TECHNOMETRICS
2023, VOL. 00, NO. 0, 1–12
https://doi.org/10.1080/00401706.2023.2257765

Model Mixing Using Bayesian Additive Regression Trees

John C. Yannotty, Thomas J. Santner, Richard J. Furnstahl, and Matthew T. Pratola

The Ohio State University, Columbus, OH

ABSTRACT

In modern computer experiment applications, one often encounters the situation where various models of a physical system are considered, each implemented as a simulator on a computer. An important question in such a setting is determining the best simulator, or the best combination of simulators, to use for prediction and inference. Bayesian model averaging (BMA) and stacking are two statistical approaches used to account for model uncertainty by aggregating a set of predictions through a simple linear combination or weighted average. Bayesian model mixing (BMM) extends these ideas to capture the localized behavior of each simulator by defining input-dependent weights. One possibility is to define the relationship between inputs and the weight functions using a flexible nonparametric model that learns the local strengths and weaknesses of each simulator. This article proposes a BMM model based on Bayesian Additive Regression Trees (BART). The proposed methodology is applied to combine predictions from Effective Field Theories (EFTs) associated with a motivating nuclear physics application. Supplementary materials for this article are available online. Source code is available at *https://github.com/jcyannotty/OpenBT*.

ARTICLE HISTORY
Received December 2022
Accepted September 2023

KEYWORDS
Computer experiments; Effective field theories; Model stacking; Uncertainty quantification

1. Introduction

In statistical learning problems, one often considers a set of plausible models, each designed to explain the system of interest. A common practice is to select a best performing model based on some pre-specified criteria. The ensuing inference for quantities of interest is then carried out using the selected model as if it were the true data generating mechanism. The resulting uncertainty quantification ignores any variability due to the underlying model structure (Draper 1995). The misrepresentation of uncertainties associated with such quantities can ultimately lead to misguided interpretation or inappropriate decisions. Another shortcoming of the typical approach to modeling is that the resulting inference may strongly depend on the selection criteria. In other words, different sets of criteria could lead to noticeably different final models and inferential results. To account for such uncertainties, one may elect to combine information across the set of models in some manner.

Any model set can be classified as $\mathcal{M}$-closed, $\mathcal{M}$-complete, or $\mathcal{M}$-open (Bernardo and Smith 1994). These three categories differ in their underlying assumptions regarding a true model, $\mathcal{M}_\dagger$, and its relation to the model set. The $\mathcal{M}$-closed setting assumes a mathematical representation of $\mathcal{M}_\dagger$ can be formulated and it is included in the model set. In this setting, model selection is appropriate because $\mathcal{M}_\dagger$ can be recovered from the set of models under consideration. The $\mathcal{M}$-complete setting also assumes it is possible to construct $\mathcal{M}_\dagger$, however, it is not included in the model set. For example, an expression for $\mathcal{M}_\dagger$ may exist, however, it may be computationally intensive or intractable compared to the models under consideration. The $\mathcal{M}$-open case assumes the true model may exist, however, a lack of knowledge or resources prevents one from constructing its mathematical representation. Consequently, $\mathcal{M}_\dagger$ is excluded from the model set. This work is motivated by applications in nuclear physics which tend to fall within the $\mathcal{M}$-open class as the underlying truth regarding the physical system may not yet be understood. In such cases, one may desire to leverage the known information about the physical system which is contained in the model set along with experimental data to further understand the nuclear phenomena.

Assume a set of K models are considered when studying a particular system of interest. One approach to account for model uncertainty is to combine the information across these K models. This may involve combining the individual point predictions or probability density functions from each model, usually in some additive manner. Traditional frequentist and Bayesian approaches use global weighting schemes, where each model is weighted by a value intended to reflect overall (global) model performance. For example, a classical global weighting scheme is Bayesian model averaging (BMA) (Raftery, Madigan, and Hoeting 1997), which combines the individual posterior densities from each model using a convex combination. The BMA weights are given by the individual posterior model probabilities, each which can be interpreted as the probability the individual model is the true data generating one. Hence, BMA implicitly assumes the true model is contained within the model set, which renders this method inappropriate outside of the $\mathcal{M}$-closed setting (Bernardo and Smith 1994). More recent Bayesian global weighting schemes

CONTACT John C. Yannotty *yannotty.1@buckeyemail.osu.edu* The Ohio State University, Columbus, OH.
Supplementary materials for this article are available online. Please go to *www.tandfonline.com/r/TECH*.

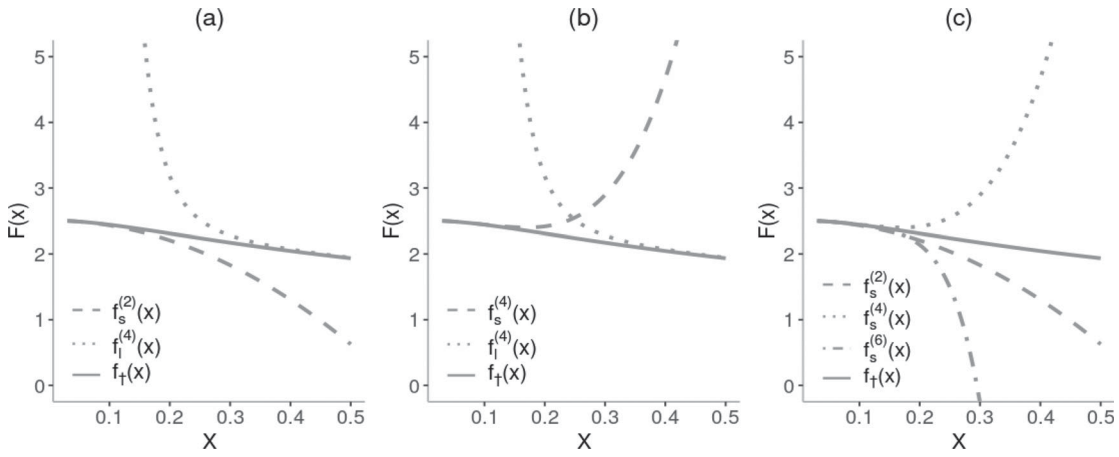

Figure 1. Three different EFT experimental settings. Each panel displays the true physical system (solid) and the mean predictions from the EFTs under consideration (non-solid).

adopt a model stacking approach, where model weights are assigned to minimize a specified posterior expected loss. This decision theory viewpoint of global weighting can be used for combining point predictions (Le and Clarke 2017) or probability densities (Yao et al. 2018). Under some assumptions, stacking methods have been shown to be more appropriate for both the $\mathcal{M}$-open and $\mathcal{M}$-closed settings (Yao et al. 2018).

Though global weighting methods are effective, they still might lead to poor approximations of the true system when the individual model performance is localized. In such a case, one may wish to select a weighting scheme that reflects the localized characteristics of the models by constructing input-dependent weights. With input-dependent weights, one would expect an individual model to receive a higher weight in input regions where it exhibits strong predictive performance, while receiving a weight close to 0 in regions of poor performance. Localized weighting schemes are more appropriate for the $\mathcal{M}$-open or $\mathcal{M}$-complete settings where the true model may be better characterized as a localized mixture of the model set under consideration.

This work is motivated by problems in nuclear physics modeled using a technique known as Effective Field Theory (EFT) (Georgi 1993; Petrov and Blechman 2016; Burgess 2020). EFTs are designed to perform well in a particular subregion(s) of the input domain, yet diverge in the rest of the input domain. Prototypes of such models are the weak and strong coupling finite-order expansions for the partition function of the zero-dimensional ϕ^4 theory presented by Honda (2014). Examples of this problem are shown in Figure 1 where the various dashed and dotted lines represent the mean predictions from a finite-order expansion and the solid line denotes the true physical system. One can see that these models are highly accurate descriptions of the true system in some regions of the domain, yet they are unable to provide a globally accurate model. Most EFT problems fall within the $\mathcal{M}$-open setting, as the true underlying description of the system across the entire domain is unknown and thus is not contained within the model set. Instead, multiple EFTs can be constructed based on the known physics to recover the true system across subsets of the domain.

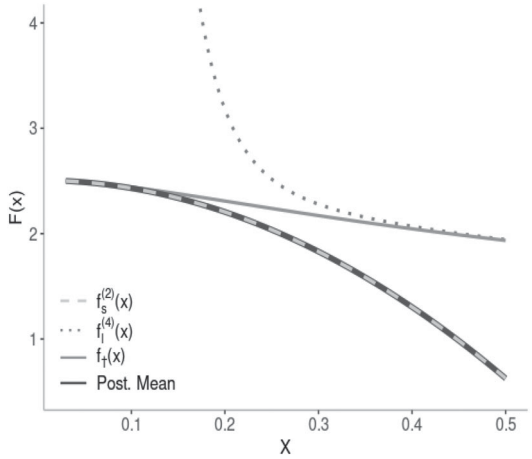

Figure 2. The posterior mean prediction of $f_t(x)$ when applying BMA to the 2nd order weak and 4th order strong coupling expansions.

This poses the question as to how to combine the predictions from multiple EFTs in order to obtain a globally accurate prediction. Various interpolation methods (Honda 2014) exist, however, no data-driven approaches are currently available for EFTs.

To demonstrate why problems falling in the $\mathcal{M}$-open class may not be suited for model averaging schemes, consider applying BMA to the model set involving the two expansions as shown in Figure 1(a). The posterior mean prediction from BMA results in a poor estimate of the true system as shown in Figure 2. Essentially, BMA selects the dashed model rather than leveraging the localized strengths contained in the model set. Given the characteristics of EFTs and the $\mathcal{M}$-open setting associated with these problems, a simple weighted average of the predictions from each model is insufficient for recovering the true physical system. A better approach is to use an input-dependent weighting scheme which leverages the localized behaviors of each model to ascertain appropriate mean prediction and uncertainty quantification. Such an approach falls under the general class of problems known as Bayesian model mixing (BMM) (Yao et al. 2021).

A key challenge in BMM is to define the relationship between the inputs and the weight functions. This work proposes a Bayesian treed model which specifies the weight functions as a sum-of-trees. This representation relies on tree bases which are used to learn the localized model behavior. Additionally, this flexible and nonparametric approach allows the user to avoid having to specify a more restrictive model for the weight functions, such as a generalized linear model. Maintaining the traditional conjugacy properties associated with Bayesian Additive Regression Tree (BART) models, the weight functions are regularized via a multivariate Gaussian prior. The prior is calibrated so that the weight functions prefer the interval $[0, 1]$ without imposing any further constraints. Additionally, this framework includes a simple strategy for incorporating prior information about localized model performance when available. All together, this approach highlights the localized behaviors of the candidate models and yields significant improvements in prediction, interpretation, and uncertainty quantification compared to traditional model averaging methods.

In addition to proposing a novel nonparametric BMM method, this work introduces a new data-driven approach for combining predictions from various EFTs. This is not only important for prediction of the system, but also for the resulting inference. In particular, practitioners can better understand the accuracy of each EFT while also advancing their knowledge about the underlying physical system across areas which are not well explained by the EFTs under consideration.

The remainder of this article is organized in the following manner. Section 2 highlights some relevant work related to model averaging, model mixing, and BART. Section 3 introduces the essential features of EFTs, while Section 4 outlines the specifics of the proposed BART-based framework. Three motivating EFT examples are presented in Section 5. Finally, Section 6 provides a detailed discussion of the results presented throughout this work. The online supplement includes full derivations of the methodology along with additional information regarding EFTs. The source code is available at *https://github.com/jcyannotty/OpenBT*.

2. Background

This section provides an overview of the primary statistical methods discussed throughout this work. Section 2.1 details popular model averaging and model mixing techniques. Section 2.2 summarizes the primary features of Bayesian tree models, which play an integral role in the proposed model mixing approach described in this work.

2.1. Model Averaging and Model Mixing

Methods to address model uncertainty have been widely studied throughout the past few decades. The majority of work in this area combines competing models through either mean or density estimation. In either case, the combined result is generally computed via linear combination of the individual predictive means or densities from the models under consideration. The weights in this linear combination may or may not depend on the inputs for each model and are learned using the set of training data $\mathcal{D} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$. Many frequentist and Bayesian methods exist for estimating the model weights. Popular frequentist approaches such as *stacking* (Breiman 1996) and *model aggregation* (Bunea, Tsybakov, and Wegkamp 2007) estimate the weights by minimizing a specified loss function. Additionally, one may elect to impose constraints such as a nonnegativity or sum-to-one constraint on the weights or apply regularization techniques. Other frequentist approaches estimate the weights using evaluation metrics such as the Akaike information criteria (Burnham and Anderson 1998) or Mallow's CP (Hansen 2007). These methods generally fall under the model averaging regime, as the weights are independent of the model inputs, with the exception of Sill et al. (2009). The remainder of this section reviews popular Bayesian methods in further detail.

Bayesian Model Averaging: A classical approach for combining models $\mathcal{M}_1,\ldots,\mathcal{M}_K$ is *Bayesian Model Averaging* (Raftery, Madigan, and Hoeting 1997). Suppose Q is a quantity of interest. The posterior density of Q is defined by $\pi(Q \mid \mathcal{D}) = \sum_{l=1}^{K} w_l \, \pi(Q \mid \mathcal{D}, \mathcal{M}_l)$, which is a weighted average of the posterior densities with respect to each model. Each weight is defined in terms of its corresponding posterior model probability, that is, $w_l = \pi(\mathcal{M}_l \mid \mathcal{D})$ where

$$\pi(\mathcal{M}_l \mid \mathcal{D}) = \frac{p(\mathcal{D} \mid \mathcal{M}_l)\pi(\mathcal{M}_l)}{\sum_{k=1}^{K} p(\mathcal{D} \mid \mathcal{M}_k)\pi(\mathcal{M}_k)}$$

and $p(\mathcal{D} \mid \mathcal{M}_l)$ is the marginal likelihood of the data with respect to the lth model. Though BMA is useful, it has been criticized for emphasizing a fit to the training data as opposed to out-of-sample prediction, asymptotically selecting a single model (inappropriate in the $\mathcal{M}$-complete and $\mathcal{M}$-open settings, e.g., Figure 2), and being sensitive to prior specification.

Bayesian Mean Stacking: Recent work has extended *stacking* to the Bayesian paradigm as an approach for mean estimation (Clyde and Iversen 2013; Le and Clarke 2017). Given K competing models, the stacked mean for a future observation $\tilde{y}$ at input $\tilde{\boldsymbol{x}}$ is constructed as a linear combination of individual model predictors $E[\tilde{y} \mid \tilde{\boldsymbol{x}}, \mathcal{D}] = \sum_{l=1}^{K} w_l \, f_l(\tilde{\boldsymbol{x}})$, where $E[\tilde{y} \mid \tilde{\boldsymbol{x}}, \mathcal{D}, \mathcal{M}_l] = f_l(\tilde{\boldsymbol{x}})$. When the individual models are unknown, stacking is conducted in a two-step procedure: (i) independently fitting the individual models $\mathcal{M}_l$, $l = 1, \ldots K$, given the set of training data $\mathcal{D}$, and (ii) estimating the weights $\boldsymbol{w} = (w_1, \ldots, w_K)^\top$ for the stacked predictor given the fitted models.

In the first step, each model is fit and their corresponding mean predictions, $\hat{f}_l(\boldsymbol{x}_i)$, are obtained at each of the training points. In practice, cross validation techniques are used to reduce the risk of overfitting the stacked predictor to the training data. In the second step, the coefficient vector $\boldsymbol{w} = (w_1, \ldots, w_K)^\top$ is defined as the minimzer of a specified posterior expected loss. Additionally, one may impose various constraints such as a simplex, nonnegativity, or sum-to-m constraint on the weights (Le and Clarke 2017). Other approaches include regularization via a penalty term or a prior (Breiman 1996; Yang and Dunson 2014).

Bayesian Complete Stacking: Complete Stacking was motivated by the shortcomings of BMA (Yao et al. 2018). This Bayesian

stacking model emphasizes prediction, as the weights are selected to minimize the Kullback-Leibler (KL) divergence between the true predictive density and the stacked predictive density $p(\tilde{y} \mid \tilde{\boldsymbol{x}}) = \sum_{l=1}^{K} w_l \; p(\tilde{y} \mid \tilde{\boldsymbol{x}}, \mathcal{D}, \mathcal{M}_l)$, where $\tilde{y}$ is a future observation with input $\tilde{\boldsymbol{x}}$. Similar to mean stacking, the leave-one-out (LOO) cross validated predictive density can be used in place of $p(\tilde{y} \mid \tilde{\boldsymbol{x}}, \mathcal{D}, \mathcal{M}_l)$ when the individual models are unknown. Given training data, the weights are constrained to a $K-$dimensional simplex S_K and estimated as $\widehat{\boldsymbol{w}} = \text{argmax}_{\boldsymbol{w} \in S_K} \sum_{i=1}^{n} \log \sum_{l=1}^{K} w_l \;\; p(y_i \mid \boldsymbol{x}_i, \mathcal{D}^{(-i)}, \mathcal{M}_l)$, where $\mathcal{D}^{(-i)}$ denotes the training set excluding the pair $(\boldsymbol{x}_i, y_i)$.

Bayesian Hierarchical Stacking: Hierarchical Stacking (Yao et al. 2021) is a model mixing approach which extends Complete Stacking by defining input-dependent weights that are estimated in a fully Bayesian manner. One way to define the weight functions is through a parametric model. First, $K-1$ unconstrained weight functions are defined as,

$$w_l^*(\boldsymbol{x}_i) = \mu_l + \sum_{j=1}^{J} \alpha_{lj} \, g_j(\boldsymbol{x}_i),$$

which depend on the sets of hyperparameters $\{\alpha_{lj}\}$ and $\{\mu_l\}$ along with user-specified basis functions $g_j(\boldsymbol{x}_i)$, where $j = 1, \ldots, J$ and $l = 1, \ldots, K-1$. The Kth function $w_K^*(\boldsymbol{x}_i)$ is set to 0 to serve as a baseline. Then, a softmax transformation is applied to the unconstrained weights in order to confine each model weight to the K-dimensional simplex, namely

$$w_l(\boldsymbol{x}_i) = \frac{\exp\left(w_l^*(\boldsymbol{x}_i)\right)}{\exp\left(w_1^*(\boldsymbol{x}_i)\right) + \cdots + \exp\left(w_K^*(\boldsymbol{x}_i)\right)}, \quad l = 1, \ldots, K.$$

The methods discussed above outline a number of strategies one can take to combine the information across multiple models. In the setting of EFT experiments, the localized nature of the predictions suggests an input-dependent weighing scheme like Bayesian Hierarchical Stacking is more suitable. However, specifying the required basis functions may not be trivial. Thus, the proposed method will adopt the notion of mean stacking within an additive tree basis model to achieve localized weighting in a flexible and nonparametric manner.

2.2. Bayesian Tree Models

Bayesian additive regression trees (BART) have become increasingly popular for modeling complex and high dimensional systems (Chipman, George, and McCulloch 2010). This additive approach involves summing together the predictions made from m trees and is facilitated through a Bayesian backfitting algorithm (Hastie and Tibshirani 2000). Each tree T_j is characterized by its structure, comprised of internal and terminal nodes, along with its associated set of terminal node parameters, M_j. The internal nodes define binary partitions of the input space according to a specified splitting rule. A given node η is defined to be an internal node with probability $p(\eta \text{ is internal}) = \alpha(1+d_\eta)^{-\beta}$ where d_η is the depth of η and α and β are tuning parameters. By construction, this prior penalizes tree complexity and thus ensures each tree maintains a shallow and simple structure. Given d different predictors, $x_1, \ldots, x_d$, splitting rules are of the form $x_v < c$ for $v \in \{1, \ldots, d\}$ and cutpoint c from a discretized subset of $\mathbb{R}$. In the simplest approach, the predictor and cutpoint associated with each splitting rule are randomly selected from discrete uniform distributions. The probabilities associated with the designation of each node along with the splitting rules for internal nodes are used to define the stochastic tree-generating prior for each tree.

The m trees are learned through MCMC, where a slight modification to each structure is proposed at every iteration of the simulation. Generally, such modifications to the tree include birth, death, perturb, or rotate as described by Chipman, George, and McCulloch (1998) and Pratola (2016). Proposals are then accepted or rejected using a Metropolis-Hastings step. To avoid a complex reversible jump MCMC, the algorithm depends on the integrated likelihood, which is obtained by integrating over the terminal node parameters associated with the given tree. A closed form expression for this density can be obtained with conditional conjugate priors for the terminal node parameters.

Given the tree structure, prior distributions can be assigned to each terminal node parameter. In the BART model, the priors ensure each tree explains a small yet different source of variation in the data. For continuous data, BART assigns Gaussian priors to the terminal node parameters. Assuming the data is mean centered, the prior assigned to terminal node parameter μ_{pj} in node η_{pj} is given by $\mu_{pj} \mid T_j \sim N(0, \tau^2)$ where $\tau = (y_{\max} - y_{\min})/(2k\sqrt{m})$ with tuning parameter k. Additionally, a scaled inverse Chi-squared prior is assigned to the variance, that is, $\sigma^2 \sim \nu\lambda/\chi_\nu^2$.

The traditional Bayesian regression tree model can be extended to allow for a more complex structure in the terminal nodes. Existing extensions include linear regression (Chipman, George, and McCulloch 2002; Prado, Moral, and Parnell 2021) and Gaussian processes (Gramacy and Lee 2008). For the setting of model mixing, this work uses a multivariate Gaussian terminal node model.

3. Toward Model Mixing with EFTs

An EFT forms an expansion (or multiple expansions) as a ratio of an input parameter to a physically relevant scale. Computer models implement EFTs as simulators. The theoretical predictions of the physical system are approximations from each simulator plus a discrepancy term, which is designed to account for the remaining unexplained portions of a system. These two components may have specific properties which can be leveraged when working with observational data. This section summarizes these details in the context of EFTs (see also Supplement C).

3.1. Motivating EFT Example

Consider the EFT example where the true physical system is the partition function of the zero-dimensional ϕ^4 theory defined by

$$f_\dagger(x) = \int_{-\infty}^{\infty} \exp\left(-\frac{u^2}{2} - x^2 u^4\right) du, \tag{1}$$

where x denotes the coupling constant (Honda 2014). Two types of finite-order expansions exist for this partition function and

are given by (2) and (3) for n_s or $n_l \geq 1$, namely

$$h_s^{(n_s)}(x) = \sum_{t=0}^{n_s} s_t x^t \quad \text{where}$$

$$s_t = \begin{cases} \frac{\sqrt{2}\Gamma(t+0.5)}{(t/2)!}(-4)^{(t/2)} & t \text{ is even} \\ 0 & t \text{ is odd} \end{cases} \tag{2}$$

$$h_l^{(n_l)}(x) = \sum_{t=0}^{n_l} l_t x^{-t} \quad \text{where}$$

$$l_t = \frac{\Gamma(0.5t + 0.25)}{2t!}\Big(-\frac{1}{2}\Big)^t, \quad t = 0, \ldots, n_l. \tag{3}$$

The weak coupling expansion in (2) is an asymptotic Taylor-like series of order n_s centered about zero. Thus, $h_s^{(n_s)}(x)$ will yield high-fidelity predictions for smaller coupling constants and diverge as the value increases. The reverse behavior is observed for the strong coupling expansion in (3), $h_l^{(n_l)}(x)$, which is convergent. Example predictions of the physical system using these finite-order expansions can be seen in Figure 1 and are discussed in detail in Section 3.2.

The theoretical predictions of the physical system using the weak and strong coupling expansions are expressed using (4) and (5), respectively.

$$f_s^{(n_s)}(x) = h_s^{(n_s)}(x) + \delta_s^{(n_s)}(x) \tag{4}$$

$$f_l^{(n_l)}(x) = h_l^{(n_l)}(x) + \delta_l^{(n_l)}(x). \tag{5}$$

where the truncation errors $\delta_s^{(n_s)}(x)$ and $\delta_l^{(n_l)}(x)$ are modeled with Gaussian processes (GPs) (Santner, Williams, and Notz 2018; Gramacy 2020). As described by Melendez et al. (2019), the parameters in both truncation error models are dependent upon the evaluations of their corresponding finite-order expansions (described in (2) and (3), respectively) over a sparse grid of points. The discrepancy model also depends on physical quantities, $Q(x)$ and $y_{\text{ref}}(x)$, which are chosen based on domain expertise. The relationship between these quantities and the discrepancy are summarized in the supplementary material (Supplement C). When $Q(x)$ and $y_{\text{ref}}(x)$ are unknown, one can alternatively use the error approximation described by Semposki, Furnstahl, and Phillips (2022).

The features present in this example from Honda (2014) are commonly found across the landscape of EFT problems. For instance, the physical system can be expressed as an additive model involving a finite-order expansion and the induced truncation error. The finite-order expansions are designed to provide high-fidelity predictions in specific subregions of the domain. There exists a subregion of the domain where none of the finite-order expansions yield accurate theoretical predictions. All together, this motivating example serves as a prototype for the EFTs that may be encountered in a general experimental setting.

3.2. The Model Set for EFT Experiments

One may encounter various experimental settings when working with EFTs. Such scenarios are introduced in the context of the motivating example presented in Section 3.1. First, consider the most basic case where the model set contains a single EFT. With one EFT, the overall predictive accuracy of the true system is poor, despite the good performance in a localized region. For example, suppose the model set $\mathcal{M}$ contains the 2nd order weak coupling expansion $f_s^{(2)}(x)$. Mean predictions constructed from (2) and (4) are shown by the dashed line in Figure 1(a). Clearly, this model is limited to strong predictive accuracy in only the left subregion of the domain.

When available, one can consider different finite-order approximations of the same EFT. For example, consider the 2nd, 4th, and the 6th order coupling expansions which are shown in Figure 1(c). The three models are very similar for lower coupling constants yet drastically differ in the remainder of the domain. Despite each expansion's poor theoretical predictions, one can still leverage the available information to improve the overall prediction of the physical system. For instance, the 2nd and 6th order expansions (dashed and dashed-dotted) are concave functions while the 4th order expansion (dotted) is convex. This suggests the true physical system lies between the expansions under consideration and can be recovered by re-weighting the corresponding predictions.

A third situation is to consider EFTs centered about different areas of the domain. For example, a model set can contain a finite-order weak coupling expansion (dashed) and the 4th order strong coupling expansion (dotted) as shown in Figures 1(a) and 1(b). The addition of the strong coupling expansion allows for a high-fidelity approximation of the physical system to be considered in the rightmost subregion of the domain. The model set listed in panel (a) implies the true system lies between the two expansions. This is particularly useful in the intermediate range where neither of the EFTs are accurate. Meanwhile, the set in panel (b) presents an interesting case where the physical system lies below both EFTs in the intermediate range. In this case, the information in the observational data can be leveraged to help recover the true system.

In this example, the predictions from the weak coupling expansion degrade slowly compared to those from the strong coupling expansions. Consequently, the weak coupling expansions generally appear to have a better overall predictive performance across the entirety of the domain. When combining these two types of EFTs using global weighting schemes such as BMA, the resulting prediction will favor the weak coupling expansion due to its drastic advantage in the overall model performance. The undesirability of the BMA solution is evident in Figure 2, which demonstrates that BMA effectively matches the 2nd order weak coupling expansion. Hence, a weighting scheme which captures the localized behaviors of each model is preferred in the EFT setting.

The proceeding sections consider a general set of K different EFTs, which are denoted by $f_1(x), \ldots, f_K(x)$. In this motivating example, $f_l(x) = h_l(x) + \delta_l(x)$ where $h_l(x)$ can denote either a weak or strong coupling expansion of order N_l, where $l = 1, \ldots, K$. Meanwhile, $\delta_l(x)$ is the associated truncation error and is modeled by a GP (Supplement C).

3.3. Predictions from EFTs

Prior to model mixing, each of the K EFTs are independently emulated. Without loss of generality, consider the lth EFT denoted by $f_l(\boldsymbol{x})$. It is assumed this EFT is accompanied by a set of simulator runs across a fixed set of inputs $\boldsymbol{x}^c_{l1}, \ldots, \boldsymbol{x}^c_{ln_l}$. Information regarding the design of the computer experiment for each EFT can be found in Melendez et al. (2021). The simulator runs are evaluations of the finite-order expansion, $h_l(\cdot)$, at the specified inputs. Using these runs, one can extract the set of $N_l + 1$ coefficients $c_0(\cdot), \ldots, c_{N_l}(\cdot)$ at each of the fixed inputs. The training set for the lth EFT is then defined by $\mathcal{D}_l = \left\{\left(\boldsymbol{x}^c_{l1}, \boldsymbol{C}(\boldsymbol{x}^c_{l1})\right), \ldots, \left(\boldsymbol{x}^c_{ln_l}, \boldsymbol{C}(\boldsymbol{x}^c_{ln_l})\right)\right\}$ where $\boldsymbol{C}(\cdot)$ denotes the vector of known finite-order coefficients at the specified model input. The resulting coefficients and the set of inputs can differ across the K models, thus, the sets $\mathcal{D}_1, \ldots, \mathcal{D}_K$ will contain different information.

As described in Supplement C, an EFT is fit using the finite-order coefficients to learn the unknown parameters which characterize the GP assigned to the truncation error. This information can be extracted from $\mathcal{D}_l$, which implies the set of field observations is not required to fit each EFT. Consequently, the desired theoretical predictions across the input domain can be obtained without using any of the observational data. The resulting posterior predictive distribution is a Gaussian process, which can be characterized by the corresponding mean and covariance functions as described in Melendez et al. (2019). The predictions for an EFT are then computed through the posterior mean.

4. Bayesian Additive Model Mixing Trees

4.1. Defining a Mixed Model

The proposed BMM model is trained using a set of observational data, $Y_1, \ldots, Y_n$, which are assumed to be independently generated at fixed inputs $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ according to

$$Y_i = f_\dagger(\boldsymbol{x}_i) + \epsilon_i, \quad \epsilon_i \overset{\text{iid}}{\sim} N(0, \sigma^2) \tag{6}$$

where $f_\dagger(\boldsymbol{x}_i)$ represents the true and unknown physical system and σ^2 denotes the constant variance associated with the independent observational errors. Conditional on the theoretical predictions at a given point, $f_1(\boldsymbol{x}_i), \ldots, f_K(\boldsymbol{x}_i)$, the data can be modeled as

$$Y_i \mid \boldsymbol{f}(\boldsymbol{x}_i), \boldsymbol{w}(\boldsymbol{x}_i), \sigma^2 \overset{\text{ind}}{\sim} N\left(\boldsymbol{f}^\top(\boldsymbol{x}_i)\boldsymbol{w}(\boldsymbol{x}_i), \sigma^2\right) \tag{7}$$

where $\boldsymbol{f}(\boldsymbol{x}_i) = \left(f_1(\boldsymbol{x}_i), \ldots, f_K(\boldsymbol{x}_i)\right)^\top$ and $\boldsymbol{w}(\boldsymbol{x}_i) = (w_1(\boldsymbol{x}_i), \ldots, w_K(\boldsymbol{x}_i))^\top$. This formulation is an example of Bayesian mean stacking with an input-dependent weighting scheme. In practice, the predictions from each model are unknown and must be estimated.

The proposed BMM model relies on a two-step approach for combining the predictions across K EFTs. This implies each EFT is first fit independently and the estimated predictions $\hat{f}_l(\boldsymbol{x}_i)$ are obtained for $l = 1 \ldots, K$ and $i = 1, \ldots, n$ prior to learning the weight functions $w_1(\boldsymbol{x}_i), \ldots, w_K(\boldsymbol{x}_i)$. The proposed two-step approach is tailored to EFTs by taking advantage of the sources of data described in Section 3.3 as well as the properties described in Supplement C. Conditional on the estimated predictions, the model for the observational data becomes

$$Y_i \mid \hat{\boldsymbol{f}}(\boldsymbol{x}_i), \boldsymbol{w}(\boldsymbol{x}_i), \sigma^2 \overset{\text{ind}}{\sim} N\left(\hat{\boldsymbol{f}}^\top(\boldsymbol{x}_i)\boldsymbol{w}(\boldsymbol{x}_i), \sigma^2\right)$$

where $\hat{\boldsymbol{f}}(\boldsymbol{x}_i) = \left(\hat{f}_1(\boldsymbol{x}_i), \ldots, \hat{f}_K(\boldsymbol{x}_i)\right)^\top$. The weight functions are then learned using the set of field data. The next section outlines the proposed model mixing scheme which defines the weight functions using Bayesian Additive Regression Trees (BART).

4.2. Model Mixing Using BART

The weight functions $\boldsymbol{w}(\boldsymbol{x}) = (w_1(\boldsymbol{x}), \ldots, w_K(\boldsymbol{x}))^\top$ are modeled using a sum-of-trees

$$\boldsymbol{w}(\boldsymbol{x}_i) = \sum_{j=1}^{m} \boldsymbol{g}(\boldsymbol{x}_i, T_j, M_j), \tag{8}$$

where $\boldsymbol{g}(\boldsymbol{x}_i, T_j, M_j)$ is the K-dimensional output of the jth tree using the set of terminal node parameters, M_j, at the input, $\boldsymbol{x}_i$. This approach defines the weight functions using tree bases which are learned from the data. The amount of flexibility in the weight functions can be controlled by changing the number of trees or tuning the hyperparameters in the prior distributions.

In this application of BART, each terminal node parameter is a K-dimensional vector which is assigned a multivariate Gaussian prior. The parameter is regularized so that each tree accounts for a small amount of variation in the weight functions. For the proceeding statements, let η_{pj} represent the pth terminal on the jth tree and define its corresponding parameter by $\boldsymbol{\mu}_{pj} = (\mu_{pj1}, \ldots, \mu_{pjK})^\top$. Now assume the observations $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_{n_p}, y_{n_p})$ lie in the hyper-rectangle defined by η_{pj}, where n_p is the number of observations assigned to this subregion. The model at each terminal node amounts to fitting a localized Bayesian linear regression with parameter vector $\boldsymbol{\mu}_{pj}$. Due to conditional independence, the likelihood in this node is defined by

$$L(r_1, \ldots, r_{n_p} \mid T_j, \boldsymbol{\mu}_{pj}, \sigma^2) = (2\pi\sigma^2)^{-n_p/2} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n_p}\left(r_i - \hat{\boldsymbol{f}}^\top(\boldsymbol{x}_i)\boldsymbol{\mu}_{pj}\right)^2\right)$$

where $\hat{\boldsymbol{f}}(\boldsymbol{x}_i) = \left(\hat{f}_1(\boldsymbol{x}_i), \ldots, \hat{f}_K(\boldsymbol{x}_i)\right)^\top$ is a vector of mean predictions from each EFT and r_i is the ith residual given by $r_i = y_i - \sum_{q \neq j} \hat{\boldsymbol{f}}^\top(\boldsymbol{x}_i) g(\boldsymbol{x}_i, T_q, M_q)$.

Conditional on the tree structure, T_j, the terminal node parameter, $\boldsymbol{\mu}_{pj}$ is assigned a conjugate multivariate Gaussian prior, namely

$$\boldsymbol{\mu}_{pj} \mid T_j \overset{\text{ind}}{\sim} N_K\left(\boldsymbol{\beta}, \tau^2 \boldsymbol{I}_K\right) \tag{9}$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_K)^\top$ is a K-dimensional mean vector and $\boldsymbol{I}_K$ is the identity matrix. This prior is noninformative in the sense that the mean is fixed regardless of how the input space is partitioned.

In model mixing, each simulator may perform strongly in one subregion of the input space but weakly in another. This belief can be reflected in the prior distribution of $\boldsymbol{\mu}_{pj}$ by allowing

the hyperparameters to depend on the partition of input space assigned to the given terminal node. Thus, an informative prior for $\boldsymbol{\mu}_{pj}$ can be constructed as

$$\boldsymbol{\mu}_{pj} \mid T_j \overset{\text{ind}}{\sim} N_K\left(\boldsymbol{\beta}_{pj}, \tau^2 \boldsymbol{I}_K\right)$$

where $\boldsymbol{\beta}_{pj} = (\beta_{pj1}, \ldots, \beta_{pjK})^\top$. This allows the prior mean to vary depending on the tree partitions and thus reflect some sense of localized model performance. Meanwhile, the assumed covariance structure implies the K vector components $\mu_{pj1}, \ldots, \mu_{pjK}$ are independent apriori.

Both of the proposed priors are conjugate, which is an important choice in BART, as it allows for a closed form expression for the marginal likelihood for the vector of residuals $\boldsymbol{R}_{pj} = (r_1, \ldots, r_{n_p})^\top$, Additionally, the conjugate priors result in closed form expressions for the full conditional distributions of the terminal node parameters and the error variance. The derivations of these distributions are found in the online supplement (Supplement B). In particular, the full conditional distribution for the pth terminal node in T_j is given by

$$\boldsymbol{\mu}_{pj} \mid \boldsymbol{R}_{pj}, T_j, \sigma^2 \overset{\text{ind}}{\sim} N_K\Bigg(\Big(\frac{1}{\sigma^2}\hat{\boldsymbol{F}}_{pj}^\top \hat{\boldsymbol{F}}_{pj} + \frac{1}{\tau^2}\boldsymbol{I}_K\Big)^{-1}\Big(\frac{1}{\tau^2}\boldsymbol{\beta}_{pj} + \frac{1}{\sigma^2}\hat{\boldsymbol{F}}_{pj}^\top \boldsymbol{R}_{pj}\Big), \Big(\frac{1}{\sigma^2}\hat{\boldsymbol{F}}_{pj}^\top \hat{\boldsymbol{F}}_{pj} + \frac{1}{\tau^2}\boldsymbol{I}_K\Big)^{-1}\Bigg)$$

where $\hat{\boldsymbol{F}}_{pj}$ is the $n_p \times K$ design matrix with the ith row vector given by the vector $\hat{\boldsymbol{f}}^\top(\boldsymbol{x}_i)$. The full conditional distribution for σ^2 is a scaled inverse Chi-squared, that is, $\sigma^2 \mid \cdot \sim \nu'\lambda'/\chi^2_{\nu'}$, where

$$\nu' = n + \nu \quad \text{and} \quad \lambda' = \frac{1}{n+\nu}\Big(\sum_{i=1}^{n}\Big(y_i - \hat{\boldsymbol{f}}^\top(\boldsymbol{x}_i)\boldsymbol{w}(\boldsymbol{x}_i)\Big)^2 + \nu\lambda\Big),$$

with ν and λ denoting the prior shape and scale parameters, respectively.

4.3. Calibrating Priors

First consider the prior for the terminal node parameters. The calibration of the hyperparameters differs for the noninformative and informative priors, however, both approaches are designed to ensure that each model weight $w_l(\boldsymbol{x})$ should prefer the interval $[0, 1]$ and be centered at a value within this region. Moreover, the functions $w_1(\boldsymbol{x}), \ldots, w_K(\boldsymbol{x})$ are assumed to be independent apriori at a fixed input. This enables the prior for each weight to be calibrated marginally.

4.3.1. Noninformative Prior

Consider a noninformative prior for the terminal node parameters. In this setting,

$\boldsymbol{\mu}_{pj} \mid T_j \overset{\text{iid}}{\sim} N_K(\boldsymbol{\beta}, \tau^2\boldsymbol{I}_K)$ for the pth terminal node parameter in the jth tree. First, fix $l \in \{1, \ldots, K\}$ and $i \in \{1, \ldots, n\}$ to calibrate the prior for $w_l(\boldsymbol{x}_i)$. Since the terminal node parameters are independent and identically distributed with a diagonal covariance structure, the prior induced on $w_l(\boldsymbol{x}_i)$ is the same for the remaining weight and input combinations. From (8) and (9), the induced prior on the lth model weight is $w_l(\boldsymbol{x}_i) \sim N(m\beta_l, m\tau^2)$. Since it is believed $w_l(\boldsymbol{x}_i) \in [0, 1]$ with high probability, it is plausible to set $m\beta_l = 0.5$. Consequently, $\beta_l = 0.5/m$. Thus, each weight has an equal chance to reach the "extreme" values of 0 or 1 regardless of the input location. The prior standard deviation, τ, can be selected so that $w_l(\boldsymbol{x}_i) \in [0, 1]$ with high probability. To do this, a confidence interval for $w_l(\boldsymbol{x}_i)$ is constructed such that $0 = 0.5 - k\tau\sqrt{m}$ and $1 = 0.5 + k\tau\sqrt{m}$. Subtracting the first equation from the second and solving for τ yields $\tau = 1/2k\sqrt{m}$. This calibration approach is very similar to the one proposed by Chipman, George, and McCulloch (2010). The main difference is due to the context of the problem, as it is believed the weights are predominately contained in an interval $[0, 1]$ rather than the observed range of the data, $[y_{\min}, y_{\max}]$.

4.3.2. Informative Prior

In the informative setting, the prior mean directly depends on the partitions of the input space induced by the given tree, that is $\boldsymbol{\mu}_{pj} \mid T_j \sim N_K(\boldsymbol{\beta}_{pj}, \tau^2\boldsymbol{I}_K)$. This prior is tailored toward EFTs, where the functional variance, $v_l(\boldsymbol{x}_i)$, indicates the severity of the truncation error. A larger variance within a particular subregion of the domain indicates the presence of larger truncation error meaning the EFT provides a poor approximation of the true system.

Given this interpretation of the truncation error variances, one strategy for combining EFTs is precision weighting (Phillips et al. 2021). For example, the precision weight for the lth EFT at $\boldsymbol{x}_i$ is given by

$$\beta_l(\boldsymbol{x}_i) = \frac{1/v_l(\boldsymbol{x}_i)}{1/v_1(\boldsymbol{x}_i) + \ldots + 1/v_K(\boldsymbol{x}_i)}.$$

The precision weight $\beta_l(\boldsymbol{x}_i)$ can be interpreted as an initial guess for the weight function $w_l(\boldsymbol{x}_i)$ for $l = 1, \ldots, K$ and $i = 1, \ldots, n$.

Since the prior of the terminal node parameter changes conditional on the tree structure, each $\boldsymbol{\beta}_{pj}$ is chosen separately from the other terminal node parameters. Given the precision weights for the EFTs and n training points, each component of the prior mean vector, β_{pjl}, is chosen by

$$\beta_{pjl} = \frac{1}{m\sum_{i=1}^{n}\mathbf{1}(\boldsymbol{x}_i \in \eta_{pj})}\sum_{i=1}^{n}\beta_l(\boldsymbol{x}_i)\,\mathbf{1}(\boldsymbol{x}_i \in \eta_{pj}),$$

where $\mathbf{1}(\boldsymbol{x}_i \in \eta_{pj})$ is the indicator that $\boldsymbol{x}_i$ is assigned to the terminal node η_{pj}. A confidence interval for each terminal node parameter can be set to have a length of $1/m$ in order to ensure each tree is a weak learner. This is done by setting $\tau = 1/2km$.

4.3.3. Variance Prior

A conjugate scaled inverse Chi-squared distribution with hyperparameters ν and λ is assigned to the error variance σ^2. To calibrate the prior, first select a value of ν to reflect the desired shape of the distribution. Common values of ν range from 3 to 10. Before selecting a value for λ, one needs an initial estimate of the error variance to help set the prior around a range of plausible values of σ^2. Given the model set and the corresponding point predictions at each of the training points $\hat{\boldsymbol{f}}_l(\boldsymbol{x}_i)$, one can use a lightly data informed prior by setting $\hat{\sigma}^2 = \max_{l=1,\ldots,K}\left\{\min_{i=1,\ldots,n}\left(y_i - \hat{f}_l(\boldsymbol{x}_i)\right)^2\right\}$. Since a common belief is that each model yields accurate approximations of the true

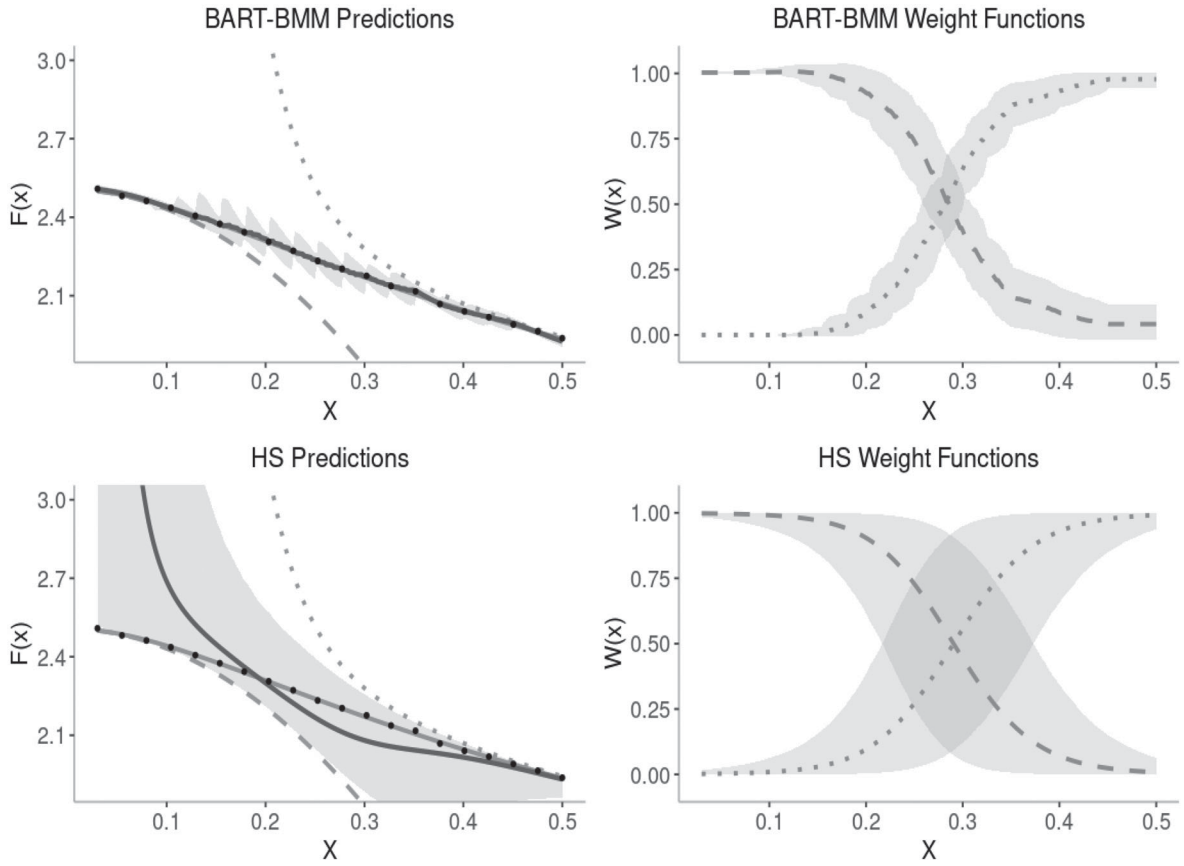

Figure 3. The predicted mean (dark gray) and 95% credible intervals (shaded) when mixing $f_s^{(2)}(x)$ (dashed) and $f_l^{(4)}(x)$ (dotted). Results are obtained from a BART-BMM model with 10 trees and a Hierarchical Stacking model with a linear unconstrained weight function (bottom).

system over some subregion of the domain, one should expect the set of minimum squared differences across the K models will unveil reliable information about the true error variance. Given this information, one strategy is to set $\hat{\sigma}^2$ to be the mean or mode of a $\lambda\nu/\chi^2_\nu$ distribution. The value of λ is then found by solving the resulting equation.

5. EFT Examples

This section applies the proposed model mixing methodology to three different examples. Section 5.1 demonstrates the success of the BART-based mixing approach on two univariate EFT examples, which are introduced in Section 3. A multi-dimensional example is highlighted in Section 5.2 using simulators which are based on Taylor series expansions of a trigonometric function. Though this last example does not involve a true underlying physical system, the model set considers simulators which have similar qualities of EFTs with double expansions (see Burgess 2020). Each example highlights specific features of the proposed BART-based mixing model such as flexible basis functions for the weights and the associated prior regularization.

5.1. Example 1: Mixing Univariate EFTs

This section applies the BART model mixing (BART-BMM) method to various EFTs over a one-dimensional domain. For comparison, Hierarchical Stacking (HS) is also applied to the same set of EFTs. In both EFT examples, 20 observations are independently generated according to

$$Y_i = f_\dagger(x_i) + \epsilon_i, \quad \epsilon_i \sim N(0, \sigma^2)$$

where $i = 1, \ldots, 20$, $\sigma = 0.005$, and $f_\dagger(x)$ is defined in (1). The 20 training points are located at inputs which are evenly spaced over the interval of 0.03 to 0.50. The error standard deviation of 0.005 was selected to mimic a controlled experiment setting. Each EFT model is fit using $n_c = 4$ evaluations of the corresponding finite-order expansion.

5.1.1. Example 1a: Mixing Two EFTs

First consider mixing the EFTs based on the 2nd order weak coupling expansion, $f_s^{(2)}(x)$ and the 4th order strong coupling expansion, $f_l^{(4)}(x)$ as shown in Figure 1(a). The true system $f_\dagger(x)$ lies between both EFTs across the entire domain, hence, a convex combination of the predictions from both EFTs is appropriate for recovering the true system. The BART-BMM model is fit using 10 trees and $k = 5.0$. Meanwhile, the HS unconstrained weight function is defined by $w_1^*(x) = \mu_1 + \alpha_1 x$. The results of the BART-BMM method and HS are shown in Figure 3.

In terms of the root mean squared error (RMSE) between the predicted system and the true $f_\dagger(x)$, the BART-BMM model results in more accurate mean predictions compared to HS, which have RMSE values of 0.0053 and 1.9460, respectively. The RMSE for the HS result is inflated by the diverging mixed prediction in the left portion of the domain. For example, the

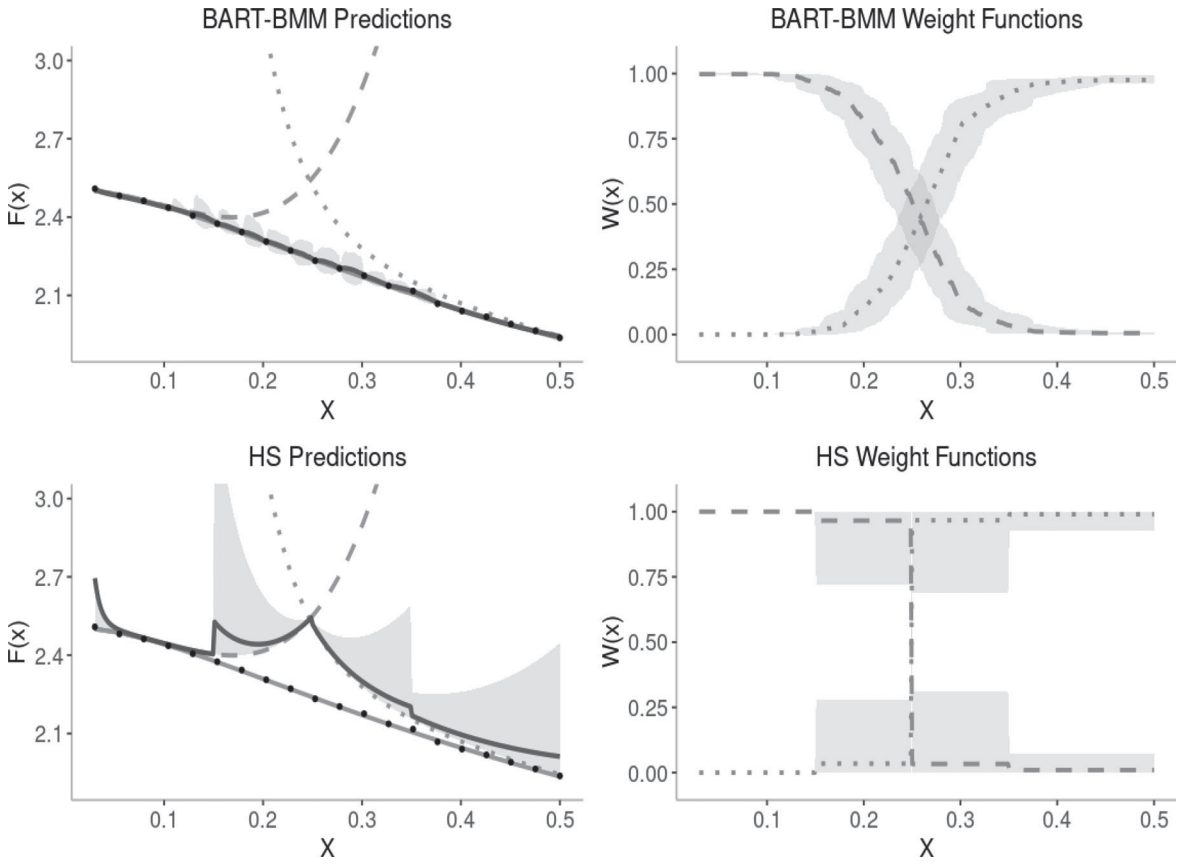

Figure 4. The predicted mean (dark gray) and 95% credible intervals (shaded) when mixing $f_s^{(4)}(x)$ (dashed) and $f_l^{(4)}(x)$ (dotted). Results are obtained from a BART-BMM model with 10 trees (top) and a Hierarchical Stacking model with a piecewise unconstrained weight function (bottom).

RMSE for the HS model over the interval $[0.1, 0.5]$ drops to 0.0717. Additionally, from Figure 3 it is evident BART-BMM results in predictions of $f_\dagger(x)$ which have lower uncertainty than those from HS.

The weight functions in Figure 3 also take similar sigmoid-like shapes, however, the HS solution displays a high degree of uncertainty. The most noticeable difference between the two methods can be seen in the weight function of $f_l^{(4)}(x)$ (dotted). In particular, the curve in the BART-BMM result increases at a quicker rate in the sub-region $[0.3, 0.4]$ compared to the HS result. This slower rate of increase contributes to the poor prediction from HS in this sub-region. Another difference is observed in the region of $[0.03, 0.15]$, as the weight of $f_l^{(4)}(x)$ under the HS approach is near 0, however, it is not small enough to negate the effect of the drastically diverging mean prediction from $f_l^{(4)}(x)$. Meanwhile, the BART-BMM weight is shrunk close to 0 with minimal uncertainty due to the mean estimation objective, which directly re-weights the mean prediction from an individual model, and the lack of a simplex constraint.

Another advantage of BART-BMM is that the weight functions are learned throughout the MCMC via the tree models. This differs from HS, which requires specification of a basis for the unconstrained weights a priori. In this example, one may consider a different basis function, as the specified linear basis appears to be inadequate for ascertaining high-fidelity mean predictions across the entire domain.

5.1.2. Example 1b: Mixing Two Convex EFTs

Now, consider a second model set which is shown in Figure 1(b) and replaces $f_s^{(2)}(x)$ with $f_s^{(4)}(x)$. Both EFTs overestimate $f_\dagger(x)$ in the intermediate range, hence, weights which are confined to a simplex are unable to recover the true system. In this case, a piecewise basis function is assigned to the unconstrained HS weight as shown below,

$$w_1^*(x) = \mu_1 + \alpha_1 \mathbf{1}(x < 0.15) + \alpha_2 \mathbf{1}(0.15 \le x < 0.25) + \alpha_3 \mathbf{1}(0.25 \le x < 0.35).$$

This basis was chosen to roughly reflect the areas where the mean predictions begin to change at differing rates. Other selections of the partitions for a piecewise basis are equally valid.

The BART-BMM and HS results are shown in Figure 4. Once more, the BART-BMM approach outperforms HS in terms of mean prediction, with RMSE values of 0.0057 and 0.1141, respectively. Most notably, the HS solution is unable to accurately predict the true system in the intermediate range of the domain due to the simplex constraint on the model weights. Meanwhile, the BART-BMM approach is able to recover the system across the entirety of the domain due to the prior regularization approach taken with the weights, which does not impose such strict constraints.

In this HS result, it appears the piecewise basis was more effective than the linear basis in terms of predicting the true system in the left and right portion of the domain. This further poses the question of how to select the partitions induced by

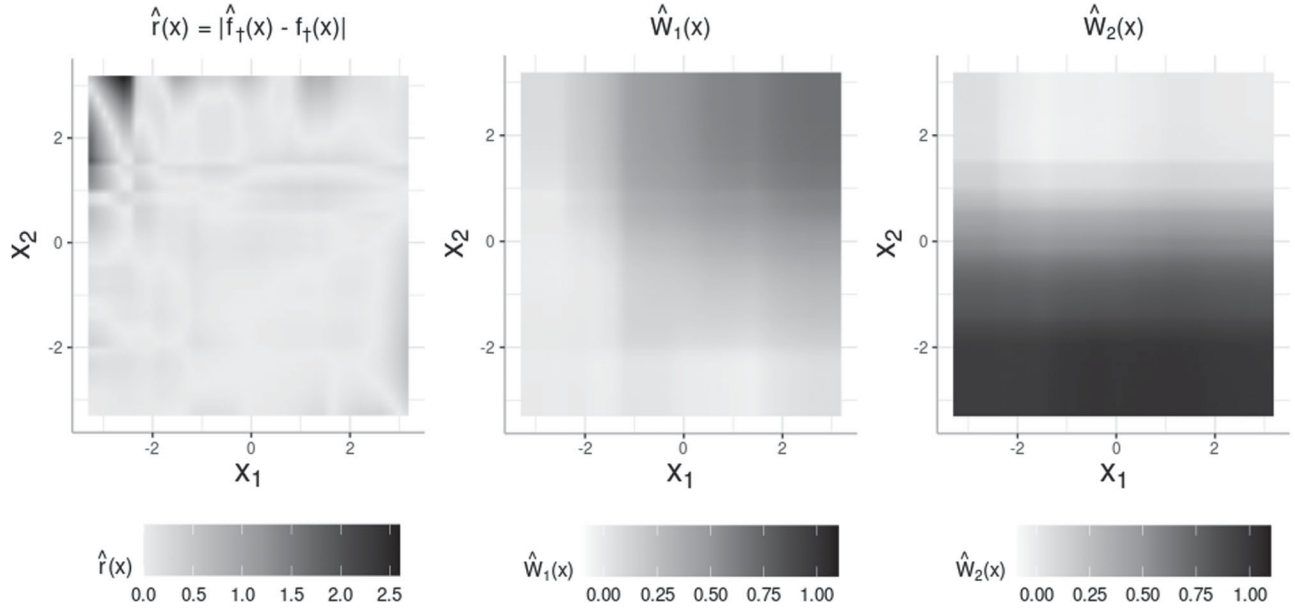

Figure 5. (Left) The mean difference between the predicted system $\hat{f}_\dagger(\boldsymbol{x})$, and the true system $f_\dagger(\boldsymbol{x})$. (Center) The mean weight function for $h_1(\boldsymbol{x})$. (Right) The mean weight function for $h_2(\boldsymbol{x})$.

the piecewise basis, as different choices may lead to drastically different results. This question served as the motivation for defining a BART-based model, which adaptively learns these partitions based on the observational data and the model set.

5.2. Example 2: Multi-Dimensional Mixing

The proposed model mixing approach is also applicable for computer experiments which depend on multi-dimensional inputs. To demonstrate this, consider a two-dimensional problem where the true underlying system is defined by

$$f_\dagger(\boldsymbol{x}) = \sin(x_1) + \cos(x_2),$$

where $\boldsymbol{x} = (x_1, x_2)^\top \in [-\pi, \pi] \times [-\pi, \pi]$. A set of 80 training points are generated from this true system with observational error standard deviation of 0.1. Additionally two candidate models are considered, each with simulators defined in terms of Taylor series expansions of $s(x_i) := \sin(x_1)$ and $c(x_2) := \cos(x_2)$. For this example, the simulators are defined by

$$h_1(\boldsymbol{x}) = \sum_{j=0}^{7} \frac{s^{(j)}(x_1)}{j!}(x_1 - \pi)^j + \sum_{k=0}^{10} \frac{c^{(k)}(x_2)}{k!}(x_2 - \pi)^k$$

$$h_2(\boldsymbol{x}) = \sum_{j=0}^{13} \frac{s^{(j)}(x_1)}{j!}(x_1 + \pi)^j + \sum_{k=0}^{6} \frac{c^{(k)}(x_2)}{k!}(x_2 + \pi)^k$$

where $s^{(j)}(x_1)$ and $c^{(k)}(x_2)$ denote the jth and kth derivatives of $\sin(x_1)$ and $\cos(x_2)$, respectively. Note, the first simulator $h_1(\boldsymbol{x})$ centers both Taylor series expansions about π, hence, it produces relatively accurate predictions of the system in upper right corner of the domain and diverges when moving toward the negative portion of the domain. Meanwhile, the $h_2(\boldsymbol{x})$ is composed of Taylor series expansions centered about $-\pi$ which produces accurate predictions in the negative portion of the domain. One key difference between the simulators is that $h_2(\boldsymbol{x})$ contains a highly accurate approximation of $\sin(x_1)$ across the entire interval $[-\pi, \pi]$ because its corresponding Taylor series expansion is composed of seven nonzero terms. Thus, even though the expansion $\sin(x_1)$ and $\cos(x_2)$ are centered about $-\pi$, one would expect $h_2(\boldsymbol{x})$ to result in accurate predictions of $f_\dagger(\boldsymbol{x})$ across the rectangle $[-\pi, \pi] \times [-\pi, 0]$.

The theoretical predictions from each model $f_1(\boldsymbol{x})$ and $f_2(\boldsymbol{x})$ can be defined using the additive form $f_l(\boldsymbol{x}) = h_l(\boldsymbol{x}) + \delta_l(\boldsymbol{x})$, where $\delta_l(\boldsymbol{x})$ represents the unknown higher-order corrections and $l = 1, 2$. Due to the nature of this example, no model is postulated for $\delta_l(\boldsymbol{x})$. Consequently, the estimated theoretical predictions at each training point $\boldsymbol{x}_i$ are obtained by $\hat{f}_l(\boldsymbol{x}_i) = h_l(\boldsymbol{x}_i)$. Note, in a multi-dimensional EFT setting, the strategy discussed from Section 3 remains applicable.

The results from a 30-tree BART-BMM model are shown in Figure 5. The leftmost plot displays the absolute value of the mean residuals, $|\hat{f}_\dagger(\boldsymbol{x}) - f_\dagger(\boldsymbol{x})|$ where $\hat{f}_\dagger(\boldsymbol{x})$ denotes the mean prediction from the BMM model. Based on the residual plot, it appears $f_\dagger(\boldsymbol{x})$ is adequately recovered across the majority of the domain with an RMSE of 0.2575. As expected, the error in the mean prediction noticeably increases in the upper left corner of the domain, where only two training points are included and both simulators are inaccurate.

The second and third plots illustrate the posterior mean weight functions for each simulator. Based on the middle plot, the first simulator has increasing utility as x_1 and x_2 both increase. This is to be expected, as $h_1(\boldsymbol{x})$ is composed of two expansions centered about π. Note, the mean value of $w_1(\boldsymbol{x})$ does not reach 1 in the upper right corner of the domain because the simulator slightly overestimates the peak of $f_\dagger(\boldsymbol{x})$ in this region. Meanwhile, the posterior mean of $w_2(\boldsymbol{x})$ indicates $h_2(\boldsymbol{x})$ has high utility for $\boldsymbol{x} \in [-\pi, \pi] \times [-\pi, 0]$, which is to be expected given the nature of the expansions included in this simulator. Moreover, the predictions from $h_2(\boldsymbol{x})$ appear to align closely with the data, and thus $f_\dagger(\boldsymbol{x})$, as is evident by the weights approaching values near 1 in the bottom half of the domain.

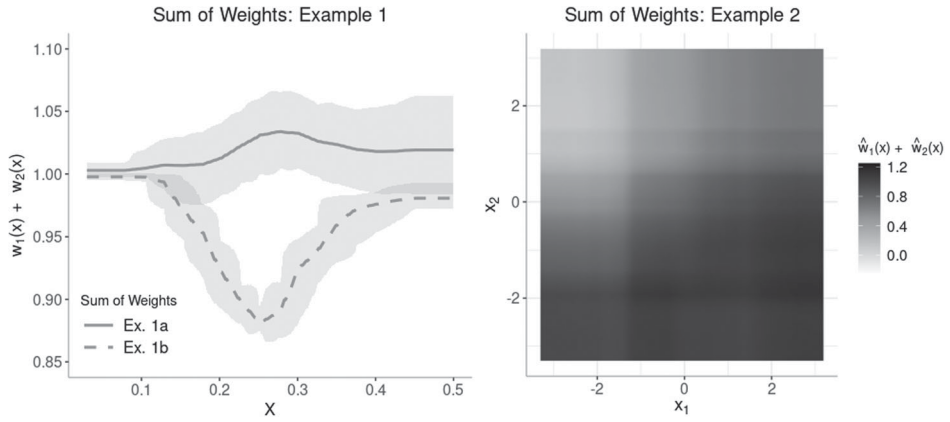

Figure 6. (Left) The posterior mean estimates and 95% credible intervals (shaded) of the sum of weight functions from Examples 1a and 1b (solid and dashed). (Right) The posterior mean estimate of the sum of weight functions in Example 2.

6. Discussion

A variety of frequentist and Bayesian approaches are available for model averaging and mixing. Each method involves estimating the overall predictive mean or density based on the individual models. The selection between these two objectives should ultimately be guided by the underlying statistical inference one wishes to ascertain. In computer experiments, a primary objective is to recover the underlying system, which is generally expressed as the mean function in an additive model for the observational data. Hence, a mean estimation approach is more desirable when working within this setting compared to a predictive density estimation, which is modeled with the intention of predicting a future observation $\tilde{y}$.

Example 5.1 compares the proposed mean estimation method versus a density estimation method in Hierarchical Stacking (HS). In HS, the weight functions are learned relative to leave-one-out (LOO) predictive densities under a simplex constraint. These LOO densities incorporate information regarding the mean and variance of each EFT at a given x. In portions of the domain where a model may rapidly diverge, the resulting LOO predictive density is shrunk toward 0. In turn, the corresponding weight function will approach 0, however, it may struggle to obtain a small enough value to shrink out the effect of the diverging mean. Meanwhile, shrinking the effect of a diverging prediction appears to be easier when mixing the mean predictions from each EFT.

The primary objective of the weight functions is to re-scale the predictions given by each individual model so that a linear combination of these predictions can adequately recover the true system. Given the prior regularization method applied to the weight functions, exact interpretation of the resulting values can be unclear. However, using this regularization perspective, one can conclude that weight functions which fall close to 0 within a particular subregion indicate that the corresponding model is unnecessary for the overall prediction. Meanwhile, a model which is the unique local expert within a particular region should be weighted by values close to 1. Overall, a joint interpretation of the weight functions is appropriate, particularly in regions where the weights concentrate around values away from 0 or 1. These features are observed across each example.

The benefit of the proposed regularization approach can further be understood through the posterior distribution of the sum of the weight functions, $w_{\text{sum}}(\boldsymbol{x}) = \sum_{l=1}^{K} w_l(\boldsymbol{x})$, as shown in Figure 6. The posterior of $w_{\text{sum}}(x)$ from Example 1a (left panel, solid) is centered very close to 1 with relatively small amounts of uncertainty. This results because: (i) the prior regularization and (ii) $f_\dagger(x)$ lies between the selected EFTs, which indicates a convex combination is appropriate. Even though a sum-to-one property is not strictly imposed, it appears to naturally occur in this situation where an interpolation of the competing models is appropriate. Meanwhile, the posterior of $w_{\text{sum}}(x)$ from Example 1b (left, dashed) significantly drops below 1 in the intermediate range of the domain because both EFTs overestimate the true system, which renders a convex combination to be inappropriate. Similar features are observed in the two-dimensional example, as the mean of $w_{\text{sum}}(\boldsymbol{x})$ concentrates around 1 in areas where at least one of the simulators aligns well with the true system. Meanwhile, when neither simulator is accurate (i.e., the top left corner) the mean value of $w_{\text{sum}}(\boldsymbol{x})$ is drastically below 1. From these observations, it appears the BART-BMM approach benefits by not imposing strict assumptions, such as a simplex constraint, on the weights. The examples in Section 5 emphasize the two-model case, however, analogous interpretations of the weight functions are applicable when considering model sets of size K.

Finally, the weight functions can be used to better understand the $\mathcal{M}$-open assumption associated with the model set. An initial confirmation of the $\mathcal{M}$-open setting can be made when the weight functions noticeably change as a function of the inputs. This observation indicates localized performance of each model, hence, one can confirm the true system is not contained in the set. If the weight functions are nearly constant, one may also wish to check the posterior of $w_{\text{sum}}(\boldsymbol{x})$ to see if the sum of the weights is fixated close to 1. Such a case may suggest model averaging with a simplex constraint could also be an appropriate solution. This alone is not enough to confirm or deny the $\mathcal{M}$-open assumption, however, it may indicate that the $\mathcal{M}$-complete or $\mathcal{M}$-closed assumptions are possible for the model set. A final case to consider is the situation where a single model receives a weight near 1 while the effects of the competing models are shrunk to 0 across a subregion of the domain. This situation may indicate the model set is $\mathcal{M}$-closed conditional on the subregion of interest despite falling in the $\mathcal{M}$-open case when considering the entire domain.

In conclusion, this work proposes a Bayesian treed framework to mix predictions from a set of competing models, each of which are intended to explain the physical system across a subregion of the domain. This approach falls within the class of problems referred to as Bayesian model mixing, as input-dependent weights are defined to reflect the localized behavior of each model. The weight functions are modeled using a sum-of-trees and are regularized via a multivariate Gaussian prior. The tree bases coupled with the regularization approach allows for the weights to be learned in a flexible nonparametric manner free of strict constraints. Using the weight functions, predictions from the individual models are mixed via a linear combination. The success of this mixing approach is demonstrated on three examples, each of which considers models with localized predictive performances. Leveraging the localized behavior of the individual models leads to significant improvements in the posterior prediction and uncertainty quantification of $f_\dagger(\boldsymbol{x})$ and the overall interpretation of the system compared to existing global and local weighting schemes.

Supplementary Materials

The supplementary material includes the essential derivations of the methodology along with additional information regarding EFTs. Code implementing the method and reproducing the examples is also available online.

Acknowledgments

The authors would like to thank the Editor, an Associate Editor, and two referees for helpful comments on this work.

Disclosure Statement

No potential conflict of interest was reported by the author(s).

Funding

The work of JCY and RJF work was supported in part by the National Science Foundation under Agreement OAC-2004601. The work of MTP was supported in part by the National Science Foundation under Agreements DMS-1916231, DMS-1564395, OAC-2004601, and in part by the King Abdullah University of Science and Technology (KAUST) Office of Sponsored Research (OSR) under Award No. OSR-2018-CRG7-3800.3. The work of TJS was supported in part by the National Science Foundation under Agreement DMS-1564395 (The Ohio State University).

References

Bernardo, J. M., and Smith, A. F. (1994), *Bayesian Theory*, Chichester: Wiley. [1]

Breiman, L. (1996), "Stacked Regressions," *Machine Learning*, 24, 49–64. [3]

Bunea, F., Tsybakov, A. B., and Wegkamp, M. H. (2007), "Aggregation for Gaussian Regression," *The Annals of Statistics*, 35, 1674–1697. [3]

Burgess, C. P. (2020), *Introduction to Effective Field Theory: Thinking Effectively about Hierarchies of Scale*, Cambridge: Cambridge University Press. [2,8]

Burnham, K. P., and Anderson, D. R. (1998), "Practical Use of the Information-Theoretic Approach," in *Model Selection and Inference*, pp. 75–117, New York: Springer. [3]

Chipman, H. A., George, E. I., and McCulloch, R. E. (1998), "Bayesian CART Model Search," *Journal of the American Statistical Association*, 93, 935–948. [4]

——— (2002), "Bayesian Treed Models," *Machine Learning*, 48, 299–320. [4]

Chipman, H., George, E., and McCulloch, R. (2010), "BART: Bayesian Additive Regression Trees," *The Annals of Applied Statistics*, 4, 266–298. [4,7]

Clyde, M., and Iversen, E. S. (2013), "Bayesian Model Averaging in the M-Open Framework," in *Bayesian Theory and Applications*, eds. P. Damien, P. Dellaportas, N. G. Polson, and D. A. Stephens, 484–498, Oxford: Oxford University Press. [3]

Draper, D. (1995), "Assessment and Propagation of Model Uncertainty," *Journal of the Royal Statistical Society*, Series B, 57, 45–70. [1]

Georgi, H. (1993), "Effective Field Theory," *Annual Review of Nuclear and Particle Science*, 43, 209–252. [2]

Gramacy, R. B. (2020), *Surrogates: Gaussian Process Modeling, Design, and Optimization for the Applied Sciences*, Boca Raton, FL: Chapman and Hall/CRC. [5]

Gramacy, R. B., and Lee, H. K. H. (2008), "Bayesian Treed Gaussian Process Models with an Application to Computer Modeling," *Journal of the American Statistical Association*, 103, 1119–1130. [4]

Hansen, B. E. (2007), "Least Squares Model Averaging," *Econometrica*, 75, 1175–1189. [3]

Hastie, T., and Tibshirani, R. (2000), "Bayesian Backfitting" (with comments and a rejoinder), *Statistical Science*, 15, 196–223. [4]

Honda, M. (2014), "On Perturbation Theory Improved by Strong Coupling Expansion," *Journal of High Energy Physics*, 2014, 1–44. [2,4,5]

Le, T., and Clarke, B. (2017), "A Bayes Interpretation of Stacking for M-complete and M-open Settings," *Bayesian Analysis*, 12, 807–829. [2,3]

Melendez, J. A., Furnstahl, R. J., Phillips, D. R., Pratola, M. T., and Wesolowski, S. (2019), "Quantifying Correlated Truncation Errors in Effective Field Theory," *Physical Review C*, 100, 044001. [5,6]

Melendez, J., Furnstahl, R., Grießhammer, H., McGovern, J., Phillips, D., and Pratola, M. (2021), "Designing Optimal Experiments: An Application to Proton Compton Scattering," *The European Physical Journal A*, 57, 1–24. [6]

Petrov, A. A., and Blechman, A. E. (2016), *Effective Field Theories*, Singapore: World Scientific. *https://www.worldscientific.com/doi/abs/10.1142/8619* [2]

Phillips, D., Furnstahl, R., Heinz, U., Maiti, T., Nazarewicz, W., Nunes, F., Plumlee, M., Pratola, M., Pratt, S., Viens, F. et al. (2021), "Get on the BAND Wagon: A Bayesian Framework for Quantifying Model Uncertainties in Nuclear Dynamics," *Journal of Physics G: Nuclear and Particle Physics*, 48, 072001. [7]

Prado, E. B., Moral, R. A., and Parnell, A. C. (2021), "Bayesian Additive Regression Trees with Model Trees," *Statistics and Computing*, 31, 1–13. [4]

Pratola, M. T. (2016), "Efficient Metropolis–Hastings Proposal Mechanisms for Bayesian Regression Tree Models," *Bayesian Analysis*, 11, 885–911. [4]

Raftery, A., Madigan, D., and Hoeting, J. (1997), "Bayesian Model Averaging for Linear Regression Models," *Journal of the American Statistical Association*, 92, 179–191. [1,3]

Santner, T. J., Williams, B. J., and Notz, W. I. (2018), *The Design and Analysis of Computer Experiments* (2nd ed.), New York: Springer. [5]

Semposki, A. C., Furnstahl, R. J., and Phillips, D. R. (2022), "Interpolating Between Small- and Large-g Expansions Using Bayesian Model Mixing," *Physical Review C*, 106, 044002. *https://link.aps.org/doi/10.1103/PhysRevC.106.044002* [5]

Sill, J., Takács, G., Mackey, L., and Lin, D. (2009), "Feature-Weighted Linear Stacking," arXiv preprint arXiv:0911.0460 . [3]

Yang, Y., and Dunson, D. B. (2014), "Minimax Optimal Bayesian Aggregation," arXiv preprint arXiv:1403.1345. [3]

Yao, Y., Pirš, G., Vehtari, A., and Gelman, A. (2021), "Bayesian Hierarchical Stacking: Some Models are (somewhere) Useful," *Bayesian Analysis*, 1, 1–29. [2,4]

Yao, Y., Vehtari, A., Simpson, D., and Gelman, A. (2018), "Using Stacking to Average Bayesian Predictive Distributions," *Bayesian Analysis*, 13, 917–1007. [2,3]