Bayesian Probabilistic Numerical Integration with Tree-Based Models

Harrison Zhu, Xing Liu
Imperial College London
{hbz15,xl6116}@ic.ac.uk

Ruya Kang
Brown University
ruya_kang@brown.edu

Zhichao Shen
University of Oxford
zhichao.shen@new.ox.ac.uk

Seth Flaxman
Imperial College London
s.flaxman@imperial.ac.uk

François-Xavier Briol
University College London
f.briol@ucl.ac.uk

Abstract

Bayesian quadrature (BQ) is a method for solving numerical integration problems in a Bayesian manner, which allows users to quantify their uncertainty about the solution. The standard approach to BQ is based on a Gaussian process (GP) approximation of the integrand. As a result, BQ is inherently limited to cases where GP approximations can be done in an efficient manner, thus often prohibiting very high-dimensional or non-smooth target functions. This paper proposes to tackle this issue with a new Bayesian numerical integration algorithm based on Bayesian Additive Regression Trees (BART) priors, which we call BART-Int. BART priors are easy to tune and well-suited for discontinuous functions. We demonstrate that they also lend themselves naturally to a sequential design setting and that explicit convergence rates can be obtained in a variety of settings. The advantages and disadvantages of this new methodology are highlighted on a set of benchmark tests including the Genz functions, and on a Bayesian survey design problem.

1 Introduction

Numerical integration algorithms are key tools for modern statistics and machine learning. They allow us to tackle many of the integrals commonly arising in those fields and which cannot be computed in closed form. Examples include the computation of posterior expectations in Bayesian statistics, the marginalisation or conditioning of random variables, the computation of normalisation constants, calculations for the EM algorithm, or even the approximation of solutions to differential equations.

This paper will focus on the task of approximating the integral of some function $f : \mathcal{X} \to \mathbb{R}$ which is integrable with respect to some distribution Π (whose density with respect to the Lebesgue measure is denoted π) over some domain $\mathcal{X} \subseteq \mathbb{R}^d$ ($d \in \mathbb{N}_+ = \mathbb{N}\backslash\{0\}$):

$$\Pi[f] := \int_{\mathcal{X}} f(x)d\Pi(x) = \int_{\mathcal{X}} f(x)\pi(x)dx. \tag{1}$$

A large number of methods have been developed to tackle this problem. Classical quadrature rules [18] tend to be limited to low-dimensional problems and integrals against a small class of probability measures. Alternatively, Monte Carlo integration (MI) [92] only requires sampling independently from Π and enjoys a convergence rate of $\mathcal{O}(n^{-1/2})$. When π is unnormalised, sequential Monte Carlo (SMC) [20] or Markov chain Monte Carlo (MCMC) [92] samplers can be used and enjoy a similar rate. Quasi-Monte Carlo (QMC) [23] can improve on this convergence rate, but is limited to integration against a uniform measure in a hyper-cube (or simple transformations of this problem). We note that all of these methods are *quadrature rules*: they take the form $\hat{\Pi}[f] := \sum_{i=1}^{n} w_i f(x_i)$ where

$\{x_i\}_{i=1}^n \subset \mathcal{X}$ are design points and $\{w_i\}_{i=1}^n \subset \mathbb{R}$ are weights. Finally, the Laplace approximation [99] and variational inference [4] can be used, but these cannot be guaranteed to provide accurate estimates as n grows.

Although all of the methods above are commonly used in practice, they all lack a straightforward non-asymptotic approach to quantifying our uncertainty about the value of $\Pi[f]$ after a finite number of function evaluations. This is not necessarily a problem for applications where n can be taken to be large relative to the difficulty of the problem. However, those methods will be sub-optimal when f is both difficult to approximate and expensive to evaluate since the asymptotic results will not hold.

An alternative approach comes from the field of probabilistic numerics [22; 45; 58; 76; 103], and in particular Bayesian probabilistic numerical methods [16], which frame problems in numerical analysis as statistical estimation tasks. This allows for the quantification of uncertainty surrounding the value of a quantity of interest using probabilistic statements valid for finite n. For integration, the main approach is called Bayesian quadrature (BQ) [6; 22; 75; 90]. Its name comes from the fact that the estimator takes the form of a quadrature rule, but also carries a Bayesian interpretation. However, we will see that not all Bayesian estimators have this property. For this reason, we prefer the general name of *Bayesian probabilistic numerical integration (BPNI)*.

BPNI starts by positing a prior distribution for the integrand f. It then computes the posterior distribution given values of f at some design points, and finally considers the implied distribution on $\Pi[f]$. The main advantage of BPNI is that all our knowledge about the problem can be straightforwardly implemented in a prior, and the posterior distribution on $\Pi[f]$ can allow us to quantify our uncertainty about the exact value of this quantity. BPNI critically relies upon a flexible statistical model for the integrand f, which is almost always chosen to be a Gaussian process (GP). In this case, we have a quadrature rule which we will call GP-BQ. GPs have the convenient property that the posterior distribution can be obtained in closed form for interpolation and regression with Gaussian noise. They also have well-studied concentration rates which carry over to the corresponding GP-BQ methods [6; 51; 52; 111].

GP-BQ has received a lot of attention in recent years. Efficient deterministic [53; 54], adaptive [5; 29; 40; 50] and randomised [3] point-selection schemes have been designed. The method was extended to cases where π is not available [72], and when multiple integrals are computed simultaneously [38; 112]. It has been applied to fields ranging from econometrics [73] to computer graphics [7] and robotics [86]. While GPs are virtually the only choice that has been explored in the BPNI literature, in practice they suffer from a number of challenges. These include:

1. **Discontinuities:** GPs are not suited for integration problems where the integrand is discontinuous – a common challenge in applications [17; 66]. As a result, they may lead to poor predictions and unreliable uncertainty estimates for these problems.

2. **Computational cost:** A major issue with GPs is their cubic cost in the number of data points. This can be mitigated using approximate GPs, but these do not necessarily lead to closed-form estimators. Exceptions include the work of [53] and [48] which allow for near-linear computational cost, but require Π and $\{x_i\}_{i=1}^n$ to exhibit symmetry properties which may not hold in practice.

3. **High dimensions:** Few applications of GP-BQ exists where $d > 10$ due to the curse of dimensionality. This issue is closely linked to the computational cost issue since the number of points needed to approximate functions well will grow exponentially with dimension. However, it is sometimes possible to use sparsity of f to lower this cost [6].

To tackle some of these issues, we propose to use an alternative model based on trees. The specific model is called Bayesian additive regression trees (BART) [14; 46], and as a result we call this BPNI method *BART integration (BART-Int)*. BART is a sum-of-trees model similar in spirit and effectiveness to random forests. It has been successfully deployed in a number of settings including for causal inference [26; 47], genomics [24], behavioural sciences [113] and Bayesian optimisation [15]. Extensive experimental results in the literature have shown that BART does not usually overfit, thanks to its ensemble structure, and that it only requires a limited amount of hyperparameter tuning.

Contributions of the paper: This paper derives a novel BPNI algorithm based on Bayesian priors with a tree structure. We show that BART is a natural choice for BPNI which can be preferable to GPs in any of the three settings highlighted. This is done through a mix of theory and experiments. On the theoretical side, we prove asymptotic convergence rates for BPNI with MCMC approximations. This

result may be of independent interest since it is applicable for any nonparametric method for which a concentration rate is known. On the experimental side, we compare our method to GP-BQ on a range of problems including the Genz test functions, a rare-event simulation problem and a Bayesian survey design problem.

2   Background: Bayesian Regression with GPs and BART

Before presenting BART-Int, this section introduces background material on Bayesian regression with GPs and BART. Given data (X, y) consisting of design points $X = (x_1, \ldots, x_n)^\top \subset \mathcal{X}^n$ and responses $y = (y_1, \ldots, y_n)^\top \subset \mathbb{R}^n$, the regression problem is to recover an unknown function $f : \mathcal{X} \to \mathbb{R}$ for which we have access to noisy measurements y. These are usually assumed to be independently and identically distributed (i.i.d.) Gaussian: $y_i = f(x_i) + \epsilon_i$, $\epsilon_i \sim \mathcal{N}(0, \sigma^2)$ for $i = 1, \ldots, n$, where $\mathcal{N}(a, b)$ denotes a Gaussian distribution with mean a and variance b, and $N(\cdot | a, b)$ will denote its density.

Bayesian (nonparametric) regression [39] consists of specifying a (nonparametric) prior distribution over the function f and conditioning on the observations to obtain a posterior distribution on f. This prior is often chosen to be some (real-valued) stochastic process $g : \mathcal{X} \times \Omega \to \mathbb{R}$ where $(\Omega, \Sigma, \mathbb{P})$ is some probability space. These can be thought of as a random function since $\forall \omega \in \Omega$, $g(\cdot; \omega)$ will be a function, and $\forall x \in \mathcal{X}$, $g(x; \cdot)$ will be a random variable.

A common choice of prior is a GP [89]. A GP is fully determined by its mean function $\mu : \mathcal{X} \to \mathbb{R}$ and covariance function (or kernel) $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$, and hence often denoted by $\mathcal{GP}(\mu, k)$. Due to conjugacy properties of the Gaussian distribution, the posterior f after conditioning on (X, y) is once again a GP with posterior mean $\tilde{\mu}(x) = \mu(x) + k_{x,X}(k_{X,X} + \sigma^2 I)^{-1}(y - \mu_X)$ and covariance $\tilde{k}(x, x') = k(x, x') - k_{x,X}(k_{X,X} + \sigma^2 I)^{-1} k_{X,x'}$, where $k_{X,X} \coloneqq (k(x_i, x_j))_{i,j=1}^n$, $k_{x,X} \coloneqq (k(x, x_1), \ldots, k(x, x_n))$, $k_{X,x} = k_{x,X}^\top$, $\mu_X \coloneqq (\mu(x_1), \ldots, \mu(x_n))^\top$ and I is the $n \times n$ identity matrix. The properties of GP posteriors are inherited from μ and k. Whenever these are continuous, the posterior mean will also be continuous. This is the case for the most common choices of kernels, such as the Gaussian or Matérn kernels with sufficient smoothness. In those cases, discontinuities would have to be inputted manually or inferred from data [17], making the modelling of functions with a large number of discontinuities a challenge for most GP models.

In this paper, we consider instead models based on tree structures. Popular examples include Bayesian CART [21], dynamic regression trees [104] and Mondrian forests [57]. We will focus on BART [14; 46] due to its strong empirical performance and theoretical results. BART is a model which consists of a combination of regression trees. A *regression tree* is any step function $g_{\mathcal{T}, \beta} : \mathcal{X} \to \mathbb{R}$ with K leaves/partitions: $g_{\mathcal{T}, \beta}(x) \coloneqq \sum_{k=1}^K \beta_k \mathbb{1}_{\chi_k}(x)$. Here, we denote by $\mathcal{T} \coloneqq \{\chi_k\}_{k=1}^K$, where $\chi_k \subset \mathcal{X}$, the partition of the domain, and by $\beta \coloneqq (\beta_1, \ldots, \beta_K)^\top \in \mathbb{R}^K$ the leaf values. The function $\mathbb{1}_{\chi_k} : \mathcal{X} \to \mathbb{R}$ is an indicator function taking value 1 whenever $x \in \chi_k$ and 0 otherwise.

Similarly to decision trees and random forest, a single tree comprising of a large number of leafs might overfit to the training data. As a result, it is common to use an ensemble of shallow trees. We will call T-*additive regression tree* any function which takes the form of a sum of regression trees: $g_{\mathcal{E}, \mathcal{B}}(x) \coloneqq \sum_{t=1}^T g_{\mathcal{T}_t, \beta_t}(x)$, where $\mathcal{E} \coloneqq \{\mathcal{T}_t\}_{t=1}^T$ and $\mathcal{B} \coloneqq \{\beta_t\}_{t=1}^T$. Finally, we call *Bayesian additive regression tree* (BART) any distribution on the family of T-additive regression trees. Such a distribution can be constructed by specifying a (prior) distribution on the partition $\mathcal{E}$ and leaf values $\mathcal{B}$. BART is hence a stochastic process whose sample space Ω consists of the product space of K_t-partitions of $\mathcal{X}$ and $\mathbb{R}^{K_t}$. For simplicity, BART is usually restricted to an approximation domain $\mathcal{X} = [0, 1]^d$, but this is not a requirement in full generality.

Denote by p the density of this prior distribution. We will follow the majority of the BART literature [13; 14; 95] and use prior models which factorise in the following way: $p(\mathcal{E}, \mathcal{B}) \coloneqq \prod_{t=1}^T p(\mathcal{T}_t) p(\beta_t | \mathcal{T}_t)$ and $p(\beta_t | \mathcal{T}_t) \coloneqq \prod_{k=1}^{K_t} N(\beta_{t,k} | 0, 1/(16T))$, where T is the number of trees and K_t is the number of leaves in the tth tree. The construction of the distribution on partitions is usually itself done via a tree generating stochastic process; see further details in [14] (and a full description in Appendix A).

Given this prior $\mathbb{P}$ on T-additive trees, we can condition on (X, y) to obtain a posterior $\mathbb{P}_n$ (with density p_n). We focus on regression with i.i.d. Gaussian noise, but several generalisations exist

[83]. The corresponding posterior distribution on parameters will then imply a posterior distribution on T-additive trees. BART is hence another Bayesian model which we can use to approximate the integrand, using for example the posterior mean: $g^n(x) = \mathbb{E}_{\mathbb{P}_n}[f(x)] = \mathbb{E}[f(x)|X,y] = \int_\Omega g_{\mathcal{E},\mathcal{B}}(x)p_n(\mathcal{E},\mathcal{B})d\mathcal{E}d\mathcal{B}$.

Unfortunately, this is not available in closed form and needs to be approximated. This is usually done using Markov chain Monte Carlo (MCMC) methods [14; 81] (although optimisation methods can also be used [44]). At each iteration j, the MCMC algorithm produces T regression trees. More precisely, for the tth regression tree, the algorithm returns a set of leaf values $\{\beta^j_{t,k}\}^{K_{t,j}}_{k=1}$ and a partition of the domain $\{\chi^j_{t,k}\}^{K_{t,j}}_{k=1}$ where $K_{t,j}$ is the number of leaves for tree t at iteration j. This gives a T-additive regression tree which we will denote g^n_j. After m iterations, the MCMC mean is given by:

$$\hat{g}^n(x) = \tfrac{1}{m}\sum_{j=1}^m g^n_j(x) = \tfrac{1}{m}\sum_{j=1}^m \sum_{t=1}^T \sum_{k=1}^{K_{t,j}} \beta^j_{t,k}\mathbb{1}_{\chi^j_{t,k}}(x). \qquad (2)$$

A similar expression can be obtained for the variance of our posterior on f at $x \in \mathcal{X}$. However, when compared to GPs, the entire BART posterior can in fact provide us with a more complex, possibly multimodal, distribution on f.

3   Numerical Integration with Bayesian Additive Regression Trees

The Bayesian approach to numerical integration can be succinctly summarised in the following steps:

1. *Posit a Bayesian prior on the integrand f.* This consists of a stochastic process $g : \mathcal{X} \times \Omega \to \mathbb{R}$ together with a prior measure $\mathbb{P}$ on Ω.
2. *Condition this prior on the data (X,y) to get a posterior on f.* This is given in terms of the posterior measure $\mathbb{P}_n$, and the posterior mean is $g^n := \mathbb{E}_{\mathbb{P}_n}[g] = \mathbb{E}[g|X,y]$.
3. *Obtain the posterior distribution on $\Pi[f]$.* This is given by the marginal distribution obtained by integrating out Π, and a point estimate for the posterior mean $\Pi[f]$, given by $\hat{\Pi}_{\text{BPNI}}[f] = \Pi[g^n]$.

We remark that the posterior distribution on $\Pi[f]$, or even just the estimate $\hat{\Pi}_{\text{BPNI}}[f]$, may not be available in closed-form. In that case, MCMC estimates can be used (see the following section). In the case of GP-BQ with a prior $\mathcal{GP}(\mu, k)$, the posterior on $\Pi[f]$ is available in closed-form [6] and is given by a Gaussian with mean $\mathbb{E}[\Pi[f]|X,y] = \Pi[k_{\cdot,X}](k_{X,X} + \sigma^2 I)^{-1}(y - \mu_X)$ and variance $\mathbb{V}[\Pi[f]|X,y] = \Pi\bar{\Pi}[k] - \Pi[k_{\cdot,X}](k_{X,X} + \sigma^2 I)^{-1}\Pi[k_{X,\cdot}]$, with $\Pi[k_{\cdot,X}] = (\Pi[k(\cdot,x_1)], \ldots, \Pi[k(\cdot,x_n)])$, $\Pi[k_{X,\cdot}] = \Pi[k_{\cdot,X}]^\top$ and $\bar{\Pi}$ denotes the integration functional with respect to the second input. The posterior mean can be expressed as a quadrature rule whose weights depend on X, k and σ. This expression will be available in closed form whenever $\Pi[k(\cdot,x)]$, known as the *kernel mean*, is available in closed form. This is potentially a restrictive condition, but these integrals can themselves be approximated using quadrature, or the kernels could be constructed with this condition in mind [70; 71].

A significant challenge for GP-BQ is the computational cost, which is $\mathcal{O}(n^3)$ due to the need to invert an $n \times n$ matrix. This can possibly be alleviated using fast GP algorithms, but often at the cost of introducing approximations to the posterior. The GP-BQ method is hence more expensive than most standard MC methods, but this should be balanced with significantly faster convergence rates; see [6; 51; 111]. Overall, this method should hence be preferred to MC methods whenever the number of quadrature points n is small or moderate, or when the integrand is itself expensive.

We now derive a new BPNI method via BART. Suppose that g, $\mathbb{P}_0$ are a BART prior (as described in Sec. 2) and we have data (X,y); the posterior on f implies a posterior on $\Pi[f]$. An important distinction with GP-BQ is that the posterior on $\Pi[f]$ is not available in closed-form, and must be approximated. It is also not fully specified by the mean and variance, and may, in fact, be multi-modal. However, the mean of this posterior can be useful if a point estimate for $\Pi[f]$ is required, and the variance can be a useful summary of uncertainty. Expressions for both of these quantities are included in the proposition below for completeness.

Proposition 1 (**MCMC for BART-Int**). *The MCMC approximation of $\Pi[f]$ consists of samples:*

$$s_j = \Pi\left[g^n_j\right] = \sum_{t=1}^T \sum_{k=1}^{K_{t,j}} \beta^j_{t,k}\Pi\left[\mathbb{1}_{\chi^j_{t,k}}\right], \qquad j \in \{1, \ldots, m\}.$$

4

These lead to estimates of the mean $\widehat{\mathbb{E}}_m[\Pi[f]|X,y] = \Pi[\hat{g}^n] = \frac{1}{m}\sum_{j=1}^m s_j$ *and variance* $\widehat{\mathbb{V}}_m[\Pi[f]|X,y] = \frac{1}{(m-1)}\sum_{j=1}^m (s_j - \Pi[\hat{g}^n])^2$.

The posterior samples $\{s_j\}_{j=1}^m$ can only be obtained in a closed form whenever probabilities of the form $\Pi(\chi)$ can be computed for any χ in the partition. This issue corresponds to the well-known issue of intractable kernel means for GP-BQ. The simplest case for which $\Pi(\chi)$ can be computed is when Π is the uniform distribution. If π is available in closed form, one option is to model $f\pi$ as the integrand, in which case the integral is once again against a uniform. However, this makes the specification of a BART prior challenging since very little will usually be known about $f\pi$, and as a result the uncertainty quantification may be unreliable.

Another solution, used in this paper and commonly used for intractable kernel means, is to approximate these probabilities using samples $\{\tilde{x}_i\}_{i=1}^l$ representative of Π. This leads to approximate MCMC samples and an estimate of $\Pi[f]$ of the form:

$$\hat{s}_j^l = \frac{1}{l}\sum_{i=1}^l g_j^n(\tilde{x}_i) = \frac{1}{l}\sum_{i=1}^l \sum_{t=1}^T \sum_{k=1}^{K_{t,j}} \beta_{t,k}^j \mathbb{1}_{\chi_{t,k}^j}(\tilde{x}_i),$$

$$\widehat{\mathbb{E}}_{\mathrm{m}}^l[\Pi[f]|X,y] = \frac{1}{m}\sum_{j=1}^m \hat{s}_j^l. \qquad \widehat{\mathbb{V}}_{\mathrm{m}}^l[\Pi[f]|X,y] = \frac{1}{(m-1)}\sum_{j=1}^m \left(\hat{s}_j^l - \left(\frac{1}{m}\sum_{k=1}^m \hat{s}_k^l\right)\right)^2.$$

Using this approach might be counter-intuitive since it means that some of the uncertainty is not quantified in a Bayesian manner. However, when the integrand f is expensive but a very large number of data points are available (i.e. $n \ll l$), the additional error can be made to be negligible relative to the overall integration error. These settings are common in practice, as will be shown in Sec. 5.

A significant advantage of BART-Int over GP-BQ is the computational cost. For BART, the cost is $\mathcal{O}(Tmn)$ (or $\mathcal{O}(Tm(n+l))$), although the constant depends on the properties of the trees (see [82] for a detailed analysis). When n is large, this can be much cheaper than the $\mathcal{O}(n^3)$ for GP-BQ. We note that $\{x_i\}_{i=1}^n$ can be selected through any quadrature rule. However, when f is computationally expensive, it may be preferable to adaptively select $\{x_i\}_{i=1}^n$ using tools from experimental design and active learning [100]. Recent advances in this direction in numerical integration focus on minimising some distance between Π and $\{x_i\}_{i=1}^n$ (see e.g. kernel herding [11], Stein Variational Gradient Descent [63] and Stein points [10]). All of these methods could be combined with our BART-Int estimators, but such an approach may be sub-optimal in the sense that our objective is not to approximate Π, but only $\Pi[f]$. We propose instead a method which focuses on improving the fit of the BART posterior mean. Our approach is hence closer to previous active learning strategies for GP-BQ, including the work of [5; 40]. It is summarised in Algorithm 1.

Algorithm 1 Sequential Design for BART-Int

1: **Inputs:** Initial point(s) X (set of size n_{ini}), response(s) y, number of candidate design points S, total number of points n, acquisition criterion J and number of MCMC samples m.
2: **for** $i \in \{n_{\mathrm{ini}}, \ldots, n-1\}$ iterations **do**
3: $\quad$ Obtain m posterior samples (given X^i, y^i) & pick a candidate set $\mathcal{C} = \{c_1, \ldots, c_S\} \subset \mathcal{X}$.
4: $\quad$ Find $c^* = \operatorname{argmax}_{c \in \mathcal{C}} J(c)$ & set $X^{i+1} \leftarrow X^i \cup \{c^*\}$, $y^{i+1} \leftarrow y^i \cup \{y_c^*\}$, $y_c^* = f(c^*) + \epsilon^*$.
5: **end for**

The second step consists of comparing the suitability of each point in $\mathcal{C}$ according to some criterion $J : \mathcal{X} \rightarrow \mathbb{R}$, then adding the best point to our design set. At iteration i, we propose to use

$$J(c) = \widehat{\mathbb{V}}_m[f(c)\pi(c)|X^i,y^i], \tag{3}$$

which leads to selecting points where the uncertainty in f is highest. Alternatively, one could choose c which minimises the posterior variance: $J(c) = -\widehat{\mathbb{V}}_m[\Pi[f]|X^i \cup \{c\}, y^i \cup \{y_c\}]$. However, this would require training a different MCMC sampler for each point in $\mathcal{C}$, which would significantly increase the computational cost.

4 Theoretical Results

We now introduce novel concentration results. All proofs are given in Appendix C. These results hold for BART-Int, but are also of independent interest since they can lead to rates for other BPNI

5

methods. The results will hold for an integrand in a normed subspace $\mathcal{H}$ of $L^2(\Pi) := \{h : \mathcal{X} \rightarrow \mathbb{R}$ s.t. $\|h\|_{L^2(\mathcal{X})} := \int_{\mathcal{X}} h^2(x)\pi(x)dx < \infty\}$. They will depend on the contraction rate of the posterior onto the integrand, as measured in some empirical norm: $\|f\|_n = (\frac{1}{n}\sum_{i=1}^{n} f(x_i)^2)^{1/2}$. Below, we will use (X^n, y^n) to denote a dataset (X, y) of size n.

Theorem 1 (Concentration Bound for BPNI). *Suppose we have a normed space $\mathcal{H} \subseteq L^2(\Pi)$ such that the integrand $f \in \mathcal{H}$. Furthermore, suppose that our BPNI posterior is based on a stochastic process g (jointly measurable over $\mathcal{X} \times \Omega$) and a sequence of data $\{(X^n, y^n)\}_{n\in\mathbb{N}}$ such that $\forall \omega \in \Omega, g(\cdot, \omega) \in \mathcal{H}$ and $\exists N \in \mathbb{N}_+$ such that:*

A1. $\exists\{\varepsilon_n\}_{n\geq N}$ such that $\lim_{n\to\infty} \mathbb{P}_n[\|f - g\|_n > A_n\varepsilon_n] = 0$ for any $A_n \to \infty$ as $n \to \infty$.

A2. $\exists\{\gamma_n\}_{n\geq N}$ with $\gamma_n \to 0$ as $n \to \infty$ such that $\sup_{\|h\|_{\mathcal{H}}\leq 1} |\frac{1}{n}\sum_{i=1}^{n} h(x_i) - \Pi[h]| = O(\gamma_n)$.

Then, we have $\lim_{n\to\infty} \mathbb{P}_n[|\Pi[f] - \Pi[g]| > C_n \max(\varepsilon_n, \gamma_n)] = 0$ for any $C_n \to \infty$ as $n \to \infty$.

We now make a number of remarks on the assumptions and statement of the theorem:

1. To guarantee concentration of a BPNI posterior, we require two ingredients: (i) the Bayesian posterior should concentrate on f as n grows, and (ii) the sequence of point sets X^n should give a sequence of quadrature rules which can integrate both f and g^n for any n large enough. The rates at which (i) and (ii) occur then control the overall concentration rate of the BPNI posterior.

2. The assumptions in this theorem are stated in a way such that the order in n at which concentration of the BPNI posterior occurs is explicit. In particular, A1 is a standard assumption in Bayesian nonparametric and guarantees concentration of the posterior on f at rate controlled by $O(\varepsilon_n)$. Similarly, A2 is a common assumption for quadrature rules and guarantees a quadrature rate of $O(\gamma_n)$. The resulting concentration rate of the BPNI posterior is then controlled by $O(\max(\varepsilon_n, \gamma_n))$, the slowest of these two rates.

3. A number of quadrature rules can integrate functions in $L^2(\Pi)$ at a rate $O(n^{-1/2})$ (i.e. $\gamma_n = n^{-1/2}$), but have faster worst-case integration rates for subspaces of $L^2(\Pi)$ which are sufficiently regular. For this reason, we introduced the subspace $\mathcal{H}$ and the conditions that $f \in \mathcal{H}$ and $\forall \omega \in \Omega$, $g(\cdot, \omega) \in \mathcal{H}$, in order to allow for faster rates for γ_n.

4. Results of the form in A1 are available for a variety of Bayesian nonparametric models [39], including GPs [107] and Bayesian neural networks [80]. This is a rapidly evolving field and most results were only derived in recent years. The advantage of Thm. 1 is that any contraction result derived in the future can be plugged in to understand the corresponding BPNI method.

5. Assumption A2 will hold for most reasonable choices of design points. For example, it would hold for V-uniformly ergodic MCMC when $\mathcal{H} = \{h \in L^2(\Pi) : \sup_{x\in\mathcal{X}} |h(x)|/V(x) < \infty\}$ (under regularity conditions given in [93]). Note that Assumption A1 may also depend on the design points, and a discussion of various cases can be found in [49].

6. As for BART-Int, we might sometimes need to approximate the BPNI using samples $\{\tilde{x}_i\}_{i=1}^{l}$. In that case, it is sufficient for $\{\tilde{x}_i\}_{i=1}^{l}$ to satisfy an assumption of the form of A2 for Thm. 1 to hold. In general, it will be possible to take l much larger than n, so we could potentially even allow for slower quadrature rates than $O(\gamma_n)$.

Now that we have discussed our result on concentration of BPNI posteriors, we consider implications for BPNI with tree-based models. Most results are presented for the space of α-Hölder continuous functions, denoted $\mathcal{H}^\alpha$, where $\|f\|_{\mathcal{H}^\alpha} := \sup_{x,y\in\mathcal{X}} |f(x) - f(y)|/\|x - y\|_2^\alpha < \infty$ (when $\alpha = 1$, we recover Lipschitz functions). Several results are of interest:

- [95; 96] showed that Bayesian CART and BART concentrate on targets which are sums of functions in $\mathcal{H}^\alpha$ ($0 < \alpha \leq 1$) that are constant in $d - d_0$ coordinates at a rate $\varepsilon_n = n^{-\frac{\alpha}{2\alpha+d_0}} \log^{\frac{1}{2}} n$. This does not require a-priori knowledge of the coordinates in which the function is constant.

- [62] showed that the soft-BART posterior raised to a fractional power concentrates at the rate $\varepsilon_n = n^{-\frac{\alpha}{2\alpha+d_0}} \log^\beta n + \sqrt{n^{-1}d_0 \log d}$ for any $\beta \geq \alpha(d_0 + 1)/(2\alpha + d_0)$, even when $\alpha > 1$.

These results are not directly comparable with the most of the literature for GP-BQ [6; 51; 52] since these consider the interpolation setting (where $y_i = f(x_i)$). The closest result is in [111], which

presents a result for regression with i.i.d. Gaussian noise and convergence in expectation. The rate in [111] is identical to those above except for for sparse high-dimensional functions where it is slower.

Recent Bernstein-von Mises results for BART presented in [94] (Thm. 5.2) also provide guarantees on the uncertainty quantification provided by the BART-Int posterior distribution on $\Pi[f]$, and show that in the asymptotic limit of $n \to \infty$, the Bayesian credible intervals will coincide with frequentist confidence intervals. Similar results were obtained for sparse deep networks in [109].

Before concluding, we note that any MCMC approximation required to approximate the posterior will also impact the convergence. This is made more precise through the following result.

Proposition 2 (**MCMC Approximation for BPNI**). *Let $\hat{\mathbb{E}}_m[\Pi[g]]$ denote the MCMC approximation of $\mathbb{E}[\Pi[g]]$ where the expectation is with respect to some measure $\mathbb{P}$. Assume $\Pi[g] \in L^2(\mathbb{P})$, where the Markov chain targets $\mathbb{P}$ and is geometrically ergodic and reversible. Then, $\exists \sigma_{MCMC} > 0$ such that*

$$\sqrt{m}|\mathbb{E}[\Pi[g]] - \hat{\mathbb{E}}_m[\Pi[g]]| \to \mathcal{N}(0, \sigma_{MCMC}^2)$$

in distribution as $m \to \infty$.

Geometric ergodicity is a well-studied concept in MCMC theory; see Appendix C.3. Combining Thm. 1 and Prop. 2 gives us some intuition as to how to balance the computational cost of increasing n and m in order to obtain the smallest possible approximation error. Unfortunately, to the best of our knowledge, the ergodicity of MCMC samplers commonly used for BART has not yet been studied.

5 Numerical Experiments

We now illustrate BART-Int and GP-BQ on a range of synthetic problems and a Bayesian survey design problem. We emphasise that our goal is to compare methods which can provide a Bayesian quantification of uncertainty; as a result, it is very much possible that non-Bayesian methods could have better point estimate performance on some of these problems. For BART-Int, we used the default prior settings in dbarts [25], whereas for GP-BQ we used a Matérn kernel whose lengthscale was chosen through maximum likelihood. Further details can be found in Appendix D.

Setup	Method	Cont	Copeak	Disc	Gaussian	Oscil	Prpeak	Step
$d = 1$	BART-Int	1.21	**7.32e-01**	**6.06e-02**	**6.80e-01**	**228**	2.58	**7.85e-03**
$n_{\text{ini}} = 20$	MI	**7.11e-01**	8.99e-01	1.55e-01	8.41e-01	251	1.28	5.90e-02
$n_{\text{seq}} = 20$	GP-BQ	9.38e-01	3.41	1.61e-01	7.42e-01	229	**9.88e-01**	2.88e-02
$d = 10$	BART-Int	**9.10e-04**	43.8	**1.31e-02**	3.37e-03	1.19e-02	1.77e-02	**4.33e-03**
$n_{\text{ini}} = 200$	MI	2.78e-03	**2.12e-01**	1.33e-01	1.18e-02	8.43e-02	3.70e-03	2.46e-02
$n_{\text{seq}} = 200$	GP-BQ	1.52e-03	2.79e-01	1.84e-01	**3.02e-03**	**2.39e-03**	**6.59e-04**	8.36e-03

Table 1: Integration of the six Genz test functions and a step function. The values correspond to the mean absolute percentage error (MAPE) over 20 separate runs.

Genz Functions We start with a standard benchmark set of multi-dimensional integrands proposed by Genz [36] (see Appendix D). This consists of six functions in $\mathcal{X} = [0,1]^d$, where d can be varied to adjust the level of difficulty of the problem. We also add a step function: $f_{\text{step}}(x) = \mathbb{1}_{\{x_1 \in (0.5,1]\}}(x)$, where x_1 is the first component of x. We use $n_{\text{ini}} = 20d$ design points, with $n_{\text{seq}} = 20d$ additional points selected by the sequential scheme in Algorithm 1.

Table 1 shows the mean absolute percentage error (MAPE) of BART-Int ($m = 1500$, $T = 200$ $m = 1000$, $T = 50$, with a burn-in of 1000 and keeping every 5 samples afterwards), GP-BQ and Monte Carlo Integration (MI) for all six Genz functions in $d = 1$ and 10. The MAPE is given by given by $\frac{1}{r}\sum_{t=1}^r |\Pi[f] - \hat{\Pi}_t[f]|/|\Pi[f]|$, where $\hat{\Pi}_t[f]$ for $t = 1, \ldots, r$, are estimates of $\Pi[f]$ for r different initial i.i.d. uniform point sets.

Figure 1: Integration of f_{step} against a uniform, $d = 1$, $n_{\text{ini}} = 20$ and $n_{\text{seq}} = 20$.

BART-Int outperforms GP-BQ when f is not continuous, e.g. the Step functions and the Discontinuous functions. In those cases, the posterior distribution of BART (red dots) is also more concentrated
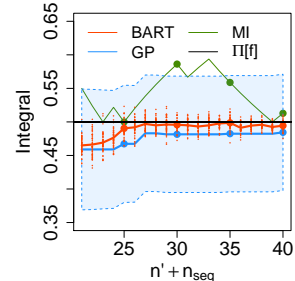
7

around the truth than GP-BQ (whose 95% credible intervals are in shaded blue); see Fig 1. However, GP-BQ is strongest when estimating integrals of smooth continuous in $d = 1$ (see all other integrands). This is to be expected since these functions are smooth enough to be well modelled by the GP (see convergence rates in [111]), while the step-function nature of BART makes it more appropriate for non-smooth functions. Finally, in $d = 10$, BART-Int outperforms GP-BQ for all integrands as it is adaptive to the important features of the integrand, with the largest gains being once again for discontinuous functions.

Computational Complexity   The computational complexity of BART-Int is $\mathcal{O}(Tmn)$, which is much cheaper than the $\mathcal{O}(n^3)$ for GP-BQ with even moderately large n. This behaviour can be seen empirically in Figure 2, noting T and m have been fixed. The computational time of BART-Int is based on the assumption that the tree-based operations are constant, which is reasonable so long as the tree sizes are moderate [82]. Furthermore, it has been shown empirically that the run time of BART is almost independent of the dimension, d, of the data, see e.g. [82] and section 6 of [14]. However, with larger d, it would require a longer burn-in period for the parameter space to be fully explored, and thus the run-time for BART (thus also BART-Int) would increase for the purpose of approximation accuracy.
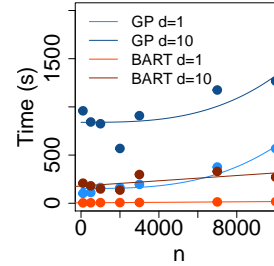

Figure 2: Run-times of GP-BQ & BART-Int for $\Pi[f_{\text{step}}]$ (w.o. seq. design).

Rare Events Simulation   Rare-event simulation is an important tool in risk management. It is used in a range of applications, such as for safety testing autonomous driving systems [77], queuing systems [97] and finance [9].   Traditionally, rare-event simulation is tackled using Monte Carlo methods [97], but we propose to use BART-Int instead.

Let $\ell : \mathbb{R}^d \to \mathbb{R}$ be a measurable function representing our loss. The central quantity of interest here is the probability of obtaining a loss larger than some constant $\gamma \in \mathbb{R}$, which can be written as $p_\gamma = \int_{\mathcal{X}} \mathbb{1}_{\{\ell(x) > \gamma\}}(x) \Pi(dx)$.

We consider a problem of high-dimensional portfolio management adapted from [9]. Suppose we have d loans to obligors, each with value c_i for $i = 1, \ldots, d$. Let x_i denote the financial strain on loan i, and suppose d_i is a thresholds after which default occurs. We assume that the distribution of financial strains is $\text{Exp}(1)$. We can define the portfolio loss as $\ell(x) = \sum_{i=1}^{d} c_i \mathbb{1}_{\{x_i > d_i\}}(x)$. Given a threshold γ, we can compute the probability of making a loss greater than γ as p_γ.

	Method	MAPE	Std. Err.
	BART-Int	1.71e-01	2.56e-02
$d = 5$	MI	1.95e-01	2.29e-02
$n = 2500$	GP-BQ	**1.68e-01**	2.09e-02
	BART-Int	**1.56e-02**	2.35e-03
$d = 10$	MI	9.98e-01	4.47e-04
$n = 5000$	GP-BQ	2.72e-02	5.20e-03
	BART-Int	**8.40e-03**	1.60e-03
$d = 20$	MI	9.94e-01	6.34e-04
$n = 10000$	GP-BQ	2.92e-02	4.90e-03

Table 2: Performance on the portfolio loss event over 20 runs.

We set $\gamma = 2$, $d_i = 0.5i$ and $c_i = 0.2i$. We use the same experimental settings as for the Genz functions, but do not use a sequential design and instead set $n = 500d$. The number of post-burn-in samples is chosen to be 10^4. We can see that for $d = 5$ most of the algorithms perform similarly, but as d increases BART-Int gradually performs better, especially when compared to GP-BQ.

Bayesian Survey Design   In surveys, response variables are collected from a subset of a population based on answers to a set of questions. Survey design concerns sampling strategies to obtain population-representative estimates from these samples. The standard approach is simple or stratified random (Monte Carlo) sampling. Bayesian hierarchical models are also often used in this setting to analyse survey data in order to stabilise estimates and to borrow strength when making sub-population estimates for underrepresented subgroups or locations [35].

We propose a new approach we term "Bayesian Survey Design", using BART-Int to adaptively choose the next individual to survey. Assume we have access to a small set of n_{ini} survey responses with a mixture of continuous and categorical covariates (educational
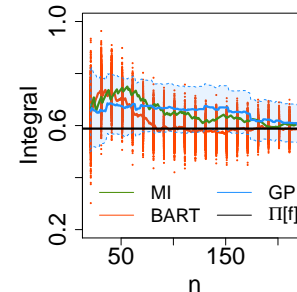

Figure 3: Proportion of individuals with log-salary >10.

8

attainment, age, etc). In addition, we assume there is a much larger set of S individuals for whom demographic variables are known but the response variable is unknown but available for surveying. Such active surveying has previously also been explored in Bayesian active learning [34]. To demonstrate this approach, we use individual-level anonymised census data from the United States [106] and estimate the proportion of the population whose income is higher than around $22,000, or log-income greater than 10. This is equivalent to computing the integral of the following indicator function: $f(x) = \mathbb{1}_{\{\text{log-income}>10\}}(x)$ where the input consists of all other covariates in the survey and Π is the distribution of these covariates in the population.

We consider a universe of $454,816$ possible respondents with 8 demographic covariates. After a one-hot encoding of categorical variables, the dimensionality is $d = 24$. We randomly select our initial set (of size $n_{\text{ini}} = 20$) and candidate set (of size $S = 10,000$). We compute ground truth using all $454,816$ observations and use Algorithm 1 to select $n_{\text{seq}} = 200$ new individuals to survey. As seen in Table 3 and Figure 3, BART-Int outperforms both MI

Method	MAPE	Std. Err.
BART-Int	**4.66e-02**	7.38e-03
MI	5.68e-02	7.01e-03
GP-BQ	9.73e-02	1.44e-02

Table 3: Performance on the Survey Design problem over 20 runs.

and GP-BQ. As expected, the posterior distributions contract to the population mean as n increases. Furthermore, the BART-Int posterior (red dots) is centered around the truth, whereas the 95% credible intervals (blue shaded region) for GP-BQ seem overconfident at the wrong value around $n = 100$.

6 Related Work

Before concluding, we briefly comment on the connections between BART-Int and other tree-based algorithms for integration. Our work is most closely related to GP-BQ, but with tree-based priors. Although trees had not been considered in this literature, the nearest-neighbors method in [64] uses GP kernels which depends on indicator functions at Voronoi partitions, and yield approximations to $\Pi[f]$ which closely resembles that in our paper.

We note that tree-based methods have received significant interest in the literature in numerical integration, but previous work does not come with a Bayesian interpretation. For example, the VEGAS algorithm [59] from the physics literature can be thought of as a tree-based method to reduce the variance of MI. More recent work also includes MCMC and SMC samplers with proposals estimated using trees [28; 42]. Finally, [30] also uses regression trees as surrogates, but again in a non-Bayesian framework.

7 Discussion

We proposed a novel BPNI algorithm which uses BART instead of a GP to model the integrand. BART has several advantages in settings where GPs do not perform well. It is easy to tune, robust to overfitting and enjoys a low computational complexity. It is also a natural model for discontinuous functions and can automatically adapt to sparse high-dimensional functions. We demonstrated these advantages through theoretical results, including contraction of the posterior distribution onto the value of the integral, and through a set of benchmark tests and a survey design experiment. However, it is also important to note that BART-Int did not perform as well as GP-BQ in certain low dimensional and smooth settings. We therefore see BART-Int as a useful addition to the toolset of BPNI which complements, rather than replaces, GP-BQ.

There are a number of potential directions for future research. From an applications-viewpoint, there are interesting parallels to be made with algorithms used to estimate treatment effects in causal inference, and we foresee some possible use for BART-Int in this field. This was discussed in more details in an opinion piece published in [41]. In terms of theory, little is known about the ergodicity of the MCMC sampler used to sample posterior trees. Further work in this area could help practitioners understand the impact of the MCMC approximation on the estimate of the integral for BART-Int. In terms of methodology, while the sequential design approach proposed in this paper uses the (estimated) posterior variance to select new query points, it would be interesting to explore how the use of other acquisition functions may improve the estimates. Similarly, extensions of BART, such as soft-BART [62] and heteroscedastic-BART [83], should also be of practical interest, as they may provide additional advantages over GP-based methods. Finally, some tree-based models can naturally handle categorical variables without the need for one-hot encoding; this could significantly improve the performance for problems such as the Bayesian survey design experiment in this paper.

Broader Impact

Our paper is about numerical integration, a common computational problem in machine learning. It provides an approach to improve the accuracy of such approximations, as well as obtaining credible intervals representing our uncertainty about the value of the integral. The increased accuracy of the method may allow the users to reduce their computational requirements, which may, further down the line, have some impact on mitigating the impact of large computer clusters on climate change.

However, the broader impact of the method will mostly depend on the applications of the algorithms that it will be used to enhance: applications to ethical algorithms will have a positive impact, but application to unethical algorithms will also have a negative impact. For example, in the Bayesian survey design problem studied in the numerical experiments section, we have shown that there is potential for application in predicting population proportions resulting from binary outcome variables. The categorical variables for instance, however, may be unethically used as explanatory variables for certain problems, and this would be an issue depending on how the conclusions are drawn.

References

[1] Wolfram alpha: computational intelligence. Accessed: 2018-01-23.

[2] G. Baio and M. Blangiardo. Bayesian hierarchical model for the prediction of football results. *Journal of Applied Statistics*, 37(2):253–264, 2010.

[3] A. Belhadji, R. Bardenet, and P. Chainais. Kernel quadrature with DPPs. In *Advances in Neural Information Processing Systems*, pages 12927–12937, 2019.

[4] D. M. Blei, A. Kucukelbir, and J. D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112(518):859–877, 2017.

[5] F.-X. Briol, C. Oates, M. Girolami, and M. A. Osborne. Frank-Wolfe Bayesian quadrature: Probabilistic integration with theoretical guarantees. In *Advances in Neural Information Processing Systems*, pages 1162–1170, 2015.

[6] F.-X. Briol, C. J. Oates, M. Girolami, M. A. Osborne, and D. Sejdinovic. Probabilistic integration: A role in statistical computation? (with discussion). *Statistical Science*, 34(1):1–22, 2019.

[7] J. Brouillat, C. Bouville, B. Loos, C. Hansen, and K. Bouatouch. A Bayesian Monte Carlo approach to global illumination. *Computer Graphics Forum*, 28(8):2315–2329, 2009.

[8] U. C. Bureau. AMERICAN COMMUNITY SURVEY 2012-2016 ACS 5-YEAR PUMS FILES, 2018.

[9] J. C. Chan and D. P. Kroese. Improved cross-entropy method for estimation. *Statistics and computing*, 22(5):1031–1040, 2012.

[10] W. Y. Chen, L. Mackey, J. Gorham, F.-X. Briol, and C. J. Oates. Stein points. In *Proceedings of the International Conference on Machine Learning, PMLR 80*, pages 843–852, 2018.

[11] Y. Chen, M. Welling, and A. Smola. Super-samples from kernel herding. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 2012.

[12] H. Chipman, P. Ranjan, and W. Wang. Sequential design for computer experiments with a flexible bayesian additive model. *Canadian Journal of Statistics*, 40(4):663–678, 2012.

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

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

[15] H. A. Chipman, P. Ranjan, and W. Wang. Sequential design for computer experiments with a flexible Bayesian additive model. *Canadian Journal of Statistics*, 40(4):663–678, 2012.

[16] J. Cockayne, C. Oates, T. Sullivan, and M. Girolami. Bayesian probabilistic numerical methods. *SIAM Review*, 61(4):756–789, 2019.

[17] D. Cornford, I. T. Nabney, and C. K. I. Williams. Adding constrained discontinuities to Gaussian process models of wind fields. *Advances in Neural Information Processing Systems*, pages 861–867, 1999.

[18] P. J. Davis and P. Rabinowitz. *Methods of numerical integration*. Courier Corporation, 2007.

[19] A. C. Davison and R. Huser. Statistics of extremes. *Annual Review of Statistics and Its Application*, 2(1):203–235, 2015.

[20] P. Del Moral, A. Doucet, and A. Jasra. Sequential Monte Carlo samplers. *Journal of the Royal Statistical Society Series B: Statisitical Methodology*, 68(3):411–436, 2006.

[21] D. G. T. Denison, B. K. Mallick, and A. F. M. Smith. A Bayesian CART algorithm. *Biometrika*, 85(2):363–377, 1998.

[22] P. Diaconis. Bayesian numerical analysis. *Statistical Decision Theory and Related Topics IV*, 1:163–175, 1988.

[23] J. Dick, F. Y. Kuo, and I. H. Sloan. High-dimensional integration: The quasi-Monte Carlo way. *Acta Numerica*, 22:133–288, 2013.

[24] J. Ding, A. Bashashati, A. Roth, A. Oloumi, K. Tse, T. Zeng, G. Haffari, M. Hirst, M. A. Marra, A. Condon, S. Aparicio, and S. P. Shah. Feature-based classifiers for somatic mutation detection in tumour-normal paired sequencing data. *Bioinformatics*, 28(2):167–175, 2012.

[25] V. Dorie, H. Chipman, and R. McCulloch. dbarts: Discrete Bayesian additive regression trees sampler, 2018.

[26] V. Dorie, J. Hill, U. Shalit, M. Scott, D. Cervone, et al. Automated versus do-it-yourself methods for causal inference: Lessons learned from a data analysis competition. *Statistical Science*, 34(1):43–68, 2019.

[27] R. Durrett. *Probability: theory and examples*. Cambridge University Press, New York, NY, USA, 4th edition, 2010.

[28] J. Felip, N. Ahuja, and O. Tickoo. Tree pyramidal adaptive importance sampling. *arXiv preprint arXiv:1912.08434*, 2019.

[29] M. A. Fisher, C. J. Oates, C. Powell, and A. Teckentrup. A locally adaptive Bayesian cubature method. In *Twenty Third International Conference on Artificial Intelligence and Statistic*, 2020.

[30] T. Foster, C. L. Lei, M. Robinson, D. Gavaghan, and B. Lambert. Model evidence with fast tree based quadrature. *arXiv preprint arXiv:2005.11300*, 2020.

[31] J. H. Friedman. Multivariate adaptive regression splines. *The Annals of Statistics*, 19(1):1–67, 1991.

[32] G.-A. Fuglstad, D. Simpson, F. Lindgren, and H. Rue. Does non-stationary spatial data always require non-stationary random fields? *Spatial Statistics*, 14:505–531, 2015.

[33] J. Gardner, G. Pleiss, K. Q. Weinberger, D. Bindel, and A. G. Wilson. Gpytorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. In *Advances in Neural Information Processing Systems*, pages 7576–7586, 2018.

[34] R. Garnett, Y. Krishnamurthy, X. Xiong, J. Schneider, and R. Mann. Bayesian optimal active search and surveying. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 843–850, 2012.

[35] A. Gelman and J. Hill. *Data analysis using regression and multilevel/hierarchical models*. Cambridge university press, 2006.

[36] A. Genz. Testing multidimensional integration routines. In *Proceedings of the International Conference on Tools, Methods and Languages for Scientific and Engineering Computation*, pages 81–94, 1984.

[37] E. George, P. Laud, B. Logan, R. McCulloch, and R. Sparapani. *Fully nonparametric Bayesian additive regression trees*, pages 89–110. 10 2019.

[38] A. Gessner, J. Gonzalez, and M. Mahsereci. Active multi-information source Bayesian quadrature. In *Uncertainty in Artificial Intelligence*, 2019.

[39] S. Ghosal and A. van der Vaart. *Fundamentals of nonparametric Bayesian inference*. Cambridge University Press, 2017.

[40] T. Gunter, R. Garnett, M. Osborne, P. Hennig, and S. Roberts. Sampling for inference in probabilistic models with fast Bayesian quadrature. In *Advances in Neural Information Processing Systems*, pages 2789–2797, 2014.

[41] P. R. Hahn, J. S. Murray, and C. M. Carvalho. Bayesian regression tree models for causal inference: Regularization, confounding, and heterogeneous effects (with discussion). *Bayesian Anal.*, 15(3):965–1056, 09 2020.

[42] T. E. Hanson, J. V. Monteiro, and A. Jara. The polya tree sampler: Toward efficient and automatic independent metropolis–hastings proposals. *Journal of Computational and Graphical Statistics*, 20(1):41–62, 2011.

[43] W. K. Hastings, editor. *Classification, subtype discovery, numerical recipes: The art of scientific computing (3rd Edition)*. Oxford University Press on behalf of Biometrika Trust, New York, 2007.

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

[45] P. Hennig, M. A. Osborne, and M. Girolami. Probabilistic numerics and uncertainty in computations. *Journal of the Royal Society A*, 471(2179):20150142, 2015.

[46] J. Hill, A. Linero, and J. Murray. Bayesian additive regression trees: a review and look forward. *Annual Reviews in Statistics and Its Applications*, 7(6):1–28, 2020.

[47] J. L. Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 2011.

[48] R. Jagadeeswaran and F. J. Hickernell. Fast automatic Bayesian cubature using lattice sampling. *Statistics and Computing*, 29(6):1215–1229, 2019.

[49] S. Jeong and V. Rockova. The art of bart: On flexibility of bayesian forests. *arXiv preprint arXiv:2008.06620*, 2020.

[50] S. Jiang, H. Chai, J. Gonzalez, and R. Garnett. BINOCULARS for efficient, nonmyopic sequential experimental design. *arXiv:1909.04568*, 2019.

[51] M. Kanagawa and P. Hennig. Convergence guarantees for adaptive Bayesian quadrature methods. In *Advances in Neural Information Processing Systems*, pages 6234–6245, 2019.

[52] M. Kanagawa, B. K. Sriperumbudur, and K. Fukumizu. Convergence analysis of deterministic kernel-based quadrature rules in misspecified settings. *Foundations of Computational Mathematics*, pages 1–40, 2019.

[53] T. Karvonen and S. Särkkä. Fully symmetric kernel quadrature. *SIAM Journal on Scientific Computing*, 40(2):A697–A720, 2018.

[54] T. Karvonen, S. Särkkä, and C. J. Oates. Symmetry exploits for Bayesian cubature methods. *Statistics and Computing*, 29:1231–1248, 2019.

[55] D. P. Kroese, T. J. Brereton, T. Taimre, and Z. I. Botev. Why the monte carlo method is so important today. *Wiley Interdisciplinary Reviews: Computational Statistics*, 6:386–392, 2014.

[56] A. R. L. Bayesian regression trees for high-dimensional prediction and variable selection. *Journal of the American Statistical Association*, 113(522):626–636, 2018.

[57] B. Lakshminarayanan, D. M. Roy, and Y. W. Teh. Mondrian forests for large-scale regression when uncertainty matters. *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, 51:1478–1487, 2016.

[58] F. M. Larkin. Gaussian measure in Hilbert space and applications in numerical analysis. *Rocky Mountain Journal of Mathematics*, 2(3):379–422, 1972.

[59] G. P. Lepage. A new algorithm for adaptive multidimensional integration. *Journal of Computational Physics*, 27(2), 1978.

[60] G. P. Lepage. A new algorithm for adaptive multidimensional integration. *Journal of Computational Physics*, 27(2):192–203, 1978.

[61] G. P. Lepage. Adaptive multidimensional integration: Vegas enhanced. *arXiv preprint arXiv:2009.05112*, 2020.

[62] A. R. Linero and Y. Yang. Bayesian regression tree ensembles that adapt to smoothness and sparsity. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(5):1087–1110, 2018.

[63] Q. Liu and D. Wang. Stein variational gradient descent: A general purpose Bayesian inference algorithm. In *Advances In Neural Information Processing Systems*, pages 2370–2378, 2016.

[64] F. Llorente, L. Martino, V. Elvira, D. Delgado, and J. Lopez-Santiago. Adaptive quadrature schemes for bayesian inference via active learning. *arXiv preprint arXiv:2006.00535*, 2020.

[65] S. Meyn and R. L. Tweedie. *Markov Chains and Stochastic Stability*. Springer, 1993.

[66] H. Mohammadi, P. Challenor, M. Goodfellow, and D. Williamson. Emulating computer models with step-discontinuous outputs using Gaussian processes. *arXiv:1903.02071*, 2019.

[67] T. K. Moon. The expectation-maximization algorithm. *IEEE Signal processing magazine*, 13(6):47–60, 1996.

[68] A. S. Naeem, M. Kegg metabolic reaction network (undirected) data set.

[69] C. J. Oates, J. Cockayne, R. G. Aykroyd, and M. Girolami. Bayesian probabilistic numerical methods in time-dependent state estimation for industrial hydrocyclone equipment. *Journal of the American Statistical Association*, 114(528):1518–1531, 2019.

[70] C. J. Oates, J. Cockayne, F.-X. Briol, and M. Girolami. Convergence rates for a class of estimators based on Stein's identity. *Bernoulli*, 25(2):1141–1159, 2019.

[71] C. J. Oates, M. Girolami, and N. Chopin. Control functionals for Monte Carlo integration. *Journal of the Royal Statistical Society B: Statistical Methodology*, 79(3):695–718, 2017.

[72] C. J. Oates, S. Niederer, A. Lee, F.-X. Briol, and M. Girolami. Probabilistic models for integration error in the assessment of functional cardiac models. In *Advances in Neural Information Processing*, pages 110–118, 2017.

[73] J. Oettershagen. *Construction of optimal cubature algorithms with applications to econometrics and uncertainty quantification*. PhD thesis, Rheinischen Friedrich-Wilhelms-Universität Bonn, 2017.

[74] A. O'Hagan. Bayes–hermite quadrature. *Journal of Statistical Planning and Inference*, 29(3):245–260, 1991.

[75] A. O'Hagan. Bayes–Hermite quadrature. *Journal of Statistical Planning and Inference*, 29(3):245–260, 1991.

[76] A. O'Hagan. Some Bayesian numerical analysis. *Bayesian Statistics*, 4:345–363, 1992.

[77] M. O'Kelly, A. Sinha, H. Namkoong, R. Tedrake, and J. C. Duchi. Scalable end-to-end autonomous vehicle testing via rare-event simulation. In *Advances in Neural Information Processing Systems*, pages 9827–9838, 2018.

[78] M. A. Osborne. *Bayesian Gaussian processes for sequential prediction, optimisation and quadrature*. PhD thesis, Oxford University, 2010.

[79] D. Paulin. Concentration inequalities for Markov chains by Marton couplings and spectral methods. *Electronic Journal of Probability*, 20(79):1–32, 2015.

[80] N. Polson and V. Rockova. Posterior concentration for sparse deep learning. In *Advances in Neural Information Processing Systems*, pages 930–941, 2018.

[81] M. T. Pratola. Efficient Metropolis-Hastings proposal mechanisms for Bayesian regression tree models. *Bayesian Analysis*, 11(3):885–911, 2016.

[82] M. T. Pratola, H. A. Chipman, J. R. Gattiker, D. M. Higdon, R. McCulloch, and W. N. Rust. Parallel Bayesian additive regression trees. *Journal of Computational and Graphical Statistics*, 23(3):830–852, 2014.

[83] M. T. Pratola, H. A. Chipman, E. George, and R. McCulloch. Heteroscedastic BART using multiplicative regression trees. *Journal of Computational and Graphical Statistics*, 09 2017.

[84] S. A. V. W. T. F. B. P. Press, W.H.; Teukolsky, editor. *Classification, subtype discovery, numerical recipes: The art of scientific computing (3rd Edition)*. Cambridge University Press, New York, 2007.

[85] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge University Press, New York, NY, USA, 3 edition, 2007.

[86] M. Quann, L. Ojeda, W. Smith, D. Rizzo, M. Castanier, and K. Barton. Ground robot terrain mapping and energy prediction in environments with 3-D topography. *Proceedings of the American Control Conference*, pages 3532–3537, 2018.

[87] R Core Team. *R: A language and environment for statistical computing*. R Foundation for Statistical Computing, Vienna, Austria, 2018.

[88] C. Rasmussen and Z. Ghahramani. Bayesian Monte Carlo. In *Advances in Neural Information Processing Systems*, pages 489–496, 2002.

[89] C. Rasmussen and C. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[90] C. E. Rasmussen and Z. Ghahramani. Bayesian Monte Carlo. In *Advances in Neural Information Processing Systems*, pages 505–512, 2002.

[91] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning (Adaptive computation and machine learning)*, volume 2. The MIT Press, 2006.

[92] C. Robert and G. Casella. *Monte Carlo statistical methods*. Springer, 2004.

[93] G. O. Roberts and J. S. Rosenthal. General state space Markov chains and MCMC algorithms. *Probability Surveys*, 1(1):20–71, 2004.

[94] V. Rockova. On semi-parametric Bernstein-von Mises theorems for BART. *arXiv:1905.03735*, 2019.

[95] V. Rockova and E. Saha. On theory for BART. In *Proceedings of Machine Learning Research, PMLR 89*, pages 2839–2848, 2019.

[96] V. Rockova and S. van der Pas. Posterior concentration for Bayesian regression trees and forests. *The Annals of Statistics*, 2017.

[97] G. Rubino and B. Tuffin. *Rare event simulation using Monte Carlo methods*. John Wiley & Sons, 2009.

[98] R. Y. Rubinstein and D. P. Kroese. The cross-entropy method: A unified approach to monte carlo simulation, randomized optimization and machine learning. *Information Science & Statistics, Springer Verlag, NY*, 2004.

[99] H. Rue, A. Riebler, S. H. Sørbye, J. B. Illian, D. P. Simpson, and F. K. Lindgren. Bayesian computing with INLA: A review. *Annual Reviews of Statistics and Its Applications*, 4:395–421, 2016.

[100] T. J. Santner, B. J. Williams, and W. I. Notz. *The design and analysis of computer experiments*. Springer, 2nd edition, 2018.

[101] R. Schürer. Parallel High-dimensional integration: quasi-Monte Carlo versus adaptive cubature rules. In V. N. Alexandrov, J. J. Dongarra, B. A. Juliano, R. S. Renner, and C. J. K. Tan, editors, *Computational Science — ICCS 2001*, pages 1262–1271, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[102] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2016.

[103] A. B. Suldin. Wiener measure and its applications to approximation methods. I. *I. Izvestiya Vysshikh Uchebnykh Zavedenii. Matematika*, 1959(6 (13)):145–158, 1959.

[104] M. A. Taddy, R. B. Gramacy, and N. G. Polson. Dynamic trees for learning and design. *Journal of the American Statistical Association*, 106(493):109–123, 2011.

[105] A. L. Teckentrup. Convergence of Gaussian process regression with estimated hyperparameters and applications in Bayesian inverse problems. *arXiv:1909.00232*, 2019.

[106] U.S. Census Bureau. American Community Survey, 2012-2016 ACS 5-Year PUMS Files. Technical report, U.S. Department of Commerce, Janurary 2018.

[107] A. van Der Vaart and H. van Zanten. Information rates of nonparametric Gaussian process methods. *Journal of Machine Learning Research*, 12(Jun):2095–2119, 2011.

[108] A. W. van der Vaart and J. H. van Zanten. Rates of contraction of posterior distributions based on gaussian process priors. *The Annals of Statistics*, 36(3):1435–1463, 2008.

[109] Y. Wang and V. Ročková. Uncertainty quantification for sparse deep learning. *Artificial Intelligence and Statistics*, 2020.

[110] J. T. Wilson, F. Hutter, and M. P. Deisenroth. Maximizing acquisition functions for bayesian optimization, 2018.

[111] G. Wynne, F.-X. Briol, and M. Girolami. Convergence guarantees for Gaussian process means with misspecified likelihoods and smoothness. *arXiv:2001.10818*, 2020.

[112] X. Xi, F.-X. Briol, and M. Girolami. Bayesian quadrature for multiple related integrals. In *International Conference on Machine Learning, PMLR 80*, pages 5369–5378, 2018.

[113] D. S. Yeager, P. Hanselman, G. M. Walton, J. S. Murray, R. Crosnoe, C. Muller, E. Tipton, B. Schneider, C. S. Hulleman, C. P. Hinojosa, et al. A national experiment reveals where a growth mindset improves achievement. *Nature*, 573(7774):364–369, 2019.

[114] H. Zhu, X. Liu, R. Kang, Z. Shen, S. Flaxman, and F.-X. Briol. Bayesian Probabilistic Numerical Integration with Tree-Based Models. *arXiv preprint arXiv:2006.05371*, 2020.

Appendices

This supplementary material is separated into four sections. First, in Appendix A, we provided a detailed description of the BART prior considered in the paper. Secondly, in Appendix B we give a concise description of the posterior sampling procedure. Then, in Appendix C, we provide proofs for the theoretical result in the main text. Finally, in Appendix D, we provide additional details on the experimental setting as well as additional numerical results.

A The Bayesian Additive Regression Trees Prior

Following our definition in Section 3, for a fixed number of regression trees T, a *T-additive regression tree* $g_{\mathcal{E},\mathcal{B}}$ is essentially determined by $(\mathcal{T}_1, \beta_1), \ldots, (\mathcal{T}_T, \beta_T)$, where for each tree t, $\mathcal{T}_t$ is a partition of $\mathcal{X}$ into K_t subsets and β_t is a vector of leaf values. Additionally, σ is the standard deviance of the Gaussian observational noise. This section summarizes the discussion in [14] on how the prior $\mathbb{P}((\mathcal{T}_1, \beta_1), \ldots, (\mathcal{T}_T, \beta_T), \sigma)$ is chosen. An appropriate prior can effectively prevent the individual trees from being overly influential, thus regularizing the fit as an ensemble model. We refer the reader to [14] for more details. Details about our specific choice of hyper-parameters for the priors can be found in Section D.1.

We will assume independence amongst tree components $(\mathcal{T}_t, \beta_t)$ and σ, and amongst each tree's terminal node parameters β_t. The distribution of the sum-of-trees model can hence be simplified as:

$$p((\mathcal{T}_1, \beta_1), \ldots, (\mathcal{T}_T, \beta_T), \sigma) = p(\sigma) \prod_t p(\beta_t | \mathcal{T}_t) p(\mathcal{T}_t)$$
$$p(\beta_t | \mathcal{T}_t) = \prod_k p(\beta_{t,k} | \mathcal{T}_t),$$

where $\beta_{t,k}$ is the k-th terminal node of the t-th tree. We further assume an identical form for each of the component $p(\mathcal{T}_t)$ and for $p(\beta_t | \mathcal{T}_t)$. The prior distributions to be specified are therefore $p(\mathcal{T}_t)$, $p(\beta_t | \mathcal{T}_t)$ and $p(\sigma)$.

The $\mathcal{T}_t$ Prior The trees in BART are k-d trees and have axis-aligned splits. For the prior on $\mathcal{T}_t$, we will use what is commonly known as the "uniform" prior in the literature. This is usually specified in the form of a generative model which consists of three components. Firstly, we specify the probability that a given node of depth $l \in \mathbb{N} \cup \{0\}$ is terminal. This takes the form $p_{\text{split}} = \alpha(1 + l)^{-\beta}$, where $\alpha \in (0, 1)$ and $\beta \in [0, \infty)$. Secondly, a uniform distribution on $\{1, \ldots, d\}$ is used to decide which of the available variables $x_1, \ldots, x_d$ to split on at each interior node. Lastly, a uniform distribution on the set of possible values of that variable is used for the splitting rule assignment. See [13; 25] for more details. Other priors are also possible, such as the Galton-Watson prior [95].

The $\beta_t | \mathcal{T}_t$ Prior For the distribution of leaf values given a tree, we use a Gaussian distribution:

$$p(\beta_t | \mathcal{T}_t) = \prod_{k=1}^{K_t} N\left(\beta_{t,k}; 0, \sigma_\beta^2\right).$$

Let $y_{\min} = \min_i y_i^n$ and $y_{\max} = \max_i y_i^n$. In [14], the authors suggested to rescale y to have zero mean and to ensure that $y_{\min} = -0.5$ and $y_{\max} = 0.5$, then taking $\sigma_\beta = 0.25/\sqrt{T}$. This ensures that the prior on $g(x)$ is within the interval $(y_{\min}, y_{\max})$ with high probability.

The σ Prior For the prior on σ, we use the inverse chi-square distribution $\sigma^2 \sim \nu\lambda/\chi_\nu^2$. This is also a conjugate prior which, again, reduces the required computational effort in the MCMC procedure, as elaborated in [14]. To find the appropriate hyper-parameters, we introduce $q \in (0, 1)$, take $\hat{\sigma}$ of σ as the sample standard deviation of y^n and calibrate for ν and λ such that $\mathbb{P}(\sigma < \hat{\sigma}) = q$. The authors of [14] recommended $(\nu, q) = (3, 0.90)$ as it tends to avoid extremes and we hence follow these recommendations.

Number of Trees T Finally, the number of regression trees T can either be chosen to be the default value $T = 200$, or through cross-validation. [14] further pointed out that, in general, the model performs reasonably well on prediction tasks so long as the value of T is not too small.

B Bayesian Additive Regression Trees Posterior Sampling Procedure

In this appendix, we now describe the Bayesian backfitting MCMC algorithm first introduced in [14], and which is used throughout our experiments. We will make use of the notation: $\mathcal{T}_{-j} = \mathcal{E} \setminus \{\mathcal{T}_j\}$ and similarly $\beta_{-j} = \mathcal{B} \setminus \{\beta_j\}$ for $j = 1, \ldots, T$. Our target posterior is the distribution $\mathcal{E}, \mathcal{B}, \sigma|y^n$. To sample from this posterior, we will make use of a Gibbs sampler, which, at each iteration, draws $(\mathcal{T}_j, \beta_j)|(\mathcal{T}_{-j}, \beta_{-j}, \sigma, y^n)$ sequentially for $t = 1, \ldots, T$, then draws σ from $\sigma|(\mathcal{E}, \mathcal{B}, y^n)$.

Sampling individual trees We note that drawing from $(\mathcal{T}_j, \beta_j)|(\mathcal{T}_{-j}, \beta_{-j}, \sigma, y^n)$ is equivalent to drawing from

$$(\mathcal{T}_j, \beta_j)|R_j, \sigma, \qquad \text{where} \qquad (R_j)_k = y_k - \sum_{t \neq j} g_{\mathcal{T}_t, \beta_t}(x_k),$$

for $k = 1, \ldots, n$, which is the partial residual obtained without the j-th tree. Drawing from this distribution is the same as drawing from the posterior of the residuals regression model with the j-th tree $R_j = g_{\mathcal{T}_j, \beta_j}(x) + \epsilon$, where $\epsilon \sim \mathcal{N}(0, \sigma^2)$. Since the prior distribution $p(\beta_j)$ and the likelihood $p(R_j|\beta_j, \mathcal{T}_j, \sigma)$ are Gaussian, the posterior distribution attains a closed form up to a normalising constant

$$p(\mathcal{T}_j|R_j, \sigma) \propto p(\mathcal{T}_j) \int_{\mathbb{R}^{K_j}} p(R_j|\beta_j, \mathcal{T}_j, \sigma)p(\beta_j|\mathcal{T}_j, \sigma)d\beta_j,$$

where $d\beta_j$ is the Lebesgue measure on $\mathbb{R}^{K_j}$ and K_j is the number of leaf nodes for tree j. Therefore we can now sample $\mathcal{T}_j$ and β_j via the following two-step procedure: first sample from $\mathcal{T}_j|R_j, \sigma$, then sample from $\beta_j|\mathcal{T}_j, R_j, \sigma$.

The draw of $\mathcal{T}_j|R_j, \sigma$ is done via a Metropolis-Hastings algorithm with the following proposal. Given the current tree, grow a terminal node with probability 0.25, prune a pair of terminal nodes with probability 0.25, change a non-terminal node's split rule with probability 0.40, and finally swap a split rule between parent and child with probability 0.10.

Sampling the standard deviation of the observational noise To sample from $\sigma|(\mathcal{E}, \mathcal{B}, y^n)$, we simply draw from the inverse chi-squared distribution defined in Appendix A.

C Proofs of Theoretical Results

In this appendix, we present the proofs of our theoretical results. Section C.1 provides a proof of Proposition 1, Section C.2 provides a proof of Theorem 1, and Section C.3 provides a proof of Proposition 2.

C.1 Proof of Proposition 1

Proof. We begin by deriving the integral of some arbitrary T-additive tree $g_{\mathcal{E}, \mathcal{B}} : \mathcal{X} \times \Omega \to \mathbb{R}$ against the probability measure Π on $\mathcal{X}$. Fix $\omega \in \Omega$. Then, using linearity of integration, we get:

$$\Pi[g_{\mathcal{E}, \mathcal{B}}(\cdot, \omega)] = \int_{\mathcal{X}} g_{\mathcal{E}, \mathcal{B}}(x, \omega)d\Pi(x) = \int_{\mathcal{X}} \sum_{t=1}^{T} \sum_{k=1}^{K} \beta_{t,k} \mathbb{1}_{\chi_{t,k}}(x)d\Pi(x)$$
$$= \sum_{t=1}^{T} \sum_{k=1}^{K} \beta_{t,k} \int_{\mathcal{X}} \mathbb{1}_{\chi_{t,k}}(x)d\Pi(x) = \sum_{t=1}^{T} \sum_{k=1}^{K} \beta_{t,k} \Pi(\chi_{t,k}).$$

We can use this expression to derive the posterior mean and variance for $\Pi[g]$ given the data X, y:

$$\mathbb{E}[\Pi[g_{\mathcal{E}, \mathcal{B}}]|X, y] = \mathbb{E}\left[\sum_{t=1}^{T} \sum_{k=1}^{K} \beta_{t,k} \Pi(\chi_{t,k}) \Big| X, y\right],$$
$$\mathbb{V}[\Pi[g_{\mathcal{E}, \mathcal{B}}]|X, y] = \mathbb{E}\left[\left(\Pi[g_{\mathcal{E}, \mathcal{B}}] - \mathbb{E}\left[\Pi[g_{\mathcal{E}, \mathcal{B}}]|X, y\right]\right)^2 \Big| X, y\right].$$

Using a U-statistic estimate of these quantities based on posterior samples for the parameters leads to our desired result. $\qquad\square$

C.2 Proof of Theorem 1

Proof. Fixing $\omega \in \Omega$ (i.e. fixing a realisation from the stochastic process) and starting with the triangle inequality, we can decouple the integration error into several terms depending on $g(\cdot, \omega)$:

$$|\Pi[f] - \Pi[g(\cdot, \omega)]|$$
$$= \left| \Pi[f] - \Pi[g(\cdot, \omega)] + \frac{1}{n}\sum_{i=1}^{n} g(x_i, \omega) - \frac{1}{n}\sum_{i=1}^{n} g(x_i, \omega) + \frac{1}{n}\sum_{i=1}^{n} f(x_i) - \frac{1}{n}\sum_{i=1}^{n} f(x_i) \right|$$
$$\leq \left| \Pi[f] - \frac{1}{n}\sum_{i=1}^{n} f(x_i) \right| + \left| \Pi[g(\cdot, \omega)] - \frac{1}{n}\sum_{i=1}^{n} g(x_i, \omega) \right|$$
$$+ \left| \frac{1}{n}\sum_{i=1}^{n} g(x_i, \omega) - \frac{1}{n}\sum_{i=1}^{n} f(x_i) \right|. \tag{4}$$

Note that by definition of the worst-case integration error in $\mathcal{H}$, we have that for any $h \in \mathcal{H}$:

$$\left| \Pi[h] - \frac{1}{n}\sum_{i=1}^{n} h(x_i) \right| \leq \|h\|_{\mathcal{H}} \times \sup_{\|h\|_{\mathcal{H}} \leq 1} \left| \Pi[h] - \frac{1}{n}\sum_{i=1}^{n} h(x_i) \right| \tag{5}$$

First, using Equation 5 in Equation 4, then using assumption A2 gives us that whenever $n \geq N$, $\exists B > 0$ such that:

$$|\Pi[f] - \Pi[g(\cdot, \omega)]| \leq B\|f\|_{\mathcal{H}}\gamma_n + B\|g(\cdot, \omega)\|_{\mathcal{H}}\gamma_n + \left| \frac{1}{n}\sum_{i=1}^{n} g(x_i, \omega) - \frac{1}{n}\sum_{i=1}^{n} f(x_i) \right|$$
$$= B(\|f\|_{\mathcal{H}} + \|g(\cdot, \omega)\|_{\mathcal{H}})\gamma_n + \left| \frac{1}{n}\sum_{i=1}^{n} g(x_i, \omega) - \frac{1}{n}\sum_{i=1}^{n} f(x_i) \right|. \tag{6}$$

To tackle the third term, we can use the Cauchy-Schwartz inequality, which states that $\forall u, v \in \mathbb{R}^n$, we have $(\sum_{i=1}^{n} u_i v_i)^2 \leq (\sum_{i=1}^{n} u_i^2)(\sum_{i=1}^{n} v_i^2)$. Taking $u_i = g(x_i, \omega) - f(x_i)$ and $v_i = 1$, we get:

$$\left(\sum_{i=1}^{n} g(x_i, \omega) - \sum_{i=1}^{n} f(x_i) \right)^2 \leq n \sum_{i=1}^{n} (g(x_i, \omega) - f(x_i))^2.$$

Multiplying both sides by n^{-2} and taking square roots, we end up with:

$$\left| \frac{1}{n}\sum_{i=1}^{n} g(x_i, \omega) - \frac{1}{n}\sum_{i=1}^{n} f(x_i) \right| \leq \left(\frac{1}{n}\sum_{i=1}^{n}(g(x_i, \omega) - f(x_i))^2 \right)^{\frac{1}{2}} = \|g(\cdot, \omega) - f\|_n. \tag{7}$$

Plugging in Equation 7 into Equation 6, we get:

$$|\Pi[f] - \Pi[g(\cdot, \omega)]| \leq B(\|f\|_{\mathcal{H}} + \|g(\cdot, \omega)\|_{\mathcal{H}})\gamma_n + \|g(\cdot, \omega) - f\|_n$$
$$\leq B(\|f\|_{\mathcal{H}} + \|g(\cdot, \omega)\|_{\mathcal{H}})\max(\varepsilon_n, \gamma_n) + \|g(\cdot, \omega) - f\|_n,$$

Using this inequality, we have that:

$$\mathbb{P}_n \left(|\Pi[f] - \Pi[g(\cdot, \omega)]| > C_n \max(\varepsilon_n, \gamma_n) \right)$$
$$\leq \mathbb{P}_n \left(B(\|f\|_{\mathcal{H}} + \|g(\cdot, \omega)\|_{\mathcal{H}})\max(\varepsilon_n, \gamma_n) + \|g(\cdot, \omega) - f\|_n \geq C_n \max(\varepsilon_n, \gamma_n) \right)$$
$$\leq \mathbb{P}_n \left(\|g - f\|_n \geq (C_n - B(\|f\|_{\mathcal{H}} + \|g(\cdot, \omega)\|_{\mathcal{H}}))\max(\varepsilon_n, \gamma_n) \right)$$
$$= \mathbb{P}_n \left(\|g - f\|_n \geq A_n \max(\varepsilon_n, \gamma_n) \right)$$
$$\leq \mathbb{P}_n \left(\|g - f\|_n \geq A_n \varepsilon_n \right), \tag{8}$$

where we have taken $A_n = C_n - B(\|f\|_{\mathcal{H}} + \|g(\cdot, \omega)\|_{\mathcal{H}})$. Clearly, since $C_n \to \infty$ as $n \to \infty$ and $B, \|f\|_{\mathcal{H}}, \|g(\cdot, \omega)\|_{\mathcal{H}} < \infty$, we must have $A_n \to \infty$ as $n \to \infty$. Hence, combining Assumption A1 and the upper bound of Equation 8, we have:

$$\lim_{n \to \infty} \mathbb{P}_n[|\Pi[f] - \Pi[g]| > C_n \max(\varepsilon_n, \gamma_n)] = 0$$

which concludes our proof.

$\square$

C.3 Proof of Proposition 2 and Discussion

Proof. Theorem 25 of [93] states that if a Markov chain with stationary distribution $\mathbb{P}$ is reversible and geometrically ergodic, then a central limit theorem holds for any $h \in L^2(\mathbb{P})$. Taking $h = \Pi[g]$, which is in $L^2(\mathbb{P})$ by assumption, concludes the proof. $\square$

Geometric ergodicity is a well-studied concept in MCMC theory which ensures that the chain mixes at a fast rate; see [93], Section 3.4., for a discussion of sufficient conditions, and Section 5.2. for alternative sufficient conditions to obtain a CLT. Stronger results such as convergence almost surely or in probability could also be obtained using stronger conditions on the Markov chain; see for example Theorem 4 of [93] or Theorem 17.0.1 of [65]. Finally, all of the results aforementioned hold in the asymptotic setting where the number of MCMC samples $m \to \infty$. However, finite m results could be obtained using concentration inequalities; see for example [79].

D Additional Numerical Experiments

In this appendix, we provide additional details on the numerical experiments in the paper including the Genz functions in Section D.1, the step function in Section D.2 and the Bayesian survey design experiment in Section D.3.

Figure 4 provides a summary of the algorithm. We first observe some data pairs $\{(x_i, y_i)\}_{i=1}^n$ and then obtain the posterior on , which can be approximated by m MCMC samples. We then integrate each of the samples and then take the mean to obtain an estimate of the integral $\hat{\Pi}_{\text{BPNI}}[f]$.
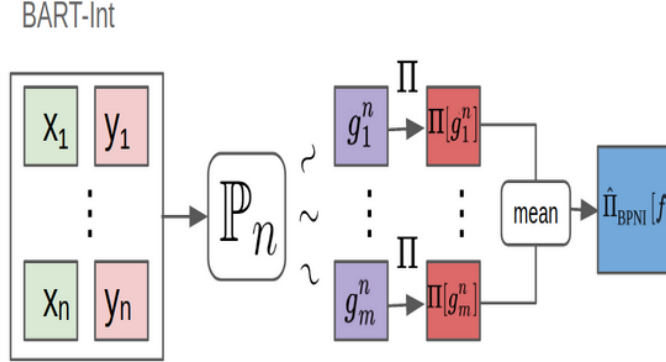

Figure 4: Bayesian Probabilistic Numerical Integration using Bayesian Additive Regression Trees (BART-Int). Here, Π could either be known or replaced with an estimate $\hat{\Pi}$.

Our code relies on `gpytorch` [33] for kernel hyper-parameter (lengthscale) tuning for the GP, and `dbarts` [25] as backend for implementing BART-Int. We use the Imperial College London High Performance Computing and the Department of Mathematics NextGen High Performance Computing servers to conduct our experiments. Our code is available on GitHub `https://github.com/ImperialCollegeLondon/BART-Int` and is subject to spontaneous maintenance.

D.1 Genz Integrand Families

The Genz functions [36] were taken from `http://www.sfu.ca/~ssurjano`, and are presented in Table 4. They have two sets of parameters — d "ineffective" parameters u and d "effective" parameters a which vary the level of difficulty. We use the default setting of $u = (0.5, \ldots, 0.5)^\top$ and scale a suitably as the dimension increases to ensure numerical stability. Specifically, this is done by bounding the L_1-norm of a so that numerical stability is obtained (see [101] for details). As ground truth, we analytically compute the integrals for these Genz test functions, which are again given in Table 4. We compare the performance of BART-Int for each function and make comparisons with two baselines: Monte Carlo integration (MI) and GP-BQ. Following the literature, we choose Π to be the uniform distribution on $[0, 1]^d$.

For BART-Int, we used an MCMC sampler described in Appendix B with a burn-in of 1000 samples and took 5000 samples afterwards. These were then thinned by keeping every 5 samples. This led to $m = 1000$. For the BART model, we used $T = 50$ trees, and the pair $(\alpha, \beta) = (0.95, 2)$ for the terminating probability (see Appendix A for further details). We set $\sigma = 0.1$ to calibrate its inverse-chi-squared prior [25] due to our knowledge that there is no observation noise, but keep it to be non-zero to preserve the statistical properties of BART. For the rest of the hyper-parameters, we used the default setting from `dbarts`. Note that we have applied very little tuning to the fitting of BART.

For GP-BQ, we used a prior mean $\mu(x) = 0$ and the Matérn kernel with smoothness $3/2$:

$$k(x, y) = \left(1 + \frac{\sqrt{3}\|x-y\|_2}{\rho}\right) \exp\left(-\frac{\sqrt{3}\|x-y\|_2}{\rho}\right),$$

where $\| \cdot \|_2$ is the Euclidean norm. The parameter ρ is called the lengthscale, which was selected by maximising the marginal likelihood. To compute the kernel means, we used a MI estimate with $l = 10^6$ randomly sampled points from Π.

All of our results are presented in the main text. To complement these, we show the empirical distribution of the number of leaf nodes of the BART-Int method for each function in Figure 5. Recall that we chose the hyper-parameter values of $(\alpha, \beta) = (0.95, 2)$ for the prior on trees, which guarantees that trees with $1, 2, 3, 4$ and ≥ 5 terminal nodes receive prior probability of $0.05, 0.55,$ $0.28, 0.09$ and 0.03 respectively. As we can see, the posterior distribution of number of leaves per tree varies across target functions, demonstrating that BART is able to adapt to the target function.

For fixed targets, we see very little difference between the distribution for $d = 1$ and $d = 10$. This is sub-optimal since, as mentioned in [96], the optimal number of leaves is $\mathcal{O}\left(n^{\frac{d}{2\alpha+d}}\right)$ where α is the Hölder smoothness of the target function. For fixed α, this suggests we should take a larger number of leaves in larger dimensions. This suggests further improvements in performance could be obtained by adapting the prior distribution as a function of d. On the other hand, the small number of leaves may also be seen as an advantage from a computational viewpoint.
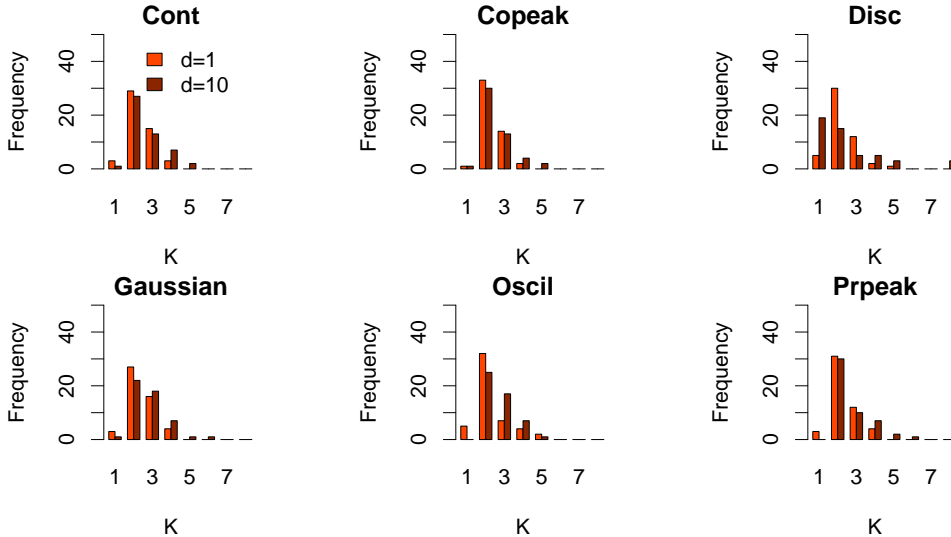

Figure 5: Histogram distribution of the number of leaf nodes, K over $T = 50$ trees for the Continuous, Corner Peak, Discontinuous, Gaussian, Oscillatory and Product Peak Genz function in dimensions $d = 1, 10$.

D.2 Step Function

For the step function, we conducted experiments when integrating the function against either a uniform or a truncated Gaussian distribution. It is clear that with Π being the uniform measure, $\Pi[f] = 0.5$ for all dimensions. When Π is a multivariate Gaussian distribution with mean $x = (0.5, \ldots, 0.5)^\top$ and identity variance matrix truncated to $[0, 1]^d$, the integral is $\Pi[f] = 0.5$ (by symmetry).

We first provide additional results when Π is uniform. The performance of BART-Int in this case is presented in Section 5. Figure 6 illustrates the posterior estimates for both the step function and its integral with BART-Int and GP-BQ. We can see that the posterior distribution of the integral for BART is more concentrated around the value of the true integral than the GP. We can also see that the GP has trouble estimating the discontinuity at $x = 0.5$. Finally a disadvantage of BART-Int is that we see that uncertainty for both algorithms enlarge at areas where data is not observed, but for the end regions near 0 and 1 BART exhibits lower uncertainty due to its stepwise property. It is true that tree-based algorithms do not perform well for extrapolation tasks and this is quite evidently shown with the uncertainty intervals at the end points.

Family	Integrand	Parameter	Integral
Continous (cont)	$\exp\left(-\sum_{i=1}^d a_i \lvert x_i - u_i\rvert\right)$	$a_i = \dfrac{150}{d^3}$	$\prod_{i=1}^d \dfrac{1}{a_i}\left(2 - \exp(-a_i u_i) - \exp(a_i(u_i - 1))\right)$
Corner Peak (copeak)	$\left(1 + \sum_{i=1}^d a_i x_i\right)^{-(d+1)}$	$a_i = \dfrac{600}{d^3}$	$\left(\dfrac{1}{d!\prod_{i=1}^d a_i}\right) \sum_{k=0}^d \sum_{\substack{I \subset \{1,\ldots,d\} \\ \lvert I\rvert=k}} (-1)^{k+d}\left(1 + \sum_{i=1}^d a_i - \sum_{j\in I} a_j\right)^{-1}$
Discontinuous (disc)	$\begin{cases} 0, & \text{if } x_i > u_i \text{ for } i = 1,\ldots,\min(2, d) \\ \exp\left(\sum_{i=1}^d a_i x_i\right), & \text{otherwise} \end{cases}$	$a_i = \dfrac{10}{d^3}$	$\begin{cases} \prod_{i=1}^d \frac{1}{a_i}\left(\exp(a_i \min(1, u_i)) - 1)\right) & \text{if } d > 1 \\ \frac{1}{a_1}\left(\exp(a_1 u_1) - 1\right) & \text{if } d = 1 \end{cases}$
Gaussian Peak (gaussian)	$\exp\left(-\sum_{i=1}^d a_i^2 (x_i - u_i)^2\right)$	$a_i = \dfrac{100}{d^2}$	$\pi^{\frac{d}{2}} \prod_{i=1}^d \dfrac{1}{a_i}\left(\Phi(\sqrt{2}a_i(1 - u_i)) - \Phi(-\sqrt{2}a_i u_i)\right)$
Oscillatory (oscil)	$\cos\left(2\pi u_1 + \sum_{i=1}^d a_i x_i\right)$	$a_i = \dfrac{110}{d^{5/2}}$	$\dfrac{1}{\prod_{i=1}^d a_i} \sum_{k=0}^d \sum_{\substack{I \subset \{1,\ldots,d\} \\ \lvert I\rvert=k}} (-1)^k h_d\left(2\pi u_1 + \sum_{i=1}^d a_i - \sum_{j\in I} a_j\right)$ where $h_d(\cdot) = \begin{cases} \sin(\cdot), & d \equiv 1 \mod 4 \\ -\cos(\cdot), & d \equiv 2 \mod 4 \\ -\sin(\cdot), & d \equiv 3 \mod 4 \\ \cos(\cdot), & d \equiv 0 \mod 4 \end{cases}$
Product Peak (prpeak)	$\prod_{i=1}^d (a_i^{-2} + (x_i - u_i)^2)^{-1}$	$a_i = \dfrac{600}{d^3}$	$\prod_{i=1}^d a_i \left(\arctan(a_i(1 - u_i)) - \arctan(-a_i u_i)\right)$

Table 4: The Genz functions and their true integrals, defined with respect to a uniform measure on $[0, 1]^d$. The default u-parameter is $u = (0.5, \ldots, 0.5)$ and the a-parameter is chosen such that the order of the integral values is comparable across dimension. Here, $\Phi(x) = \int_{-\infty}^x (2\pi)^{-1/2} \exp(-x^2/2)dx$ refers to the cumulative distribution function of a standard Gaussian random variable.

Furthermore, Figure 7 illustrates the sequentially selected design points for each method. As we can see, both the BART and GP methods adaptively select points in areas not covered by the initial design points, and where the uncertainty about f is hence greatest.
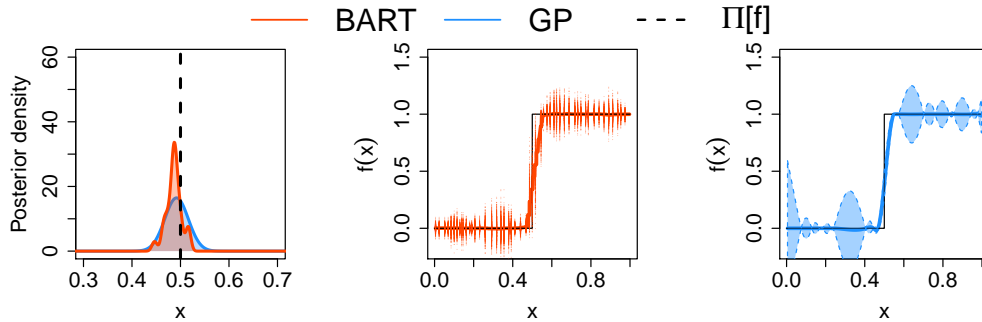

Figure 6: Integration of the step function against a uniform distribution over $[0, 1]$ with BART-Int and GP-BQ with $n = 20$ points. *Left:* The posterior distribution on $\Pi[f]$. *Middle:* The BART posterior distribution on f. Posterior samples for BART are plotted as red points. *Right:* The GP posterior distribution on f. The lines represent the posterior mean, the shaded areas give 95% credible regions for the GP.
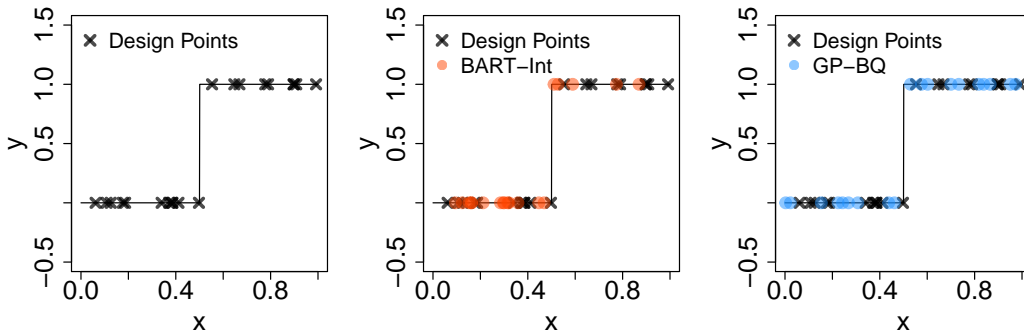

Figure 7: Illustration of adaptive selection of design points through Algorithm 1 on the step function integrated against a uniform measure over $[0, 1]$ with $n_{\text{ini}} = 20$ and $n_{\text{seq}} = 20$.
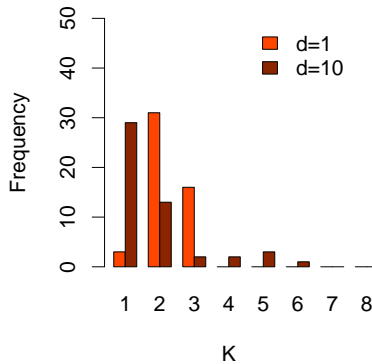

Figure 8: Histogram distribution of the number of leaf nodes, K over $T = 200$ trees for the step function in dimensions $d = 1$ and $d = 10$.

As another toy example, we ran BART-Int, GP-BQ and MI on the step function integrated against a truncated Gaussian measure. We started with $n_{\text{ini}} = 20$ design points and selected sequentially

$n_{\text{seq}} = 20$ points according to the scheme introduced in Algorithm 1. The experimental set-ups for BART-Int and GP-BQ were the same as in the previous experiment. The experiment was repeated with 20 sets of initial points that were sampled randomly and independently from Π. Over these runs, BART-Int achieved the smallest MAPE of 1.81e-02, whereas MI gave 1.15e-01 and GP-BQ yielded 2.83e-02.

D.3 Bayesian Survey Design

To process the dataset in our experiments, we first randomly selected $n_{\text{ini}} = 20$ points as our design points and another $10,000$ points as a candidate set. We then computed the logarithm of the income and created an indicator for each person: 1 if their log income is above 10 and 0 otherwise.

All of the variables education, age, sex, own child, health insurance, marital status, employment and disability are categorical and we hence used a one-hot encoding. The education variable is an ordinal variable but we encoded it as a continuous variable for convenience. We then sampled $n_{\text{seq}} = 200$ new points via sequential design using BART-Int and GP-BQ, and sampled randomly for MI.

As a baseline ground truth, we used all $454,816$ points in the dataset and estimated the mean via MI. We also double-checked by using BART-Int with $10,020$ points from the design and candidate sets, which yielded very similar results.

We repeated this set-up 20 times over different random initial points but the same candidate set. For GP-BQ, we set the lengthscale and σ by maximising the marginal likelihood. For BART-Int, we mostly followed the default settings but used $T = 50$ trees, 1000 burn-in points, 5000 posterior draws after burn-in and kept every 3 draws from the posterior (thinning) so that $m = 1666$.