BAMDT: Bayesian Additive Semi-Multivariate Decision Trees for Nonparametric Regression

Zhao Tang Luo [1] **Huiyan Sang** [1] **Bani Mallick** [1]

Abstract

Bayesian additive regression trees (BART; Chipman et al., 2010) have gained great popularity as a flexible nonparametric function estimation and modeling tool. Nearly all existing BART models rely on decision tree weak learners with axis-parallel univariate split rules to partition the Euclidean feature space into rectangular regions. In practice, however, many regression problems involve features with multivariate structures (e.g., spatial locations) possibly lying in a manifold, where rectangular partitions may fail to respect irregular intrinsic geometry and boundary constraints of the structured feature space. In this paper, we develop a new class of Bayesian additive multivariate decision tree models that combine univariate split rules for handling possibly high dimensional features without known multivariate structures and novel multivariate split rules for features with multivariate structures in each weak learner. The proposed multivariate split rules are built upon stochastic predictive spanning tree bipartition models on reference knots, which are capable of achieving highly flexible nonlinear decision boundaries on manifold feature spaces while enabling efficient dimension reduction computations. We demonstrate the superior performance of the proposed method using simulation data and a Sacramento housing price data set.

1. Introduction

In this paper, we focus on a nonparametric regression problem with response $Y \in \mathbb{R}$ (e.g., housing price) and a number of features. We consider features $\mathbf{s} \in \mathcal{M}$ with *known* multivariate structures, where $\mathcal{M}$ may be a known d-dimensional connected compact Riemannian manifold embedded in a Euclidean space. For instance, $\mathbf{s}$ may represent the coordinates of a location in a spatial domain with a boundary constraint. In addition to $\mathbf{s}$, we also consider features $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^p$ either without multivariate structures or with *unknown* multivariate structures (e.g., square footage and housing age). To be more precise, we model Y as

$$Y = f(\mathbf{s}, \mathbf{x}) + \epsilon, \quad \epsilon \overset{\text{iid}}{\sim} \mathrm{N}(0, \sigma^2), \tag{1}$$

where $f : \mathcal{D} \to \mathbb{R}$ is an unknown function defined on the joint input feature space $\mathcal{D} \subseteq \mathcal{M} \times \mathcal{X}$, and σ^2 is an unknown noise variance. Throughout the paper, we will refer to $\mathbf{s}$ as *structured features*, $\mathbf{x}$ as *unstructured features*, and the regression setting in (1) as *semi-structured regression*.

Semi-structured regression problems are increasingly common in many applications. Examples include spatial regressions and image analysis on complex constrained domains with non-trivial geometries, such as cities with irregular boundaries or interior holes (e.g., lakes and parks), road networks, and brain cortical surfaces, as well as prediction problems on graphs/networks using both graph/network topology and node attributes as predictors. The general model formulation in (1) encompasses many classes of models as special specifications of $f(\mathbf{s}, \mathbf{x})$. Below, we focus on reviewing semi-parametric or nonparametric methods due to their flexibility in function estimation compared to parametric methods.

Related work. Spline smoothings and Gaussian process (GP) regressions are popular choices for nonparametric semi-structured regression problems, and there have been some recent extensions of these methods for data on complex domains (Wood et al., 2008; Scott-Hayward et al., 2014; Niu et al., 2019; Borovitskiy et al., 2020; Dunson et al., 2022). However, these methods often assume globally smooth true functions and thus may not fully adapt to functions with local discontinuities. And the effects of structured features and other unstructured features are usually modeled separately. For example, conventional spatial GP regressions (Gelfand et al., 2010) often assume an additive form, $f(\mathbf{s}, \mathbf{x}) = f(\mathbf{x}) + f(\mathbf{s})$, where $f(\mathbf{x})$ is modeled by a parametric function such as a linear regression and $f(\mathbf{s})$ is a spatial random effect modeled by a GP. However,

[1] Department of Statistics, Texas A&M University, College Station, TX, USA. Correspondence to: Zhao Tang Luo <luozht1015@gmail.com>, Huiyan Sang <huiyan@stat.tamu.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022. Copyright 2022 by the author(s).

parametric models for $f(\mathbf{x})$ may suffer the risk of being misspecified, and more flexible nonparametric models for $f(\mathbf{x})$ have been introduced such as spatial generalized additive models in Nandy et al. (2017) and spatial random forest in Saha et al. (2021). Nonetheless, the additive form of $f(\mathbf{s}, \mathbf{x})$ in these methods could not capture the potential interactions between the effects of $\mathbf{s}$ and $\mathbf{x}$.

Alternatively, ensemble and boosting tree methods such as random forest (Breiman, 2001) and XGBoost (Chen & Guestrin, 2016) have gained great success in nonparametric prediction tasks, owing to their ability to capture both smooth and discontinuous patterns with strong local adaptivity for function estimations. In particular, Bayesian additive regression trees (BART; Chipman et al., 2010) and their variants (see, e.g., Tan & Roy, 2019; He et al., 2019) offer a flexible Bayesian treatment of boosting to probabilistically model and estimate (latent) nonparametric functions in various modeling contexts, while producing uncertainty measures. These models are also appealing for handling a relatively large number of unstructured features; the decision tree weak learner often assumes a simple axis-parallel split rule based on a univariate feature at each decision node, allowing the method to more conveniently adapt to the increasing dimension of features. Nevertheless, the simple axis-parallel univariate split rule comes with a cost: the feature space can only be partitioned into (hyper) rectangular shapes which may not comply with irregular domain constraints and function discontinuity boundaries in the multivariate structured feature space. This limitation has motivated some attempts to relax the axis-parallel decision boundary assumption by considering more flexible decision split rules based on multivariate features (for review, see, e.g., Cañete-Sifuentes et al., 2021; Fan et al., 2021). However, these attempts often make stringent parametric assumptions such as linear or quadratic split rules (Yıldız, 2011; Blaser & Fryzlewicz, 2016), and their estimation procedures are usually not likelihood-based and hence are lacking uncertainty measures.

Most recently, Luo et al. (2021a) proposed a Bayesian additive model built upon random spanning tree partitions for each weak learner. However, the method is applicable to the case with structured features only. It is not straightforward to extend the method in Luo et al. (2021a) to semi-structured regression problems with additional unstructured features since their partition model is not explicitly formulated as decision trees. Moreover, their model is defined only for a finite number of observations. Therefore, although function estimation can be done following Bayesian inference, the out-of-sample prediction of their method is based on a two-step soft nearest neighbor approach due to the lack of a coherent Bayesian model defined on the whole manifold.

Our contributions. In light of these limitations in the current literature, we propose a new Bayesian nonparametric prior for the unknown function f in the context of semi-structured regression, which is built upon an ensemble of novel semi-multivariate decision trees (sMDTs). Specifically, our decision tree recursively splits data into tree nodes starting from a root node. We model each node split rule by a mixture model between a *multivariate split* based on the structured feature, $\mathbf{s}$, and a *univariate split* based on one unstructured feature of $\mathbf{x}$. This allows us to combine their merits for capturing the complex effects of $\mathbf{s}$ and handling possibly high dimensional $\mathbf{x}$, and more importantly, to model the interactions between $\mathbf{s}$ and $\mathbf{x}$. The multivariate split rules are built upon a novel bipartition model via predictive spanning trees. It differs from those of existing MDT methods in that: 1) it allows highly flexible decision boundary shapes while fully respecting the intrinsic geometry of the structured feature space; 2) it is built on any arbitrary subset of the manifold so that both parameter estimation and prediction can be performed under a unified framework; 3) the predictive spanning tree can be constructed on a reduced dimensional reference knot set that is allowed to vary across weak learners, which can be viewed as a multivariate extension of the binning ideas used in boosting methods such as lightGBM (Ke et al., 2017) for reduced computations. An implementation of the proposed model is available at `https://github.com/ztluostat/BAMDT`.

2. Bayesian Semi-Structured Regression with Additive Semi-Multivariate Decision Trees

2.1. Semi-Multivariate Decision Trees

In this subsection, we develop a novel semi-MDT model (sMDT), that involves both multivariate splits for structured features and univariate splits for unstructured features. An sMDT recursively divides the joint input space $\mathcal{D} \subseteq \mathcal{M} \times \mathcal{X}$ into subsets represented by tree nodes. Note that $\mathcal{D}$ may not equal to the product space of $\mathcal{M}$ and $\mathcal{X}$, because $\mathbf{x}$ and $\mathbf{s}$ may not be independent. For a generic set $\mathcal{A}$, we use $\pi_2(\mathcal{A}) = \{\mathcal{A}_1, \mathcal{A}_2\}$ to denote a bipartition of $\mathcal{A}$ that satisfies $\emptyset \subsetneqq \mathcal{A}_1 \subsetneqq \mathcal{A}$ and $\mathcal{A}_2 = \mathcal{A} \backslash \mathcal{A}_1$. Let η be a non-terminal node in an sMDT and η_1 and η_2 be its two offspring nodes. In an sMDT, η either performs a multivariate split using *all* structured features, or a univariate split using one of the unstructured features, to divide the associated subset $\mathcal{D}_\eta \subseteq \mathcal{D}$ into $\pi_2(\mathcal{D}_\eta) = \{\mathcal{D}_{\eta,1}, \mathcal{D}_{\eta,2}\}$ corresponding to η_1 and η_2, respectively. We remark that interaction effects between structured and unstructured features can be naturally captured when the hierarchical splitting of sMDTs involves both $\mathbf{s}$ and $\mathbf{x}$.

Multivariate splits using structured features. A multivariate split divides $\mathcal{D}_\eta$ by bipartitioning $\mathcal{M}_\eta$, the projection of $\mathcal{D}_\eta$ onto $\mathcal{M}$. For a given $\mathcal{M}_\eta$, we utilize a tailored spanning tree based method to be described in Section 2.2

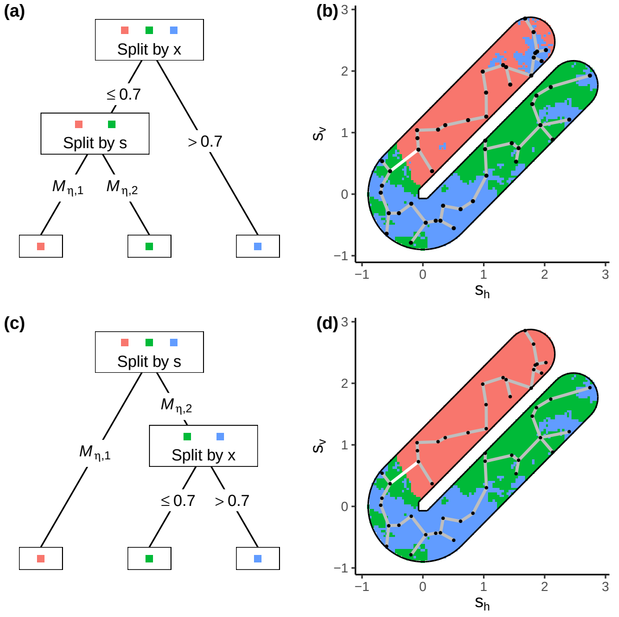

Figure 1. (a, c) Two sMDTs with an input domain $\mathcal{D} \subseteq \mathcal{M} \times \mathbb{R}$, where $\mathcal{M}$ is a two-dimensional U-shape domain. (b, d) Partitions of $\mathcal{D}$ projected onto $\mathcal{M}$ corresponding to the sMDTs in (a) and (c). The spanning tree edges removed in multivariate splits are marked in white. s_h and s_v refer to the horizontal and vertical coordinates of $\mathbf{s}$, respectively.

to split $\mathcal{M}_\eta$ into $\pi_2(\mathcal{M}_\eta) = \{\mathcal{M}_{\eta,1}, \mathcal{M}_{\eta,2}\}$, in a way that the intrinsic geometry of $\mathcal{M}_\eta$ is fully respected. We then set $\mathcal{D}_{\eta,k} = \mathcal{D}_\eta \cap (\mathcal{M}_{\eta,k} \times \mathcal{X})$ for $k = 1, 2$.

Univariate splits using unstructured features. In a univariate split, $\mathcal{D}_\eta$ is divided into

$$\mathcal{D}_{\eta,1} = \{(\mathbf{x}, \mathbf{s}) \in \mathcal{D}_\eta : x_{j(\eta)} \le c_\eta\} \tag{2}$$

$$\mathcal{D}_{\eta,2} = \mathcal{D}_\eta \setminus \mathcal{D}_{\eta,1}, \tag{3}$$

where $x_{j(\eta)}$ is the jth coordinate of $\mathbf{x}$ selected at node η, and c_η is a node-specific cutoff.

Figure 1 shows two examples of sMDTs and the $\mathcal{M}_\eta$ corresponding to their nodes. Note that the univariate splits may create disconnected $\mathcal{M}_\eta$.

2.2. Multivariate Splits via Predictive Spanning Tree Bipartitions

Recall that at a node η of an sMDT, a multivariate split rule requires a bipartition of a possibly disconnected $\mathcal{M}_\eta$. Below, we consider how to induce $\pi_2(\mathcal{M}_\eta)$ from a bipartition of a finite set of reference knots. Let $\mathcal{S}^* = \{\mathbf{s}_1^*, \ldots, \mathbf{s}_t^*\} \subseteq \mathcal{M}$ be a finite set of reference knots on $\mathcal{M}$ which may or may not coincide with the observed structured features. Typical choices of $\mathcal{S}^*$ include grid points covering $\mathcal{M}$ or a random subset of the observed values of $\mathbf{s}$. Let $d_g(\mathbf{s}, \mathcal{B}) := \inf_{\mathbf{t}\in\mathcal{B}} d_g(\mathbf{s}, \mathbf{t})$ be the distance between $\mathbf{s}$ and a non-empty subset $\mathcal{B}$ of $\mathcal{M}$, where d_g is a geodesic distance metric for the manifold $\mathcal{M}$. Given a node-specific bipartition $\pi_2^\eta(\mathcal{S}^*) = \{\mathcal{S}_1^{*,\eta}, \mathcal{S}_2^{*,\eta}\}$, $\pi_2(\mathcal{M}_\eta) = \{\mathcal{M}_{\eta,1}, \mathcal{M}_{\eta,2}\}$ can be obtained by setting:

$$\mathcal{M}_{\eta,1} = \{\mathbf{s} \in \mathcal{M}_\eta : d_g(\mathbf{s}, \mathcal{S}_1^{*,\eta}) \le d_g(\mathbf{s}, \mathcal{S}_2^{*,\eta})\}, \tag{4}$$

$$\mathcal{M}_{\eta,2} = \mathcal{M}_\eta \setminus \mathcal{M}_{\eta,1}. \tag{5}$$

We now construct the bipartition model of $\mathcal{S}^*$ at node η with two considerations in mind. First, since similar structured features tend to have similar effects on Y, it is desired to guarantee local contiguity of $\pi_2^\eta(\mathcal{S}^*)$, in the sense that each local cluster in $\mathcal{S}_k^{*,\eta}$ only contains knots that are close to each other with respect to distance d_g. Second, $\pi_2^\eta(\mathcal{S}^*)$ needs to be a *valid* partition in a way that there is at least one observation $(\mathbf{s}, \mathbf{x}) \in \mathcal{D}_\eta$ whose $\mathbf{s}$ belongs to the resulting $\mathcal{M}_{\eta,k}$ for both $k = 1, 2$.

Spanning tree partition models have recently been proposed as an effective tool to model contiguous partitions of graphs (Li & Sang, 2019; Teixeira et al., 2019; Luo et al., 2021b; Lee et al., 2021). They simplify the complicated combinatorial problem of graph partitions by representing partitions as connected components induced by pruning an edge from a spanning tree of the graph. However, there exist some major challenges that prevent us from directly applying these methods to model the bipartition of $\mathcal{S}^*$. Specifically, there may exist many knots with no nearby observations whose unstructured features are in $\mathcal{M}_\eta$. As a result, removing an arbitrary edge from the spanning tree graph on $\mathcal{S}_\eta^*$ does not necessarily lead to a *valid* $\pi_2^\eta(\mathcal{S}^*)$. In addition, one also needs to take into account the fact that $\mathcal{M}_\eta$ varies with η and can be disconnected due to possible interactions between multivariate splits and univariate splits.

In this paper, we propose a new spanning tree based bipartition model for $\pi_2^\eta(\mathcal{S}^*)$ to overcome these challenges. Specifically, let $\mathcal{G}_T^* = (\mathcal{S}^*, \mathcal{E}^*)$ be an undirected spanning tree graph on the reference knots set $\mathcal{S}^*$ with edges $\mathcal{E}^*$, where each knot is only connected to its near neighbors with respect to d_g so that it represents the topology of the structured feature space. For instance, $\mathcal{G}_T^*$ can be specified as the minimum spanning tree (MST) of a graph $\mathcal{G}^*$ on $\mathcal{S}^*$ using edge lengths under d_g as edge weights. Following Luo et al. (2021a), $\mathcal{G}^*$ can be constructed using constrained Delaunay triangulations (CDTs; Lee & Schachter, 1980) for constrained domains in $\mathbb{R}^2$ or K nearest neighbor (K-NN) graphs with respect to distance d_g for general manifolds. See Appendix A for more discussion on constructing $\mathcal{G}^*$ in practice.

At each decision node η, instead of arbitrarily removing an edge from $\mathcal{E}^*$, we first identify a subset of knots, denoted by $S_\eta^* \subseteq \mathcal{S}^*$, that contains the union of the nearest reference knot of each $\mathbf{s} \in \mathcal{M}_\eta$ under d_g such that $(\mathbf{s}, \mathbf{x})$ is an observation in $\mathcal{D}_\eta$ for some $\mathbf{x}$. We then consider a

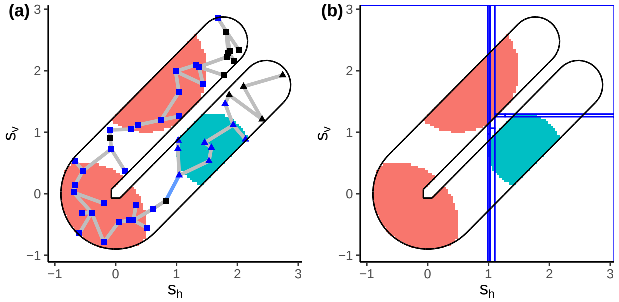

Figure 2. (a) A spanning tree graph $\mathcal{G}_T^*$ on reference knots $\mathcal{S}^*$ and a colored subset $\mathcal{M}_\eta$ of a U-shape domain. The subset $\mathcal{S}_\eta^*$ is marked by blue points. A bipartition $\pi_2^\eta(\mathcal{S}^*)$ after the blue edge is removed from $\mathcal{G}_T^*$ is shown by different point shapes, and the induced multivariate split $\pi_2(\mathcal{M}_\eta)$ is marked by different colors. (b) A univariate decision tree partition that approximates $\pi_2(\mathcal{M}_\eta)$, where blue lines represent decision boundaries.

path in $\mathcal{G}_T^*$ connecting two distinct knots $\mathbf{s}^*$ and $\mathbf{t}^*$ in $\mathcal{S}_\eta^*$, which is unique as $\mathcal{G}_T^*$ is a spanning tree. By removing an edge e^* in the path from $\mathcal{G}_T^*$, we obtain two connected components from the resulting two subgraphs of $\mathcal{G}_T^*$, which naturally defines a bipartition of $\mathcal{S}^*$ at node η by letting $\mathcal{S}_k^{*,\eta}$ ($k = 1, 2$) be the vertices in the kth connected component, and further induces a multivariate split of $\mathcal{M}_\eta$ via (4) and (5). The resulting bipartition, $\pi_2^\eta(\mathcal{S}^*)$, is guaranteed to be valid since by construction and the definition of $\mathcal{S}_\eta^*$, each $\mathcal{M}_{\eta,k}$ ($k = 1, 2$) contains an observation whose nearest knot in $\mathcal{M}$ is either $\mathbf{s}^*$ or $\mathbf{t}^*$. This property motivates a generative prior model for $\pi_2(\mathcal{M}_\eta)$ induced from $\pi_2^\eta(\mathcal{S}^*)$ to be introduced in Section 2.3. Note that the endpoints of e^* may not belong to $\mathcal{S}_\eta^*$ as $\mathcal{M}_\eta$ can be disconnected.

Since the multivariate splits rely on geodesic distance, they are capable to generate flexible shaped partitions of $\mathcal{M}_\eta$ that respect the intrinsic geometry and boundary constraints. Figure 2(a) illustrates an example of a spanning tree bipartition $\pi_2^\eta(\mathcal{S}^*)$ and the induced $\pi_2(\mathcal{M}_\eta)$, where $\mathcal{M}_\eta$ is a disconnected subset of a U-shape domain. Note that a similar partition in Figure 2(b) given by a univariate decision tree has more splits, and the univariate splits do not fully respect the intrinsic geometry of $\mathcal{M}$.

Before introducing the sMDT generating process, we remark that, although both sMDTs and spanning trees $\mathcal{G}_T^*$ are referred to as "trees", they are fundamentally different concepts for different purposes. An sMDT is a binary decision tree defining a partition of $\mathcal{D}$, and its vertices represent subsets of $\mathcal{D}$. On the other hand, $\mathcal{G}_T^*$ encodes an ordering of the multivariate structured knots that facilitates multivariate splits, and its vertices are the reference knots in $\mathcal{M}$.

2.3. A Generative Prior Model for Semi-Multivariate Decision Trees

Similar to the generative process for univariate decision trees (Chipman et al., 1998), an sMDT can be recursively generated in the following manner:

1. Start with a trivial sMDT that only contains a root node representing the full input space $\mathcal{D}$.
2. Split a terminal node η representing $\mathcal{D}_\eta$ with probability $p_{\text{split}}(\eta)$. If η splits, apply one of the following split rules to obtain $\pi_2(\mathcal{D}_\eta)$.
   (a) With probability p_m, perform a *multivariate* split using the structured features $\mathbf{s}$.
   (b) Otherwise, perform a *univariate* split using one of the unstructured features $\mathbf{x}$.
3. If η splits, apply Step 2 to each offspring node of η by setting η as η_1 and η_2, respectively.

To generate a multivariate split, we first partition $\mathcal{M}_\eta$ into $\pi_2(\mathcal{M}_\eta)$ by generating a node-specific bipartition of $\mathcal{S}^*$, $\pi_2^\eta(\mathcal{S}^*)$. Motivated by the property in Section 2.2, we assume the following generative process of $\pi_2(\mathcal{M}_\eta)$:

1. Randomly sample two distinct knots $\mathbf{s}^*$ and $\mathbf{t}^*$ from $\mathcal{S}_\eta^*$.
2. Randomly sample an edge e^* from the unique path in $\mathcal{G}_T^*$ connecting $\mathbf{s}^*$ and $\mathbf{t}^*$.
3. Remove e^* from $\mathcal{G}_T^*$ to obtain $\pi_2^\eta(\mathcal{S}^*)$ and the induced $\pi_2(\mathcal{M}_\eta)$ via (4) and (5).

Then, we let $\mathcal{D}_{\eta,k} = \mathcal{D}_\eta \cap (\mathcal{M}_{\eta,k} \times \mathcal{X})$ be the subset represented by η's offspring η_k, for $k = 1, 2$.

The generating process of splits using unstructured features follows a similar path as in Chipman et al. (1998) and Denison et al. (1998). Specifically, one of the unstructured features $x_{j(\eta)}$ is randomly chosen, and a random cutoff value c_η is uniformly drawn from its candidate set, which typically depends on the feature and training data. Then we set $\mathcal{D}_{\eta,1}$ and $\mathcal{D}_{\eta,2}$ as in (2) and (3).

Probability for splits p_{split}. Following Chipman et al. (1998), we specify $p_{\text{split}}(\eta)$ as

$$p_{\text{split}}(\eta) = \alpha(1 + d_\eta)^{-\beta}, \qquad (6)$$

where d_η is the depth of a node η, and α and β are positive constants. This specification implies that the probability of a node being non-terminal decreases exponentially with its depth and hence implicitly controls the size of an sMDT. We will discuss the choice of α and β in Section 2.4, where we adopt the sMDT generating process as a prior model.

Probability for multivariate splits p_m. This probability controls the proportions of multivariate structured splits among all decision tree nodes. The larger p_m is, the more structured information is used for growing an sMDT. When there is no *a priori* information about the true function, $p_m = d/(d+p)$ is a reasonable default choice.

2.4. A Bayesian Sum-of-multivariate-decision-trees Model

An sMDT, denoted by T, partitions the input space $\mathcal{D}$ into ℓ disjoint subsets $\{\mathcal{D}_1, \dots, \mathcal{D}_\ell\}$ represented by its ℓ terminal nodes. To apply sMDTs to nonparametric regression tasks, given T and $\boldsymbol{\mu} = (\mu_1, \dots, \mu_\ell)$, we define a piecewise constant mapping from $\mathcal{D}$ to $\mathbb{R}$ as

$$g(\mathbf{s}, \mathbf{x} | T, \boldsymbol{\mu}) = \mu_j, \quad \text{if } (\mathbf{s}, \mathbf{x}) \in \mathcal{D}_j.$$

Using g as a weak learner, a Bayesian additive semi-multivariate decision trees (BAMDT) regression model utilizes a summation of piecewise constant functions to approximate the true function f by assuming

$$\mathbb{E}(Y|\mathbf{s}, \mathbf{x}) = \sum_{m=1}^{M} g(\mathbf{s}, \mathbf{x} | T_m, \boldsymbol{\mu}_m),$$

where T_m is an sMDT with ℓ_m terminal nodes, $\boldsymbol{\mu}_m = (\mu_{m1}, \dots, \mu_{m\ell_m})$ are the terminal node specific constants for T_m, and $M \in \mathbb{N}$ is the pre-specified number of weak learners.

Like other additive tree models such as BART (Chipman et al., 2010) and gradient boosting trees (Friedman, 2001), BAMDT is able to adapt to different smoothness levels and/or discontinuities in the true function. The highly flexible sMDT partitions allow BAMDT to more effectively capture irregularly shaped decision boundaries where discontinuities or sharp changes happen, while respecting the intrinsic geometry of the structured feature space $\mathcal{M}$.

The regularization prior model of BAMDT is specified in a similar way as in BART, which admits the form

$$p\left(\{T_m, \boldsymbol{\mu}_m\}_{m=1}^{M}, \sigma^2\right) = \left\{\prod_{m=1}^{M} p(\boldsymbol{\mu}_m | T_m) p(T_m)\right\} p(\sigma^2).$$

The sMDT generating process in Section 2.3 is adopted as a prior for T_m's, that is, we assume *a priori* that each T_m is a sample from the generating process. We recommend choosing the reference set of reduced size compared to the number of observations so that computations can be done more efficiently on a reduced spanning tree graph. Nevertheless, this dimension reduction strategy leads to coarser decision tree boundaries in each weak learner. To increase the diversity of sMDTs in the ensemble, we use different sets of reference knots (and hence different spanning trees) for each T_m, which allows each weak learner to explore and learn a different portion of f so that finer discontinuity boundaries in data might be better recovered from ensembles. Following Chipman et al. (2010), we choose $\alpha = 0.95$ and $\beta = 2$ in (6), which assigns most of the prior probability to small sMDTs with 2 or 3 nodes and penalizes large T_m's. Shallow sMDTs encourage better mixing and faster convergence in Markov chain Monte Carlo.

Conditional on T_m, we place a conjugate Gaussian prior for $\boldsymbol{\mu}_m$:

$$\boldsymbol{\mu}_m | T_m \sim \mathrm{N}_{\ell_m}(\mathbf{0}, \sigma_\mu^2 \mathbf{I}_{\ell_m}),$$

where $\mathbf{I}_\ell$ is an $\ell \times \ell$ identity matrix and $\sigma_\mu^2 = 0.5/(a\sqrt{M})$ with $a > 0$. This prior imposes stronger shrinkage on $\boldsymbol{\mu}_m$ towards zero when we have more weak learners, and therefore prevents overfitting given that we rescale Y into $[-0.5, 0.5]$. We choose $a = 2$ by default, which assigns 0.95 prior probability to $\mathbb{E}(Y|\mathbf{s}, \mathbf{x})$ within $[-0.5, 0.5]$.

The shrinkage prior for $\boldsymbol{\mu}_m$'s, together with the prior for T_m's that favors small sMDTs, ensures that each weak learner only explains a small proportion of response variability, and hence prevents each ensemble membership from being too influential to the overall fit. This regularizes the model to keep it from overfitting the training data.

We complete the prior specification by choosing a conjugate inverse-χ^2 prior for σ^2 in the form of $\sigma^2 \sim \nu\lambda_s/\chi^2_\nu$ for $\lambda_s > 0$ and some degree of freedom ν. We choose $\nu = 3$ and calibrate the prior by selecting λ_s such that $\mathbb{P}(\sigma^2 < \hat{\sigma}^2) = 0.90$ *a priori*, where $\hat{\sigma}^2$ is the sample variance of the responses.

3. Bayesian Inference

Bayesian inference of BAMDT is based on a tailored backfitting Markov chain Monte Carlo (MCMC) sampler (Hastie & Tibshirani, 2000), which successively draws $(T_1, \boldsymbol{\mu}_1), \dots, (T_M, \boldsymbol{\mu}_M)$, and σ^2 from their respective full conditional distributions. To sample from $[T_m, \boldsymbol{\mu}_m | -]$, where $-$ stands for all other parameters and the response data $\mathbf{Y} = (Y_1, \dots, Y_n)$, we first draw T_m from the collapsed full conditional $p(T_m|-) = \int p(T_m, \boldsymbol{\mu}_m|-) d\boldsymbol{\mu}_m$, and then sample $\boldsymbol{\mu}_m$ from $[\boldsymbol{\mu}_m | T_m, -]$. Thanks to the conjugate priors for $\boldsymbol{\mu}_m$ and σ^2, the distributions $[\boldsymbol{\mu}_m | T_m, -]$ and $[\sigma^2 | -]$ admit straightforward closed-form expressions, which are detailed in Appendix B.

To sample a new sMDT $T_m^\star$ from $p(T_m|-)$, we randomly *grow* or *prune* the existing T_m with equal probability to obtain a tree proposal. In a growing move, one of T_m's terminal nodes, denoted by η, is randomly chosen and split into two offspring nodes following Step 2 of the sMDT generating process in Section 2.3. A pruning step does the opposite by first randomly selecting a node with two terminal offspring and then removing its children. The proposed $T_m^\star$ is then accepted or rejected following standard Metropolis-Hastings (MH) procedure, and we leave the details to Appendix B. Note that the MH acceptance probability involves a likelihood ratio $\mathcal{L}(\mathbf{Y}|T_m^\star, -)/\mathcal{L}(\mathbf{Y}|T_m, -)$, where $\mathcal{L}(\mathbf{Y}|T_m, -)$ is the likelihood with $\boldsymbol{\mu}_m$ integrated out. This ratio can be evaluated using its analytical form, thanks to the conjugate Gaussian prior for $\boldsymbol{\mu}_m$. The time complexity to draw T_m is $\mathcal{O}(\max\{n, t\})$ since we utilize a spanning tree that has $t-1$ edges for multivariate splits.

Using posterior draws of $\{(T_m, \boldsymbol{\mu}_m)\}_{m=1}^M$, we can perform prediction for Y_{new} given $(\mathbf{s}_{\text{new}}, \mathbf{x}_{\text{new}})$. A posterior sample of $\mathbb{E}(Y_{\text{new}}|\mathbf{s}_{\text{new}}, \mathbf{x}_{\text{new}})$ is obtained by summing $g(\mathbf{s}_{\text{new}}, \mathbf{x}_{\text{new}}|T_m, \boldsymbol{\mu}_m)$ over $m = 1, \ldots, M$. A point predictor of Y_{new} can be taken as the posterior mean of $\mathbb{E}(Y_{\text{new}}|\mathbf{s}_{\text{new}}, \mathbf{x}_{\text{new}})$ draws.

Similar to BART, BAMDT offers a natural importance metric for variable selection based on MCMC samples. Let r_w be a split rule involving feature w, where w can be $\mathbf{s}$ or one coordinate of $\mathbf{x} = (x_1, \ldots, x_p)$. For $w = \mathbf{s}$, r_w corresponds to a multivariate split; when $w = x_j$, r_w refers to a univariate split on x_j. The relative importance of w is measured by the proportion of r_w used in the sum-of-sMDT model, denoted by v_w. We use the posterior mean of v_w as a metric to evaluate the importance of w. A higher metric indicates that w is more favored in model fitting, and thus it is more likely that w provides more information for predicting Y.

4. Experiments

We demonstrate the performance of BAMDT using some synthetic data in Section 4.1 and 4.2, where we consider two examples of manifold structured feature spaces $\mathcal{M}$. In each example, we generate $n = 500$ random locations in $\mathcal{M}$. The geodesic distance on $\mathcal{M}$ is approximated using the method in Appendix A. The unstructured feature space is set to be $\mathcal{X} \subseteq [0, 1]^p$ with $p \in \{2, 10\}$ (but only one coordinate of $\mathbf{x}$ is involved in the true data generating process). We independently generate x_j for $j = 1, \ldots, p$, but introduce spatial dependence among locations within each x_j to mimic real applications. Using the same data generating scheme, we simulate features for a test data set of size $n_{\text{test}} = 200$. Detailed data generating process can be found in Appendix C.1. We also illustrate the application of BAMDT using a real housing price data set in Section 4.3.

4.1. U-shape Example

The first structured feature space $\mathcal{M}$ that we consider is a two-dimensional U-shape domain as shown in Figure 3(a) that is divided into three subsets by a circle centered at the origin with radius 0.9. As shown in Figure 3(a), we consider a true piecewise smooth function defined on $\mathcal{D}$, where we design two jumps across the surfaces $\{(s_h, s_v) \in \mathcal{M} : s_h^2 + s_v^2 = 0.9^2\} \times \mathcal{X}$. The true function only depends on $(\mathbf{s}, x_1)$ and their interaction. The responses in the training and test data sets are generated according to (1) with noise level $\sigma = 0.1$ (signal-to-noise ratio SNR = 33.7dB). We simulate 50 replicates for each level of p.

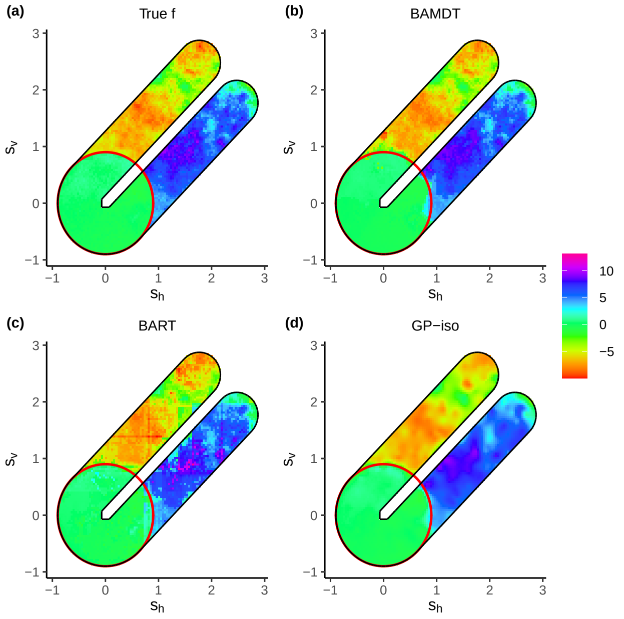

Figure 3. (a) True $f(\mathbf{s}, \mathbf{x})$ on a U-shape domain $\mathcal{M}$ in $\mathbb{R}^2$. (b-d) Predictive surfaces $\hat{f}(\mathbf{s}, \mathbf{x})$ of BAMDT, BART, and GP-iso using one data set with $p = 2$ and $\sigma = 0.1$. Red circles indicate discontinuity surfaces in the true function projected to $\mathcal{M}$.

We use $M = 50$ weak learners in BAMDT. For each weak learner, we randomly sample $t = 100$ locations from the training data as reference knots. We construct spatial graphs $\mathcal{G}^*$ on reference knots using CDTs as in Appendix A, and choose their MSTs based on geodesic distance as the spanning trees for multivariate split rules. We use 100 equally spaced grid points as candidates of univariate split cutoffs for each unstructured feature. The probability of performing a multivariate split is set to be $p_m = 2/(2 + p)$.

We compare BAMDT with the following methods:

- BART using $\mathbf{x}$ and the Cartesian coordinates of $\mathbf{s}$ as input features.
- GP-iso: Spatial GP regression (Gelfand et al., 2010) with a linear mean function of $\mathbf{x}$ and an isotropic Matérn covariance kernel for $\mathbf{s}$.
- GP-aniso: Same as GP-iso but with an anisotropic Matérn covariance kernel for $\mathbf{s}$.
- BAST-s: Bayesian additive regression spanning trees (BAST; Luo et al., 2021a) using the structured features $\mathbf{s}$ only.
- BAST-KNN: BAST using a 10-NN graph based on the Euclidean distance among the scaled joint features,

where each feature in $(\mathbf{s}, \mathbf{x})$ is scaled into $[0, 1]$.

- GAM-additive: Generalized additive models (GAM; see, e.g., Wood, 2017) with a mean function $f(\mathbf{s}, \mathbf{x}) = f_{\text{tps}}(\mathbf{x}) + f_{\text{sfs}}(\mathbf{s})$, where f_{tps} is a thin-plate spline smoother and f_{sfs} is a soap film smoother (SFS; Wood et al., 2008).
- GAM-TP: GAM with $f(\mathbf{s}, \mathbf{x}) = f_{\text{tps}}(\mathbf{x}) \otimes f_{\text{sfs}}(\mathbf{s})$, where $\otimes$ is the tensor product of the two smoothers to capture interaction effects between $\mathbf{x}$ and $\mathbf{s}$.

We use the same number of weak learners ($M = 50$) for the ensemble models BART and BAST. The implementation of the competing methods can be found in Appendix C.1. For the Bayesian methods BAMDT, BART and BAST, we run the MCMC algorithms for $30,000$ iterations, discarding the first half and retaining samples every 10 iterations.

We evaluate prediction performance of BAMDT and its competitors using the test data set. We use mean square prediction error (MSPE) and mean absolute prediction error (MAPE) to measure point prediction accuracy. Point predictors of BAMDT, BART and BAST are based on posterior means, while the ones for GP regressions are the kriging means. For the Bayesian models and GP regressions, we also compare the accuracy of probabilistic prediction using the continuous ranked probability score (CRPS; Gneiting & Raftery, 2007), which is computed using posterior samples of $\mathbb{E}(Y_{\text{new}}|\mathbf{s}_{\text{new}}, \mathbf{x}_{\text{new}})$ for Bayesian models and using the kriging distributions for GP regressions. For all the metrics, lower values indicate better performance.

Table 1(a, b) summarizes the average prediction performance of BAMDT and its competitors over 50 replicates in two settings of p. When $p = 10$, the large dimension of spline basis prevents us from fitting GAM-TP with a limited sample size using R package mgcv (Wood, 2017). In both settings of p, BAMDT outperforms other methods in terms of all performance metrics. In particular, the comparison between BAMDT and BART suggests that the proposed MDTs enhance the performance in complex restricted domains while inheriting BART's feature selection capacity. Indeed, the feature importance metric from BAMDT can better identify the truly relevant features $(\mathbf{s}, x_1)$ compared with BART. As an example, in the setting of $p = 10$ and $\sigma = 0.1$, the average percentage of splits involving $\mathbf{s}$ or x_1 in BAMDT is 73.98%, while the one in BART is 54.62%.

To better examine the prediction from all the models, we present in Figure 3(b-d) the mean predictive surfaces (as a function of $\mathbf{s}$) from BAMDT, BART, and GP-iso fitted using one randomly selected data set with $p = 2$ and $\sigma = 0.1$. All three models can recover the general pattern of the true function, but the result from BAMDT matches the ground truth best. BAMDT performs fairly well in the interior of each subregion of $\mathcal{M}$, while there are some visible errors around the discontinuity surfaces marked by the red circle, which are expected due to larger uncertainties in data around discontinuities. The predictive surface from BART displays some artificial rectangular decision boundaries such as those in the upper arm, due to the sole use of univariate split rules. There is also some noticeable "leakage" effect in the prediction of BART as evidenced by the underestimation in some regions in the lower arm that are near the upper arm. These undesired patterns are overcome in BAMDT thanks to the use of the multivariate split rules that can generate flexibly shaped partitions and respect the domain boundary. Unlike BAMDT or BART, the predictive surface from GP-iso is too smooth relative to the truth and loses some small-scale spatial patterns. We have also compared the predictive surfaces and predictive uncertainty from all competing methods in Appendix C.2. Our result suggests that the discontinuity surfaces in the true function are characterized by higher uncertainty from BAMDT, while this pattern does not appear in BART or GP regressions. We have also included additional simulation settings with different n and SNR, a sensitivity analysis of BAMDT, and some discussion on computation in Appendix C.2.

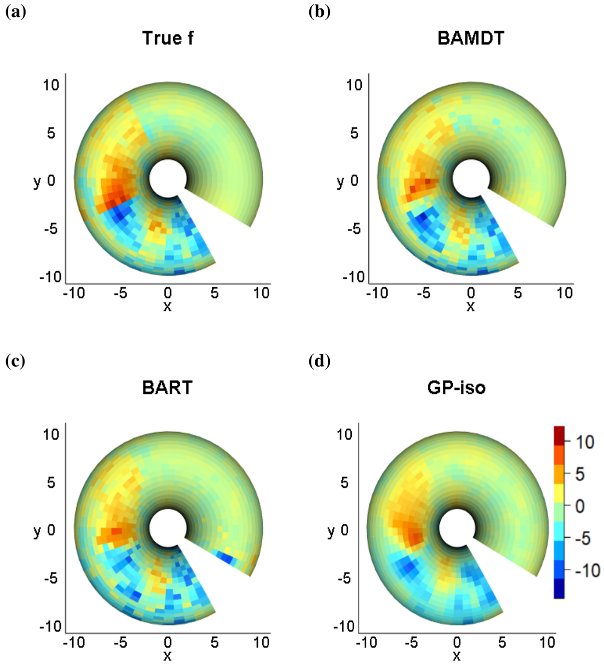

Figure 4. (a) True $f(\mathbf{s}, \mathbf{x})$ on a bitten torus domain $\mathcal{M}$ in $\mathbb{R}^3$. (b-d) Predictive surfaces $\hat{f}(\mathbf{s}, \mathbf{x})$ of BAMDT, BART, and GP-iso using one data set with $p = 2$ and $\sigma = 0.1$. All plots are viewed along the positive direction of the z-axis.

4.2. Bitten Torus Example

We consider the scenario where $\mathbf{s}$ lies in a two-dimensional manifold embedded in $\mathbb{R}^3$. Specifically, the manifold we consider is a bitten torus whose Cartesian coordinate

Table 1. Average prediction performance metrics over 50 replicate data sets in various simulation settings.

(a) U-shape example with $n = 500$, $p = 2$, and $\sigma = 0.1$

	BAMDT	BART	GP-iso	GP-aniso	BAST-s	BAST-KNN	GAM-additive	GAM-TP
MSPE	**0.374**	1.405	0.620	0.583	0.912	2.155	0.985	0.418
MAPE	**0.281**	0.612	0.499	0.488	0.543	0.641	0.720	0.345
Mean CRPS	**0.219**	0.508	0.398	0.390	0.398	0.479	—	—
(b) U-shape example with $n = 500$, $p = 10$, and $\sigma = 0.1$								
MSPE	**0.495**	1.219	0.662	0.632	0.911	1.214	1.168	—
MAPE	**0.317**	0.688	0.545	0.537	0.543	0.662	0.751	—
Mean CRPS	**0.252**	0.552	0.415	0.409	0.398	0.453	—	—
(c) Bitten torus example with $n = 500$, $p = 2$, and $\sigma = 0.1$								
MSPE	**0.967**	1.958	1.569	1.621	1.606	1.678	—	—
MAPE	**0.545**	0.693	0.782	0.805	0.814	0.666	—	—
Mean CRPS	**0.431**	0.573	0.636	0.646	0.591	0.529	—	—
(d) Bitten torus example with $n = 500$, $p = 10$, and $\sigma = 0.1$								
MSPE	**1.169**	2.092	1.563	1.601	1.606	2.056	—	—
MAPE	**0.620**	0.792	0.802	0.819	0.814	0.845	—	—
Mean CRPS	**0.493**	0.646	0.638	0.646	0.591	0.630	—	—

(x, y, z) can be parameterized by $x = (R + r\cos\theta)\cos\phi$, $y = (R + r\cos\theta)\sin\phi$, and $z = r\sin\theta$, where $\theta \in [0, 2\pi]$ is the angle for the torus, $\phi \in [-\pi/6, 5\pi/3]$ is the angle of the tube, $r = 4$ is the fixed radius of the tube, and $R = 6$ is the fixed distance from the tube center to the torus center. See Figure 4(a) for the bitten torus domain. The true mean function $f(\mathbf{s}, \mathbf{x})$ depends on $\mathbf{s}$ and x_1 and has two discontinuities across $\phi = 2\pi/3$ and $\phi = 7\pi/6$. We simulate responses with noise level $\sigma = 0.1$ (SNR = 28.9dB) for 50 replicates.

We compare BAMDT to the same competing models as in Section 4.1 except for the two GAM methods, as the SFS is only applicable to constrained domains in $\mathbb{R}^2$. As shown in Table 1(c, d), BAMDT achieves the best prediction performance in terms of all metrics. Figure 4(b-d) and Figure S3 in the appendix show the predictive surfaces using one selected data set with $p = 2$. Similar to the findings from the U-shape domain, the prediction surface of BART suffers from the "leakage" phenomenon near the bitten portion of the torus, and the ones of GP regressions and BAST models are too smooth compared to the true function. On the other hand, the prediction from BAMDT well matches the truth.

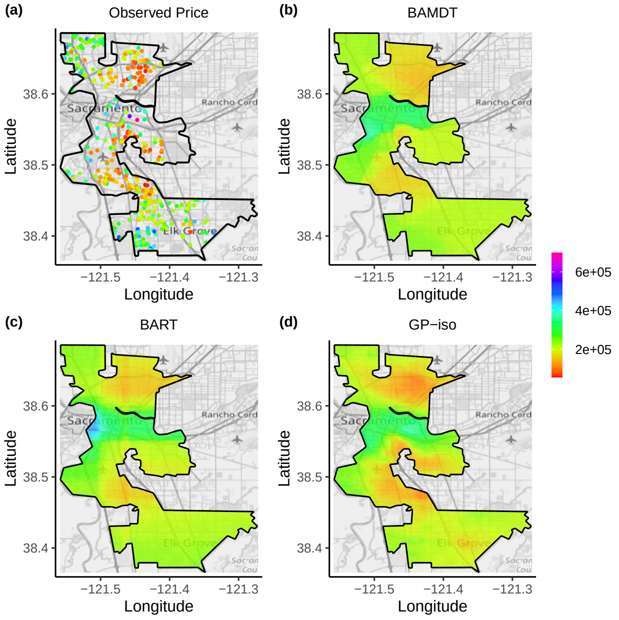

Figure 5. (a) Observed housing data price (in U.S. dollars). (b-d) Predicted price for a representative house from BAMDT, BART, and GP-iso.

4.3. Application to Sacramento Housing Price Data

We apply BAMDT to analyze housing price data in Sacramento County, California, available in R package `caret` (Kuhn, 2021). We focus on $n = 405$ data points from the Cities of Sacramento and Elk Grove. The observed housing price and city boundary [1] are shown in Figure 5(a). Note that the City of Sacramento is divided by the American River near 38.6°N. We model the logarithm of housing price (in U.S. dollars) as a function of the house location (in latitude and longitude), number of bedrooms, number of bathrooms, and square footage. We treat the location as a structured feature $\mathbf{s}$ and all other covariates as unstructured features $\mathbf{x}$. The goal is to examine BAMDT's performance in predicting housing prices with new features.

We fit BAMDT, BART, GP-iso, BAST-KNN, and GAM-additive to the data [2]. The settings of them are identical

[1] City shape file is retrieved from Sacramento County GIS (2015)

[2] We failed to fit GP-aniso using the R package `GpGp` (Guinness, 2018; 2021) due to numerical errors in Cholesky factorization. BAST-s is based on a CDT which is not applicable to duplicate observed $\mathbf{s}$ in this data set. Results of GAM-TP are not available due to the large tensor basis dimension relative to sample size.

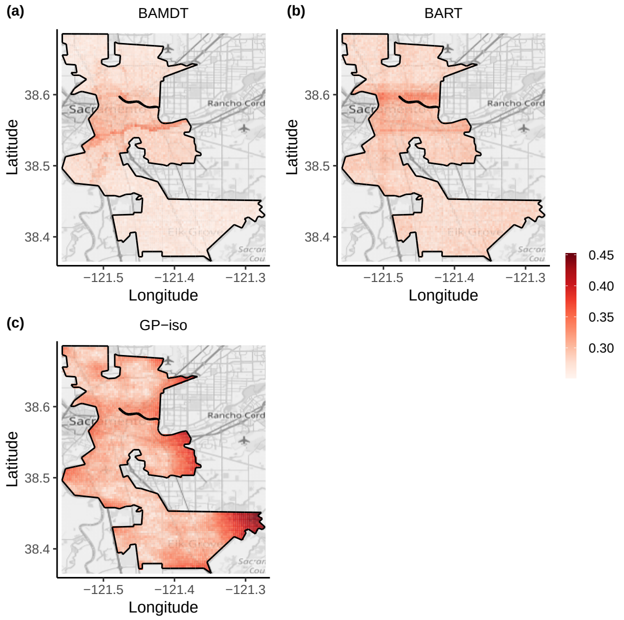

Figure 6. Posterior predictive standard deviation of log-price for a representative house from (a) BAMDT, (b) BART, and (c) GP-iso.

to those in Section 4.1, except that we use $t = 150$ knots for BAMDT. We first compare the prediction performance of the five models using 5-fold cross-validation. Table 2 shows the performance metrics computed using the original price scale (instead of log scale). BAMDT achieves better prediction accuracy than the other methods in all metrics.

Next, we turn to the mean predictive surface fitted using all the observations. We consider a representative house with median unstructured features, namely, three bedrooms, two bathrooms, and 1436 square feet. We display its predicted price at different locations from the selected models in Figure 5(b-d) to examine the marginal spatial effect on housing prices. The predictive surfaces from BART and GP-iso fail to respect the boundary constraints, especially near the American River. In contrast, there is a clear jump across the river on the surface from BAMDT. As in the simulation studies, BART only identifies axis-parallel discontinuities, while BAMDT could detect more flexible discontinuity boundaries with meaningful interpretations such as the one along U.S. Highway 50. Compared with BAMDT and BART, GP-iso tends to give lower predictions in the regions of low housing prices, possibly due to the lack of interaction between $\mathbf{s}$ and $\mathbf{x}$ in the model, and its predicted price changes smoothly near U.S. Highway 50. See Appendix D for additional analyses on predictive surfaces, feature importance, and the marginal effect of square footage.

Table 2. Prediction performance in Sacramento housing data set.

	Root MSPE	MAPE	Mean CRPS
BAMDT	**62128**	**43110**	**34107**
BART	64607	45224	35940
GP-iso	69701	48790	35633
BAST-KNN	79981	55208	43904
GAM-additive	66713	43527	—

Finally, we examine the prediction uncertainty for the representative house. Figure 6 shows the predictive standard deviation of *log-price* from the three models. There is a narrow band with high uncertainty near U.S. Highway 50 and Sacramento Zoo from BAMDT that separates the downtown area and East Sacramento from the southern regions. This band corresponds to an abrupt price change in Figure 5(b), and thus it is associated with higher prediction uncertainty.

5. Conclusion and Discussion

In this paper, we proposed a new Bayesian additive decision tree model for semi-structured regressions. The method relaxes the limitations of conventional BART methods due to axis-parallel split rules by allowing a flexible mixture of univariate and multivariate split rules in decision tree weak learners. The proposed multivariate split rules are built upon a manifold bipartition model via predictive spanning trees that is capable of complying with intrinsic geometry and boundary constraints of the structured feature space.

Thanks to its Bayesian nature, BAMDT is promising to serve as a flexible nonparametric prior for modeling latent functions in various hierarchical modeling settings. The method has great potential beyond predictive regression tasks to other machine learning tasks such as classification, density estimation, survival analysis, and causal inference, to name a few. Moreover, since BAMDT is based on a discretization of the manifold represented by graphs, it naturally applies to the case where structured features lie on a fixed graph such as in spatial areal unit data instead of a compact Riemannian manifold. Besides these extensions, future research may also include theoretical investigations of function approximation performance via Bayesian posterior concentration theories, and computational accelerations via extensions of informed MCMC and importance sampling (Zanella & Roberts, 2019; Griffin et al., 2021; Zhou & Smith, 2022), spike and slab lasso (Ročková & George, 2018), or variational Bayes inference (Blei et al., 2017).

Acknowledgements

The research of Zhao Tang Luo and Huiyan Sang was supported by NSF grant no. NSF DMS-1854655. The research of Bani Mallick was supported by NSF grant no. NSF CCF-1934904. We thank the anonymous reviewers of this paper and the anonymous reviewer Tbpc of Luo et al. (2021a) for helpful comments.

References

Blaser, R. and Fryzlewicz, P. Random rotation ensembles. *The Journal of Machine Learning Research*, 17(1):126–151, 2016.

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112:859–877, 2017.

Borovitskiy, V., Terenin, A., Mostowsky, P., and Deisenroth, M. P. Matérn Gaussian processes on Riemannian manifolds. In *Proceedings of Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Breiman, L. Random forests. *Machine learning*, 45(1): 5–32, 2001.

Cañete-Sifuentes, L., Monroy, R., and Medina-Pérez, M. A. A review and experimental comparison of multivariate decision trees. *IEEE Access*, 2021.

Chen, T. and Guestrin, C. XGBoost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

Chipman, H. A., George, E. I., and McCulloch, R. E. Bayesian CART model search. *Journal of the American Statistical Association*, 93(443):935–948, 1998.

Chipman, H. A., George, E. I., and McCulloch, R. E. BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298, 2010.

Csardi, G. and Nepusz, T. The igraph software package for complex network research. *InterJournal*, Complex Systems:1695, 2006. URL https://igraph.org.

Denison, D., Mallick, B., and Smith, A. A Bayesian CART algorithm. *Biometrika*, 85:363–377, 1998.

Dunson, D. B., Wu, H.-T., Wu, N., et al. Graph based Gaussian processes on restricted domains. *Journal of the Royal Statistical Society Series B*, 84(2):414–439, 2022.

Fan, X., Li, B., Luo, L., and Sisson, S. A. Bayesian nonparametric space partitions: A survey. In Zhou, Z.-H. (ed.), *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pp. 4408–4415. International Joint Conferences on Artificial Intelligence Organization, 8 2021. doi: 10.24963/ijcai.2021/602. URL https://doi.org/10.24963/ijcai.2021/602. Survey Track.

Friedman, J. H. Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, pp. 1189–1232, 2001.

Gelfand, A. E., Diggle, P., Guttorp, P., and Fuentes, M. *Handbook of Spatial Statistics*. CRC press, 2010.

Gneiting, T. and Raftery, A. E. Strictly proper scoring rules, prediction, and estimation. *Journal of the American statistical Association*, 102(477):359–378, 2007.

Griffin, J., Latuszyński, K., and Steel, M. In search of lost mixing time: adaptive Markov chain Monte Carlo schemes for Bayesian variable selection with very large p. *Biometrika*, 108:53–69, 2021.

Guinness, J. Permutation and grouping methods for sharpening Gaussian process approximations. *Technometrics*, 60(4):415–429, 2018.

Guinness, J. Gaussian process learning via Fisher scoring of Vecchia's approximation. *Statistics and Computing*, 31(3):1–8, 2021.

Hastie, T. and Tibshirani, R. Bayesian backfitting (with comments and a rejoinder by the authors). *Statistical Science*, 15(3):196–223, 2000.

He, J., Yalov, S., and Hahn, P. R. XBART: Accelerated Bayesian additive regression trees. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1130–1138. PMLR, 2019.

Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., and Liu, T.-Y. LightGBM: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30:3146–3154, 2017.

Kuhn, M. *caret: Classification and Regression Training*, 2021. URL https://CRAN.R-project.org/package=caret. R package version 6.0-90.

Lee, C., Luo, Z. T., and Sang, H. T-LoHo: A Bayesian regularization model for structured sparsity and smoothness on graphs. *Advances in Neural Information Processing Systems*, 34, 2021.

Lee, D.-T. and Schachter, B. J. Two algorithms for constructing a Delaunay triangulation. *International Journal of Computer & Information Sciences*, 9(3):219–242, 1980.

Li, F. and Sang, H. Spatial homogeneity pursuit of regression coefficients for large datasets. *Journal of the American Statistical Association*, 114(527):1050–1062, 2019.

Lila, E., Sangalli, L. M., Ramsay, J., and Formaggia, L. fdaPDE: Functional data analysis and partial differential equations; statistical analysis of functional and spatial data, based on regression with partial differential regularizations. *R package version 0.1-4, URL https://CRAN. R-project. org/package= fdaPDE*, 2016.

Luo, Z. T., Sang, H., and Mallick, B. BAST: Bayesian additive regression spanning trees for complex constrained domain. *Advances in Neural Information Processing Systems*, 34, 2021a.

Luo, Z. T., Sang, H., and Mallick, B. A Bayesian contiguous partitioning method for learning clustered latent variables. *Journal of Machine Learning Research*, 22(37): 1–52, 2021b.

Nandy, S., Lim, C. Y., and Maiti, T. Additive model building for spatial regression. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(3):779–800, 2017.

Niu, M., Cheung, P., Lin, L., Dai, Z., Lawrence, N., and Dunson, D. Intrinsic Gaussian processes on complex constrained domains. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 81(3):603–627, 2019.

Ramsay, T. Spline smoothing over difficult regions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 64(2):307–319, 2002.

Ročková, V. and George, E. I. The spike-and-slab LASSO. *Journal of the American Statistical Association*, 113:431–444, 2018.

Sacramento County GIS. City boundaries: Sacramento County, California, 2015, 2015. URL https://earthworks.stanford.edu/catalog/stanford-kq595nj1377.

Saha, A., Basu, S., and Datta, A. Random forests for spatially dependent data. *Journal of the American Statistical Association*, pp. 1–19, 2021.

Scott-Hayward, L. A. S., MacKenzie, M. L., Donovan, C. R., Walker, C., and Ashe, E. Complex region spatial smoother (CReSS). *Journal of Computational and Graphical Statistics*, 23(2):340–360, 2014.

Sparapani, R., Spanbauer, C., and McCulloch, R. Nonparametric machine learning and efficient computation with Bayesian additive regression trees: The BART R package. *Journal of Statistical Software*, 97(1):1–66, 2021. doi: 10.18637/jss.v097.i01.

Tan, Y. V. and Roy, J. Bayesian additive regression trees and the general BART model. *Statistics in medicine*, 38 (25):5048–5069, 2019.

Teixeira, L. V., Assunção, R. M., and Loschi, R. H. Bayesian space-time partitioning by sampling and pruning spanning trees. *Journal of Machine Learning Research*, 20: 85–1, 2019.

Tenenbaum, J. B., De Silva, V., and Langford, J. C. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

Watanabe, S. A widely applicable Bayesian information criterion. *Journal of Machine Learning Research*, 14: 867–897, 2013.

Wood, S. *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC, 2nd edition, 2017.

Wood, S. N., Bravington, M. V., and Hedley, S. L. Soap film smoothing. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(5):931–955, 2008.

Yıldız, O. T. Model selection in omnivariate decision trees using structural risk minimization. *Information Sciences*, 181(23):5214–5226, 2011.

Zanella, G. and Roberts, G. Scalable importance tempering and Bayesian variable selection. *Journal of the Royal Statistical Society Series B*, 81:489–517, 2019.

Zhou, Q. and Smith, A. Rapid convergence of informed importance tempering. In *International Conference on Artificial Intelligence and Statistics*, pp. 10939–10965. PMLR, 2022.

A. Details on Spanning Tree Bipartitions

As discussed in Section 2.2, the spanning tree graph $\mathcal{G}_T^*$ on reference knots $\mathcal{S}^*$ can be obtained by finding the geodesic distance-based MST of a graph $\mathcal{G}^* = (\mathcal{S}^*, \mathcal{E}_0^*)$, which is constructed following Luo et al. (2021a). In practice, however, when the number of knots is small or when the shape of $\mathcal{M}$ is highly irregular, the methods in Luo et al. (2021a) may result in a disconnected $\mathcal{G}^*$. To overcome this, one can augment $\mathcal{E}_0^*$ to make $\mathcal{G}^*$ connected using Algorithm 1.

Algorithm 1 Connecting connected components in $\mathcal{G}^*$

```
Input: a graph $\mathcal{G}^* = (\mathcal{S}^*, \mathcal{E}_0^*)$ with $N_c$ connected components.
Initialize $\mathcal{C}$ to be the vertices in one connected component of $\mathcal{G}^*$.
for $i = 1$ to $N_c - 1$ do
    Find the pair of vertices $\mathbf{v}_1 \in \mathcal{C}$ and $\mathbf{v}_2 \in \mathcal{S}^* \setminus \mathcal{C}$ that has the minimal geodesic distance.
    Add the edge $(\mathbf{v}_1, \mathbf{v}_2)$ to $\mathcal{E}_0^*$.
    Set $\mathcal{C} \leftarrow \mathcal{C} \cup \mathcal{C}'$, where $\mathcal{C}'$ is the connected component containing $\mathbf{v}_2$.
end for
Output: a connected graph $\mathcal{G}^*$.
```

The constructions of the aforementioned graphs and $\pi_2(\mathcal{M}_\eta)$ rely on the geodesic distance d_g in $\mathcal{M}$. For many manifolds, d_g has no analytical form. Fortunately, we can approximate d_g between any two locations in a way similar to the Isomap algorithm (Tenenbaum et al., 2000). To be more specific, we construct a dense weighted nearest neighbor graph based on Euclidean distance on some fine grids in $\mathcal{M}$ and the locations of interest, and then approximate the geodesic distance between the two locations by the length of the shortest path between them in the dense graph.

B. Details on Bayesian Inference

This appendix provides details on the Markov chain Monte Carlo (MCMC) algorithm in Section 3. Given data $(\mathbf{s}_1, \mathbf{x}_1, Y_1), \dots, (\mathbf{s}_n, \mathbf{x}_n, Y_n)$, let $\mathbf{g}_m$ be the vector of in-sample fitted values from the mth weak learner, i.e., the ith element of $\mathbf{g}_m$ is $g(\mathbf{s}_i, \mathbf{x}_i | T_m, \boldsymbol{\mu}_m)$. Define the partial residual from the mth weak learner as

$$\mathbf{r}_m = \mathbf{Y} - \sum_{k \neq m} \mathbf{g}_k.$$

As discussed in Section 3, our MCMC sampler successively draw samples from the full conditional distributions of $(T_1, \boldsymbol{\mu}_1), \dots, (T_M, \boldsymbol{\mu}_M)$, and σ^2. To sample from each $p(T_m, \boldsymbol{\mu}_m | -)$, we proceed in two steps. First, we update T_m using a Metropolis-Hastings (MH) sampler by drawing T_m from $p(T_m|-)$, the full conditional distribution of T_m with $\boldsymbol{\mu}_m$ integrated out. Specifically, we propose a new sMDT $T_m^\star$ by a growing or a pruning move as detailed in Section 3. In a *growing* move, letting η be the node we split, the MH acceptance probability is given by

$$\min\left\{1,\ \frac{\alpha(1+d_\eta)^{-\beta}[1-\alpha(2+d_\eta)^{-\beta}]^2}{1-\alpha(1+d_\eta)^{-\beta}} \cdot \frac{N_s}{N_m} \cdot \frac{\mathcal{L}(\mathbf{r}_m|T_m^\star, -)}{\mathcal{L}(\mathbf{r}_m|T_m, -)}\right\}, \tag{S1}$$

where N_s is the number of terminal nodes in T_m, N_m is the number of non-terminal nodes with two terminal children in T_m, and $\mathcal{L}(\mathbf{r}_m|T_m, -)$ is the likelihood with $\boldsymbol{\mu}_m$ marginalized out. Thanks to the conjugate prior on $\boldsymbol{\mu}_m$, $\mathcal{L}(\mathbf{r}_m|T_m, -)$ can be explicitly evaluated by

$$\mathcal{L}(\mathbf{r}_m|T_m, -) \propto |\mathbf{P}_m|^{-1/2} \exp\left(-\frac{1}{2}\mathbf{r}_m^\top \mathbf{P}_m^{-1} \mathbf{r}_m\right),$$

where $\mathbf{P}_m = \sigma^2 \mathbf{I}_n + \sigma_\mu^2 \mathbf{Z}_m \mathbf{Z}_m^\top$ and $\mathbf{Z}_m$ is an $n \times \ell_m$ binary matrix whose (i,j)th element is 1 if and only if the ith observation is assigned to the jth terminal node of T_m. In practice, utilizing the fact that $\mathbf{Z}_m$ has reduced rank ℓ_m, we use Sherman-Woodbury-Morrison formula to simplify the computation of $|\mathbf{P}_m|$ and $\mathbf{P}_m^{-1}$. The MH acceptance probability of a pruning move is analogous to (S1).

The second step to sample from $p(T_m, \boldsymbol{\mu}_m|-)$ is to draw $\boldsymbol{\mu}_m$ from $p(\boldsymbol{\mu}_m|T_m, -)$, which admits a closed form

$$[\boldsymbol{\mu}_m | T_m, -] \sim \mathrm{N}_{\ell_m}\left(\mathbf{Q}_m^{-1}\mathbf{b}_m, \mathbf{Q}_m^{-1}\right),$$

with $\mathbf{Q}_m = \mathbf{Z}_m^\mathsf{T}\mathbf{Z}_m/\sigma^2 + \mathbf{I}_{\ell_m}/\sigma_\mu^2$ and $\mathbf{b}_m = \mathbf{Z}_m^\mathsf{T}\mathbf{r}_m/\sigma^2$.

Finally, the full conditional of σ^2 is an inverse gamma distribution of the form

$$[\sigma^2|-] \sim \mathrm{IG}\left(\frac{n+\nu}{2},\ \frac{1}{2}\left[\nu\lambda_s + \|\mathbf{Y} - \sum_{m=1}^{M}\mathbf{g}_m\|^2\right]\right),$$

where $\|\cdot\|$ is the Euclidean norm.

C. Supplementary Simulation Details

C.1. Details on Simulation Setup

We consider a manifold structured feature space $\mathcal{M}$ (which is a U-shape domain in Section 4.1 and a bitten torus in Section 4.2) and generate random locations $\mathbf{s}$ in $\mathcal{M}$. Below, we discuss the generation of unstructured features $\mathbf{x}$. In many applications, there is oftentimes spatial dependence among locations within an unstructured feature. To simulate spatially correlated features, we first find a homomorphism $\mathbf{u} = h(\mathbf{s})$ from $\mathcal{M}$ to a rectangular region in $\mathbb{R}^d$, with $d = 2$ in both examples in Section 4. Then we simulate p independent realizations $\{\zeta_1\}, \ldots, \{\zeta_p\}$ from a Gaussian process using Euclidean distance on $\mathbf{u} = (u_1, u_2)$. We further use the transformation $x_j = \Phi(\zeta_j)$ to generate unstructured features within $[0, 1]$, where Φ is the cumulative distribution function of standard Gaussian distribution.

For the U-shape example in Section 4.1, motivated by Ramsay (2002), we construct a true function as $f_U(\mathbf{s}, \mathbf{x}) = b_0 + b_1(u_1x_1 + u_2^2)$ for some constants b_0 and b_1, which only depends on the structured features $\mathbf{s} = (s_h, s_v)$ and one of the unstructured features. We allow b_0 and b_1 to take different values in different subregions of $\mathcal{M}$ to create discontinuities. Specifically, we divide $\mathcal{M}$ into three subsets separated by a circle:

$$\begin{aligned}
\mathcal{M}_1 &= \{(s_h, s_v) \in \mathcal{M} : s_h^2 + s_v^2 > 0.9^2 \text{ and } s_h < s_v\},\\
\mathcal{M}_2 &= \{(s_h, s_v) \in \mathcal{M} : s_h^2 + s_v^2 > 0.9^2 \text{ and } s_h > s_v\},\\
\mathcal{M}_3 &= \{(s_h, s_v) \in \mathcal{M} : s_h^2 + s_v^2 \leq 0.9^2\}.
\end{aligned}$$

We set $b_0 = -4$ and $b_1 = 1$ in $\mathcal{M}_1$, $b_0 = 4$ and $b_1 = 1$ in $\mathcal{M}_2$, and $b_0 = 0$ and $b_1 = -0.5$ in $\mathcal{M}_3$.

For the bitten torus example in Section 4.2, we modify the true function in Niu et al. (2019) to involve the unstructured feature x_1. Specifically, the true function is given by

$$f_T(\mathbf{s}, \mathbf{x}) = \begin{cases} x_1\phi + 0.3\sin(\theta), & \text{if } \phi \in [-\pi/6, 2\pi/3] \cup [7\pi/6, 5\pi/3],\\ -x_1\phi - 0.3\sin(\theta), & \text{if } \phi \in (2\pi/3, 7\pi/6). \end{cases}$$

The spatial graph $\mathcal{G}^*$ on reference knots we use for this example is a 5-NN graph under geodesic distances.

We implement BAMDT in `R` using the packages `igraph` (Csardi & Nepusz, 2006) for graph operations and `fdaPDE` (Lila et al., 2016) to construct CDT graphs on two-dimensional constrained domains. All the competing methods are implemented in `R`, using the packages `BART` (Sparapani et al., 2021) for BART, `GpGp` (Guinness, 2018; 2021) for GP-iso and GP-aniso, and `mgcv` (Wood, 2017) for GAM-additive and GAM-TP. We use the default settings for BART except for the number of weak learners. The implementation of BAST-s and BAST-KNN follows Luo et al. (2021a). In both GAM-additive and GAM-TP, we choose regular grids as knots, and set the dimension of the SFS basis for $\mathbf{s}$ as 40. We use the default dimension of the thin-plate basis for $\mathbf{x}$ in GAM-additive and set it as 5 in GAM-TP.

C.2. Supplementary Results for U-shape Example

We first consider three additional simulation settings in the U-shape example in Section 4.1. Specifically, in the first two settings, we simulate $n = 500$ training responses and $n_{\text{test}} = 200$ test responses with noise level $\sigma = 0.5$ (reducing SNR to 19.7dB) and unstructured features $\mathbf{x}$ of dimension $p \in \{2, 10\}$. Table S1(a, b) summarizes prediction performance for these two settings, and BAMDT outperforms all its competitors in terms of all metrics. In the third setting, we use $\sigma = 0.1$ and $p = 2$ but increase the sample size to $n = 2000$ and $n_{\text{test}} = 500$. The prediction results for this setting are shown in Table S1(c). BAMDT achieves better performance in terms of MAPE and mean CRPS, while the two GP regression models have the lowest MSPE, possibly because the MSPE of BAMDT is dominated by some large errors.

Table S1. Average prediction performance metrics over 50 replicate data sets in various simulation settings. Standard deviations are in parentheses.

(a) U-shape example with $n = 500$, $p = 2$, and $\sigma = 0.5$

	BAMDT	BART	GP-iso	GP-aniso	BAST-s	BAST-KNN	GAM-additive	GAM-TP
MSPE	**0.685**	1.679	0.949	0.916	1.202	2.461	1.276	0.771
	(0.081)	(0.244)	(0.067)	(0.064)	(0.090)	(0.093)	(0.072)	(0.066)
MAPE	**0.567**	0.829	0.723	0.713	0.737	0.864	0.851	0.623
	(0.026)	(0.048)	(0.031)	(0.031)	(0.031)	(0.027)	(0.029)	(0.028)
Mean CRPS	**0.438**	0.656	0.528	0.520	0.547	0.623	—	—
	(0.023)	(0.042)	(0.019)	(0.018)	(0.023)	(0.021)		
(b) U-shape example with $n = 500$, $p = 10$, and $\sigma = 0.5$								
MSPE	**0.756**	1.580	1.008	0.979	1.203	1.509	1.454	—
	(0.131)	(0.211)	(0.070)	(0.070)	(0.094)	(0.077)	(0.087)	
MAPE	**0.584**	0.878	0.754	0.747	0.735	0.838	0.884	—
	(0.034)	(0.057)	(0.031)	(0.031)	(0.031)	(0.026)	(0.029)	
Mean CRPS	**0.453**	0.686	0.544	0.539	0.545	0.585	—	—
	(0.032)	(0.050)	(0.019)	(0.019)	(0.023)	(0.019)		
(c) U-shape example with $n = 2000$, $p = 2$, and $\sigma = 0.1$								
MSPE	0.472	0.742	**0.343**	0.347	0.370	0.440	0.734	0.353
	(0.112)	(0.186)	(0.007)	(0.006)	(0.009)	(0.007)	(0.008)	(0.018)
MAPE	**0.242**	0.332	0.392	0.393	0.397	0.304	0.589	0.274
	(0.023)	(0.034)	(0.005)	(0.005)	(0.007)	(0.004)	(0.004)	(0.005)
Mean CRPS	**0.208**	0.282	0.301	0.302	0.312	0.239	—	—
	(0.022)	(0.031)	(0.003)	(0.003)	(0.007)	(0.004)		

Next, we compare the predictive surfaces and predictive uncertainties using the same simulated data as in Section 4.1 under the setting of $n = 500$, $p = 2$ and $\sigma = 0.1$. Figure S1 shows the predictive surfaces from the competing methods that are not included in Section 4.1 due to space limitations. Similar to the predictive surface from GP-iso in Figure 3(d), the ones from GP-aniso, BAST-s and GAM-add are relatively too smooth. Due to the use of the 10-NN graph based on the Euclidean distance of the joint features $(\mathbf{x}, \mathbf{s})$ that include many irrelevant features, the prediction from BAST-KNN does not fully respect the domain boundary constraints. The predictive surface from GAM-TP is close to the one from BAMDT, but GAM-TP overpredicts in some regions inside the red circle. The predictive uncertainties at different spatial locations are shown in Figure S2. As expected, the posterior predictive standard deviation (SD) from BAMDT is higher around the discontinuity surfaces, reflecting the uncertainty due to the unknown discontinuities. The uncertainty measures from BART and GP regressions, however, fail to capture this. In the predictive SD for BART, one can observe some artificial axis-parallel high uncertainty regions probably resulting from univariate splits on $\mathbf{s}$. The uncertainty of GP-iso and GP-aniso at unobserved locations is generally higher than the one for BAMDT, possibly due to misspecification of the model, especially in the mean function. BAST-s and BAST-KNN also exhibit high uncertainty around the true discontinuities, but in other regions, their uncertainty is still larger than BAMDT, possibly due to the poor graph construction that does not use the true feature x_1 or include many irrelevant features in $\mathbf{x}$. Note that GAM-additive and GAM-TP do not provide natural uncertainty measures.

We also compare BAMDT with $M = 50$ weak learners to BART with $M = 50, 100, 200, 400$ weak learners under the setting of $n = 500$, $p = 2$ and $\sigma = 0.1$, and their prediction performance is shown in Table S2. BAMDT with $M = 50$ weak learners outperforms BART with more weak learners, suggesting that, compared with the sole use of univariate splits, the multivariate splits can help BAMDT capture complex true functions on constrained domains in a more efficient way.

To examine the sensitivity of BAMDT's prediction performance to the hyperparameters, we focus on a data set under the setting of $n = 500$, $p = 2$, and $\sigma = 0.1$, and consider different values of the number of weak learners M, the number of reference knots t, and the prior probability for performing a multivariate split p_m. Prediction metrics are shown in Table S3. We also include the value of widely applicable Bayesian information criterion (WBIC; Watanabe, 2013) for each hyperparameter combination. The models with the lowest and the second lowest WBIC values have the overall best prediction performance. When the prediction performance is a concern, we recommend using standard hyperparameter selection methods such as WBIC and cross-validation to fine tune the model.

As a final remark, the average computation time in Section 4.1 with $n = 500$, $p = 2$ and $\sigma = 0.1$ is 65 minutes using a pure

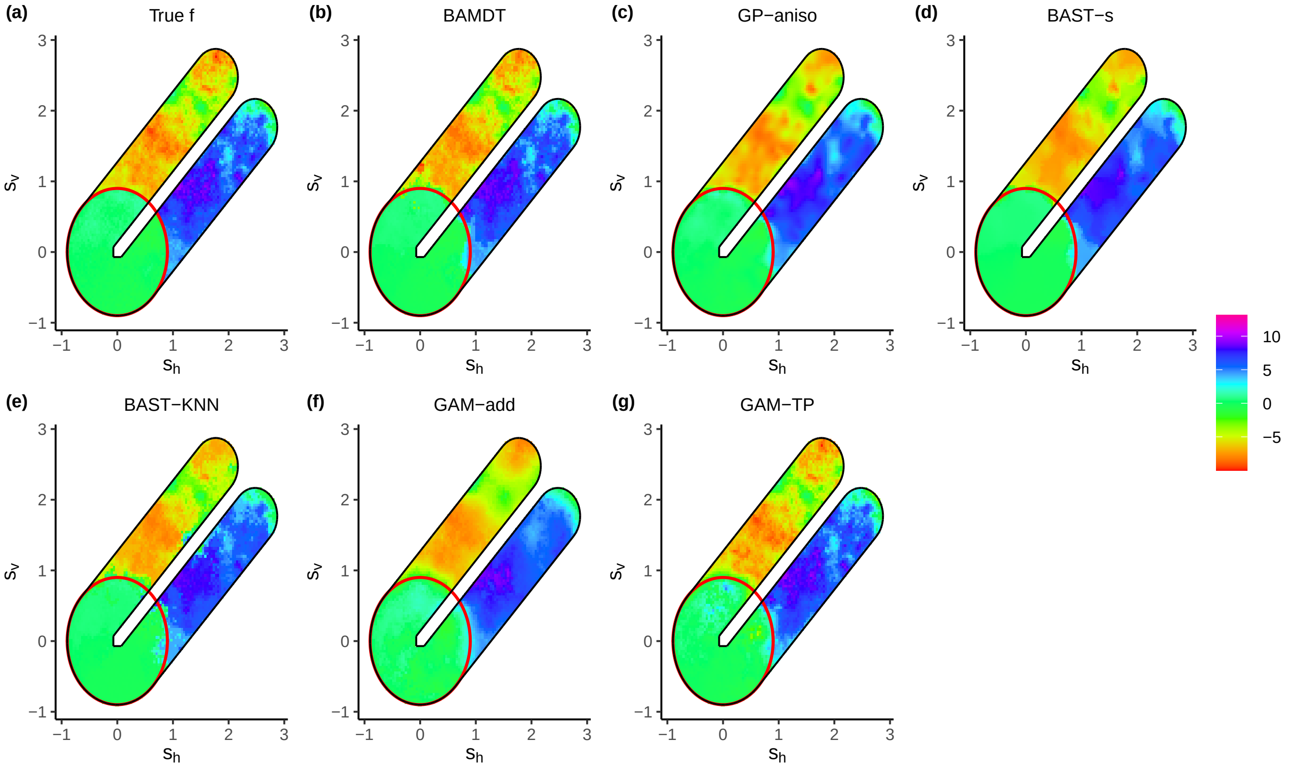

Figure S1. Predictive surfaces $\hat{f}(\mathbf{s}, \mathbf{x})$ of (b) BAMDT, (c) GP-aniso, (d) BAST-s, (e) BAST-KNN, (f) GAM-additive, and (g) GAM-TP using one data set with $n = 500$, $p = 2$ and $\sigma = 0.1$. The true function $f(\mathbf{s}, \mathbf{x})$ is also included for reference in (a). Red circles indicate discontinuity surfaces in the true function projected to $\mathcal{M}$.

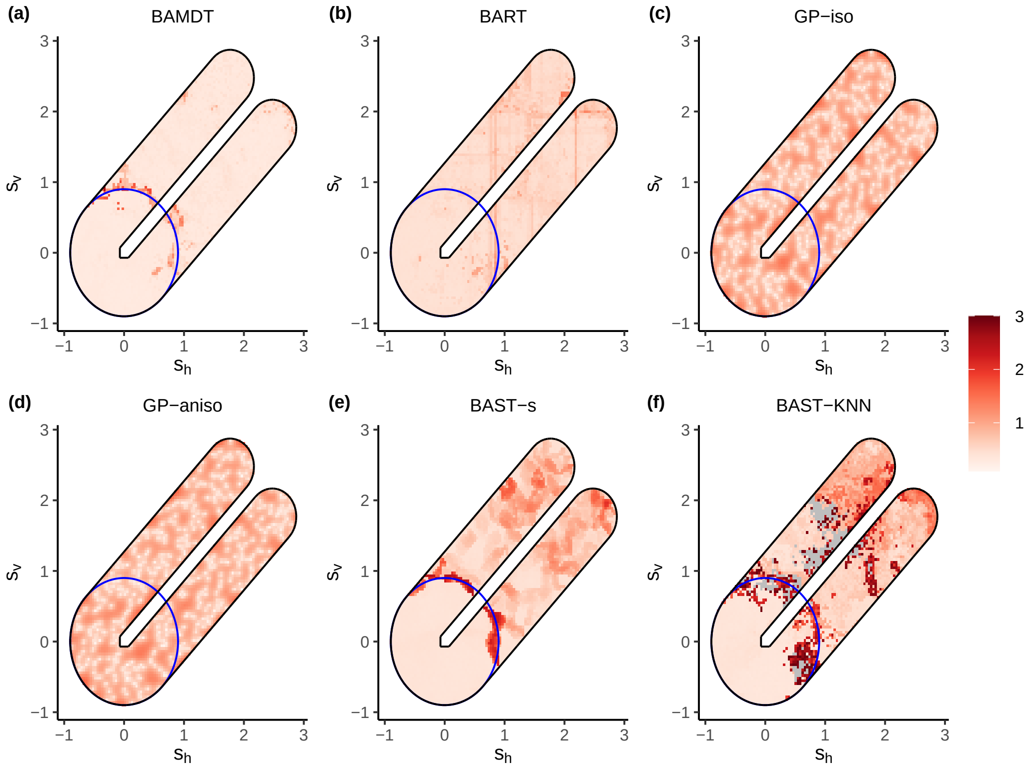

Figure S2. Posterior predictive standard deviation surfaces of (a) BAMDT, (b) BART, (c) GP-iso, (d) GP-aniso, (e) BAST-s and (f) BAST-KNN in the setting of $n = 500$, $p = 2$ and $\sigma = 0.1$. Standard deviation larger than 3 is marked by gray. Blue circles indicate discontinuity surfaces in the true function projected to $\mathcal{M}$.

`R` implementation, while fitting BART and BAST-s in `C++` takes 50 seconds and 6 minutes respectively. We are currently investigating a more efficient implementation of BAMDT in `C++`.

D. Supplementary Real Data Analysis

In this appendix, we provide more analysis of the Sacramento housing price data.

We first examine the maps of predicted price for the representative house from BAST-KNN and GAM-additive shown in Figure S4(a, b). Compared with the prediction from BAMDT, BAST-KNN underpredicts the housing price in the middle Sacramento region, possibly because the K-NN graph is not an efficient way to incorporate unstructured information. GAM-additive, on the other hand, predicts much higher housing price near $(121.42°\text{W}, 38.55°\text{N})$, possibly due to the lack of interaction between $\mathbf{s}$ and $\mathbf{x}$ in the model. As shown in Figure S4(c), the prediction from BAST-KNN has high uncertainty, which is similar to the finding in the U-shape domain example.

Next, we turn to feature importance in BAMDT. 44.92% of the splits are attributed to the structured feature $\mathbf{s}$, suggesting that a large part of the variation in Sacramento housing prices can be explained by the spatial component of the model. The square footage feature is the second important feature in BAMDT, followed by the number of bathrooms and bedrooms. A similar feature importance pattern is found using BART.

In Section 4.3, we have examined the marginal effect of the spatial location $\mathbf{s}$. Below, we focus on the marginal effect of square footage. We choose five representative locations in Downtown Sacramento (green), North Natomas (cyan), North Sacramento (red), Valley Hi / North Laguna (blue), and Elk Grove (pink), as shown in Figure S5(a). Figure S5(b) shows the predicted price of houses with three bedrooms, two bathrooms, and various square footages. As expected, there is a positive nonlinear relationship between price and square footage at each selected location, and there is a noticeable

Table S2. Average prediction performance metrics over 50 replicate data sets of BAMDT and BART with various numbers of weak learners, under the setting of $n = 500$, $p = 2$ and $\sigma = 0.1$. Standard deviations are in parentheses.

	BAMDT ($M = 50$)	BART ($M = 50$)	BART ($M = 100$)	BART ($M = 200$)	BART ($M = 400$)
MSPE	**0.374**	1.405	1.234	1.162	1.126
	(0.106)	(0.249)	(0.217)	(0.161)	(0.069)
MAPE	**0.281**	0.612	0.590	0.586	0.627
	(0.025)	(0.056)	(0.051)	(0.029)	(0.024)
Mean CRPS	**0.219**	0.508	0.475	0.458	0.481
	(0.023)	(0.052)	(0.046)	(0.026)	(0.018)

Table S3. Prediction performance and WBIC of BAMDT under different settings of hyperparameters.

M	t	p_m	MSPE	MAPE	Mean CRPS	WBIC
50	100	0.75	0.318	0.268	0.214	364.676
100	100	0.75	**0.258**	**0.232**	**0.178**	**296.928**
50	200	0.75	0.475	0.292	0.232	301.478
100	200	0.75	0.442	0.288	0.222	311.126
50	100	0.50	**0.222**	**0.236**	**0.175**	**295.329**
100	100	0.50	0.305	0.239	0.183	331.875
50	200	0.50	0.395	0.300	0.244	325.121
100	200	0.50	0.409	0.270	0.210	331.168
50	100	0.25	0.482	0.305	0.239	343.953
100	100	0.25	0.327	0.261	0.199	342.134
50	200	0.25	0.457	0.304	0.235	308.955
100	200	0.25	0.361	0.259	0.196	350.442

change in the relationship near 1600 square feet. The marginal effect of square footage also depends on the locations: Downtown Sacramento has the highest price per square feet, while North Sacramento has the lowest. 95% predictive credible intervals of these two locations are also shown. The credible intervals are wider for larger houses, probably because of the log-transformation of price in the model.

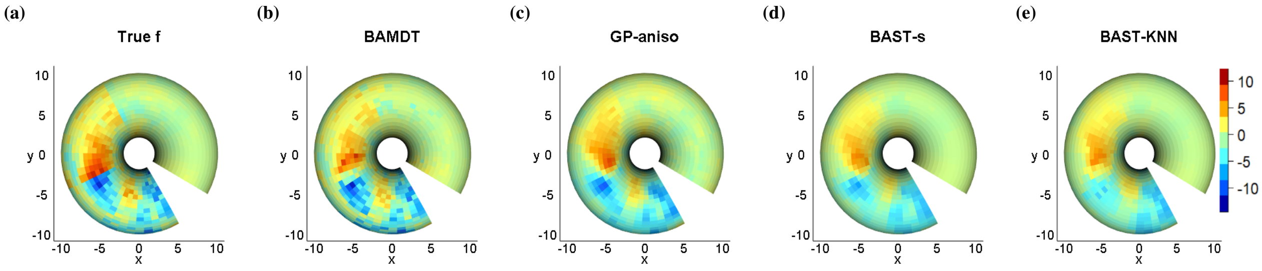

Figure S3. Predictive surfaces $\hat{f}(\mathbf{s}, \mathbf{x})$ of (b) BAMDT, (c) GP-aniso, (d) BAST-s and (e) BAST-KNN using one data set with $p = 2$ and $\sigma = 0.1$. The true function $f(\mathbf{s}, \mathbf{x})$ is also included for reference in (a). All plots are viewed along the positive direction of the z-axis.

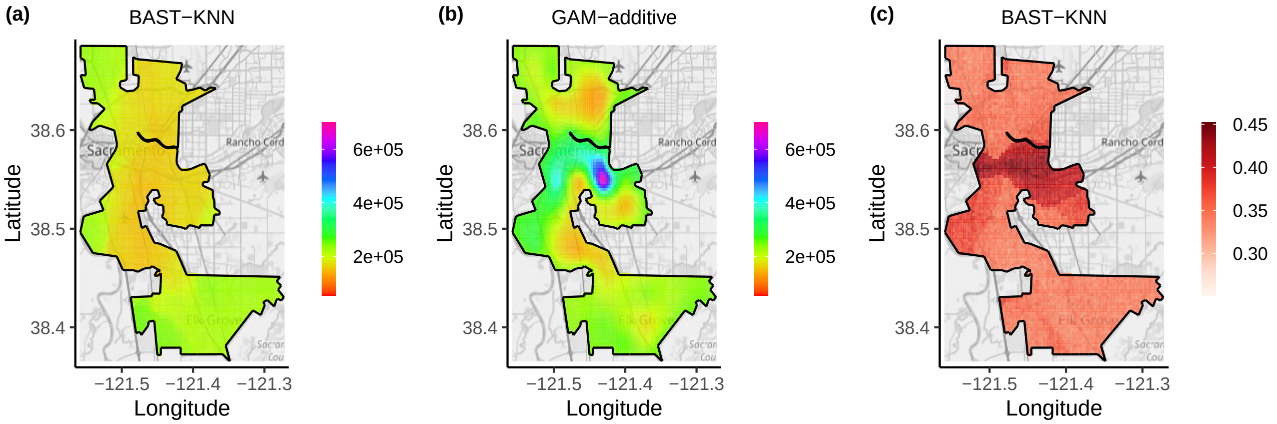

Figure S4. (a, b) Predicted price for a representative house from BAST-KNN and GAM-additive. (c) Posterior predictive standard deviation of log-price from BAST-KNN.

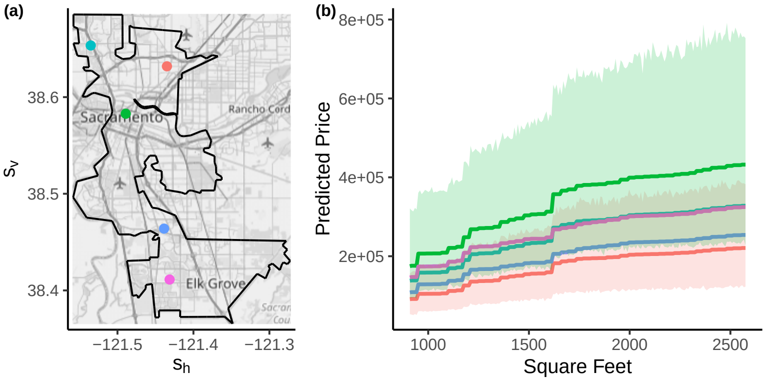

Figure S5. (a) Map of five representative locations. (b) Predicted price versus square footage of the houses. Colored ribbons represent 95% predictive credible intervals of two representative locations.