Assumption 4.3 (Integrity)**.** Assume

1. (Structure-Value Isolation) The density for $\mathbf{s}_b(\mathbf{x})$ is independent from the terminal leaf value $\mathbf{y} \in \mathbb{R}$.
2. (Non-adaptivity) For each feature k, there exists a set of probability measures $\mathcal{Q}_n^{(k)}$ for tree structures such that all tree structures $\mathbf{S}_b^{(k)}$ are independent draws from $\mathcal{Q}_n^{(k)}$ after some finite b'.

The former can be satisfied by refitting leaf values on a hold-out dataset Wager and Athey (2017). Non-adaptivity may occur naturally, but can also be enforced by sampling from previous tree structures after a burn-in period. In practice, our algorithms tend to perform well even in the absence of these assumptions.

Tree Geometry We control the geometry and size of leaves to ensure locality and stable averages. These assumptions control the MSE of the final estimate – Zhou and Hooker (2019) and Fang et al. (2025) make spiritually similar assumptions, but with different scalings in dimension that do not avoid the curse of dimensionality. Again, we avoid this problem due to the GAM structure of our problem setup:

Assumption 4.4 (Bounded Leaf Diameter)**.** Write $\mathsf{diam}(A) = \sup_{\mathbf{x}_1,\mathbf{x}_2 \in A} \|\mathbf{x}_1 - \mathbf{x}_2\|$. For any leaf A in a tree with structure $\Pi \in Q_n^{(k)}, \forall k = 1, \cdots, p$, we need $\sup_{A\in\Pi} \mathsf{diam}(A) = O(d_n) = O(n^{-1/3}) = o(\frac{1}{\log n})$.

Assumption 4.5 forces trees to perform only *local* averaging by keeping leaves well-balanced. Assumption 2.1 allows depths as large as $\Omega(\log n)$, which is natural for balanced binary trees.

Assumption 4.5 (Increased Minimal Leaf Size)**.** For any $\nu > 0$, the leaf geometric volume v_n is bounded such that $v_n = n^{-\frac{2}{3}+\nu} < n^{-\frac{1}{3}} = O(d_n)$.

This is natural: the leaf count cannot exceed the effective histogram resolution. Enforcing at least $n^{2/3-\nu}$ points per leaf rules out overly fine, high-variance partitions.

Assumption 4.6 (Restricted Tree Support)**.** For a tree constructed using feature k, the cardinality of the tree space $Q_n^{(k)}$ is bounded by $O\Big(n^{-1/3}\exp(n^{2/3-\epsilon_n})\Big)$, for some small $\epsilon_n \to 0$.

This is a technically convenient assumption for us to generalize it in random design case. It bounds the effective complexity of the tree class so we can use uniform LLNs and contraction arguments. The assumption still permits an super-polynomial large class in n, which is far looser than practice: vanilla EBMs use very shallow trees (typically depth 1–2) on a fixed histogram grid with at most a few hundred bins per feature, so the effective class is much smaller (roughly polynomial in n).

Unlike Zhou and Hooker (2019) and Fang et al. (2025), the assumptions above are dimension-free. They enforce asymptotic locality of the partition, allowing for control over bias and variance. Assumption 4.5 ensures stability of the structure matrix spectrum, allowing us to recover the minimax-optimal rate later.

Warmup: Theory for Algorithm 3. As mentioned earlier, the theory for Algorithm 3 is the easiest to prove. With the following observation, this amounts to a straightforward extension of Zhou and Hooker (2019). The learner updates a feature's kernel at random each round, so the tree kernel chosen by Algorithm 3 amounts to $\mathbb{E}\mathbf{S} = \frac{1}{p}\sum_{k=1}^{p} \mathbb{E}[\mathbf{S}^{(k)}]$.

This observation allows us to apply the theory of Zhou and Hooker (2019) almost verbatim. As such, Algorithm 3 gives a spiritually similar fixed point iteration $\tilde{\mathbf{y}}^* := \sum_k \hat{\mathbf{y}}_k^* = \lambda \mathbf{J}_n \mathbb{E}\mathbf{S}(\mathbf{y} - \tilde{\mathbf{y}}^*)$[5], modulo centering. This leads to a ensemble-wise fixed point of $\tilde{\mathbf{y}}^* = (\mathbf{I} + \mathbf{J}_n\mathbb{E}\mathbf{S})^{-1}\mathbf{J}_n\mathbb{E}\mathbf{S}\mathbf{y}$ and the subsequent feature-wise fixed points $\hat{\mathbf{y}}_k^* = \lambda\mathbf{J}_n\mathbb{E}\mathbf{S}(\mathbf{I} - (\mathbf{I} + \mathbf{J}_n\mathbb{E}\mathbf{S})^{-1}\mathbf{J}_n\mathbb{E}\mathbf{S})\mathbf{y} = \lambda\mathbf{J}_n\mathbb{E}\mathbf{S}^{(k)}(\mathbf{I} + \mathbf{J}_n\mathbb{E}\mathbf{S})^{-1}\mathbf{y}$. Finite sample convergence and asymptotic normality holds following the same recipe discussed in Zhou and Hooker (2019).

However, this sequential random update does not parallelize over features. To address these concerns we return to Algorithm 1 and analyze its convergence, which we will discuss in Section 4.

Feature-specific (specialized) kernels. Still, the above intuition is instructive. Unlike Zhou and Hooker (2019) and Fang et al. (2025), which consider a single expected kernel $\mathbb{E}\mathbf{S}$, EBMs induce *feature-specific* kernels $\mathbf{K}^{(k)} := \mathbb{E}[\mathbf{S}^{(k)}]$. This is a double-edged sword. On one hand, one has to control the way these kernels interact in our algorithms. On the other, it enables a clearer characterization of partial dependence, while allowing us to demonstrate feature-wise convergence, which in turn yields feature-wise confidence intervals.

Theory for Algorithm 1. As mentioned earlier in Section 3, introducing parallelization introduces theoretical challenges. One way to prove that Algorithm 1 converges is to use a learning rate of $\lambda = \frac{1}{p}$. However, this yields no speedup from Algorithm 3 on average, as sampling with probability $1/p$ and averaging across p updates are the same in expectation.

[5] We drop intercepts as they are orthogonal to centering.

Another way, that allows us to use a learning rate of $\lambda \approx 1$, is to make the following assumption:

Assumption 4.7 (Projector-like feature kernels and conditional orthogonality)**.** Let $\mathbf{K}^{(k)} = \mathbb{E}[\mathbf{S}^{(k)}]$ and $\mathbf{J}_n := \mathbf{I}_n - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$. For each feature k, let

$$\mathcal{H}_k := \left\{ \mathbf{v} \in \mathbb{R}^n : \exists\, g : \mathbb{R} \to \mathbb{R}, \mathbf{v}_i = g(\mathbf{x}_i^{(k)}),\ \mathbf{1}^\top \mathbf{v} = 0 \right\}$$

with P_k the orthogonal projector onto $\mathcal{H}_k$. Assume:

- **Projector-like approximation.** There exists $\varepsilon_{n,k} \to 0$ with $\left\| \mathbf{J}_n \mathbf{K}^{(k)} \mathbf{J}_n - P_k \right\|_{\text{op}} \le \varepsilon_{n,k}$.
- **Conditional orthogonality.** For $a \neq k$, the additive subspaces are orthogonal: $P_a P_k = \mathbf{0}$.

Assumption 4.7 is technically convenient but admittedly strong, especially the conditional orthogonality requirement. We therefore emphasize that it is used only to analyze Algorithm 1. As a fallback, Algorithm 2 removes this requirement and needs only Assumption 4.9. Intuitively, the projector-like condition says the centered expected tree kernel for feature k behaves like the orthogonal projector onto mean-zero functions of $\mathbf{x}^{(k)}$, while conditional orthogonality asserts these per-feature subspaces have negligible overlap across k (similar in spirit to "no concurvity" conditions for uniqueness in additive models, e.g., Hastie and Tibshirani, 1986); this can hold exactly under feature independence and can be a reasonable approximation under weak dependence when partitions are shallow and not aligned across features.

Now we discuss the finite-sample convergence for Algorithm 1 as promised. A way forward is to first establish almost surely convergence as we keep boosting with $b \to \infty$, then seek asymptotic normality when sample size grows. The first part is usually handled by showing that the difference to a set of postulated fixed points is a stochastic contraction mapping, decaying to 0. However, by the design of Algorithm 1, stepwise mean convergence can be fragile when $\lambda \approx 1$, hence we enforce Assumption 4.7 to ensure stability. Formally, the convergence behavior can be characterized in Theorem 4.8, stated below.

Theorem 4.8. *Define* $\widehat{\mathbf{y}}_b := \widehat{\boldsymbol{\beta}}_b + \sum_{a=1}^p \widehat{\mathbf{y}}_b^{(a)}, \mathbf{J}_n = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$. *The fixed points are*

$$\widetilde{\mathbf{y}}_k^* = \mathbf{J}_n \mathbf{K}^{(k)} [\lambda^{-1}\mathbf{I} + \mathbf{J}_n \mathbf{K}]^{-1}\mathbf{y}, \qquad \mathbf{K} := \sum_{k=1}^p \mathbf{K}^{(k)},$$

and $\widehat{\mathbf{y}}^* = \widehat{\boldsymbol{\beta}}^* \mathbf{1} + \sum_{k=1}^p \widetilde{\mathbf{y}}_k^*$, *with* $\widehat{\boldsymbol{\beta}}^* = \bar{\mathbf{y}}$. *For each* k,

$$\widehat{\mathbf{y}}_b^{(k)} \xrightarrow{a.s.} \widetilde{\mathbf{y}}^{(k)*}, \qquad \widehat{\boldsymbol{\beta}}_b \xrightarrow{a.s.} \bar{\mathbf{y}} = \tfrac{1}{n}\mathbf{1}^\top \mathbf{y},$$

and hence $\widehat{\mathbf{y}}_b \xrightarrow{a.s.} \widehat{\mathbf{y}}^*$ *almost surely.*

This fixed point carries two different relevant and practical perspectives. When setting $\lambda = \frac{1}{p}$, one essentially implements a coordinate descent. It also recovers the vanilla boulevard randomized tree kernel by assigning uniform mass on featurewise kernels. Thus convergence can be shown without additional assumptions. Taking $\lambda = 1$, however, requires a more delicate characterization of tree behavior specified in Assumption 4.7. This condition implies that featurewise signals do not overlap with each other asymptotically, removing the need for averaging the final feature functions.[6]

Theory for Algorithm 2. As discussed before, one may argue that Assumption 4.7 is unrealistic. If one is not willing to use $\lambda = 1/p$ then, Algorithm 2 poses a remedy. This does not rely on Assumption 4.7, and only depends on a weaker assumption below.

Assumption 4.9 (Feature-kernel distinctness)**.** Let $\mathbf{K}^{(k)} = \mathbb{E}[\mathbf{S}^{(k)}]$. There exists $c_3 < 1$ such that, for all $a \neq k$, $\left\| \mathbb{E}[\mathbf{S}^{(a)}]\, \mathbb{E}[\mathbf{S}^{(k)}] \right\|_{\text{op}} \le c_3, \quad (p-1)\, c_3 < 1.$

Assumption 4.9 is a weaker version of Assumption 4.7: instead of requiring (approximate) orthogonality of feature-wise subspaces, it only rules out the degenerate case where different features induce nearly identical smoothing operators in expectation. We verified this assumption empirically with plausible results. See Table 10. This milder assumption is sufficient for establishing a similar finite-sample convergence result below.

Theorem 4.10. *Denote the prediction using feature* k *at boosting round* b *as* $\widehat{\mathbf{y}}_b^{(k)} = \frac{1}{b}\sum_{s=1}^b t^{(s,k)}(\mathbf{x}^{(k)})$. *Define the aggregated kernel* $\mathcal{K} := \sum_{k=1}^p (\mathbf{I} - \mathbb{E}\mathbf{S}^{(k)})^{-1}\mathbf{J}_n \mathbb{E}\mathbf{S}^{(k)}$. *Define feature-wise fixed points*

$$\widetilde{\mathbf{y}}_k^* = (\mathbf{I} - \mathbb{E}\mathbf{S}^{(k)})^{-1}\mathbf{J}_n \mathbb{E}\mathbf{S}^{(k)} \left[\mathbf{I} - (I + \mathcal{K})^{-1}\mathcal{K}\right] \mathbf{y}.$$

Then, $\widehat{\mathbf{y}}_b^{(k)} \xrightarrow{a.s.} \widetilde{\mathbf{y}}_k^*$, $\widehat{\mathbf{y}}_b \xrightarrow{a.s.} \widehat{\mathbf{y}}^*$, *and* $\widehat{\boldsymbol{\beta}}^* \xrightarrow{a.s.} \bar{\mathbf{y}} := \frac{1}{n}\mathbf{1}^\top \mathbf{y}$.

The operator $(\mathbf{I} - \mathbf{S}^{(k)})^{-1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}$, which we can interpret as a feature-wise KRR, is stacked with the fixed point of the residual created on the whole ensemble. In this product, $(\mathbf{I} - \mathbf{S}^{(k)})^{-1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}$ measures the marginal contribution of a particular feature k contribute to the ensemble prediction.

A Central Limit Theorem. We are now in a position to provide central limit theorems for all three algorithms. Unlike Zhou and Hooker (2019) and Fang et al. (2025), the GAM structure of our problem allows us to relax the rates within the assumptions on tree geometries. While the following lemma allows us

[6] This can be thought of as parallel coordinate descent. Averaging is conservative, so when can we not do so?

to check the Lindeberg-Feller conditions, it also allows us to recover the optimal minimax rate $n^{-2/3}$ for univariate non-parametric Lipschitz regression Yuan and Zhou (2015). Below, we use the subscript $E = \{A, B\}$ to label relevant variables in Algorithm 1 and 2.

Lemma 4.11 (Rate of Convergence). *Let* $\mathbf{B}_n^{(k)} := \{i : \left\|\mathbf{x}^{(k)} - \mathbf{x}_i^{(k)}\right\| \le d_n\}$ *be the points within distance* d_n *from test point* $\mathbf{x}$ *on feature* k. *Under Assumption 4.6, if* $|\mathbf{B}_n| = \Omega(n\, d_n)$ *and* $\inf_{A \in \Pi \in Q_n^{(k)}} \sum_i \mathbf{I}(\mathbf{x}_i \in A) = \Omega(n^{\frac{2}{3}})$, *then* $\left\|\mathbf{k}^{(k)}\right\|_2, \left\|\mathbf{r}_E^{(k)}\right\|_2 = \Theta(n^{-\frac{1}{3}})$ *almost surely.*

The rate of the boulevard vector enables a conditional CLT, informally $\frac{\mathbf{r}_E^{(k)}(\mathbf{x})^\top \mathbf{f}(\mathbf{X}) - \mathbf{f}^{(k)}(\mathbf{x})}{\left\|\mathbf{r}_E^{(k)}(\mathbf{x})\right\|} \xrightarrow{d} \mathcal{N}(0, \sigma^2)$. Working with either Assumption 4.7 or Assumption 4.9, we analyze the form of the combined kernel, and conclude with the following central limit theorem.

Theorem 4.12 (Asymptotic Normality). *Define* $c_A = \frac{\lambda}{1+\lambda}, c_B = 1$. *For all* $k = 1, \cdots, p$, *we have that:*

$$\left\|\mathbf{r}_E^{(k)}(\mathbf{x})\right\|^{-1} \left(\widehat{\mathbf{f}}_E^{(k)}(\mathbf{x}) - c_E \mathbf{f}^{(k)}(\mathbf{x})\right) \xrightarrow{d} \mathcal{N}(0, \sigma^2)$$

Normality of the intercept term $\hat{\boldsymbol{\beta}}$ is also established and discussed in Lemma F.15. Similar to Fang et al. (2025), our CLTs yields a subsequent risk bound:

Corollary 4.13. $\mathbb{E}[(\frac{1}{c_E}\hat{\mathbf{f}}_E^{(k)}(\mathbf{x}) - \mathbf{f}^{(k)}(\mathbf{x}))^2] \lesssim \frac{\sigma^2}{c_E^2} p\, n^{-\frac{2}{3}}$.

This result shows that our algorithms achieve the minimax-optimal MSE of $O(pn^{-2/3})$ for fitting Lipschitz p-dimensional GAMs Yuan and Zhou (2015). This avoids the curse of dimensionality, significantly improving upon the rates of Zhou and Hooker (2019); Fang et al. (2025).

5 INFERENCE AND COMPUTATION

We can now derive confidence and prediction intervals from the result on asymptotic normality above. Here, we construct practical intervals and tests, and shows how to compute the required standard errors in *bin space* so that inference time is independent of n. Throughout the section, we estimate the noise variance with a consistent variance estimates $\hat{\sigma}^2$ via in-sample or out-of-bag estimation.

Confidence Intervals. We construct $(1-\alpha)$ confidence intervals for each of {individual feature effects, the intercept, the values of a prediction at a new $\mathbf{x}$}.

- $c_E^{-1}\widehat{\mathbf{f}}_E^{(k)}(\mathbf{x}) \pm z_{1-\alpha/2}\, c_E^{-1}\hat{\sigma}\, \|\mathbf{r}_E^{(k)}(\mathbf{x})\|$ yields a $(1-\alpha)$-confidence interval for the feature effect. By similar arguments in Fang et al. (2025), the coverage rate will converge to $(1-\alpha)$ almost surely. This is validated by an example in Figure 3.
- For the intercept term, by Lemma F.15, $\widehat{\boldsymbol{\beta}}$ is asymptotically normal with variance σ^2/n, yielding the usual interval $\widehat{\boldsymbol{\beta}} \pm z_{1-\alpha/2}\, \hat{\sigma}/\sqrt{n}$
- For the overall response, write $\widehat{f}_E(\mathbf{x}) = \widehat{\boldsymbol{\beta}} + c_E^{-1} \sum_{k=1}^p \widehat{\mathbf{f}}_E^{(k)}(\mathbf{x})$. A $(1-\alpha)$ confidence interval is $\widehat{f}_E(\mathbf{x}) \pm z_{1-\alpha/2} c_E^{-1} \hat{\sigma} \sqrt{\frac{1}{n} + \|\sum_{k=1}^p \mathbf{r}_E^{(k)}(\mathbf{x})\|_2^2}$.

Recall that $\mathbf{r}_E^{(k)}(\mathbf{x})$ denotes the per-feature influence vector. Under Assumption 4.7, $\|\sum_{k=1}^p \mathbf{r}_E^{(k)}(\mathbf{x})\|_2^2 = \sum_{k=1}^p \|\mathbf{r}_E^{(k)}(\mathbf{x})\|_2^2$, so the variance contribution is the sum of feature-wise terms. Without Assumption 4.7, this replacement is conservative by Cauchy–Schwarz.

Fang et al. (2025) examines two further variants that are immediately implementable:

- **Reproduction Intervals.** All intervals above can be modified to be *reproduction* intervals for the predictions of a new model trained on an independent dataset by scaling their widths by $\sqrt{2}$.
- **Prediction Intervals.** We can construct a prediction interval for a new y given $\mathbf{x}$ by computing $c_E{}^{-1}\widehat{\mathbf{f}}_E^{(k)}(\mathbf{x}) \pm z_{1-\alpha/2}\, c_E{}^{-1}\hat{\sigma}\sqrt{1 + \|\mathbf{r}_E^{(k)}(\mathbf{x})\|_2^2}$.

Binning. The current implementation of EBMs Nori et al. (2019) utilizes histogram trees. As we worked off this implementation, it was natural to construct intervals with the binned data, instead of working with the original data. A more detailed discussion of the discretization error induced by binning is deferred to Appendix D.3.

As such, we compress all n samples along feature k into $m_k \ll n$ histogram bins and work entirely in *bin space*. In Appendix D, we construct a decomposition of the structure matrix that allows us to show that computing the norm of $\mathbf{r}^{(k)}(\mathbf{x})$ reduces to solving a $m_k \times m_k$ system of linear equations, and computing two m_k-length dot products.

After this compression, the dependence on n disappears. Denote m as the maximum number of bins among all k binnings. Caching $\mathbf{M}$ takes $O(pm^3)$, while each per-point inference query takes only $O(m^2)$ time. As bin numbers are often capped to 255 or 511 (independent of n), we can compute our intervals in time independent of the number of samples. This outperforms Fang et al. (2025)'s runtime of $O(nd_{\text{eff}}^2)$.

Practical Guidance. In practice, Algorithm 1 relies on conditional orthogonality; we recommend mon-

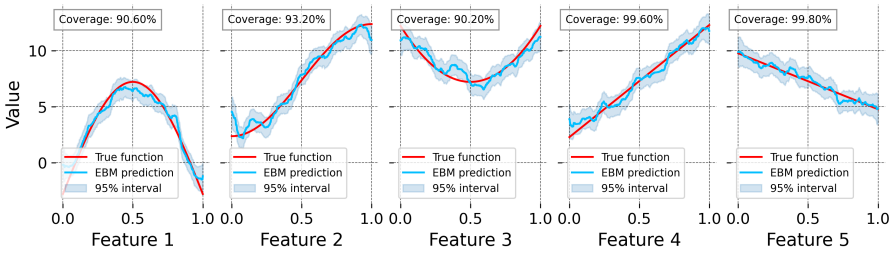

Figure 3: Per-feature CIs and coverages for $f(x) = -5+10\sin(\pi x^{(1)})+5\cos(\pi x^{(2)})+20(x^{(2)}-0.5)^2+10x^{(3)}-5x^{(4)}$.

itoring simple convergence diagnostics, and if convergence appears unstable, falling back to Algorithm 2. In our experiments, the number of boosting rounds B in the range 100–300 worked well. A smaller learning rate λ typically improves finite-sample stability but can yield slightly wider intervals. Subsample rates in $[0.5, 0.8]$ were robust. For binning, we found that using either the usual EBM defaults (e.g., 64 or 255 bins) or scaling the bin count as $m \asymp n^{1/3}$ works well; the latter matches our assumptions.

6 NUMERICAL EXPERIMENTS

Predictive Accuracy We compare Algorithm 1 to other benchmark models on real world data from the UCI Machine Learning Repository in Figure 5. Algorithm 1 is competitive with EBMs across the Wine Quality, Obesity and Air Quality datasets, while converging much faster with parallelization. However, these datasets exhibit higher order interactions. Algorithm 1 and EBMs ultimately underperform random forests and gradient boosting, though their added flexibility allows them to outperform a linear model.

However, the above setup had hyperparameters extensively tuned by Optuna over 100 trials. When this is not feasible, gradient boosting can be susceptible to overfitting. Figure 4 depicts two examples where this is the case, on $f(x) = \sin(4\pi x)$ and the diabetes dataset from Efron et al. (2004). Here, the performances are flipped. Algorithm 1 and EBMs perform the best with no need for early stopping, while the gradient boosting methods quickly overfit. EBMs achieve this via a tiny learning rate during cyclic boosting, while our implementation parallelizes over features for faster runtime.

Interval Coverage Rates To validate the proposed central limit theorems, we first construct feature-wise confidence intervals via binning and compute their coverage rates. Figure 3 shows good coverage and width. In combination with Figure 2, we can qualitatively argue that their performance appears desirable.

Figure 1 and Figure 7 yield an example on real-world data from the obesity dataset of Palechor and la Hoz Manotas (2019), regressing weight on other covariates. Our intervals offer much-needed uncertainty estimates for each effect.

Figure 6 compares the coverage and width of the CIs and PIs for the overall response, and RIs for another realization. We use the same conformal correction that Fang et al. (2025) use for the PIs only. Coverage is close to nominal for the CIs and PIs, while the RIs exhibit some overcoverage.

We further evaluate scalability in higher-dimensional, correlated synthetic additive models for Algorithm 2 under the realistic Assumption 4.9. In Figure 1, bootstrap EBMs achieve low RMSE but severely undercover and are slow due to repeated refits, while Algorithm 2 (and BRATD) attain near-nominal coverage; moreover, Algorithm 2 yields better RMSE and tighter intervals than BRATD at comparable or lower runtime. Compared with Fang et al. (2025), our intervals are also faster and better-behaved, since we avoid the curse of dimensionality and our runtime is independent of n.

Figure 2 shows that on real datasets with $p \approx 110$–130, Algorithm 2 maintains near-nominal 95% coverage (0.948–0.956), achieves RMSE comparable to the baselines, and is substantially faster than vanilla EBM. Relative to BRATD, it attains similar RMSE with lower training time on three of the four datasets, especially on the largest example.

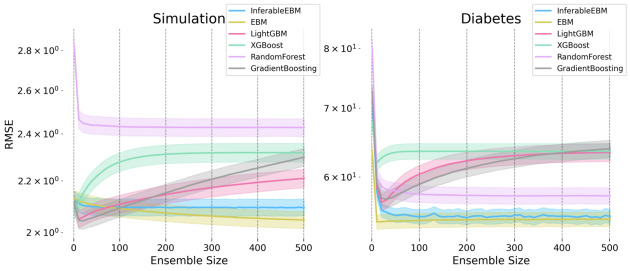

Figure 4: RMSE for Algorithm 1 and benchmarks with 2 s.e. bands over 50 trials, on $f(x) = \sin(4\pi x)$ and the diabetes dataset from Efron et al. (2004). Both EBM and Algorithm 1 are highly resistant to overfitting.

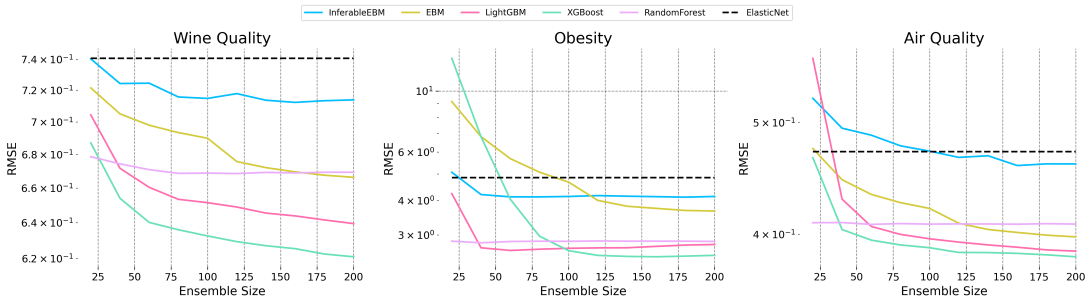

Figure 5: MSE horse races on UCI machine learning repository datasets. All hyperparameters tuned by Optuna.

7 CONCLUSION

Explainable boosting machines have been an important tool for producing glass-box models, used both to establish insight into the underlying patterns that generate data, and to diagnose potential non-causal behavior prior to deployment. The understanding presented by such methods must be qualified by appropriate uncertainty quantification. Current UQ for EBMs relies on bootstrapping which is both computationally intensive and lacks theoretical backing.

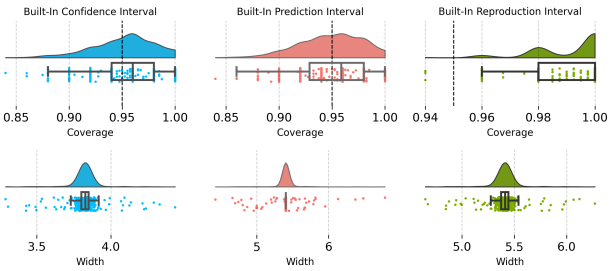

Figure 6: Coverage and width of our intervals over 50 trials, $n = 2000$, on the same function as Figure 3.

The Boulevard regularization method that we adopt to EBM here both removes the need to choose stopping times (see Figure 4), and generates a limit that can be characterized as a kernel ridge regression. This enables us to directly derive a central limit theorem for predictions of the resulting model and their additive components, and to thereby both derive consistent estimates of their asymptotic variance and provide confidence intervals and tests.

This work opens several extensions. We can readily extend the function class to more complex additive structure; including isolated interaction terms is straightforward, although additive components with shared features require new identifiability conditions (e.g. Lengerich et al., 2020). More generally, these methods are extensible to mixtures of learners, such as the class of varying coefficient models discussed in Zhou and Hooker (2022). We can an also broaden the class of models to discrete outcomes and models with non-constant variance. Finally, the kernel form of the limit also admits combinations and comparisons across models as described in Ghosal et al. (2022).

p	Method	RMSE	Coverage	Width	Time (s)
5	Bootstrap EBM	0.514	0.290	0.386	26.42
	Algorithm 2	0.661	0.951	2.609	3.00
	BRATD	0.611	0.956	2.899	10.41
10	Bootstrap EBM	0.537	0.373	0.545	21.24
	Algorithm 2	0.788	0.951	3.069	5.90
	BRATD	0.866	0.935	3.531	13.58
15	Bootstrap EBM	0.563	0.453	0.675	31.51
	Algorithm 2	0.900	0.945	3.520	8.77
	BRATD	1.112	0.946	4.398	10.29
20	Bootstrap EBM	0.583	0.511	0.801	43.14
	Algorithm 2	1.004	0.948	3.901	11.56
	BRATD	1.332	0.948	5.121	10.70

Table 1: Synthetic Lipschitz GAMs (bootstrap EBM vs. Algorithm 2 vs. BRATD). Synthetic additive model with $n = 1000$ and $p \in \{5, 10, 15, 20\}$ (base functions are cycled): $f_1(x) = 2\sin(\pi x)$, $f_2(x) = (x - 0.5)^2$, $f_3(x) = 0.5 \times \mathbf{1}\{x > 0.3\}$, $f_4(x) = \sqrt{x}$, $f_5(x) = -1.5x$. We report mean RMSE, empirical coverage at nominal 95%, average interval width, and runtime over 10 repetitions ($\alpha = 0.05$; 250 bootstrap resamples for EBM).

Dataset	n	p	RMSE (Alg. 2)	Cov. (Alg. 2)	Time (s) (Alg. 2)
taiwanese_bankruptcy	6819	95	0.151	0.948	18.01
myocardial_infarction	1700	111	0.309	0.943	3.53
communities_and_crime	1994	127	0.140	0.953	5.68
dota2_results	50000	116	0.142	0.956	28.78

Dataset	RMSE (EBM)	Time (s) (EBM)	RMSE (BRATD)	Time (s) (BRATD)
taiwanese_bankruptcy	0.153	197.31	0.154	24.58
myocardial_infarction	0.309	52.62	0.311	3.78
communities_and_crime	0.135	178.41	0.144	7.80
dota2_results	0.013	3736.10	0.147	82.62

Table 2: We report RMSE, 95% empirical coverage, and training time for Algorithm 2 (top), and baseline RMSE and training time for vanilla EBM and BRATD (bottom). Baseline coverage is omitted because large-scale uncertainty estimation is not computationally scalable.

References

Apley, D. W. and Zhu, J. (2020). 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.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 785–794. ACM.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *The Annals of Statistics*, 32(2).

Fang, H., Tan, K., and Hooker, G. (2025). Statistical inference for gradient boosting regression.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.

Fryer, D., Strümke, I., and Nguyen, H. (2021). Shapley values for feature selection: The good, the bad, and the axioms.

Ghosal, I., Zhou, Y., and Hooker, G. (2022). The infinitesimal jackknife and combinations of models. *arXiv preprint arXiv:2209.00147*.

Hastie, T. and Tibshirani, R. (1986). Generalized Additive Models. *Statistical Science*, 1(3):297 – 310.

Hooker, G., Mentch, L., and Zhou, S. (2021). Unrestricted permutation forces extrapolation: Variable importance requires at least one more model, or there is no free variable importance.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. (2017). Lightgbm: A highly efficient gradient boosting decision tree. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Lengerich, B., Tan, S., Chang, C.-H., Hooker, G., and Caruana, R. (2020). Purifying interaction effects with the functional anova: An efficient algorithm for recovering identifiable additive models. In *International Conference on Artificial Intelligence and Statistics*, pages 2402–2412. PMLR.

Lou, Y., Caruana, R., and Gehrke, J. (2012). Intelligible models for classification and regression. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '12, page 150–158, New York, NY, USA. Association for Computing Machinery.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

Mentch, L. and Hooker, G. (2015). Quantifying uncertainty in random forests via confidence intervals and hypothesis tests.

Musco, C. and Musco, C. (2017). Recursive sampling for the nyström method.

Nori, H., Jenkins, S., Koch, P., and Caruana, R. (2019). Interpretml: A unified framework for machine learning interpretability.

Nychka, D. (1988). Bayesian confidence intervals for smoothing splines. *Journal of the American Statistical Association*, 83(404):1134–1143.

Palechor, F. M. and la Hoz Manotas, A. D. (2019). Estimation of Obesity Levels Based On Eating Habits and Physical Condition . UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5H31Z.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). " 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.

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

Silverman, B. W. (1985). Some aspects of the spline smoothing approach to non-parametric regression curve fitting. *Journal of the Royal Statistical Society: Series B (Methodological)*, 47(1):1–21.

Ustimenko, A., Beliakov, A., and Prokhorenkova, L. (2023). Gradient boosting performs gaussian process inference.

Ustimenko, A. and Prokhorenkova, L. (2022). Sglb: Stochastic gradient langevin boosting.

Wager, S. and Athey, S. (2017). Estimation and inference of heterogeneous treatment effects using random forests.

Wahba, G. (1983). Bayesian "confidence intervals" for the cross-validated smoothing spline. *Journal of the Royal Statistical Society. Series B (Methodological)*, 45(1):133–150.

Wood, S. N. (2006). On confidence intervals for generalized additive models based on penalized regression splines. *Australian & New Zealand Journal of Statistics*, 48(4):445–464.

Yoshida, T. and Naito, K. (2012). Asymptotics for penalized splines in generalized additive models.

Yuan, M. and Zhou, D.-X. (2015). Minimax optimal rates of estimation in high dimensional additive models: Universal phase transition.

Zhou, Y. and Hooker, G. (2019). Boulevard: Regularized stochastic gradient boosted trees and their limiting distribution.

Zhou, Y. and Hooker, G. (2022). Decision tree boosted varying coefficient models. *Data Mining and Knowledge Discovery*, 36(6):2237–2271.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [Yes; See Sections 2 and 3, and Appendices D and C]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [Yes. Otherwise, the remaining algorithms are modification of existing boosting methods and share their properties with those already in the literature – namely a runtime of $O(Bnp \log n)$, and auxiliary space of $O(Bnp)$.]
   (c) (Optional) Annonymized source code, with specification of all dependencies, including external libraries. [Yes. Source code provided as a .zip file. All experiments were run on a single machine with an Intel i9-13900K CPU, 128 GB of RAM, and an NVIDIA GeForce RTX 3090Ti GPU, though no GPU acceleration was employed.]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [Yes; See sections 2 and 4]
   (b) Complete proofs of all theoretical results. [Yes; see Appendices E, F, and G]
   (c) Clear explanations of any assumptions. [Yes; See section 4]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [Yes; see provided .zip file]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [Yes; see provided .zip file]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [Yes]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [Yes]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Not Applicable]
   (b) The license information of the assets, if applicable. [Not Applicable]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Not Applicable]
   (d) Information about consent from data providers/curators. [Not Applicable]
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Not Applicable]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. [Not Applicable]
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Not Applicable]
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Not Applicable]

A Notations

Symbol	Meaning / definition (with dimensions when helpful)
n	Sample size; data $(\mathbf{X}_n, \mathbf{y}_n) = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$ are i.i.d.
p	Number of features (coordinates) in $\mathbf{x} \in \mathbb{R}^p$.
i, j, ℓ	Generic indices: $i, j \in \{1, \dots, n\}$ for samples; $\ell \in \{1, \dots, p\}$ for features.
$\mathbf{x}_i$	i-th covariate vector: $\mathbf{x}_i = (\mathbf{x}_i^{(1)}, \dots, \mathbf{x}_i^{(p)})^\top \in \mathbb{R}^p$.
$\mathbf{x}_i^{(k)}$	Feature-k coordinate of $\mathbf{x}_i$ (scalar in the univariate EBM setting).
$\mathbf{X} \in \mathbb{R}^{n \times p}$	Design matrix with rows $\mathbf{x}_i^\top$.
y_i	Response for sample i (scalar).
$\mathbf{y} \in \mathbb{R}^n$	Response vector $(y_1, \dots, y_n)^\top$.
ϵ / ε	Noise; i.i.d. sub-Gaussian errors with $\mathbb{E}[\varepsilon] = 0$, $\mathrm{Var}(\varepsilon) = \sigma^2 < \infty$.
σ^2	Noise variance in Assumption 1.
$\mu(\mathbf{x})$	Density of covariates on their support; bounded as $0 < c_1 \le \mu(\mathbf{x}) \le c_2 < \infty$.
$\boldsymbol{\beta}$	True intercept: $\boldsymbol{\beta} := \mathbb{E}_{\mathbf{x},\epsilon}[y]$ (unconditional mean of y).
$\widehat{\boldsymbol{\beta}}_b$	Estimated intercept at boosting round b (Algorithm 1 / Alg. in appendix). Initialized as $\widehat{\boldsymbol{\beta}}_0 = \frac{1}{n}\mathbf{1}^\top \mathbf{y}$.
$\bar{\mathbf{y}}$	Sample mean of responses: $\bar{\mathbf{y}} := \frac{1}{n}\mathbf{1}^\top \mathbf{y}$.
$\mathbf{f}(\mathbf{x})$	True regression function in GAM form: $\mathbf{f}(\mathbf{x}) = \boldsymbol{\beta} + \sum_{k=1}^p \mathbf{f}^{(k)}(\mathbf{x}^{(k)})$.
$\mathbf{f}^{(k)}$	True univariate component function for feature k; L-Lipschitz and mean-zero w.r.t. marginal of $\mathbf{x}^{(k)}$.
$\widehat{\mathbf{f}}$	Generic fitted predictor; in Algorithm 1 output is $\widehat{\mathbf{f}}(\mathbf{x}) = \widehat{\boldsymbol{\beta}}_B + \frac{1+\lambda}{\lambda}\sum_{k=1}^p \mathbf{f}_B^{(k)}(\mathbf{x}^{(k)})$.
$\mathbf{f}_b^{(k)}$	Algorithm's stored additive component for feature k after round b (a function of $\mathbf{x}^{(k)}$). Initialized $\mathbf{f}_0^{(k)} \equiv 0$.
$\widehat{\mathbf{y}}_b^{(k)}$	Feature-k *vector* prediction on training points at round b (used in Theorems): $\widehat{\mathbf{y}}_b^{(k)} \in \mathbb{R}^n$.
$\widehat{\mathbf{y}}_b$	Total fitted values on training set at round b: $\widehat{\mathbf{y}}_b := \widehat{\boldsymbol{\beta}}_b + \sum_{a=1}^p \widehat{\mathbf{y}}_b^{(a)}$ (Theorem 1).
$\mathbf{1}$	All-ones vector in $\mathbb{R}^n$.
$\mathbf{I}_n$ / $\mathbf{I}$	Identity matrix in $\mathbb{R}^{n \times n}$ (often written $\mathbf{I}$ when n is clear).
$\mathbf{J}_n$ / $\mathbf{J}_n$	Centering projector: $\mathbf{J}_n = \mathbf{I}_n - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$.

Table 3: Basic data & model notation.

Symbol	Meaning / definition (with dimensions when helpful)
b	Boosting round index ($b = 1, 2, \dots$).
B	Number of boosting rounds (loop bound in pseudocode).
λ	Learning rate in Boulevard-style moving average update.
ξ	Subsample rate: each $G_{b,k} \subset \{1, \dots, n\}$ includes each index i.i.d. with prob. ξ.
M	Truncation / clipping level: $\widetilde{t} \leftarrow \max\{-M, \min\{\widetilde{t}, M\}\}$.
$G_{b,k}$	Subsample index set used to fit tree for feature k at round b.
$r_{i,k}$	Residual used to fit feature-k tree at round b. Alg 1: $r_{i,k} = y_i - \big(\widehat{\boldsymbol{\beta}}_{b-1} + \sum_\ell \mathbf{f}_{b-1}^{(\ell)}(\mathbf{x}_i^{(\ell)})\big)$. Alg 2 (leave-one-out): sum excludes $\ell = k$.
$\mathbf{t}^{(b,k)}$	Regression tree trained at round b on feature k using pairs $\{(\mathbf{x}_i^{(k)}, r_{i,k})\}_{i \in G_{b,k}}$.
$\mu_{b,k}$	Mean (centering constant) of tree predictions on training set: $\mu_{b,k} = \frac{1}{n}\sum_{i=1}^n \mathbf{t}^{(b,k)}(\mathbf{x}_i^{(k)})$.
$\widetilde{\mathbf{t}}_b^{(k)}$	Centered (and then clipped) tree: $\widetilde{\mathbf{t}}_b^{(k)}(\mathbf{x}) = \mathbf{t}^{(b,k)}(\mathbf{x}) - \mu_{b,k}$, then clipped to $[-M, M]$.

Table 4: Algorithmic hyperparameters & indices.

Symbol	Meaning / definition (with dimensions when helpful)
$\mathbf{t}_n$	A regression tree trained on n datapoints; induces a partition $\{A_i\}_{i=1}^m$ of $[0,1]^d$ (Def. 1).
$A(\mathbf{x})$	The leaf (rectangle) containing $\mathbf{x}$ under the tree partition.
$s_{n,j}(\mathbf{x})$	Structure weight of training point j for query $\mathbf{x}$: $s_{n,j}(\mathbf{x}) = \frac{\mathbb{1}(\mathbf{x}_j \in A(\mathbf{x}))}{\sum_{k=1}^n \mathbb{1}(\mathbf{x}_k \in A(\mathbf{x}))}$.
$\mathbf{s}_n(\mathbf{x})$	Structure vector: $\mathbf{s}_n(\mathbf{x}) = (s_{n,1}(\mathbf{x}), \ldots, s_{n,n}(\mathbf{x}))^\top \in \mathbb{R}^n$.
$\mathbf{S}_n$	Structure matrix: $\mathbf{S}_n = (s_{n,j}(\mathbf{x}_i))_{i,j=1}^n \in \mathbb{R}^{n\times n}$; row i is $\mathbf{s}_n(\mathbf{x}_i)^\top$.
$\mathbf{S}_b^{(k)}$	Feature-k structure matrix at boosting round b (Assumption 2). **[I interpret $\mathbf{S}_b^{(k)}$ as $\mathbf{S}_n$ for the univariate tree fit on feature k at round b.]**
$\mathbf{K}^{(k)}$	Expected feature-k kernel: $\mathbf{K}^{(k)} := \mathbb{E}[\mathbf{S}^{(k)}]$ (also written $\mathbb{E}[\mathbf{S}^{(k)}]$).
$\mathbf{K}$	Aggregated kernel (main text / Thm 1): $\mathbf{K} := \sum_{k=1}^p \mathbf{K}^{(k)}$.
$\mathcal{Q}_n^{(k)}$	Distribution over feature-k tree structures after burn-in (Assumption 2).
$Q_n^{(k)}$	Tree-structure support / "tree space" for feature k (Assumptions 3–5).
b'	Burn-in iteration after which structures are i.i.d. from $\mathcal{Q}_n^{(k)}$ (Assumption 2).

Table 5: Structure vectors & matrices.

Symbol	Meaning / definition (with dimensions when helpful)
Π	A tree partition (set of leaves) drawn from $Q_n^{(k)}$.
$\mathsf{diam}(A)$	Leaf diameter: $\sup_{\mathbf{x}_1,\mathbf{x}_2\in A} \|\mathbf{x}_1 - \mathbf{x}_2\|$.
d_n	Target diameter scale: $d_n = O(n^{-1/3}) = o(1/\log n)$ (Assumption 3).
v_n	Leaf geometric volume bound, with $v_n = n^{-2/3+\nu}$ for any $\nu > 0$ (Assumption 4).
ϵ_n	Small sequence $\epsilon_n \to 0$ in restricted tree support bound (Assumption 5).

Table 6: Geometric & complexity parameters for trees.

Symbol	Meaning / definition (with dimensions when helpful)
$\widetilde{\mathbf{y}}_k^*$	Feature-k fixed point vector (Thm 1 / Thm 2). In Thm 1: $\widetilde{\mathbf{y}}_k^* = \mathbf{J}_n \mathbf{K}^{(k)}[\lambda^{-1}\mathbf{I} + \mathbf{J}_n\mathbf{K}]^{-1}\mathbf{y}$.
$\widehat{\boldsymbol{\beta}}^*$	Intercept fixed point: $\widehat{\boldsymbol{\beta}}^* = \bar{\mathbf{y}}$.
$\widehat{\mathbf{y}}^*$	Total fixed point fit on training set: $\widehat{\mathbf{y}}^* = \widehat{\boldsymbol{\beta}}^*\mathbf{1} + \sum_{k=1}^p \widetilde{\mathbf{y}}_k^*$.
$\mathcal{K}$	"combined kernel" for Algorithm 2: $\mathcal{K} := \sum_{k=1}^p (\mathbf{I} - \mathbb{E}\mathbf{S}^{(k)})^{-1}\mathbf{J}_n\mathbb{E}\mathbf{S}^{(k)}$.
$\mathcal{K}_{\text{bin}}$	$\mathcal{K}_{\text{bin}} = \sum_{k=1}^p \mathbb{E}[\mathbf{b}_n^{(k)}\mathbf{B}^{(k)}\mathbf{b}_n^{(k)\top}]$.
$\mathbf{r}^{(k)}$ / $\mathbf{r}_E^{(k)}(\mathbf{x})$	Kernel-ridge "influence / weight" vector used in CLTs and intervals; satisfies (example for Alg 1): $\mathbf{r}^{(k)\top}\mathbf{y} = \mathbb{E}[\mathbf{s}^{(k)}(\mathbf{x})]^\top\mathbf{J}_n[\mathbf{I}+\mathbf{J}_n\mathbf{K}]^{-1}\mathbf{y}$. Subscript $E \in \{A, B\}$ labels algorithm family (Alg 1 vs Alg 2) in Theorem 3.
$\mathbf{k}^{(k)}$ / $\mathbf{k}_n^{(k)}(\mathbf{x})$	Kernel vector (expected structure vector / feature-k similarity to training points). In binning appendix: $\mathbf{k}_n^{(k)}(\mathbf{x}) = \mathbb{E}[\mathbf{b}^{(k)}(\mathbf{x})^\top\mathbf{D}^{(k)}\mathbf{b}_n^{(k)\top}] \in \mathbb{R}^{1\times n}$.
c_E	Algorithm-dependent scaling constant in CLT: $c_A = \lambda/(1+\lambda)$ for Algorithms 1, 3; $c_B = 1$ for Algorithm 2 (Theorem 3).
$z_{1-\alpha/2}$	Standard normal quantile used for $(1-\alpha)$ intervals.

Table 7: Fixed points, CLT scaling, weight vectors.

Symbol	Meaning / definition (with dimensions when helpful)
$\mathcal{H}_k$	Centered feature-k additive subspace: $\mathcal{H}_k = \{\mathbf{v} \in \mathbb{R}^n : \exists g : \mathbb{R} \to \mathbb{R},\ \mathbf{v}_i = g(\mathbf{x}_i^{(k)}),\ \mathbf{1}^\top \mathbf{v} = 0\}$.
P_k	Orthogonal projector onto $\mathcal{H}_k$.
$\varepsilon_{n,k}$	Operator-norm approximation error s.t. $\|\mathbf{J}_n \mathbf{K}^{(k)} \mathbf{J}_n - P_k\|_{\text{op}} \le \varepsilon_{n,k} \to 0$.

Table 8: Additive subspaces and projectors; Assumption 4.7).

Symbol	Meaning / definition (with dimensions when helpful)
m_k	Number of histogram bins for feature k; $m := \max_k m_k$.
$B_r^{(k)}$	The r-th bin for feature k (a subset/interval of $\mathbb{R}$).
$r_i^{(k)}$	Bin index of sample i on feature k: unique r with $\mathbf{x}_i^{(k)} \in B_r^{(k)}$.
$\mathcal{L}^{(k)}(r)$	Set of bin indices in the same tree leaf as bin r (feature k): $\{s \in \{1, \dots, m_k\} : B_s^{(k)} \in A(B_r^{(k)})\}$.
$n_s^{(k)}$	Bin count: number of samples in bin s for feature k, $n_s^{(k)} = \sum_{j=1}^n \mathbf{1}(\mathbf{x}_j^{(k)} \in B_s^{(k)})$.
$N_{\mathcal{L}(r)}^{(k)}$	Leaf sample count for the leaf containing bin r: $N_{\mathcal{L}(r)}^{(k)} = \sum_{q \in \mathcal{L}^{(k)}(r)} n_q^{(k)}$.
$\lvert\mathcal{L}^{(k)}(r)\rvert$	Leaf bin count (number of bins in that leaf).
$\mathbf{Z}_n^{(k)} \in \{0,1\}^{n \times m_k}$	Sample-to-bin assignment matrix: $\mathbf{Z}_{i,j}^{(k)} = \mathbf{1}(r_i^{(k)} = j) = \mathbf{1}(\mathbf{x}_i^{(k)} \in B_j^{(k)})$.
$\mathbf{D}^{(k)}$ $=$ $\operatorname{diag}(\xi_1^{(k)}, \dots, \xi_{m_k}^{(k)})$	Diagonal rescaling to convert bin-normalization to sample-normalization, with $\xi_r^{(k)} = \sqrt{\frac{\lvert\mathcal{L}^{(k)}(r)\rvert}{N_{\mathcal{L}(r)}^{(k)}}}$.
$\mathbf{B}^{(k)} \in \mathbb{R}^{m_k \times m_k}$	Bin-structure matrix: $\mathbf{B}_{i,j}^{(k)} = \frac{\mathbf{1}(B_j^{(k)} \in A(B_i^{(k)}))}{\sum_{l=1}^{m_k} \mathbf{1}(B_l^{(k)} \in A(B_i^{(k)}))}$.
$\mathbf{b}_n^{(k)}$	Matrix mapping bin-space to sample-space. Initially given as $\mathbf{b}_n^{(k)} \in \mathbb{R}^{n \times m_k}$; later simplified to $\mathbf{b}_n^{(k)} = \mathbf{Z}_n^{(k)} \mathbf{D}^{(k)}$.
$\mathbf{b}^{(k)}(\mathbf{x}) \in \mathbb{R}^{m_k}$	Bin-space structure vector for query $\mathbf{x}$: $\mathbf{b}^{(k)}(\mathbf{x})^\top = \left(\frac{\mathbf{1}(B_j^{(k)} \in A(\mathbf{x}))}{\sum_{l=1}^{m_k} \mathbf{1}(B_l^{(k)} \in A(\mathbf{x}))} \right)_{j=1}^{m_k}$.
$\mathbf{h}^{(k)}(\mathbf{x})$	Defined in appendix lemma: $\mathbf{h}^{(k)}(\mathbf{x}) = \mathbb{E}[(\mathbf{D}^{(k)})^2 \mathbf{b}^{(k)}(\mathbf{x})] \in \mathbb{R}^{m_k}$.
$\mathbf{H}$	Sum of expected bin-kernels: $\mathbf{H} = \sum_{k=1}^p \mathbb{E}[\mathbf{D}^{(k)} \mathbf{B}^{(k)} \mathbf{D}^{(k)}]$.
$\mathbf{c}^{(k)}$	Bin counts vector: $\mathbf{c}^{(k)} := \mathbf{Z}_n^{(k)\top} \mathbf{1} \in \mathbb{R}^{m_k}$.

Table 9: Binning & histogram-tree appendix.

B Additional Literature

GAMs. Wahba (1983) established the asymptotic distribution of smoothing spline estimators via a Bayesian posterior covariance argument, Silverman (1985) developed the equivalent kernel representation and convergence theory, and Nychka (1988) analyzed the frequentist coverage of Bayesian confidence intervals for smoothing splines. Built on these developments, Wood (2006) investigated the coverage of penalized regression spline intervals in the GAM setting, while Yoshida and Naito (2012) provided a full asymptotic distribution theory for penalized spline estimators, establishing asymptotic normality for each additive component.

C Algorithms

Algorithm 2 Leave-one-out Inferable EBM

Input: $\mathbf{X} \in \mathbb{R}^{n\times p}$, $\mathbf{y} \in \mathbb{R}^n$, subsample rate ξ, rounds B, truncation level $M > 0$.
Init: $\widehat{\boldsymbol{\beta}}_0 \leftarrow \frac{1}{n}\sum_{i=1}^n \mathbf{y}_i$; $\mathbf{f}_k^{(0)} \leftarrow 0$ for all k.
for $b = 1$ **to** B **do**
  for $k = 1$ **to** p **in parallel do**
    Sample $G_{b,k} \subset \{1, \dots, n\}$ i.i.d. w.p. ξ.
    $r_{i,k} \leftarrow \mathbf{y}_i - (\widehat{\boldsymbol{\beta}}_{b-1} + \sum_{\ell\neq k} \mathbf{f}_{b-1}^{(\ell)}(\mathbf{x}_i^{(\ell)}))$.
    Fit tree $\mathbf{t}^{(b,k)}$ to $\{(\mathbf{x}_i^{(k)}, r_{i,k})\}_{i\in G_{b,k}}$.
    $\mu_{b,k} \leftarrow \frac{1}{n}\sum_{i=1}^n \mathbf{t}^{(b,k)}(\mathbf{x}_i^{(k)})$.
    $\widetilde{\mathbf{t}}_b^{(k)}(\mathbf{x}) \leftarrow t_b^{(k)}(\mathbf{x}) - \mu_{b,k}$.
    $\widetilde{\mathbf{t}}_b^{(k)}(x) \leftarrow \max\{-M,\ \min\{\widetilde{\mathbf{t}}_b^{(k)}(\mathbf{x}), M\}\}$.
    $\widehat{\boldsymbol{\beta}}_b \leftarrow \widehat{\boldsymbol{\beta}}_{b-1} + \mu_{b,k}$.
    $\mathbf{f}_b^{(k)} \leftarrow \frac{b-1}{b}\mathbf{f}_{b-1}^{(k)} + \frac{1}{b}\widetilde{\mathbf{t}}_b^{(k)}$.
  end for
end for
Output: $\widehat{\mathbf{f}}(x) \leftarrow \widehat{\boldsymbol{\beta}}_B + 2\sum_{k=1}^p \mathbf{f}_k^{(B)}(\mathbf{x}^{(k)})$.

Algorithm 3 Random Cyclic Inferable EBM

Input: $\mathbf{X} \in \mathbb{R}^{n\times p}$, $\mathbf{y} \in \mathbb{R}^n$, subsample rate ξ, rounds B, truncation level $M > 0$.
Init: $\widehat{\boldsymbol{\beta}}_0 \leftarrow \frac{1}{n}\sum_{i=1}^n \mathbf{y}_i$; $\mathbf{f}_k^{(0)} \leftarrow 0$ for all k.
for $b = 1$ **to** B **do**
  Randomly sample $k \sim$ Categorical$(1/p, ..., 1/p)$
  Sample $G_{b,k} \subset \{1, \dots, n\}$ i.i.d. w.p. ξ.
  $r_{i,k} \leftarrow \mathbf{y}_i - (\widehat{\boldsymbol{\beta}}_{b-1} + \sum_{\ell} \mathbf{f}_{b-1}^{(\ell)}(\mathbf{x}_i^{(\ell)}))$.
  Fit tree $\mathbf{t}^{(b,k)}$ to $\{(\mathbf{x}_i^{(k)}, r_{i,k})\}_{i\in G_{b,k}}$.
  $\mu_{b,k} \leftarrow \frac{1}{n}\sum_{i=1}^n \mathbf{t}^{(b,k)}(\mathbf{x}_i^{(k)})$.
  $\widetilde{\mathbf{t}}_b^{(k)}(\mathbf{x}) \leftarrow t_b^{(k)}(\mathbf{x}) - \mu_{b,k}$.
  $\widetilde{\mathbf{t}}_b^{(k)}(x) \leftarrow \max\{-M,\ \min\{\widetilde{\mathbf{t}}_b^{(k)}(\mathbf{x}), M\}\}$.
  $\widehat{\boldsymbol{\beta}}_b \leftarrow \widehat{\boldsymbol{\beta}}_{b-1} + \mu_{b,k}$.
  $\mathbf{f}_b^{(k)} \leftarrow \frac{b-1}{b}\mathbf{f}_{b-1}^{(k)} + \frac{1}{b}\widetilde{\mathbf{t}}_b^{(k)}$.
end for
Output: $\widehat{\mathbf{f}}(x) \leftarrow \widehat{\boldsymbol{\beta}}_B + 2\sum_{k=1}^p \mathbf{f}_k^{(B)}(\mathbf{x}^{(k)})$.

D Binning

D.1 Bin-level Decomposition of the Structure Matrix

In the special case where one uses histogram trees, there is a simpler form for the structure vector and matrix. When one performs binning automatically during pre-fitting, we can store the structure matrix in such a decomposition:

$$\mathbf{S}^{(k)} = \mathbf{b}_n^{(k)}\mathbf{B}^{(k)}{\mathbf{b}_n^{(k)}}^\top, \mathbf{B}_{i,j}^{(k)} = \left[\frac{\mathbf{1}(B_j^{(k)} \in A(B_i^{(k)}))}{\sum_{l=1}^{m_k}\mathbf{1}(B_l^{(k)} \in A(B_i^{(k)}))}\right]_{m_k\times m_k}, (\mathbf{b}_n^{(k)})_{i,j} = \left[\mathbf{1}(x_i^{(k)} \in B_j^{(k)})\frac{\sum_{l=1}^{m_k}\mathbf{1}\left(B_l^{(k)} \in A(B_j^{(k)})\right)}{\sum_{q=1}^{n}\mathbf{1}\left(\mathbf{x}_q^{(k)} \in A(B_j^{(k)})\right)}\right]_{n\times m_k},$$

where $A(\cdot)$ is the leaf that the bin (or sample) in its argument is in. The matrix $\mathbf{B}^{(k)}$ then is the bin-structure

matrix where the i,j-th entry contains the indicators of whether bin i is in the same leaf as bin j, normalized by the number of bins in the same leaf as bin i.

We now introduce a few more definitions:

1. **Bin index of a sample.** For each sample i, let $r_i^{(k)} \in \{1,\dots,m_k\}$ be the unique bin index with $\mathbf{x}_i^{(k)} \in B^{(k)}_{r_i^{(k)}}$. That is, this is the index of the histogram bin that sample i belongs to along feature k.

2. **The set of bins in a given leaf.** For a bin index r, define the set of bins in the unique tree leaf containing the bin $B_r^{(k)}$ with index r along feature k:

$$\mathcal{L}^{(k)}(r) := \left\{ s \in \{1,\dots,m_k\} : B_s^{(k)} \in A\left(B_r^{(k)}\right) \right\}.$$

   These are exactly the bins merged together by the tree into the same terminal node as bin r.

3. **How many samples are in each bin, and how many bins are there in each leaf?**

   The bin count $n_s^{(k)}$, or the number of samples in bin s along feature k, is given by:

$$n_s^{(k)} := \sum_{j=1}^{n} \mathbf{1}\left(\mathbf{x}_j^{(k)} \in B_s^{(k)}\right).$$

   The leaf sample count $N^{(k)}_{\mathcal{L}(r)}$ and leaf bin count $\left|\mathcal{L}^{(k)}(r)\right|$ are given by:

$$N^{(k)}_{\mathcal{L}(r)} := \sum_{q \in \mathcal{L}^{(k)}(r)} n_q^{(k)}, \quad \left|\mathcal{L}^{(k)}(r)\right| := \sum_{q=1}^{m_k} \mathbf{1}\left(q \in \mathcal{L}^{(k)}(r)\right).$$

   The leaf sample count $N^{(k)}_{\mathcal{L}(r)}$ denotes how many training samples ended up in the leaf containing bin r. The leaf bin count $\left|\mathcal{L}^{(k)}(r)\right|$ denotes how many bins that the leaf containing bin r aggregates.

4. **The assignment matrix of samples to bins.**

$$\mathbf{Z}_n^{(k)} \in \{0,1\}^{n \times m_k}, \text{ with } \mathbf{Z}_{i,j}^{(k)} := \mathbf{1}\left(r_i^{(k)} = j\right) = \mathbf{1}\left(\mathbf{x}_i^{(k)} \in B_j^{(k)}\right).$$

   This is the matrix where the i,j-th entry is 1 of sample i is in bin j, and 0 otherwise. Note that each sample lives in exactly one bin, so each row of $\mathbf{Z}_n^{(k)}$ has a single 1.

5. **Leaf-wise rescaling from bin-space to sample-space.**

   Define a diagonal matrix $\mathbf{D}^{(k)} = \operatorname{diag}\left(\xi_1^{(k)},\dots,\xi_{m_k}^{(k)}\right)$ with

$$\xi_r^{(k)} := \sqrt{\frac{\left|\mathcal{L}^{(k)}(r)\right|}{N^{(k)}_{\mathcal{L}(r)}}} = \frac{\sum_{l=1}^{m_k} \mathbf{1}\left(B_l^{(k)} \in A(B_r^{(k)})\right)}{\sum_{i=1}^{n} \mathbf{1}\left(\mathbf{x}_i^{(k)} \in A(B_r^{(k)})\right)} = \sqrt{\frac{\text{Number of bins in leaf containing bin } r}{\text{Number of training samples in leaf containing bin } r}}.$$

   The intuition is as follows. $\mathbf{B}^{(k)}$ normalizes over bins in a leaf, while $\mathbf{S}^{(k)}$ normalizes over samples in a leaf. Some bins have more samples than others. We must therefore correct the bin-structure matrix so that it normalizes over samples rather than bins, and so we construct $\mathbf{D}^{(k)}$ to do exactly that. Note $\xi_r^{(k)}$ is constant within a leaf, in the sense that if bins s and r are in the same leaf, we will then have $\xi_r^{(k)} = \xi_s^{(k)}$.

We then have the following result.

Lemma D.1 (Bin-Level Decomposition of Structure Matrix)**.** *We can decompose the structure matrix as follows:*

$$\mathbf{S}_n^{(k)} = \mathbf{b}_n^{(k)} \mathbf{B}^{(k)} \mathbf{b}_n^{(k)\top}, \quad \mathbf{B}_{i,j}^{(k)} = \left[\frac{\mathbf{1}(B_j^{(k)} \in A(B_i^{(k)}))}{\sum_{l=1}^{m_k} \mathbf{1}(B_l^{(k)} \in A(B_i^{(k)}))} \right]_{m_k \times m_k}, \quad \mathbf{b}_n^{(k)} = \mathbf{Z}_n^{(k)} \mathbf{D}^{(k)} \in \mathbb{R}^{n \times m_k}.$$

Similarly, we can also decompose the structure vector into the structure vector in bin space transformed to sample space:

$$\mathbf{s}_n^{(k)}(\mathbf{x}) = \mathbf{b}_n^{(k)} \mathbf{D}^{(k)} \mathbf{b}^{(k)}(\mathbf{x}), \quad \mathbf{b}^{(k)}(\mathbf{x})^\top = \left(\frac{\mathbf{1}(B_j^{(k)} \in A(\mathbf{x}))}{\sum_{l=1}^{m_k} \mathbf{1}(B_l^{(k)} \in A(\mathbf{x}))} \right)_{j=1,\ldots,m_k}.$$

Proof. Examine the i, j-th entry of the decomposition.

$$\left(\mathbf{b}_n^{(k)} \mathbf{B}^{(k)} \mathbf{b}_n^{(k)\top} \right)_{i,j} = \sum_{r=1}^{m_k} \sum_{s=1}^{m_k} \mathbf{Z}_{i,r}^{(k)} \xi_r^{(k)} \mathbf{B}_{r,s}^{(k)} \xi_s^{(k)} \mathbf{Z}_{j,s}^{(k)}$$

Since $\mathbf{Z}_{i,\cdot}^{(k)}$ and $\mathbf{Z}_{j,\cdot}^{(k)}$ are both one-hot vectors, the double sum collapses to

$$\left(\mathbf{b}_n^{(k)} \mathbf{B}^{(k)} \mathbf{b}_n^{(k)\top} \right)_{i,j} = \xi_{r_i^{(k)}}^{(k)} \mathbf{B}_{r_i^{(k)}, r_j^{(k)}}^{(k)} \xi_{r_j^{(k)}}^{(k)},$$

where $r_i^{(k)}$ is the bin index of the i-th sample across feature k.

There are two cases. First, if the i-th and j-th samples are not in the same leaf, then $\mathbf{B}_{r_i^{(k)}, r_j^{(k)}}^{(k)} = 0$, so the entry is 0 . This matches $\left(\mathbf{S}_n^{(k)} \right)_{i,j} = 0$.

Otherwise, both bins lie in the same leaf $L := \mathcal{L}^{(k)} \left(r_i^{(k)} \right) = \mathcal{L}^{(k)} \left(r_j^{(k)} \right)$. Then

$$\mathbf{B}_{r_i^{(k)}, r_j^{(k)}}^{(k)} = \frac{1}{|L|}, \quad \xi_{r_i^{(k)}}^{(k)} = \xi_{r_j^{(k)}}^{(k)} = \sqrt{\frac{|L|}{N_L^{(k)}}}.$$

Hence

$$\xi_{r_i^{(k)}}^{(k)} \mathbf{B}_{r_i^{(k)}, r_j^{(k)}}^{(k)} \xi_{r_j^{(k)}}^{(k)} = \left(\sqrt{\frac{|L|}{N_L^{(k)}}} \right) \cdot \frac{1}{|L|} \cdot \left(\sqrt{\frac{|L|}{N_L^{(k)}}} \right) = \frac{1}{N_L^{(k)}}.$$

But

$$\left(\mathbf{S}_n^{(k)} \right)_{i,j} = \frac{\mathbf{1}\left(\mathbf{x}_j^{(k)} \in A\left(\mathbf{x}_i^{(k)} \right) \right)}{\sum_{\ell=1}^{n} \mathbf{1}\left(\mathbf{x}_\ell^{(k)} \in A\left(\mathbf{x}_i^{(k)} \right) \right)} = \frac{1}{N_L^{(k)}} \quad \text{when } \mathbf{x}_j^{(k)} \text{ shares leaf } L \text{ with } \mathbf{x}_i^{(k)}.$$

Thus $\left(\mathbf{b}_n^{(k)} B^{(k)} \mathbf{b}_n^{(k)\top} \right)_{i,j} = \left(\mathbf{S}_n^{(k)} \right)_{i,j}$ in all cases.

The proof for the structure vector follows analogously. □

D.2 Computing the Kernel Vector in Bin Space

We are interested in computing the expectations

$$\mathbf{k}_n^{(k)}(\mathbf{x})^{\cdot} = \mathbb{E}\left[\mathbf{b}^{(k)}(\mathbf{x})^\top \mathbf{D}^{(k)} \mathbf{b}_n^{(k)\top} \right] \in \mathbb{R}^{1 \times n}, \quad \mathbf{K}_n^{(k)} := \mathbb{E}[\mathbf{S}_n^{(k)}] = \mathbb{E}\left[\mathbf{b}_n^{(k)} \mathbf{B}^{(k)} \mathbf{b}_n^{(k)\top} \right], \quad \mathcal{K}_{\text{bin}} = \sum_{k=1}^{p} \mathbb{E}\left[\mathbf{b}_n^{(k)} \mathbf{B}^{(k)} \mathbf{b}_n^{(k)\top} \right],$$

and the norm of the weighting vector

$$\mathbf{r}^{(k)}(\mathbf{x})^\top := \mathbb{E}[\mathbf{b}_n^{(k)}\mathbf{D}^{(k)}\mathbf{b}^{(k)}(\mathbf{x})]^\top \mathbf{J}_n[\mathbf{I} + \mathbf{J}_n\mathcal{K}_{\text{bin}}]^{-1},$$

where $\mathbf{J}_n$ is the centering matrix $\mathbf{J}_n = \mathbf{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^\top$.

Structure vectors and matrices.

Lemma D.2 (Norm of the weight vector in bin-space)**.** *Write* $\mathbf{h}^{(k)}(\mathbf{x}) = \mathbb{E}[{\mathbf{D}^{(k)}}^2\mathbf{b}^{(k)}(\mathbf{x})] \in \mathbb{R}^m$, $\mathbf{H} = \sum_{k=1}^p \mathbb{E}\left[\mathbf{D}^{(k)}\mathbf{B}^{(k)}\mathbf{D}^{(k)}\right]$. *The weight vector* $\mathbf{r}_n^{(k)}(\mathbf{x})$ *can be written as:*

$$\mathbf{r}_n^{(k)}(\mathbf{x}) = \left[\mathbf{I}_n + \mathbf{J}_n\mathbf{Z}_n^{(k)}\mathbf{H}{\mathbf{Z}_n^{(k)}}^\top\right]^{-1}\mathbf{J}_n\mathbf{k}_n^{(k)}(\mathbf{x}).$$

Proof. We first turn our attention to the structure matrix. First note that as the expectation is taken over the ensemble, and the bin assignments are the same for every member of the ensemble, we can pull $\mathbf{Z}_n^{(k)}$ out of the expectation. We then have

$$\mathbf{K}_n^{(k)} = \mathbb{E}[\mathbf{S}_n^{(k)}] = \mathbb{E}\left[\mathbf{b}_n^{(k)}\mathbf{B}^{(k)}{\mathbf{b}_n^{(k)}}^\top\right] = \mathbf{Z}_n^{(k)}\mathbb{E}\left[\mathbf{D}^{(k)}\mathbf{B}^{(k)}\mathbf{D}^{(k)}\right]{\mathbf{Z}_n^{(k)}}^\top.$$

The overall kernel matrix then can be written as:

$$\mathcal{K}_{\text{bin}} = \sum_{k=1}^p \mathbb{E}\left[\mathbf{b}_n^{(k)}\mathbf{B}^{(k)}{\mathbf{b}_n^{(k)}}^\top\right] = \mathbf{Z}_n^{(k)}\left(\sum_{k=1}^p \mathbb{E}\left[\mathbf{D}^{(k)}\mathbf{B}^{(k)}\mathbf{D}^{(k)}\right]\right){\mathbf{Z}_n^{(k)}}^\top$$

The weight vector can then be written as

$$\begin{aligned}\mathbf{r}_n^{(k)}(\mathbf{x})^\top &:= \mathbb{E}[\mathbf{b}_n^{(k)}\mathbf{D}^{(k)}\mathbf{b}^{(k)}(\mathbf{x})]^\top\mathbf{J}_n[\mathbf{I} + \mathbf{J}_n\mathcal{K}_{\text{bin}}]^{-1} \\ &= \left(\mathbf{Z}_n^{(k)}\mathbb{E}[{\mathbf{D}^{(k)}}^2\mathbf{b}^{(k)}(\mathbf{x})]\right)^\top\mathbf{J}_n\left[\mathbf{I} + \mathbf{J}_n\mathbf{Z}_n^{(k)}\left(\sum_{k=1}^p \mathbb{E}\left[\mathbf{D}^{(k)}\mathbf{B}^{(k)}\mathbf{D}^{(k)}\right]\right){\mathbf{Z}_n^{(k)}}^\top\right]^{-1}\end{aligned}$$

Now write $\mathbf{h}^{(k)}(\mathbf{x}) = \mathbb{E}[{\mathbf{D}^{(k)}}^2\mathbf{b}^{(k)}(\mathbf{x})] \in \mathbb{R}^m$, $\mathbf{H} = \sum_{k=1}^p \mathbb{E}\left[\mathbf{D}^{(k)}\mathbf{B}^{(k)}\mathbf{D}^{(k)}\right]$, and note that we can express $\mathbf{k}_n^{(k)}(\mathbf{x}) = \mathbf{Z}_n^{(k)}\mathbf{h}^{(k)}(\mathbf{x}) \in \mathbb{R}^n$. The weight vector is then:

$$\mathbf{r}_n^{(k)}(\mathbf{x}) = \left[\mathbf{I}_n + \mathbf{J}_n\mathbf{Z}_n^{(k)}\mathbf{H}{\mathbf{Z}_n^{(k)}}^\top\right]^{-1}\mathbf{J}_n\mathbf{k}_n^{(k)}(\mathbf{x}).$$

□

Bin–space compression (for the norm).

Lemma D.3 (Bin-space norm computation)**.** *Let* $\mathbf{c}^{(k)} := {\mathbf{Z}_n^{(k)}}^\top\mathbf{1}_n \in \mathbb{R}^{m_k}$ *be the bin counts. Also define*

$$\mathbf{M}^{(k)} := \mathbf{H}^{-1} + \operatorname{diag}(\mathbf{c}^{(\mathbf{k})}) - \frac{1}{n}\mathbf{c}^{(k)}{\mathbf{c}^{(k)}}^\top, \qquad \mathbf{z}^{(k)}(\mathbf{x}) := \operatorname{diag}(\mathbf{c}^{(k)})\mathbf{h}^{(k)}(\mathbf{x}) - \frac{{\mathbf{c}^{(k)}}^\top\mathbf{h}^{(k)}(\mathbf{x})}{n}\mathbf{c}^{(k)}.$$

Setting $\mathbf{q}^{(k)}(\mathbf{x}) := \mathbf{h}^{(k)}(\mathbf{x}) - \mathbf{w}^{(k)}(\mathbf{x})$, *where* $\mathbf{w}^{(k)}$ *is the solution to the small* $m_k \times m_k$ *system of linear equations*

$$\mathbf{M}^{(k)}\mathbf{w}^{(k)} = \mathbf{z}^{(k)}(\mathbf{x}),$$

allows us to express

$$\|\mathbf{r}_n^{(k)}(\mathbf{x})\|_2^2 = \mathbf{q}^{(k)}(\mathbf{x})^\top \operatorname{diag}(\mathbf{c}^{(k)})\,\mathbf{q}^{(k)}(\mathbf{x}) - \frac{1}{n}\left({\mathbf{c}^{(k)}}^\top\mathbf{q}^{(k)}(\mathbf{x})\right)^2.$$

Proof. An application of the Woodbury formula ($(I+UCV)^{-1} = I - U\left(C^{-1}+VU\right)^{-1}V$ with $U = \mathbf{J}_n\mathbf{Z}_n^{(k)}, C = \mathbf{H}, V = \mathbf{Z}_n^{(k)^\top}$) yields:

$$\left[\mathbf{I}_n + \mathbf{J}_n\mathbf{Z}_n^{(k)}\mathbf{H}\mathbf{Z}_n^{(k)^\top}\right]^{-1} = \mathbf{I}_n - \mathbf{J}_n\mathbf{Z}_n^{(k)}\underbrace{\left[\mathbf{H}^{-1} + \mathbf{Z}_n^{(k)^\top}\mathbf{J}_n\mathbf{Z}_n^{(k)}\right]^{-1}}_{:=\mathbf{M}^{(k)-1}}\mathbf{Z}_n^{(k)^\top}.$$

Let $\mathbf{c}^{(k)} := \mathbf{Z}_n^{(k)\top}\mathbf{1}_n \in \mathbb{R}^{m_k}$ be the bin counts. Then

$$\mathbf{Z}_n^{(k)\top}\mathbf{J}_n\,\mathbf{Z}_n^{(k)} = \mathbf{Z}_n^{(k)\top}\mathbf{Z}_n^{(k)} - \frac{1}{n}\mathbf{Z}_n^{(k)\top}\mathbf{1}_n\mathbf{1}_n^\top\mathbf{Z}_n^{(k)} = \mathrm{diag}(\mathbf{c}^{(k)}) - \frac{1}{n}\,\mathbf{c}^{(k)}\mathbf{c}^{(k)^\top}.$$

So defining

$$\mathbf{M}^{(k)} := \mathbf{H}^{-1} + \mathrm{diag}(\mathbf{c}^{(k)}) - \frac{1}{n}\,\mathbf{c}^{(k)}\mathbf{c}^{(k)^\top},$$

we can express

$$\mathbf{r}_n^{(k)}(\mathbf{x}) = \left[\mathbf{I}_n - \mathbf{J}_n\mathbf{Z}_n^{(k)}\mathbf{M}^{(k)^{-1}}\mathbf{Z}_n^{(k)^\top}\right]\mathbf{J}_n\mathbf{k}_n^{(k)}(\mathbf{x}) = \mathbf{J}_n\mathbf{k}_n^{(k)}(\mathbf{x}) - \mathbf{J}_n\mathbf{Z}_n^{(k)}\mathbf{M}^{(k)^{-1}}\mathbf{Z}_n^{(k)^\top}\mathbf{J}_n\mathbf{k}_n^{(k)}(\mathbf{x}).$$

It now falls to us to note that the quantity on the very right of the above expression can be expressed as:

$$\begin{aligned}
\mathbf{z}^{(k)}(\mathbf{x}) &:= \mathbf{Z}_n^{(k)^\top}\mathbf{J}_n\mathbf{k}_n^{(k)}(\mathbf{x}) \\
&= \mathbf{Z}_n^{(k)^\top}\mathbf{J}_n\mathbf{Z}_n^{(k)}\mathbf{h}^{(k)}(\mathbf{x}) \\
&= \left[\mathrm{diag}(\mathbf{c}^{(k)}) - \frac{1}{n}\,\mathbf{c}^{(k)}\mathbf{c}^{(k)^\top}\right]\mathbf{h}^{(k)}(\mathbf{x}) \\
&= \mathrm{diag}(\mathbf{c}^{(k)})\mathbf{h}^{(k)}(\mathbf{x}) - \frac{\mathbf{c}^{(k)^\top}\mathbf{h}^{(k)}(\mathbf{x})}{n}\mathbf{c}^{(k)}.
\end{aligned}$$

To recall, we now have, for

$$\mathbf{M}^{(k)} := \mathbf{H}^{-1} + \mathrm{diag}(\mathbf{c}^{(\mathbf{k})}) - \frac{1}{n}\,\mathbf{c}^{(k)}\mathbf{c}^{(k)^\top}, \qquad \mathbf{z}^{(k)}(\mathbf{x}) := \mathrm{diag}(\mathbf{c}^{(k)})\mathbf{h}^{(k)}(\mathbf{x}) - \frac{\mathbf{c}^{(k)^\top}\mathbf{h}^{(k)}(\mathbf{x})}{n}\mathbf{c}^{(k)},$$

we can express

$$\mathbf{r}_n^{(k)}(\mathbf{x}) = \mathbf{J}_n\mathbf{Z}_n^{(k)}\mathbf{h}^{(k)}(\mathbf{x}) - \mathbf{J}_n\mathbf{Z}_n^{(k)}\mathbf{M}^{(k)^{-1}}\mathbf{z}^{(k)}(\mathbf{x}).$$

Now, we can solve the small $m_k \times m_k$ system of linear equations for $\mathbf{w}^{(k)}$ given by

$$\mathbf{M}^{(k)}\mathbf{w}^{(k)} = \mathbf{z}^{(k)}(\mathbf{x}),$$

to obtain the solution $\mathbf{w}^{(k)} := \mathbf{M}^{(k)^{-1}}\mathbf{z}^{(k)}(\mathbf{x})$. We can then set $\mathbf{q}^{(k)}(\mathbf{x}) := \mathbf{h}^{(k)}(\mathbf{x}) - \mathbf{w}^{(k)}(\mathbf{x})$, and observe that

$$\mathbf{r}_n^{(k)}(\mathbf{x}) = \mathbf{J}_n\,\mathbf{Z}_n^{(k)}\mathbf{q}^{(k)}(\mathbf{x}).$$

To find its norm we simply compute:

$$\begin{aligned}
\|\mathbf{r}_n^{(k)}(\mathbf{x})\|_2^2 &= \mathbf{r}_n^{(k)^\top}\mathbf{r}_n^{(k)} \\
&= \mathbf{q}^{(k)}(\mathbf{x})^\top\mathbf{Z}_n^{(k)^\top}\mathbf{J}_n\,\mathbf{J}_n\,\mathbf{Z}_n^{(k)}\mathbf{q}^{(k)}(\mathbf{x}) \\
&= \mathbf{q}^{(k)}(\mathbf{x})^\top\mathbf{Z}_n^{(k)^\top}\mathbf{J}_n\,\mathbf{Z}_n^{(k)}\mathbf{q}^{(k)}(\mathbf{x}) \\
&= \mathbf{q}^{(k)}(\mathbf{x})^\top\left[\mathrm{diag}(\mathbf{c}^{(k)}) - \frac{1}{n}\,\mathbf{c}^{(k)}\mathbf{c}^{(k)^\top}\right]\mathbf{q}^{(k)}(\mathbf{x}) \\
&= \mathbf{q}^{(k)}(\mathbf{x})^\top\,\mathrm{diag}(\mathbf{c}^{(k)})\,\mathbf{q}^{(k)}(\mathbf{x}) - \frac{1}{n}\,\left(\mathbf{c}^{(k)^\top}\mathbf{q}^{(k)}(\mathbf{x})\right)^2.
\end{aligned}$$

□

The final algorithm.

1. During training, construct and cache diagonal matrix $\mathbf{D}^{(k)} = \operatorname{diag}\left(\xi_1^{(k)}, \ldots, \xi_{m_k}^{(k)}\right)$, with

$$\xi_r^{(k)} := \sqrt{\frac{\left|\mathcal{L}^{(k)}(r)\right|}{N_{\mathcal{L}(r)}^{(k)}}} = \frac{\sum_{l=1}^{m_k} \mathbf{1}\left(B_l^{(k)} \in A\left(B_r^{(k)}\right)\right)}{\sum_{i=1}^{n} \mathbf{1}\left(\mathbf{x}_i^{(k)} \in A\left(B_r^{(k)}\right)\right)} = \sqrt{\frac{\text{Number of bins in leaf containing bin } r}{\text{Number of training samples in leaf containing bin } r}}.$$

2. During training, construct and cache bin-level structure matrix and transform

$$\mathbf{B}_{i,j}^{(k)} = \left[\frac{\mathbf{1}\left(B_j^{(k)} \in A\left(B_i^{(k)}\right)\right)}{\sum_{l=1}^{m_k} \mathbf{1}\left(B_l^{(k)} \in A\left(B_i^{(k)}\right)\right)}\right]_{m_k \times m_k}, \quad \mathbf{b}_n^{(k)} = \mathbf{Z}_n^{(k)} {\mathbf{D}^{(k)}}^2.$$

3. During training, construct and cache an estimate of $\mathbf{H} = \sum_{k=1}^{p} \mathbb{E}\left[\mathbf{D}^{(k)}\mathbf{B}^{(k)}\mathbf{D}^{(k)}\right]$ over the ensemble.

4. During training, construct and cache bin-counts $\mathbf{c}^{(k)} := {\mathbf{Z}_n^{(k)}}^\top \mathbf{1}_n$.

5. Post-training, construct and cache $\mathbf{M}^{(k)} := \mathbf{H}^{-1} + \operatorname{diag}\left(\mathbf{c}^{(k)}\right) - \frac{1}{n}\mathbf{c}^{(k)}{\mathbf{c}^{(k)}}^\top$.

6. On the fly, index into the row/column of $\mathbf{H}$ corresponding the bin that test point $\mathbf{x}$ is in to obtain $\mathbf{h}^{(k)}(\mathbf{x})$.

7. On the fly, compute $\mathbf{z}^{(k)}(\mathbf{x}) := \operatorname{diag}\left(\mathbf{c}^{(k)}\right)\mathbf{h}^{(k)}(\mathbf{x}) - \frac{{\mathbf{c}^{(k)}}^\top \mathbf{h}^{(k)}(\mathbf{x})}{n}\mathbf{c}^{(k)}$.

8. On the fly, solve $\mathbf{M}^{(k)}\mathbf{w}^{(k)} = \mathbf{z}^{(k)}(\mathbf{x})$ for $\mathbf{w}^{(k)}$.

9. On the fly, set $\mathbf{q}^{(k)}(\mathbf{x}) := \mathbf{h}^{(k)}(\mathbf{x}) - \mathbf{w}^{(k)}(\mathbf{x})$.

10. On the fly, compute

$$\|\mathbf{r}_n^{(k)}(\mathbf{x})\|_2^2 = \mathbf{q}^{(k)}(\mathbf{x})^\top \operatorname{diag}(\mathbf{c}^{(k)})\,\mathbf{q}^{(k)}(\mathbf{x}) - \frac{1}{n}\left({\mathbf{c}^{(k)}}^\top \mathbf{q}^{(k)}(\mathbf{x})\right)^2.$$

11. Sum over all features and take the square root: $\|\mathbf{r}_n(\mathbf{x})\|_2 = \sqrt{\sum_{k=1}^{p} \|\mathbf{r}_n^{(k)}(\mathbf{x})\|_2^2}$.

D.3 What we lose by binning.

Binning induces a discretization: all points within the same histogram bin are treated as indistinguishable along that feature. This can (i) introduce an approximation bias at sub-bin scales, and (ii) prevent recovery of within-bin variation even when n is large. However, our bin-space computations do not introduce an additional approximation beyond the estimator we analyze. The reasoning is as follow: the practical EBM implementation we build on already trains *histogram trees*, so the learned model is itself defined on bins, and our inference targets the sampling variability of this histogram-EBM estimator.

If one instead wants inference for a hypothetical *unbinned* (continuous) version of the model, an additional discretization error must be accounted for. A simple bound can be stated under the smoothness assumption in Assumption 2.1. Suppose the bin width along each feature is $O(1/m)$. Then an application of the mean value inequality yields a uniform approximation error $\|\hat{f} - \hat{f}^{(\text{bin})}\|_\infty = O(1/m)$ (assuming the relevant derivatives/densities are bounded), so the accumulated increase in MSE scales as $O(p/m)$. In particular, as the bin resolution increases (e.g., $m \to \infty$ with $m \gg n^{1/3}$), this discretization effect becomes negligible, recovering the same MSE order as in the unbinned idealization.

E Finite Sample Convergence

E.1 Structural assumptions and identities for finite-sample convergence

Throughout the finite-sample convergence analysis we will use a common set of structural assumptions and consequences. Recall $\mathbf{J}_n := \mathbf{I}_n - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$, $\mathbf{K}^{(k)} := \mathbb{E}[\mathbf{S}^{(k)}]$, and $\mathbf{K} := \sum_{k=1}^p \mathbf{K}^{(k)}$. For each feature k, define the additive (centered) subspace

$$\mathcal{H}_k \;:=\; \big\{\, \mathbf{v}\in\mathbb{R}^n :\; \exists\, g:\mathbb{R}\to\mathbb{R} \text{ s.t. } v_i = g(x_i^{(k)}),\; \mathbf{1}^\top\mathbf{v}=0 \,\big\}, \qquad \mathcal{S} \;:=\; \operatorname{span}(\mathcal{H}_1,\dots,\mathcal{H}_p) \subseteq \{\mathbf{1}\}^\perp,$$

and let P_k denote the orthogonal projector onto $\mathcal{H}_k$. One can show that under Assumptions 4.7, the aggregated kernel $\mathbf{K}$ is asymptotically identity.

Lemma E.1 (Centered+ones decomposition and identity on $\mathcal{S}$)**.** *Under Assumptions 4.7, there exist A_n and $\mathbf{E}_n$ with*

$$\mathbf{K} \;=\; A_n \;+\; \frac{p}{n}\,\mathbf{1}\mathbf{1}^\top \;+\; \mathbf{E}_n, \qquad \mathbf{J}_n A_n \mathbf{J}_n = A_n, \qquad \|\,\mathbf{E}_n\,\|_{\mathrm{op}} \;\le\; C\left(\sum_{k=1}^p \varepsilon_{n,k} + \frac{p}{n}\right), \tag{1}$$

and, restricted to $\mathcal{S}$,

$$\|\,A_n - \mathbf{I}\,\|_{\mathcal{S}\to\mathcal{S}} \;\le\; \sum_{k=1}^p \varepsilon_{n,k} \;\xrightarrow[n\to\infty]{}\; 0. \tag{2}$$

Consequently,

$$\mathbf{J}_n\mathbf{K}\mathbf{J}_n \;=\; A_n + \mathbf{J}_n\mathbf{E}_n\mathbf{J}_n, \qquad \big\|\,\mathbf{J}_n\mathbf{K}\mathbf{J}_n - \mathbf{I}\,\big\|_{\mathcal{S}\to\mathcal{S}} \;\le\; C\left(\sum_{k=1}^p \varepsilon_{n,k} + \frac{p}{n}\right) =: \eta_n \;\xrightarrow[n\to\infty]{}\; 0.$$

Proof. For any M, $M = \mathbf{J}_n M \mathbf{J}_n + \frac{1}{n}\mathbf{1}\mathbf{1}^\top + R(M)$ with $\|R(M)\|_{\mathrm{op}} \le c/n$ (the remainder absorbs the $O(1/n)$ column-sum deviation). Apply to $\mathbf{K}^{(k)}$ and sum over k to get (1) with $A_n := \sum_k \mathbf{J}_n\mathbf{K}^{(k)}\mathbf{J}_n$ and $\mathbf{E}_n := \sum_k R(\mathbf{K}^{(k)})$. By Assumption 4.7, $A_n = \sum_k P_k + \sum_k(\mathbf{J}_n\mathbf{K}^{(k)}\mathbf{J}_n - P_k)$, so on $\mathcal{S}$ we have $\sum_k P_k = \mathbf{I}$ (Assumption 4.7) and (2) follows. Left/right multiplying by $\mathbf{J}_n$ kills the rank-one term. □

Lemma E.1 supplies the *identity-on-$\mathcal{S}$* bound

$$\|\,\mathbf{J}_n\mathbf{K}\mathbf{J}_n - \mathbf{I}\,\|_{\mathcal{S}\to\mathcal{S}} \le \eta_n \to 0,$$

which we plug into the mean contraction step for Algorithm 1.

E.2 A Fixed-point Iteration

To investigate the finite sample convergence behavior of the algorithm, we begin with a set of postulated fixed point. For each subensemble's prediction on the training set, assume the existence of a fixed point $\widetilde{\mathbf{y}}_k^*$. Then the Inferable EBM update rule leads to the following fixed point iteration:

$$\widetilde{\mathbf{y}}_k^* = \frac{1}{b+1}\Big((\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}(\mathbf{y}-\widehat{\mathbf{y}}) + b\widetilde{\mathbf{y}}_k^*\Big)$$

Rearranging the equation gives the following identity

$$\widetilde{\mathbf{y}}_k^* = (\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}(\mathbf{y}-\widehat{\mathbf{y}}) \tag{3}$$

Since we do not know $\widehat{\mathbf{y}}$, we will solve for it and plug it back later. Sum the equations over k yields:

$$\widehat{\mathbf{y}} := \widehat{\boldsymbol{\beta}}^* + \sum_{k=1}^p \widetilde{\mathbf{y}}_k^* = \widehat{\boldsymbol{\beta}}^* + (\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)(\sum_{k=1}^p \mathbf{S}^{(k)})(\mathbf{y}-\widehat{\mathbf{y}})$$

And by the design of identifiability constraint, at fixed points, all residuals should have mean zero:

$$\frac{1}{n}\mathbf{1}^\top(\mathbf{y}-\widehat{\mathbf{y}})=0\to\frac{1}{n}\mathbf{1}^\top(\mathbf{y}-\widehat{\boldsymbol{\beta}}^*-\sum_{k=1}^{p}\mathbf{f}^{(k)})=0\to\widehat{\boldsymbol{\beta}}^*=\bar{\mathbf{y}}=\frac{1}{n}\mathbf{1}\mathbf{1}^\top\mathbf{y}$$

Plugging this back we have

$$\widehat{\mathbf{y}}=\Big[\mathbf{I}+(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)(\sum_k\mathbf{S}^{(k)})\Big]^{-1}\Big[(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)(\sum_k\mathbf{S}^{(k)})+\frac{1}{n}\mathbf{1}\mathbf{1}^\top\Big]\mathbf{y} \tag{4}$$

Plugging it back to equation 7 produces the fixed-points for each subensemble:

$$\widetilde{\mathbf{y}}_k^*=(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}\Big[\mathbf{I}-\big[\mathbf{I}+(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)(\sum_k\mathbf{S}^{(k)})\big]^{-1}\big[(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)(\sum_k\mathbf{S}^{(k)})+\frac{1}{n}\mathbf{1}\mathbf{1}^\top\big]\Big]\mathbf{y} \tag{5}$$

The intercept term is canceled out following the algebra below.

First, notice that $\Big(\mathbf{I}+(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)(\sum_{k=1}^{p}\mathbf{S}^{(k)})\Big)(\frac{1}{n}\mathbf{1}\mathbf{1}^\top)=(\frac{1}{n}\mathbf{1}\mathbf{1}^\top)$, since the row sum of $\mathbf{S}^{(k)}$ is always one and thus eliminated by the centering operator.

Multiplying both sides by $\Big(\mathbf{I}+(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)(\sum_{k=1}^{p}\mathbf{S}^{(k)})\Big)^{-1}$ implies $\Big(\mathbf{I}+(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)(\sum_{k=1}^{p}\mathbf{S}^{(k)})\Big)^{-1}(\frac{1}{n}\mathbf{1}\mathbf{1}^\top)=(\frac{1}{n}\mathbf{1}\mathbf{1}^\top)$. This, once again, is killed by the composition $(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}$. Hence the intercept would not matter in the final fixed point derivation.

Assuming the invertibility of $\big[\mathbf{I}+(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)(\sum_k\mathbf{S}^{(k)})\big]$, the resolvent identity yields

$$\widetilde{\mathbf{y}}_k^*=(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}\big[\mathbf{I}+(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)(\sum_k\mathbf{S}^{(k)})\big]^{-1}\mathbf{y}:=\mathbf{J}_n\mathbf{S}^{(k)}[\mathbf{I}+\mathbf{J}_n\mathbf{K}]^{-1}\mathbf{y} \tag{6}$$

The corresponding pointwise prediction for $\forall\mathbf{x}\in\mathbb{R}^p$ is

$$\widehat{\mathbf{f}}^{(k)}(\mathbf{x})=\mathbf{s}^{(k)}(\mathbf{x})\mathbf{J}_n[\mathbf{I}+\mathbf{J}_n\mathbf{K}]^{-1}\mathbf{y}$$

E.3 Another Fixed-point Iteration

Here is a fixed point for the leave-one-out algorithm. For each subensemble's prediction on the training set, assume the existence of a fixed point $\widetilde{\mathbf{y}}_k^*$. Then the update rule leads to the following fixed point iteration:

$$\widetilde{\mathbf{y}}_k^*=\frac{1}{b+1}\Big((\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbb{E}\mathbf{S}^{(k)}(\mathbf{y}-\sum_{a\neq k}\widetilde{\mathbf{y}}_a^*)+b\widetilde{\mathbf{y}}_k^*\Big)$$

Rearranging the equation gives the following identity

$$\widetilde{\mathbf{y}}_k^*=(\mathbf{I}-\mathbf{S}^{(k)})^\dagger(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}(\mathbf{y}-\widehat{\mathbf{y}}) \tag{7}$$

Since we do not know $\widehat{\mathbf{y}}$, we will solve for it and plug it back later. Sum the equations over k yields:

$$\widehat{\mathbf{y}}:=\sum_{k=1}^{p}\widetilde{\mathbf{y}}_k^*=\sum_{k=1}^{p}(\mathbf{I}-\mathbf{S}^{(k)})^{-1}(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}(\mathbf{y}-\widehat{\mathbf{y}})$$

which represents our ensemble prediction as a kernel ridge regression, with the group-level kernel being a sum of feature-wise kernel ridge operator (i.e. $\mathbf{S}_n=\sum_{k=1}^{p}(\mathbf{I}-\mathbf{S}^{(k)})^{-1}\mathbf{S}^{(k)}$)

$$\widehat{\mathbf{y}} = \Big(\mathbf{I} + \sum_{k=1}^{p} (\mathbf{I} - S_n^{(k)})^{-1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}\Big)^{-1}\Big(\sum_{k=1}^{p}(\mathbf{I} - \mathbf{S}^{(k)})^{-1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}\Big)\mathbf{y}$$

Plugging it back to equation 6 produces the fixed-points for each subensemble:

$$\begin{aligned}\widetilde{\mathbf{y}}_k^* &= (\mathbf{I} - \mathbf{S}^{(k)})^{-1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top\mathbf{S}^{(k)}\Big[\mathbf{I} - \Big(\mathbf{I} + \sum_{k=1}^{p}(\mathbf{I} - \mathbf{S}^{(k)})^{-1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}\Big)^{-1}\Big(\sum_{k=1}^{p}(\mathbf{I} - \mathbf{S}^{(k)})^{-1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}\Big)\Big]\mathbf{y} \\ &:= (\mathbf{I} - \mathbf{S}^{(k)})^{-1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}^{(k)}R\mathbf{y}\end{aligned}$$

E.4 Contraction

E.4.1 Algorithm 1

For the vanilla Inferable EBM, we make use of the fixed point iteration conjecture and prove that all feature functions do converge to the set of fixed points. Formally, we state the theorem below:

Theorem E.2. *Define* $\widehat{\mathbf{y}}_b := \widehat{\boldsymbol{\beta}}_b + \sum_{a=1}^{p}\widehat{\mathbf{y}}_b^{(a)}, \mathbf{J}_n = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$*.Let the per-feature Boulevard averages of the new algorithm be*

$$\widehat{\mathbf{y}}_{b+1}^{(k)} \;=\; \frac{b}{b+1}\,\widehat{\mathbf{y}}_b^{(k)} \;+\; \frac{\lambda}{b+1}\,\mathbf{J}_n\,\mathbf{S}^{(b+1,k)}\Big(\mathbf{y} - \widehat{\mathbf{y}}_b\Big),$$

With Assumption 4.3 4.7 hold, Let the postulated fixed points be

$$\widetilde{\mathbf{y}}_k^* \;=\; \mathbf{J}_n\,\mathbf{K}^{(k)}\,[\lambda\mathbf{I} + \mathbf{J}_n\,\mathbf{K}]^{-1}\mathbf{y}, \qquad \mathbf{K} \;:=\; \sum_{k=1}^{p}\mathbf{K}^{(k)},$$

and $\widehat{\mathbf{y}}^* \;=\; \widehat{\boldsymbol{\beta}}^*\mathbf{1} + \sum_{k=1}^{p}\widetilde{\mathbf{y}}_k^*$*, with* $\widehat{\boldsymbol{\beta}}^* = \bar{\mathbf{y}}$*. Then, almost surely, for each* k*,*

$$\widehat{\mathbf{y}}_b^{(k)} \;\longrightarrow\; \widetilde{\mathbf{y}}^{(k)\,*}, \qquad \widehat{\boldsymbol{\beta}}_b \;\longrightarrow\; \bar{\mathbf{y}} = \tfrac{1}{n}\mathbf{1}^\top\mathbf{y},$$

and hence $\widehat{\mathbf{y}}_b \to \widehat{\mathbf{y}}^*$ *almost surely.*

Proof. Since $\mathbf{S}^{(k)}\mathbf{1} = \mathbf{1}$ and $\mathbf{J}_n\mathbf{1} = 0$, we have $\mathbf{J}_n\mathbf{S}^{(k)} = \mathbf{J}_n\mathbf{S}^{(k)}\mathbf{J}_n$. Thus inserting or omitting $\widehat{\boldsymbol{\beta}}_b\mathbf{1}$ in the residual does not affect updates; we drop the intercept in what follows.

Fix the deterministic fixed points $\{\widetilde{\mathbf{y}}_k^*\}$ above and set $\mathbf{D}_b^{(k)} := \widehat{\mathbf{y}}_b^{(k)} - \widetilde{\mathbf{y}}_k^*$, $\mathbf{D}_b := [(\mathbf{D}_b^{(1)})^\top, \ldots, (\mathbf{D}_b^{(p)})^\top]^\top$.

$$\begin{aligned}\mathbb{E}[\mathbf{D}_{b+1}^{(k)} \mid \mathcal{F}_b] &= \mathbb{E}[\widehat{\mathbf{y}}_{b+1}^{(k)} - \widetilde{\mathbf{y}}_k^* \mid \mathcal{F}_b] \\ &= \mathbb{E}\Big[\frac{b}{b+1}\widehat{\mathbf{y}}_b^{(k)} + \frac{1}{b+1}(\mathbf{I} - \tfrac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}_n^{b+1,(k)}\Big(\mathbf{y} - \Gamma_M\Big(\sum_{a=1}^{p}\widehat{\mathbf{y}}_b^{(a)}\Big)\Big) - \widetilde{\mathbf{y}}_k^* \,\Big|\, \mathcal{F}_b\Big] \\ &= \frac{b}{b+1}\widehat{\mathbf{y}}_b^{(k)} + \frac{1}{b+1}(\mathbf{I} - \tfrac{1}{n}\mathbf{1}\mathbf{1}^\top)\,\mathbb{E}[\mathbf{S}_n^{b+1,(k)} \mid \mathcal{F}_b]\Big(\mathbf{y} - \Gamma_M\Big(\sum_{a=1}^{p}\widehat{\mathbf{y}}_b^{(a)}\Big)\Big) - \widetilde{\mathbf{y}}_k^* \\ &= \frac{b}{b+1}\mathbf{D}_b^{(k)} + \frac{1}{b+1}\Big((\mathbf{I} - \tfrac{1}{n}\mathbf{1}\mathbf{1}^\top)\big(\mathbb{E}[\mathbf{S}_n^{b+1,(k)} \mid \mathcal{F}_b] - \mathbb{E}[\mathbf{S}_n^{(\infty,k)}]\big)\big(\mathbf{y} - \sum_{a=1}^{p}\widetilde{\mathbf{y}}_a^*\big) \\ &\qquad + (\mathbf{I} - \tfrac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbb{E}[\mathbf{S}_n^{b+1,(k)} \mid \mathcal{F}_b]\Big(\sum_{a=1}^{p}\widetilde{\mathbf{y}}_a^* - \Gamma_M\Big(\sum_{a=1}^{p}\widehat{\mathbf{y}}_b^{(a)}\Big)\Big)\Big) \\ &= \frac{b}{b+1}\mathbf{D}_b^{(k)} + \frac{1}{b+1}(\mathbf{I} - \tfrac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbb{E}\mathbf{S}^{(k)}\,\Delta_b,\end{aligned}$$

$$\Delta_b \;:=\; \sum_{a=1}^{p} \widetilde{\mathbf{y}}_a^{*} - \Gamma_M\Big(\sum_{a=1}^{p} \widehat{\mathbf{y}}_b^{(a)}\Big), \qquad \|\Delta_b\| \;\le\; \Big\|\sum_{a=1}^{p} \mathbf{D}_b^{(a)}\Big\|.$$

Now we stack the coordinates. The associated $np \times np$ block operator is

$$\mathbf{L} \;=\; \mathbf{J}_n \otimes \mathbf{S}, \qquad (\mathbf{SD})_k \;:=\; \big(\mathbf{J}_n\, \mathbf{K}^{(k)}\, \mathbf{J}_n\big)\left(\sum_{a=1}^{p} \mathbf{D}^{(a)}\right), \quad k = 1, \ldots, p,$$

so that

$$\mathbb{E}[\mathbf{D}_{b+1} \mid \mathcal{F}_b] \;=\; \Big(\frac{b}{b+1}\mathbf{I}_{np} + \frac{1}{b+1}\mathbf{L}\Big)\mathbf{D}_b.$$

Since $\|\mathbf{J}_n\|_{\mathrm{op}} = 1$, it suffices to bound $\|\mathbf{S}\|_{\mathrm{op}}$.

By Assumption 4.7, for each k we have $J\,\mathbf{K}^{(k)}\,J = P_k + E_k$ with $\|E_k\|_{\mathrm{op}} \le \varepsilon_{n,k} \to 0$, and $P_a P_k = 0$ for $a \neq k$. Fix any centered block vector $\mathbf{D} = (\mathbf{D}^{(1)}, \ldots, \mathbf{D}^{(p)})$ and set $S := \sum_{a=1}^{p} \mathbf{D}^{(a)}$. Then

$$\|\mathbf{SD}\|_2^2 = \sum_{k=1}^{p} \|(P_k + E_k)S\|_2^2 \;\le\; \sum_{k=1}^{p} \big(\|P_k S\|_2 + \|E_k\|\,\|S\|_2\big)^2 \;\le\; \sum_{k=1}^{p} \|P_k S\|_2^2 \;+\; 2\varepsilon_n \|S\|_2 \sum_{k=1}^{p} \|P_k S\|_2 \;+\; p\varepsilon_n^2 \|S\|_2^2,$$

where $\varepsilon_n := \max_k \varepsilon_{n,k}$. By orthogonality of the ranges of the P_k's, $\sum_{k=1}^{p} \|P_k S\|_2^2 = \|\sum_{k=1}^{p} P_k S\|_2^2 = \|S\|_2^2$, and $\sum_{k=1}^{p} \|P_k S\|_2 \le \sqrt{p}\,\|S\|_2$. Hence

$$\|\mathbf{SD}\|_2^2 \;\le\; \big(1 + 2\sqrt{p}\,\varepsilon_n + p\varepsilon_n^2\big)\,\|S\|_2^2 \;\le\; \big(1 + c_p\,\varepsilon_n\big)\,\Big\|\sum_{a=1}^{p} \mathbf{D}^{(a)}\Big\|_2^2,$$

for a constant $c_p = 2\sqrt{p} + p\varepsilon_n$. In particular, for all sufficiently large n, $\|\mathbf{S}\|_{\mathrm{op}} \le 1 - \delta_n$ with $\delta_n := 1 - \sqrt{1 + c_p\,\varepsilon_n} > 0$ and $\delta_n \to 0$. Therefore

$$\|\mathbf{L}\|_{\mathrm{op}} \;=\; \|J\|_{\mathrm{op}}\,\|\mathbf{S}\|_{\mathrm{op}} \;\le\; 1 - \delta_n \;<\; 1,$$

and the mean update is a contraction:

$$\Big\|\mathbb{E}[\mathbf{D}_{b+1} \mid \mathcal{F}_b]\Big\| \;\le\; \frac{b + \|\mathbf{L}\|_{\mathrm{op}}}{b+1}\,\|\mathbf{D}_b\| \;\le\; \Big(1 - \frac{\delta_n}{b+1}\Big)\|\mathbf{D}_b\|.$$

For the deviation condition, if leaf predictions are bounded by M, then $\|\epsilon_{b+1}\| \le \frac{2M\sqrt{p}}{b+1}$, so $\sup_b \|\epsilon_b\| \to 0$ and $\sum_b \mathbb{E}[\|\epsilon_b\|^2] < \infty$. This verifies the stochastic contraction mapping conditions.

At the fixed point $\frac{1}{n}\mathbf{1}^\top \widetilde{\mathbf{y}}^{(k)\,*} = 0$ for all k, so $\frac{1}{n}\mathbf{1}^\top \widehat{\mathbf{y}}^{*} = \widehat{\boldsymbol{\beta}}^{*}$. The residual mean at convergence is 0, hence $\widehat{\boldsymbol{\beta}}^{*} = \bar{\mathbf{y}}$. The online update for $\widehat{\boldsymbol{\beta}}_b$ therefore converges almost surely to $\bar{\mathbf{y}}$. □

Corollary E.3. *The feature-wise fixed point prediction:*

$$\widehat{\mathbf{f}}_A^{(k)}(\mathbf{x}) = \mathbb{E}[\mathbf{s}^{(k)}(\mathbf{x})]^\top \mathbf{J}_n[\mathbf{I} + \mathbf{J}_n\mathbf{K}]^{-1}\mathbf{y}$$

and the ensemble prediction

$$\widehat{\mathbf{f}}_A(\mathbf{x}) := \bar{\mathbf{y}} + \sum_{k=1}^{p} \mathbb{E}[\mathbf{s}^{(k)}(\mathbf{x})]^\top \mathbf{J}_n[\mathbf{I} + \mathbf{J}_n\mathbf{K}]^{-1}\mathbf{y}$$

E.4.2 Algorithm 2

Under stated assumptions, we can prove that Algorithm 1's prediction converge almost surely to a postulated fixed point. Formally, we make use of Theorem G.1, called a stochastic contraction mapping theorem, to demonstrate the difference between the current prediction and the fixed point is a stochastic contraction mapping.

Theorem E.4. *Denote the prediction using feature k at boosting round b as $\widehat{\mathbf{y}}_b^{(k)} = \frac{1}{b}\sum_{s=1}^{b} t^{(s,k)}(\mathbf{x}^{(k)})$. Define the aggregated kernel $\mathcal{K} := \sum_{k=1}^{p}(\mathbf{I} - \mathbb{E}\mathbf{S}^{(k)})^{-1}\mathbf{J}_n\mathbb{E}\mathbf{S}^{(k)}$. Suppose for each feature, there exists a fixed point:*

$$\widetilde{\mathbf{y}}_k^* = (\mathbf{I} - \mathbb{E}\mathbf{S}^{(k)})^{-1}\mathbf{J}_n\mathbb{E}\mathbf{S}^{(k)}\Big[\mathbf{I} - \Big(I + \mathcal{K}\Big)^{-1}\mathcal{K}\Big]\mathbf{y}$$

which implies a global fixed point

$$\widehat{\mathbf{y}}^* = \widehat{\boldsymbol{\beta}}^* + \mathcal{K}\Big[\mathbf{I} - \Big(\mathbf{I} + \mathcal{K}\Big)^{-1}\mathcal{K}\Big]\mathbf{y}$$

Then $\widehat{\mathbf{y}}_b^{(k)} \overset{a.s.}{\to} \widetilde{\mathbf{y}}_k^*$ and $\widehat{\mathbf{y}}_b \overset{a.s.}{\to} \widehat{\mathbf{y}}^*$ under stated assumptions in 4. In the meantime, $\widehat{\boldsymbol{\beta}}^* \overset{a.s.}{\to} \bar{\mathbf{y}} := \frac{1}{n}\mathbf{1}^\top\mathbf{y}$.

Proof. Denote $\mathbf{D}_b^{(k)} = \hat{\mathbf{y}}_b^{(k)} - \tilde{\mathbf{y}}_k^*$. Define the stacked distance vector $\mathbf{D}_b = [\mathbf{D}_b^{(1)}, \cdots, \mathbf{D}_b^{(p)}]^\top \in \mathbb{R}^{np}$. Showing this random vector is a stochastic contraction mapping suffices to establish coordinate-wise convergence.

For each coordinate, the conditional mean satisfies

$$\begin{aligned}
\mathbb{E}[\mathbf{D}_{b+1}^{(k)} \mid \mathcal{F}_b] &= \mathbb{E}[\hat{\mathbf{y}}_{b+1}^{(k)} - \tilde{\mathbf{y}}_k^* \mid \mathcal{F}_b] \\
&= \mathbb{E}\Big[\frac{b}{b+1}\hat{\mathbf{y}}_b^{(k)} + \frac{1}{b+1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbf{S}_n^{b+1,(k)}(\mathbf{y} - \Gamma_M\Big(\sum_{a\neq k}\hat{\mathbf{y}}_b^{(a)}\Big)) - \tilde{\mathbf{y}}_k^* \mid \mathcal{F}_b\Big] \\
&= \frac{b}{b+1}\hat{\mathbf{y}}_b^{(k)} + \frac{1}{b+1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)E[\mathbf{S}_n^{b+1,(k)} \mid \mathcal{F}_b](\mathbf{y} - \Gamma_M\Big(\sum_{a\neq k}\hat{\mathbf{y}}_b^{(a)}\Big)) - \tilde{\mathbf{y}}_k^* \\
&= \frac{b}{b+1}\mathbf{D}_b^{(k)} + \frac{1}{b+1}\Big((\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbb{E}[\mathbf{S}_n^{b+1,(k)} \mid \mathcal{F}_b](\mathbf{y} - \Gamma_M\Big(\sum_{a\neq k}\hat{\mathbf{y}}_b^{(a)}\Big)) - \tilde{\mathbf{y}}_k^*\Big) \\
&= \frac{b}{b+1}\mathbf{D}_b^{(k)} + \frac{1}{b+1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\big(\mathbb{E}[\mathbf{S}_n^{(b,k)} \mid \mathcal{F}_b] - \mathbb{E}[\mathbf{S}_n^{(\infty,k)} \mid \mathcal{F}_b]\big)(\mathbf{y} - \sum_{a\neq k}\tilde{\mathbf{y}}_a^*) \\
&+ \frac{1}{b+1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbb{E}[\mathbf{S}_n^{(b,k)} \mid \mathcal{F}_b](\sum_{a\neq k}\tilde{\mathbf{y}}_a^* - \Gamma_M\Big(\sum_{a\neq k}\hat{\mathbf{y}}_b^{(a)}\Big)) \\
&= \frac{b}{b+1}\mathbf{D}_b^{(k)} + \frac{1}{b+1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbb{E}[\mathbf{S}_n^{(b,k)} \mid \mathcal{F}_b](\sum_{a\neq k}\tilde{\mathbf{y}}_a^* - \Gamma_M\Big(\sum_{a\neq k}\hat{\mathbf{y}}_b^{(a)}\Big)) \\
&:= \frac{b}{b+1}\mathbf{D}_b^{(k)} + \frac{1}{b+1}(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\mathbb{E}[\mathbf{S}_n^{(b,k)} \mid \mathcal{F}_b]\Delta_b^{(-k)}
\end{aligned}$$

where $\Big\|\Delta_b^{(-k)}\Big\| \leq \Big\|\sum_{a\neq k}\mathbf{D}_b^{(a)}\Big\|$. Without truncation, stacking across all features yields

$$\mathbb{E}[\mathbf{D}_{b+1} \mid \mathcal{F}_b] = \left(\frac{b}{b+1}\mathbf{I}_{np} + \frac{1}{b+1}\mathbf{L}\right)\mathbf{D}_b,$$

where $\mathbf{L}$ is the $np \times np$ block matrix with diagonal blocks equal to 0 and off–diagonal blocks given by $\mathbb{E}\mathbf{S}^{(k)}$, multiplied by the averaging operator $(I - \frac{1}{n^2}\mathbf{1}\mathbf{1}^\top)$.

The Partition Distinctness assumption implies that each off–diagonal block of $\mathbf{L}$ has operator norm at most c_3, so by a Schur–type bound

$$\|\mathbf{L}\| \;\leq\; (p-1)c_3 < 1$$

Consequently the stacked update satisfies

$$\|\mathbb{E}[\mathbf{D}_{b+1} \mid \mathcal{F}_b]\| \leq \frac{b + \|\mathbf{L}\|}{b+1}\|\mathbf{D}_b\|$$

For the deviation condition, if leaf predictions are bounded by M, then $\|\epsilon_{b+1}\| \leq \frac{2M\sqrt{p}}{b+1}$, so $\sup_b \|\epsilon_b\| \to 0$ and $\sum_b \mathbb{E}[\|\epsilon_b\|^2] < \infty$. This verifies the stochastic contraction mapping conditions.

It remains to verify the convergence of the intercept term $\hat{\boldsymbol{\beta}}$. The update rule for the intercept is given by

$$\hat{\boldsymbol{\beta}}_{b+1} := \hat{\boldsymbol{\beta}}_b + \frac{1}{n}\sum_{k=1}^{p}\sum_{i=1}^{n}\hat{\mathbf{f}}_b^{(k)}(x_i)$$

To identify this limit, recall that at the postulated fixed point we have

$$\hat{\mathbf{y}}^* = \hat{\boldsymbol{\beta}}^* + \sum_{k=1}^{p}\tilde{\mathbf{y}}_k^*,$$

with $\frac{1}{n}\mathbf{1}^\top\tilde{\mathbf{y}}_k^* = 0$ for all k. Taking the average over i gives

$$\frac{1}{n}\mathbf{1}^\top\hat{\mathbf{y}}^* = \hat{\boldsymbol{\beta}}^* + \frac{1}{n}\sum_{k=1}^{p}\mathbf{1}^\top\tilde{\mathbf{y}}_k^* = \hat{\boldsymbol{\beta}}^*.$$

But $\frac{1}{n}\mathbf{1}^\top\hat{\mathbf{y}}^* = \frac{1}{n}\mathbf{1}^\top\mathbf{y} = \bar{y}$ since the residuals at convergence have mean zero. Therefore the intercept converges almost surely to the global mean of the response,

$$\hat{\boldsymbol{\beta}}_b \xrightarrow{a.s.} \bar{y} = \frac{1}{n}\mathbf{1}^\top\mathbf{y}.$$

□

We drop the subscript n if no ambiguity is caused.

Corollary E.5. *The feature-wise fixed point prediction:*

$$\widehat{\mathbf{f}}_B^{(k)}(\mathbf{x}) = \mathbb{E}[\mathbf{s}^{(k)}(\mathbf{x})]^\top(\mathbf{I} - \mathbb{E}\mathbf{S}^{(k)})^{-1}\mathbf{J}_n\Big[\mathbf{I} - (\mathbf{I}+\mathcal{K})^{-1}\mathcal{K}\Big]\mathbf{y}$$

and the ensemble prediction

$$\widehat{\mathbf{f}}_B(\mathbf{x}) := \bar{\mathbf{y}} + \sum_{k=1}^{p}\mathbb{E}[\mathbf{s}^{(k)}(\mathbf{x})]^\top(\mathbf{I} - \mathbb{E}\mathbf{S}^{(k)})^{-1}\mathbf{J}_n\Big[\mathbf{I} - (\mathbf{I}+\mathcal{K})^{-1}\mathcal{K}\Big]\mathbf{y}$$

Intuitively, after centering each signal vector, we would have a decomposition with additive feature-shaped functionals plus a intercept, which is $\bar{\mathbf{y}}$. If we plot the partial-dependency figure, we would see all predicted signal oscillating around 0. In that case, interpretation is as follow: If we observe no features, the best guess would be the baseline $\widehat{\boldsymbol{\beta}}^* = \bar{\mathbf{y}}$. As we adding more and more features, we are able to recover greater parts of the function landscape.

Empirical verification of Assumption 4.9 (Partition Distinctness). In the proof above we use Assumption 4.9 to control the bound of the operator norm of the cross-feature blocks $P_k = \mathbb{E}[\mathbf{J}_n\mathbf{S}^{(k)}\mathbf{J}_n]$, i.e. $c_3 = \max_{a\neq k}\|P_aP_k\|$ and ensure $(p-1)c_3 < 1$. Table 10 reports empirical estimates of c_3 and the corresponding $(p-1)c_3$ across representative real and synthetic datasets.

Table 10: Empirical verification of Assumption 4.9 across real and synthetic datasets. All experiments use `learning_rate=0.1`, `max_rounds=200`, `subsample_rate=0.7`, `truncation=3.0`.

Dataset	**n**	**p**	$\mathbf{c_3}$	$\mathbf{(p-1)c_3}$
adult	48842	14	5.227×10^{-4}	6.795×10^{-3}
bike_sharing	17379	13	9.513×10^{-5}	1.142×10^{-3}
energy_efficiency	768	8	2.838×10^{-4}	1.987×10^{-3}
breast_cancer	569	30	5.553×10^{-2}	$1.610\times10^{+0}$
california_housing	20640	8	4.434×10^{-5}	3.104×10^{-4}
$\mathbf{y} = 0.9\mathbf{X} + \sqrt{1-0.81}\epsilon, \epsilon\sim\mathcal{N}(0,1)$	5000	10	7.181×10^{-4}	6.463×10^{-3}

Across datasets we observe that the cross-products are typically negligible. There are several potential theories for such observations:

1. While multicollinearity is almost surely present in most datasets, their statistical dependence can be weak across features after centering and preprocessing. This is widely noted in GAM literature: Hastie and Tibshirani (1986) often assumes low dependence so that model specifications do not run into identifiability issues.

2. Even when features are correlated, the resulting feature-wise kernels can still point to different directions in the RKHS, which results in a zero cross-product in expectation. Here is a simple GAM as an example: $\mathbf{y} = \mathbf{f}_1(\mathbf{X}_1) + \mathbf{f}_2(\mathbf{X}_2) + \epsilon$, where $\mathbf{f}_1(\mathbf{x}) = \mathbf{x}, \mathbf{f}_2(\mathbf{x}) = \mathbf{x}^2 + 1$, and $(\mathbf{X}_1, \mathbf{X}_2)$ are jointly Gaussian with non-zero correlation. The covariance $\mathrm{Cov}\left(\mathbf{f}_1(\mathbf{X}_1), \mathbf{f}_2(\mathbf{X}_2)\right)$ can be shown to equal to 0 for any ρ. So unless two covariates are nearly identical in both their distribution and main effects to the signal, the induced smoothing kernel can theoretically be small.

3. Last but not least, Gradient Boosting learns by approximating residuals, which are approximately orthogonal to the previously fitted learners.

F Limiting Distribution

In Section E, we have shown that as we keep boosting, our feature estimators will converge to a stacked kernel-ridge-regression

$$\widehat{\mathbf{y}}^* = \widehat{\boldsymbol{\beta}}^* + \sum_{k=1}^{p} \widehat{\mathbf{f}}^{(k)}(\mathbf{X}) = \bar{\mathbf{y}} + \mathbf{J}_n \mathbf{S}^{(k)} [\mathbf{I} + \mathbf{J}_n \mathcal{K}]^{-1} \mathbf{y}$$

with

$$\mathcal{K} := \sum_{k=1}^{p} \mathbb{E}\mathbf{S}^{(k)}$$

F.1 Preliminaries for Asymptotic Normality

To show asymptotic normality, consider the following decomposition:

$$\frac{\widehat{\mathbf{f}}_n^{(k)}(\mathbf{x}) - \mathbf{f}^{(k)}(\mathbf{x})}{\left\|\mathbf{r}^{(k)}\right\|} = \frac{\widehat{\mathbf{f}}_n^{(k)}(\mathbf{x}) - \langle \mathbf{r}^{(k)}(\mathbf{x}), \mathbf{f}(\mathbf{X}_n) \rangle}{\left\|\mathbf{r}^{(k)}\right\|} + \frac{\langle \mathbf{r}^{(k)}(\mathbf{x}), \mathbf{f}(\mathbf{X}_n) \rangle - \mathbf{f}^{(k)}(\mathbf{x})}{\left\|\mathbf{r}^{(k)}\right\|} \tag{8}$$

The first term can be easily shown by constraining leaf size, which we discuss in Section F.2. The later term is trickier. Generalizing from training set to the whole input space, we expect a convergence that is at least of a weak. Bounded kernel, fortunately, allows us to do so.

F.2 Conditional Asymptotic Normality

We begin analysis by showing asymptotic normality on the training set. For simplicity, we use the subscript $E = \{A, B\}$ to specify the ingredients generated by Algorithm 1 and Algorithm 2 Formally, denote $\mathbf{f}(\mathbf{X}) = [\mathbf{f}(\mathbf{x}_1), \cdots, \mathbf{f}(\mathbf{x}_n)]^\top$. What we want to show is that, given any $\mathbf{x} \in \mathbb{R}^p, \forall k$:

$$\frac{\widehat{\mathbf{f}}_E^{(k)}(\mathbf{x}) - \mathbf{r}_E^{(k)}(\mathbf{x})^\top \mathbf{f}(\mathbf{X})}{\left\|\mathbf{r}_E^{(k)}(\mathbf{x})\right\|} \xrightarrow{d} \mathcal{N}(0, \sigma^2)$$

F.2.1 Subsampling

The key component of this proof is the rate of $\left\|\mathbf{r}_E^{(k)}\right\|_1, \left\|\mathbf{r}_E^{(k)}\right\|, \left\|\mathbf{r}_E^{(k)}\right\|_\infty$. These rates will be built on a result shown in Zhou and Hooker (2019) on the behavior of subsampling rate.

Lemma F.1. *Considering a subsampled regression tree. Assume that each leaf contains no fewer than $n^{\frac{1}{p+2}}$ sample points before subsampling. If we are subsampling at rate at least $\xi = n^{-\frac{1}{p+2}\log n}$, then the expected structure vector's norm follows the rate below:*

$$\left\|\mathbf{k}^{(k)}\right\| = 1 - O\left(\frac{1}{n}\right)$$

For each algorithm, our plan to conditional CLT is as follow: Lemma F.1 specifies a proper subsample rates that enables norm convergence. Lindeberg condition simply asks us to check the truncation level $\frac{\mathbf{r}_i^{(k)}}{\left\|\mathbf{r}^{(k)}\right\|}$. We will first study the norm of $\mathbf{k}^{(k)}$. To bound the ℓ_1 norm, it suffices to imposing conditions on the leaf size (with respect to the Lebesgue measure of data points). To bound the ℓ_2 norm. A simple algebra utilizing the rate of ℓ_1 norm will help with that, too.If this norm is well controlled, notice that in both algorithm, the Boulevard vector $\mathbf{r}^{(k)}$ is simply a sum of rows in the matrix $\mathbf{P}_1 := \mathbf{J}_n[\mathbf{I}+\mathbf{J}_n\mathbf{K}]^{-1}$, $\mathbf{P}_2 := (\mathbf{I}-\mathbf{K}^{(k)})^{-1}(\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top)\Big[\mathbf{I}-(\mathbf{I}+\mathcal{K})^\dagger\mathcal{K}\Big]$ with the weighting vector being $\mathbf{k}^{(k)}$. Hence it suffices to study the eigenvalues of the operator $\mathbf{P}_1, \mathbf{P}_2$, which we discuss below. We will first show that the spectrum of $\mathbf{K}^{(k)}$ is strictly bounded way from 1.

F.2.2 Expected Feature Tree Kernels

Lemma F.2 (Spectrum of $\mathbf{K}^{(k)}$)**.** *For any feature k, the expected structure matrix $\mathbf{K}^{(k)} = \mathbb{E}[\mathbf{S}^{(k)}]$ is a symmetric operator on $\mathbb{R}^n$ with spectrum contained in $[0,1]$. It always holds that $\mathbf{K}^{(k)}\mathbf{1} = \mathbf{1}$, so 1 is an eigenvalue with eigenvector $\mathbf{1}$. Restricting to the mean–zero subspace $\mathsf{H}_0 = \{v : \mathbf{1}^\top v = 0\}$, the supremum of the spectrum is bounded by*

$$\sup\{\lambda : \lambda \in \sigma(\mathbf{K}^{(k)}|_{\mathsf{H}_0})\} \;=\; \rho_k \;<\; 1,$$

provided the distribution of partitions under $\mathcal{Q}_n^k$ is not almost surely degenerate.

Proof. Fix any unit $v \in \mathsf{H}_0$. Because $\mathbf{S}^{(k)}$ is an orthogonal projector, $v^\top\mathbf{S}^{(k)}v = \|\mathbf{S}^{(k)}v\|^2$, with equality $\|\mathbf{S}^{(k)}v\| = 1$ if and only if $v \in \text{range}(\mathbf{S}^{(k)})$. Hence

$$v^\top\mathbf{K}^{(k)}v \;=\; \mathbb{E}\Big[v^\top\mathbf{S}^{(k)}v\Big] \;=\; \mathbb{E}\Big[\|\mathbf{S}^{(k)}v\|^2\Big] \;\le\; 1.$$

Assume toward a contradiction that $\lambda_{\max}(\mathbf{K}^{(k)}|_{\mathsf{H}_0}) = 1$. Then there exists a sequence of unit vectors $v_m \in \mathsf{H}_0$ with $v_m^\top\mathbf{K}^{(k)}v_m \to 1$. By the display, $\mathbb{E}[\|\mathbf{S}^{(k)}v_m\|^2] \to 1$. The map $\Pi \mapsto \|\mathbf{S}^{(k)}v_m\|^2$ takes values in $[0,1]$, so if its expectation tends to 1, we must have $\|\mathbf{S}^{(k)}v_m\|^2 \to 1$ in probability (with respect to m). Passing to a subsequence if necessary and using compactness of the unit sphere, take $v_m \to v_\star$ with $\|v_\star\| = 1$ and $v_\star \in \mathsf{H}_0$. By lower semicontinuity and the projector property, $\|\mathbf{S}^{(k)}v_\star\| = 1$ almost surely, hence $\mathbf{S}^{(k)}v_\star = v_\star$ almost surely. Therefore $v_\star \in \text{range}(\mathbf{S}^{(k)})$ for almost every draw of $\mathbf{S}^{(k)}$.

If the distribution of $\mathbf{S}^{(k)}$ charges at least two distinct partitions with positive probability, their ranges (as subspaces of $\mathbb{R}^n$) intersect on H_0 only in $\{0\}$ unless $v_\star$ is proportional to $\mathbf{1}$, which it is not. Thus $v_\star$ cannot lie in the range of both with positive probability, a contradiction. Hence $\lambda_{\max}(\mathbf{K}^{(k)}|_{\mathsf{H}_0}) < 1$. □

Lemma F.2 gives a strict bound on the eigenvalues of the expected kernel, which we can utilize to stabilize the behavior of the aggregated kernel $\mathbf{K}$ and $\mathcal{K}$, which is described in the following sections.

F.2.3 Operator $\mathbf{P}_1$ for Algorithm 1

For $\mathbf{P}_1$, the operator norm is bounded naturally by definition.

Corollary F.3 (Eigenvalue bounds for $\mathbf{P}_1$)**.** *Let $\mathbf{J}_n = \mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$ be the centering projector and*

$$\mathbf{K} \;=\; \sum_{k=1}^{p} \mathbb{E}[\mathbf{S}^{(k)}], \qquad \mathbf{P}_1 \;:=\; \mathbf{J}_n\left(\mathbf{I} + \mathbf{J}_n\,\mathbf{K}\right)^{-1}.$$

Under the assumption that $\sum_{k=1}^{p} \lambda_2(\mathbb{E}\mathbf{S}^{(k)})^2 < 1$, $\|\mathbf{P}_1\|_{op} \leq 1$

F.2.4 Operator $\mathbf{P}_2$ for Algorithm 2

Lemma F.4. *Work on the mean–zero subspace* $\mathsf{H}_0 = \{v \in \mathbb{R}^n : \mathbf{1}^\top v = 0\}$. *The aggregated kernel*

$$\mathcal{K} \;:=\; \sum_{k=1}^{p} (\mathbf{I} - \mathbf{K}^{(k)})^{-1} \mathbf{J}_n \mathbf{K}^{(k)}$$

is symmetric and positive semidefinite. Since $\mathbf{J}_n = \left(\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top\right)$ *is the identity on* H_0, *we may write on* H_0 *simply* $\mathcal{K} = \sum_k (\mathbf{I} - \mathbf{K}^{(k)})^\dagger \mathbf{K}^{(k)}$. *If* $\rho_k := \|\mathbf{K}^{(k)}\|_{\mathsf{H}_0} < 1$, *then on* H_0 *one has the operator sandwich*

$$\sum_{k=1}^{p} \mathbf{K}^{(k)} \;\preceq\; \mathcal{K} \;\preceq\; \sum_{k=1}^{p} \frac{1}{1-\rho_k}\, \mathbf{K}^{(k)},$$

hence $\lambda_{\min}^{+}(\mathcal{K}) \;\geq\; \lambda_{\min}^{+}\Big(\sum_a \mathbf{K}^{(k)}\Big)$ *and* $\lambda_{\max}(\mathcal{K}) \;\leq\; \sum_k \frac{\rho_k}{1-\rho_k}$.

Proof. Each summand $(\mathbf{I}-\mathbf{K}^{(k)})^\dagger \mathbf{K}^{(k)}$ is symmetric because $\mathbf{K}^{(k)}$ is symmetric and $(\mathbf{I}-\mathbf{K}^{(k)})^\dagger$ is a Borel function of $\mathbf{K}^{(k)}$; positivity follows from $x \mapsto (1-x)^\dagger x$ being nonnegative on $[0,1)$. On H_0, the eigenvalues of $(\mathbf{I} - \mathbf{K}^{(k)})^\dagger$ lie in $[1,\, 1/(1-\rho_k)]$, so for any $v \in \mathsf{H}_0$,

$$v^\top \mathbf{K}^{(k)} v \;\leq\; v^\top (\mathbf{I} - \mathbf{K}^{(k)})^\dagger \mathbf{K}^{(k)} v \;\leq\; \frac{1}{1-\rho_k}\, v^\top \mathbf{K}^{(k)} v.$$

Summing over k gives the stated Loewner bounds, from which the eigenvalue bounds follow. □

Combining those bounds above, we are able to claim that our EBM operator $\mathbf{P}$ will not explode in some directions. Formally, we have the following lemma.

Lemma F.5 (Eigenvalues of $\mathbf{P}_2$)**.** *Work on the mean–zero subspace* $\mathsf{H}_0 = \{v \in \mathbb{R}^n : \mathbf{1}^\top v = 0\}$. *Define*

$$\mathbf{P} \;:=\; (\mathbf{I} - \mathbf{K}^{(k)})^\dagger \Big(\mathbf{I} - \tfrac{1}{n}\mathbf{1}\mathbf{1}^\top\Big)\Big[\mathbf{I} - (\mathbf{I}+\mathcal{K})^\dagger \mathcal{K}\Big], \qquad \mathcal{K} \;:=\; \sum_{k=1}^{p} (\mathbf{I} - \mathbf{K}^{(k)})^\dagger \Big(\mathbf{I} - \tfrac{1}{n}\mathbf{1}\mathbf{1}^\top\Big)\mathbf{K}^{(k)}.$$

On H_0 *the centering projector is the identity and* $(\mathbf{I}+\mathcal{K})^\dagger = (\mathbf{I}+\mathcal{K})^{-1}$, *hence*

$$\mathbf{P} \;=\; (\mathbf{I} - \mathbf{K}^{(k)})^\dagger\, (\mathbf{I}+\mathcal{K})^{-1} \qquad \text{on } \mathsf{H}_0.$$

The matrix $\mathbf{P}$ *is symmetric and positive semidefinite on* H_0. *Writing* $\rho_k := \|\mathbf{K}^{(k)}\|_{\mathsf{H}_0} < 1$, *its eigenvalues lie in the interval*

$$\frac{1}{1+\lambda_{\max}(\mathcal{K})} \;\leq\; \lambda(\mathbf{P}) \;\leq\; \frac{1}{(1-\rho_k)\,\big(1+\lambda_{\min}^{+}(\mathcal{K})\big)}.$$

In particular, $\|\mathbf{P}\| \leq \big((1-\rho_k)\,(1+\lambda_{\min}^{+}(\mathcal{K}))\big)^{-1} \leq (1-\rho_k)^{-1}$. *If, in addition,* $\mathbf{K}^{(k)}$ *and* $\mathcal{K}$ *commute on* H_0, *then* $\mathbf{P}$ *is diagonalizable in a common eigenbasis and*

$$\lambda_i(\mathbf{P}) \;=\; \frac{1}{1-\lambda_i(\mathbf{K}^{(k)})} \cdot \frac{1}{1+\lambda_i(\mathcal{K})}, \qquad i = 1, \ldots, n-1.$$

Proof. On H_0 we have $\mathbf{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^\top = \mathbf{I}$ and

$$\mathbf{I} - (\mathbf{I}+\mathcal{K})^\dagger \mathcal{K} \;=\; \mathbf{I} - (\mathbf{I}+\mathcal{K})^{-1}\mathcal{K} \;=\; (\mathbf{I}+\mathcal{K})^{-1},$$

so $\mathbf{P} = (\mathbf{I} - \mathbf{K}^{(k)})^\dagger (\mathbf{I}+\mathcal{K})^{-1}$ on H_0. Since $\mathbf{K}^{(k)}$ and $\mathcal{K}$ are symmetric, by the spectral theorem we may write $\mathbf{K}^{(k)} = U\Lambda U^\top$ with $\Lambda = \mathrm{diag}(\lambda_i)$ and define $(\mathbf{I}-\mathbf{K}^{(k)})^\dagger = U(\mathbf{I}-\Lambda)^\dagger U^\top$, where $(1-\lambda_i)^\dagger = 1/(1-\lambda_i)$ if $\lambda_i \neq 1$ and 0 otherwise. Thus $(\mathbf{I} - \mathbf{K}^{(k)})^\dagger$ is symmetric and positive semidefinite on H_0, with eigenvalues in $[1, (1-\rho_k)^{-1}]$. Likewise, $\mathcal{K}$ is symmetric positive semidefinite, so $(\mathbf{I}+\mathcal{K})^{-1}$ is symmetric positive definite on H_0, with eigenvalues in $[1/(1+\lambda_{\max}(\mathcal{K})),\, 1/(1+\lambda_{\min}^{+}(\mathcal{K}))]$. The product of symmetric positive semidefinite matrices has nonnegative spectrum; moreover, for any matrices A, B the extremal singular values satisfy $s_{\min}(AB) \geq s_{\min}(A)s_{\min}(B)$ and $s_{\max}(AB) \leq s_{\max}(A)s_{\max}(B)$. Applying these to $A = (\mathbf{I}-\mathbf{K}^{(k)})^\dagger$ and $B = (\mathbf{I}+\mathcal{K})^{-1}$ yields the stated eigenvalue interval and the norm bound. If $\mathbf{K}^{(k)}$ and $\mathcal{K}$ commute, they are simultaneously diagonalizable on H_0, and the eigenvalue formula follows by multiplying the diagonal entries. □

F.2.5 Rate of Convergence

Then our analysis proceed with the assistance of the following result, modified from the original Boulevard paper as well.

Lemma F.6 (Rate of Convergence)**.** *Let* $\mathbf{B}_n^{(k)} := \{i : \left\|\mathbf{x}^{(k)} - \mathbf{x}_i^{(k)}\right\| \leq d_n\}$ *be the points within distance* d_n *from test point* $\mathbf{x}$ *split on feature* k*. If* $|\mathbf{B}_n| = \Omega(n \cdot d_n)$ *and* $\inf_{A \in \Pi \in Q_n^{(k)}} \sum_i \mathbf{I}(\mathbf{x}_i \in A) = \Omega(n^{\frac{2}{3}})$*, then* $\left\|\mathbf{k}^{(k)}\right\|_2 = \Theta(n^{-\frac{1}{3}}), \left\|\mathbf{r}_E^{(k)}\right\|_2 = \Theta(n^{-\frac{1}{3}})$

This lemma holds true regardless of the design of our boosting process. The key argument is the minimal leaf size. A sketch of proof is appended below.

Proof. To bound $\left\|\mathbf{k}^{(k)}\right\|$, recall:

$$\mathbf{k}_{nj}^{(k)} = \mathbb{E}[\mathbf{s}_{n,j}^{(k)}(\mathbf{x})] = \mathbb{E}\left[\frac{\mathbf{1}(\mathbf{x}_j \in A)}{\sum_{j=1}^n \mathbf{1}(\mathbf{x}_j \in A)}\right], \mathbf{x} \in A$$

encodes the expected influence of point x_j on a point of interest x among other n points. Then the condition

$$\inf_{A \in \Pi \in Q_n^{(k)}} \sum_{i=1}^n \mathbf{1}(\mathbf{x}_i \in A) \geq \Omega(n^{2/3})$$

implies that $\mathbf{k}_{nj}^{(k)} = O(n^{-2/3})$. By Lemma F.1, $\left\|\mathbf{k}^{(k)}\right\|_1 \leq 1$, then an ℓ_2 norm bound yields

$$\left\|\mathbf{k}^{(k)}\right\| \leq \sqrt{\left\|\mathbf{k}^{(k)}\right\|_1 \left\|\mathbf{k}^{(k)}\right\|_\infty} = O(n^{-\frac{1}{3}}).$$

By assertion $|\mathbf{B}_n| = \Omega(n \cdot d_n)$, there are at most

$$\Omega(n \cdot d_n) = \Omega(n^{2/3})$$

k_{nj} non-zero elements, with the votes $\mathbf{k}_{nj}^{(k)}$ being lower bounded by $\Omega(n^{-2/3})$. Given $\left\|\mathbf{k}^{(k)}\right\|_1 = 1 - O(n^{-1})$,

$$\left\|\mathbf{k}^{(k)}\right\| = \Omega(\sqrt{\left(n^{-2/3}\right)^2 \cdot n^{2/3}}) = \Omega(n^{-1/3}).$$

By corollary F.3 and lemma F.5, we know that $\mathbf{r}_E^{(k)}$ are linear transformations from $\mathbf{k}^{(k)}$ by $\mathbf{P}_1, \mathbf{P}_2$ with bounded eigenvalues. Hence $\left\|\mathbf{r}_E^{(k)}\right\|$ enjoys similar rates. □

Having access to the rate of convergence of $\mathbf{r}^{(k)}$, we can fill in the blank of the weighting vector argument in the Lindeberg conditions. Concretely, it controls any error deviate too much from 0 that contributes to non-normality. We state the theorem below.

Theorem F.7 (Conditional Asymptotic Normality)**.** *For any* $\mathbf{x} \in [0,1]^p$*, write* $\mathbf{f}(\mathbf{X}) = (\mathbf{f}(\mathbf{x}_1), \ldots, \mathbf{f}(\mathbf{x}_n))^\top$*. Then under SVI, Non-Adaptivity, Minimal Leaf Size and assumptions in Lemma 4.11, we have*

$$\frac{\widehat{\mathbf{f}}_E^{(k)}(\mathbf{x}) - \langle \mathbf{r}_E^{(k)}(\mathbf{x}), \mathbf{f}(\mathbf{X})\rangle}{\left\|\mathbf{r}_E^{(k)}\right\|} \xrightarrow{d} \mathcal{N}(0, \sigma^2).$$

Proof. Write

$$\widehat{\mathbf{f}}(x) - {\mathbf{r}^{(k)}}_E^\top \mathbf{f}(\mathbf{X}) = {\mathbf{r}^{(k)}}_E^\top \vec{\epsilon}_n.$$

For simplicity, we drop the subscript E in this proof since both $\mathbf{r}^{(k)}$ maintains the same decaying rate. To obtain a CLT we check the Lindeberg-Feller condition of ${\mathbf{r}^{(k)}}^\top \vec{\epsilon}_n$, i.e. for any $\delta > 0$,

$$\lim_n \frac{1}{\left\|\mathbf{r}^{(k)}\right\|^2 \sigma^2} \sum_{i=1}^n \mathbb{E}\left[(\mathbf{r}_i^{(k)}\epsilon_i)^2 \mathbf{1}(|\mathbf{r}_i^{(k)}\epsilon_i| > \delta \left\|\mathbf{r}^{(k)}\right\| \sigma)\right] = 0.$$

By 4.11 we know that

$$\left\|\mathbf{r}^{(k)}\right\|_\infty \le \left\|\mathbf{k}^{(k)}\right\|_\infty \cdot \|\mathbf{P}\|_1 = \Theta(n^{-2/3}), \quad \left\|\mathbf{r}^{(k)}\right\| = \Theta(n^{-\frac{1}{3}}),$$

Then the maximal deviation's ratio is upper bounded by

$$\frac{\left\|\mathbf{r}^{(k)}\right\|_\infty}{\left\|\mathbf{r}^{(k)}\right\|} = O\left(n^{-\frac{1}{3}}\right),$$

This allows us to check out the Lindeberg-Feller conditions:

$$\begin{aligned}
\sum_{i=1}^n \mathbb{E}\left[(\mathbf{r}_i^{(k)}\epsilon_i)^2 \mathbf{1}(|\mathbf{r}_i^{(k)}\epsilon_i| > \delta \left\|\mathbf{r}^{(k)}\right\| \sigma)\right] &\le \sum_{i=1}^n {\mathbf{r}_i^{(k)}}^2 \sqrt{\mathbb{E}[\epsilon_i^4] \cdot \mathbb{E}[\mathbf{1}(|\mathbf{r}_i^{(k)}\epsilon_i| > \delta \left\|\mathbf{r}^{(k)}\right\| \sigma)]} \\
&\le \sum_{i=1}^n {\mathbf{r}_i^{(k)}}^2 \sqrt{\mathbb{E}[\epsilon_i^4]} \cdot \sqrt{\mathbb{P}\left(|\epsilon_i| \ge \frac{\delta \|\mathbf{r}^{(k)}\| \sigma}{\mathbf{r}_i^{(k)}}\right)} \\
&\le \sum_{i=1}^n {\mathbf{r}_i^{(k)}}^2 \sqrt{\mathbb{E}[\epsilon_i^4]} \sqrt{2\exp\left(-\frac{1}{2\sigma^2} \cdot \left(\frac{\delta \|\mathbf{r}^{(k)}\| \sigma}{\mathbf{r}_i^{(k)}}\right)^2\right)} \\
&\le \left\|\mathbf{r}^{(k)}\right\|^2 \exp\left(-O(n^{2/3})\right) \longrightarrow 0
\end{aligned}$$

where the second last line is the concentration of measure by definition of sub-Gaussian noises. □

F.3 Bias Control and Asymptotic Unbiasedness

In Section F.2 we established a conditional central limit theorem: conditioning on the training covariates $\mathbf{X}$, the noise contribution $\mathbf{r}^{(k)}(\mathbf{x})^\top \vec{\epsilon}$ is asymptotically normal with variance σ^2. To extend this to an unconditional CLT for the estimator, it remains to verify that all *bias terms* arising from the random design vanish at rate $o_p(1)$.

Take Algorithm 1 as example. recall the decomposition

$$\widehat{\mathbf{f}}_A^{(k)}(\mathbf{x}) - \mathbf{f}^{(k)}(\mathbf{x}^{(k)}) = \mathbf{r}_A^{(k)}(\mathbf{x})^\top \boldsymbol{\beta} + \left(\mathbf{r}_A^{(k)}(\mathbf{x})^\top \mathbf{f}^{(k)}(\mathbf{X}^{(k)}) - \frac{\lambda}{1+\lambda}\mathbf{f}^{(k)}(\mathbf{x}^{(k)})\right) + \mathbf{r}_A^{(k)}(\mathbf{x})^\top \sum_{a \ne k} \mathbf{f}^{(a)}(\mathbf{X}^{(a)}) + \mathbf{r}_A^{(k)}(\mathbf{x})^\top \vec{\epsilon}.$$

The last term is the conditional CLT contribution. We now show that the three deterministic terms (baseline, same-feature, cross-feature) all converge to zero at suitable rates.

F.3.1 Baseline Bias

Lemma F.8 (Baseline Bias)**.** *Let $\beta \in \mathbb{R}^n$ denote the baseline vector. With subsampling rate specified in Section F.2.1, $\mathbf{r}_E^{(k)}(\mathbf{x})^\top \beta = O(\frac{1}{n})$ for all $k = 1, \dots, p$.*

Proof. We show this for 2 first. By definition,

$$\mathbf{r}_B^{(k)}(\mathbf{x})^\top = \mathbf{k}^{(k)}(\mathbf{x})^\top (I - \mathbb{E}\mathbf{S}^{(k)})^{-1} \mathbf{J}_n \left[\mathbf{I} - (\mathbf{I} + \mathcal{K})^{-1}\mathcal{K}\right],$$

where $\mathcal{K} = \sum_{k=1}^p (\mathbf{I} - \mathbb{E}\mathbf{S}^{(k)})^{-1} \mathbf{J}_n \mathbb{E}\mathbf{S}^{(k)}$. Since every structure matrix has row sum 1, the centering operator annihilates $\mathbf{1}$, so $\mathcal{K}\beta = 0$ and likewise $(I - \frac{1}{n}\mathbf{1}\mathbf{1}^\top)\beta = 0$. With subsampling, the whole inner product will go to 0 at rate $O(\frac{1}{n})$ since $\left\|\mathbf{k}^{(k)}\right\|_1 = 1 - O(\frac{1}{n})$. Thus the entire product vanishes.

Next we show for Algorithm 1.Row–stochasticity gives $\mathbf{S}^{(k)}\mathbf{1} = \mathbf{1} + O(\frac{1}{n})\mathbf{1}$ for all k, hence $\mathbb{E}[\mathbf{S}^{(k)}]\mathbf{1} = \mathbf{1} + O(\frac{1}{n})\mathbf{1}$. Because $(I + \mathbf{J}_n\mathbf{K})$ maps $\boldsymbol{\beta}$ to itself, its inverse (which exists since $(I + \mathbf{J}_n\mathbf{K})$ is the identity on $\text{span}\{\mathbf{1}\}$ and equals $(I + \mathbf{K}_0)$ on the orthogonal complement, with $\mathbf{K}_0 := \mathbf{J}_n\mathbf{K}\mathbf{J}_n \succeq 0$) maps $\boldsymbol{\beta}$ back to $\boldsymbol{\beta}$: $(I + \mathbf{J}_n\mathbf{K})^{-1}\boldsymbol{\beta} = (1 + O(\frac{1}{n}))\boldsymbol{\beta}$.. This is once again killed by $\mathbf{J}_n$. □

The same feature term bias can be killed by building deeper trees as we collect more and more observations from the underlying data generating process. This can be realized by inheriting assumptions on leaf size from Zhou and Hooker (2019), which we present below:

F.3.2 Same-feature Bias for Algorithm 1

Lemma F.9 (Same–feature Bias for Algorithm 1)**.** *Let $\mathbf{f}^{(k)}$ be L–Lipschitz in its argument, and let $A_n(\mathbf{x})$ denote the (feature-k) leaf containing $\mathbf{x}$. Then*

$$\left\| \mathbf{r}_E^{(k)}(\mathbf{x})^\top \mathbf{f}^{(k)}(\mathbf{X}^{(k)}) - \frac{\lambda}{1+\lambda}\mathbf{f}^{(k)}(\mathbf{x}^{(k)}) \right\| = O(\frac{1}{n})$$

Proof. We drop the subscript for E again. To begin with, by the locality assumption,

$$\left|\mathbf{r}^{(k)}(\mathbf{x})^\top \mathbf{f}^{(k)}(\mathbf{X}^{(k)}) - \mathbf{f}^{(k)}(\mathbf{x}^{(k)})\right| = \left| \sum_i \mathbf{r}_{n,i}^{(k)}(\mathbf{x}) \left(\mathbf{f}^{(k)}(\mathbf{x}_i^{(k)}) - \mathbf{f}^{(k)}(\mathbf{x}^{(k)})\right)\right| < |\sum_i \mathbf{r}_{n,i}^{(k)} - \frac{\lambda}{1+\lambda}|M$$

By Lemma E.1 (a direct consequence of Assumption 4.7), on H_0 we have

$$(I + \mathbf{J}_n\mathbf{K})^{-1} \;=\; \tfrac{1}{2}I + \mathbf{E}_n, \qquad \|\mathbf{E}_n\|_{\mathrm{op}} \le C\,\varepsilon_n,$$

And $\mathbf{r}^{(k)}$ is a linear combinationwith weights in $\mathbf{k}^{(k)}$ of the $\frac{\lambda}{1+\lambda}\mathbf{I}$ matrix. Hence the same feature bias goes to 0 at rate $O(\frac{1}{n})$. □

F.3.3 Same-feature Bias for Algorithm 2

Lemma F.10 (Same–feature Bias for Algorithm 2)**.** *Let $f^{(k)}$ be L–Lipschitz in its argument and let $A_n(\mathbf{x})$ be the (feature–k) leaf containing $\mathbf{x}$. Denote*

$$\mathbf{r}_B^{(k)}(\mathbf{x})^\top := \mathbb{E}[\mathbf{s}^{(k)}(\mathbf{x})]^\top (\mathbf{I} - \mathbf{K}^{(k)})^{-1}\mathbf{J}_n(\mathbf{I} + \mathcal{K})^{-1}, \qquad \mathcal{K} := \sum_{a=1}^{p}(\mathbf{I} - \mathbf{K}^{(a)})^{-1}\mathbf{J}_n\mathbf{K}^{(a)}.$$

Then

$$\left|\mathbf{r}_B^{(k)}(\mathbf{x})^\top \mathbf{f}^{(k)}(\mathbf{X}^{(k)}) - \mathbf{f}^{(k)}(\mathbf{x}^{(k)})\right| = O(\frac{1}{n})$$

Proof sketch. Work on the centered subspace where $\mathbf{J}_n$ acts as identity. By the Neumann series,

$$(\mathbf{I} - \mathbf{K}^{(k)})^{-1} \;=\; \sum_{s\ge 0}(\mathbf{K}^{(k)})^s, \qquad (\mathbf{I} + \mathcal{K})^{-1} \;=\; \sum_{t\ge 0}(-1)^t\mathcal{K}^t,$$

with operator norms uniformly bounded under the assumed spectral gap. Hence

$$\mathbf{P}_k \;=\; \sum_{s\ge 0}(\mathbf{K}^{(k)})^s\,\mathbf{J}_n\,\sum_{t\ge 0}(-1)^t\mathcal{K}^t.$$

Left–multiply by $\mathbf{1}^\top$ to take column sums. Since $\mathbf{1}^\top\mathbf{J}_n = 0$, the $s = 0$ term vanishes. For $s \ge 1$, the near–stochasticity yields $\mathbf{1}^\top(\mathbf{K}^{(k)})^s = \mathbf{1}^\top + O\big(\frac{1}{n}\big)$, and multiplying by the uniformly bounded $\mathbf{J}_n(\mathbf{I}+\mathcal{K})^{-1}$ preserves the $O\big(\frac{s}{n}\big)$ size. Summing the geometric tail over s gives an $O\big(\frac{1}{n}\big)$ remainder. Thus $\mathbf{1}^\top\mathbf{P}_k = \mathbf{1}^\top + O\big(\frac{1}{n}\big)^\top$, i.e., each column sum equals $1 + O\big(\frac{1}{n}\big)$. □

F.3.4 Rate of Leakage for Algorithm 1

Notice that

$$\mathbf{r}^{(k)}(\mathbf{x})^\top \sum_{a\ne k}\mathbf{f}^{(a)}(\mathbf{X}^{(a)}) = \sum_{a\ne k}\mathbf{k}^{(k)}(\mathbf{x})^\top\mathbf{f}^{(a)}(\mathbf{X}^{(a)})$$

The idea is to specify a close-by neighborhood, where the cross-feature leakage decays inside this hyperball fast enough under Lipschitz conditions and slowly expands the border of this ball to eliminate contributions outside this ball at rate $O(\frac{1}{n})$. Concretely, given an index set D_n, for any vector $\mathbf{v} \in \mathbb{R}^n$ we define the notation

$$\mathbf{v}|_{D_n} = \begin{bmatrix} \mathbf{I}(1 \in D_n)v_1 \\ \vdots \\ \mathbf{I}(n \in D_n)v_n \end{bmatrix}$$

which implies a decomposition. $\mathbf{v} = \mathbf{v}|_{D_n} + \mathbf{v}_{D_n^c}$ For a fixed test point $\mathbf{x}$ and feature k, set $l_n := \lceil c_1 \log n \rceil$ with $c_1 > 0$, let d_n be the locality radius from Assumption 4.4, and define

$$D_n := \Big\{ i : |x_i^{(k)} - x^{(k)}| \le l_n \, d_n \Big\}.$$

To account for the diminishing composition of points outside the rim, we first show that on the centered mean subspace H_0, the operator $(\mathbf{I} + \mathcal{K})^{-1}$ has exponentially decaying resolvent along feature k.

The following lemma quantifies the behavior desired.

Lemma F.11 (Rate of Leakage for Algorithm 1)**.** *Under specified assumptions in section 4, the rate of the leakage term is*

$$|\mathbf{r}_A^{(k)}(\mathbf{x})^\top \sum_{a \neq k} \mathbf{f}^{(a)}(\mathbf{X})| = o(1) + O(\frac{1}{n}) = o(1)$$

Proof. We will make use of the radius factor $l_n = \lceil c_1 \log n \rceil$. The leakage term admits the decomposition

$$\mathbf{r}_A^{(k)}(\mathbf{x})^\top \sum_{a \neq k} \mathbf{f}^{(a)}(\mathbf{X}) = \mathbf{r}_A^{(k)}(\mathbf{x})|_{D_n}^\top \sum_{a \neq k} \mathbf{f}^{(a)}(\mathbf{X}) + \mathbf{r}_A^{(k)}(\mathbf{x})|_{D_n^c}^\top \sum_{a \neq k} \mathbf{f}^{(a)}(\mathbf{X})$$

Without loss of generality, we control the behavior of a single dimension a. First bound the near-by term $\mathbf{r}_A^{(k)}(\mathbf{x})|_{D_n}^\top \mathbf{f}^{(a)}(\mathbf{X})$. By assertion, the underlying additive terms are L-Lipschitz, which guarantees deterministically the rate

$$|\mathbf{r}_A^{(k)}(\mathbf{x})|_{D_n}^\top \mathbf{f}^{(a)}(\mathbf{X})| \le L \cdot l_n d_n \left\| \mathbf{r}_A^{(k)}(\mathbf{x})|_{D_n} \right\|_1 = O(d_n \log n) = o(\log^{-1} n) O(\log n) = o(1)$$

Next, we show that the peripheral contribution also goes to 0. Recall by definition

$$|\mathbf{r}_A^{(k)}(\mathbf{x})|_{D_n^c}^\top = \mathbf{k}^{(k)}(\mathbf{x})|_{D_n^c}^\top \mathbf{J}_n [\mathbf{I} + \mathbf{J}_n \mathbf{K}]^{-1} \mathbf{f}^{(a)}(\mathbf{X}^{(a)})$$

By default, we drop the subscript A here for simplicity. Work on the centered space, $\mathbf{J}_n$ acts as the identity. Hence we can bound the $\mathbf{r}_A^{(k)}$ using the following strategy

$$\begin{aligned}
\left\| \mathbf{r}^{(k)}|_{D_n^c} \right\| &= \sum_{|\mathbf{x} - \mathbf{x}_i| > l_n \cdot d_n} |\mathbf{r}_{ni}^{(k)}| \\
&= \sum_{|\mathbf{x} - \mathbf{x}_i| > l_n \cdot d_n} |\sum_j \mathbf{k}_{nj}^{(k)} \Big[\mathbf{J}_n [\mathbf{I} + \mathbf{J}_n \mathbf{K}]^{-1} \Big]_{j,i}| \\
&\le C \times \sum_{|\mathbf{x} - \mathbf{x}_i| > l_n \cdot d_n} |\sum_{|\mathbf{x} - \mathbf{x}_j| \le d_n} \mathbf{k}_{nj}^{(k)} \Big[\mathbf{J}_n [\mathbf{I} + \mathbf{J}_n \mathbf{K}]^{-1} \Big]_{j,i}| \\
&\le C \times \sum_{|\mathbf{x} - \mathbf{x}_j| > l_n \cdot d_n} \mathbf{k}_{nj}^{(k)} \sum_{|\mathbf{x} - \mathbf{x}_i| > l_n \cdot d_n} |\Big[[\mathbf{I} + \mathbf{J}_n \mathbf{K}]^{-1} \Big]_{j,i}| \\
&\le C \times \sum_{|\mathbf{x} - \mathbf{x}_j| > l_n \cdot d_n} \mathbf{k}_{nj}^{(k)} \sum_{|\mathbf{x}_i - \mathbf{x}_j| > (l_n - 1) \cdot d_n} |\Big[[\mathbf{I} + \mathbf{J}_n \mathbf{K}]^{-1} \Big]_{j,i}|
\end{aligned}$$

By Lemma F.3, we know that $\mathbf{I}+\mathbf{J}_n\mathbf{K}$ is positive semidefinite, has bounded inverse and has bounded spectrum from $[1+\lambda_{\min}(\mathcal{K}), 1+\lambda_{\max}(\mathcal{K})]$. Hence by proposition G.2, the bound above proceeds as

$$\begin{aligned}\left\|\mathbf{r}^{(k)}(\mathbf{x})\big|_{D_n^c}\right\|_1 &\le C \sum_{|x_j^{(k)}-x^{(k)}|<d_n} k_{nj}^{(k)}(\mathbf{x}) \sum_{|x_i^{(k)}-x^{(k)}|>\ell_n d_n} C_0\, q_1^{\ell_n-1} \\ &\le O(n q_1^{\ell_n-1}) \\ &= O(q_1^{-1}\cdot n\cdot q_1^{c_1\log n}) \\ &= O(q_1^{-1} n^{1-c_1\log q_1})\end{aligned}$$

where $q_1 = \frac{\sqrt{(1+\lambda_{\max}(\mathbf{K}))/((1+\lambda_{\min}(\mathbf{K}))}-1}{\sqrt{(1+\lambda_{\max}(\mathbf{K}))/((1+\lambda_{\min}(\mathbf{K}))}+1}$. Choose $c_1 = \frac{2}{\log q_1}$ we have this contribution being $O(\frac{1}{n}) = o(1)$. And since $\mathbf{f}^{(k)}(\mathbf{X}^{(k)})$ are bounded for all k below M, hence the peripheral term is also of rate $o(1)$. Adding the inner-rim term preserves the rate $o(1)$.

□

F.3.5 Rate of Leakage for Algorithm 2

Recall

$$\mathbf{r}_B^{(k)}(\mathbf{x})^\top \sum_{a\neq k}\mathbf{f}^{(a)}(\mathbf{X}^{(a)}) = \sum_{a\neq k}\mathbf{k}^{(k)}(\mathbf{x})^\top(\mathbf{I}-\mathbb{E}\mathbf{S}^{(k)})^\dagger(\mathbf{I}-\tfrac{1}{n}\mathbf{1}\mathbf{1}^\top)\big[\mathbf{I}-(\mathbf{I}+\mathcal{K})^\dagger\mathcal{K}\big]\mathbf{f}^{(a)}(\mathbf{X}^{(a)})$$

The strategy to bound the peripheral contribution is similar. This time we choose c_2 such that $c_2 = \frac{2}{\log q_2}$ where $q_2 = \frac{\sqrt{(1+\lambda_{\max}(\mathcal{K}))/((1+\lambda_{\min}(\mathcal{K}))}-1}{\sqrt{(1+\lambda_{\max}(\mathcal{K}))/((1+\lambda_{\min}(\mathcal{K}))}+1}$

Lemma F.12 (Rate of Leakage for Algorithm 2)**.** *Under specified assumptions in section 4, the rate of the leakage term is*

$$|\mathbf{r}_B^{(k)}(\mathbf{x})^\top \sum_{a\neq k}\mathbf{f}^{(a)}(\mathbf{X})| = o(1) + O(\frac{1}{n}) = o(1)$$

Proof. For simplicity, we drop the subscript B here. The inner rim follows the same bound:

$$|\mathbf{r}^{(k)}(\mathbf{x})|_{D_n}^\top \mathbf{f}^{(a)}(\mathbf{X})| \le L\cdot l_n d_n \left\|\mathbf{r}^{(k)}(\mathbf{x})|_{D_n}\right\|_1 = O(d_n\log n) = o(\log^{-1} n)O(\log n) = o(1)$$

And the peripheral contribution

$$|\mathbf{r}^{(k)}(\mathbf{x})|_{D_n^c}^\top = \mathbf{k}^{(k)}(\mathbf{x})|_{D_n^c}^\top(\mathbf{I}-\mathbb{E}\mathbf{S}^{(k)})^{-1}(\mathbf{I}-\tfrac{1}{n}\mathbf{1}\mathbf{1}^\top)\big[\mathbf{I}-(\mathbf{I}+\mathcal{K})^{-1}\mathcal{K}\big]f^{(a)}(\mathbf{X}^{(a)})$$

can be bound in the same manner, since F.2 guarantees $\left\|\mathbf{I}-\mathbf{K}^{(k)}\right\|_{\text{op}} \le \frac{1}{1-\rho_k}$ and $\left\|\mathbf{I}-\frac{1}{n}\mathbf{1}\mathbf{1}^\top\right\|_{\text{op}} \le 1$. Hence the goal is to control the rate of the last linear transformation.

Working on the centered subspace $H_0 = \{\mathbf{v} : \mathbf{v}^\top\mathbf{1} \neq 0\}$. We've shown that on the subspace H_0 it holds true that $[\mathbf{I}-(\mathbf{I}+\mathcal{K})^{-1}\mathcal{K}] = (\mathbf{I}+\mathcal{K})^{-1}$. Combining the bound above, we consider the Neumann expansion of the kernel

$$
\begin{aligned}
\left\|\mathbf{r}^{(k)}|_{D_n^c}\right\| &= \sum_{|\mathbf{x}-\mathbf{x}_i|>l_n\cdot d_n} |\mathbf{r}_{ni}^{(k)}| \\
&= \sum_{|\mathbf{x}-\mathbf{x}_i|>l_n\cdot d_n} |\sum_j \mathbf{k}_{nj}^{(k)}\Big[(\mathbf{I}-\mathbb{E}\mathbf{S}^{(k)})^{-1}(\mathbf{I}-\tfrac{1}{n}\mathbf{1}\mathbf{1}^\top)\big[\mathbf{I}-(\mathbf{I}+\mathcal{K})^{-1}\mathcal{K}\Big]_{j,i}| \\
&\le C\times \sum_{|\mathbf{x}-\mathbf{x}_i|>l_n\cdot d_n} |\sum_{|\mathbf{x}-\mathbf{x}_j|\le d_n} \mathbf{k}_{nj}^{(k)}\Big[\big[\mathbf{I}-(\mathbf{I}+\mathcal{K})^{-1}\mathcal{K}\Big]_{j,i}| \\
&= C\times \sum_{|\mathbf{x}-\mathbf{x}_j|>l_n\cdot d_n} |\sum_{|\mathbf{x}-\mathbf{x}_j|\le d_n} \mathbf{k}_{nj}^{(k)}\Big[(\mathbf{I}+\mathcal{K})^{-1}\Big]_{j,i}| \\
&\le C\times \sum_{|\mathbf{x}-\mathbf{x}_j|>l_n\cdot d_n} \mathbf{k}_{nj}^{(k)} \sum_{|\mathbf{x}-\mathbf{x}_i|>l_n\cdot d_n} |\Big[(\mathbf{I}+\mathcal{K})^{-1}\Big]_{j,i}| \\
&\le C\times \sum_{|\mathbf{x}-\mathbf{x}_j|>l_n\cdot d_n} \mathbf{k}_{nj}^{(k)} \sum_{|\mathbf{x}_i-\mathbf{x}_j|>(l_n-1)\cdot d_n} |\Big[(\mathbf{I}+\mathcal{K})^{-1}\Big]_{j,i}|
\end{aligned}
$$

By Lemma F.5, we know that $\mathbf{I}+\mathcal{K}$ is positive semidefinite, has bounded inverse and has bounded spectrum from $[1+\lambda_{\min}(\mathcal{K}), 1+\lambda_{\max}(\mathcal{K})]$. Hence by proposition G.2, the bound above proceeds as

$$
\begin{aligned}
\left\|\mathbf{r}^{(k)}(\mathbf{x})\big|_{D_n^c}\right\|_1 &\le C \sum_{|x_j^{(k)}-x^{(k)}|\le d_n} k_{nj}^{(k)}(\mathbf{x}) \sum_{|x_i^{(k)}-x^{(k)}|>\ell_n d_n} C_0\, q_2^{\ell_n-1} \\
&\le O(n\, q_2^{\ell_n-1}) \\
&= O(q_2^{-1}\cdot n\cdot q_2^{c_2\log n}) \\
&= O(q_2^{-1} n^{1-c_2\log q_2})
\end{aligned}
$$

where $q_2 = \frac{\sqrt{(1+\lambda_{\max}(\mathcal{K}))/((1+\lambda_{\min}(\mathcal{K}))}-1}{\sqrt{(1+\lambda_{\max}(\mathcal{K}))/((1+\lambda_{\min}(\mathcal{K}))}+1}$. Choose $c_2 = \frac{2}{\log q_2}$ we have this contribution being $O(\frac{1}{n}) = o(1)$, concluding the proof.

□

Collecting Lemmas F.8–F.12 and the we conclude that the random-design bias terms vanish in probability. Together with the conditional CLT for the noise term, Slutsky's theorem implies the full estimator is asymptotically normal (formalized in the next subsection).

F.4 Unconditional Asymptotic Normality

In Section F.3, we have shown that all bias terms vanish to 0 deterministically, given the rate on $\left\|\mathbf{k}^{(k)}\right\|$ and $\left\|\mathbf{r}^{(k)}\right\|$ hold. However the rate does not come for free. To extend them to a random design case, one would need a Kolmogorov extension theorem to define a new probability space and the infinte series $(\mathbf{x}_i, \mathbf{y}_i), i = 1, \cdots, \infty$'s projection onto finite series. See Zhou and Hooker (2019) for detailed construction. It can be shown that conditional asymptotic normality can be extended to the random design case. We will drop the subscript A, B now since they share the same $O_p(\cdot)$ rate established in Section F.3.

Lemma F.13. *For given $\mathbf{x}\in[0,1]^p$, suppose we have random sample $(\mathbf{x}_1,\mathbf{y}_1),\ldots,(\mathbf{x}_n,\mathbf{y}_n)$ for each n. If the cardinality of the tree space $Q_n^{(k)}$ is bounded by $O\Big(n^{-1/3}\exp(n^{2/3-\epsilon_n})\Big)$, for some small $\epsilon_n\to 0$, we can show that*

$$
\frac{\widehat{\mathbf{f}}^{(k)}(\mathbf{x}) - \mathbf{r}_E^{(k)}(\mathbf{x})^\top \mathbf{f}(\mathbf{X})}{\left\|]\mathbf{r}_E^{(k)}(\mathbf{x})\right\|} \xrightarrow{d} \mathcal{N}(0,\sigma_\epsilon^2).
$$

To show the minimal leaf capacity, consider the probability bound

$$\begin{aligned}\mathbb{P}(\exists A \in \Pi \in Q_n^{(k)}; |A| \leq cn^{2/3}, c \in \mathbb{R}^+) &\leq |Q_n^{(k)}| n^{1/3} \exp(-\Theta(n^{-2/3})) \\ &\leq \exp(-\Theta(\epsilon_n)), \epsilon_n > 0\end{aligned}$$

which is absolutely summable and per Borel-Canteli will happen almost surely.

Proof. Conditional asymptotic normality holds under the rate of $|\mathbf{B}_n|$ and the minimal leaf capacity. So it suffices to show that these two statements holds almost surely.

To show the neighborhood statement. We know that the mean of the cardinality is not 0 for sure, so it's safe to assume it scales with the neighborhood volume(length). So $\mathbb{E}[|\mathbf{B}_n|] = nd_n = o(n \log^{-1} n)$. By the multiplicative Chernoff bound,

$$\mathbb{P}(|\mathbf{B}_m| < cn \log^{-1} n) \leq \Big(- \frac{(1-c)^2 n \log^{-1} n}{2} \Big)$$

Bound this infinite sum by the integral below for some $M > 0$

$$\sum_{n=1}^{\infty} \exp\left(-\frac{(1-c)^2 n \log^{-1} n}{2} \right) \leq \int_{n=0}^{\infty} \exp\left(-\frac{(1-c)^2 M}{2} \cdot n \log^{-1} n \right) dn < \infty$$

By Borel-Canteli, the assertion of $|\mathbf{B}_n|$ holds almost surely. For the minimal capacity, this is automatic since any stump will do. □

Combining all results from section F.3, F.2, F.4.

Theorem F.14 (Asymptotic Normality)**.** *Under assumptions on tree space, and all assumptions from section 2, we have*

$$\frac{\widehat{\mathbf{f}}_A^{(k)}(\mathbf{x}) - \frac{\lambda}{1+\lambda}\mathbf{f}^{(k)}(\mathbf{x})}{\left\|\mathbf{r}_A^{(k)}(\mathbf{x})\right\|} \xrightarrow{d} \mathcal{N}(0, \sigma^2), \quad \frac{\widehat{\mathbf{f}}_B^{(k)}(\mathbf{x}) - \mathbf{f}^{(k)}(\mathbf{x})}{\left\|\mathbf{r}_B^{(k)}(\mathbf{x})\right\|} \xrightarrow{d} \mathcal{N}(0, \sigma^2)$$

for all $k = 1, \cdots, p$.

Proof. We show the proof for Algorithm 1 as an example. We break the numerator into three terms.

$$\frac{\widehat{\mathbf{f}}_A^{(k)}(\mathbf{x}) - \frac{\lambda}{1+\lambda}\mathbf{f}^{(k)}(\mathbf{x})}{\left\|\mathbf{r}_A^{(k)}(\mathbf{x})\right\|} = \frac{1}{\left\|\mathbf{r}_A^{(k)}(\mathbf{x})\right\|} (\mathbf{r}_A^{(k)}(\mathbf{x})^\top \boldsymbol{\beta} + \big(\mathbf{r}_A^{(k)}(\mathbf{x})^\top \mathbf{f}^{(k)}(\mathbf{X}^{(k)}) - \frac{\lambda}{1+\lambda}\mathbf{f}^{(k)}(\mathbf{x}^{(k)})\big) + \mathbf{r}_A^{(k)}(\mathbf{x})^\top \sum_{a \neq k} \mathbf{f}^{(a)}(\mathbf{X}^{(a)}) + \mathbf{r}_A^{(k)}(\mathbf{x})^\top \vec{\epsilon})$$

Under restricted tree space assumptions, all terms except the last term are $o_p(1)$, and the last term is conditionally asymptotically normal. By Slutsky's theorem, the sum has a limiting distribution being $\mathcal{N}(0, \sigma^2)$. The proof for Algorithm 2 follows the same arguments. □

The CLT holds trivially for the baseline vector as well, by definition of the fixed point it converge to.

Theorem F.15 (CLT for Baseline Vector)**.** *Let $\widehat{\boldsymbol{\beta}} = \frac{1}{n}\sum_{i=1}^n \mathbf{y}_i$ denote the baseline estimator returned by Algorithm 1. Under Assumption 4.3 4.7, the random vector for Algorithm 1 and Algorithm 2 $\sqrt{n}\,(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta})$ converges in distribution to a Gaussian:*

$$\sqrt{n}\,(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}) \quad \xrightarrow{d} \quad \mathcal{N}(0, \sigma^2),$$

where $\beta = \mathbb{E}[Y]$ is the population intercept.

Proof. By construction of the algorithm, the intercept is always updated by the global mean of residuals. At convergence this coincides with the empirical mean of the responses:

$$\widehat{\boldsymbol{\beta}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{y}_i.$$

Since $\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i$ with $\mathbb{E}[\epsilon_i] = 0$ and $\mathrm{Var}(\epsilon_i) = \sigma^2 < \infty$, we have $\mathbb{E}[\mathbf{y}_i] = \boldsymbol{\beta}$. Because the $\{\mathbf{y}_i\}$ are i.i.d. sub–Gaussian with finite variance, the Lindeberg–Feller central limit theorem applies, yielding

$$\sqrt{n}\,(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}) = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}(\mathbf{y}_i - \mathbb{E}[\mathbf{y}_i]) \quad \xrightarrow{d} \quad \mathcal{N}(0, \sigma^2).$$

□

Proof. By construction of the algorithm, the intercept is always updated by the global mean of residuals. At convergence this coincides with the empirical mean of the responses:

$$\widehat{\boldsymbol{\beta}} = \frac{1}{n}\sum_{i=1}^{n} \mathbf{y}_i.$$

Since $\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i$ with $\mathbb{E}[\epsilon_i] = 0$ and $\mathrm{Var}(\epsilon_i) = \sigma^2 < \infty$, we have $\mathbb{E}[\mathbf{y}_i] = \beta$. Because the $\{\mathbf{y}_i\}$ are i.i.d. sub–Gaussian with finite variance, the Lindeberg–Feller central limit theorem applies, yielding

$$\sqrt{n}\,(\widehat{\boldsymbol{\beta}} - \boldsymbol{\beta}) = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}(\mathbf{y}_i - \mathbb{E}[\mathbf{y}_i]) \quad \xrightarrow{d} \quad \mathcal{N}(0, \sigma^2).$$

□

G Stochastic Contraction Mapping Theorem

Theorem G.1. *Given $\mathbb{R}^d$-valued stochastic process $\{\mathbf{z}_t\}_{t\in\mathbb{N}}$, a sequence of $0 < \lambda_t \leq 1$, define*

$$\mathcal{F}_0 = \emptyset, \mathcal{F}_t = \sigma(\mathbf{z}_1, \ldots, \mathbf{z}_t),$$
$$\epsilon_t = \mathbf{z}_t - \mathbb{E}[\mathbf{z}_t|\mathcal{F}_{t-1}].$$

We call $\mathbf{z}_t$ a stochastic contraction if the following properties hold

1. *Vanishing coefficients*
$$\sum_{t=1}^{\infty}(1-\lambda_t) = \infty, \ \textit{i.e.} \ \prod_{t=1}^{\infty}\lambda_t = 0.$$
2. *Mean contraction*
$$||\mathbb{E}[\mathbf{z}_t|\mathcal{F}_{t-1}]|| \leq \lambda_t \left\|\mathbf{z}_{t-1}\right\|, a.s..$$
3. *Bounded deviation*
$$\sup\|\epsilon_t\| \to 0, \quad \sum_{t=1}^{\infty}\mathbb{E}[\|\epsilon_t\|^2] < \infty.$$

In particular, a multidimensional stochastic contraction exhibits the following behavior

1. *Contraction*
$$\mathbf{z}_t \overset{a.s.}{\to} 0.$$

2. Kolmogorov inequality

$$P\left(\sup_{t\geq T}\|\mathbf{z}_t\| \leq \|\mathbf{z}_T\| + \delta\right) \geq 1 - \frac{4\sqrt{d}\sum_{t=T+1}^{\infty}\mathbb{E}[\epsilon_t^2]}{\min\{\delta^2, \beta^2\}} \tag{9}$$

holds for all $T, \delta > 0$ *s.t.* $\beta = \|\mathbf{z}_T\| + \delta - \sqrt{d}\sup_{t>T}\|\epsilon_t\| > 0$.

Proposition G.2 (Demko–Moss–Smith, General Sparse Exponential Decay)**.** *Let* A *be positive definite, bounded, and boundedly invertible on* $\ell^2(S)$*. For any such* A*, define the support sets*

$$S_n(A) := \bigcup_{k=0}^{n}\{(i,j) : (A^k)_{ij} \neq 0\}, \qquad D_n(A) := (S \times S) \setminus S_n(A).$$

Then there exist constants $C_0 < \infty$ *and* $q \in (0,1)$*, depending only on the spectral bounds of* A*, such that*

$$\sup\{\,|A^{-1}(i,j)| : (i,j) \in D_n(A)\,\} \;\leq\; C_0\, q^{\,n+1}.$$

Here the decay factor q *is given explicitly by*

$$q \;=\; \frac{\sqrt{r}-1}{\sqrt{r}+1}, \qquad r = \frac{b}{a},$$

where $[a,b]$ *is the smallest interval containing the spectrum of* A.

H Additional Figures

H.1 Study of Obesity Data Palechor and la Hoz Manotas (2019)

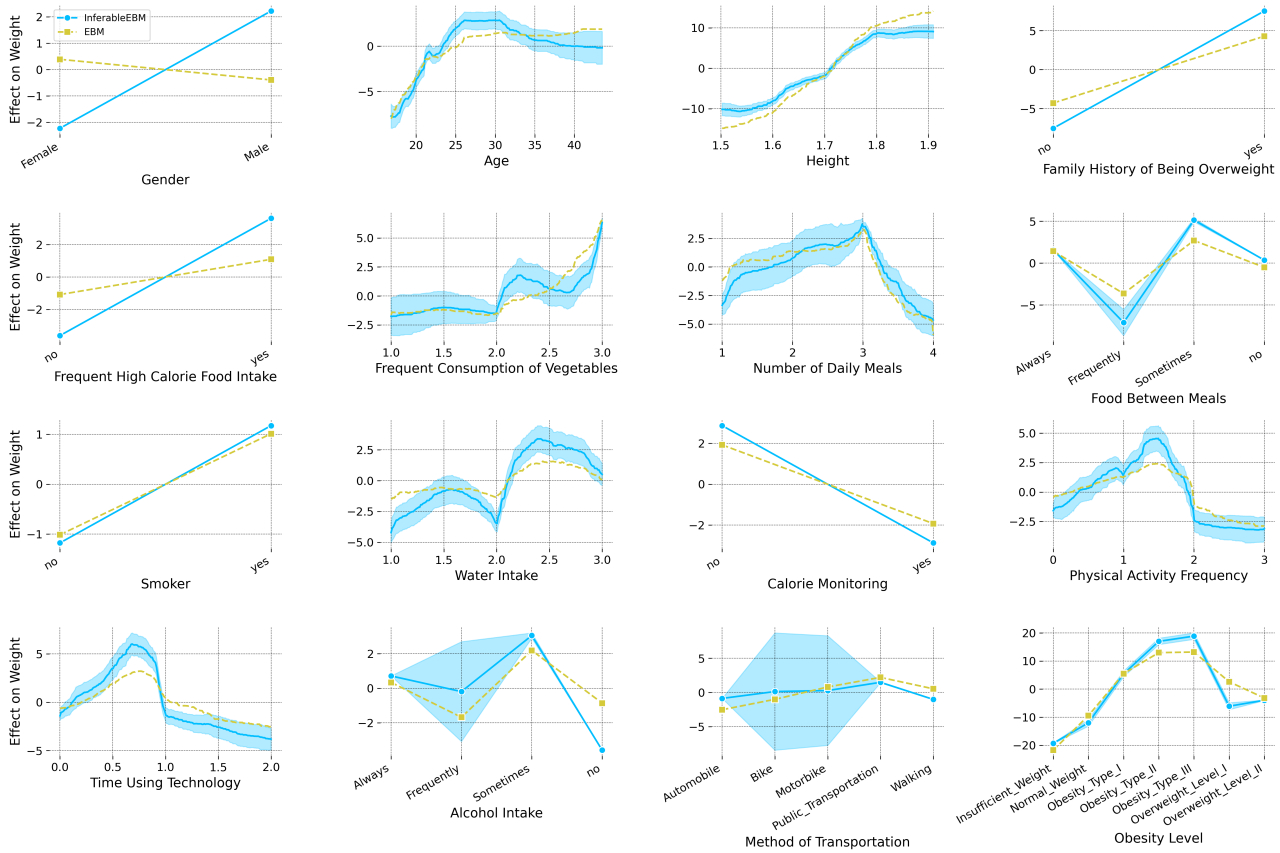

Figure 7: Full set of estimated feature effects for Algorithm 1 and EBM proper Nori et al. (2019), when regressing weight on the given set of covariates. Algorithm 1 by-and-large yields qualitatively similar estimates, with confidence intervals for each term displayed in the shaded area. The terms roughly agree apart from the effect of gender on weight, which we argue should be higher for men than for women.