Maynooth University
National University
of Ireland Maynooth

Extensions to Bayesian tree-based machine learning algorithms

A dissertation submitted for the degree of
Doctor of Philosophy

By:

Estevão Batista do Prado

Under the supervision of:

Prof. Andrew C. Parnell

Dr. Rafael A. Moral

Hamilton Institute
National University of Ireland Maynooth
Ollscoil na hÉireann, Má Nuad

August 2022

To my family, for the infinite love and respect.

In memory of grandpa Arlindo and aunt Sônia.

Declaration

I hereby declare that I have produced this manuscript without the prohibited assistance of any third parties and without making use of aids other than those specified.

The thesis work was conducted from October 2018 to August 2022 under the supervision of Professor Andrew C. Parnell and Dr. Rafael A. Moral in the Hamilton Institute, National University of Ireland Maynooth.

Estevão Batista do Prado.

Maynooth, Ireland,

August 2022.

Sponsor

This work was supported by a Science Foundation Ireland Career Development Award grant number 17/CDA/4695.

Collaborations

Andrew C. Parnell: As my supervisor, Professor Parnell (Maynooth University) supervised and collaborated on the work of all chapters.

Rafael A. Moral: As my supervisor, Dr. Moral (Maynooth University) supervised and collaborated on the work of all chapters.

Danilo A. Sarti: Dr. Sarti (Maynooth University) collaborated on the work of Chapter 4 by writing the literature review and interpreting the model results in the case study section.

Alan N. Inglis: Alan contributed with R scripts to create the plots with marginal and interaction effects of the proposed model in Chapter 4.

Alessandra L. dos Santos: Alessandra contributed to the understanding of the data used in the case study section in Chapter 4.

Catherine Hurley: As Alan's co-supervisor, Dr. Hurley (Maynooth University) collaborated on the work of Chapter 4 by providing insights for the plots.

Keefe Murphy: Dr. Murphy (Maynooth University) collaborated on Chapter 5, by organising and guiding the writing, providing R code to optimise model performance, and contributing to the design and interpretation of the simulated and real data analyses.

Nathan McJames: Nathan McJames contributed to the understanding of the data used in the real-world application in Chapter 5.

Ann O'Shea: As Nathan's co-supervisor, Professor O'Shea (Maynooth University) collaborated on the work of Chapter 5 by providing insights for the interpretation of the proposed model results.

Publications

The chapters contained in this thesis have been either published or submitted to peer-reviewed journals. Chapter 3 has been published in the journal *Statistics and Computing* and Chapter 4 has been accepted in the journal *The Annals of Applied Statistics* and is due to appear. Chapter 5 has been submitted to the journal *The Annals of Applied Statistics* and is currently under review. In Chapter 4, Danilo A. Sarti and Estevão B. Prado are joint first authors.

Peer-reviewed journal article:

- Prado, E.B., Moral, R.A. & Parnell, A.C. Bayesian additive regression trees with model trees. *Statistics and Computing* 31, 20 (2021). `https://doi.org/10.1007/s11222-021-09997-3`.
- Sarti, D.A., Prado, E.B., Inglis, A.N., dos Santos, A.A.L, Hurley, C., Moral, R.A. & Parnell, A.C. Bayesian additive regression trees for genotype by environment interaction models. *The Annals of Applied Statistics* (to appear).

Submitted articles (under review):

- Prado, E.B., Parnell, A.C., Murphy, K., McJames, N., O'Shea, A., & Moral, R.A. (2022). Accounting for shared covariates in semi-parametric Bayesian additive regression trees. Under review in the journal *The Annals of Applied Statistics*. *arXiv* preprint: `https://arxiv.org/abs/2108.07636`.

Contents

Abstract

Bayesian additive regression trees (BART) is a Bayesian tree-based algorithm which can provide high predictive accuracy in both classification and regression problems. Unlike other machine learning algorithms based on an ensemble of trees, such as random forests and gradient boosting, BART is not based on recursive partitioning. Rather, it is a fully Bayesian model built upon a likelihood function and diligently specified prior distributions.

In this thesis, we propose methodological extensions to BART to deal with two main limitations of tree-based methods: the limited ability to fit smooth functions, which is inherently associated with how methods based on trees are built, as well as the lack of adequate mechanisms that enable to quantify in an interpretable fashion the impact of certain inputs of primary interest on the output.

Firstly, we present an extension that aims to deal with linear effects at the terminal nodes level. By considering linear piecewise functions instead of piecewise constants, local linearities are captured more efficiently and fewer trees are required to achieve equal or better performance than BART. Secondly, motivated by an agricultural application, we develop a semi-parametric BART model in which marginal genotypes and environment effects are estimated along with their interactions.

Last, motivated by data collected in 2019 under the seventh cycle of the quadrennial Trends in International Mathematics and Science Study, we extend semi-parametric models based on BART, which generally assume that the set of covariates in the linear predictor and the BART model are mutually exclusive, to account for shared covariates. In particular, we change the tree-generation moves in BART to deal with bias/confounding between the parametric and non-parametric components, even when they have covariates in common.

Acknowledgements

First of all, I would like to thank God for the gift of life, for all the blessings and for the wonderful family.

To my supervisors, Prof. Andrew Parnell and Dr. Rafael Moral, for choosing me among other qualified students back in 2018 and for the great opportunity to work with them. Their ideas, guidance, motivation, patience and support contributed enormously to my learning during the PhD, and I will be eternally grateful for all.

To my collaborators for all the help and hard work. In special, I would like to thank Dr. Keefe Murphy for his time, patience, thorough criticism and constructive feedback on my work. His thousands of comments (literally!) along with suggestions and guidance made my work better, and I have no words to express my gratitude.

To my internal and external examiners, Dr. Niamh Cahill and professor Hugh Chipman, for the time invested in reading my work and for the feedback provided.

To my friends and staff members in the Hamilton Institute/Department of Mathematics and Statistics at Maynooth University. I feel so fortunate to be surrounded by such smart and talented students and professionals. In special, I would like to thank Alessandra for all the brotherhood; Alan and Hochan for all the camaraderie inside and outside the office; Mateus for the numerous talks/discussions about BART and football; Akash, Amin, Carl and Fredito for the gym/football time together; Eleni and my housemate Anna for being such nice, positive and pleasant friends; Rosemary, Kate, Janice and Gráinne for their kindness and for all the help.

To my family, to which I would like address in Portuguese. Galera, se eu não tivesse cada um de vocês em minha vida, qualquer esforço seria em vão. Obrigado, vô Arlindo, vó Zélia, mãe, irmãos, irmã, primos, primas, priminhos em segundo grau, tios, tias e, claro, a mais nova membra da família, a joia da coroa, minha sobrinha Clara. Me sinto muito abençoado por tê-los em minha vida. Sou muito grato por todo amor e respeito que existem entre nós. Em especial e *in memoriam*, eu gostaria de agradecer a tia Sônia por ter me acolhido em sua casa como um filho lá em 2016 durante o meu período de mestrado em Belo Horizonte. Infelizmente, ela não está mais entre nós, mas seu amor e apoio foram fundamentais para que eu conseguisse avançar em meus estudos. Sua generosidade e suporte estarão sempre em minhas memórias e coração. Obrigado por tudo, tia!

Finalmente, eu gostaria de fazer um agradecimento especial *in memoriam* ao vô Arlindo, que nos deixou semanas antes da minha defesa. Infelizmente, sua partida foi rápida e não consegui vê-lo pela última vez ou mesmo dar um último beijo e abraço. O vô Arlindo foi o melhor entre os melhores. O nosso camisa 10! Um avô maravilhoso, generoso, paciente e muito, muito amoroso com sua família, especialmente com seus netos. Me lembro de quando eu era piazinho e vivia para baixo e para cima em sua garupa, como um chiclete na sola de um sapato. Lembro também de quando caímos de bicicleta em uma valeta antes de chegarmos à igreja – eu na garupa e o vô Arlindo na boleia. Não sei o que aconteceu, mas acho que aquele dia passamos perto demais da beira da rua. Maior, pude contemplar todo o amor que ele dispensou aos seus bisnetos, mesmo não tendo a mesma energia de outrora. Aos finais de semana sempre em sua casa, pude ver, religiosamente, o quanto ele amava os pequenos e o quanto era amado por eles. Independente de onde estiver, vô, saiba que eu te amei, te amo e te amarei por toda minha vida. O senhor foi um exemplo para mim e tenho certeza que para todos da nossa família! O amor que sinto pelo senhor jamais morrerá e jamais poderá ser expresso em palavras. Neste momento, não consigo continuar escrevendo pois lágrimas embaçam o meu olhar, mas, enquanto eu viver, o senhor estará sempre em minhas memórias. Que o senhor possa descansar em paz! Obrigado, obrigado, obrigado!

List of Figures

List of Tables

CHAPTER 1

Introduction

1.1 Motivation

Tree-based methods are in general recursive partitioning algorithms that split the data into homogeneous partitions given a response variable and a set of covariates. These methods are flexible, scale well as the numbers of observations and variables grow, and have been part of the toolkit of researchers and applied data analysis practitioners due to, among other things, the wide availability of fast, easy-to-use packages in free and private software. Ensemble-of-trees algorithms, such as random forests (Breiman, 2001) and gradient boosting (Friedman, 2001), are widely used in non-parametric regression problems as they make minimal assumptions about the data and are known for their great prediction accuracy in real-world applications.

Motivated by their contemporaries, Bayesian approaches based on a single tree emerged in the late 1990s in the pioneering works of Chipman et al. (1998) and Denison et al. (1998). In both papers, the aim was to propose a Bayesian version of the classification and regression tree algorithm (CART; Breiman et al., 1984) in which a stochastic search procedure and prior distributions replace the recursive partitioning. Conceptually, the Bayesian CARTs are full statistical models since all elements of a Bayesian model are present, such as a probability distribution for the response, prior distributions, and a posterior distribution. With the advent of methods combining multiple trees rather than a single tree, namely boosting (Freund and Schapire, 1997), bagging (Breiman, 1996), and random forests, an 'ensemble-of-trees Bayesian model' seemed to be the next step in the early 2000s.

First proposed in a seminal paper by Chipman et al. (2010) more than a decade ago, Bayesian additive regression trees (BART) is a Bayesian non-parametric model based on an ensemble of trees. Initially proposed for regression and classification settings, BART predicts a univariate response variable by using a set of regularised, shallow trees which generate the fit through an additive structure. As with any Bayesian model, a probability distribution is assumed for the response and prior distributions are placed on the necessary quantities. The regularisation is established via the prior distributions on both the tree structure and the terminal node parameters, and it is a key element behind the excellent predictive performance of the BART model. Through an iterative Bayesian backfitting Markov Chain Monte Carlo algorithm (Hastie and Tibshirani, 2000; Brooks et al., 2011), the trees are learned and samples from the posterior distribution are generated.

Since its proposal in 2010, BART has drawn attention from researchers and professionals from various fields. Early applications/extensions involve credit risk modelling (Zhang and Härdle, 2010), causal inference (Hill, 2011; Green and Kern, 2012; Hill and Su, 2013), survival analysis (Bonato et al., 2011), and spam-detection (Abu-Nimeh et al., 2008). More recently, BART has been explored in proteomic discovery (Hernández et al., 2018), hospitals' evaluation (Liu et al., 2015), preeclampsia and stillbirth risk (Starling et al., 2019, 2020), time-series analysis (Prüser, 2019; Clark et al., 2021), to list a few. Methodologically, BART has also been extended to account for polychotomous response (Kindo et al., 2016b), multivariate skewed response (Um, 2021), density regression (Orlandi et al., 2021), count/semi-continuous zero-inflated data (Linero et al., 2020; Murray, 2021), high-dimensional data (Linero and Yang, 2018; He et al., 2019), and heterocedastic data (Pratola et al., 2020). There are also works combining BART with varying coefficient models (Deshpande et al., 2020), quantile regression (Kindo et al., 2016a), and semi-parametric models (Zeldow et al., 2019; Tan and Roy, 2019).

Due to the its flexibility and great predictive capabilities, BART has received considerations from the theoretical point of view. For instance, Linero and Yang (2018), Ročková and Saha (2019), Ročková and van der Pas (2020), and Jeong and Ročková (2020) study the speed at which the posterior distribution of 'Bayesian forests' (i.e., Bayesian methods based on ensemble of trees, of which BART is a prominent member) contracts to the true posterior, which contributes to a better understanding of why BART does so well in practice. On the other hand, easy-to-use software implementations, such as the R (R Core Team, 2020) packages `dbarts` (Dorie, 2020), `bartMachine` (Kapelner and Bleich, 2016), and `BART` (McCulloch et al., 2020), are efficient, well-implemented tools that make BART a reality for practitioners.

This thesis presents proposals for overcoming some key limitations of BART and some of its semi-parametric versions. Though BART is a popular algorithm and has been widely applied in a variety of real-world data, it is well-known (Hastie et al., 2009; Linero and Yang, 2018; Tan and Roy, 2019) that it is a black-box model where i) local smooth effects are not efficiently estimated as well as ii) the interpretation of effects of covariates of interest on the response is not straightforward as in a parametric model.

The first point is due to the step-function estimates commonly found in tree-based methods, which can offer some degree of smoothness, especially when the fit is based on a large number of trees. However, to approximate a linear effect, many splits are usually required on the same variable so as to attain a good fit. Also, due to the learning process in BART where the splitting rules are randomly determined, to estimate linear effects becomes more challenging particularly when the number of covariates is large. This drawback is not exclusive to BART and has been dealt with in random forests by Friedberg et al. (2020) and Künzel et al. (2022), who replace the piecewise constants with local linear regressions. Regarding Bayesian tree-based models, Chipman et al. (1998) also adopt local linear regressions at the terminal node levels but in the context of a single tree.

The second limitation is alleviated by semi-parametric BART models (Zeldow et al., 2019; Tan and Roy, 2019), where, for interpretational purposes, the response is approximated by a parametric (linear predictor) and non-parametric (BART) components. From the practical point of view, the reasoning is to consider that covariates for which one might be interested in measuring the effects on the response are specified in the linear predictor, whereas covariates that are thought to be of non-primary interest are dealt with by the non-parametric component. However, the semi-parametric BART models assume that the two components (linear predictor and BART) cannot share covariates, which prevents BART from capturing both interactions among covariates in the linear predictor and between covariates of the two components.

1.2 Outline of the thesis

The remainder of this thesis is organised as follows. In Chapter 2, we provide some background on the development of BART, elaborate on the essential definitions, and clarify related notation used throughout the thesis. In Section 2.1, we provide a brief introduction to decision trees, which are an essential element for tree-based methods. We initially present some terminology and how trees are learned under the CART algorithm. In Sections 2.2 and 2.3, we provide an extensive review of the Bayesian CART (BCART; Chipman et al., 1998; Denison et al., 1998) and BART models. In Section 2.4, we summarise how we approach the aforementioned limitations of BART. Finally, the subsequent chapters are presented in the format of three journal articles and were adapted in an effort to avoid redundancy, when possible, and to keep coherence throughout the thesis.

In Chapter 3, we extend the work of Chipman et al. (2002) by considering observation-specific predictions at the terminal node level within the BART framework to address the issue related to the difficulty of approximating smooth effects. Here, the novel MOTR-BART framework is introduced from the perspective of improving the predictive performance of BART in settings where interpretability is not of such great importance. Instead of estimating one piecewise constant as the predicted value for each terminal node, a local Bayesian linear regression is considered where the covariates in the linear predictors are chosen based on the tree structure. Under the new formulation, smooth effects are captured more efficiently, while the recommended number of trees required to predict the response variable is drastically reduced. Through a simulation study based on the Friedman equation (Friedman, 1991), which contains linear and non-linear effects, we show that MOTR-BART consistently outperforms BART and other tree-based competitors regardless of the sample size and number of noise covariates. We also explore the performance of MOTR-BART on well-known real-world applications, with MOTR-BART presenting the lowest or the second lowest out-of-sample root mean squared errors on almost all datasets considered.

Chapters 4 and 5 also make use of linear components but under the semi-parametric framework so as to lend interpretability to covariates of primary interpretational interest. This is achieved in Chapter 4 by using a linear predictor along with BART to estimate the main effects and induce interactions in a genotype by environment setting. One of the motivations for this new model lies in the structure of the additive main effects multiplicative interactions model (AMMI; Mandel, 1971), which is a commonly adopted model in the agricultural literature to analyse crop yield, where the response variable, usually production of a crop in tonnes per hectare, is estimated by the sum of linear and bi-linear terms. The latter term is responsible for capturing interactions between genotypes and environments in the AMMI models and its components are obtained via a singular value decomposition. Under our approach, we replace the bi-linear term with BART by significantly modifying it to account for potential interactions. Via simulation studies, we show that the proposed model AMBARTI presents great predictive performance since it is able to estimate more complex interaction structures than AMMI and its Bayesian counterpart. As a case study, we analyse wheat data kindly provided by the Irish Department of Agriculture from 2010 to 2019. We compare AMBARTI with other interaction detection models in terms of out-of-sample error and observe it provided the smallest errors for 7 out of 10 periods analysed. Finally, we introduce new visualisations that help assess the marginal and interaction effects from the proposed model, which is helpful as it can assist professionals with no quantitative background.

In Chapter 5, we extend BART-based semi-parametric models to account for shared covariates between the parametric and non-parametric components. The rationale behind semi-parametric BART models is to combine a linear predictor and a BART model so that covariates of primary interpretational interest are specified in the linear predictor, while covariates of non-primary interpretational interest are dealt with by BART. Broadly speaking, the design matrix is split into two subsets, where the first is used in the linear predictor and the second by the BART model. This approach has advantages over, for example, generalised linear models (GLMs; Nelder and Wedderburn, 1972; McCullagh and Nelder, 1989), since interactions and non-linearities are estimated without pre-specification, while the linear predictor offers a level of interpretability similar to parametric models. However, one well-known limitation of these models is that they assume that covariates of primary interest, which are specified via linear predictor, cannot be part of the subset used by BART. Implicitly, these models assume there is no interaction among the covariates in the linear predictor nor interactions between covariates in the linear predictor and BART model. We circumvent this issue by changing the moves of the BART model specifically for when covariates of primary interest are selected to be part of the structure of the trees. We also introduce checks on the topology of the trees to guarantee that there is no confounding between the linear and BART components. We demonstrate the unbiasedness of the coefficient estimates in the linear predictor through two simulation experiments and show enhanced prediction performance on real data from an international education study, where we point out the benefits of sharing covariates across the two components.

All proposed methods in this thesis were implemented using the R (R Core Team, 2020) software and are accessible on the author's Github[1] via three public repositories. The repositories `MOTR-BART`, `AMBARTI`, and `CSP-BART` are related to Chapters 3, 4, and 5, respectively. To avail interested practitioners, R scripts are made available such that the analyses and plots presented throughout are reproducible. In addition, all datasets are available, either through R packages, which are presented in the R scripts, or files in the aforementioned repositories, with the wheat data from the Irish Department of Agriculture in Chapter 4 being anonymised to prevent the identification of genotypes and environments.

Finally, in Chapter 6, we conclude the thesis indicating topics for future research.

[1] https://github.com/ebprado

CHAPTER 2

Bayesian additive regression trees

Before introducing BART, we first introduce decision trees under the CART recursive partitioning algorithm to give the overall picture of a decision tree and also to present terminologies and notations. We then review the Bayesian CART in order to contrast how Bayesian and non-Bayesian trees are built as well as a preparation to introduce BART, since many elements of Bayesian CART are part of the BART model. Finally, we present an extensive review of BART.

2.1 Decision trees

The idea behind CART or any other recursive partitioning procedure is to divide a population based on a set of covariates into disjoint subsamples where the observations within each subsample are homogeneous in relation to a response variable. Let $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ denote a p-dimensional vector corresponding to the i-th row of the design matrix $\mathbf{X}$ and let $y_i \in \mathbf{y}$ be the response variable for observation i, where $i = 1, \ldots, n$.

For the sake of simplicity, we illustrate in Figure 2.1 the growing process of a CART tree in the context of regression. Initially, all observations belong to a tree with a single node, as shown in panel (a). Next, in panel (b), the data are split into two disjoint parts based on a splitting variable (x_2) along with a splitting value (1), which are shown inside the rectangle. The combination of a splitting variable and a splitting value is generally called a splitting rule, since together they completely define a rule to partition the data. The observations go to the left if they satisfy the splitting rule, otherwise they go to the right. In panel (b), the tree has an internal/non-terminal node (rectangle) at the top and two terminal/child nodes (circles) at the bottom. From panel (b) to (c), the observations in the left-most terminal node are partitioned and new terminal nodes are created. This binary growing process is recursively applied until a stopping criterion is met as, for example, the minimum number of observations in a node. Conventionally, all observations that belong to a terminal node have a constant μ_ℓ as predicted value.

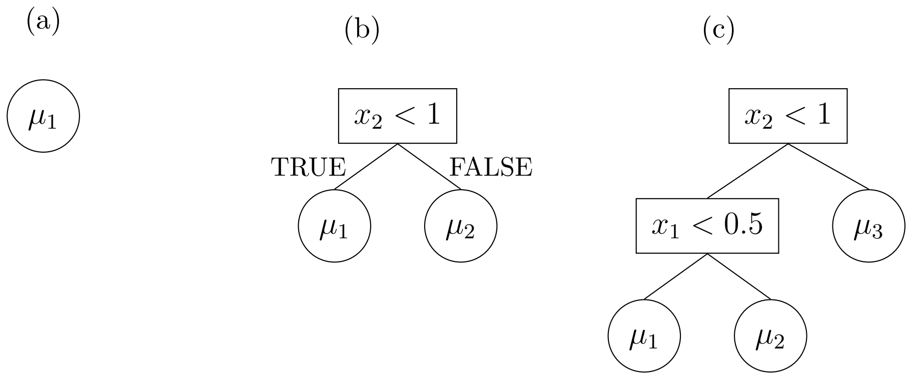


Figure 2.1: An example of a decision tree built by the CART algorithm. In panel (a), the data initially are allocated into a stump. In panel (b), the growing process splits the data into two disjoint subsamples based on the splitting rule $x_2 < 1$. In panel (c), the left-most subsample in panel (b) is grown considering the splitting rule $x_1 < 0.5$, which generates new terminal nodes. The parameters μ in the terminal nodes indicate the prediction applied to that subsample.

In Figure 2.1, we can see that the tree structure in panel (c) is based on two splitting rules which involve x_1 and x_2. The choice of the covariates that are part of the tree structure is a minimisation problem given by

$$\min_{k,j} \left(\sum_{i:\mathbf{x}_i \in \mathcal{P}_1(x_{ik}, c_j)} (y_i - \mu_1)^2 + \sum_{i:\mathbf{x}_i \in \mathcal{P}_2(x_{ik}, c_j)} (y_i - \mu_2)^2 \right), \tag{2.1}$$

where x_{ik} denotes the k-th covariate for observation i, c_j the j-th split point in the domain of $x_{\cdot k}$, and μ_ℓ is the average of the response variable y_i for observations that belong to terminal node ℓ. We denote $\mathcal{P}_\ell$ the partition formed by the splitting rules that define the terminal node ℓ in a tree so that all observations that belong to $\mathcal{P}_\ell$ have the same predicted value. From above, the optimal splitting rule is obtained after an exhaustive search over all covariates and their split points. Once the best splitting rule is obtained, the same process is repeated until a stopping rule is reached.

It is relatively straightforward with minor changes to use the same recursive rationale for a categorical response variable with K distinct classes. To do so, we first need to replace the sum of squared errors in (2.1) for some other measure appropriate for classification and then change slightly the minimisation problem. Let $\hat{p}_{\ell k}$ be the proportion of observations in terminal node ℓ of class k, which can be calculated as

$$\hat{p}_{\ell k} = \frac{1}{n_\ell} \sum_{i:\mathbf{x}_i \in \mathcal{P}_\ell} \mathbb{I}(y_i = k),$$

where $\mathbb{I}(\cdot)$ is the indicator function and n_ℓ denotes the number of observations in node ℓ. In the classification case, the predicted value for all observations in node ℓ is $\mu_\ell = \{k : \hat{p}_{\ell k} = \max_k \hat{p}_{\ell k}\}$; i.e., the most common class in the node. Regarding the measures that can be used as loss functions for classification, Breiman et al. (1984) suggest

$$\begin{aligned} \text{Misclassification error} &= 1 - \hat{p}_{\ell k}, \\ \text{Gini index} &= 1 - \sum_{k=1}^{K} \hat{p}_{\ell k}^2, \\ \text{Entropy} &= - \sum_{k=1}^{K} \hat{p}_{\ell k} \log(\hat{p}_{\ell k}). \end{aligned} \tag{2.2}$$

The Gini and Entropy are usually preferred over misclassification error as they are differentiable and more sensitive to small changes (Hastie et al., 2009). Here, rather than minimising the sum of squared errors to recursively grow the tree, the metrics above are optimised instead. For instance, for a given terminal node, the lower the Gini index, the more homogeneous the observations in that node are.

Let $\psi(\mathcal{P}_{\ell L})$ and $\psi(\mathcal{P}_{\ell R})$ be any metric in (2.2) applied to two child nodes obtained from a node ℓ, which was partitioned based on a splitting rule defined by k-th covariate and j-th split point. The best splitting rule is chosen based on the optimisation of

$$\Delta(k,j) = \Big\{\psi(\mathcal{P}_\ell) - \frac{n_{\ell L}}{n_\ell}\psi(\mathcal{P}_{\ell L}) - \frac{n_{\ell R}}{n_\ell}\psi(\mathcal{P}_{\ell R})\Big\}, \tag{2.3}$$

where $n_{\ell L}$ and $n_{\ell R}$ denote the size of the left and right child nodes. The maximisation of (2.3) is equivalent to finding the best splitting rule so that the new nodes defined by $\mathcal{P}_{\cdot L}$ and $\mathcal{P}_{\cdot R}$ contain observations that are more homogeneous among themselves when compared to $\mathcal{P}_\ell$. For instance, when a splitting rule generates two child nodes where all observations in both nodes have a unique class, (2.3) is maximised as $\psi(\mathcal{P}_{\ell L}) = \psi(\mathcal{P}_{\ell R}) = 0$, $\Delta(k,j) = \psi(\mathcal{P}_\ell)$ and no further splitting is needed. In contrast, $\Delta(k,j) = 0$ (the lowest possible value) if a splitting rule creates two child nodes where the classes of the response variable in both nodes are evenly distributed.

After the binary recursive partitioning is employed in CART, either for a regression or classification tree, a cost-complexity pruning is applied to avoid over-fitting. The criterion used is based on a tuning parameter, number of terminal nodes in the tree, and on the contribution of each node to the overall sum of squared errors (regression) or any of the three measures in (2.2) (classification). The rationale is to remove nodes that have a small contribution to the overall prediction based on cross-validation (Hastie et al., 2009).

2.2 Bayesian CART

Unlike CART, decision trees built upon the Bayesian paradigm can be seen as statistical models rather than algorithmic recursive procedures. This is because a probability distribution is associated to the response variable and prior distributions are placed on the node-level parameters and binary tree structure. In this Section, we focus on the Bayesian CART proposed by Chipman et al. (1998) and Denison et al. (1998), giving more emphasis to Chipman et al. (1998) since it is the backbone of the BART model. In the end of this Section, however, we point out the main differences between the two approaches.

For regression trees, BCART models assume that observations which belong to the same terminal node are independent and identically distributed (i.i.d) and follow a probability distribution as

$$\begin{aligned} y_i|\mathbf{x}_i, \mathcal{T}, \mathcal{M} &\sim \mathrm{N}(\mu_\ell, \sigma^2), \text{ or} \\ y_i|\mathbf{x}_i, \mathcal{T}, \mathcal{M} &\sim \mathrm{N}(\mu_\ell, \sigma_\ell^2), \text{ for all } i \in \mathcal{P}_\ell. \end{aligned} \tag{2.4}$$

In the first model, the terminal nodes have different means but share the same variance, while in the second both mean and variance are specific to each terminal node. For convenience, we consider the model in (2.4) when presenting the priors and posterior conditional distributions. Given the design matrix $\mathbf{X}$ and tree structure $\mathcal{T}$, the node-level predicted values $\mathcal{M} = \{\mu_1, \cdots, \mu_b\}$ and σ^2 are the parameters of interest. To ensure that their posterior conditional distributions have closed-form expressions and marginalisations can be carried out analytically, the respective conjugate priors (2.5) and (2.6) below are placed on both parameters:

$$\mu_\ell|\mathcal{T}, \sigma^2 \sim \mathrm{N}(\bar{\mu}, \sigma^2/a), \tag{2.5}$$

$$\sigma^2 \sim \mathrm{IG}(\nu/2, \nu\lambda/2). \tag{2.6}$$

For regression trees, Chipman et al. (1998) use the observed y_i to guide the choice of the hyperparameters in both priors. For instance, $\bar{\mu}$ and $a > 0$ are specified so that the resulting prior distribution assigns substantial probability around the possible values of the y_i. On the other hand, the choices of ν and λ are based on the empirical variance of the y_i and a residual variance from some over-fitted model. The idea is to choose ν and λ so that the prior distribution for σ^2 assigns considerable probability to the interval between the two variances.

Another important component in BCART is the prior on the tree structure, which is responsible for controlling how deep/shallow the tree can be. A branching process prior is adopted where the probability of observing an internal node at depth d_ℓ is $\alpha(1 + d_\ell)^{-\beta}$, where $\alpha \in (0, 1)$ and $\beta \geq 0$ are user-defined hyperparameters. Thus, the prior on the tree topology is given by

$$p(\mathcal{T}|\alpha, \beta) = \prod_{\ell \in \mathcal{S}_I} \left[\alpha(1 + d_\ell)^{-\beta}\right] \times \prod_{\ell \in \mathcal{S}_T} \left[1 - \alpha(1 + d_\ell)^{-\beta}\right], \tag{2.7}$$

where $\mathcal{S}_I$ and $\mathcal{S}_T$ represent the sets of internal and terminal nodes, respectively. Abusing notation slightly, we refer to (2.7) as $p(\mathcal{T})$ throughout. Under this prior, nodes at the same depth are equally likely to be split *a priori*, which tends to induce trees with terminal nodes at similar depth (i.e., balanced trees). By changing α and β, the user controls how harsh should be the penalisation on the creation of new nodes as the tree gets deeper, which inevitably impacts the expected number of nodes in the tree. Equipped with the conditional distribution in (2.4) and the priors in (2.5), (2.6), and (2.7), the joint posterior distribution of the BCART model can be written as

$$p(\mathcal{T}, \mathcal{M}, \sigma^2|\mathbf{y}, \mathbf{X}) \propto p(\mathbf{y}|\mathbf{X}, \mathcal{M}, \sigma^2\mathcal{T})p(\mathcal{M}|\mathcal{T}, \sigma^2)p(\mathcal{T})p(\sigma^2).$$

As the above expression does not have a closed-form distribution, it is possible to sample from it through the following posterior conditional distributions:

$$p(\mathcal{M}|\mathbf{y}, \mathcal{T}, \sigma^2) \propto p(\mathbf{y}|\mathbf{X}, \mathcal{T}, \mathcal{M}, \sigma^2)p(\mathcal{M}|\mathcal{T}, \sigma^2), \tag{2.8}$$

$$p(\sigma^2|\mathbf{X}, \mathbf{y}, \mathcal{M}) \propto p(\mathbf{y}|\mathbf{X}, \mathcal{T}, \mathcal{M}, \sigma^2)p(\sigma^2), \tag{2.9}$$

$$p(\mathcal{T}|\mathbf{X}, \mathbf{y}) \propto p(\mathcal{T}) \int\int p(\mathbf{y}|\mathbf{X}, \mathcal{T}, \mathcal{M}, \sigma^2)p(\mathcal{M}|\mathcal{T}, \sigma^2)p(\sigma^2)d\mathcal{M}d\sigma^2. \tag{2.10}$$

The expression in (2.8) allows the sampling of the terminal node parameters. As the μ_ℓ are assumed to be i.i.d, it is possible to write $p(\mathcal{M}|\mathcal{T}, \sigma^2) = \prod_{\ell=1}^{b} p(\mu_\ell|\mathcal{T}, \sigma^2)$, where b denotes the number of terminal nodes. Using the fact that the conditional distribution of y_i is normal as well as the prior on its mean, the posterior conditional distribution for μ_ℓ is also a normal distribution. The equation in (2.9) is an inverse gamma and is used to update the residual variance.

A significant difference between CART and BCART is how the splitting rules that are used to create the tree structure are chosen. This fact is directly related to why the posterior conditional distribution for the tree in (2.10) is needed and plays an important role. Recall that in CART the best splitting rules are selected based on an exhaustive recursive search over the covariates and their split points. However, in BCART the splitting rules are based on a uniform specification whereby both the splitting covariate and the split point are determined by randomly selecting one covariate and one split point from the sets of covariates and split points available.

As the splitting rules are formed at random, BCART samples from (2.10) via a Metropolis-Hastings (Metropolis et al., 1953; Hastings, 1970) step to compare a proposed tree with its previous version so that only 'good' splitting rules are part of the tree structure. For the sake of visualisation, we illustrate this recalling Figure 2.1. At the beginning of the BCART model, we could imagine that all observations are altogether in a stump, as shown in panel a). The only possible move for a stump is a grow step, where a splitting rule is employed to create two terminal nodes. In panel b), a tree with a splitting rule defined by $x_2 < 1$ is presented. Recall that this splitting rule under the uniform specification is randomly formed, which means that we do not know whether the two disjoint partitions defined by it are actually any different. To check for their possible difference, we evaluate (2.10) twice, one for tree $\mathcal{T}$ in panel a) and one for tree $\mathcal{T}^\star$ in panel b) as

$$\alpha(\mathcal{T}, \mathcal{T}^\star) = \min\left\{1, \frac{p(\mathcal{T}^\star|\mathbf{X}, \mathbf{y})q(\mathcal{T}^\star \to \mathcal{T})}{p(\mathcal{T}|\mathbf{X}, \mathbf{y})q(\mathcal{T} \to \mathcal{T}^\star)}\right\}, \tag{2.11}$$

where $q(\mathcal{T}_A \to \mathcal{T}_B)$ represents the transition probability from tree $\mathcal{T}_A$ to tree $\mathcal{T}_B$, which is a function of the probability of the moves grow (0.25), prune (0.25), change (0.4), and swap (0.1), and the tree topology. If the move is grow and the resulting tree structure is valid[2], then

$$\begin{aligned} q(\mathcal{T} \to \mathcal{T}^\star) &= \frac{\mathbb{P}(\text{grow})}{b}, \\ q(\mathcal{T}^\star \to \mathcal{T}) &= \frac{\mathbb{P}(\text{prune})}{w^\star}, \end{aligned} \tag{2.12}$$

where $\mathbb{P}(\text{grow}) = \mathbb{P}(\text{prune}) = 0.25$, b is the number of terminal node in $\mathcal{T}$, and $w^\star$ and w are the numbers of internal nodes which are parents of two terminal nodes in $\mathcal{T}^\star$ and $\mathcal{T}$, respectively (Kapelner and Bleich, 2016). If the move is prune, then

$$\begin{aligned} q(\mathcal{T} \to \mathcal{T}^\star) &= \frac{\mathbb{P}(\text{prune})}{w}, \\ q(\mathcal{T}^\star \to \mathcal{T}) &= \frac{\mathbb{P}(\text{grow})}{b-1}. \end{aligned} \tag{2.13}$$

If the moves are change or swap, the transition kernels cancel out as the ratio of the $q(\cdot)$ is always 1 (Chipman et al., 1998). Notably, the conditional distribution for the tree works as a mechanism of comparison which rejects modifications in the tree structure that do not help reduce the residual variance.

[2] In this context, it refers to a tree with non-empty terminal nodes which have at least a minimum number of observations.

Another point that differs is the moves employed by BCART to form the tree structure. Recall that in CART the tree is firstly grown based on recursive partitioning and then pruned back to avoid over-fitting. In BCART, however, the tree can be learned based on a grow, prune, change, or swap step. For instance, the grow move is employed by randomly choosing a terminal node and splitting it into two new nodes. In contrast, during pruning a pair of two terminal nodes is selected at random and then collapsed to its parent. The change move modifies the splitting rule of any internal node in the tree. Finally, in the swap move the splitting rules of any pair of parent-child internal nodes are exchanged. For the change and swap, the internal/pair of nodes that have their splitting rules changed/swapped are selected at random. In all cases, the new learned tree is compared to its previous version as mentioned above to guarantee that only alterations that help reduce the residual variance are kept.

Finally, we recall that Chipman et al. (1998) and Denison et al. (1998) proposed Bayesian versions of the CART algorithm that present some similarities. First, both assume that the response variable in each terminal node is normally distributed with unknown mean and variance. They also share the inverse gamma distribution as prior on the variance of the terminal nodes. However, Denison et al. (1998) adopt as priors for the terminal node-level parameters a uniform distribution as well as a zero-truncated Poisson on the number of terminal nodes. Unlike the branching process prior utilised by Chipman et al. (1998), the truncated Poisson does not account for the tree topology but only for number of nodes in the tree. Under the approach of Denison et al. (1998), two trees with completely different topologies have the same probability *a priori* of being observed as long as they have the same number of nodes. Another difference is related to the sampling from the posterior conditional distribution for the tree. While Denison et al. (1998) use a plug-in strategy and reversible jump Markov Chain Monte Carlo (MCMC) algorithms (Green, 1995), Chipman et al. (1998) place conjugate priors on the mean and variance of each terminal node so that they can be analytically marginalised out of the likelihood function as in (2.10). Since the tree structure changes whenever it is learned by a grow or prune step, the number of terminal node parameters also changes. In this case, without the marginalisation of the parameters that vary by the terminal nodes, reversible jump MCMC algorithms are needed to account for the different number of parameters in the tree. By integrating out the node-level parameters, Chipman et al. (1998) simplify the posterior sampling and avoid additional computational cost.

2.3 BART

BART is a Bayesian statistical model based on an ensemble of trees where the trees are set up so that they are all (roughly) equally important. Unlike BCART, which considers one tree to predict the response variable, BART uses a set of decision trees thus providing a much better prediction. Given a design matrix $\mathbf{X}$ and a response vector $\mathbf{y}$, the BART model can be defined as:

$$y_i = \sum_{t=1}^{T} g(\mathbf{x}_i; \mathcal{T}_t, \mathcal{M}_t) + \epsilon_i, \quad \epsilon_i \sim \mathrm{N}(0, \sigma^2), \tag{2.14}$$

where $g(\cdot; \mathcal{T}_t, \mathcal{M}_t)$ is a function that returns the terminal node parameter $\mu_{t\ell}$ for any observation that belongs to the terminal node ℓ of tree t. The number of trees used to predict y_i is represented by T; Chipman et al. (2010) recommend $T = 200$ as a default but they also mention it can be chosen via cross-validation, depending on the problem at hand. The binary structure of the t-th tree is denoted by $\mathcal{T}_t$ and $\mathcal{M}_t = \{\mu_{t1}, \cdots, \mu_{tb_t}\}$, where b_t represents the number of terminal nodes of $\mathcal{T}_t$. As a Bayesian model, BART has prior distributions on $\mu_{t\ell}$, $\mathcal{T}_t$, and σ^2. These priors are carefully set up so that each tree works a weak learner, which prevents one tree from dominating the fit resulting from the additive structure in (2.14).

2.3.1 Prior distributions

The contribution of each tree is constrained by the priors on the tree structure and terminal node parameters. The prior on the trees forces them to be shallow and balanced, while the prior on the node-level parameters shrinks them towards zero thereby forcing each tree to provide a small contribution to the final fit. The prior on the tree topology is the same as in (2.7), where a non-terminal node is observed at depth $d_{t\ell}$ with $\alpha(1 + d_{t\ell})^{-\beta}$ probability, where $\alpha \in (0, 1)$ and $\beta \geq 0$; Chipman et al. (2010) recommend as a default $\alpha = 0.95$ and $\beta = 2$, which favours *a priori* shallow trees of 1, 2, 3, 4, and > 4 terminal nodes, with probability of 0.05, 0.55, 0.27, 0.09, and 0.03, respectively.

If we set $\alpha = 0.95$ and $\beta = 2$ and use the expression in (2.7), the probability of getting a stump is calculated as $1 - 0.95 \times (1+0)^{-2} = 0.05$. We recall that if a non-terminal node is observed at depth d with $\alpha(1+d)^{-\beta}$ probability, then a terminal node at the same depth is found $1 - \alpha(1+d)^{-\beta}$ probability. In Figure 2.2, we present tree structures which vary in terms of the number of nodes and depth. For example, the prior probability to observe a tree with 2 terminal nodes is $0.95 \times (1+0)^{-2} \times (1 - 0.95(1+1)^{-2})^2 = 0.55$, since there is 1 internal node at depth zero and 2 terminal nodes at depth one. For trees with 3 terminal nodes, the probability is $2 \times 0.95 \times (1+0)^{-2} \times (0.95 \times (1+1)^{-2}) \times (1 - 0.95 \times (1+1)^{-2}) \times (1 - 0.95 \times (1+2)^{-2})^2 = 0.27$, as there are 2 possible topologies, shown in panels (b) and (c), with two internal nodes (one at depth zero and one at depth one) and three terminal nodes (one at depth one and two at depth two). For trees up to 4 terminal nodes, it is possible to easily calculate the probabilities listed above, but as the number of terminal nodes increases, the number of possible tree structures grows rapidly and analytical calculation becomes impractical[3].

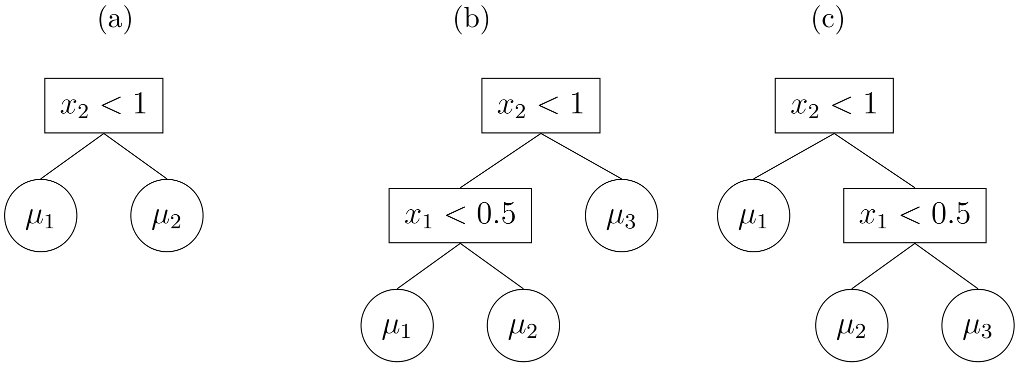


Figure 2.2: Examples of tree structures with different numbers of nodes and depth. The terminal nodes are identified as circles and the internal/non-terminal nodes are represented by rectangles. In all panels, the root node has depth zero and for every level down the tree the depth is incremented by one. The tree topology is crucial for computing the probabilities in (2.7).

[3] As noticed by Castillo and Ročková (2019), the number of different structures of binary trees with v non-terminal nodes is given by $\frac{1}{v+1}\binom{2v}{v}$, also known as the Catalan number. Recall that the number of terminal nodes in a binary tree is $b = v+1$. Thus, for a tree with $b = 1, 2, 3, 4, 5,$ and 6 terminal nodes, the number of trees with distinct shapes is $1, 1, 2, 5, 14, 42,$ and 132, respectively.

The prior on the node-level parameters $\mu_{t\ell}$ assumes they are i.i.d and normally distributed with mean μ_μ and variance σ_μ^2. By assuming that $\mu_{t\ell}|\mathcal{T}_t \overset{i.i.d}{\sim} \mathrm{N}(\mu_\mu, \sigma_\mu^2)$, it implies *a priori* that $\hat{y}_i \sim \mathrm{N}(T\mu_\mu, T\sigma_\mu^2)$, where $\hat{y}_i = \sum_{t=1}^{T} g(\mathbf{x}_i; \mathcal{T}_t, \mathcal{M}_t)$. Given the prior on $\hat{y}_i$, the rationale for choosing μ_μ and σ_μ^2 is that $\mathrm{N}(T\mu_\mu, T\sigma_\mu^2)$ should assign high probability for values of y_i between $y_{(1)}$ and $y_{(n)}$, where $y_{(1)}$ is the minimum and $y_{(n)}$ is the maximum values of the observed y_i. Thus, μ_μ is specified as $(y_{(1)} + y_{(n)})/2T$. In contrast, the specification of σ_μ^2 is carried out so that $y_{(1)} = T\mu_\mu - k\sqrt{T}\sigma_\mu$ and $y_{(n)} = T\mu_\mu + k\sqrt{T}\sigma_\mu$, where $k \in [1, 3]$ controls the amount of prior probability on the interval $(y_{(1)}, y_{(n)})$. In order to facilitate more straightforward elicitation of the prior distributions, Chipman et al. (2010) apply a transformation on the y_i so that they are constrained to the interval -0.5 and 0.5. With this re-scaling, $\mu_{t\ell} \overset{i.i.d}{\sim} \mathrm{N}(0, \sigma_\mu^2)$, where $\sigma_\mu = 0.5/(k\sqrt{T})$.

The prior on the residual variance assumes that $\sigma^2 \sim \mathrm{IG}(\nu/2, \nu\lambda/2)$, which has the same form of that adopted by Chipman et al. (1998) in (2.6). Recall that in BCART $p(\sigma^2)$ is set up to assign high probability to values in an interval where the residual variance from an over-fitted model is the lower bound and the empirical variance of the response is the upper bound. Conversely, the choice of ν and λ in BART is guided by the data in such a way that the prior assigns high probability to values less than the residual variance $\hat{\sigma}^2$ from a least-squared based model. The rationale is to first choose ν between 3 and 10 and then λ so that $p(\sigma^2 < \hat{\sigma}^2) = q$. The choice of $\nu \in [3, 10]$ is to prevent the prior from concentrating probability to very small values of σ^2; Chipman et al. (2010) recommend $\nu = 3$ and $q = 0.9$ as a default setting.

2.3.2 Posterior computation

To sample from the posterior distribution of the BART model, the Bayesian backfitting (Hastie and Tibshirani, 2000) and the Metropolis-within-Gibbs (Müller, 1991, 1992) algorithms are used. For notational convenience, let $\mathcal{T} = \{\mathcal{T}_1, \cdots, \mathcal{T}_T\}$ and $\mathcal{M} = \{\mathcal{M}_1, \cdots, \mathcal{M}_T\}$ denote the sets of all trees and all terminal node parameters, respectively. Similarly, define $\mathcal{T}_{(-j)} = \{\mathcal{T}_t : t \neq j\}$ and $\mathcal{M}_{(-j)} = \{\mathcal{M}_t : t \neq j\}$ the set of all trees and node-level parameters, excluding quantities associated with the j-th tree. Thus, the posterior distribution can be written as

$$\begin{aligned} p(\mathcal{T},\mathcal{M},\sigma^2|\mathbf{X},\mathbf{y}) &\propto p(\mathbf{y}|\mathbf{X},\mathcal{T},\mathcal{M},\sigma^2)p(\mathcal{M}|\mathcal{T})p(\sigma^2)p(\mathcal{T}) \qquad (2.15)\\ &\propto \left[\prod_{t=1}^{T}\prod_{\ell=1}^{b_t}\prod_{i\in\mathcal{P}_{t\ell}} p(y_i|\mathbf{X},\mathcal{T}_t,\mathcal{M}_t,\sigma^2)\right]\left[\prod_{t=1}^{T}\left[\prod_{\ell=1}^{b_t} p(\mu_{t\ell}|\mathcal{T}_t)\right]p(\mathcal{T}_t)\right]p(\sigma^2)\\ &\propto \left[\prod_{i=1}^{n}\mathrm{N}\left(y_i|\sum_{t=1}^{T} g(\mathbf{x}_i;\mathcal{T}_t,\mathcal{M}_t),\sigma^2\right)\right]\left[\prod_{t=1}^{T}\left[\prod_{\ell=1}^{b_t}\mathrm{N}(\mu_{t\ell}|0,\sigma_\mu^2)\right]p(\mathcal{T}_t)\right]\mathrm{IG}(\sigma^2|\nu,\lambda), \end{aligned}$$

where $\mathcal{P}_{t\ell}$ denotes the set of splitting rules that define terminal node ℓ of tree t and $p(\mathcal{T}_t)$ is given in (2.7). The expression above does not have a known distributional form, but it is possible to draw samples from (2.15) by decomposing it. First, it is possible to sequentially sample, one at a time, from $p(\mathcal{T}_t,\mathcal{M}_t|\mathbf{y},\mathcal{T}_{(-j)}$, $\mathcal{M}_{(-j)},\sigma^2)$, for $t=1,\cdots,T$. This can be carried out by noticing that sampling from $p(\mathcal{T}_t,\mathcal{M}_t|\mathbf{y},\mathcal{T}_{(-j)}$, $\mathcal{M}_{(-j)},\sigma^2)$ is equivalent to sample from $p(\mathcal{T}_t,\mathcal{M}_t|\mathbf{R}_t,\sigma^2)$, where $\mathbf{R}_t=\mathbf{y}-\sum_{j\neq t}^{T} g(\mathbf{x}_i;\mathcal{T}_j,\mathcal{M}_j)$. The vector of partial residuals $\mathbf{R}_t$ works like the response variable and takes into account the dependence on the other trees. In addition, it is possible to sample from $p(\mathcal{T}_t,\mathcal{M}_t|\mathbf{R}_t,\sigma^2)$, which does not present a closed-form distribution either, by further decomposing it into

$$p(\mathcal{M}_t|\mathbf{R}_t,\mathcal{T}_t,\sigma^2) \propto p(\mathbf{R}_t|\mathbf{X},\mathcal{T}_t,\mathcal{M}_t,\sigma^2)p(\mathcal{M}_t|\mathcal{T}_t), \qquad (2.16)$$

$$p(\mathcal{T}_t|\mathbf{R}_t,\sigma^2) \propto \int p(\mathcal{T}_t|\mathbf{R}_t,\mathcal{M}_t,\sigma^2)p(\mathcal{M}_t|\mathcal{T}_t)d\mathcal{M}_t. \qquad (2.17)$$

To sample from (2.16) is straightforward as the $\mu_{t\ell}\in\mathcal{M}_t$ are i.i.d and follow a normal distribution, whereas for (2.17) a Metropolis-Hastings step is required. Then, after samples of all T trees are obtained as well as for their terminal node parameters, it is possible to update σ^2 from $p(\sigma^2|\mathcal{T},\mathcal{M},\mathbf{y})$.

In Algorithm 1, we present the structure of the BART model for one MCMC iteration considering that the response variable is continuous. Given the design matrix $\mathbf{X}$, the response variable $\mathbf{y}$, and the user-specified hyper-parameters, a tree is learned by a grow, prune, change, or swap step, where each step is proposed with probability of 0.25, 0.25, 0.4, and 0.1, respectively. A common practice is to initially set all trees to stumps. Then, the proposed tree $\mathcal{T}_t^\star$ is compared to its previous version $\mathcal{T}_t$ via a Metropolis-Hastings step, and it is accepted with probability $\alpha(\mathcal{T}_t,\mathcal{T}_t^\star)$. Given the tree structure, the node-level parameters $\mu_{t\ell}$ are updated. After this process is repeated for all T trees, the residual variance is updated as well as the predicted values. For more than one iteration, it suffices to repeat steps 3-11 until convergence is achieved.

Algorithm 1 BART model for regression

```
1: **Input**: $\mathbf{y}$, $\mathbf{X}$, and number of trees $T$.
2: **Initialise**: $\{\mathcal{T}_t\}_1^T$ and set all hyperparameters of the prior distributions.
3: **for** ($t = 1$ to $T$) **do**
4:     Compute $\mathbf{R}_t = \mathbf{y} - \sum_{j \neq t}^{T} g\left(\mathbf{X}, \mathcal{M}_j, \mathcal{T}_j\right)$.
5:     Propose a new tree $\mathcal{T}_t^\star$ by a grow, prune, change, or swap move.
6:     Compare the current ($\mathcal{T}_t$) and proposed ($\mathcal{T}_t^\star$) trees via Metropolis-
       Hastings with
       $$\alpha\left(\mathcal{T}_t, \mathcal{T}_t^\star\right) = \min\left\{1, \frac{p\left(\mathcal{T}_t^\star \mid \mathbf{R}_t, \sigma^2\right) q\left(\mathcal{T}_t^\star \to \mathcal{T}_t\right)}{p\left(\mathcal{T}_t \mid \mathbf{R}_t, \sigma^2\right) q\left(\mathcal{T}_t \to \mathcal{T}_t^\star\right)}\right\}.$$
7:     Sample $u \sim$ Uniform $(0,1)$: If $\alpha\left(\mathcal{T}_t, \mathcal{T}_t^\star\right) < u$, set $\mathcal{T}_t = \mathcal{T}_t$, otherwise
       set $\mathcal{T}_t = \mathcal{T}_t^\star$.
8:     Update all node-level parameters $\mu_{t\ell}$ via $p(\mu_{t\ell} | \mathbf{R}_t, \mathcal{T}_t, \sigma^2)$, for $\ell = 1, \ldots, b_t$.
9: **end for**
10: Update $\sigma^2$ via $p(\sigma^2 | \mathcal{T}, \mathcal{M}, \mathbf{y})$.
11: Update $\hat{\mathbf{y}} = \sum_{t=1}^{T} g\left(\mathbf{X}, \mathcal{M}_t, \mathcal{T}_t\right)$.
```

BART can also be extended to deal with binary or multi-class (Kindo et al., 2016b; Murray, 2021) response via the data augmentation approach of Albert and Chib (1993). For $y_i \in \{0, 1\}$, the strategy consists of introducing a latent variable $z_i \sim \mathrm{N}(\sum_{t=1}^{T} g(\mathbf{x}_i; \mathcal{T}_t, \mathcal{M}_t), 1)$, for $i = 1, \cdots, n$, and defining $y_i = 1$ if $z_i > 0$, $y_i = 0$ otherwise. Under this formulation, we assume that $p(y_i = 1 | \mathbf{x}_i) = \Phi\left(\sum_{t=1}^{T} g\left(\mathbf{X}, \mathcal{M}_t, \mathcal{T}_t\right)\right)$, where $\Phi(\cdot)$ denotes the standard normal cumulative distribution function, and there is no need to estimate the residual variance σ^2 as it is set to 1. Furthermore, the conditional distribution of $z_i | y_i$ is a truncated-normal distribution, conditioned on the value of y_i. For instance, $z_i | [y_i = 1] \sim \mathrm{N}_{(0,\infty)}(\sum_{t=1}^{T} g(\mathbf{x}_i; \mathcal{T}_t, \mathcal{M}_t), 1)$ and $z_i | [y_i = 0] \sim \mathrm{N}_{(-\infty,0)}(\sum_{t=1}^{T} g(\mathbf{x}_i; \mathcal{T}_t, \mathcal{M}_t), 1)$, where $\mathrm{N}_{(a,b)}(\cdot)$ denotes a truncated-normal constrained to the interval (a, b).

With a few changes, it is possible to adapt Algorithm 1 to deal with a binary response. First, the prior distribution $p(\mu_{t\ell} | \mathcal{T}_t)$ needs to be changed, and Chipman et al. (2010) recommend setting it up so that the induced prior on $\hat{\mathbf{y}}$ assigns high probability to the interval $\Phi(-3)$ and $\Phi(3)$. Second, the partial residuals $\mathbf{R}_t$ in line 4 now take into account z_i rather than y_i, and are defined as $\mathbf{R}_t = \mathbf{z} - \sum_{j \neq t}^{T} g\left(\mathbf{X}, \mathcal{M}_j, \mathcal{T}_j\right)$. Third, in place of the update of σ^2 in line 10, which can be removed since $\sigma^2 = 1$, the update of the latent variables z_i can be carried out via $p(z_i|y_i)$. Finally, the predicted values in line 11 are calculated as $\hat{\mathbf{y}} = \Phi\left(\sum_{t=1}^{T} g\left(\mathbf{X}, \mathcal{M}_t, \mathcal{T}_t\right)\right)$.

2.4 Chapter summaries

In this Section, we go through some of the aforementioned limitations of the BART model and provide an overview of how we tackle them. First, we start by stating how we approach the issue of estimating linear effects more efficiently. Second, we present how we make the AMMI model more flexible by replacing its bi-linear term with a BART model. Finally, we outline BART-based semi-parametric models and how we overcome some of their limitations.

2.4.1 Chapter 3: MOTR-BART

We recall that the BART model uses piecewise constants to estimate any association that may exist between a response and a set of covariates. For instance, to marginally approximate a non-linear effect, BART does not require, as regression models usually do, polynomial terms and/or basis expansions (e.g., splines). Rather, BART splits on the covariate itself several times, and this can take place either in one or multiple trees, or even multiple branches within a given tree. In the case of a two-way interaction, the same rationale applies in the sense that BART splits on the two covariates that interact but the splits occur in the same tree.

To estimate a linear effect, BART also requires many splits on the same covariate. When the number of covariates p is small, a linear effect is reasonably approximated, especially if the number of trees T is sufficiently large. However, when p is large, to estimate linear and non-linear effects becomes challenging due to the uniform specification, where a covariate x_j is sampled to form a splitting rule with probability $s_j = 1/p$ for all j. To effectively identify the covariates that help predict the response when p is large, Linero (2018) proposes to replace the uniform distribution over the vector of splitting probabilities $s = (s_1, s_2, \cdots, s_p)$ by a Dirichlet distribution. The aim is to favour the covariates that are part of the structure of the trees over those covariates that are irrelevant.

Our novel MOTR-BART model can be viewed as an extension to existing models from two perspectives: i) an extension to BART with linear regression models at the terminal nodes, or ii) an extension to the Bayesian treed regression model (Chipman et al., 2002) to a setting where an ensemble of trees is used rather than a single tree. In our novel approach, we adopt the Dirichlet prior from Linero (2018) and consider that

$$y_i = \sum_{t=1}^{T} g(\mathbf{x}_i, \mathcal{T}_t, \mathcal{B}_t) + \epsilon_i,$$

where $\epsilon_i \sim \mathrm{N}(0, \sigma^2)$ and $g(\mathbf{x}_i, \mathcal{T}_t, \mathcal{B}_t) = \mathbf{X}_{t\ell}\boldsymbol{\beta}_{t\ell}$. We define $\mathbf{X}_{t\ell}$ as a subset of the design matrix $\mathbf{X}$ containing only the observations that belong to terminal node ℓ of tree t. The set of terminal node parameters is represented by $\mathcal{B}_t = (\boldsymbol{\beta}_{t1}, \cdots, \boldsymbol{\beta}_{tb_t})$, where b_t denotes the number of terminal nodes in tree t. In addition, we consider that $\boldsymbol{\beta}_{t\ell}$ is a q-dimensional row vector and $\mathbf{X}_{t\ell}$ is a matrix of dimension $n_{t\ell} \times q$, with $q = p_{t\ell} + 1$, where the additional dimension accounts for an intercept.

In terms of priors, we assume that $\boldsymbol{\beta}_{t\ell} \sim \mathrm{MVN}\left(\mathbf{b}, \sigma^2\mathbf{V}\right)$, where $\mathrm{MVN}(\cdot, \cdot)$ denotes a multivariate normal distribution. We follow Chipman et al. (2010) and also scale the response to lie between -0.5 and 0.5, which allows to set $\mathbf{b} = \mathbf{0}$. To facilitate the specification of the variance-covariance matrix $\mathbf{V}$, we standardise the covariates so that they have mean zero and standard deviation one. With this, we set $\mathbf{V}$ in two ways. First, we consider that $\mathbf{V} = \sigma_b^2\mathbf{I}_q$, where $\mathbf{I}_q$ is a q-dimensional identity matrix and $\sigma_b^2 = T^{-1}$ is responsible for balancing the importance of each tree by shrinking the components within $\boldsymbol{\beta}_{t\ell}$ towards zero in the spirit of the prior adopted by Chipman et al. (2010). Second, we set intercept- and slope-specific variances through $\mathbf{V}_{1,1} = \sigma_{\beta_0}^2/T$ and $\mathbf{V}_{j+1,j+1} = \sigma_\beta^2/T$, and place conjugate inverse gamma priors on both parameters. In this setting, samples from the conditional posterior distributions for $\sigma_{\beta_0}^2$ and σ_β^2 can be obtained via Gibbs updates. For σ^2 and $\mathcal{T}_1 \ldots \mathcal{T}_t$, we make use of those priors adopted by Chipman et al. (2010), which were introduced in Section 2.3.

Figure 2.3 visually illustrates the difference between BART and MOTR-BART. Now, rather than a piecewise constant at the terminal nodes, the predicted values are based on the local linear predictors. Thus, for any two observations that fall into the same terminal node, their predicted values will be different as long as the values of the covariates in $\mathbf{X}_{t\ell}$ are different. Under MOTR-BART, linear effects do not require as many splits as in BART, which in turn significantly reduces the number of trees used in the ensemble. Furthermore, replacing the piecewise constants with linear predictors allows us to capture/explore more complex structures at the terminal node level, which generally leads to improvements in the predictive performance.

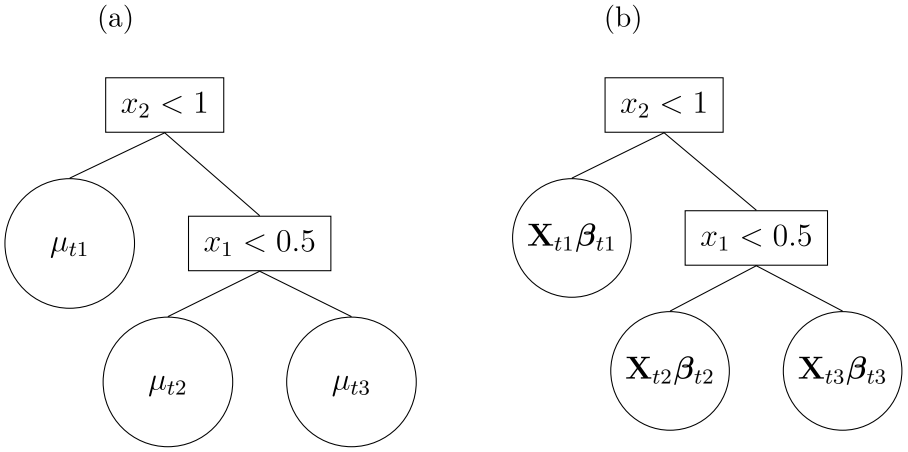

(a)
(b)
$x_2 < 1$
μ_{t1}
$x_1 < 0.5$
μ_{t2}
μ_{t3}
$x_2 < 1$
$\mathbf{X}_{t1}\boldsymbol{\beta}_{t1}$
$x_1 < 0.5$
$\mathbf{X}_{t2}\boldsymbol{\beta}_{t2}$
$\mathbf{X}_{t3}\boldsymbol{\beta}_{t3}$

Figure 2.3: Example of trees under (a) BART and (b) MOTR-BART. In panel (a), the observations which belong to the same node have one predicted value, while in panel (b) the predicted values are calculated based on the local linear predictors. We remark that both BART and MOTR-BART predict a univariate response by using a set of trees.

Finally, we tackle the specification of covariates in the linear predictors of each terminal node in two ways. First, we specify all linear predictors of a tree using the covariates used in the splitting rules of the corresponding tree. Second, we set the linear predictors in a given tree based on their ancestor nodes. Both approaches are clarified in further details in Chapter 3.

2.4.2 Chapter 4: AMBARTI

The additive main effects multiplicative interactions models (AMMI; Mandel, 1971) are commonly used to analyse crop data. These models consider that a phenotypic response y_{ij} can be predicted based on genetic (g_i) and environmental (e_j) factors as

$$y_{ij} = \mu + g_i + e_j + \sum_{q=1}^{Q} \lambda_q \gamma_{iq} \delta_{jq} + \epsilon_{ij}, \ \ \epsilon_{ij} \sim \mathrm{N}(0, \sigma^2). \tag{2.18}$$

The first three components in (2.18) are referred to as the linear term, whereas the component involving the sum over q is called the bi-linear term. In the context of plant-based genetics, the genetic factor is called genotype and, in general, the response variable y_{ij} represents the amount harvested for $i = 1, \cdots, I$ genotypes which were cultivated in $j = 1, \cdots, J$ environments.

The components g_i and e_j represent the individual effects of each genotype and environment, respectively, while the bi-linear term accounts for the interaction between them. Were the model in (2.18) parameterised as $y_{ij} = \mu + g_i + e_j + (g_i \times e_j)$, there would not be enough degrees of freedom to estimate all parameters in the model, since the AMMI model cannot handle replicates for any combination of g_i and e_j. With this, the bi-linear term in (2.18) requires some constraints on the three sets of parameters (i.e., λ_q, γ_{iq}, and δ_{jq}) so that the interaction effects between g_i and e_j are adequately estimated.

In terms of estimation, the parameters in the linear term are obtained using ordinary least squares as in a linear regression model. In contrast, the parameters in the bi-linear term are estimated via a singular value decomposition performed on a residual $I \times J$ matrix, where each entry is given by $r_{ij} = y_i - \hat{\mu} - \hat{g}_i - \hat{e}_j$. Thus, the λ_q are the singular values of the residual matrix and γ_{iq} and δ_{jq} are the left and right singular vectors.

In our novel AMBARTI approach, we extend the AMMI model to allow for richer interactions between genotypes and environments by replacing the bi-linear term with a BART model as

$$y_{ij} = \mu + g_i + e_j + \sum_{t=1}^{T} h(\mathbf{x}_{ij}, \mathcal{M}_t, \mathcal{T}_t) + \epsilon_{ij}, \epsilon_{ij} \sim \mathrm{N}(0, \sigma^2), \tag{2.19}$$

where $\mathbf{x}_{ij}$ denotes a row of the design matrix $\mathbf{X}$ associated with genotype i and environment j which contains only dummy variables associated with the g_i and e_j.

The model in (2.19) can also be seen as a semi-parametric BART (SSP-BART) model, in which the response variable is predicted through the sum of a linear predictor and a BART model; see subsection 2.4.3 and Chapter 5 for further details about SSP-BART. Nonetheless, AMBARTI is substantially different from SSP-BART. First, we add two new moves to the tree-generation process in the BART model. This change is necessary in AMBARTI by the fact that its BART component is expected to solely estimate interactions between the g_i and e_j, since the linear term is responsible for the main effects. The rationale behind the double grow and the double prune moves is to make sure that BART will always have either stumps or trees with at least one g_i and one e_j. Without the double moves, the trees from BART could have splits only on the g_i or only on the e_j, which is not desirable since the role of the bi-linear in the AMMI model is to exclusively induce interaction between g_i and e_j. Regarding the change and swap moves, no modifications were made/needed, although we introduced validity checks in our implementation to guarantee that the proposed trees have a valid structure.

Second, we use the response variable only to update the parameter estimates in the linear term, as opposed to the full residual $\tilde{\mathbf{r}} = \mathbf{y} - \sum_{t=1}^{T} h(\mathbf{X}, \mathcal{M}_t, \mathcal{T}_t)$. This specification is motivated by the two-stage procedure used to estimate the parameters in AMMI. We recall that in the AMMI model, the linear term is obtained from least squares and then the bi-linear term is estimated based on a singular value decomposition performed on the residuals; i.e., since the bi-linear term estimates have been obtained, the estimation procedure is finished and the estimates for the λ_q, γ_{iq}, and δ_{jq} are not fed back into the model. From this perspective, AMBARTI is equivalent to the cut feedback model described in Plummer (2015), since the linear term is uniquely updated taking into account the response, while the BART component is fed back by the linear term. However, we show the validity of the posterior sampling under the 'naive cut algorithm' according to Plummer (2015).

Finally, we introduce new visualisations that help easily assess which are the best genotypes and environments taking into account both the main and interaction effects. Motivated by the data from a set of experiments carried out in Ireland from 2010 to 2019, we present the new visualisations as a more complete alternative to biplots (Gabriel, 1971), which are a visual tool frequently utilised to identify the best genotypes/environments interactions based only on the bi-linear term of the AMMI model. The new plots are especially useful for professionals with no quantitative background as they help determine which genotypes/environments should be prioritised in order to maximise the production of a crop.

2.4.3 Chapter 5: CSP-BART

The aforementioned semi-parametric BART models (Zeldow et al., 2019; Tan and Roy, 2019) offer interpretability and predictive performance by combining a linear predictor and a BART model. Here, the design matrix $\mathbf{X}$ is split into two subsets $\mathbf{X}_1$ and $\mathbf{X}_2$, where the first contains the covariates of primary interpretational interest and the second consists of a subset of variables that are not of primary interpretational interest but may also be important to predict the response variable. Under the SSP-BART model, a univariate continuous response variable is predicted as

$$y_i = \mathbf{x}_{i1}\boldsymbol{\beta} + \sum_{t=1}^{T} g(\mathbf{x}_{i2}, \mathcal{M}_t, \mathcal{T}_t) + \epsilon_i,$$

where $\epsilon_i \sim \mathrm{N}(0, \sigma^2)$ and $\mathbf{x}_{i1}$ and $\mathbf{x}_{i2}$ denote the i-th row of $\mathbf{X}_1$ and $\mathbf{X}_2$, respectively.

The advantage of SSP-BART over BART and traditional modelling techniques, such as GLMs and/or generalised additive models (GAMs; Hastie and Tibshirani, 1990; Wood, 2017), is two-fold. First, SSP-BART offers more interpretability than BART since the effects of the covariates of primary interest can be explicitly quantified via a linear predictor, while its BART component deals with covariates of non-primary interest. Second, SSP-BART provides more flexibility than GLMs/GAMs as interactions and non-linearities are naturally estimated by the BART component without pre-specification, which does not take place in GLMs nor GAMs as they require the specification of all main and interactions effects in the model.

One of the key limitations of the SSP-BART is the premise that $\mathbf{X}_1$ and $\mathbf{X}_2$ cannot share covariates (i.e., $\{\mathbf{X}_1 \cap \mathbf{X}_2\} = \emptyset$), which is introduced as an attempt to avoid undercoverage and bias. Without this assumption, the two components in the model (linear predictor and BART) would try to estimate the marginal effects associated with the covariates common to both $\mathbf{X}_1$ and $\mathbf{X}_2$, and this would lead to confounding/identifiability issues. However, by assuming that $\mathbf{X}_1$ and $\mathbf{X}_2$ are mutually exclusive, SSP-BART prevents the BART component from estimating interactions among covariates in $\mathbf{X}_1$ and between the covariates in $\mathbf{X}_1$ and $\mathbf{X}_2$.

In our novel CSP-BART, we allow for sharing covariates across the components by adding the aforementioned double grow and double prune moves to the BART model. We recall that the doubles moves in AMBARTI are exclusively to induce interactions between the g_i and e_j, since the design matrix $\mathbf{x}_{ij}$ in (2.19) contains dummy variables representing the g_i and e_j only; that is, in the AMBARTI model the only covariates are the genotypes and environments, which are both present in the linear predictor and BART component. In contrast, the double grow and double prune moves in CSP-BART aim to induce interactions either across the covariates in $\mathbf{X}_1$ or between the covariates common to $\mathbf{X}_1$ and $\mathbf{X}_2$, since in CSP-BART we allow $\{\mathbf{X}_1 \cap \mathbf{X}_2\} \neq \emptyset$, or even $\mathbf{X}_1 \subseteq \mathbf{X}_2$. We point out that the interactions among covariates which belong to $\mathbf{X}_2$ only (i.e., the remaining possible interactions) are accounted for by the standard moves of the BART component as in SSP-BART.

In Figure 2.4, we illustrate how the new moves work in a tree. In panel (b), we see how a tree looks like after a 'single' grow move where the splitting rule is based on x_2. Under the CSP-BART model, if $x_2 \in \mathbf{X}_2$ and $x_2 \notin \mathbf{X}_1$, the single grow move can be applied, as it generates a valid tree. However, if $x_2 \in \{\mathbf{X}_1 \cap \mathbf{X}_2\}$, the tree in panel (b) is not valid, since the main effect of x_2 is now estimated in both the linear predictor and BART.

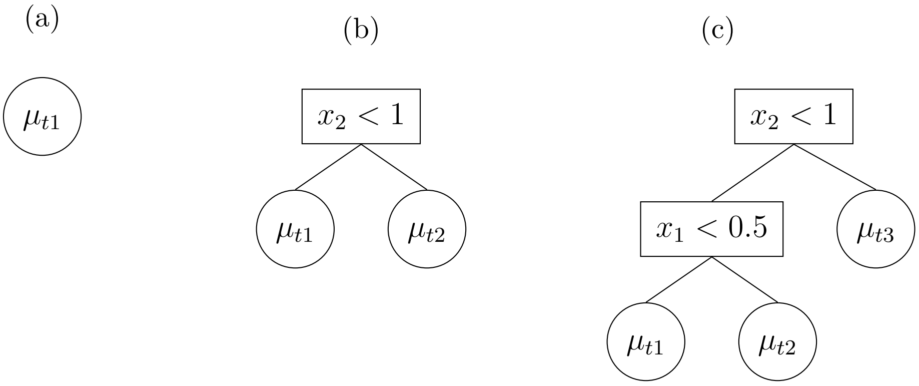


Figure 2.4: Example of a tree in the BART component of the CSP-BART model in different instances. Panel (a) shows a stump. Panel (b) illustrates a tree after a single grow move. In panels (b) and (c), we assume that $x_2 \in \{\mathbf{X}_1 \cap \mathbf{X}_2\}$ and that x_1 can be either in $\mathbf{X}_2$ or $\{\mathbf{X}_1 \cap \mathbf{X}_2\}$. From panels (a) to (c), a double grow move is shown. In contrast, from panels (c) to (a) the double prune move is illustrated. In panel (c), μ_{t3} is set to zero to avoid identifiability issues since $x_2 \in \{\mathbf{X}_1 \cap \mathbf{X}_2\}$.

To circumvent the issue of introducing bias/confounding into the model, we take the tree in panel (b) and grow it again in order to have a valid tree. After employing a double grow move, panel (c) shows a valid tree containing $x_2 \in \{\mathbf{X}_1 \cap \mathbf{X}_2\}$ and x_1, which can be either in $\mathbf{X}_2$ or $\{\mathbf{X}_1 \cap \mathbf{X}_2\}$. Conversely, the double prune move is applied to prevent a tree from having an invalid topology after a single prune move. For instance, if $x_2 \in \{\mathbf{X}_1 \cap \mathbf{X}_2\}$ and the tree in panel (c) is single pruned, the resulting tree in panel (b) would be invalid. To make sure the corresponding tree is valid, it is necessary to prune the tree in panel (b) again, thus getting back to the tree in panel (a). The rationale behind the double moves is to force BART to induce interactions whenever a covariate common to $\mathbf{X}_1$ and $\mathbf{X}_2$ is in a tree by itself.

We point out that the addition of the double moves to the tree-generation process also requires some changes in the prior for the terminal node parameters. For instance, in Figure 2.4 panel (c), even with the double grow move being employed, only the left-most nodes result from interactions; that is, the right-most terminal node is based on x_2 only, which would still bring some bias into the effect of x_2 if no change on the prior for μ_{t3} is done. Details of the modifications made to the priors are provided in the Chapter 5. Furthermore, although the change and swap moves remain the same, we introduce additional validity checks on the structure of the trees to guarantee that only valid structures are kept.

CHAPTER 3

Bayesian additive regression trees with model trees

Bayesian additive regression trees (BART) is a tree-based machine learning method that has been successfully applied to regression and classification problems. BART assumes regularisation priors on a set of trees that work as weak learners and is very flexible for predicting in the presence of non-linearities and low-order interactions. In this paper, we introduce an extension of BART, called model trees BART (MOTR-BART), that considers a linear function at node level instead of a constant. In MOTR-BART, rather than having a unique value at node level for the prediction, a linear predictor is estimated considering the covariates that have been used as the split variables in the corresponding tree. In our approach, local linearities are captured more efficiently and fewer trees are required to achieve equal or better performance than BART. Via simulation studies and real data applications, we compare MOTR-BART to its main competitors. R code for MOTR-BART implementation is available at https://github.com/ebprado/MOTR-BART.

3.1 Introduction

Bayesian additive regression trees (BART) is a statistical method proposed by Chipman et al. (2010) that has become popular in recent years due to its competitive performance on regression and classification problems, when compared to other supervised machine learning methods, such as random forests (RF; Breiman, 2001) and gradient boosting (GB; Friedman, 2001). BART differs from other tree-based methods as it controls the structure of each tree via a prior distribution and generates the predictions via an iterative Bayesian backfitting MCMC algorithm. In practice, BART can be used for predicting a continuous/binary response variable through R packages, such as `dbarts` (Dorie, 2020), `BART` (McCulloch et al., 2020), and `bartMachine` (Kapelner and Bleich, 2016).

In essence, BART is a non-parametric Bayesian algorithm that generates a set of trees by choosing the covariates and the split-points at random. To generate the predicted values for each terminal node, the normal distribution is adopted as the conditional probability distribution and prior distributions are placed on the structure of the trees, predicted values, and residual variance. Through a Bayesian backfitting MCMC algorithm, the predictions from each tree are obtained by combining Gibbs sampling (Gelfand and Smith, 1990) and Metropolis-Hastings (Metropolis et al., 1953; Hastings, 1970) steps. The final prediction is then calculated as the sum of the predicted values over all trees. In parallel, samples from the posterior distributions of the quantities of interest are naturally generated along the MCMC iterations.

In this paper, we introduce the algorithm MOTR-BART, which combines model trees (Quinlan, 1992) with BART to deal with local linearity at node levels. In MOTR-BART, rather than estimating a constant as the node parameter as BART does, for each terminal node a linear predictor is estimated, which includes only the covariates that have been used as a split in the corresponding tree. With this approach, we aim to capture linear associations between the response and covariates and then improve the final prediction. We observe that MOTR-BART requires fewer trees to achieve equal or better performance than BART. Through simulation experiments that consider different number of observations and covariates, MOTR-BART outperforms its main competitors in terms of RMSE on out-of-sample data, even using fewer trees. In the real data applications, MOTR-BART is competitive compared to BART and other tree-based methods.

This paper is organised as follows. In Section 3.2, we briefly introduce BART, some related works, and model trees. Section 3.3 presents the mathematical details of BART and how it may be implemented in the regression context. In Section 3.4, we introduce the MOTR-BART, providing the mathematical expressions needed for regression and classification. Section 3.5 shows comparisons between MOTR-BART and other algorithms via simulated scenarios and real data applications. Finally, in Section 3.6, we conclude with a discussion.

3.2 Tree-based methods

3.2.1 Related work

BART considers that a univariate response variable can be approximated by a sum of predicted values from a set of trees as $\hat{y} = \sum_{t=1}^{T} g(\mathbf{X}; \mathcal{M}_t, \mathcal{T}_t)$, where $g(\cdot)$ is a function that assigns a predicted value based on $\mathbf{X}$ and $\mathcal{T}_t$, $\mathbf{X}$ is the design matrix, $\mathcal{M}_t$ is the set of predicted values of the tree t, and $\mathcal{T}_t$ represents the structure of the tree t. In BART, a tree $\mathcal{T}_t$ can be modified through four moves (growing, pruning, changing, or swapping), and the splitting rules used to create the terminal/internal nodes are randomly chosen. To sample from the full conditional distribution of $\mathcal{T}_t$, the Metropolis-Hastings algorithm is used. Further, each component $\mu_{t\ell} \in \mathcal{M}_t$ is sampled from its full conditional via a Gibbs sampling step. Then, the final prediction is calculated by adding up the values of $\mu_{t\ell}$ from all the T trees. Further details are given in Section 3.3.

BART's versatility has made it an attractive option with applications in credit risk modelling (Zhang and Härdle, 2010), identification of subgroup effects in clinical trials (Sivaganesan et al., 2017; Schnell et al., 2016), competing risk analysis (Sparapani et al., 2019), survival analysis of stem cell transplantation (Sparapani et al., 2016), proteomic biomarker discovery (Hernández et al., 2015), and causal inference (Hill, 2011; Green and Kern, 2012; Hahn et al., 2020). In this context, many extensions have been proposed, such as BART for estimating monotone and smooth surfaces (Starling et al., 2019, 2020; Linero and Yang, 2018), categorical and multinomial data (Murray, 2021; Kindo et al., 2016b), high-dimensional data (Hernández et al., 2018; He et al., 2019; Linero, 2018), zero-inflated and semi-continuous responses (Linero et al., 2020), heterocedastic data (Pratola et al., 2020), BART with quantile regression and varying coefficient models (Kindo et al., 2016a; Deshpande et al., 2020), among others. Recently, some papers have developed theoretical aspects related to BART (Linero, 2017b; Ročková and van der Pas, 2020; Ročková and Saha, 2019; Linero and Yang, 2018).

Some of the works mentioned above are somewhat related to MOTR-BART. For instance, Linero and Yang (2018) introduce the soft BART (SBART) in order to provide an approach suitable for both estimating a target smooth function and dealing with sparsity. In SBART, the observations are not allocated deterministically to the terminal nodes, as it is commonly done in the conventional trees. Instead, the observations are assigned to the terminal nodes based on a probability measure, which is a function of a bandwidth parameter and of the distance between the values of the covariates and the cut-offs defined by the splitting rules. Through empirical and theoretical results, they show that SBART is capable to smoothly approximate linear and non-linear functions as well as that its posterior distribution concentrates, under mild conditions, at the minimax rate. The main differences between MOTR-BART and SBART are: i) MOTR-BART does not use the idea of soft trees, where the observations are assigned to the terminal based on a probability measure, and ii) MOTR-BART uses linear predictors rather than constants to locally generate the predictions at the node level.

In this sense, Starling et al. (2020) propose the BART with Targeted Smoothing (tsBART) by introducing smoothness over a covariate of interest. In their approach, rather than predicting a step function as the standard BART, univariate smooth functions of a certain covariate of interest are used to generate the node-level predictions. In tsBART, they place a Gaussian process prior over the smooth function associated with each terminal node and learn the trees using all available covariates, apart from the one over which they wish to introduce the smoothness. Although tsBART and MOTR-BART have some similarities, since both do not base their predictions on step functions and both aim to provide more flexibility at the node-level predictions, they differ as MOTR-BART allows for more than one covariate to be used in the linear predictors and does not assume a Gaussian process prior on each linear predictor.

In addition, Deshpande et al. (2020) propose an extension named VCBART that combines varying coefficient models and BART. In their approach, rather than approximating the response variable itself, each covariate effect in the linear predictor is estimated using BART. They also provide theoretical results about the near minimax optimal rate associated with the posterior concentration of the VCBART considering non-i.i.d errors. Although the linear model is a particular case of the varying coefficients model, MOTR-BART and VCBART are structurally different. For instance, VCBART considers that a univariate response variable can be approximated via an overall linear predictor in which the coefficients are estimated via BART. In contrast, MOTR-BART approximates the response by estimating a linear predictor for each terminal node in each tree, where normal priors are placed on the coefficients in order to estimate them.

Regarding non-Bayesian methods, we highlight the algorithms introduced by Friedberg et al. (2020) and Künzel et al. (2022), named local linear forests (LLF) and linear random forests (LRF), respectively. In their work, RF-based algorithms are proposed, where the predictions for each terminal node are generated from a local ridge regression. Furthermore, the LLF algorithm also provides a point-wise confidence interval based on the RF delta method proposed by Athey et al. (2019) and theoretical results related to asymptotic consistency and rates of convergence of the forest.

3.2.2 Model trees and treed models

Quinlan (1992) introduced the term model trees when proposing the M5 algorithm, which is a tree-based method that estimates a linear equation for each terminal node and then computes the final prediction based on piecewise linear models and a smoothing process. Initially introduced in the context of regression, extensions and generalisations for classification were presented by Wang et al. (1997) and Landwehr et al. (2005).

Unlike BART, RF, and GB, where multiple trees are generated to predict the outcome, the algorithm M5 generates only one tree. For the growing process, the variance reduction is adopted as the splitting criterion. When estimating the coefficients for the linear equation at a terminal node, the covariates are selected based on tests and, depending on their significance, the linear equation can be reduced to a constant, if all covariates do not show any significance. At the end, the predictions are averaged over the piecewise linear predictors from the terminal nodes along the path to the root.

In the context of Bayesian methods, Chipman et al. (2002) proposed the Bayesian treed model by combining the structure of a decision tree with linear models within each terminal node. This model uses one tree to predict either a binary or continuous response variable and many elements of the Bayesian CART of Chipman et al. (1998), such as the branching process prior on the tree topology, tree-generation moves, and the MCMC scheme used to sample from the posterior distribution. From this perspective, the MOTR-BART model proposed in this work can also be viewed as an extension to the Bayesian treed model to a setting where multiple trees are utilised to predict the response variable.

3.3 BART

Introduced by Chipman et al. (2010), BART is a tree-based machine learning method that considers that a univariate response variable $\mathbf{y} = (y_1, ..., y_n)^\top$ can be approximated by a sum-of-trees as

$$y_i = \sum_{t=1}^{T} g(\mathbf{x}_i; \mathcal{T}_t, \mathcal{M}_t) + \epsilon_i, \;\; \epsilon_i \sim \mathrm{N}(0, \sigma^2),$$

where $g(\mathbf{x}_i; \mathcal{T}_t, \mathcal{M}_t) = \mu_{t\ell}$ is a function that assigns a predicted value $\mu_{t\ell}$ based on $\mathbf{x}_i$, $\mathbf{x}_i = (x_{i1}, ..., x_{id})$ represents the i-th row of the design matrix $\mathbf{X}$, $\mathcal{T}_t$ is the set of splitting rules that defines the t-th tree, and $\mathcal{M}_t = (\mu_{t1}, ..., \mu_{tb_t})$ is the set of predicted values for all nodes in the tree t, with μ_{tb_t} representing the predicted value for the terminal node b_t. The splitting rules that define the terminal nodes for the tree t can be defined as partitions $\mathcal{P}_{t\ell}$, with $\ell = 1, ..., b_t$, and $g(\mathbf{x}_i; \mathcal{T}_t, \mathcal{M}_t) = \mu_{t\ell}$ for all observations $i \in \mathcal{P}_{t\ell}$, based on the values of $\mathbf{x}_i$.

In BART, each regression tree is generated as in Chipman et al. (1998) (see Figure 3.1) where, through an iterative Bayesian backfitting algorithm, a binary tree can be learned by four movements: grow, prune, change or swap. A new tree is proposed by one of these four movements and then compared to its previous version via a Metropolis-Hastings step on the partial residuals. In the growing process, a terminal node is randomly selected and then separated into two new nodes. Here, the covariate that is used to create the new terminal nodes is picked uniformly as is its associated split-point. In other words, the splitting rule is fully defined assuming the uniform distribution over both the set of covariates and the set of their split-points. During a prune step, a parent of two terminal nodes is randomly chosen and then its child nodes are removed. In the change move, an internal node (of any kind) is picked at random and its splitting rule is changed. In the swap process, a pair of parent-child internal nodes is randomly selected and the splitting rules of the two nodes are exchanged.

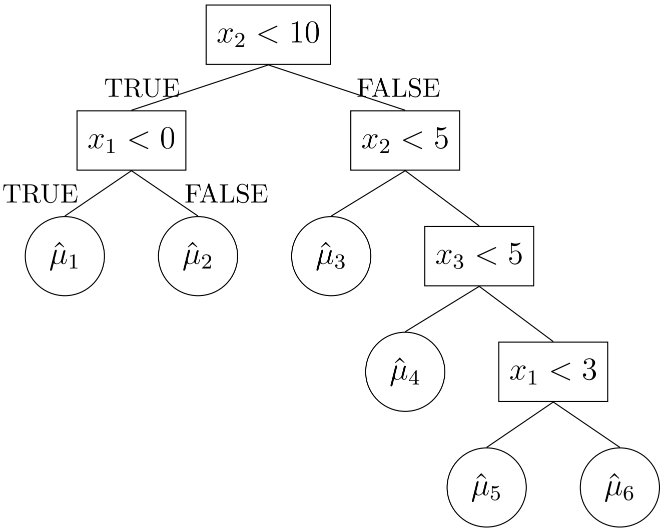


Figure 3.1: An example of a single tree generated by BART. In practice, BART generates multiple trees for which the predictions are added together. The covariates and split-points that define the terminal nodes are proposed uniformly and optimised via an MCMC algorithm. The quantities x_1, x_2, and x_3 represent covariates; $\hat{\mu}_\ell$ is the predicted value of node ℓ.

In order to control the depth of the tree, a regularisation prior is considered as

$$p(\mathcal{T}_t) = \prod_{\ell \in S_I} \left[\alpha(1+d_{t\ell})^{-\beta}\right] \times \prod_{\ell \in S_T} \left[1-\alpha(1+d_{t\ell})^{-\beta}\right], \tag{3.1}$$

where S_I and S_T denote the sets of indices of the internal and terminal nodes, respectively, $d_{t\ell}$ is the depth of node ℓ in tree t, $\alpha \in (0,1)$, and $\beta \geq 0$. Chipman et al. (2010) recommend $\alpha = 0.95$ and $\beta = 2$. In essence, $\alpha(1+d_{t\ell})^{-\beta}$ represents the probability of the node ℓ being internal at depth $d_{t\ell}$.

To estimate the terminal node parameters, $\mu_{t\ell}$, and residual variance, σ^2, conjugate priors are used:

$$\begin{aligned} \mu_{t\ell}|\mathcal{T}_t &\sim \mathrm{N}(0,\sigma_\mu^2),\\ y \qquad & \\ \sigma^2 &\sim \mathrm{IG}(\nu/2,\nu\lambda/2), \end{aligned}$$

where $\sigma_\mu = 0.5/(c\sqrt{T})$, $1 \leq c \leq 3$, $\mathrm{IG}(\cdot)$ denotes the inverse gamma distribution, and T is the number of trees. The division by T has the effect of reducing the predictive power of each tree and forcing each to be a weak learner. The joint posterior distribution of the trees and predicted values is given by

$$\begin{aligned} p((\mathcal{T},\mathcal{M}),\sigma^2|\mathbf{y},\mathbf{X}) &\propto p(\mathbf{y}|\mathbf{X},\mathcal{T},\mathcal{M},\sigma^2)p(\mathcal{M}|\mathcal{T})p(\mathcal{T})p(\sigma^2)\\ &\propto \left[\prod_{t=1}^{T}\prod_{\ell=1}^{b_t}\prod_{i:\mathbf{x}_i\in\mathcal{P}_{t\ell}} p(y_i|\mathbf{x}_i,\mathcal{T}_t,\mathcal{M}_t,\sigma^2)\right]\times\\ &\quad\times\left[\prod_{t=1}^{T}\prod_{\ell=1}^{b_t} p(\mu_{t\ell}|\mathcal{T}_t)p(\mathcal{T}_t)\right]p(\sigma^2). \end{aligned}$$

Chipman et al. (2010) initially decompose this joint posterior into two full conditionals. The first one generates all $\mu_{t\ell}$ for each tree $t = 1,...,T$, and is given by

$$p(\mathcal{T}_t,\mathcal{M}_t|\mathcal{T}_{(-t)},\mathcal{M}_{(-t)},\sigma^2,\mathbf{X},\mathbf{y}), \tag{3.2}$$

where $\mathcal{T}_{(-t)}$ represents the set of all trees without the component t; similarly for $\mathcal{M}_{(-t)}$. To sample from (3.2), Chipman et al. (2010) noticed that the dependence of the full conditional of $(\mathcal{T}_t,\mathcal{M}_t)$ on $\mathcal{T}_{(-t)},\mathcal{M}_{(-t)}$ is given by the partial residuals through

$$\mathbf{R}_t = \mathbf{y} - \sum_{k\neq t}^{T} g(\mathbf{X};\mathcal{T}_k,\mathcal{M}_k).$$

Thus, rather than depending on the other trees and their predicted values, the joint full conditional of $(\mathcal{T}_t, \mathcal{M}_t)$ may be rewritten as $p(\mathcal{T}_t, \mathcal{M}_t|\mathbf{R}_t, \sigma^2, \mathbf{X})$, with $\mathbf{R}_t$ acting like the response variable. This simplification allows us to sample from $p(\mathcal{T}_t, \mathcal{M}_t|\mathbf{R}_t, \sigma^2, \mathbf{X})$ in two steps:

a) Propose a new tree either growing, pruning, changing, or swapping terminal nodes via

$$\begin{aligned} p(\mathcal{T}_t|\mathbf{R}_t, \sigma^2) &\propto p(\mathcal{T}_t) \int p(\mathbf{R}_t|\mathcal{M}_t, \mathcal{T}_t, \sigma^2) p(\mathcal{M}_t|\mathcal{T}_t) d\mathcal{M}_t \\ &\propto p(\mathcal{T}_t) p(\mathbf{R}_t|\mathcal{T}_t, \sigma^2) \\ &\propto p(\mathcal{T}_t) \prod_{\ell=1}^{b_t} \left[\left(\frac{\sigma^2}{\sigma_\mu^2 n_{t\ell} + \sigma^2} \right)^{1/2} \right. \\ &\qquad \left. \times \exp\left(\frac{\sigma_\mu^2 \left[n_{t\ell} \bar{R}_\ell \right]^2}{2\sigma^2(\sigma_\mu^2 n_{t\ell} + \sigma^2)} \right) \right], \end{aligned}$$

where $\bar{R}_\ell = \sum_{i \in \mathcal{P}_{t\ell}} r_i / n_{t\ell}$, $r_i \in \mathbf{R}_t$, and $n_{t\ell}$ is the number of observations that belong to $\mathcal{P}_{t\ell}$. This sampling is carried out through a Metropolis-Hastings step, as the expression does not have a known distributional form.

b) Generate the predicted values $\mu_{t\ell}$ for all terminal nodes in the corresponding tree. As all $\mu_{t\ell}$ are independent from each other, it is possible to write $p(\mathcal{M}_t|\mathcal{T}_t, \mathbf{R}_t, \sigma^2) = \prod_{\ell=1}^{b_t} p(\mu_{t\ell}|\mathcal{T}_t, \mathbf{R}_t, \sigma^2)$. Hence,

$$\begin{aligned} p(\mu_{t\ell}|\mathcal{T}_t, \mathbf{R}_t, \sigma^2) &\propto p(\mathbf{R}_t|\mathcal{M}_t, \mathcal{T}_t, \sigma^2) p(\mu_{t\ell}) \\ &\propto \exp\left(-\frac{1}{2\sigma_\star^2} \left(\mu_{t\ell} - \mu_{t\ell}^\star \right)^2 \right), \end{aligned}$$

which represents the kernel of the following distribution:

$$\mathrm{N}\left(\frac{\sigma^{-2} \sum_{i \in \mathcal{P}_{t\ell}} r_i}{n_{t\ell}/\sigma^2 + \sigma_\mu^{-2}}, \frac{1}{n_{t\ell}/\sigma^2 + \sigma_\mu^{-2}} \right). \tag{3.3}$$

Then, after generating all predicted values for all trees, σ^2 can be updated based on

$$p(\sigma^2|\mathcal{T},\mathcal{M},\mathbf{X},\mathbf{y}) \propto p(\mathbf{y}|\mathbf{X},\mathcal{T},\mathcal{M},\sigma^2)p(\sigma^2)$$
$$\propto (\sigma^2)^{-\left(\frac{n+\nu}{2}+1\right)} \exp\left(-\frac{S+\nu\lambda}{2\sigma^2}\right), \tag{3.4}$$

where $S=\sum_{i=1}^{n}(y_i-\hat{y}_i)^2$ and $\hat{y}_i=\sum_{t=1}^{T} g(\mathbf{x}_i;\mathcal{T}_t,\mathcal{M}_t)$. The expression in (3.4) is an $\text{IG}((n+\nu)/2,(S+\nu\lambda)/2)$, and drawing samples from it is straightforward.

In Algorithm 2, we present the full structure of the BART model. Firstly, the response variable and design matrix are required. The trees, hyper-parameters, and the number of MCMC iterations M have to be initialised. Later, within each MCMC iteration, candidate trees ($\mathcal{T}_t^\star$) are sequentially generated, which might be accepted (or rejected) as the current trees with probability $\alpha(\mathcal{T}_t,\mathcal{T}_t^\star)$. After that, the predicted values $\mu_{t\ell}$ are generated for all terminal nodes. Finally, the final predictions and σ^2 are obtained.

Algorithm 2 BART model

```
**Input**: $\mathbf{y}$, $\mathbf{X}$, number of trees $T$, and number of MCMC iterations $M$.
**Initialise**: $\{\mathcal{T}_t\}_1^T$ and set all hyperparameters of the prior distributions.
**for** $(m = 1$ to $M)$ **do**
    **for** $(t = 1$ to $T)$ **do**
        Update $\mathbf{R}_t = \mathbf{y} - \sum_{j\neq t}^{T} g(\mathbf{X};\mathcal{T}_j,\mathcal{M}_j)$.
        Propose a new tree $\mathcal{T}_t^\star$ by a grow, prune, change or swap move, where
         each move has probability of 0.25, 0.25, 0.4, and 0.1, respectively.
        Compute $\alpha(\mathcal{T}_t,\mathcal{T}_t^\star) = \min\left\{1, \frac{p(\mathcal{T}_t^\star \mid \mathbf{R}_t,\sigma^2)q(\mathcal{T}_t^\star\to\mathcal{T}_t)}{p(\mathcal{T}_t \mid \mathbf{R}_t,\sigma^2)q(\mathcal{T}_t\to\mathcal{T}_t^\star)}\right\}$.
        Sample $u \sim \text{Uniform}(0,1)$: if $\alpha(\mathcal{T}_t,\mathcal{T}_t^\star) < u$, set $\mathcal{T}_t = \mathcal{T}_t$, otherwise
        set $\mathcal{T}_t = \mathcal{T}_t^\star$.
        **for** $(\ell = 1$ to $b_t)$ **do**
            Update $\mu_{t\ell}$ from $p(\mu_{t\ell}|\mathcal{T}_t,\mathbf{R}_t,\sigma^2)$.
        **end for**
    **end for**
    Update $\sigma^2$ via $p(\sigma^2|\mathcal{T},\mathcal{M},\mathbf{X},\mathbf{y})$.
    Update $\hat{\mathbf{y}} = \sum_{t=1}^{T} g(\mathbf{X},\mathcal{T}_t,\mathcal{M}_t)$.
**end for**
**Output**: samples of the posterior distribution of $\mathcal{T}$.
```

3.4 Model trees BART

In MOTR-BART, we consider that the response variable is a sum-of-trees in the form of

$$\mathbf{y} = \sum_{t=1}^{T} g(\mathbf{X}; \mathcal{T}_t, \mathcal{B}_t) + \epsilon,$$

where $\mathcal{B}_t$ is the set of parameters of all linear predictors of the tree t. In terms of partial residuals, MOTR-BART can be represented as

$$r_i | \mathbf{x}_i, \boldsymbol{\beta}_{t\ell}, \sigma^2 \sim \mathrm{N}(\mathbf{x}_i \boldsymbol{\beta}_{t\ell}, \sigma^2),$$

where $r_i = y_i - \sum_{j \neq t}^{T} g(\mathbf{x}_i; \mathcal{T}_j, \mathcal{B}_j)$, $\boldsymbol{\beta}_{t\ell}$ is the parameter vector associated with the terminal node ℓ of the tree t. In this sense, all observations $i \in \mathcal{P}_{t\ell}$ will have predicted values based on $\boldsymbol{\beta}_{t\ell}$ and the values of their covariates $\mathbf{X}_{t\ell}$; i.e., each observation $i \in \mathcal{P}_{t\ell}$ may have different predicted value. The priors for $\boldsymbol{\beta}_{t\ell}$ and σ^2 are:

$$\begin{aligned} \boldsymbol{\beta}_{t\ell} | \mathcal{T}_t, \mathbf{V} &\sim \mathrm{MVN}(\mathbf{0}, \sigma^2 \mathbf{V}), \\ \sigma^2 | \mathcal{T}_t &\sim \mathrm{IG}(\nu/2, \nu\lambda/2), \end{aligned} \tag{3.5}$$

where MVN$(\cdot)$ denotes the multivariate normal distribution, $\mathbf{V} = \tau_b^{-1} \times \mathbf{I}_q$ and $q = p_{t\ell} + 1$, with $p_{t\ell}$ representing the number of covariates in the linear predictor of the terminal node ℓ of the tree t. The additional dimension in $\mathbf{V}$ is due to a column filled with ones in the design matrix $\mathbf{X}_{t\ell}$. Here, the role of the parameter τ_b is to balance the importance of each tree on the final prediction by keeping the components of $\boldsymbol{\beta}_{t\ell}$ close to zero, thus avoiding that one tree contributes more than other. We scale the predictors in $\mathbf{X}$ as our prior on $\boldsymbol{\beta}_{t\ell}$ assumes that all entries have the same variance. In our simulations and real data applications, we have found that $\tau_b = T$ worked well.

Another possibility is to penalise the intercept and the slopes differently. In this sense, the specification of intercept- and slope-specific variances may be done by setting $\mathbf{V}$ as a $q \times q$ diagonal matrix with $\mathbf{V}_{1,1} = (\tau_{\beta_0} T)^{-1}$ and $\mathbf{V}_{j+1,j+1} = (\tau_\beta T)^{-1}$. In addition, we may assume conjugate priors, such as $\tau_{\beta_0} \sim \mathrm{G}(a_0, b_0)$ and $\tau_\beta \sim \mathrm{G}(a_1, b_1)$, to be able to estimate both variances via Gibbs-sampling steps. In this case, we would end up with the following full conditionals:

$$\tau_{\beta_0}|- \sim \mathrm{G}\left(\frac{\sum_{t=1}^{T} b_t}{2} + a_0, \frac{\boldsymbol{\beta}_0^\top \boldsymbol{\beta}_0}{2\sigma^2} + b_0\right),$$

$$\tau_{\beta}|- \sim \mathrm{G}\left(\frac{\sum_{t=1}^{T}\sum_{\ell=1}^{b_t} p_{t\ell}}{2} + a_1, \frac{\boldsymbol{\beta}_\star^\top \boldsymbol{\beta}_\star}{2\sigma^2} + b_1\right),$$

where $\boldsymbol{\beta}_0$ is a column vector with the intercepts from all terminal nodes of all trees and $\boldsymbol{\beta}_\star$ contains the slopes from all linear predictors of all trees. In our software, we have implemented an option, through the argument `vars_inter_slope = TRUE/FALSE`, that allows the user to either estimate τ_{β_0} and τ_β or use $\tau_b = T$, with `vars_inter_slope = TRUE` as the default. In Section 3.5, we show the results of MOTR-BART using both approaches.

Hence, the full conditionals are

$$p(\boldsymbol{\beta}_{t\ell}|\mathbf{X}_{t\ell}, \mathbf{R}_t, \sigma^2, \mathcal{T}_t) \propto p(\mathbf{R}_t|\mathbf{X}_{t\ell}, \boldsymbol{\beta}_{t\ell}, \sigma^2, \mathcal{T}_t)p(\boldsymbol{\beta}_{t\ell}),$$

which is identifiable as a multivariate normal distribution:

$$\mathrm{MVN}\left(\boldsymbol{\mu}_{t\ell}, \sigma^2\boldsymbol{\Lambda}_{t\ell}\right),$$

where $\boldsymbol{\mu}_{t\ell} = \boldsymbol{\Lambda}_{t\ell}(\mathbf{X}_{t\ell}^\top \mathbf{r}_{t\ell})$, $\boldsymbol{\Lambda}_{t\ell} = (\mathbf{X}_{t\ell}^\top\mathbf{X}_{t\ell} + \mathbf{V}^{-1})^{-1}$, and $\mathbf{X}_{t\ell}$ is an $n_{t\ell} \times q$ matrix with all elements of the design matrix such that $i \in \mathcal{P}_{t\ell}$. The full conditional of σ^2 is similar to the expression in (3.4), but with $\hat{y}_i = \sum_{t=1}^{T} g(\mathbf{x}_i; \mathcal{T}_t, \mathcal{B}_t)$. Finally, the full conditional for $\mathcal{T}_t$ is given by

$$\begin{aligned} p(\mathcal{T}_t|\mathbf{X}, \mathbf{R}_t, \sigma^2) &\propto p(\mathcal{T}_t)\int p(\mathbf{R}_t|\mathbf{X}, \mathcal{B}_t, \sigma^2, \mathcal{T}_t)p(\mathcal{B}_t)d\mathcal{B}_t, \\ &\propto p(\mathcal{T}_t)p(\mathbf{R}_t|\mathbf{X}, \sigma^2, \mathcal{T}_t), \end{aligned}$$

where

$$\begin{aligned} p(\mathbf{R}_t|\mathbf{X}, \sigma^2, \mathcal{T}_t) = (\sigma^2)^{-n/2} \prod_{\ell=1}^{b_t} \Big[|\mathbf{V}|^{-1/2}|\boldsymbol{\Lambda}_{t\ell}|^{1/2} \times \\ \times \exp\left(-\frac{1}{2\sigma^2}\left[-\boldsymbol{\mu}_{t\ell}^\top\boldsymbol{\Lambda}_{t\ell}^{-1}\boldsymbol{\mu}_{t\ell} + \mathbf{r}_{t\ell}^\top\mathbf{r}_{t\ell}\right]\right)\Big]. \end{aligned}$$

The main difference between BART and MOTR-BART can be seen in Figure 3.2. Now, rather than having a constant as the predicted value for each terminal node,

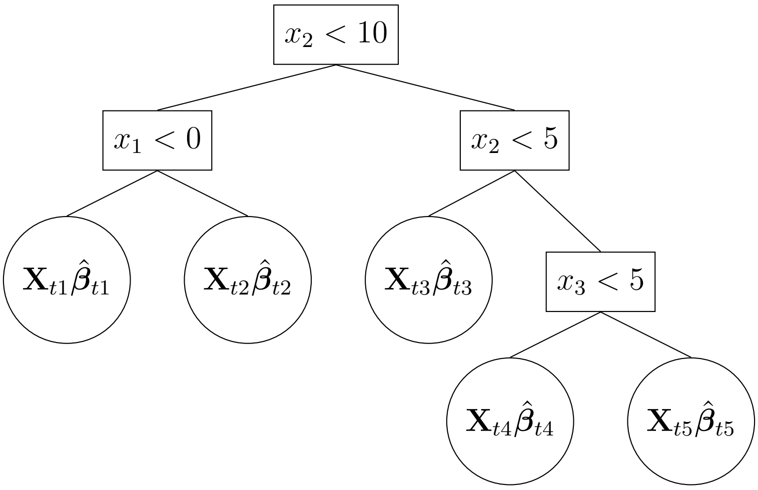

$x_2 < 10$
$x_1 < 0$
$x_2 < 5$
$\mathbf{X}_{t1}\hat{\boldsymbol{\beta}}_{t1}$
$\mathbf{X}_{t2}\hat{\boldsymbol{\beta}}_{t2}$
$\mathbf{X}_{t3}\hat{\boldsymbol{\beta}}_{t3}$
$x_3 < 5$
$\mathbf{X}_{t4}\hat{\boldsymbol{\beta}}_{t4}$
$\mathbf{X}_{t5}\hat{\boldsymbol{\beta}}_{t5}$

Figure 3.2: An example of a tree generated based on MOTR-BART. The quantities x_1, x_2, and x_3 represent covariates; $\mathbf{X}_{t\ell}$ is a subset of the design matrix $\mathbf{X}$ such that $i \in \mathcal{P}_{t\ell}$ and $\hat{\boldsymbol{\beta}}_{t\ell} = (\hat{\beta}_{0t\ell}, \hat{\beta}_{1t\ell}, ..., \hat{\beta}_{p_{t\ell}t\ell})^\top$ is the parameter vector associated with the node ℓ of the tree t.

the prediction will be obtained from a linear predictor at node level. The purpose of introducing a linear predictor is to try to capture local linearity, reduce the number of trees, and then possibly improve the prediction at node level.

The key point in MOTR-BART is which covariates should be considered in the linear predictor of each terminal node. Our idea to circumvent this issue is to consider in the linear predictor only covariates that have been used as a split in the corresponding tree. For instance, in Figure 3.2 three covariates are used as a split (x_1, x_2, and x_3). The plan is to include these three covariates in each of the five linear predictors. The intuition in doing so is that if a covariate has been utilised as a split, it means that it improves the prediction either because it has a linear or a non-linear relation with the response variable. If this relation is linear, this will be captured by the linear predictor. However, if the relation is non-linear, the coefficient associated with this covariate will be close to zero and the covariate will not have an impact on the prediction.

We have also explored using only the ancestors of the terminal nodes in the linear predictor as well as replacing the uniform distribution over the splitting probabilities, where the covariates are selected with equal probability, by the sparsity-inducing Dirichlet prior proposed by Linero (2018). To illustrate the first approach, we recall Figure 3.2, where there are five terminal nodes and three covariates are used in the splitting rules. For the two left-most terminal nodes, only the covariates x_1 and x_2 would be considered in both linear predictors. For the terminal node 3, only x_2. For the right-most terminal nodes, x_3 and x_2 would be used. In relation to Linero's approach, rather than selecting the covariates with probability $1/p$, a Dirichlet prior is placed on the vector of splitting probabilities so that the covariates that are frequently used to create the internal nodes are more likely to be chosen.

At first glance, one might think that it would be advantageous to use conventional shrinkage/regularisation techniques, such as ridge regression, lasso (Tibshirani, 1996) or horseshoe (Carvalho et al., 2010). Under the Bayesian perspective, these methods assume different priors on the regression coefficient vector and then estimate its components. In the ridge and horseshoe regressions, a Gaussian with mean zero is assumed as the prior on the parameter vector. For lasso regression, a Laplace distribution is considered. For MOTR-BART, we assume a normal distribution with mean zero on $\boldsymbol{\beta}_{t\ell}$, which is equivalent to performing a local ridge regression at the node level, but as the trees might change their dimensions depending on the moves growing and pruning, it is not possible to obtain the posterior distribution associated with each component of $\boldsymbol{\beta}_{t\ell}$ and then perform the variable selection.

3.4.1 MOTR-BART for classification

The version of MOTR-BART that was presented in Section 3.4 assumes that the response variable is continuous. In this Section, we provide the extension to the case when it is binary following the idea of Chipman et al. (2010), which used the strategy of data augmentation (Albert and Chib, 1993). Firstly, we consider that $y_i \in \{0, 1\}$ and we introduce a latent variable

$$z_i \sim \mathrm{N}\left(\sum_{t=1}^{T} g(\mathbf{x}_i, \mathcal{T}_t, \mathcal{B}_t), 1\right), \text{ with } i = 1, ..., n$$

such that $y_i = 1$ if $z_i > 0$ and $y_i = 0$ if $z_i \leq 0$. With this formulation, we have that $p(y_i = 1|\mathbf{x}_i) = \Phi(\sum_{t=1}^{T} g(\mathbf{x}_i, \mathcal{T}_t, \mathcal{B}_t))$, where $\Phi(\cdot)$ is the cumulative distribution function (cdf) of the standard normal, which works as the link function that limits the output to the interval $(0, 1)$. Here, there is no need to estimate the variance component as it is equal to 1. The priors on $\mathcal{T}_t$ and $\mathcal{B}_t$ are the same as in (3.1) and (3.5), respectively. Finally, as the latent variable z_i is introduced, it is necessary to compute its full conditional, which is given by

$$z_i|[y_i = 0] \sim \mathrm{N}_{(-\infty,0)}\left(\sum_{t=1}^{T} g(\mathbf{x}_i, \mathcal{T}_t, \mathcal{B}_t), 1\right),$$
$$z_i|[y_i = 1] \sim \mathrm{N}_{(0,\infty)}\left(\sum_{t=1}^{T} g(\mathbf{x}_i, \mathcal{T}_t, \mathcal{B}_t), 1\right),$$

where $\mathrm{N}_{(a,b)}(\cdot)$ denotes a truncated normal distribution constrained to the interval (a, b). Going back to Algorithm 2, some steps need to be modified or included:

1. The update of σ^2 is no longer needed, because it is set to one following the definition of the data augmentation scheme.
2. The predicted values now consider the cdf of the standard normal as a probit model in the form of $\hat{\mathbf{y}}^{(k)} = \Phi\left(\sum_{t=1}^{T} g(\mathbf{X}, \mathcal{T}_t, \mathcal{B}_t)\right)$.
3. A Gibbs sampling step needs to be added to update the latent variables at each MCMC iteration. The update is done by drawing samples from $p(z_i|y_i)$.
4. Rather than calculating the partial residuals taking into account the response variable, we have that $\mathbf{R}_t = \mathbf{z} - \sum_{j\neq t}^{T} g(\mathbf{X}, \mathcal{T}_j, \mathcal{B}_j)$, where $\mathbf{z}$ is the vector with all latent variables.

3.5 Results

In this Section, we compare MOTR-BART to BART, RF, GB, lasso regression, SBART, and LLF via simulation scenarios and real data applications using the root mean squared errors (RMSE) as the accuracy measure. All results were generated by using R (R Core Team, 2020) version 3.6.3 and the packages `dbarts` (Dorie, 2020), `ranger` (Wright and Ziegler, 2017), `gbm` (Greenwell et al., 2019), `glmnet` (Friedman et al., 2010), `SoftBart` (Linero, 2017a), and `grf` (Tibshirani et al., 2020). We use the default behaviour of these packages, except where otherwise specified below. We also tried running the linear random forests (LRF; Künzel et al., 2022) algorithm. However, error messages were reported when using the `forestry` R package. Thus, LRF is not considered further in our comparisons.

Throughout this Section, we present results for two versions of our method. The first one is MOTR-BART (10 trees), which uses the sparsity-inducing Dirichlet prior and estimates τ_{β_0} and τ_{β}, while the second is MOTR-BART (10 trees, fixed var), which assumes the uniform distribution on the splitting probabilities and sets $\tau_b = T$. As a default version, we recommend the MOTR-BART (10 trees).

3.5.1 Simulation

To compare the algorithms, we simulate data from the equation proposed by Friedman (1991). This dataset is widely used in testing tree-based models and has been used repeatedly to evaluate the performance of BART and extensions (Friedman, 1991; Chipman et al., 2010; Linero, 2018). We generate the response variable considering five covariates via:

$$y_i = 10\text{sin}(\pi x_{i1} x_{i2}) + 20(x_{i3} - 0.5)^2 + 10x_{i4} + 5x_{i5} + \epsilon_i,$$

where the covariates $x_{ip} \sim \text{Uniform}(0, 1)$, with $p = 1, \ldots, 5$, and $\epsilon_i \sim \text{N}(0, 1)$. For this simulation experiment, we created 9 datasets with different numbers of observations (200, 500, and 1000) and covariates (5, 10, and 50). For those scenarios with 10 and 50 covariates, the additional x values do not have any impact on the response variable.

Each simulated dataset was split into 10 different training (80%) and test (20%) partitions. For MOTR-BART, 10 trees were considered, 1000 iterations as burn-in, 5000 as post-burn-in, with `alpha` $= \alpha = 0.95$ and `beta` $= \beta = 2$. To choose the number of trees (10) for MOTR-BART, we initially tested a range of possible values, such as 3, 10, and 50, and then used cross-validation to select the setting which presented the lowest RMSE. The set up for `dbarts` was similar to MOTR-BART, except for the number of trees (10 and 200, with the latter being the default). For the packages `ranger` and `gbm`, the default options were kept, except for the number of trees (200) and the parameter `interaction.depth = 3`. For the `glmnet`, we followed the manual and used a 10-fold cross-validation with `type.measure = "mse"` to obtain the estimate of the regularisation parameter `lambda.min`, which is the value that minimises the cross-validated error under the loss function chosen in `type.measure`. As in Chipman et al. (2010), we evaluate the convergence of MOTR-BART and BART by eye from the plot of σ^2 after the burn-in period.

In Figure 3.3, we present the comparison of the algorithms MOTR-BART, BART, RF, GB, Lasso, SBART, and LLF in terms of RMSE on test data. Note that we have BART (10 trees) and BART (200 trees; default). The first version considers 10 trees and was run to see how BART would perform with the equivalent number of trees of MOTR-BART. We can see that for different combinations of number of observations (n) and covariates (p), SBART and MOTR-BART (10 trees) consistently presented the best results for all scenarios. When compared to both versions of the original BART, both versions of MOTR-BART present lower median values of RMSE and slightly greater variability. However, the variability reduces as n and p increase. Further, we notice that MOTR-BART (10 trees) benefits from penalising the intercepts and slopes differently. In addition, it is possible to observe that the number of noisy covariates impacts on the performance of RF and LLF. For all values of n, their RMSEs increase with the number of covariates.

To further analyse the improvements given by MOTR-BART over standard BART, in Appendix 3.A we present Table 3.A.4, which shows the mean of the total number of terminal nodes utilised for BART to calculate the final prediction taking into account all the 5000 iterations. The idea of Table 3.A.4 is to show that MOTR-BART has similar or better performance whilst using fewer parameters than standard BART. As the default version of BART defaults to 200 trees, which is far more trees than MOTR-BART uses, we created Table 3.A.4 to highlight that although MOTR-BART estimates fewer parameters, it still remains competitive to the default BART. For MOTR-BART, we consider the mean of the number of parameters estimated in the linear predictors. As BART and SBART predict a constant for each terminal node, the number of 'parameters' estimated is equal to the number of terminal nodes. On the other hand, MOTR-BART estimates an intercept, which is equivalent to the constant that BART predicts, plus the parameters associated with those covariates that have been used as a split in the

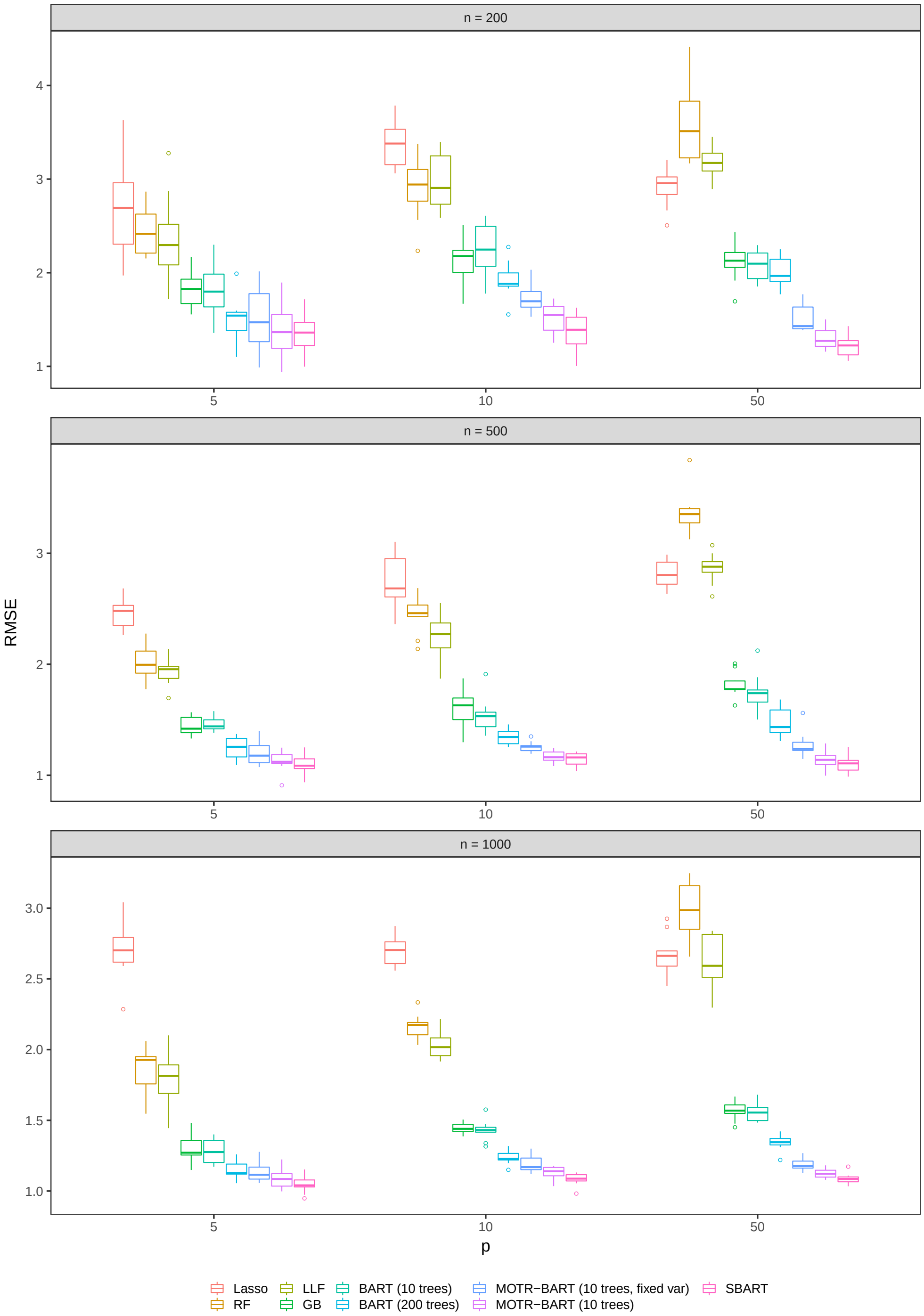


Figure 3.3: Comparison of RMSE for the Friedman datasets on test data for different combinations of n (200, 500, and 1000) and p (5, 10, and 50).

corresponding tree. For instance, if a tree has 5 terminal nodes and 2 numeric variables are used in the splitting rules, MOTR-BART will estimate 15 parameters. For BART, we set the argument `keepTrees = TRUE` and then we extracted from the sampler object `fit` the content of `getTrees()`. For both MOTR-BART and BART, we firstly summed the number of parameters for all trees along the MCMC iterations and then averaged it over the 10 sets.

In Table 3.A.4, we can observe, for example, for the Friedman dataset with $n = 1000$ and $p = 50$ that BART (10 trees) utilised 212,421 parameters on average to calculate the final prediction, while MOTR-BART (10 trees), BART (200 trees), and SBART used 391,193, 2,371,140, and 255,155, respectively. For all simulated datasets, MOTR-BART presented lower RMSE than BART (200 trees), even though it estimates far fewer parameters. From Table 3.A.4, it is possible to obtain the mean number of terminal nodes per tree by dividing the column 'Mean' by the number of MCMC iterations (5000) times the number of trees (10 or 200). In this case, we note that both versions of BART produce small trees with the mean number of terminal nodes per tree varying between 2 and 5. Due to the greater number of trees, BART (200 trees) has the lowest mean, regardless of the number of observations and covariates. In contrast, MOTR-BART has the mean number of parameters per tree varying from 5 to 8. Comparatively speaking, this is somewhat expected once MOTR-BART estimates a linear predictor. In this way, the trees from MOTR-BART tend to be shallower than those from BART (10 trees), but with more parameters estimated overall. It is important to highlight that the numbers from BART and MOTR-BART cannot be compared to those from RF, as the former work with the residuals and the latter with the response variable itself. The numbers for GB are not shown as the quantity of terminal nodes in each tree is fixed due to `interaction.depth = 3`.

In our simulations, MOTR-BART utilised just 10 trees and its results were better than RF, GB, BART (10 and 200 trees) and LLF. In practice, different number of trees may be compared via cross-validation and hence a choice can be made such that the cross-validated error is minimised.

3.5.2 Applications

In this Section, we compare the predictive performance of MOTR-BART to RF, GB, BART, SBART, and LLF in terms of RMSE on four real datasets. The first one (Ankara) has 1,609 observations and contains weather information for the city of Ankara from 1994 to 1998. The goal is to predict the mean temperature based on 9 covariates. The second is the Boston Housing dataset, where the response variable is the median value of properties in suburbs of Boston according to the 1970 U.S. census. This dataset has 506 rows and 18 explanatory variables. The third dataset (Ozone) has 330 observations and 8 covariates and is about ozone concentration in Los Angeles in 1976. The aim is to predict the amount of ozone in parts per million (ppm) based on wind speed, air temperature, pressure gradient, humidity and other covariates. The fourth dataset (Compactiv) refers to a multi-user computer that had the time of its activity measured under different tasks. The goal is to predict the portion of time that the computer runs in user mode for 8,192 observations based on 21 covariates. These datasets are a subset of 9 sets considered by Kapelner and Bleich (2016). As with the Friedman data, we consider two versions of BART (10 and 200 trees) and MOTR-BART (10 trees and 10 trees, fixed var), and we split the data into 10 different train (80%) and test (20%) sets. Furthermore, all results are based on the test data.

Figure 3.4 shows the results of RMSE on test sets. It is possible to note that MOTR-BART (10 trees) presents the lowest or second lowest median RMSE on all datasets, except for Ozone. For Ankara, RF and GB have quite similar results and Lasso presents the highest RMSE. For Boston, Lasso regression shows the highest RMSE, while MOTR-BART (10 trees) and SBART do not differ much in terms of median and quartiles. For Ozone, it can be seen that MOTR-BART (10 trees) presents the highest RMSE and that LLF, RF, and MOTR-BART (10 trees, fixed var) have the lowest median values. For Compactiv, RF and GB show similar results, while MOTR-BART (10 trees, fixed var) presents the lowest RMSE. To facilitate the visualisation, the results for Lasso are not shown for the dataset Compactiv, as it has RMSEs greater than 9. In Appendix 3.B, however, Table 3.B.1 reports the median and the first and third quartiles of the RMSE for all algorithms and datasets.

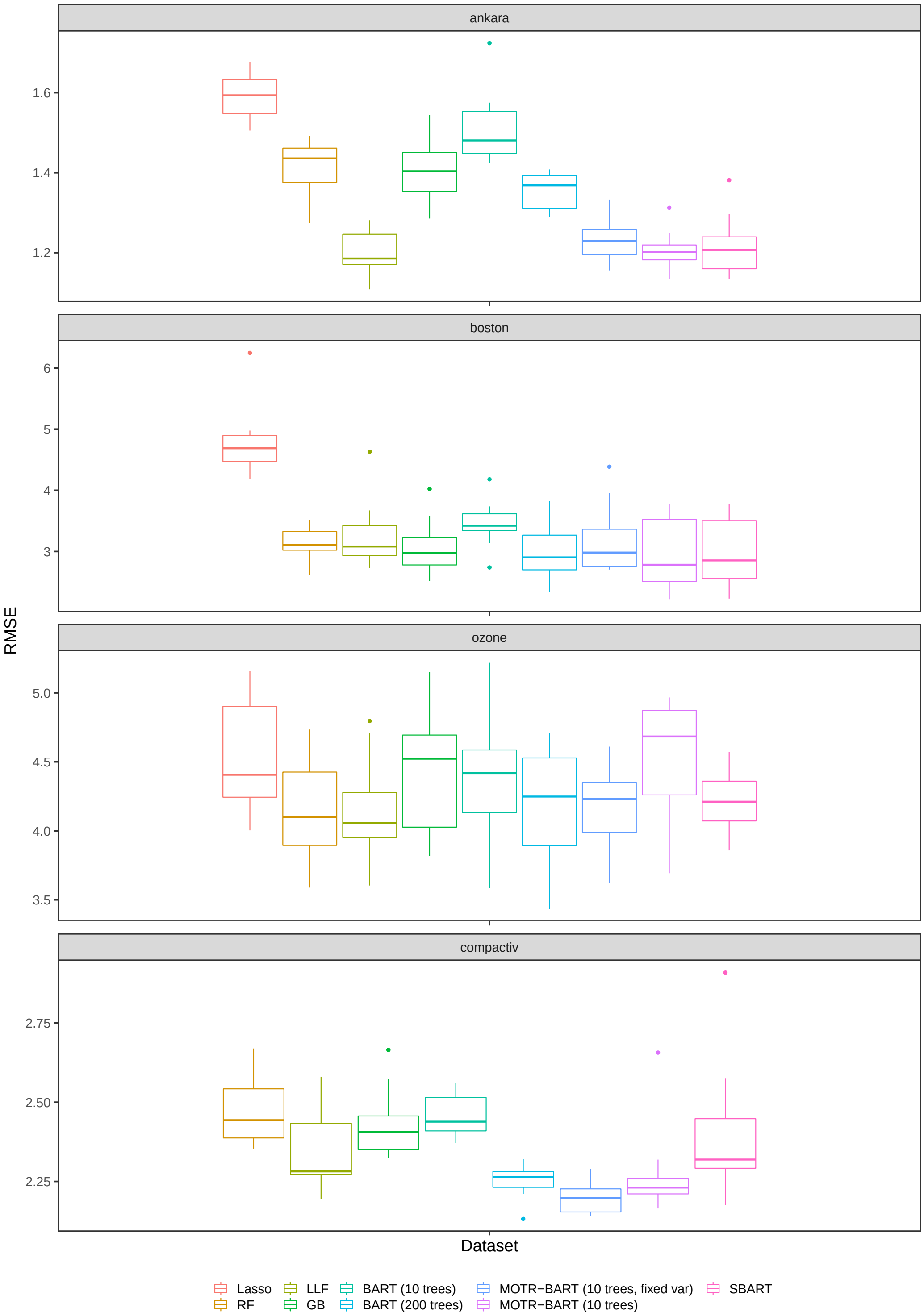


Figure 3.4: Comparison of RMSE for the Ankara, Boston, Ozone, and Compactiv datasets on test data.

In Table 3.B.2 (see Appendix 3.B), we show the mean of the total number of parameters/terminal nodes created for BART to generate the final prediction for each dataset. For MOTR-BART, the numbers correspond to the mean of the total of parameters estimated. For instance, for the dataset Ankara, 304,696 terminal nodes were used on average by BART (10 trees), while BART (200 trees) estimated 2,250,599 and MOTR-BART 546,959. As can be seen, MOTR-BART estimates more parameters than BART (10 trees) for all datasets, as we expect. However, when compared to BART (200 trees), MOTR-BART in general uses significantly fewer parameters to obtain similar or better performance, except for Compactiv.

To finish, we point out one drawback of the MOTR-BART model when compared to BART. Due to the linear models in the terminal nodes of the trees, MOTR-BART extrapolates linearly beyond the most extreme values of the training data. This is in contrast with BART, whose piecewise constants extrapolate with constant terms, which in turn do not rely on any covariates. This disadvantage is especially pertinent when the distributions of the covariates in the training and test datasets differ. Thus, if the distribution of the covariates of the training and test datasets are moderately or significantly different, this could negatively impact the MOTR-BART predictive performance, especially for squared error measures like RMSE.

3.6 Discussion

In this paper, we have proposed an extension to BART, called MOTR-BART, that can be seen as a combination of BART and model trees. In MOTR-BART, rather than having a constant as predicted value for each terminal node, a linear predictor is estimated considering only those covariates that have been used as a split in the corresponding tree. Furthermore, MOTR-BART is able to capture linear associations between the response and covariates at node level and requires fewer trees to achieve equivalent or better performance when compared to other methods.

Via simulation studies and real data applications, we showed that MOTR-BART is highly competitive when compared to BART, random forests, gradient boosting, lasso, SBART, and local linear forests. In simulation scenarios, MOTR-BART outperformed the other tree-based methods, except SBART. In the real data applications, four datasets were considered and MOTR-BART provided great predictive performance.

Due to the structure of MOTR-BART, to evaluate variable importance or even to select the covariates that should be included in the linear predictors is not straightforward. Recall that model trees was introduced in the context of one tree, where statistical methods of variable selection, such as forward, backward or stepwise can be performed at node level. Compared to other tree-based methods that consider only one tree, model trees produces much smaller trees (Landwehr et al., 2005), which helps to alleviate the computational time required by the variable selection procedures. In theory, one might think that it would be possible to use such a procedure for MOTR-BART, but in practice they would be a burden as they would have to be performed for each terminal node (and for all trees).

In the Bayesian context, Chipman et al. (2010) propose to use the inclusion probability as a measure to evaluate variable importance in BART. Basically, this metric is the proportion of times that a covariate is used as a split out of all splitting rules over all trees and MCMC iterations. However, this measure gives us an overall idea about the covariates that are important for the splitting rules and does not say anything about which covariates that should be included in the local linear predictors.

In this sense, the variable selection/importance remains as a challenge that may be investigated in future work, since conventional procedures are not suitable. One might try using spike-and-slab priors (George and McCulloch, 1997; Ishwaran and Rao, 2005) on the vector $\boldsymbol{\beta}_{t\ell}$ in order to further optimise the proposed variable selection presented in Section 3.4. The rationale would be to zero out coefficients that are irrelevant at the node level and only keep those that are significant. Another extension could be replacing the linear functions by splines to provide even further flexibility and capture local non-linear behaviour, which is a subject of ongoing work. Finally, model trees can be incorporated to other BART extensions, such as BART for log-linear models (Murray, 2021), SBART (Linero and Yang, 2018), and BART for log-normal and gamma hurdle models (Linero et al., 2020).

Appendix

3.A Simulation results

In this Section, we present results related to the simulation scenarios shown in Subsection 3.5.1. In total, 9 datasets were created based on Friedman's equation considering some combinations of sample size (n) and number of covariates (p). In Tables 3.A.1, 3.A.2 and 3.A.3, the medians and quartiles of the RMSE are shown for the algorithms MOTR-BART, BART, GB, RF, Lasso, SBART, and LLF. The values in these tables were graphically shown in Figure 3.3. In addition, Tables 3.A.4 and 3.A.5 present the mean number of parameters utilised by BART, MOTR-BART, and SBART to calculate the final prediction.

Table 3.A.1: The median of the RMSE on test data of the Friedman datasets when $n = 200$. The values in parentheses are the first and third quartiles, respectively. The two lowest RMSEs are highlighted in boldface font. The acronym ‘fv’ stands for ‘fixed var’.

Algorithm	p	RMSE
	$n = 200$	
MOTR-BART	5	**1.36 (1.19;1.55)**
MOTR-BART (fv)	5	1.47 (1.26;1.78)
BART (10 trees)	5	1.80 (1.63;1.99)
BART (200 trees)	5	1.54 (1.38;1.58)
GB	5	1.83 (1.67;1.93)
RF	5	2.41 (2.21;2.63)
Lasso	5	2.69 (2.30;2.96)
SBART	5	**1.36 (1.22;1.47)**
LLF	5	2.30 (2.08;2.52)
MOTR-BART	10	**1.55 (1.39;1.64)**
MOTR-BART (fv)	10	1.70 (1.63;1.80)
BART (10 trees)	10	2.25 (2.07;2.49)
BART (200 trees)	10	1.88 (1.86;2.00)
GB	10	2.18 (2.00;2.24)
RF	10	2.94 (2.76;3.10)
Lasso	10	3.38 (3.15;3.53)
SBART	10	**1.39 (1.24;1.52)**
LLF	10	2.91 (2.73;3.25)
MOTR-BART	50	**1.27 (1.21;1.38)**
MOTR-BART (fv)	50	1.43 (1.40;1.63)
BART (10 trees)	50	2.10 (1.94;2.21)
BART (200 trees)	50	1.97 (1.90;2.14)
GB	50	2.13 (2.06;2.22)
RF	50	3.51 (3.23;3.83)
Lasso	50	2.96 (2.84;3.02)
SBART	50	**1.22 (1.12;1.27)**
LLF	50	3.17 (3.09;3.28)

Table 3.A.2: The median of the RMSE on test data of the Friedman datasets when $n = 500$. The values in parentheses are the first and third quartiles, respectively. The two lowest RMSEs are highlighted in boldface font. The acronym 'fv' stands for 'fixed var'.

Algorithm	p	RMSE
	$n = 500$	
MOTR-BART	5	**1.12 (1.11;1.19)**
MOTR-BART (fv)	5	1.18 (1.11;1.27)
BART (10 trees)	5	1.44 (1.42;1.50)
BART (200 trees)	5	1.26 (1.17;1.33)
GB	5	1.42 (1.38;1.52)
RF	5	2.00 (1.92;2.12)
Lasso	5	2.48 (2.35;2.53)
SBART	5	**1.09 (1.06;1.15)**
LLF	5	1.96 (1.87;1.98)
MOTR-BART	10	**1.16 (1.14;1.21)**
MOTR-BART (fv)	10	1.26 (1.22;1.27)
BART (10 trees)	10	1.53 (1.44;1.57)
BART (200 trees)	10	1.35 (1.28;1.39)
GB	10	1.63 (1.50;1.70)
RF	10	2.46 (2.43;2.53)
Lasso	10	2.68 (2.61;2.95)
SBART	10	**1.16 (1.10;1.19)**
LLF	10	2.27 (2.15;2.37)
MOTR-BART	50	**1.14 (1.10;1.18)**
MOTR-BART (fv)	50	1.24 (1.22;1.30)
BART (10 trees)	50	1.74 (1.66;1.77)
BART (200 trees)	50	1.43 (1.38;1.59)
GB	50	1.78 (1.77;1.85)
RF	50	3.35 (3.27;3.40)
Lasso	50	2.80 (2.72;2.92)
SBART	50	**1.11 (1.05;1.13)**
LLF	50	2.88 (2.83;2.92)

Table 3.A.3: The median of the RMSE on test data of the Friedman datasets when $n = 1000$. The values in parentheses are the first and third quartiles, respectively. The two lowest RMSEs are highlighted in boldface font. The acronym 'fv' stands for 'fixed var'.

Algorithm	p	RMSE
	$n = 1000$	
MOTR-BART	5	**1.09 (1.03;1.12)**
MOTR-BART (fv)	5	1.11 (1.08;1.17)
BART (10 trees)	5	1.28 (1.20;1.36)
BART (200 trees)	5	1.13 (1.12;1.19)
GB	5	1.27 (1.25;1.36)
RF	5	1.93 (1.76;1.95)
Lasso	5	2.70 (2.62;2.79)
SBART	5	**1.04 (1.03;1.08)**
LLF	5	1.81 (1.69;1.89)
MOTR-BART	10	**1.14 (1.11;1.17)**
MOTR-BART (fv)	10	1.17 (1.15;1.23)
BART (10 trees)	10	1.43 (1.42;1.45)
BART (200 trees)	10	1.23 (1.22;1.27)
GB	10	1.44 (1.42;1.47)
RF	10	2.17 (2.10;2.19)
Lasso	10	2.70 (2.61;2.76)
SBART	10	**1.09 (1.07;1.12)**
LLF	10	2.02 (1.96;2.08)
MOTR-BART	50	**1.12 (1.10;1.15)**
MOTR-BART (fv)	50	1.18 (1.16;1.21)
BART (10 trees)	50	1.55 (1.50;1.59)
BART (200 trees)	50	1.35 (1.33;1.37)
GB	50	1.57 (1.55;1.61)
RF	50	2.99 (2.85;3.16)
Lasso	50	2.66 (2.59;2.70)
SBART	10	**1.09 (1.07;1.10)**
LLF	10	2.59 (2.51;2.81)

Table 3.A.4: Friedman data: Mean and standard deviation of the total number of terminal nodes created for BART and SBART to generate the final prediction over 5,000 iterations for $n = 200$. For MOTR-BARTs, the values correspond to the mean of the total number of parameters estimated in the linear predictors. The acronym ‘fv’ stands for ‘fixed var’.

Algorithm	p	Mean	Std
	$n = 200$		
MOTR-BART	5	302,447	18,210
MOTR-BART (fv)	5	263,079	11,990
BART (10 trees)	5	163,468	7,393
BART (200 trees)	5	2,468,707	6,734
SBART	5	250,615	6,599
MOTR-BART	10	326,678	20,309
MOTR-BART (fv)	10	258,380	13,530
BART (10 trees)	10	145,458	7,380
BART (200 trees)	10	2,470,333	3,670
SBART	10	256,391	5,577
MOTR-BART	50	327,751	28,627
MOTR-BART (fv)	50	251,469	12,376
BART (10 trees)	50	134,809	4,447
BART (200 trees)	50	2,428,259	5,368
SBART	50	256,184	6,754

Table 3.A.5: Friedman data: Mean and standard deviation of the total number of terminal nodes created for BART and SBART to generate the final prediction over 5,000 iterations for $n = 500$ and $n = 1000$. For MOTR-BARTs, the values correspond to the mean of the total number of parameters estimated in the linear predictors. The acronym 'fv' stands for 'fixed var'.

Algorithm	p	Mean	Std
	$n = 500$		
MOTR-BART	5	364,786	21,978
MOTR-BART (fv)	5	364,258	17,478
BART (10 trees)	5	203,625	7,950
BART (200 trees)	5	2,470,900	8,739
SBART	5	257,769	6,727
MOTR-BART	10	382,528	24,768
MOTR-BART (fv)	10	354,755	25,238
BART (10 trees)	10	206,394	8,694
BART (200 trees)	10	2,448,212	9,171
SBART	10	256,465	2,829
MOTR-BART	50	384,828	18,099
MOTR-BART (fv)	50	330,434	33,566
BART (10 trees)	50	178,779	8,700
BART (200 trees)	50	2,407,661	14,314
SBART	50	254,164	9,334
	$n = 1000$		
MOTR-BART	5	389,479	23,247
MOTR-BART (fv)	5	396,280	29,040
BART (10 trees)	5	271,878	8,977
BART (200 trees)	5	2,425,863	8,276
SBART	5	257,656	9,881
MOTR-BART	10	410,517	30,346
MOTR-BART (fv)	10	390,274	22,442
BART (10 trees)	10	256,511	7,812
BART (200 trees)	10	2,415,372	8,575
SBART	10	255,604	6,549
MOTR-BART	50	391,193	16,127
MOTR-BART (fv)	50	380,365	40,069
BART (10 trees)	50	212,421	5,959
BART (200 trees)	50	2,371,140	14,287
SBART	50	255,155	4,400

3.B Real data results

This Appendix presents two tables with results associated with the datasets Ankara, Boston, Ozone, and Compactiv. In Table 3.B.1, it is reported the median and quartiles of the RMSE computed on 10 test sets. The values in this table are related to the Figure 3.4 from Subsection 3.5.2. Further, Table 3.B.2 shows the mean number of parameters utilised by BART, MOTR-BART, and SBART to calculate the final prediction for the aforementioned datasets.

Table 3.B.1: Real datasets: Median RMSE (and first and third quartiles) for the Ankara, Boston, Ozone, and Compactiv datasets on test data. The two lowest RMSEs are highlighted in boldface font. The acronym 'fv' stands for 'fixed var'.

Dataset	Algorithm	RMSE	rank
Ankara	MOTR-BART	**1.20 (1.18;1.22)**	2
	MOTR-BART (fv)	1.23 (1.20;1.26)	4
	BART (200 trees)	1.37 (1.31;1.39)	5
	BART (10 trees)	1.48 (1.45;1.55)	8
	GB	1.40 (1.35;1.45)	6
	RF	1.44 (1.38;1.46)	7
	Lasso	1.59 (1.55;1.63)	9
	SBART	1.21 (1.16;1.24)	3
	LLF	**1.19 (1.17;1.25)**	1
Boston	MOTR-BART	**2.78 (2.51;3.53)**	1
	MOTR-BART (fv)	2.98 (2.75;3.36)	5
	BART (200 trees)	2.90 (2.70;3.27)	3
	BART (10 trees)	3.42 (3.34;3.62)	8
	GB	2.97 (2.78;3.22)	4
	RF	3.10 (3.02;3.33)	6
	Lasso	4.69 (4.47;4.89)	9
	SBART	**2.85 (2.56;3.50)**	2
	LLF	3.08 (2.93;3.42)	7
Ozone	MOTR-BART	4.68 (4.26;4.87)	9
	MOTR-BART (fv)	4.23 (3.99;4.35)	3
	BART (200 trees)	4.25 (3.89;4.53)	4
	BART (10 trees)	4.42 (4.13;4.59)	6
	GB	4.52 (4.03;4.69)	8
	RF	**4.10 (3.89;4.43)**	2
	Lasso	4.41 (4.24;4.90)	7
	SBART	4.21 (4.07;4.36)	5
	LLF	**4.06 (3.95;4.28)**	1
Compactiv	MOTR-BART	**2.23 (2.21;2.26)**	2
	MOTR-BART (fv)	**2.20 (2.15;2.23)**	1
	BART (200 trees)	2.26 (2.23;2.28)	3
	BART (10 trees)	2.44 (2.41;2.51)	7
	GB	2.41 (2.35;2.46)	6
	RF	2.44 (2.39;2.54)	8
	Lasso	9.97 (9.51;10.09)	9
	SBART	2.32 (2.29;2.45)	5
	LLF	2.28 (2.27;2.43)	4

Table 3.B.2: Real datasets: Mean and standard deviation of the total number of terminal nodes created for BART and SBART to generate the final prediction over 5,000 iterations. For MOTR-BARTs, the values correspond to the mean of the total number of parameters estimated in the linear predictors. The acronym ‘fv’ stands for ‘fixed var’.

Dataset	Algorithm	Mean	Std
Ankara	MOTR-BART	546,959	36,977
	MOTR-BART (fv)	485,743	40,840
	BART (10 trees)	304,696	8,872
	BART (200 trees)	2,250,599	11,798
	SBART	312,927	11,539
Boston	MOTR-BART	748,468	58,945
	MOTR-BART (fv)	414,762	50,705
	BART (10 trees)	204,038	10,143
	BART (200 trees)	2,389,130	14,244
	SBART	318,171	17,078
Ozone	MOTR-BART	272,370	42,809
	MOTR-BART (fv)	182,189	8,093
	BART (10 trees)	137,239	2,642
	BART (200 trees)	2,343,350	5,128
	SBART	268,948	5,667
Compactiv	MOTR-BART	2,990,494	298,221
	MOTR-BART (fv)	1,529,666	102,940
	BART (10 trees)	539,621	15,759
	BART (200 trees)	2,649,167	29,989
	SBART	711,860	49,087

CHAPTER 4

Bayesian additive regression trees for genotype by environment interaction models

We propose a new class of models for the estimation of genotype by environment interactions in plant-based genetics. Our approach uses semi-parametric Bayesian additive regression trees to accurately capture marginal genotypic and environment effects along with their interaction in a cut Bayesian framework. We demonstrate that our approach is competitive or superior to similar models widely used in the literature via both simulation and a real world dataset. Furthermore, we introduce new types of visualisation to properly assess both the marginal and interactive predictions from the model. An R package that implements our approach is available at `https://github.com/ebprado/ambarti`*.*

4.1 Introduction

The interaction between genotypes and environments (GxE) is a key parameter in plant breeding (Allard and Bradshaw, 1992). Poor understanding of GxE can lead to sub-optimal selection of new genotypes and inbred lines. Understanding the GxE interactions is crucial for germplasm management, having strong genetic and economic impacts on seed production and crop yield (Sarti, 2013). Many models have been proposed for studying GxE in the context of multi-environmental experiments (METs). One special case is the additive main effects multiplicative interactions model (AMMI; Mandel, 1971).

The classical AMMI models combine features of analysis of variance (ANOVA) with a bilinear term to represent GxE interactions. Such interactions will be named here bilinear interactions. In addition, these models allow for estimation of main effects of genotypes and environments as well as the decomposition of the interaction through a bilinear term. Many extensions to the AMMI models have been proposed, including robust AMMI (Rodrigues et al., 2016) and weighted AMMI (Sarti, 2019).

Bayesian additive regression trees (BART; Chipman et al., 2010) is a non-parametric Bayesian model that generates a set of trees and uses random splitting rules to produce predictions for a univariate response. Given its flexibility to deal with non-linear structures and richer, non-multiplicative interactions terms, the use of BART and its extensions has increased with applications in many areas including proteomic studies (Hernández et al., 2018), hospital performance evaluation (Liu et al., 2015), credit scores (Zhang and Härdle, 2010) among many others.

In this paper, we extend the AMMI model to allow for richer GxE interactions, and similarly sidestep the model choice complexity term present in all AMMI-type approaches. We achieve this goal by including a new variant of BART, which we term 'double-grow' BART. The new proposed method, named AMBARTI, provides a cut Bayesian model (Bayarri et al., 2009; Plummer, 2015), where the 'double-grow' BART component is solely responsible for GxE interactions. In Sections 4.2.4 and 4.2.4.1, we contrast cut and fully Bayesian models using as example the proposed model and the work of Plummer (2015). .

We compare our newly proposed model with the traditional AMMI approaches and other competing interaction detection models, and we show that its performance is superior (judged on out-of-sample error) in both simulated and real-world example data. The real dataset we use is taken from the value of cultivation and usage (VCU) experiments of the Irish Department of Agriculture, which were conducted in the years between 2010 and 2019. Furthermore, the output of AMBARTI leads us to suggest several new forms of visualisation that are easier to interpret for non-specialists.

The remainder of this paper is structured as follows. In Section 4.2, we describe the framework used to collect evidence from METs, including the classic genetic equation to describe the relationship between phenotypes, genotypes, and environments. We also outline the formulation of the AMMI model in its classic form. In Section 4.2.3, we briefly describe the standard Bayesian additive regression trees model. In Section 4.2.4, we present the structure of our novel AMBARTI approach. Sections 4.3 and 4.4 contain the main results from the simulation experiments and real datasets, respectively. In Section 4.5, we conclude with a discussion and outline further opportunities.

4.2 Methods

4.2.1 GxE interactions and MET

The phenotypic expression of a genetic character can be theoretically decomposed in terms of genetic factors, environmental factors, and the interactions between them as

$$p = g + e + (ge),$$

where p is the phenotypic response, g is the genetic factor, e is the environmental factor, and (ge) is the interaction between genotypes and environments. The last term is necessary due to the different response of genotypes across different environments. For instance, if we produce a rank that orders the performance of a set of genotypes into a set of environments, we will usually notice that the order of the best to worst genotype is different in each environment. The presence of GxE interactions is known to be capable of having large effects on the phenotypic response (Falconer and Mackay, 1996; Dias, 2005).

The (ge) terms can be estimated in a MET design where several environments and genotypes are evaluated for a given phenotype (Isik et al., 2017). In plant breeding, the need for METs is constant given the fact that the germplasm generates new genotypes every year and the pressure of diseases and other factors are dynamic.

Such experiments require a complex set of logistical activities, leading to high costs of implementation. These trials thus have strong regulatory appeal in the seed and biotech industries around the world (Sarti, 2013).

Reliable information about GxE can help breeders make decisions on cultivar recommendations. Thus, models for the study of GxE need to be able to answer questions such as which genotypes can perform well across a set of environments and which are specifically recommended for a given environment. The answers to these questions are crucial both to broad breeding strategies, i.e., to obtain one or more genotypes that perform well in a set of environments, and to target breeding, where the best genotype is determined for a given environment (Sarti, 2019).

4.2.2 Traditional AMMI models

A simple statistical linear model can be used to estimate GxE effects from METs. The model can be written as

$$y_{ij} = \mu + g_i + e_j + (ge)_{ij} + \epsilon_{ij}, \;\; i = 1, \ldots, I, j = 1, \ldots, J, \tag{4.1}$$

where $\epsilon_{ij} \sim \mathrm{N}(0, \sigma^2)$, y_{ij} is the phenotypic response, which represents, for example, the production of a crop in tonnes per hectare, μ is the grand mean, g_i is the effect of genotype i, e_j is the effect of environment j, and $(ge)_{ij}$ represents the interaction between genotype i and environment j.

In the specification of the Equation (4.1), the term (ge) can be thought of as representing a decomposition of the residual from a more basic linear model. Gollob (1968) and Mandel (1971) proposed a method to decompose the residual term as a sum of multiplicative factors that includes the (ge) term. This yields the decomposition:

$$(ge)_{ij} = \sum_{q=1}^{Q} \lambda_q \gamma_{iq} \delta_{jq}, \tag{4.2}$$

where Q is the number of components to be considered in the analysis, λ_q is the strength of the interaction of component q, γ_{iq} represents the importance of genotype i in component q, and δ_{jq} represents the importance of environment j in component q; see Appendix 4.C for the restrictions imposed on γ_{iq}, δ_{jq} and λ_q to make the model identifiable. Hence, the complete AMMI model is written as

$$y_{ij} = \mu + g_i + e_j + \sum_{q=1}^{Q} \lambda_q \gamma_{iq} \delta_{jq} + \epsilon_{ij}. \tag{4.3}$$

The interaction terms in (4.3) are estimated by a singular value decomposition (SVD) of a matrix $\mathbf{M}$, which contains the residual values of a two-factor ANOVA model that considers the genotypes and environments as main effects. Here, λ_q is the q-th eigenvalue of the matrix $\mathbf{M}$, γ_{ik} is the i-th element of the left singular vector, and δ_{jk} is j-th element of the right singular vector obtained in the SVD (Good, 1969). In practice, the classical AMMI model can be run in `R` (R Core Team, 2020) using the package `agricolae` (de Mendiburu, 2019) or via functions programmed by the user as in Onofri and Ciriciofolo (2007).

The protocol for estimation of the terms in a standard AMMI model is given by Gauch Jr (2013). This involves the following steps:

1. Obtain the grand mean and main effects of the genotypes and environments using ANOVA with two factors based on a matrix of means containing the means of each genotype within each environment;
2. Obtain the residuals from the model above that will comprise the interaction matrix, where each row is an environment and each column a genotype;
3. Choose an appropriate value for the number of components Q;
4. Form the multiplicative terms that represent the reduced-dimension interactions via an SVD of the matrix of interaction residuals.

The rank of the matrix $\mathbf{M}$ is assumed to be $r = \min(I-1, J-1)$. Thus, the number of components Q may vary from $1, \ldots, r$. The quantity r establishes the minimum number of non-zero eigenvectors to be obtained in the SVD. Taking $Q = r$, the AMMI model would capture all the variance related to the interaction, and it would result in over-fitting. This problem is ameliorated by using a limited number of components Q. The choice of Q is related to the amount of total variability captured by the principal components (PCs) and, in general, is recommended to use a number of PCs that captures at least 80% of the total variability (Lal et al., 2020; Shafii and Price, 1998; Love et al., 2004; Tyagi et al., 2016; Dias and Krzanowski, 2006). Usually, the value of Q varies from 1 to 3.

AMMI models have been extensively used for evaluation of phenotype performance of cultivars. Nachit et al. (1992) used AMMI models to assess the performance of wheat germplasm from the International Maize and Wheat Improvement Center. Farshadfar and Sutka (2003) explored quantitative trait loci (QTL) related to adaptation in wheat. Rad et al. (2013) studied GxE for wheat in the context of drought and normal conditions. Brancourt-Hulmel and Lecomte (2003) evaluated the impact of environmental conditions in the stability of winter wheat. Badu-Apraku et al. (2012) used AMMI to study the stability of early maize genotypes in Africa. Mitroviaã et al. (2012) evaluated the performance of experimental maize hybrids using AMMI models, and Sarti (2019) studied the performance of the AMMI model in the context of simulated data. The applications of AMMI models can also be found in several other species including: a) rice (Mahalingam et al., 2006), b) barley (Mahalingam et al., 2006; Romagosa et al., 1996; Sato and Takeda, 1993; Anbessa et al., 2009), and c) sugarcane (Silveira et al., 2013).

4.2.3 Tree-based methods and BART

Introduced by Chipman et al. (2010), BART is a Bayesian model that uses a sum of trees to approximate a univariate response. In BART, each tree works as a weak learner that yields a small contribution to the final prediction. Based on a design matrix $\mathbf{X}$, the model is able to capture interactions and non-linear relationships. The BART model can be written as

$$y_i|\mathbf{x}_i, \mathcal{M}, \mathcal{T}, \sigma^2 \sim \mathrm{N}\left(\sum_{t=1}^{T} h(\mathbf{x}_i, \mathcal{M}_t, \mathcal{T}_t), \sigma^2\right), \; i = 1, \dots, n, \tag{4.4}$$

where $\mathbf{x}_i$ is the i-th row of the design matrix $\mathbf{X}$, $\mathcal{M}_t$ denotes the set of terminal node parameters of tree t, $\mathcal{T}_t$ is the set of binary splitting rules that define the tree t, and $h(\cdot) = \mu_{t\ell}$ is a function that assigns the predicted values $\mu_{t\ell} \in \mathcal{M}_t$ based on the design matrix $\mathbf{X}$ and tree structure $\mathcal{T}_t$. We let $\mathcal{M} = (\mathcal{M}_1, \dots, \mathcal{M}_T)$ and $\mathcal{T} = (\mathcal{T}_1, \dots, \mathcal{T}_T)$ denote the sets of all predicted values and trees, respectively. Chipman et al. (2010) recommend $T = 200$ as a default for the number of trees since it works remarkably well in a wide variety of applications. However, they also suggest that T can be selected through cross-validation.

Unlike other tree-based methods where a loss function is optimised to grow the trees, in BART the trees are learned using a Bayesian backfitting Markov Chain Monte Carlo (MCMC) algorithm (Hastie and Tibshirani, 2000; Gamerman and Lopes, 2006; Robert and Casella, 2013). The trees are either accepted or rejected via a Metropolis-Hastings step. In addition, the trees can be learned by four moves: grow, prune, change or swap. It is important to highlight that in all moves the splitting rule is defined by randomly selecting one covariate and one split point. In the grow move, a terminal node is selected and then two children nodes are created below it. When pruning, a parent of two terminal nodes is selected and its children nodes are removed. During the change move, a parent node is picked and its splitting rule (i.e., covariate and split point) is redefined. In the swap move, a pair of parent-child internal nodes is chosen and the splitting rules associated to the two nodes are swapped.

As a fully Bayesian model, BART assumes prior distributions on all quantities of interest. First, the node-level parameters $\mu_{t\ell}$ are assumed to be i.i.d $\mathrm{N}(0, \sigma_\mu^2)$, where $\sigma_\mu = 0.5/(k\sqrt{T})$ and $k \in [1, 3]$. Second, the residual variance σ^2 is assumed to be distributed as $\mathrm{IG}(\nu/2, \nu\lambda/2)$, where $\mathrm{IG}(\cdot)$ denotes an Inverse Gamma distribution. Third, to control how shallow/deep a tree may be, each non-terminal node has a prior probability of $\alpha(1 + d)^{-\beta}$ of being observed, where $\alpha \in (0, 1)$, $\beta \geq 0$, and d corresponds to the depth of the node; Chipman et al. (2010) recommends $\alpha = 0.95$ and $\beta = 2$ as default values. These hyperparameter values tend to select trees which are not too deep.

We remark that the priors above are a crucial element to identify σ^2 as they control the tree topology and the variability of the prediction at the terminal node level. Were these priors to be set too vague (e.g., so that deep trees were favoured), then the value of σ^2 would shrink towards zero and we would be left with an over-fitted model. Thus, these priors force the trees to be shallow and shrink the terminal node parameters towards zero such that each tree only explains a small component of the data, which leaves some variation in the residuals and resolves the identifiability issue of σ^2.

The identification of the interaction effects in BART is powered by the full conditional of the trees, which we denote by $p(\mathcal{T}_t|\mathbf{R}_t, \sigma^2)$, where $\mathbf{R}_t = \mathbf{y} - \sum_{k \neq t}^{T} h(\mathbf{X}; \mathcal{T}_k, \mathcal{M}_k)$ represents the vector of the partial residuals excluding tree t; see Appendix 4.A for details. As in BART the trees are learned by using splitting rules that are created by randomly selecting a covariate and a split point, $p(\mathcal{T}_t|\mathbf{R}_t, \sigma^2)$ is used to select only 'good' trees (i.e., trees that help reduce the residual variance).

Finally, the structure of the BART model for a continuous response can be summarised as follows. First, all $\mathcal{T}_t$ are initialised as stumps. Then, the trees are learned, one at a time, through one of the four moves previously described (grow, prune, change or swap). For each tree, the newly proposed $\mathcal{T}_t^\star$ is compared to its previous version $\mathcal{T}_t$ via a Metropolis-Hastings step taking into account the partial residuals $\mathbf{R}_t$ and the structure/depth of $\mathcal{T}_t$ and $\mathcal{T}_t^\star$. Hence, the predicted values for each terminal node ℓ of the tree t are generated. After doing that for all trees, σ^2 is updated. For a binary outcome, the data augmentation strategy of Albert and Chib (1993) can be used; see Tan and Roy (2019) and Prado et al. (2021) for more details.

Due to its flexibility and excellent performance on regression and classification problems, BART has been applied and extended to credit modelling (Zhang and Härdle, 2010), survival analysis (Sparapani et al., 2016; Linero et al., 2021; Basak et al., 2021), proteomic biomarker analysis (Hernández et al., 2015), polychotomous response (Kindo et al., 2016b), and large datasets (Hernández et al., 2018; Linero, 2018). More recently, papers exploring the theoretical aspects of BART have been developed by Jeong and Ročková (2020); Ročková and van der Pas (2020); Ročková and Saha (2019); Linero (2018).

4.2.4 AMBARTI

To insert the BART model inside an AMMI approach, we make some fundamental changes to the way the trees are learned and structured. As a first step, we can write the sum of trees inside the Bayesian version of the AMMI model as

$$y_{ij}|\mathbf{x}_{ij}, \mu, g_i, e_j, \Theta \sim \mathrm{N}\left(\mu + g_i + e_j + \sum_{t=1}^{T} h(\mathbf{x}_{ij}, \mathcal{M}_t, \mathcal{T}_t), \sigma^2\right), \tag{4.5}$$

where y_{ij} denotes the response for genotype i and environment j, $\Theta = (\mathcal{M}_t, \mathcal{T}_t, \sigma^2)$, μ is the grand mean, and g_i and e_j denote the effects of genotypes and environments, respectively. The component $\sum_{t=1}^{T} h(\mathbf{x}_{ij}, \mathcal{M}_t, \mathcal{T}_t)$ is similar to that presented in (4.4) and $\mathbf{x}_{ij}$ contains dummy variables that represent combinations of g_i and e_j. In order to get the posterior distribution of the new parameters, we assume that $\mu \sim \mathrm{N}(0, \sigma_m^2)$, $g_i \sim \mathrm{N}(0, \sigma_g^2)$ and $e_j \sim \mathrm{N}(0, \sigma_e^2)$, as well as that $\sigma_g^2 \sim \mathrm{IG}(a_g, b_g)$ and $\sigma_e^2 \sim \mathrm{IG}(a_e, b_e)$.

To facilitate the specification of the prior distributions, we transform the response variable to lie between -0.5 and 0.5 as in Chipman et al. (2010), and we adopt the prior specification for (most) $\mu_{t\ell}$ and σ^2 from BART. We set the priors for μ, g_i, and e_j to have mean zero, and we specified the prior distributions for their variance to have mean one and diffuse variance. Although the choices above seem ad-hoc, they worked remarkably well across different synthetic and real-world datasets.

At first look our model is similar to the semi-parametric BART proposed by Zeldow et al. (2019). However, our approach differs in that i) we do not partition $\mathbf{x}_{ij}$ into two distinct subsets, as the dummy variables (g_i and e_j) that are used in the linear predictor are also contained in $\mathbf{x}_{ij}$; ii) most importantly, we add to the tree-generation process in BART a 'double grow' and a 'double prune' steps so that we guarantee that the trees will include at least one g_i and one e_j as splitting criteria and; iii) unlike Zeldow et al. (2019), we do not use the full residuals $\mathbf{R} = \mathbf{y} - \sum_{t=1}^{T} h(\mathbf{X}, \mathcal{M}_t, \mathcal{T}_t)$ to update the linear predictor estimates, but rather the response variable only. This last modification makes AMBARTI a modularised (Bayarri et al., 2009) or a cut Bayesian (Plummer, 2015) model since only $\mathbf{y}$ is used to update the main effects and not $\mathbf{R}$. Hence, AMBARTI is a 'cut' Bayesian model as opposed to a fully Bayesian approach. Nevertheless, we provide details below about the motivation and validity of our model.

The rationale of the double grow and double prune moves is to force the trees to exclusively work on the interactions between g_i and e_j. In doing so, we remove the chance that the 'single' moves split on a single g_i or e_j variable, which would lead to confounding with the main marginal genomic or environment effects. For example, in the double grow, rather than randomly selecting one covariate and one split point when growing a stump, a variable $g^\star$ is chosen and then another variable $e^\star$ is randomly selected and both define the splitting rules of the corresponding tree. Here, the dummy variables $g^\star$ and $e^\star$ are sampled from the sets of all possible combinations of g_i and e_j, respectively. Conversely, in the double prune move a tree is pruned twice to prevent it from having a single g_i or e_j. Thus, the resulting tree from this double move will always be a stump. Regarding the change and swap moves, we point out that they were kept as 'single' moves since their double counterparts would not help induce interactions between g_i and e_j. However, we introduced validity checks on the structure of the trees resulting from these moves to guarantee that only valid splitting rules are proposed/accepted.

We also modified the prior on the node-level parameters so that terminal nodes which the ancestor nodes do not have an interaction between g_i and e_j are set to zero. To illustrate this, we refer to Figure 4.2.1. The left-most terminal nodes (μ_1 and μ_2) are defined by splitting rules which contain a set of genotypes ($g_{1,3}$) and environments ($e_{4,5,6}$). However, the terminal node on the right hand side (μ_3) has only $g_{1,3}$ as its ancestor, which is not desirable since the main effects for the genotypes are estimated in the linear predictor. To avoid this, we modify the prior only for terminal node parameters like μ_{t3} so that their posterior values are shrunk to zero (i.e., we assume that $\mu_{t3} \sim \mathrm{N}(0, \sigma_\mu^2 \approx 0)$).

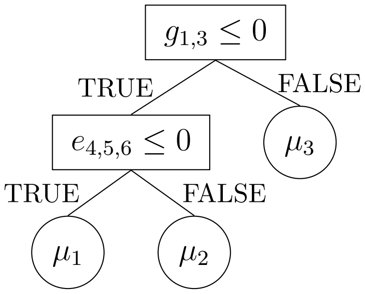


Figure 4.2.1: An example of a tree generated from a double grow move based on the sets of dummy variables generated for g and e. $g_{1,3}$ denotes a dummy variable that combines genotypes 1 and 3, while $e_{4,5,6}$ binds environments 4, 5, and 6.

We allow for extra flexibility in the identification of the interaction effects by setting up the design matrix of the BART component to contain all possible combinations of genotype and environment effects. For instance, when $I = 5$ genotypes and $J = 6$ environments, $2^{I-1} - 1 = 15$ and $2^{J-1} - 1 = 31$ unique dummy variables are generated for g and e, respectively; see Wright and König (2019) for splitting methods on categorical covariates. In this case, there are 46 dummy variables available so that BART can use to identify possible interaction between genotypes and environments. To illustrate this, in Figure 4.2.1 a dummy variable is sampled from the set of 15 dummy variables related to g along with a dummy variable from the set of 31 dummy variables associated with e.

The current R packages that offer implementations of BART, such as `bartMachine` (Kapelner and Bleich, 2016), `BART` (McCulloch et al., 2020) and `dbarts` (Dorie, 2020), represent categorical variables with $k > 2$ levels into k dummy variables. In contrast, in random forests the standard approach to deal with categorical variables is to create all the $2^{k-1} - 1$ 2-partition combinations (Wright and König, 2019) subject to a not too big k.

The idea of performing small modifications to the trees through k dummy variables as opposed to $2^{k-1} - 1$ is tempting, especially when the numbers of genotypes and/or environments are large. One important aspect is that in BART the default prior on the tree structure favours shallow trees over deep trees so that small modifications through the grow move (considering k dummy variables) would not be ideal since deep trees are unlikely. Thus, the learning process should take place in the form of swap and change moves, since both do not change the topology of the tree but only the splitting rules. However, any modifications proposed through the change move incur the same issue of having to sample from the subset that contains $2^{k-1} - 1$ combinations since we are interested in finding/grouping similar genotypes and environments.

In the initial versions of AMBARTI, we considered the set of k dummy variables over the set of $2^{k-1} - 1$ combinations, and we also modified the prior on the trees so that it would not overly penalise deep trees. However, the results with k dummy variables were not encouraging, whilst the change to the set of $2^{k-1} - 1$

combinations vastly improved the prediction performance.

We recall that before estimating the parameters in the bilinear term of the AMMI model in (4.3), the effects of g_i and e_j are estimated via a linear regression. Hence, the residuals are organised into an $I \times J$ matrix, in which the rows and columns sum to zero, and then an SVD is performed on the residual matrix. Due to this sum-to-zero constraint, the resulting output from the bilinear term sums to zero within each g_i and e_j. In a similar fashion, a 'fully Bayesian AMBARTI'[4] would have to guarantee that the predictions from the BART component sum to zero within the g_i and e_j in order to be comparable to AMMI. However, if we impose the sum-to-zero constraint on the BART predictions within the g_i and e_j, this is equivalent to cutting feedback (i.e., remove without marginalisation) the BART predictions from the full conditionals of g_i and e_j; see (4.11) and (4.12) in Appendix 4.A. In Section 4.4, we compare AMBARTI with a set of comparators that includes a fully Bayesian AMMI model, which performs poorly compared to AMBARTI and other interactions models.

The motivation behind cutting feedback in Bayesian models is studied in Bayarri et al. (2009) and Plummer (2015). In the first work, the authors list a set of situations where the 'modularisation' (i.e., cut feedback) of Bayesian models can be useful. They motivate the use of modularisation through examples in the analysis of computer models and present possible reasons for performing it. They point out that Bayesian models that eventually incur identifiability/confounding issues might require modularisation, especially when there is interest in determining effects of interest and not only the overall prediction. We highlight that were AMBARTI fully Bayesian rather than a cut Bayesian model, there would be some bias/identifiability issues, due to the sum-to-zero condition mentioned above, between the g_i and e_j and the BART component. Finally, Plummer (2015) highlights some issues that can affect the sampling of the parameters from a cut Bayesian model and also proposes solutions to circumvent some of the issues. In Section 4.2.4.1, we tackle Plummer's concerns by showing that the posterior samples from

[4] Recall iii) which states that the full residuals $\mathbf{R} = \mathbf{y} - \sum_{t=1}^{T} h(\mathbf{X}, \mathcal{M}_t, \mathcal{T}_t)$ are not used to estimate the main effects of g_i and e_j. Hence, a fully Bayesian AMBARTI would use $\mathbf{R}$ to update the linear predictor estimates as opposed to the response variable only.

AMBARTI are valid.

An appealing advantage of AMBARTI over AMMI is that it does not require the specification of the number of components Q in the bilinear sum and does not require complex orthonormality constraints on the interaction structure; see Appendix 4.C for the constraints of the AMMI model. In a Bayesian context, these constraints can lead to complex prior distribution choices for implementation of AMMI (as in Josse et al., 2014; Crossa et al., 2011). Furthermore, although AMBARTI adds a computational cost to the BART model, we have found this to be negligible for standard MET datasets that usually have values of I and J up to the low tens or hundreds.

An additional advantage of using a (cut) Bayesian approach as in AMBARTI is that we have access to the posterior distribution of each parameter. As the model is fitted, we are thus able to ascertain the general levels of uncertainty in each g_i or e_j component, which may assist with future experimental designs. Similarly, the interaction term is also estimated probabilistically, and so may avoid interpretation errors associated with, e.g., biplots from a traditional AMMI model.

In terms of estimation, the AMBARTI model can be fitted as follows. First, the parameter estimates g_i and e_j are sampled taking into account the response variable $\mathbf{y}$ (not the residuals). Then, one at a time, the trees are updated via partial residuals $\mathbf{R}_t = \mathbf{y} - \mu - g_i - e_j - \sum_{k \neq t}^{T} h(\mathbf{X}, \mathcal{M}_k, \mathcal{T}_k)$. Hence, the terminal node parameters are generated and the sample variance is updated. In the end, posterior samples associated with μ, g_i, e_j, σ_g^2, σ_e^2, and $\mathcal{T}_t$ are available, which allow for the calculation of credible intervals and evaluation of the significance of the parameter estimates; see Algorithm 3 in Appendix 4.A for more details.

4.2.4.1 Validity of the posterior sampling in AMBARTI

The AMBARTI model estimates the genotype and environment main effects taking into account only the response and disregarding the BART component. In contrast, when fitting the BART component, the effects of g and e are taken into account. We set up AMBARTI in this way as we aimed to compare its results to the frequentist AMMI model. We recall that in the classical AMMI the estimation of

g and e is first carried out and then the interactions between them are estimated without the bilinear term being fed back into the process to update the estimates of g and e. As briefly mentioned above, AMBARTI can be seen as a modularised model (Bayarri et al., 2009) or a cut Bayesian model which uses the 'naive cut algorithm' (Plummer, 2015). Nonetheless, we show that our model, under the naive cut algorithm, generates valid posterior samples following Plummer (2015).

Before introducing the AMBARTI model as a cut Bayesian approach following the work of Plummer (2015), we state what a cut Bayesian model is. Broadly speaking, a cut Bayesian model differs from a full Bayesian model in terms of how it samples the parameters in the model. To illustrate this, let's consider that we observe a univariate response y_i in the following form:

$$\mathbf{y} = \mathbf{W}\phi + \mathbf{X}\theta + \epsilon, \epsilon_i \sim \mathrm{N}(0, \sigma^2),$$

where $\mathbf{W}$ and $\mathbf{X}$ are known quantities and ϕ and θ are unknown parameter vectors of interest. In a full Bayesian model, the parameters in the model above can be estimated via three full conditional distributions: i) $p(\theta|\mathbf{y}, \mathbf{W}, \mathbf{X}, \phi, \sigma^2)$, ii) $p(\phi|\mathbf{y}, \mathbf{W}, \mathbf{X}, \theta, \sigma^2)$ and iii) $p(\sigma^2|\mathbf{y}, \mathbf{W}, \mathbf{X}, \phi, \theta)$. In this setting, a natural way of sampling, say, θ but not conditioning it on, say, ϕ would be to integrate ϕ out of the update of θ (i.e., $p(\theta|\mathbf{y}, \mathbf{W}, \mathbf{X}, \sigma^2) = \int p(\theta|\mathbf{y}, \mathbf{W}, \mathbf{X}, \phi, \sigma^2)p(\phi)d\phi$). However, under a cut Bayesian model, the 'full' conditional distribution of θ simply disregards the information from ϕ, but without marginalising ϕ. This is approach can be useful in situations where there is conflicts of information in the model or convergence/missing issues in the MCMC scheme; see Plummer (2003) and Bayarri et al. (2009) for examples.

A simple way of writing our model, given the tree structures, is:

$$\mathbf{y} = \underbrace{\mathbf{W}\phi}_{\text{g + e}} + \underbrace{\mathcal{T}\theta}_{\text{BART}} + \epsilon, \tag{4.6}$$

where $\epsilon_i \sim \mathrm{N}(0, \sigma^2)$, $\mathbf{W}$ is a binary matrix of values allocating $\mathbf{y}$ to the correct genotype/environment, ϕ are the g and e parameters, $\mathcal{T}$ is a matrix that relates each of the trees to the individual observations, and θ are parameters of the interactions. For simplicity and without loss of generality, we assume that σ^2 is known.

Under this formulation, we have that the quantities of interest ϕ, $\mathcal{T}$ and θ can be estimated via their full conditional distributions:

$$\begin{gathered} p(\phi|\mathbf{y}, \mathbf{W}, \mathcal{T}, \theta, \sigma^2), \\ p(\mathcal{T}|\mathbf{y}, \theta, \phi, \sigma^2), \\ p(\theta|\mathbf{y}, \mathcal{T}, \phi, \sigma^2). \end{gathered} \tag{4.7}$$

Figure 4.2.2 presents AMBARTI from the perspective of a cut Bayesian model. The likelihood function depends on θ, $\mathcal{T}$ and ϕ, and all parameters are of interest. The graph is divided in two parts (G_1 and G_2). The idea of the naive cut algorithm is that when constructing the full conditionals for parameters in G_1, likelihood terms involving random variables in G_2 are ignored (Plummer, 2015).

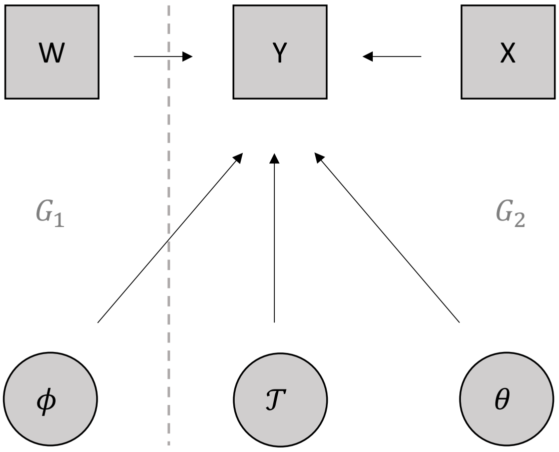


Figure 4.2.2: Graphical representation of the AMBARTI model.

The 'cut' conditional distributions for ϕ and θ are given by:

$$\begin{gathered} p(\phi|\mathbf{y}, \mathbf{W}, \sigma^2), \\ p(\mathcal{T}|\mathbf{y}, \phi, \sigma^2), \\ p(\theta|\mathbf{y}, \mathcal{T}, \phi, \sigma^2), \end{gathered} \tag{4.8}$$

where $p(\mathcal{T}|\mathbf{y}, \phi, \sigma^2) = \int p(\mathcal{T}|\mathbf{y}, \phi, \theta, \sigma^2)p(\theta)d\theta$. The full conditional in (4.7) considers the *full* Bayesian model, whilst in (4.8) there is no dependence/influence of $\mathcal{T}$ and θ over ϕ. Plummer (2015) describes three situations when the feedback from one component may not be helpful to the other. First, when the relation between the response and θ is speculative. Second, when there is conflict between the sources such that $p(\phi|\mathbf{y}, \mathbf{W}, \mathcal{T}, \theta, \sigma^2)$ is very different from $p(\phi|\mathbf{y}, \mathbf{W}, \sigma^2)$. Third, when there are computational issues in terms of convergence and mixing.

Plummer (2015) states that the cut in the feedback of $\mathcal{T}$ and θ into the conditional distribution for ϕ may lead to lack of convergence to the (cut) posterior distribution. However, for the cut algorithm to draw approximate samples from the cut model, one of the two conditions below need to be verified.

1. The transition from ϕ to ϕ', $\mathcal{T}$ to $\mathcal{T}'$ and σ^2 to $\sigma^{2\prime}$ cannot cause significant changes in the conditional distribution for $\mathcal{T}$ and θ, i.e., the limit of

$$\frac{p(\mathcal{T}'|\mathbf{y}, \phi', \sigma^{2\prime})}{p(\mathcal{T}'|\mathbf{y}, \phi, \sigma^2)} \to 1 \text{ and} \qquad (4.9)$$
$$\frac{p(\theta'|\mathbf{y}, \mathcal{T}', \phi', \sigma^{2\prime})}{p(\theta'|\mathbf{y}, \mathcal{T}, \phi, \sigma^2)} \to 1.$$

   Under the cut Bayesian model framework, this condition makes sense to parameters that do not have closed-form full conditionals (e.g., the full conditional for the tree in AMBARTI). In an attempt to satisfy this condition, Plummer (2015) proposes an algorithm based on tempered transitions which only allows small steps for ϕ, $\mathcal{T}$ and σ^2.

2. The probabilities of moving from $\mathcal{T}$ to $\mathcal{T}'$ and θ to θ', denoted by $\mathcal{T} \to \mathcal{T}'$ and $\theta \to \theta'$, do not depend on $\mathcal{T}$ and θ, respectively, i.e., the limit of

$$\frac{p(\mathcal{T} \to \mathcal{T}'|\phi, \sigma^2)}{p(\mathcal{T}'|\mathbf{y}, \phi, \sigma^2)} \to 1 \text{ and}$$
$$\frac{p(\theta \to \theta'|\phi, \mathcal{T}, \sigma^2)}{p(\theta'|\mathbf{y}, \mathcal{T}, \phi, \sigma^2)} \to 1. \qquad (4.10)$$

We remark that all parameters in the AMBARTI model attain the condition in (4.10), except $\mathcal{T}$. For example, the probability of $\theta \to \theta'$ depends exclusively on the posterior conditional distribution $p(\theta'|\mathbf{y}, \mathcal{T}, \phi, \sigma^2)$, which in turn does not depend on the previous θ since it has closed-form. To make our point clearer, recall expression (4.6) and that in BART, *a priori*, $\mu_{t\ell} \sim \mathrm{N}(0, \sigma_\mu^2)$, which is completely specified with no dependence on its previous values. In addition, the posterior conditional distributions of g_i, e_j, $\mu_{t\ell}$, and σ^2 have all closed-form expressions, which are presented in Appendix 4.A. In relation to the transition probability of an individual tree $\mathcal{T}_t \to \mathcal{T}_t'$, we point out that it does rely upon the previous tree $\mathcal{T}_t$ since the transition kernel $q(\mathcal{T}_t \to \mathcal{T}_t')$, specifically for the grow and prune moves, depends on the number of terminal and internal nodes of $\mathcal{T}_t$; see Appendix A of Kapelner and Bleich (2016) for further details on the transition probabilities of the moves in BART. However, we empirically show in Appendix 4.B that the transition $\mathcal{T}_t \to \mathcal{T}_t'$ under the naive and tempered cut algorithms do not differ, which supports the condition in (4.9) for AMBARTI.

4.3 Simulation Study

In this Section, we compare AMMI, the Bayesian version of AMMI (B-AMMI) proposed by Josse et al. (2014), and AMBARTI using the root mean squared errors (RMSE) for predicted values $\hat{y}$ and for the interaction term on out-of-sample data. Our simulation experiment was carried out considering two scenarios. In the first, we simulated data from the AMMI model with $Q = \{1, 2, 3\}$ and then fitted AMBARTI, B-AMMI, and AMMI. In the second scenario, we simulated data from the AMBARTI model and then fitted AMBARTI and three AMMI (and B-AMMI) models with different number of components to describe the interactions (i.e., $Q = \{1, 2, 3\}$). In both scenarios, we fitted the models to a training set with $I \times J$ observations and evaluated the performance on an out-of-sample test set of the same size.

For both scenarios, we set $I = J = \{10, 25\}$, $\mu = 100$ and generated g_i and e_j from $\mathrm{N}(0, \sigma_g^2)$ and $\mathrm{N}(0, \sigma_e^2)$, respectively, where $\sigma_g = \sigma_e = \{1, 5\}$. The parameters γ_{ik} and δ_{jk} were generated from $\mathrm{N}(0, 1)$ and then the orthornormality constraints were applied following the results presented in Appendix 4.B. In addition, for $Q = 1$, we consider two values for λ (i.e., $\lambda = \{8, 12\}$); for $Q = 2$, $\lambda = (\{12, 8\}, \{12, 10\})$ and; for $Q = 3$, $\lambda = \{12, 10, 8\}$. In the simulation from AMBARTI, we set $T = 200$ trees and generated each tree by using the 'double grow' move considering $2^{I-1} - 1$ possible covariates for g_i and $2^{J-1} - 1$ for e_j.

Finally, the AMMI model used in the simulations is presented in Equation 4.3, which represents a completely randomised trial design. The AMBARTI model used is shown in Equation 4.5.

4.3.1 Simulation results

We start simulating synthetic data from the AMMI system, which is the harshest test for the AMBARTI model. Figure 4.3.1 shows the RMSE values for $\hat{y}$ based on the out-of-sample sets of three models considering 10 Monte Carlo repetitions. The data sets considered in this Figure were simulated considering $I = 10$ genotypes and $J = 10$ environments, with different values of $Q = \{1, 2, 3\}$, and two values for the genotypic and environmental standard errors $\sigma_g = \{1, 5\}$ and $\sigma_e = \{1, 5\}$, respectively.

As the data were simulated from the AMMI equation, we would expect that the AMMI model would perform exceedingly well, and this is what we see in general considering all the results of Figure 4.3.1. More specifically, we can see in the first upper panel that AMBARTI has higher RMSEs compared to AMMI for all values of Q. Further, we see that B-AMMI has similar performance to the frequentist AMMI only for $Q = 1$. As the number of components in the bilinear term increases, the results from B-AMMI deteriorate compared to AMMI and AMBARTI. In addition, it is possible to note that there is no clear effect of σ_g or σ_e on the RMSEs. However, even with the AMMI model presenting the best results, AMBARTI demonstrates highly competitive performance, with RMSE values around 17% higher than that of the AMMI model.

Figure 4.3.2 shows the results of the second simulation scenario, where the data were simulated from the AMBARTI equation. Again, different combinations of parameters were used in the simulation of the training and out-of-sample sets. The upper panels show results for $I = 10$ genotypes and $J = 10$ environments; the lower ones for $I = 25$ and $J = 25$. Furthermore, three AMMI and three B-AMMI models were fitted considering Q from 1 to 3. In this case, the AMMI model, even with high values of Q, performs very poorly with RMSE values 3 times higher on average than that of AMBARTI for $I = 10$. For $I = 25$, the AMMI model with $Q = 3$ is competitive with AMBARTI, with the latter being slightly better. In addition, we can see that AMBARTI presents better results when compared with

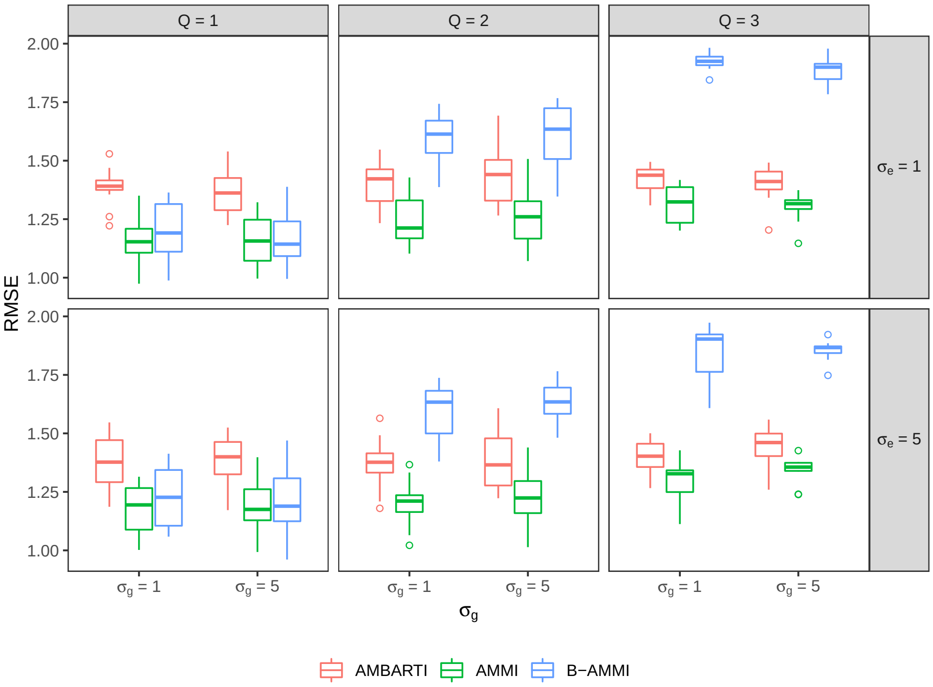


Figure 4.3.1: Out-of-sample RMSE for $\hat{y}$ based on the results of AMMI, B-AMMI, and AMBARTI for data simulated from the AMMI model with $I = J = 10$. The different panels contain 10 Monte Carlo repetitions and represent different combinations of the simulated parameters for the creation of the data set. Unsurprisingly, AMMI performs very well here, with AMBARTI having RMSE values around 17% higher.

B-AMMI, regardless of the value of Q. In this comparison, it is worth mentioning that more complex possibilities of interactions may be obtained when simulating from AMBARTI compared to AMMI.

The next important comparison to be made between AMBARTI and AMMI is related to the interaction term (i.e., the bilinear term for AMMI and the BART component for AMBARTI). Such tests are shown in Figures 4.3.3 and 4.3.4, where we show the RMSE performance only for the interaction component.

Figure 4.3.3 presents the RMSEs associated solely with the interaction terms from AMMI, B-AMMI, and AMBARTI when the data are simulated from AMMI (which has a bilinear structure for the interactions). The results are presented consider-

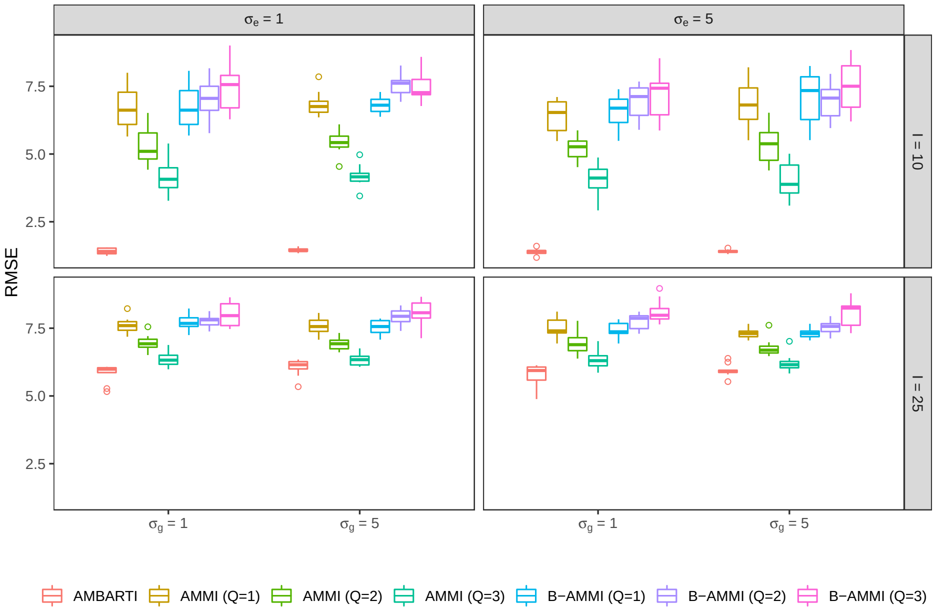


Figure 4.3.2: Out-of-sample RMSE for $\hat{y}$ based on the results of AMMI (with varying Q), B-AMMI, and AMBARTI for data simulated from the AMBARTI model with $I = J = 10$ and $I = J = 25$. The boxplots contain 10 Monte Carlo repetitions. The AMMI RMSE values are on average 3 times higher than that of AMBARTI for $I = 10$.

ing 10 genotypes and 10 environments with different combinations of genotypic and environmental variances. The performance of AMMI is optimal compared to AMBARTI, though the difference between the two is lessened with more complex AMMI model structures (i.e., $Q = 3$). In Figure 4.3.4, the values of RMSE are presented for data sets simulated from AMBARTI. In the margins of the figure, the parameters used in the simulations can be found. The RMSE values show that AMMI performs worse than AMBARTI in all scenarios, and in the same cases AMMI RMSEs are three times higher on average than those of AMBARTI for the scenario with $I = 10$ genotypes.

In summary, the information presented in Figures 4.3.2 and 4.3.4 shows that the AMMI model fails for the complex interactions that can be obtained in the AMBARTI simulated data sets. From a quantitative genetics/biological perspective, there is no reason for the structure of interactions between genotypes and environ-

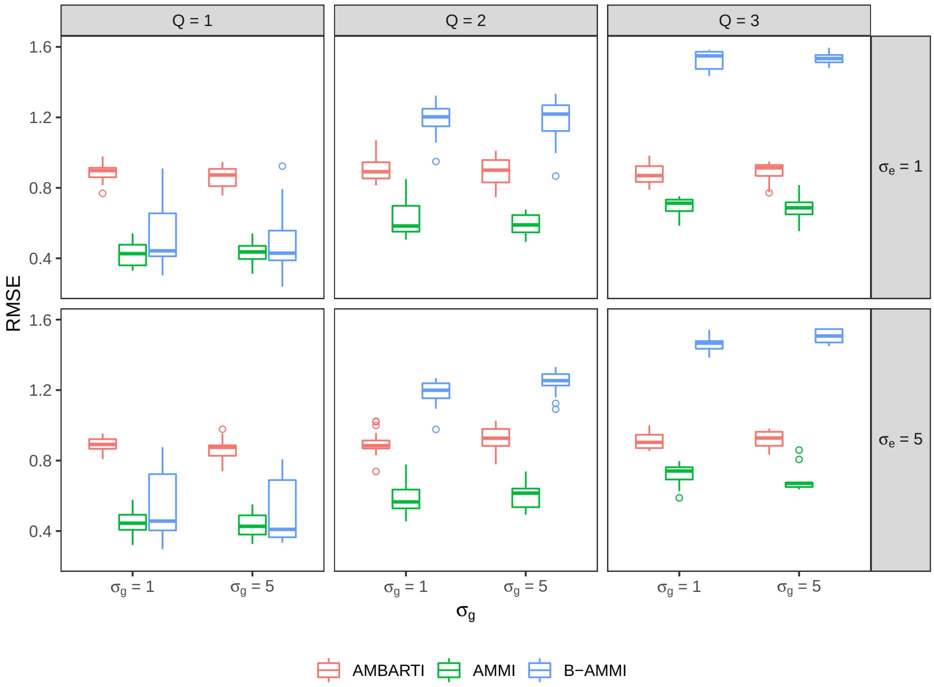


Figure 4.3.3: Out-of-sample RMSE of the interaction term of AMMI models for data simulated from AMMI. The different panels show the different parameter values used in the simulation. The performance of AMMI here is optimal, with AMBARTI performing slightly worse than AMMI when $Q = 3$.

ments to be modelled strictly by a bilinear structure, as more complex structures can be assumed to be present in nature. Thus, AMBARTI may be a more suitable model to estimate the interaction structure in real-world applications.

4.4 Case study: Irish VCU InnoVar wheat data

In addition to the simulation study, real data sets were used to evaluate the performance of AMBARTI. We compare our new approach not only to AMMI and B-AMMI, but also to more sophisticated interaction detection models including smoothing splines ANOVA models (SS-ANOVA; Gu, 2014) and Bayesian multivariate adaptive regression splines (B-MARS; Denison et al., 1998). To run SS-ANOVA and B-MARS, we used the R packages `gss` (Francom and Sansó, 2020) and `BASS` (Gu, 2014), respectively. A set of value of cultivation and usage (VCU) experiments conducted in Ireland between the years of 2010 and 2019 were con-

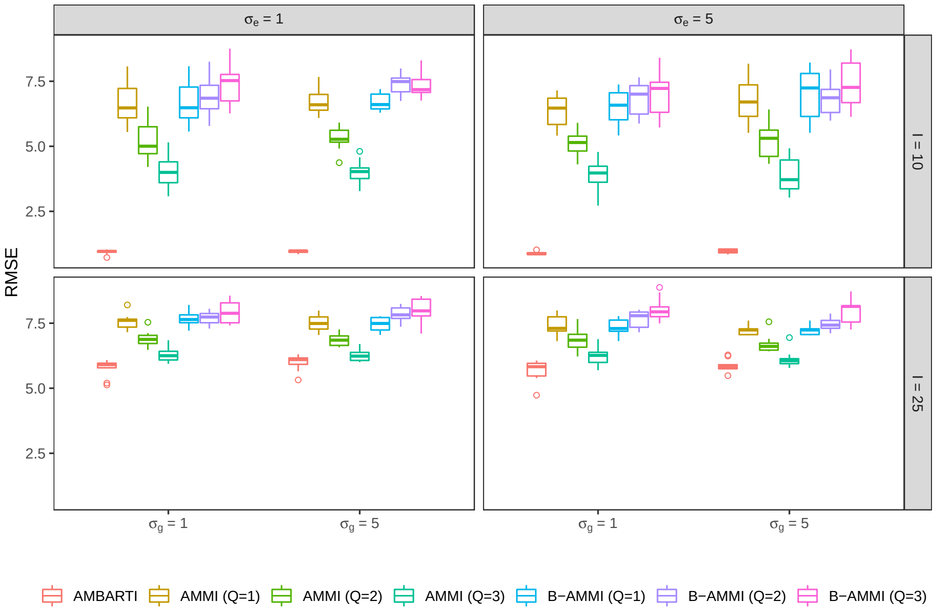


Figure 4.3.4: Out-of-sample RMSE of the interaction term of AMBARTI and AMMI models for data simulated from AMBARTI. The different panels show the different parameter values used for the simulation. It appears that the AMMI structure, even with $Q = 3$ cannot capture the interaction behaviour present in the AMBARTI model for $I = 10$.

sidered, and such experiments evaluated the performance of genotypes of wheat *Triticum aestivum L.* across the country for regulatory purposes (i.e., registration of new varieties). Here, our phenotypic response variable is the production of wheat in tonnes per hectare. The design of experiments used was that of a randomised complete block design with 4 replicates. VCUs alongside distinctness, uniformity and stability (DUS) are the most important kind of regulatory multi environmental trials conducted around the world.

The data were kindly provided by the Irish Department of Agriculture, Food, and Marine. Both genotypes and environments were anonymised. These historical Irish VCUs form part of the Horizon2020 EU InnoVar project database (www.h2020innovar.eu). The project aims to build and improve technical solutions for cultivar recommendation based on genomic and phenomic parameters. The models were fitted for all years available (summarised in Table 4.4.1), but for brevity we show detailed plots only for the year 2015, which has 4 (replicates) × 9 (enviroments) × 18 (genotypes) = 648 observations.

We compare the models by evaluating the estimated values of the genotype and environment effects, and the predictions of interaction behaviour evaluated as: $\hat{(ge)}_{ij} = y_{ij} - \hat{g}_i - \hat{e}_j$. As the Irish data from the InnoVar project correspond to a block design with 4 replicates, to fit all models we randomly selected two replicates and then averaged the response variable across them. This is a common practice in the analysis of GxE experiments with AMMI and B-AMMI models (Josse et al., 2014; Crossa et al., 2011) as these models cannot deal with replicates in an experiment. However, this pre-processing is not needed for AMBARTI. To validate the models, we use the remaining two replicates to calculate the RMSE for $\hat{y}$.

The estimates of the genotype effects g_i and environment effects e_j are shown in Figure 4.4.1 and 4.4.2, respectively. The results for AMBARTI show the samples obtained from the posterior distributions, while for AMMI, as it is a frequentist method, we adopted the approach of Goodman and Haberman (1990) where the results correspond to samples from the estimator distributions of g_i and e_j. The rationale of using samples from the estimator distribution is to be able to compare AMBARTI and AMMI results not only in terms of point estimates but also in terms of the uncertainty associated to the point estimates. Although we have tested different number of components Q for AMMI, we remark the parameter estimates g_i and e_j do not change regardless of Q (i.e., only the bilinear term depends on Q; see Equation (4.2)).

In Figure 4.4.1, we can see that the credible intervals associated with the main effects of genotypes for AMBARTI are narrower than the confidence intervals from the AMMI model. This is somewhat expected and occurs as BART is able to capture the interactions between g_i and e_j along the MCMC process, which in turn decreases the residual variance σ^2. Hence, as the full conditionals of the g_i and e_j depend on σ^2, their estimates become less uncertain as σ^2 gets smaller; see Equations (4.11) and (4.12) in Appendix 4.A. For the sake of visualisation, we decided to omit the B-AMMI results as they presented too many values that exceeded −10 and 10.

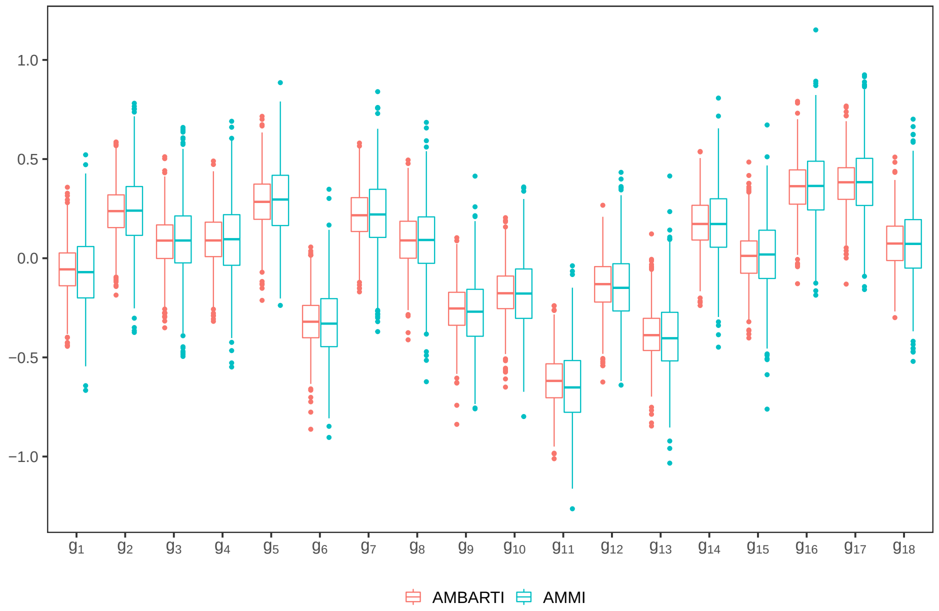


Figure 4.4.1: For AMBARTI, the boxplots represent the posterior distribution of genotype effects for the Irish VCU Innovar data set for 2015. As the parameters estimates from the AMMI model are obtained under the frequentist paradigm, their boxplots summarise samples from the estimator distribution of the genotype effects as presented in Goodman and Haberman (1990).

A more complete comparison across all years is shown in Table 4.4.1. In this table, we calculated the predicted values $\hat{y}$ on the out-of-sample data. We can see that RMSEs obtained with AMBARTI are smaller than the ones returned by the AMMI model for most years, thus highlighting that the AMBARTI model can more accurately estimate the marginal effects along with interaction component. Further, we can note that the B-AMMI presents the worst results among all methods considered. One could expect the results from AMMI and B-AMMI would be similar, but the Bayesian version is different in spirit to its frequentist counterpart, as it does not estimate the main effects and bilinear term in a two-stage approach. Instead, B-AMMI obtains the parameters through a one-stage procedure and assumes priors that do not take into account the orthormality constraints, which are fundamental for the identifiability of the frequentist AMMI model. In the B-AMMI, the orthormality constraints are applied via a post-processing on the

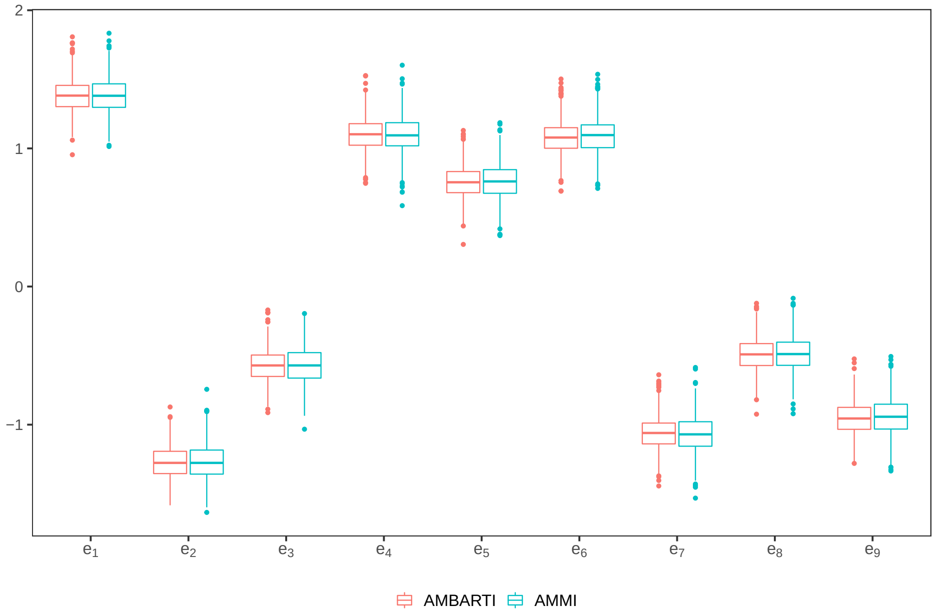


Figure 4.4.2: For AMBARTI, the boxplots represent the posterior distribution of environment effects for the Irish VCU Innovar data set for 2015. As the parameters estimates from the AMMI model are obtained under the frequentist paradigm, their boxplots summarise samples from the estimator distribution of the environment effects as presented in Goodman and Haberman (1990).

parameter estimates after the model is fitted. However, we observed the B-AMMI model with and without the post-processing may have very different performance.

Although the results for SS-ANOVA and B-MARS are competitive with AMBARTI, they work differently and do not return results in the format needed to fit the purpose of the analysis. For instance, SS-ANOVA returned interactions that do not make sense (e.g., interactions between environments or interactions between genotypes). In contrast, B-MARS is similar to BART in the sense that it does not require any specification of main/interaction effects via a linear predictor but with the advantage that it decomposes the impact of individual and interaction effects on the response.

Another modelling alternative (not shown here) is a 2-way ANOVA model, which can be used when there are replicates for each combination of genotype and environment, since there is sufficient degrees of freedom to estimate the main and

Year	AMBARTI	AMMI	B-AMMI	SS-ANOVA	B-MARS
2010	**0.81**	0.83	2.21	0.82	0.87
2011	**0.62**	0.63	2.66	0.65	0.69
2012	**0.45**	0.47	2.53	0.60	0.65
2013	0.57	0.57	1.48	**0.55**	0.56
2014	0.55	0.54	1.93	**0.53**	0.54
2015	**0.46**	0.46	1.75	0.52	0.55
2016	**0.44**	0.45	1.98	0.50	0.50
2017	**0.51**	0.53	2.24	0.60	0.62
2018	0.74	0.73	1.09	**0.70**	0.70
2019	**0.56**	0.58	1.26	0.72	0.70

Table 4.4.1: RMSE for $\hat{y}$ on out-of-sample data considering all years in the historical Irish VCU Innovar data. The values of RMSE obtained with AMBARTI are smaller than the ones obtained via AMMI and Bayesian AMMI models (both with $Q = 3$) for all years considered. The values in bold correspond to the smallest RMSE within each year.

interaction effects. However, this model does not fit the purpose of our analysis as i) GxE data might not have replicates and ii) this model decomposes the interaction term by combining one genotype and one environment only, since higher order interactions cannot be considered.

Regarding the computational time, AMBARTI took about 6 minutes on average to run, considering 50 trees, 1000 iterations as burn-in and 1000 iterations as post burn-in. This time was registered in a MacBook Pro 2.3 GHz Dual-Core Intel Core i5 with 8GB memory. AMMI took just seconds. This difference could be reduced by optimising the AMBARTI implementation using routines in C++ similar to those for BART implementations in R packages `BART` (McCulloch et al., 2020) and `dbarts` (Dorie, 2020). However, we believe AMBARTI's superior performance and posterior estimation of uncertainties outweighs the longer computational time.

4.4.1 New visualisations for AMBARTI main effects and interactions

One of the key outputs of the standard AMMI model is the biplot (Gabriel, 1971), which assists in the determination of important GxE interactions and may be used for cultivar recommendation. In Figure 4.4.3, a biplot for the Irish VCU Innovar data set for 2015 based on the results of the AMMI model shows how the genotypes and environments interact. In the upper-right quadrant, the biplot suggests that e_6 interacts positively with g_{15} and g_{11}, while the lower-left quadrant indicates that e_2 and e_5 interact negatively with g_2 and g_{12}. In practice, knowing which genotypes and environments interact positively/negatively is a valuable information, since it helps farmers in the decision making. However, these plots display only the interaction measure, thus missing i) the key marginal effects that may also come into play and ii) the uncertainty associated with both the marginal and interaction effects. For example, a certain genotype and environment may have a strong positive interaction, but if the genotype is consistently poor in all environments, this may not be clear in the biplot. Instead, we introduce new types of plots that give this full consideration.

Our first new plot is based on a heat map adapted to display both the GxE interactions (along with the marginal effects) and the predicted yields from the AMBARTI model. In Figure 4.4.4, we display the GxE interactions in the centre of the plot and the marginal effects for both environment and genotype as separate bars in the margins. The ordering applied to Figure 4.4.4 is in terms of the marginal effects for both environment and genotype, and displays low values in red to high values in blue. As both the GxE interactions and marginal effects are on the same scale and are centred around zero, we display them using only one legend and use a divergent colour palette. A diverging colour palette uses two diverging hues to represent the extremes and highlights the midpoint with a light colour. This allows for quick identification of the GxE interactions and to observe which of the environments or genotypes are the most or least optimal.

In Figure 4.4.5, we show the ordered heat map of the predicted yields (as opposed to their component parts shown in Figure 4.4.4) for each combination of environment and genotype for the AMBARTI model. In this case, we use the same ordering as that in Figure 4.4.4 with high values being generally displayed in the top left, moving to low values at the bottom right, with the units for the plot being the same as that of the phenotype (i.e., yield/production of grains in tonnes per hectare). For this plot, we use the same diverging colour palette as

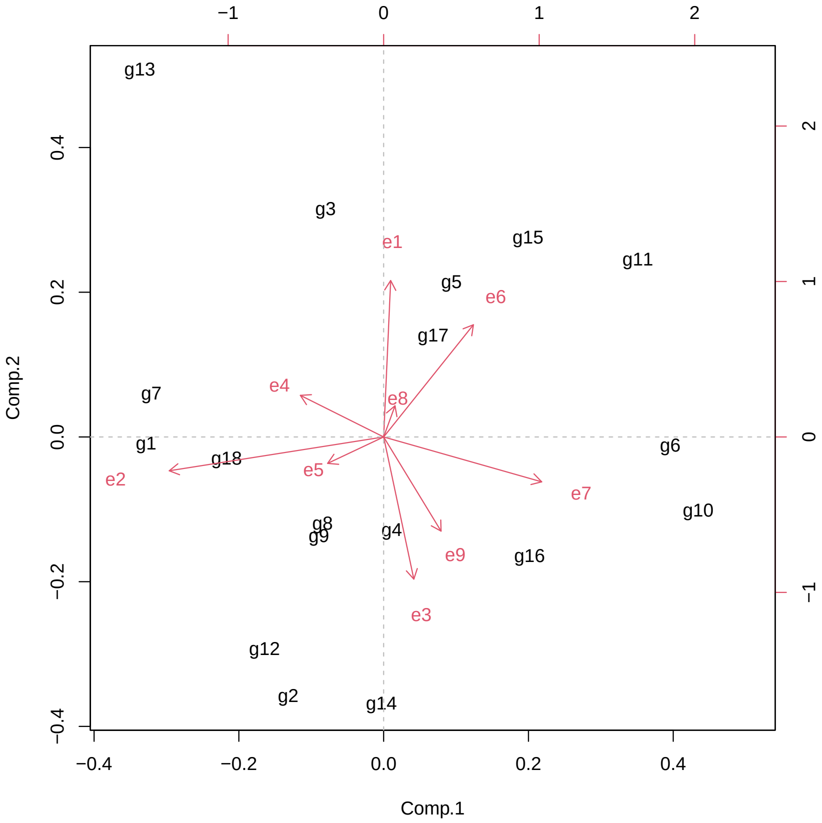


Figure 4.4.3: Biplot for the Irish VCU InnoVar data based on the results of the AMMI model. The x-axis (Comp. 1) and y-axis (Comp. 2) display the first and second principal components, respectively, obtained from an $I \times J$ matrix of residual values of a two-factor ANOVA model that considers the genotypes and environments as main effects.

in Figure 4.4.4 as the scale is centred around the mean, and when combined with the ordering, this gives a clear identification as to which environment and genotype produce high or low yields. Additionally, the AMBARTI methodology makes it possible to use tree methods to identify which combinations of genotypes and environments are similar considering a given phenotypic characteristic, e.g., yield. For example, the combination of genotype 1 and environment 1 and genotype 17 and environment 1 are similar in terms of predicted yield, as shown by the colours in Figure 4.4.5. Similarly, genotype 2 has similar yields for environments 4 and 5.

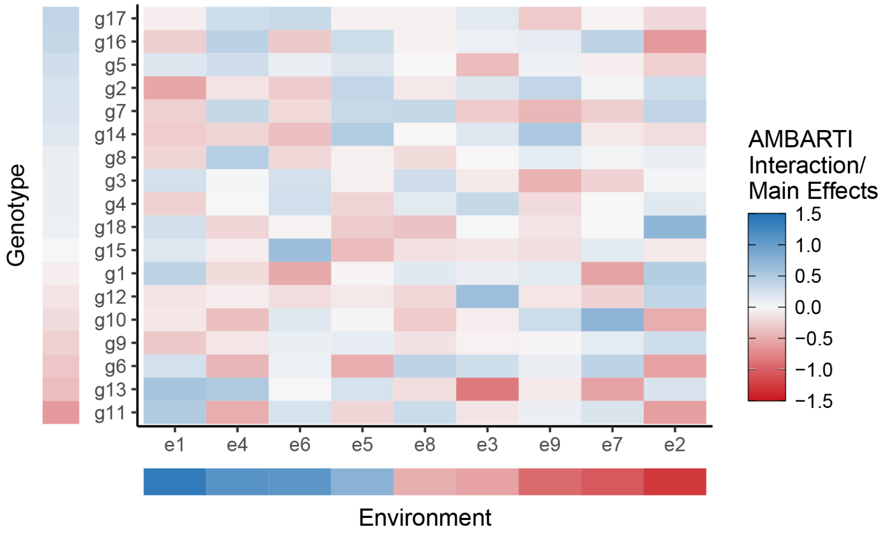


Figure 4.4.4: GxE interactions and main effects for the AMBARTI model sorted by the main effects for the Irish VCU InnoVar data in 2015. We can clearly see that environments 1, 4, 6, and 5 provide superior yields for many of the genotypes studied. Furthermore, environment 1, for example, seems to interact particularly strongly in a negative way with genotype 2. The grand mean μ is not included in this plot for ease of identification of marginal and interacting effects.

Thus, we can consider these environments similar and group them in a set named mega-environment 4-5 considering genotype 2. Such a concept is crucial for cultivar recommendations. To visualise the uncertainty associated with Figures 4.4.4 and 4.4.5, we provide plots which show the median, 5%, and 95% quantiles for the predicted response and GxE interactions and main effects in Appendix 4.D.

In Figure 4.4.6, we show a bipartite plot of the information displayed in Figure 4.4.4, but showing only the extremes of the high and low values. In this case, we display just the top 2% and the lowest 2% of the interactions. We employ the same diverging colour palette as Figure 4.4.4 except in this case the colour of the nodes represents the marginal effects and the size of each node represents the absolute value of the marginal effects. Similarly, the colour of the connecting edges represents the interaction values and the width of each edge represents the absolute interaction value. That is, larger magnitudes of the marginal effects will result in larger nodes (and vice-versa), and larger magnitudes of the interactions will result

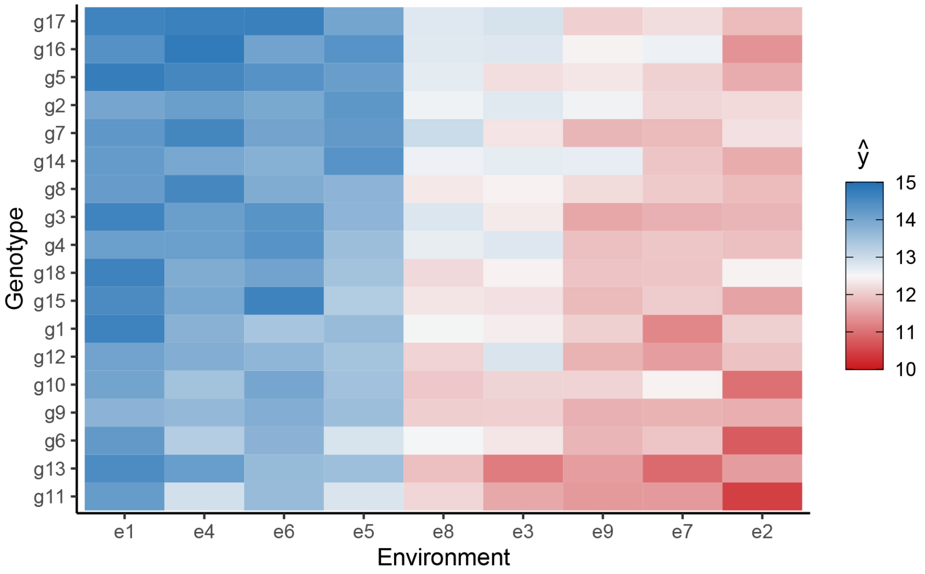


Figure 4.4.5: Predicted yields from the AMBARTI model for the Irish VCU InnoVar data in 2015. Values are sorted by the main effects. We can see, for example, a high value for the predicted yield for environment 1 with genotype 5 and a low value between environment 2 and genotype 11.

in thicker edges (and vice-versa). The aim of this plot is to allow the reader to easily and quickly identify which of the environments are the most and least optimal for each genotype and to also identify where there are clear interactions. Quantile versions of Figure 4.4.6 could also be plotted to assess uncertainty.

The visualisation perspective proposed here helps construct easily interpretable agronomic recommendations. Figure 4.4.5 can help users with no background in statistics identify that the best genotypes considering yield are the ones in the top left corner: g_{17}, g_{16}, g_5, g_2. These genotypes will have a tendency to have a better acceptance by farmers, considering solely the yield in tonnes per hectare assuming higher yields are economically preferred. Figure 4.4.4 shows us that environments e_1, e_4, e_6, e_5 are related to higher marginal effects and should be considered preferential to crop the list of wheat genotypes evaluated.

Figures 4.4.4 and 4.4.6 are also useful to establish combinations of genotypes and environments that should be avoided when the interaction is negative, indicating that a given genotype does not perform well in a given environment. This nega-

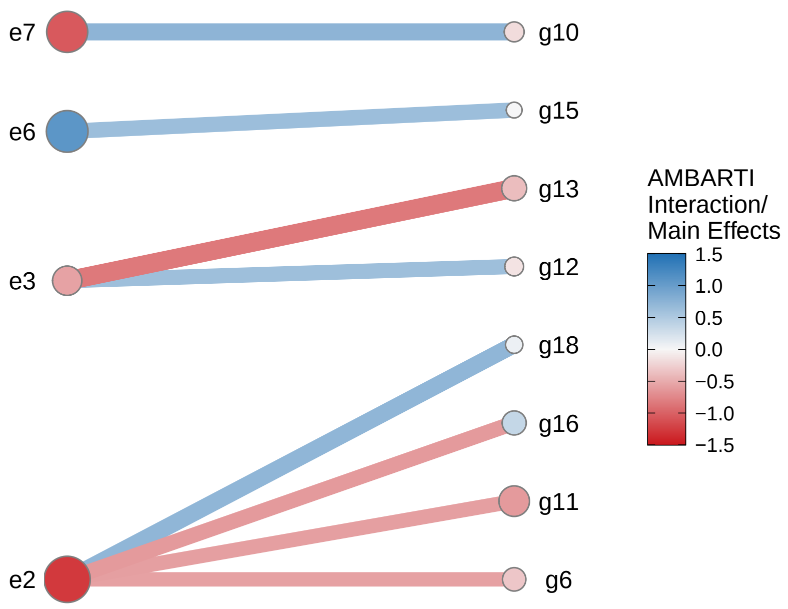


Figure 4.4.6: Bipartite network plot showing the top (in blue) and bottom (in red) 2% GxE interactions and main effects from the AMBARTI model for the Irish VCU InnoVar data in 2015. We can see that environment 3 has strong positive and negative interactions with genotypes 12 and 13, respectively.

tive interaction increases the risk of low yield and consequent economic impacts. Combinations to be avoided exist even for environments and genotypes with high marginal effects. For instance, the combination of g_2, e_1 should be avoided even though g_2 and e_1 have high marginal effects; see Figures 4.4.4 and 4.4.5. This is an important information for regulators who may be responsible for a variety's commercialisation approval or agents that promote credit or insurance for farmers given to risks that the negative interaction implies. Farmers who produce a genotype not indicated for their environment can end having a worse score or risk. On the other hand, Figure 4.4.5 is also useful to spot the combinations of genotypes and environments that should be encouraged once the signal of the interactions is positive.

In adaptability breeding, the breeder seeks to find the best genotype for a specific environment or a small set of environments. In broad target strategies, the aim is to find genotypes that perform well across several environments. For example, in Figure 4.4.4, g_5 has high marginal effect and performs well (and interacts positively) with environments e_1, e_4, e_6, e_9. Similarly g_{16}, the second best genotype considering marginal effects, performs well in environments e_4, e_5, e_9, e_7. Genotypes which present better performance across several environments are classified as high stability genotypes. They tend to be preferred by breeders because they allow optimisation of processes in the chain of seed production.

4.5 Discussion

We have introduced a new model named additive main effects Bayesian additive regression trees interaction (AMBARTI). AMBARTI is a cut Bayesian semi-parametric machine learning approach that estimates main effects of genotypes and environments and interactions with an adapted regression tree-like structure. This approach to interactions allows the treatment of more complex structures than the ones considered by traditional models.

Given the fact that GxE interactions are the result of a tangled myriad of genetics, proteomics, biochemical, environmental, and additional factors, the flexibility of AMBARTI in dealing with more complex interactions can be seen as an important improvement in the understanding of the complexities associated to GxE phenomenon. In practice, the choice between a low-rank model (AMMI) and a model that is able to deal with a sparse interaction structure (AMBARTI) is a modelling choice. However, in the real data examples upon we have tested the models, AMBARTI performed slightly or much better than AMMI, which suggests that a bilinear term is perhaps an oversimplification to study the relations that arise from GxE interactions. We believe that AMBARTI is a useful candidate to expand the understanding of experimental data in quantitative genetics.

The main novelty in AMBARTI comes from its semi-parametric structure which enables the uncertainty to be shared between the main effects and the interaction trees. More specifically, we design the trees so that they are forced to split on both a combination of genotypes and a combination of environments. We have shown in simulation experiments that this yields similar estimates to traditional models for the marginal effects and superior estimates for the interaction terms, which are no longer restricted to be linear in a restricted dimensional space. This removes the need for, e.g., the arbitrary selection of the Q parameter in a standard AMMI formulation.

A second novelty is that we have introduced new displays that simultaneously allow for interpretation of the marginal and joint effects. We have created both a heatmap and a bipartite network-style plot of the results, which are not limited for use with an AMBARTI model and could be applied to any suitable model (for example, AMMI, B-AMMI, etc.) and are a useful tool for deciphering complex model structures. From these plots, we hope to enable those using the output of AMBARTI models to make more informative decisions about which genotype and environments are most compatible.

We believe that there are many possible extensions of the AMBARTI approach. Other more advanced methods, such as PARAFAC (Basford et al., 1991; Harshman and Lundy, 1994), are available for higher dimensional interactions, such as with time. These are different versions of tensor regression (Guhaniyogi et al., 2017) and, in theory, there is no reason why the AMBARTI approach cannot be used for higher dimensional tensor-type interactions, though this is not currently possible in our code. Similar enhancements for multivariate outputs and time-series like-structure seem promising, and we hope to explore these in future work. Finally, modelling approaches which do not rely upon the BART model could also be explored. For instance, the spike-and-slab priors (George and McCulloch, 1997; Ishwaran and Rao, 2005) commonly used for Bayesian variable selection could be adapted to identify important GxE interactions. Based on some preliminary analyses, the results were promising and this a subject of ongoing work.

Appendix

4.A AMBARTI implementation

In this Section, we detail the AMBARTI model. Firstly, the conditional probability distribution associated with y_{ij} is

$$y_{ij}|\mathbf{x}_{ij}, \mu, g_i, e_j, \Theta \sim \mathrm{N}\left(\mu + g_i + e_j + \sum_{t=1}^{T} h(\mathbf{x}_{ij}, \mathcal{M}_t, \mathcal{T}_t), \sigma^2\right),$$

where y_{ij} denotes the response for genotype i and environment j, $\Theta = (\mathcal{M}_t, \mathcal{T}_t, \sigma^2)$, μ is the grand mean, $\mathbf{x}_{ij}$ is the row of the design matrix $\mathbf{X}$ associated to observations with genotype i and environment j, and $h(\cdot) = \mu_{t\ell}$ is a function that assigns the predicted values $\mu_{t\ell} \in \mathcal{M}_t$ to observations that belong to $\mathcal{P}_{t\ell}$, with $\mathcal{P}_{t\ell}$ denoting the set of rules that define the node ℓ of the tree t. In order to obtain the posterior distributions needed for the model, we assume the following prior distributions:

$$\begin{aligned}
\mu &\sim \mathrm{N}(m = 0, \sigma_m^2),\\
\mu_{t\ell}|\mathcal{T}_t &\sim \mathrm{N}(0, \sigma_\mu^2),\\
g_i|\mathcal{T}_t &\sim \mathrm{N}(\mu_g = 0, \sigma_g^2),\\
e_j|\mathcal{T}_t &\sim \mathrm{N}(\mu_e = 0, \sigma_e^2),\\
\sigma_g^2 &\sim \mathrm{IG}(a_g, b_g),\\
\sigma_e^2 &\sim \mathrm{IG}(a_e, b_e),\\
\sigma^2 &\sim \mathrm{IG}(\nu/2, \nu\lambda/2).
\end{aligned}$$

The prior distribution on the tree structure depends on the depth and number of terminal and internal nodes, and is given by

$$p(\mathcal{T}_t) = \prod_{\ell \in \mathcal{S}_I} \left[\alpha(1 + d_{t\ell})^{-\beta}\right] \times \prod_{\ell \in \mathcal{S}_T} \left[1 - \alpha(1 + d_{t\ell})^{-\beta}\right],$$

where $\mathcal{S}_I$ and $\mathcal{S}_T$ denote the sets of indices of the internal and terminal nodes, respectively, and $d_{t\ell}$ represents the depth of the node ℓ of the tree t. Furthermore, let $\mathbf{R}_t = \mathbf{y} - \left(\mu + \mathbf{g} + \mathbf{e} + \sum_{k \neq t}^{T} h(\mathbf{X}; \mathcal{T}_k, \mathcal{M}_k)\right)$ denote the vector of the partial residuals, where $\mathbf{g}$ and $\mathbf{e}$ are vectors containing the main effects g_i and e_j for all i and j. Below, we present the cut full conditional of μ.

$$\begin{aligned} p(\mu|-) &\propto p(\mathbf{y}|g_i, e_j, \mathbf{x}_{ij}, \sigma^2)p(\mu) \\ &\propto \exp\left(-\frac{1}{2\sigma^2_{m\star}}\left(\mu - \mu^{\star}\right)^2\right), \end{aligned}$$

which is a

$$\mathrm{N}\left(\frac{\sum_i \sum_j (y_{ij} - g_i - e_j)/\sigma^2 + m/\sigma^2_m}{n/\sigma^2 + 1/\sigma^2_m}, \frac{1}{n/\sigma^2 + 1/\sigma^2_m}\right).$$

Hence, the cut full conditional of g_i is given by

$$\begin{aligned} p(g_i|-) &\propto p(\mathbf{y}|g_i, e_j, \mathbf{x}_{ij}, \sigma^2)p(g_i) \\ &\propto \exp\left(-\frac{1}{2\sigma^2_{g\star}}\left(g_i - g_i^{\star}\right)^2\right), \end{aligned}$$

which is a

$$\mathrm{N}\left(\frac{\sum_j \left[y_{ij} - \mu - e_j\right]/\sigma^2}{n_{g_i}/\sigma^2 + 1/\sigma^2_g}, \frac{1}{n_{g_i}/\sigma^2 + 1/\sigma^2_g}\right), \tag{4.11}$$

where n_{g_i} is the number of observations that belong to g_i; similarly to n_{e_j}. Analogously, the cut full conditional of e_j can be written as

$$\begin{aligned} p(e_j|-) &\propto p(\mathbf{y}|g_i, e_j, \mathbf{x}_{ij}, \sigma^2)p(e_j) \\ &\propto \exp\left(-\frac{1}{2\sigma^2_{e\star}}\left(e_j - e_j^{\star}\right)^2\right), \end{aligned}$$

which is a

$$\mathrm{N}\left(\frac{\sum_i \left[y_{ij} - \mu - g_i\right]/\sigma^2}{n_{e_j}/\sigma^2 + 1/\sigma^2_e}, \frac{1}{n_{e_j}/\sigma^2 + 1/\sigma^2_e}\right). \tag{4.12}$$

The full conditional of σ^2_g is given by

$$p(\sigma^2_g|-) \propto p(\mathbf{g}|\sigma^2_g)p(\sigma^2_g),$$

which is an

$$\mathrm{IG}\left(\frac{I}{2}+a_g, \frac{\sum_{i=1}^{I} g_i^2}{2}+b_g\right).$$

The full conditional of σ_e^2 is written as

$$p(\sigma_e^2|-) \propto p(\mathbf{e}|\sigma_e^2)p(\sigma_e^2),$$

which is an

$$\mathrm{IG}\left(\frac{J}{2}+a_e, \frac{\sum_{j=1}^{J} e_j^2}{2}+b_e\right).$$

In addition, we present the full conditional of the trees. This distribution is used to compare the previous tree to the current one, as in BART the splitting rules are created by randomly selecting a covariate and a split point. Below, we present the full conditional of $\mathcal{T}_t$ as

$$\begin{aligned} p(\mathcal{T}_t|\mathbf{R}_t,\sigma^2) &\propto p(\mathcal{T}_t)\int p(\mathbf{R}_t|\mathcal{M}_t,\mathcal{T}_t,\sigma^2)p(\mathcal{M}_t|\mathcal{T}_t)d\mathcal{M}_t \qquad (4.13)\\ &\propto p(\mathcal{T}_t)p(\mathbf{R}_t|\mathcal{T}_t,\sigma^2)\\ &\propto p(\mathcal{T}_t)\prod_{\ell=1}^{b_t}\left[\left(\frac{\sigma^2}{\sigma_\mu^2 n_{t\ell}+\sigma^2}\right)^{1/2}\exp\left(\frac{\sigma_\mu^2\left[n_{t\ell}\bar{R}_\ell\right]^2}{2\sigma^2(\sigma_\mu^2 n_{t\ell}+\sigma^2)}\right)\right], \end{aligned}$$

where $\bar{R}_\ell = \sum_{(i,j)\in\mathcal{P}_{t\ell}}(r_{ij}^{(t)}-\mu-g_i-e_j)/n_{t\ell}$, $r_{ij}^{(t)} \in \mathbf{R}_t$ and $n_{t\ell}$ is the number of observations that belong to $\mathcal{P}_{t\ell}$. To sample from this expression, the Metropolis-Hastings algorithm is used because a closed-form distribution is not obtained in this case.

As all $\mu_{t\ell}$ are i.i.d, it is possible to write $p(\mathcal{M}_t|\mathcal{T}_t,\mathbf{R}_t,\sigma^2) = \prod_{\ell=1}^{b_t} p(\mu_{t\ell}|\mathcal{T}_t,\mathbf{R}_t,\sigma^2)$. Similarly to the original BART, the full conditional of $\mu_{t\ell}$ in the AMBARTI model also depends only on the information provided by all trees, except by $\mathcal{T}_t$, via partial residual as $\mathbf{R}_t$. Hence, the full conditional of $\mu_{t\ell}$ can be written as

$$\begin{aligned} p(\mu_{t\ell}|-) &\propto p(\mathbf{R}_t|\mathcal{M}_t,\mathcal{T}_t,\sigma^2)p(\mu_{t\ell})\\ &\propto \exp\left(-\frac{1}{2\sigma_\star^2}\left(\mu_{t\ell}-\mu_{t\ell}^\star\right)^2\right), \end{aligned}$$

which is a

$$\mathrm{N}\left(\frac{\sigma^{-2}\sum_{(i,j)\in\mathcal{P}_{t\ell}} r_{ij}^{(t)}}{n_{t\ell}/\sigma^2+\sigma_\mu^{-2}}, \frac{1}{n_{t\ell}/\sigma^2+\sigma_\mu^{-2}}\right).$$

Finally, after generating all predicted values for all trees, σ^2 can be updated based on

$$\begin{aligned} p(\sigma^2|-) &\propto p(\mathbf{y}|\mathbf{g},\mathbf{e},\mathbf{X},\mathcal{M}_t,\mathcal{T}_t,\sigma^2)p(\sigma^2) \\ &\propto (\sigma^2)^{-\left(\frac{n+\nu}{2}+1\right)}\exp\left(-\frac{S+\nu\lambda}{2\sigma^2}\right), \end{aligned} \tag{4.14}$$

where $S=\sum_{i=1}^{I}\sum_{j=1}^{J}(y_{ij}-\hat{y}_{ij})^2$ and $\hat{y}_{ij}=\mu+g_i+e_j+\sum_{t=1}^{T}h(\mathbf{x}_{ij};\mathcal{T}_t,\mathcal{M}_t)$. The expression in (4.14) is an $\mathrm{IG}((n+\nu)/2,(S+\nu\lambda)/2)$. In algorithm 3, the full structure of the AMBARTI model is presented.

Algorithm 3 AMBARTI model

```
Input: $\mathbf{y}$, $\mathbf{X}$, number of trees $T$, and number of MCMC iterations $M$.
Initialise: $\{\mathcal{T}_t\}_1^T$ and set all hyperparameters of the prior distributions.
for ($m = 1$ to $M$) do
    Update the parameters $\mu$, $g_i$ and $e_j$.
    Update the variances $\sigma_g^2$ and $\sigma_e^2$.
    for ($t = 1$ to $T$) do
        Compute $\mathbf{R}_t = \mathbf{y} - \mathbf{g} - \mathbf{e} - \sum_{j\neq t}^{T} g(\mathbf{X},\mathcal{M}_j,\mathcal{T}_j)$.
        Propose a new tree $\mathcal{T}_t^\star$ by a grow, double grow, prune, double prune,
        change, or swap move.
        Accept the proposed tree with probability
            $\alpha(\mathcal{T}_t,\mathcal{T}_t^\star) = \min\left\{1, \frac{p(\mathcal{T}_t^\star \mid \mathbf{R}_t,\sigma^2)q(\mathcal{T}_t^\star\to\mathcal{T}_t)}{p(\mathcal{T}_t \mid \mathbf{R}_t,\sigma^2)q(\mathcal{T}_t\to\mathcal{T}_t^\star)}\right\}$.
        Sample $u \sim \mathrm{Uniform}(0,1)$: if $\alpha(\mathcal{T}_t,\mathcal{T}_t^\star) < u$, set $\mathcal{T}_t = \mathcal{T}_t$, otherwise
        set $\mathcal{T}_t = \mathcal{T}_t^\star$.
        Update all node-level parameters $\mu_{t\ell}$ for $\ell = 1,\dots,b_t$.
    end for
    Update $\sigma^2$.
    Update the predicted response $\hat{\mathbf{y}}$.
end for
Output: samples of the posterior distribution of $\mathcal{T}$.
```

4.B Naive and tempered cut algorithms

We performed a simulation experiment to compare the results from the naive and tempered cut algorithms. We did this because the results in the paper are all based on the former method. The rationale of the tempered transitions, which are obtained from the tempered cut algorithm, is instead of moving from, say, $\sigma^2 \to \sigma^{2\prime}$, we move along a linear path in a sequence $\sigma^2(c) = c\sigma^2 + (1-c)\sigma^{2\prime}$, where $c = (c_1, \dots, c_m)$ with $c_k = k/m$. Then, at each step, conditioned on $\sigma^2(c)$, a sample for $\mathcal{T}$ is obtained but only the last one is kept, which becomes $\mathcal{T}'$ (Plummer, 2015). In this way, the condition in (4.9) is satisfied and approximate samples from the cut posterior distribution of interest are obtained. Since $p\left(\mathcal{T}_t|\mathbf{R}_t, \sigma^2\right)$ in AMBARTI does not have a known distributional form and due to the use of a cut model to avoid the feedback between the main effects and the BART component, Plummer (2015) recommend performing tempered transitions for g_i, e_j and σ^2.

We have carried out this comparison to show whether the posterior samples from $p\left(\mathcal{T}_t|\mathbf{R}_t, \sigma^2\right)$ differ under each method. Recall that in AMBARTI the following parameters have closed-form full conditionals: g_i (effect of genotype), e_j (effect of environment), $\mu_{t\ell}$ (BART predictions), σ^2 (residual variance). Thus, to sample from $p\left(\mathcal{T}_t|\mathbf{R}_t, \sigma^2\right)$ via the tempered cut algorithm, the transitions from $g_i \to g_i'$, $e_j \to e_j'$ and $\sigma^2 \to \sigma^{2\prime}$ need to be smooth and cannot lead to large jumps. We point out that there is no need to adjust the transitions from $\mu_{t\ell} \to \mu_{t\ell}'$ since these are marginalised out in $p\left(\mathcal{T}_t|\mathbf{R}_t, \sigma^2\right)$ as shown in (4.13).

Unlike the other full conditionals in the AMBARTI model, the full conditional for the trees is used as a mechanism to filter ‘good’ splitting rules only, as opposed to returning a value from the corresponding posterior distribution. Thus, to examine the convergence of the trees is not a straightforward task because i) the splitting rules are randomly selected and, as a consequence, ii) there is not a conventional MCMC chain of posterior samples to be examined. To illustrate this point, suppose that we ran AMBARTI more than once and then examined the structure of, say, $\mathcal{T}_2$. The structure/splitting rules of $\mathcal{T}_2$ certainly will not be the same across the runs, especially if the number of covariates is large. This is not necessarily an issue and happens frequently due to the uniform specification on the splitting rules where independent uniform priors are placed on the available covariates and split values. Conversely, the convergence of the posterior samples of g_i, e_j and σ^2 can be assessed by visualising the posterior density of these parameters along the line of Figure 4 of Plummer (2015).

Figure 4.2.1 shows some elements of the tree structure that could be used in an attempt to evaluate the convergence of the trees. Once again we point out that the predicted values $\mu_{t\ell}$ are integrated out from $p\left(\mathcal{T}_t|\mathbf{R}_t, \sigma^2\right)$. The remaining quantities, which are derived from the tree structure rather than parameters directly monitored during the MCMC run, are the depth of the tree, the number of terminal nodes (circles), the number of internal nodes (rectangles) and the splitting rules, which are represented into the internal nodes.

The maximum depth of a binary tree is directly related to the number of nodes in the tree. In addition, the number of internal nodes in the same type of tree is equal to number of terminal nodes minus 1. Thus, in an attempt to evaluate the convergence of the trees, it seems reasonable to look at the distribution of the number of terminal nodes, based on the aforementioned relations. More specifically, we are interested in empirically evaluating whether the distribution of the number of terminal nodes changes as the number of steps in the tempered transition increases for the tempered cut algorithm.

With this in mind, we simulated synthetic data from the AMMI model as it is the least favourable scenario for the AMBARTI model due to the orthonormality constraints on the bilinear term. The data were simulated considering $I = 10$ genotypes, $J = 10$ environments, $Q = 3$, $\lambda = (12, 10, 8)$, and standard errors $\sigma^2 = \sigma_g^2 = \sigma_e^2 = 1$. Furthermore, we have analysed the distribution of the number of terminal nodes for our case study Innovar data set for 2015. For both synthetic and real data, we used an increasing number of steps ($m = 1, 2, 4, 10, 20,$ and 40). We have considered steps greater than 40, but the results were not any different compared to 40.

Figure 4.B.1 shows the results of the mean number of terminal nodes for the different number of steps m in the tempered cut algorithm. In panel (a), the results are for the simulated data, while in panel (b) they correspond to the Innovar data.

As we can see, the distribution of the number of terminal nodes in both panels do not differ between $m = 1$ (naive cut algorithm) and the other number of steps (tempered cut algorithm). For instance, in panel (a), we can see that the boxplots for $m = 1, 20$ and 40 are similar in terms of the first, second and third quartiles, which highlights that the tempered transitions have no effect on the topology of the trees in AMBARTI for these data. In panel (b), the difference between the results of $m = 1$ and the others number of steps is negligible. Finally, we have also noticed that the tempered transitions (with steps greater than 1) have not changed either the estimates of the additive effects, the estimates of the interactions, the overall fitted values, or even the posterior prediction intervals.

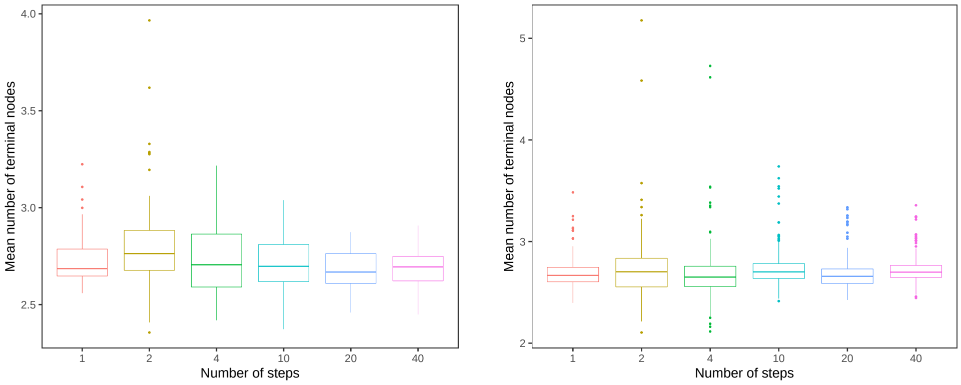


Figure 4.B.1: Mean number of terminal nodes in the AMBARTI model with tempered transitions. The numbers of steps used in the tempered cut algorithm are displayed on the x-axis. The y-axis shows the mean number of terminal nodes observed after the burn-in period. Each boxplot contains 200 values corresponding to the mean number of terminal nodes in each tree used in the ensemble of the BART component. In panel (a), the results are shown for the synthetic data simulated from the AMMI equation. In panel (b), the results are for the Innovar data for 2015.

Given the similarity of the distributions of the number of terminal nodes under the naive and tempered cut algorithms for the data considered above, we kept the results from the naive cut algorithm in the simulation and case study sections. In practice, the results shown in this Appendix provide elements that indicate that the condition in (4.9), after an appropriate burn-in period, seems reasonably attained even under the naive cut algorithm for the AMBARTI model.

We believe the explanation for the similar results is two-fold. First, the transition $\mathcal{T}_t \rightarrow \mathcal{T}_t'$ for 50% of the moves (i.e., change and swap) in the BART component have no dependence on the previous tree $\mathcal{T}_t$, since the ratio of the transition kernel for these moves is 1. Second, the prior on the trees forces them to be shallow and this is reflected in the low mean number of terminal nodes per tree. Unlike the cervical cancer example of Plummer (2015) where the Metropolis algorithm is set up to give acceptance probabilities in a certain range for the (cut) parameter of interest, in AMBARTI it is not possible to control the acceptance probability rate of the Metropolis step because the splitting rules are randomly proposed.

Although the tempered/naive cut algorithms produced similar results, we point out that the number of steps in the tempered transitions needs to be assessed for the application at hand. In our software, we have added an argument called `nsteps` which allows the user to manually specify the number of steps in the tempered transitions, with the default option being `nsteps=1`.

4.C Orthonormality constraints of the AMMI model

We recall that the AMMI model is overparameterised, so constraints need to be imposed so that the parameters can be estimated (Josse et al., 2014). In this Section, we show how to apply the orthonormality constraints on γ_{iq} and δ_{jq} when simulating from the AMMI model.

Let $\boldsymbol{\gamma}$ be an $I \times Q$ matrix, $\boldsymbol{\delta}$ a $J \times Q$ matrix, and consider that γ_{iq} and δ_{jq} are elements in row i and column q of the corresponding matrices. The following constraints are considered: i) $\sum_{i=1}^{I} \gamma_{iq} = \sum_{j=1}^{J} \delta_{jq} = 0$, for $q = 1, \dots, Q$; ii) $\boldsymbol{\gamma}^\top \boldsymbol{\gamma} = \boldsymbol{\delta}^\top \boldsymbol{\delta} = \mathbb{I}_q$, where $\mathbb{I}_q$ represents an identity matrix of dimension q; iii) $\lambda_1 \geq \lambda_2 \geq \cdots \lambda_{q-1} \geq \lambda_q \geq 0$ and; iv) $\gamma_{1q} \geq 0$, for all $q = 1, \dots, Q$.

To illustrate our strategy to meet the constraints presented above, we take the γ_{iq} as an example, but this also works for δ_{jq}. First, we create S an $I \times Q$ matrix, where $s_{iq} \sim \mathrm{N}(0, \sigma_x^2 = 1)$. Here, s_{iq} could be sampled from other distributions, such as Gamma or Beta. In addition, we define $\bar{S}$ as an $I \times Q$ matrix with each element being the mean of the corresponding q column of S. Hence, we know that

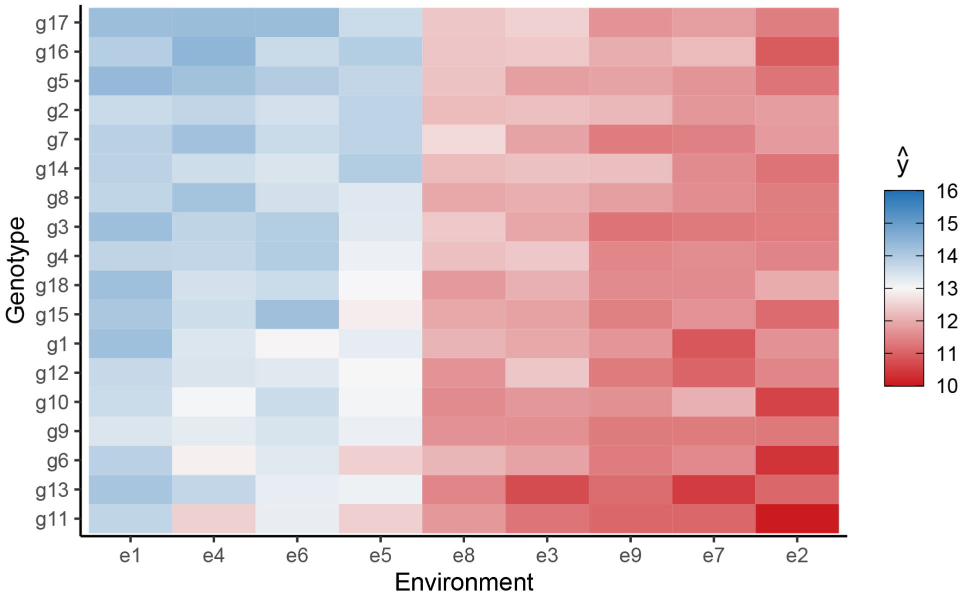
$$\begin{aligned} B &= S - \bar{S} \\ &\Rightarrow \mathbf{1}_I^\top B = 0 \\ &\not\Rightarrow B^\top B = \mathbb{I}, \end{aligned}$$

where $\mathbf{1}_I$ is a column vector of dimension I containing ones, B is, by construction, a full rank matrix and $B^\top B$ is symmetric. However, we find a matrix A such that $C = BA \Rightarrow C^\top C = \mathbb{I}$. That is, we know that

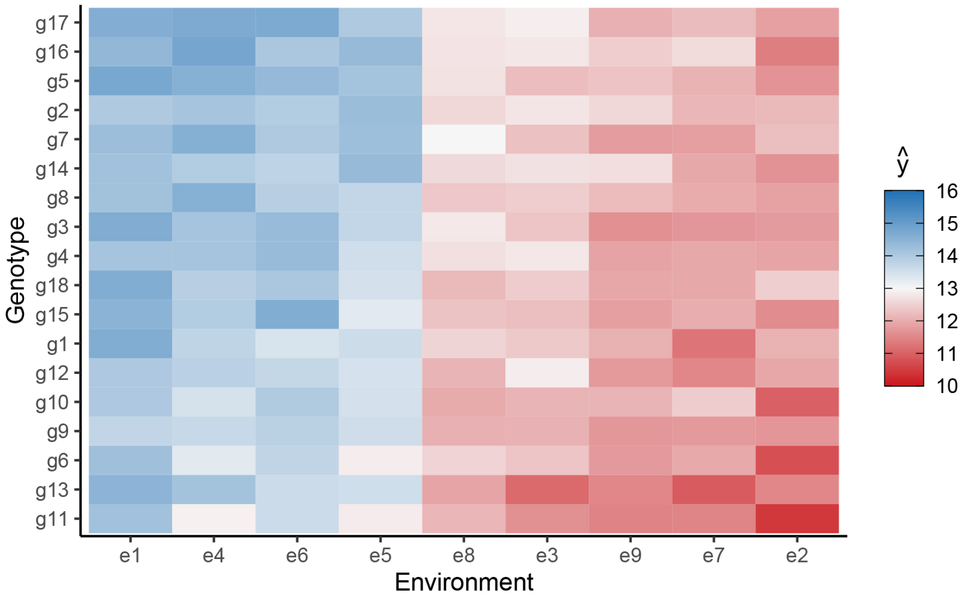
$$\begin{aligned} D &= C^\top C = \mathbb{I} \\ &\Rightarrow (BA)^\top BA = \mathbb{I} \\ &\Rightarrow A^\top B^\top BA = \mathbb{I} \\ &\Rightarrow B^\top B = A^{-\top} A^{-1} \\ &\Rightarrow B^\top B = (AA^\top)^{-1} \\ &\Rightarrow (B^\top B)^{-1} = AA^\top \\ &\Rightarrow (B^\top B)^{-1} = A^2 \text{ (by symmetry)} \\ &\Rightarrow (B^\top B)^{-1/2} = A. \end{aligned}$$

In the end, we have that $\boldsymbol{\gamma} = B(B^\top B)^{-1/2}$.

4.D Visualising the uncertainties of the parameter estimates in the AMBARTI model



(a) 5% quantile



(b) median

Figure 4.D.1: Median and 5% quantiles for predicted yields from the AMBARTI model for the Irish VCU InnoVar data in 2015. The two parts of the graph allow us to address the uncertainties associated with the predicted response in yields described in Figure 4.4.5.

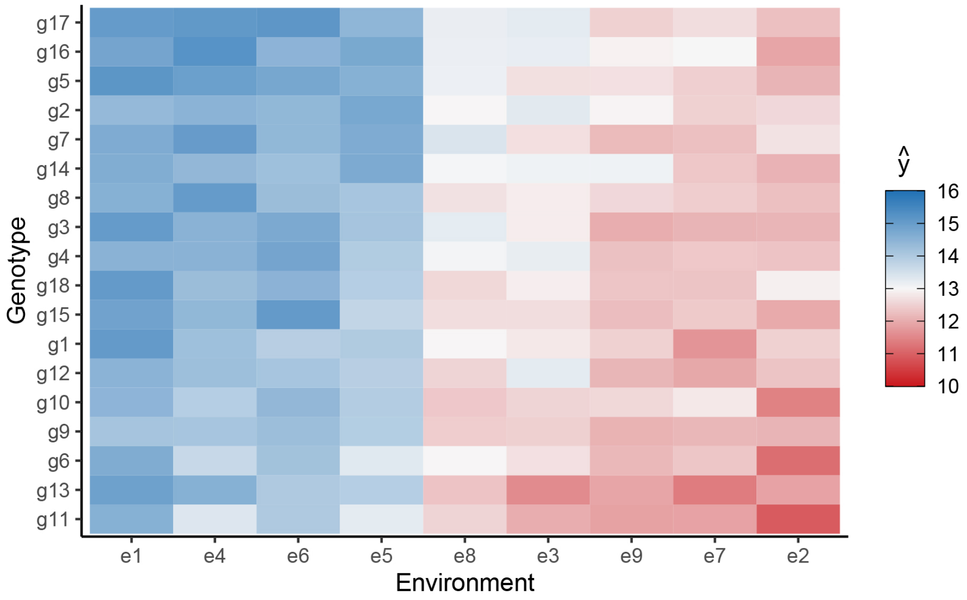
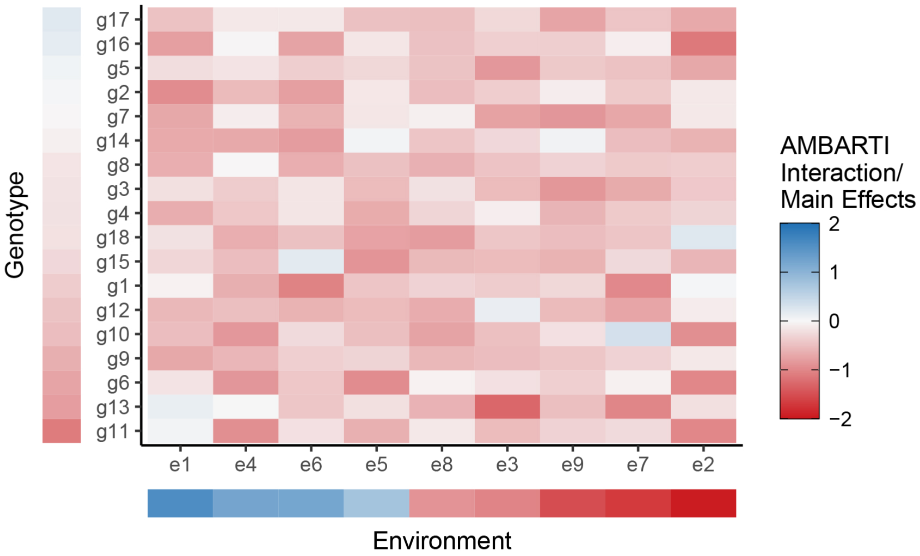
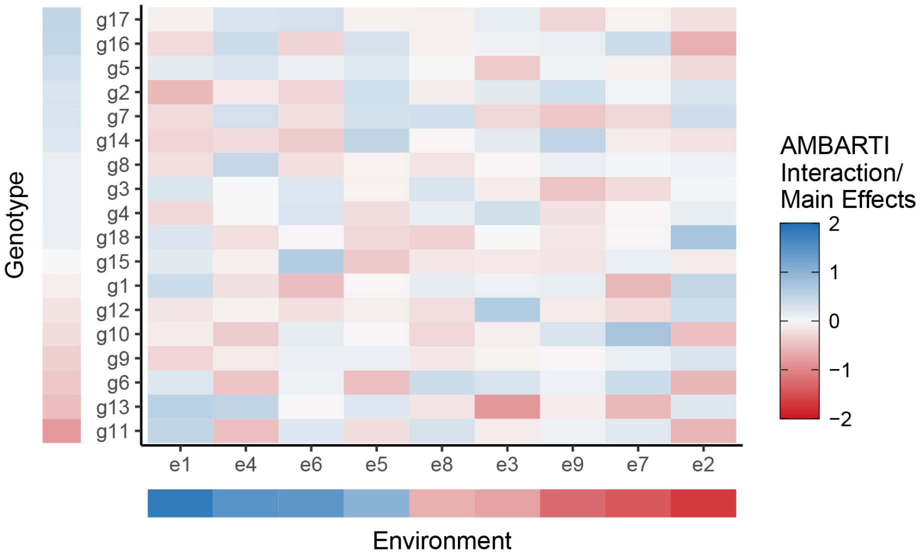


Figure 4.D.2: 95% quantile for predicted yields from the AMBARTI model for the Irish VCU InnoVar data in 2015. The graph allow us to address the uncertainties associated with the predicted response in yields described in Figure 4.4.5.



(a) 5% quantile



(b) median

Figure 4.D.3: Median and 5% quantile for predicted yields from the AMBARTI model for the Irish VCU InnoVar data in 2015. The three parts of the graph allow us to address the uncertainties associated with the GxE interactions and the main effects obtained by AMBARTI in Figure 4.4.4.

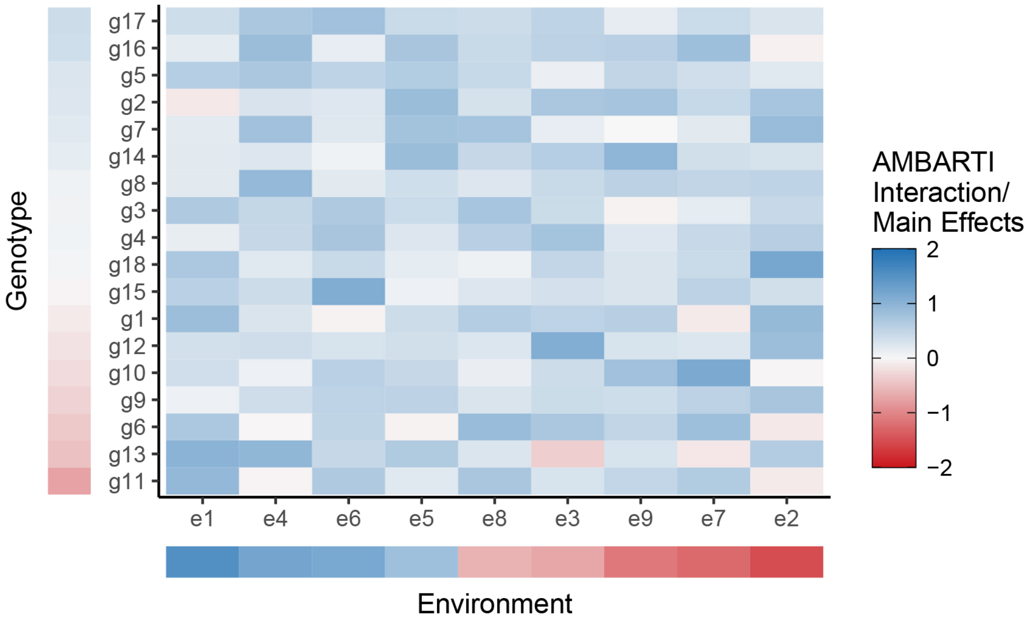


Figure 4.D.4: 95% quantile for predicted yields from the AMBARTI model for the Irish VCU InnoVar data in 2015.

CHAPTER 5

Accounting for shared covariates in semi-parametric Bayesian additive regression trees

We propose some extensions to semi-parametric models based on Bayesian additive regression trees (BART). In the semi-parametric BART paradigm, the response variable is approximated by a linear predictor and a BART model, where the linear component is responsible for estimating the main effects and BART accounts for non-specified interactions and non-linearities. Previous semi-parametric models based on BART have assumed that the set of covariates in the linear predictor and the BART model are mutually exclusive in an attempt to avoid poor coverage properties and reduce bias in the estimates of the parameters in the linear predictor. The main novelty in our approach lies in the way we change the tree-generation moves in BART to deal with this bias and resolve non-identifiability issues between the parametric and non-parametric components, even when they have covariates in common. This allows us to model complex interactions involving the covariates of primary interest, both among themselves and with those in the BART component. Our novel method is developed with a view to analysing data from an international education assessment, where certain predictors of students' achievements in mathematics are of particular interpretational interest. Through additional simulation studies and another application to a well-known benchmark dataset, we also show *competitive performance when compared to regression models, alternative formulations of semi-parametric BART, and other tree-based methods. The implementation of the proposed method is available at* `https://github.com/ebprado/CSP-BART`*.*

5.1 Introduction

Generalised linear models (GLMs; Nelder and Wedderburn, 1972; McCullagh and Nelder, 1989) are frequently used in many different applications to predict a univariate response due to the ease of interpretation of the parameter estimates as well as the wide availability of statistical software that facilitates simple analyses. A key assumption in GLMs is that the specified covariates in the linear predictor, including potential interactions and higher-order terms, have a linear relationship with the expected value of the response variable through a defined link function.

Extensions such as generalised additive models (GAMs; Hastie and Tibshirani, 1990; Wood, 2017) require the specification of the main and interaction effects via a sum of (potentially non-linear) predictors. In GAMs, the non-linear relationships are usually captured via basis expansions of the covariates and constrained by a smoothing parameter. However, in problems where the numbers of covariates and/or observations are large, it may not be simple to specify the covariates and the interactions that impact most on the response. Semi-parametric models (Harezlak et al., 2018) have been proposed for situations where a mixture of linear and non-linear trends, as well as interactions, are required for accurately fitting the data at hand.

Semi-parametric Bayesian additive regression tree (BART) models (Chipman et al., 2010; Zeldow et al., 2019; Tan and Roy, 2019; Hahn et al., 2020; Deshpande et al., 2020) are black-box type algorithms which aim to tackle some of the key limitations often encountered when using GLMs to analyse datasets with a large number of covariates. Most commonly, they are used when there is interest in quantifying the relationships between covariates and the response. It is well-known that tree-based algorithms, such as BART and random forests (Breiman, 2001), are flexible and can produce more accurate predictions, as they remove the often restrictive assumption of linearity between the covariates and the response. However, prediction is not the most important aspect in many situations (e.g., Hill, 2011; Zeldow et al., 2019; Hahn et al., 2020). Instead, knowing how covariates impact the response is crucial; but this quantification is not easily interpretable with the standard BART model or random forests. Thus, the main appeal of semi-parametric BART models is that they allow us to look inside the black-box and provide interpretations for how some key inputs of primary interest are converted into outputs.

Besides the estimation of main effects, another common use of regression models is to measure the effects that combinations of covariates may have on the response. However, most standard GLM settings require pre-specification of interaction terms, which is a complicated task with high-dimensional data. As semi-parametric BART models account for non-specified interactions automatically, they would appear to be an ideal solution to this problem.

Motivated by data collected in 2019 under the seventh cycle of the quadrennial Trends in International Mathematics and Science Study (TIMSS; Mullis et al., 2020; Fishbein et al., 2021), we extend the semi-parametric BART model introduced by Zeldow et al. (2019), which we henceforth refer to as separated semi-parametric BART (SSP-BART) for clarity. TIMSS is an international assessment which evaluates students' performance in mathematics and science at different grade levels across several countries. A large number of features pertaining to students, teachers, and schools are recorded. We aim to quantify the impact of a small number covariates of primary interpretational interest (i.e., parents' education level, minutes spent on homework, and school discipline problems) on students' performance in mathematics, in the presence of other covariates of non-primary interest.

In the previously proposed SSP-BART, the design matrix is split into two disjoint subsets $\mathbf{X}_1$ and $\mathbf{X}_2$, which contain covariates of primary and non-primary interest, respectively. The specification of these matrices should be guided by the application at hand. The covariates in $\mathbf{X}_1$ are of interest in terms of being interpretable, but their impact on the response is also relevant. The covariates of non-primary interest in $\mathbf{X}_2$ may still be strongly related to the response, but are not considered important in terms of interpretation. The primary covariates in $\mathbf{X}_1$ are specified in a linear predictor and the others are exclusively used by BART; i.e., covariates in $\mathbf{X}_2$ are the only ones allowed to form interactions. SSP-BART applied to the TIMSS data would thus prohibit interactions between (or involving) the aforementioned primary covariates. This omission of important interactions represents a major limitation of SSP-BART, given that handling interactions automatically is supposedly part of its appeal.

Our work differs from SSP-BART in that i) we do not assume that $\mathbf{X}_1$ and $\mathbf{X}_2$ are disjoint; i.e., we allow $\{\mathbf{X}_1 \cap \mathbf{X}_2\} \neq \emptyset$, or even $\mathbf{X}_1 \subset \mathbf{X}_2$. This is important because primary and non-primary covariates may also interact in complex ways and further impact the response. Unlike SSP-BART, the BART component of our model accounts for this, which yields better trees and notably improved predictive performance on the TIMSS data. Moreover, ii) we change the way the trees in BART are grown by introducing 'double grow' and 'double prune' moves, along with stricter checks on tree-structure validity, to resolve non-identifiability issues between the parametric (linear) and non-parametric (BART) components. Finally, iii) while Zeldow et al. (2019) assume that all parameters in the linear predictor have the same (diffuse) variance *a priori*, we instead place a hyperprior on the full hyper-covariance matrix of the main effects, so that we are better able to model the correlations among them.

Thus, within the semi-parametric BART paradigm, we make a distinction between SSP-BART and our combined semi-parametric BART, which we call CSP-BART. In CSP-BART, we have made fundamental structural changes to the way that the trees are grown due to the fact that $\mathbf{X}_1$ and $\mathbf{X}_2$ can have covariates in common. Specifically, we prohibit the BART component from estimating marginal effects for variables in $\mathbf{X}_1$ in order to ensure that the parameter estimates in the linear component are identifiable. We also allow the specification of both fixed and random effects in the linear predictor, as in a linear mixed model, in which the parameter estimates can vary by a grouping factor. In contrast, interactions and non-linearities are handled by the BART component.

Beyond our proposed extensions to SSP-BART, another related work in this area is the varying coefficient BART (VCBART; Deshpande et al., 2020), which combines the idea of varying-coefficient models (Hastie and Tibshirani, 1993) with BART and extends the work of Hahn et al. (2020) to a framework with multiple covariates. In VCBART, the response is modelled via a linear predictor where the effect of each covariate is approximated by a BART model based on a set of modifiers (i.e., covariates that are not of primary interest). The only similarity between VCBART and CSP-BART is the use of a linear predictor along with BART. However, our work is structurally different as we do not estimate the parameters in the linear predictor via BART. Instead, they are obtained in the same fashion as a Bayesian linear mixed model approach, so as to yield interpretable and unbiased coefficient estimates.

We show using standard performance metrics that VCBART, SSP-BART, and GAMs compare unfavourably to CSP-BART in two simulation studies, our analysis of the TIMSS data, and another application setting. In the simulation experiments, we compare CSP-BART with its main competitors and show its ability to recover the true effects in either the presence or absence of interactions. The additional application to the well-known Pima Indians Diabetes dataset (Blake, 1998) is presented to demonstrate the practical use of CSP-BART in classification rather than regression settings. Here, the goal is to determine whether or not a patient has diabetes based on eight covariates, with special interest in studying the effects of age and glucose through a linear predictor along with possible non-specified complex interactions involving age, glucose, and/or the other six covariates accounted for by BART.

The remainder of this paper is organised as follows. In Section 5.2, we summarise the BART model and introduce relevant notation. In Section 5.3, we revise the separated semi-parametric BART model and describe in detail our proposed extensions to CSP-BART. In Section 5.4, we compare the performance of CSP-BART with other relevant algorithms on synthetic data. We analyse the TIMSS dataset and explore the additional real-world application in Section 5.5. To conclude, we present a discussion in Section 5.6.

5.2 BART

BART (Chipman et al., 2010) is a Bayesian statistical model based on an ensemble of trees that was first proposed in the context of regression and classification problems. Through an iterative Bayesian backfitting MCMC algorithm, BART sequentially generates a set of trees that, when summed together, return predicted values. A branching process prior is placed on the tree structure to control the depth of the trees. In addition, the covariates and split-points used to define the tree structure (i.e., splitting rules) are randomly selected without the optimisation of a loss function, such as in random forests (Breiman, 2001) and gradient boosting (Friedman, 2001). Compared to regression models, BART is more flexible in the sense that it does not assume linearity between the covariates and the response and does not require the specification of a linear predictor. In particular, BART automatically determines non-linear main effects and multi-way interaction effects.

BART has been used and extended to different applications, and its theoretical properties have also gained attention more recently. For instance, BART has been applied to credit risk modelling (Zhang and Härdle, 2010), survival/competing analysis (Sparapani et al., 2016, 2019; Linero et al., 2021), biomarker discovery (Hernández et al., 2015), plant-based genetics (Sarti et al., 2023), and causal inference (Hill, 2011; Green and Kern, 2012; Hahn et al., 2020). Furthermore, it has also been extended to high-dimensional data (Linero, 2018; Hernández et al., 2018), polychotomous responses (Kindo et al., 2016b; Murray, 2021), zero-inflated and semi-continuous data (Murray, 2021; Linero et al., 2020), heteroscedastic data (Pratola et al., 2020), and to estimate linear, smooth, and monotone functions (Starling et al., 2020; Prado et al., 2021; Chipman et al., 2021). Regarding theoretical developments, we highlight the works of Ročková and van der Pas (2020), Ročková and Saha (2019), and Linero and Yang (2018), who provide results related to the convergence of the posterior distribution generated by the BART model. Finally, we note that BART has also been previously employed in education assessment settings (Suk et al., 2021), similar to the TIMSS application we analyse herein.

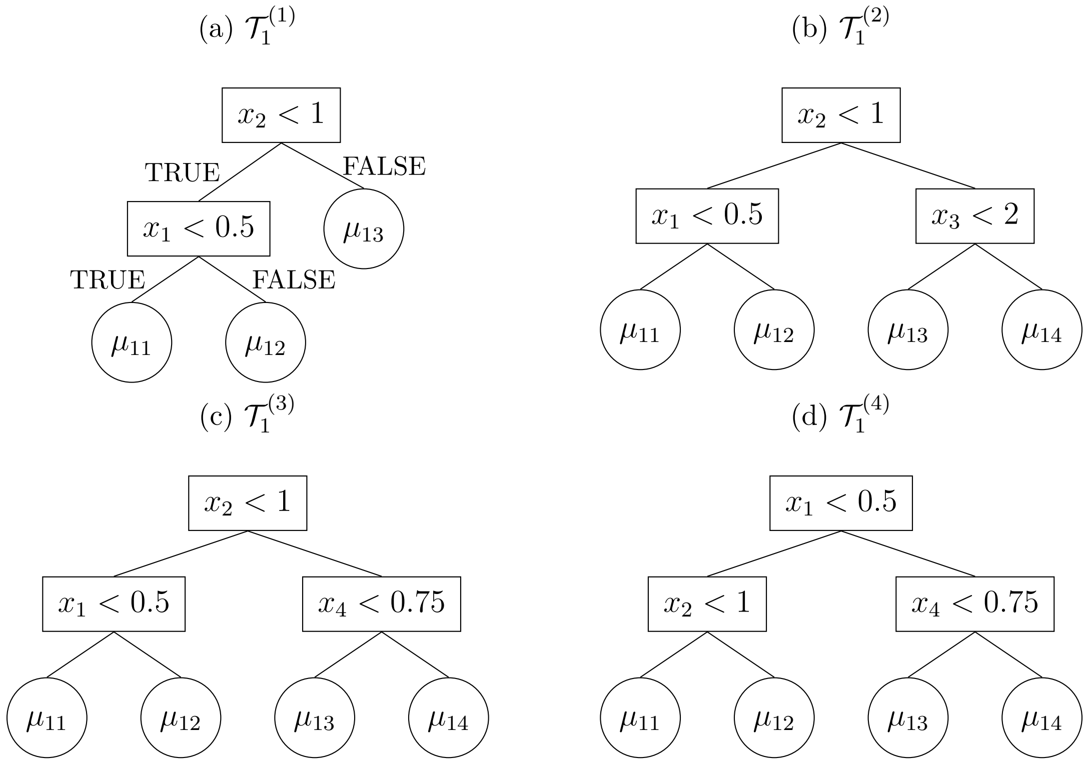
In the standard BART model, a univariate response $\{y_i\}_{i=1}^{n}$ is approximated by a sum of trees, with

$$y_i \mid \mathbf{x}_i, \mathcal{M}, \mathcal{T}, \sigma^2 \sim \mathrm{N}\left(\sum_{t=1}^{T} g\left(\mathbf{x}_i, \mathcal{M}_t, \mathcal{T}_t\right), \sigma^2\right),$$

where $\mathrm{N}(\cdot)$ denotes the Normal distribution, σ^2 is the error variance, $g(\cdot) = \mu_{t\ell}$ is a function which assigns predicted values $\mu_{t\ell}$ to all observations falling into terminal node ℓ of tree t, $\mathbf{x}_i$ denotes the i-th row of the design matrix $\mathbf{X}$, $\mathcal{T}_t$ represents the topology of tree t, and $\mathcal{M}_t = (\mu_{t1}, \ldots, \mu_{tb_t})$ is a vector comprising the predicted values from the b_t terminal nodes of tree t. For notational convenience, we let $\mathcal{T} = (\mathcal{T}_1, \ldots, \mathcal{T}_T)$ and $\mathcal{M} = (\mathcal{M}_1, \ldots, \mathcal{M}_T)$ denote the sets of all trees and all predicted values, respectively. Regarding the number of trees T, Chipman et al. (2010) recommend $T = 200$ as a default, though they suggest that T can also be selected by cross-validation, depending on the application.

Unlike other tree-based algorithms where a loss function is minimised to define the splitting rules in the growing process, in BART the splitting rules are uniformly defined (i.e., the covariates and their split-points are selected at random based on a uniform distribution). In addition, the BART model learns the structure of the trees by greedy modifications consisting of four moves: grow, prune, change, and swap (see Figure 5.2.1). For instance, in the grow move, a terminal node is randomly selected and two new terminal nodes are created below it. During a prune move, a parent of two terminal nodes is picked at random and its children are removed. In the change move, an internal node is randomly selected and its splitting rule is changed. Finally, in the swap move, the splitting rules associated with a pair of parent-child internal nodes are exchanged, with the pair being selected at random.

As a Bayesian model, BART places priors on the parameters of interest, assuming that $\sigma^2 \sim \mathrm{IG}(\nu/2, \nu\lambda/2)$ and $\mu_{t\ell} \sim \mathrm{N}(0, \sigma_\mu^2)$, where $\mathrm{IG}(\cdot)$ represents the inverse gamma distribution and $\sigma_\mu = 0.5/(k\sqrt{T})$, with $k \in [1, 3]$ such that each terminal node in each tree contributes only a small amount to the overall fit. In addition, a branching process prior is considered to control the depth of the trees. With this prior, each internal node ℓ' is observed at depth $d_{t\ell'}$ with probability $\eta(1 + d_{t\ell'})^{-\zeta}$, where $\eta \in (0, 1)$ and $\zeta \geq 0$. Chipman et al. (2010) recommend $\eta = 2$ and $\zeta = 0.95$, which tends to favour shallow trees.



(a) $\mathcal{T}_1^{(1)}$

(b) $\mathcal{T}_1^{(2)}$

(c) $\mathcal{T}_1^{(3)}$

(d) $\mathcal{T}_1^{(4)}$

Figure 5.2.1: An example of a tree generated from BART in 4 different instances. In principle, BART does not generate only one tree but rather a set of trees which, summed together, are responsible for the final prediction. As indicated in panel (a), observations are pushed to the left child node when the splitting criterion is satisfied. The tree is represented as $\mathcal{T}_1^{(r)}$, where $r = 1, 2, 3, 4$ denotes the number of the iteration in which the tree is updated. The splitting rules (covariates and their split-points) are presented in the internal nodes (rectangles). The predicted values $\mu_{t\ell}$ are shown inside the terminal nodes (circles). $\mathcal{T}_1^{(1)}$ illustrates the tree at iteration one with two internal nodes and three terminal nodes. From $\mathcal{T}_1^{(1)}$ to $\mathcal{T}_1^{(2)}$, the grow move is illustrated, as μ_{13} in $\mathcal{T}_1^{(1)}$ is split into μ_{13} and μ_{14} in $\mathcal{T}_1^{(2)}$ by using $x_3 < 2$. In addition, the prune move can be seen when $\mathcal{T}_1^{(2)}$ reverts to $\mathcal{T}_1^{(1)}$. The change move is shown when comparing $\mathcal{T}_1^{(2)}$ and $\mathcal{T}_1^{(3)}$, as the splitting rule that defines μ_{13} and μ_{14} is changed from $x_3 < 2$ to $x_4 < 0.75$. Finally, the swap move is illustrated in the comparison of $\mathcal{T}_1^{(3)}$ and $\mathcal{T}_1^{(4)}$.

Fitting and inference for BART models is accomplished via MCMC (Brooks et al., 2011). It is common to begin with all trees set as stumps and to initially only grow trees with high posterior probability. Thereafter, each tree is updated in turn by proposing a potential grow, prune, change, or swap move, whereby the type of move is chosen at random. Each modified tree is compared to its previous version considering the partial residuals $\mathbf{R}_t = \mathbf{y} - \sum_{j \neq t}^{T} g\left(\mathbf{X}, \mathcal{M}_j, \mathcal{T}_j\right)$ and the structure of both trees via a marginal likelihood calculation. This comparison is carried out via a Metropolis-Hastings step, and it is needed to select only splitting rules that improve the final prediction, since they are chosen based on a uniform distribution. Hence, all node-level parameters ($\mu_{t\ell}$) are generated. After doing this for all T trees, the error variance (σ^2) is updated from its full conditional distribution. This entire scheme is then iteratively repeated. The BART algorithm is practically implemented in the R packages `bartMachine` (Kapelner and Bleich, 2016), `dbarts` (Dorie, 2020), and `BART` (McCulloch et al., 2020).

5.3 Semi-parametric BART

The BART model above does not provide an easy way to quantify the effects of covariates on the response as in regression models, which is often the main goal in many applications. The semi-parametric BART framework aims to overcome this by adding a parametric linear component to the additive ensemble of non-parametric trees. We note that linear predictors and BART have also been previously combined by Prado et al. (2021), albeit in a different way. There, linear predictors are used at the terminal node level of each tree, with a focus more on prediction accuracy than interpretability. In this Section, we first revise briefly the SSP-BART of Zeldow et al. (2019) in Section 5.3.1 and then outline in detail our proposed extensions in the form of CSP-BART in Sections 5.3.2 and 5.3.3.

5.3.1 Separated semi-parametric BART

In the separated semi-parametric BART proposed by Zeldow et al. (2019), the design matrix $\mathbf{X}$ is split into two subsets, $\mathbf{X}_1$ and $\mathbf{X}_2$, with p_1 and p_2 columns, respectively. The matrix $\mathbf{X}_1$ contains covariates that should be included in a linear component to quantify the main effects and the $\mathbf{X}_2$ matrix contains covariates that might contribute to predicting the response but are not of primary interest. The linear predictor inside the BART framework is written as follows:

$$y_i \mid \mathbf{x}_{1i}, \mathbf{x}_{2i}, \boldsymbol{\beta}, \mathcal{M}, \mathcal{T}, \sigma^2 \sim \mathrm{N}\left(\mathbf{x}_{1i}\boldsymbol{\beta} + \sum_{t=1}^{T} g\left(\mathbf{x}_{2i}, \mathcal{M}_t, \mathcal{T}_t\right), \sigma^2\right). \tag{5.1}$$

Furthermore, $\mathbf{X}_1$ and $\mathbf{X}_2$ are assumed to be mutually exclusive, such that $\mathbf{X}_1 \cap \mathbf{X}_2 = \emptyset$, with p_2 large enough to ensure a BART model is feasible and relatively few columns in $\mathbf{X}_1$; i.e., $p_1 \ll p_2$, typically. As above, the ensemble of trees used by BART is learned by the four standard grow, prune, change, and swap moves.

The priors $\boldsymbol{\beta} \sim \text{MVN}(\mathbf{0}_{p_1}, \sigma_b^2 \mathbf{I}_{p_1})$ and $\sigma^2 \sim \text{IG}(\nu/2, \nu\lambda/2)$ are assumed for the linear regression coefficients and error variance, respectively, where $\text{MVN}(\cdot)$ represents the multivariate normal distribution, $\mathbf{0}_{p_1}$ and $\mathbf{I}_{p_1}$ respectively denote a p_1-dimensional vector of zeros and identity matrix, and ν, λ, and σ_b^2 are user-specified hyperparameters. Typically, σ_b^2 is set large enough so that the prior on $\boldsymbol{\beta}$ is diffuse. Notably, the isotropic covariance structure $\sigma_b^2 \mathbf{I}_{p_1}$ assumed by Zeldow et al. (2019) implies that i) all covariates in $\mathbf{X}_1$ have the same magnitude, which can easily be accomplished by appropriate transformations, and ii) covariates in $\mathbf{X}_1$ are uncorrelated, which may be unrealistic for many real-world applications.

5.3.2 Combined semi-parametric BART

In CSP-BART, we similarly allow for modelling covariates of primary and non-primary interest. Unlike SSP-BART, however, we consider that $\mathbf{X}_1$ and $\mathbf{X}_2$ may have covariates in common. This change is crucial as it allows primary covariates to interact both among themselves and with those in $\mathbf{X}_2$. Moreover, we change the tree-generation process in BART by introducing 'double grow' and 'double prune' moves to account for identifiability issues that may arise between the estimates from the linear and BART components. In CSP-BART, a univariate response y_i is modelled in accordance with equation (5.1), along with the following prior distributions:

$$\begin{aligned}
\boldsymbol{\beta} &\sim \text{MVN}\left(\mathbf{b}, \boldsymbol{\Omega}_\beta\right), \\
\boldsymbol{\Omega}_\beta &\sim \text{IW}\left(\mathbf{V}, v\right), \\
\sigma^2 &\sim \text{IG}\left(\nu/2, \nu\lambda/2\right),
\end{aligned}$$

where $\text{IW}(\cdot)$ represents the inverse Wishart distribution. We specify $\mathbf{V} = \mathbf{I}_{p_1}$ and $v = p_1$, while $\nu = 3$ and λ are chosen following Chipman et al. (2010). While the previous prior on $\boldsymbol{\beta}$ used in SSP-BART assumes that coefficients in the linear predictor are uncorrelated and equivariant, this assumption is sensible only when the covariates in $\mathbf{X}_1$ are standardised appropriately. Conversely, our hierarchical prior on $\boldsymbol{\beta}$ allows us to explicitly model correlation among the predictors in $\mathbf{X}_1$ (see Section 5.3.3). As an aside, the covariates in $\mathbf{X}_2$ need not be standardised under either CSP-BART or SSP-BART, as the splitting rules in BART are invariant under monotone transformations. Following Chipman et al. (2010); Linero (2018), we recommend transforming only the response to lie between -0.5 and 0.5 to facilitate specification of the prior on $\mu_{t\ell}$ and improve numerical stability.

To allow for $\mathbf{X}_1$ and $\mathbf{X}_2$ sharing covariates, we propose to change the moves of the BART model in order to resolve non-identifiability issues between the linear component and BART. Thus, if $\mathbf{X}_1 \cap \mathbf{X}_2 \neq \emptyset$, we propose a 'double grow' move only when $x \in \{\mathbf{X}_1 \cap \mathbf{X}_2\}$ is chosen to define a splitting rule for a stump. For example, if $\mathcal{T}_1$ is a stump and $x_1 \in \{\mathbf{X}_1 \cap \mathbf{X}_2\}$ is chosen to define a splitting rule, then another covariate, e.g., x_2, which can belong either to $\mathbf{X}_1$, $\mathbf{X}_2$, or $\mathbf{X}_1 \cap \mathbf{X}_2$, will also be chosen and the proposed tree will have at least x_1 and x_2 in its structure. If $\mathcal{T}_1$ is a stump and $x_1 \notin \{\mathbf{X}_1 \cap \mathbf{X}_2\}$ is chosen to define a splitting rule, a standard 'single' grow move is employed. The rationale behind double-growing is thus to induce interactions between covariates in $\mathbf{X}_1$ and others in either $\mathbf{X}_1$ or $\mathbf{X}_2$, and let only the linear component capture the main effects associated with the covariates in $\mathbf{X}_1$. With a single grow move, both components would try to estimate the effects of covariates in $\mathbf{X}_1$ whenever $\mathbf{X}_1$ and $\mathbf{X}_2$ have at least one covariate in common, which would lead to non-identifiability issues. However, the double grow move allows the linear component to estimate the main effects and forces BART to work specifically on interactions and non-linearities.

The 'double prune' move is proposed to prevent trees from containing only one covariate which belongs to $\mathbf{X}_1 \cap \mathbf{X}_2$. To illustrate this move, we recall Figure 5.2.1. In panel (a), the tree has 3 terminal nodes (circles) and 2 internal nodes (rectangles). If the parent of terminal nodes μ_{11} and μ_{12} is 'single' pruned, the new tree will have only x_2 in its structure. If $x_2 \notin \{\mathbf{X}_1 \cap \mathbf{X}_2\}$, which implies that $x_2 \in \mathbf{X}_2$, there will be no identifiability issues between the components in CSP-BART. However, if $x_2 \in \{\mathbf{X}_1 \cap \mathbf{X}_2\}$, the effect of x_2 will be estimated by both the linear predictor and BART. To avoid this issue, we prune the tree again.

Despite these double moves, non-identifiability issues may still arise in three cases: i) when categorical variables with more than two levels in $\mathbf{X}_1$ are used to define splitting rules in the BART component, ii) when an intercept is specified in $\mathbf{X}_1$, and iii) when any terminal node is associated with splitting rules, at any depth, which all involve only one covariate belonging to $\mathbf{X}_1 \cap \mathbf{X}_2$. The first issue is easily remedied by automatically rejecting proposed trees containing branches defined only by repeated splits on the same categorical variable in $\mathbf{X}_1$. Given that this further prevents the BART component from estimating marginal effects associated with categorical variables of primary interest, it is especially pertinent for the TIMSS application where the covariates in $\mathbf{X}_1$ are all categorical. To remedy the second issue, we stress that $\mathbf{X}_1$ should not be equipped with a leading column of ones corresponding to an intercept. Doing so would conflate the linear component's constant with the constant node-level $\mu_{t\ell}$ parameters in the BART component. Accordingly, our removal of the intercept circumvents the need to impose the constraint $\mathbb{E}\left(\sum_{t=1}^{T} g\left(\mathbf{x}_{2i}, \mathcal{M}_t, \mathcal{T}_t\right)\right) = 0$.

To illustrate the third issue, we recall Figure 5.2.1 and assume that $x_2 \in \{\mathbf{X}_1 \cap \mathbf{X}_2\}$. In panel (a), $\mathcal{T}_1^{(1)}$ represents a tree with two predictors (x_2 and x_1) in its structure, where x_1 can belong to either $\mathbf{X}_1$, $\mathbf{X}_2$, or $\mathbf{X}_1 \cap \mathbf{X}_2$. For simplicity, imagine that $\mathcal{T}_1^{(1)}$ was generated by a double grow move applied to a stump, where x_2 and x_1 were randomly selected to create the splitting rules. The two left-most terminal nodes have μ_{11} and μ_{12} as predicted values, with splitting rules defined by x_1 and x_2. However, the right-most terminal node, with predicted value μ_{13}, has only x_2 as its ancestor, which causes non-identifiability issues between the linear and BART components, since $x_2 \in \{\mathbf{X}_1 \cap \mathbf{X}_2\}$. We avoid such issues by modifying the prior on the relevant predicted value to $\mu_{t\ell} \sim \mathrm{N}(0, \sigma_\mu^2 \approx 0)$, which in turn shrinks the posterior predicted value towards zero. The prior on the other terminal nodes (μ_{11} and μ_{12} in the present example) would remain unchanged.

Regarding the change and swap moves, we stress that they are kept intact as 'single' moves in CSP-BART. Equivalent 'double change' and 'double swap' moves are not required to deal with non-identifiability issues that may arise between the linear and BART components. However, more stringent checks are placed on the validity of trees proposed by these moves. In particular, change and swap moves are iteratively proposed until a valid tree structure is found; i.e., one which ensures the parameters in the linear component are identifiable, with a minimum number of observations in each terminal node. If a valid tree is not found in some small number of iterations, a stump is proposed instead. In the end, proposed trees are always accepted or rejected according to a Metropolis-Hastings step, as in the standard BART model.

Overall, our proposals outlined above can be interpreted as an adjustment to the prior over the set of possible tree structures; effectively, a prior probability of zero is placed on invalid trees. The combined effect of our proposals is to ensure that main effects of primary interest are strictly isolated in the identifiable linear component, while interactions and non-linearities are strictly confined to the BART component.

Equations (5.2)–(5.4) below present the respective full conditional distributions for $\boldsymbol{\beta}$, $\boldsymbol{\Omega}_\beta$, and σ^2. These expressions are needed due to the inclusion of the linear predictor in the CSP-BART model; see Appendix 5.A for full details. An outline algorithm for the process is given by:

i) Update the linear predictor, with $\mathbf{r} = \mathbf{y} - \sum_{t=1}^{T} g\left(\mathbf{X}_2, \mathcal{M}_t, \mathcal{T}_t\right)$, via

$$\begin{aligned}\boldsymbol{\beta} \mid \mathbf{X}_1, \mathbf{r}, \sigma^2, \mathbf{b}, \boldsymbol{\Omega}_\beta \sim \text{MVN}\Bigg(\mu_\beta = \Sigma_\beta\left(\sigma^{-2}\mathbf{X}_1^\top \mathbf{r} + \boldsymbol{\Omega}_\beta^{-1}\mathbf{b}\right)&, \qquad (5.2)\\ \Sigma_\beta = \left(\sigma^{-2}\mathbf{X}_1^\top \mathbf{X}_1 + \boldsymbol{\Omega}_\beta^{-1}\right)^{-1}\Bigg)&,\\ \boldsymbol{\Omega}_\beta \mid \boldsymbol{\beta}, \mathbf{b}, \mathbf{V}, \upsilon \sim \text{IW}\left(\left(\boldsymbol{\beta} - \mathbf{b}\right)\left(\boldsymbol{\beta} - \mathbf{b}\right)^\top + \mathbf{V}, \upsilon + 1\right)&. \qquad (5.3)\end{aligned}$$

ii) Then, sequentially update all T trees, one at a time, via

$$\mathbf{R}_t = \mathbf{y} - \mathbf{X}_1\boldsymbol{\beta} - \sum_{j \neq t}^{T} g\left(\mathbf{X}_2, \mathcal{M}_j, \mathcal{T}_j\right).$$

iii) Finally, update

$$\sigma^2 \sim \text{IG}\left(\frac{n+\nu}{2}, \frac{S+\nu\lambda}{2}\right), \qquad (5.4)$$

where $S = \left(\mathbf{y} - \hat{\mathbf{y}}\right)^\top\left(\mathbf{y} - \hat{\mathbf{y}}\right)$ and $\hat{\mathbf{y}} = \mathbf{X}_1\boldsymbol{\beta} + \sum_{t=1}^{T} g\left(\mathbf{X}_2, \mathcal{M}_t, \mathcal{T}_t\right)$.

In Step i), the linear predictor's parameter estimates and covariance matrix are updated, taking into account the difference between the response and the predictions from all trees. In Step ii), each tree t is modified considering the updated parameter estimates $\boldsymbol{\beta}$. Finally, the error variance is updated in Step iii).

The main benefits of our approach are i) ease of implementation, relative to GLMs and GAMs, as we can model interactions and non-linearities without requiring pre-specification, ii) improved predictive performance relative to other tree-based methods, and iii) reduced bias relative to other semi-parametric BART models. Regarding computational cost, CSP-BART adds negligible time overhead to the standard BART model, especially if the number of columns in $\mathbf{X}_1$ is moderate. The computational cost of CSP-BART is also comparable to that of SSP-BART, as our novel double moves are not computationally intensive.

5.3.3 Incorporating random effects in CSP-BART

Although we have introduced CSP-BART considering only fixed effects, it is straightforward to extend it to a setting with additional random effects, whereby primary covariates are conditioned on categorical predictors. This yields

$$y_i \mid \mathbf{x}_{1i}, \mathbf{z}_i, \mathbf{x}_{2i}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \mathcal{M}, \mathcal{T}, \sigma^2 \sim \mathrm{N}\left(\mathbf{x}_{1i}\boldsymbol{\beta} + \mathbf{z}_i\boldsymbol{\gamma} + \sum_{t=1}^{T} g\left(\mathbf{x}_{2i}, \mathcal{M}_t, \mathcal{T}_t\right), \sigma^2\right),$$

where $\boldsymbol{\gamma}$ is the q-dimensional random effects vector with associated design matrix $\mathbf{Z}$. Conceptually, all effects are random under the Bayesian paradigm, but we use the terms 'fixed' and 'random' to distinguish between $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ nonetheless.

To fit such a model, we define $\boldsymbol{\beta}^\star = (\boldsymbol{\beta}, \boldsymbol{\gamma})^\top$ and $\mathbf{x}_{1i}^\star = (\mathbf{x}_{1i}, \mathbf{z}_i)$. With $\boldsymbol{\beta} \sim \mathrm{MVN}(\mathbf{b}, \boldsymbol{\Omega}_\beta)$ as above, and a $\mathrm{MVN}(\mathbf{0}_q, \boldsymbol{\Omega}_\gamma)$ prior assumed for $\boldsymbol{\gamma}$, a block-diagonal covariance matrix $\boldsymbol{\Omega}_{\beta^\star}$ is obtained in the induced prior for $\boldsymbol{\beta}^\star$, which implies that $\boldsymbol{\beta}$ and $\boldsymbol{\gamma}$ are correlated among themselves but not with each other. We relax this assumption by letting $\mathbf{b}^\star = (\mathbf{b}, \mathbf{0}_q)^\top$ and assuming $\boldsymbol{\beta}^\star \sim \mathrm{MVN}(\mathbf{b}^\star, \boldsymbol{\Omega}_{\beta^\star})$, where now $\boldsymbol{\Omega}_{\beta^\star} \sim \mathrm{IW}(\mathbf{V}^\star, v^\star)$. Subject to $\boldsymbol{\beta} = \boldsymbol{\beta}^\star$, $\mathbf{X}_1 = \mathbf{X}_1^\star$, and $\boldsymbol{\Omega}_\beta = \boldsymbol{\Omega}_{\beta^\star}$, both prior settings allow direct application of the model-fitting algorithm outlined in Section 5.3.2. Notably, only the $\boldsymbol{\Omega}_{\beta^\star}$ matrix under the latter approach accounts for potential correlations between the fixed and random effects, while the isotropic prior employed by Zeldow et al. (2019) under the SSP-BART framework would not. As ever, SSP-BART would also be unable capture interactions involving random effects in $\mathbf{X}_1$ and other covariates of non-primary interest in $\mathbf{X}_2$.

In our implementation[5], we adapt the mixed-model formula from the `lme4` (Bates et al., 2015) package, so that the linear fixed and random effects can be easily specified through a formula (e.g., `y` $\sim$ `0 + x1 + (x2 | x3)`, where `y` denotes a univariate response, `0` ensures that no intercept is included, `x1` and `x2` represent continuous covariates, and `x3` is a factor with multiple levels; see Table 2 in Bates et al. (2015) for more examples). When specifying the linear predictor, the user needs only to supply the main fixed and random effects, as any interactions among covariates of primary interest are also determined automatically by BART. Finally, we note that polynomial effects, if any are of primary interpretational interest, should also be specified in the linear predictor only, as splitting rules based on x_1 or x_1^3, for example, would yield equivalent trees, and it would be necessary to avoid trees whose only splits involve both x_1 and monotonic transformations thereof.

5.4 Simulation experiments

In this Section, we compare our novel CSP-BART with GAMs, SSP-BART, and VCBART in terms of bias (i.e., the difference between the posterior mean parameter estimates and the true parameter values) using two sets of synthetic data. The results were obtained using `R` (R Core Team, 2020) version 4.11 and the `R` packages `mgcv` (Wood, 2017), `semibart` (Zeldow et al., 2019), and `VCBART` (Deshpande et al., 2020). For CSP-BART and SSP-BART, we use $T = 50$ trees, 2,000 MCMC iterations as burn-in, and 2,000 as post-burn-in. We use the default arguments of the `mgcv` and `VCBART` packages, with the exception of `intercept=FALSE` being specified for `VCBART` for the sake of comparability with CSP-BART and SSP-BART. We note that the GAM is the only non-Bayesian method among the set of comparators. As GAMs require explicit specification of terms to be included in the linear predictor, we supply the true structure used to simulate the data in both experiments. This gives GAMs an unfair advantage over the other methods, but does provide a baseline that the BART-based methods can aim for.

[5] Available at `https://github.com/ebprado/CSP-BART`.

5.4.1 Friedman dataset

In this first scenario, we consider the Friedman equation (Friedman, 1991):

$$y_i = 10\sin\left(\pi x_{i1} x_{i2}\right) + 20\left(x_{i3} - 0.5\right)^2 + 10x_{i4} + 5x_{i5} + \epsilon_i,\ i = 1, \ldots, n,$$

where $x_{\cdot j} \sim \text{Uniform}(0, 1)\ \forall\ j = 1, \ldots, p$ and $\epsilon_i \sim \text{N}(0, \sigma^2)$. This equation is used for benchmarking tree-based methods using synthetic data, and has been used in many other papers, e.g., Chipman et al. (2010); Linero (2018); Deshpande et al. (2020). In this experiment, we set $n = 1000$, $p = (10, 50)$, and $\sigma^2 = (1, 10)$, totalling four scenarios. To evaluate model performance, we use the bias of the parameter estimates as the accuracy measure, across 50 replicates of the data-generation process. As the Friedman equation uses only 5 covariates to generate the response, the additional $x_{\cdot j}$ are noise, and have no impact on y_i. In this simulation, we aim to estimate the $p_1 = 2$ linear effects associated with x_4 and x_5 (denoted by $\beta_4 = 10$ and $\beta_5 = 5$, respectively) using the linear predictor, i.e., we set up $\mathbf{X}_1$ so that it contains only x_4 and x_5. In contrast, we let BART take care of the non-linear and interaction effects by setting $\mathbf{X}_2$ to contain all p covariates (including x_4 and x_5).

Figure 5.4.1 shows the results of bias exhibited by GAMs, SSP-BART, VCBART, and the novel CSP-BART for each combination of p and σ^2. As GAMs require all terms that are estimated by the model to be specified, we supply the true structure of the Friedman equation so that it can be used as a reference in the comparison. The CSP-BART and SSP-BART estimates are notably similar. We can see that the bias of the parameter estimates is low and both recover the true effects in all four scenarios. This is expected and can be attributed to the fact that x_4 and x_5 do not interact with other covariates. Consequently, the trees in CSP-BART tend not to contain x_4 and x_5 as both effects are captured solely by the linear predictor. We note also that VCBART presents larger bias for both β_4 and β_5 in all but one scenario. As VCBART estimates β_4 and β_5 using BART models that employ a set of effect modifiers (i.e., all covariates of non-primary interest), the results shown in Figure 5.4.1 are unsurprising since, in this example, β_4 and β_5 depend exclusively on x_4 and x_5, respectively.

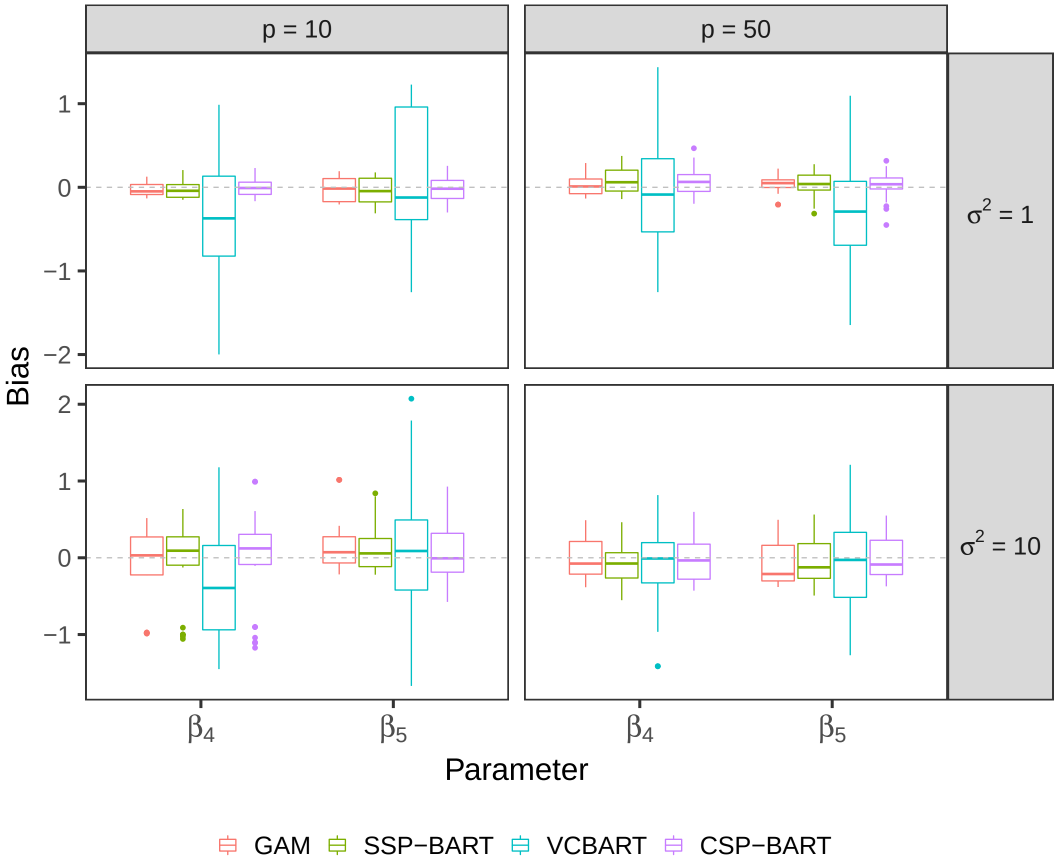


Figure 5.4.1: Boxplots of simulation results obtained across 50 replicate datasets generated according to the Friedman equation, considering $n = 1000$, $p = (10, 50)$, and $\sigma^2 = (1, 10)$. The y-axis exhibits the bias related to the parameter estimates $\hat{\beta}_4$ and $\hat{\beta}_5$ for GAM, SSP-BART, VCBART, and the novel CSP-BART. Recall that the GAM has been given the true model structure so its superior performance is expected.

5.4.2 Estimating main effects in the presence of interactions

In the scenario above, we have shown that the novel CSP-BART correctly estimates the main effects when they do not have any interactions with other effects. However, in practice, the covariates of primary interest may interact, either among themselves or with other effects, which should be taken into account. In this sense, the simulation setting which follows is likely to better reflect the nature of the TIMSS data (see Section 5.5.1) and other real-world applications.

In this scenario, we compare the methods using the regression function

$$y_i = 10x_{i1} - 5x_{i2} + (\mathcal{T}_1 \mid \mathbf{x}_i) + \epsilon_i,\ i = 1, \ldots, n, \tag{5.5}$$

where $x_{\cdot j} \sim \text{Uniform}(0, 1)\ \forall\, j = 1, \ldots, p$ and $\epsilon_i \sim \text{N}(0, \sigma^2)$, as before, and $\mathcal{T}_1 \mid \mathbf{x}_i$ represents the tree structure shown in Figure 5.4.2. As per Section 5.4.1, we consider $n = 1000$, $p = (10, 50)$, and $\sigma^2 = (1, 10)$, where the additional covariates have no impact on the response. We are now interested in estimating the effects associated with x_1 and x_2 (denoted by $\beta_1 = 10$ and $\beta_2 = -5$, respectively). This is achievable under CSP-BART by specifying $\mathbf{X}_1$ to contain only x_1 and x_2 and $\mathbf{X}_2$ to contain all p covariates, including x_1 and x_2.

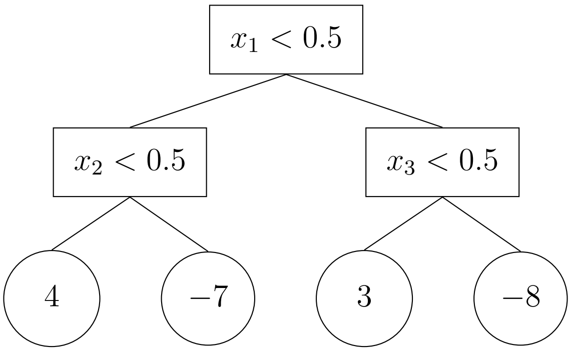


Figure 5.4.2: An illustration of the tree structure used to generate the response via (5.5). In if-else format this can be written as $\mathcal{T}_1 \mid \mathbf{x}_i = f(x_{i1}, x_{i2}, x_{i3}) = 4\mathbb{1}(x_{i1} < 0.5) \times \mathbb{1}(x_{i2} < 0.5) - 7\mathbb{1}(x_{i1} < 0.5) \times \mathbb{1}(x_{i2} \geq 0.5) + 3\mathbb{1}(x_{i1} \geq 0.5) \times \mathbb{1}(x_{i3} < 0.5) - 8\mathbb{1}(x_{i1} \geq 0.5) \times \mathbb{1}(x_{i3} \geq 0.5)$, where $\mathbb{1}(\cdot)$ denotes the indicator function. Note that the tree splits on both primary (x_1 and x_2) and non-primary (x_3) covariates.

Figure 5.4.3 shows the bias in the estimates of β_1 and β_2. While CSP-BART estimates both parameters with low bias, regardless of p and/or σ^2, SSP-BART gives large bias for β_1 and even more pronounced bias for β_2 in all scenarios. These biases occur as x_1 and x_2 are not available to the BART component of SSP-BART. We conjecture that β_2 exhibits greater bias than β_1 because x_2 appears at a lower depth than x_1 in Figure 5.4.2; i.e., in closer proximity to terminal nodes. This notion is supported by further experiments, conducted but not shown here, in which alternative tree structures with varying depth levels for x_2 were used.

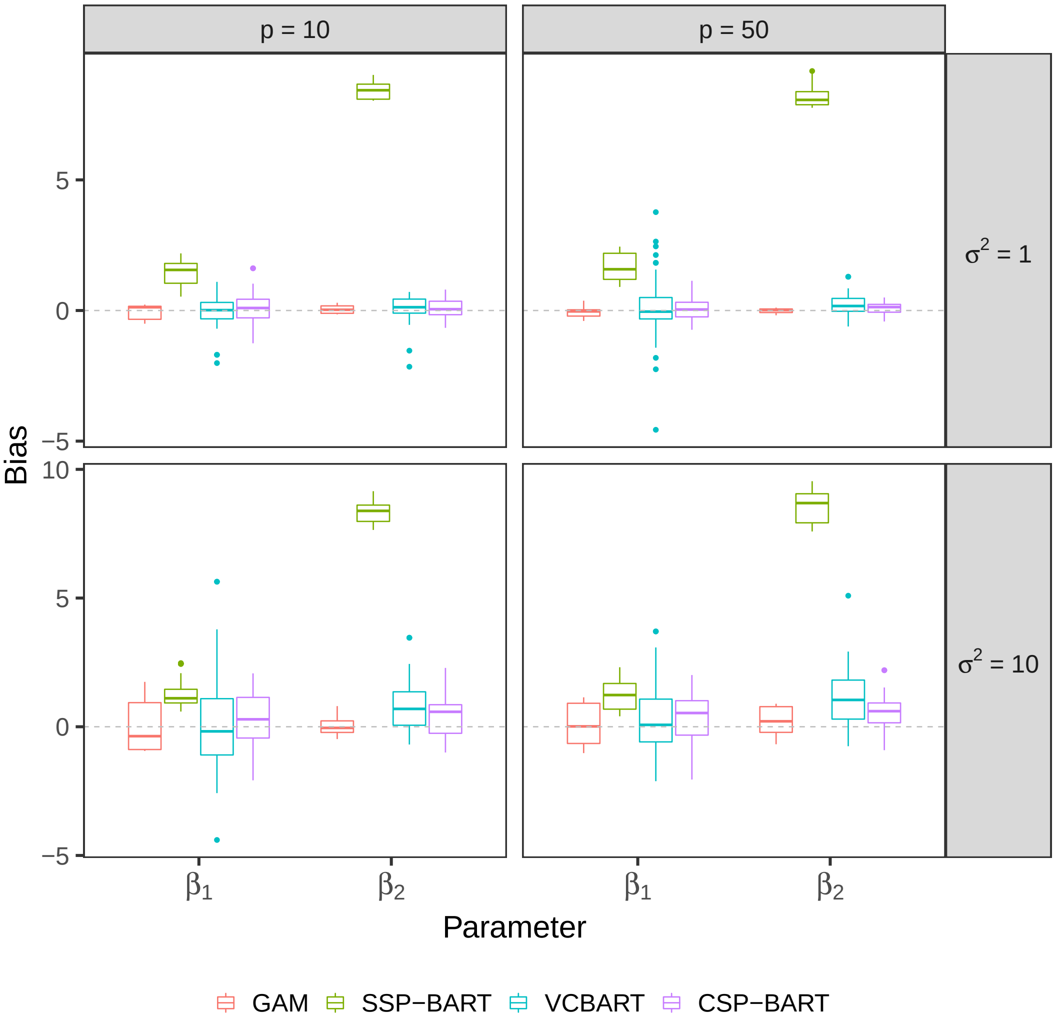


Figure 5.4.3: Boxplots of the simulation results obtained across 50 replicate datasets generated according to equation (5.5), considering $n = 1000$, $p = (10, 50)$, and $\sigma^2 = (1, 10)$. The y-axis exhibits the bias related to the parameter estimates $\hat{\beta}_1$ and $\hat{\beta}_2$ for GAM, SSP-BART, VCBART, and the novel CSP-BART. Recall that the GAM has been given the true model structure so its superior performance is expected.

Furthermore, it can be seen that VCBART and CSP-BART provide similar bias for both parameters, and match well with the baseline GAM model to which the true structure is supplied, as it is unable to capture non-specified interactions. However, it is worth recalling that VCBART uses a BART model to estimate each parameter in the linear predictor. For these data, VCBART uses $50 \times 2 = 100$ trees in total to estimate β_1 and β_2, as the `VCBART` package uses 50 trees for each parameter, by default. In this sense, the greater the number of parameters to be estimated in the linear predictor, the more computationally intensive VCBART becomes, since the total number of trees used to estimate all covariate effects is a function of the number of covariates in the linear predictor and the number of trees used to approximate each effect.

5.5 Results on real data

We now turn to the analysis of the Trends in International Mathematics and Science Study (TIMSS) dataset in Section 5.5.1, which initially motivated the development of CSP-BART and offers a large and challenging test of the model. We then demonstrate the use of CSP-BART in a classification rather than regression setting via an additional, smaller application to a well-known benchmark dataset in Section 5.5.2. SSP-BART and other previously proposed tree-based methods are used as comparators throughout.

5.5.1 TIMSS 2019

TIMSS is an international series of assessments which takes place every four years. The TIMSS 2019 dataset records students' achievements in mathematics and science at the fourth and eighth grade levels in 64 countries (Mullis et al., 2020; Fishbein et al., 2021), along with information sourced from surveys of students, teachers, and school principals. Here, we are specifically interested in quantifying the impact of some covariates on students' mathematics scores (variable 'BSMMAT01'). In our analysis, we only consider data from Ireland (where mathematics is a compulsory subject) pertaining to students at the eighth grade level, comprising 4,118 observations. As the TIMSS data were originally split by the sources of information, some data manipulation was required. During the data wrangling, covariates with a high level of missing values were discarded entirely, in order to avoid the use of imputation methods and keep as many observations as possible.

We selected three covariates of primary interpretational interest as candidates for inclusion in the linear predictor via the $\mathbf{X}_1$ matrix for this application. Notably, all three are categorical variables: 'school discipline problems' (3 levels), 'parents' education level' (6 levels), and 'minutes spent on homework' (6 levels). Our interest in these covariates follows work in the applied literature which shows that student's achievement is influenced by these factors. For instance, a previous investigation into the relation between students' performance in mathematics and various student-level and school-level factors using data from various countries gathered under the third cycle of TIMSS identified significant effects due to family background and time spent on academic activities (Martin et al., 2000).

These three factors as well as a number of others, such as the gender of the student, an index of the wealth of the school and its surrounding area, and a measure of the resources available for learning at home, have also been shown to be strongly related with the outcome in similar international education studies (Mohammadpour et al., 2015; Grilli et al., 2016). In order to identify additional predictors among those available in the present application for inclusion in the BART component, we apply a BART model to the complete cases of the aforementioned screened dataset. Reassuringly, we note that the $p_1 = 3$ covariates above are among the 20 most-used covariates under such a model. The remaining $p_2 = 17$ covariates, which may help to improve prediction but are not of primary interpretational interest, are specified in the $\mathbf{X}_2$ matrix. Full details of the chosen covariates are provided in Table B.1 in Appendix 5.B. In what follows, $n = 3224$ complete cases remain when only these 20 covariates are considered and the selected primary covariates are also shared with $\mathbf{X}_2$ when fitting the CSP-BART model but excluded from $\mathbf{X}_2$ under SSP-BART.

Initially, we compare CSP-BART with the Bayesian causal forest model (BCF; Hahn et al., 2020). To do so, we only consider the covariate 'school discipline problems' in the linear predictor of CSP-BART and as the treatment variable for BCF, with SSP-BART included as an additional comparator. CSP-BART and

SSP-BART thus differ in that this covariate is also specified in $\mathbf{X}_2$ under CSP-BART, but is exclusive to $\mathbf{X}_1$ under SSP-BART. Though this is a categorical variable with 3 levels ('hardly any problems', 'minor problems', and 'moderate to severe problems'), we binarise it by collapsing the first two levels. These modelling decisions are to the advantage of BCF, as it can only deal with a single binary covariate as the treatment effect. The goal is to quantify the impact of discipline problems on students' mathematics scores along with the other 19 covariates (i.e., the other two primary covariates are specified only in $\mathbf{X}_2$ for this preliminary analysis). In Table 5.5.1, we summarise the posterior distributions of the parameter estimates for BCF, CSP-BART, and SSP-BART. The marginal effect of school discipline problems is negative in each case, which means that students who study in schools with moderate to severe discipline issues tend to have lower mathematics scores than those in schools with hardly any or minor discipline problems. However, this covariate also defines at least one split in 2.8% of the sampled trees in the BART component of CSP-BART; i.e., it also interacts with non-primary covariates in $\mathbf{X}_2$. Notably, BCF yields a much wider credible interval (CI) than both CSP-BART and SSP-BART, though all CIs exclude zero.

Table 5.5.1: Descriptive measures of the posterior distribution of the 'school discipline problems' covariate's effect on students' mathematics scores. The estimates relate to the level 'moderate to severe problems', as the reference level merges those with 'hardly any' or 'minor' problems.

Method	**Mean**	**2.5-th percentile**	**97.5-th percentile**
BCF	−36.07	−62.05	−12.24
CSP-BART	−37.43	−54.45	−24.78
SSP-BART	−38.05	−48.09	−27.73

As we now consider all three aforementioned categorical covariates of primary interest, we note that BCF is inadequate for this application as it admits only one binary covariate. As VCBART extends BCF to allow for more covariates (of any type) in the linear predictor, we replace BCF with VCBART in the comparison with CSP-BART and SSP-BART. We use 80% of the data for training and use the remaining 20% as a test set to evaluate out-of-sample prediction performance.

Firstly, we note that the root mean squared errors on the test set are comparable for CSP-BART (58.1) and SSP-BART (58.4)[6], but VCBART (62.3) is slightly worse. Secondly, we present the parameter estimates based on the training set under each model for the three chosen primary covariates, along with associated 95% CIs, in Table 5.5.2.

Students whose parents studied at 'university or higher' or obtained 'post-secondary' qualifications tend to have higher mathematics scores than those whose parents were educated up to secondary level at most. The effects become more pronounced at lower education levels. A similar pattern of higher scores is observed for students who devote increasingly more time to homework, with the notable exception of CSP-BART; this is the only method to suggest that students who spend 'more than 90 minutes' on homework score less than those who spend less time (but still more than those who do 'no homework'). However, we point out that this finding is based on the posterior mean only (-9.07) since there is considerable uncertainty associated with it (i.e., the 95% CI includes 0). VCBART's estimates are quite extreme for these two levels, possibly due to the small numbers of observations therein. Lastly, all models suggest that students in schools with 'moderate to severe' discipline problems tend to have lower scores than those in schools with 'hardly any' or 'minor' problems.

Though their posterior mean estimates differ only in magnitude and not in sign (with only one aforementioned exception), another important aspect shown in Table 5.5.2 is the difference between the CIs from CSP-BART and SSP-BART. Notably, all CIs are much wider for SSP-BART. In particular, they all contain zero, while the effect associated with parents having 'university or higher' education is bounded away from zero by CSP-BART. As the models assume different priors for the linear regression parameters, we conducted additional experiments (not shown here, for brevity) by fitting hybrid models which swap the priors from CSP-BART and SSP-BART. In doing so, we verified that the assumption of a diffuse isotropic prior under SSP-BART is driving the disparities in these intervals. Thus, CSP-

[6] Notably, we use code from our own implementation of CSP-BART in order to fit both CSP-BART and SSP-BART, as it is not possible to predict on out-of-sample data using the R implementation of SSP-BART provided by the authors of Zeldow et al. (2019). This is achieved by adopting the prior $\boldsymbol{\beta} \sim \text{MVN}(\mathbf{0}_{p_1}, \sigma_b^2 \mathbf{I}_{p_1})$ and appropriately specifying $\mathbf{X}_1$ and $\mathbf{X}_2$.

BART allowing the covariates of primary interpretational interest to be correlated and/or have different variances *a priori* appears to have a strong impact on the posterior uncertainty of the estimates.

Table 5.5.2: Posterior mean estimates and corresponding 95% credible intervals (in parentheses) for the effects of parents' education level, minutes spent on homework, and school discipline problems on students' mathematics scores. The number of observations in each categorical level is shown in parentheses throughout the 'Category' column. The results are based on a training subset (80%) of the TIMSS 2019 dataset. Owing to the categorical nature of the covariates, a post-processing transformation is applied to the parameter estimates so that they sum to zero for each covariate, in order to facilitate an easier comparison across methods. The endpoints of the CIs are also modified accordingly. However, we stress that boldface font is used to highlight the cases where the original CI — prior to transformation — does not contain zero.

		Estimate (95% CI)		
Covariate	**Category (n)**	**CSP-BART**	**SSP-BART**	**VCBART**
Parents' education level	University or higher (870)	**22.51(0.52; 44.02)**	23.12(−63.14; 109.94)	**24.26(19.70; 30.35)**
	Post-secondary (546)	19.57(−3.00; 41.35)	21.58(−67.17; 108.77)	**20.06(6.60; 43.89)**
	Upper secondary (340)	−2.83(−25.24; 18.95)	−1.45(−88.15; 83.40)	−5.42(−38.75; 34.74)
	Lower secondary (96)	−13.36(−40.66; 10.08)	−11.82(−99.33; 74.29)	−16.62(−38.16; 13.96)
	Primary/secondary (42)	−23.03(−53.10; 9.27)	−29.53(−119.30; 59.72)	−18.82(−46.47; 91.12)
	Not informed (685)	−2.86(−25.30; 18.80)	−1.90(−89.18; 83.22)	−3.45(−10.73; 0.32)
Minutes spent on homework	No homework (21)	−20.33(−52.14; 11.02)	−29.30(−124.43; 62.04)	−127.15(−4461.47; 3372.12)
	1 to 15 minutes (862)	−3.93(−47.32; 24.60)	−0.16(−91.19; 89.26)	**19.07(−0.29; 34.23)**
	16 to 30 minutes (1158)	11.70(−11.37; 40.25)	6.82(−84.75; 96.23)	25.27(16.55; 38.31)
	31 to 60 minutes (441)	9.60(−14.40; 37.45)	4.39(−86.17; 94.53)	26.93(8.82; 67.22)
	61 to 90 minutes (62)	12.02(−29.99; 48.77)	13.78(−77.26; 103.48)	**15.45(−17.92; 24.02)**
	More than 90 minutes (35)	−9.07(−52.29; 29.77)	4.47(−88.01; 96.28)	40.43(−13.54; 395.37)
School discipline problems	Hardly any problems (1621)	15.36(−12.68; 45.64)	16.16(−86.32; 119.59)	**16.59(−15.65; 58.45)**
	Minor problems (891)	9.81(−19.41; 40.21)	10.82(−91.01; 115.98)	**6.67(−21.54; 33.23)**
	Moderate to severe (67)	−25.18(−55.30; 6.50)	−26.97(−131.20; 80.13)	−23.27(−48.81; 158.65)

To show the benefits of CSP-BART sharing covariates across components, it is of interest to detect interaction effects between covariates in $\mathbf{X}_1$ and others of both primary and non-primary interest. According to Chipman et al. (2013), an interaction exists between two variables if they are in the same tree. Here, 18.8% of trees across all MCMC samples have interactions of this sort between at least one covariate in $\mathbf{X}_1$ and another in either $\mathbf{X}_1$ or $\mathbf{X}_2$, while 0.5% are stumps and 54.7% split on one covariate only. More specifically, we detect non-spurious interactions, using a stricter definition whereby covariates must be in the same branch (Kapelner and Bleich, 2016), between 'parents' education level' and 'minutes spent on homework' (both in $\mathbf{X}_1$) and between 'school discipline problems' (in $\mathbf{X}_1$) and 'absenteeism' (5 levels, in $\mathbf{X}_2$). A major limitation of SSP-BART is that it would fail to detect key interactions such as these. Due to the assumption of mutual-exclusivity between $\mathbf{X}_1$ and $\mathbf{X}_2$, SSP-BART can only capture interactions between two or more non-primary covariates in $\mathbf{X}_2$. Our CSP-BART also detects frequent interactions of this sort in the remaining 26.0% of trees; e.g., between 'absenteeism' and 'how often the student feels hungry' (4 levels). To detect important interactions in VCBART, one would need to examine all trees for all covariates in the linear predictor. This would amount to 150 trees per iteration, as the effect associated with each of the 3 primary covariates is approximated by 50 trees (by default).

5.5.2 Pima Indians Diabetes

We now analyse the well-known Pima Indians Diabetes dataset from the UCI Machine Learning Repository (Blake, 1998), which is available in R through the `mlbench` package (Leisch et al., 2009), to demonstrate the use of CSP-BART in a classification setting. Unlike the TIMSS application, here the response is binary rather than continuous and all covariates are continuous rather than categorical. The goal is to predict whether or not a patient has diabetes based on age, blood pressure, body mass index, glucose concentration, and 4 other covariates. We analyse a corrected version of the data which treats physically impossible values of zero for a number of covariates as missing values, which we in turn omit. We are primarily interested in measuring the effects of age and glucose through the linear predictor. As the response variable is binary, we use a probit link function following the data augmentation scheme of Albert and Chib (1993).

We only compare CSP-BART and SSP-BART, as the `VCBART` package cannot deal with binary responses. Henceforth, all parameter estimates are based on a training set comprising a randomly chosen 80% of the data and misclassification rates based on the remaining 20% are used to quantify prediction accuracy. For CSP-BART, we specify age and glucose in $\mathbf{X}_1$ and supply all 8 available covariates, including age and glucose, to the BART component. For SSP-BART, we specify age and glucose in $\mathbf{X}_1$ and the 6 remaining covariates in $\mathbf{X}_2$, as SSP-BART does not allow for covariates to be shared across the linear and BART components. In both cases, the intercept is omitted from the $\mathbf{X}_1$ matrix, as described in Section 5.3.2.

We present the parameter estimates for age and glucose, with corresponding 95% CIs, in Table 5.5.3. Under both models, the estimates for both covariates indicate that, as they increase, the probability of observing positive diabetes diagnoses also increases, and *vice versa*. All CIs also have positive lower and upper limits. It is especially notable, however, that the CI for the age effect is bounded further away from zero under the CSP-BART model; i.e., we detect a more significant marginal age effect.

To highlight the efficacy of the hierarchical prior on $\boldsymbol{\beta}$, the double grow/prune moves, and our other proposals for addressing non-identifiability, we also fit a hybrid model, equipped with the isotropic prior from SSP-BART, with age and glucose in both components, but without the double moves and stringent checks on tree-structure validity used in CSP-BART. Such a model achieves a misclassification rate of 19.23% on the test set; slightly better than SSP-BART itself (20.51%), though still inferior to the proper CSP-BART (17.94%).

Under the hybrid model, we observe that the additional inclusion of age and glucose in the $\mathbf{X}_2$ matrix used by the BART component generates trees that occasionally use only age or only glucose. In this case, the linear predictor and BART component both try to estimate the effects of these covariates, which is not sensible as it generates non-identifiability issues between the two components and bias in the estimates of the parameters in the linear predictor. Overall, the benefits arising from i) sharing covariates among the components, ii) the employment of double grow and double prune moves, along with other checks on tree-structure validity, and iii) the adoption of the hierarchical prior on $\boldsymbol{\beta}$ are evident from the superior out-of-sample classification accuracy of CSP-BART.

Table 5.5.3: Posterior mean estimates of the age (years) and glucose (mg/dL) effects on the diagnosis of diabetes, with corresponding 95% CIs, according to CSP-BART and SSP-BART models fit to the training set (80%).

	CSP-BART		**SSP-BART**	
Covariate	**Estimate**	**95% CI**	**Estimate**	**95% CI**
Age	0.0634	(0.0285; 0.1006)	0.0287	(0.0016; 0.0572)
Glucose	0.0359	(0.0271; 0.0445)	0.0296	(0.0221; 0.0377)

5.6 Discussion

In this work, we have extended BART to a semi-parametric framework which circumvents many of the restrictions and identifiability issues found in other versions of semi-parametric BART. In semi-parametric BART models, the main effects are estimated via a linear predictor, while interactions and non-linearities are dealt with by a BART component. The main novelties of our CSP-BART are i) the sharing of covariates between the linear and BART components, in tandem with ii) additional double grow and double prune moves. These innovations combine to induce additional interactions between covariates of primary interest, both among themselves and with those available to the BART component, which ensures that marginal effects of primary interest are strictly isolated in the identifiable linear component while interactions and non-linearities are strictly confined to the BART component. Our modifications can be interpreted as adjustments to the prior over the set of possible tree structures; effectively, a prior probability of zero is placed on invalid trees. We have implemented CSP-BART as an R package, which is currently available at https://github.com/ebprado/CSP-BART.

Through simulation studies and two applications to a well-known benchmark dataset and novel data from an international education assessment, the ability of CSP-BART to estimate marginal effects with low bias, while not requiring prespecification of interaction effects, has been demonstrated in both regression and classification settings. Regarding the motivating TIMSS application, we note that CSP-BART offers particularly interesting insight into the effect of the 'minutes spent on homework' covariate on students' mathematics scores. While competing methods suggest that scores improve indefinitely as the time devoted to homework increases, CSP-BART suggests that the effect is reversed for those who spend more than 90 minutes on homework, which implies that students who do so might actually be weak students who struggle with their homework exercises or mathematics classes in general.

We also showed that CSP-BART captures many important interactions between covariates of primary interest and others of both primary and non-primary interest in the TIMSS application, by virtue of CSP-BART allowing the covariates of primary interest to be shared with both the linear and BART components. Such interactions cannot be captured by the SSP-BART or VCBART models, and would need to be explicitly specified if instead fitting a linear model, such as a GLM or a GAM. However, we note that mixed-effects models are widely used in the analysis of similar education assessment datasets (Mohammadpour et al., 2015; Grilli et al., 2016). As the present analysis in Section 5.5.1 only considers fixed effects in the linear component, our proposals for incorporating random effects outlined in Section 5.3.3 are thus of interest for future practical work.

In terms of future methodological extensions, other BART-based models, such as SBART (Linero, 2018), log-linear BART (Murray, 2021), and BART for gamma and log-normal hurdle data (Linero et al., 2020), could be embedded in semiparametric frameworks following a similar approach. A semi-parametric version of SBART, in particular, could prove especially fruitful for the TIMSS application. Theoretical results underlying CSP-BART could also be developed in order to explore its posterior convergence properties.

Overall, we anticipate CSP-BART enjoying great utility in a wide range of application settings, beyond the TIMSS data considered herein. The model accommodates multiple covariates, yields improved out-of-sample prediction/classification performance, and ensures accurate inference of important linear effects while accounting for additional non-specified interactions (beyond those already accounted for by other semi-parametric BART models). Furthermore, the model-fitting algorithm enables straightforward incorporation of random effects and has built-in strategies to address non-identifiability issues. Notably, its run-time is comparable or superior to its competitors BCF, SSP-BART, and VCBART, which all have one or more of these limitations and perform poorly on the TIMSS data.

Appendix

5.A Semi-parametric BART implementation

In this Section, we provide details for the implementation of the CSP-BART model, which can be written as

$$y_i \mid \mathbf{x}_{1i}, \mathbf{x}_{2i}, \boldsymbol{\beta}, \mathcal{M}, \mathcal{T}, \sigma^2 \sim \mathrm{N}\left(\mathbf{x}_{1i}\boldsymbol{\beta} + \sum_{t=1}^{T} g\left(\mathbf{x}_{2i}, \mathcal{M}_t, \mathcal{T}_t\right), \sigma^2\right).$$

We recall that CSP-BART and SSP-BART (Zeldow et al., 2019) differ in many aspects, with the latter assuming that i) $\mathbf{X}_1$ and $\mathbf{X}_2$ are disjoint matrices, such that only 'single' grow/prune moves are considered, and ii) $\boldsymbol{\beta} \sim \mathrm{MVN}(\mathbf{0}_{p_1}, \sigma^2_{\beta}\mathbf{I}_{p_1})$, where $\mathbf{0}_{p_1}$ and $\mathbf{I}_{p_1}$ respectively denote a vector of zeros and an identity matrix of appropriate dimension and σ^2_b is a fixed, large scalar, such that the prior on $\boldsymbol{\beta}$ is uninformative. In contrast, CSP-BART i) allows $\mathbf{X}_1$ and $\mathbf{X}_2$ to share covariates, which is rendered valid by the novel double grow/prune moves employed, and ii) assumes $\boldsymbol{\beta} \sim \mathrm{MVN}(\mathbf{b}, \boldsymbol{\Omega}_{\beta})$, with the associated hyperprior $\boldsymbol{\Omega}_{\beta} \sim \mathrm{IW}(\mathbf{V}, v)$. This hierarchical prior allows for more complex covariance structures for the linear predictor's parameters to be explicitly modelled. In terms of commonalities, both methods consider that $\sigma^2 \sim \mathrm{IG}(\nu/2, \nu\lambda/2)$ and define the partial residuals as $\mathbf{R}_t = \mathbf{y} - \mathbf{X}_1\boldsymbol{\beta} - \sum_{j \neq t}^{T} g(\mathbf{X}_2, \mathcal{M}_j, \mathcal{T}_j)$.

In Algorithm 4, the structure of CSP-BART is presented. Firstly, the response and the design matrices $\mathbf{X}_1$ and $\mathbf{X}_2$ are specified, along with the number of trees (e.g., $T = 200$), number of MCMC iterations M, and all hyperparameters associated with the priors for $\boldsymbol{\beta}$, $\boldsymbol{\Omega}_{\beta}$, $\mu_{t\ell}$, $\mathcal{T}_t$, and σ^2. Initially, all trees are set as stumps. Secondly, the parameter vector $\boldsymbol{\beta}$ and covariance matrix $\boldsymbol{\Omega}_{\beta}$ are updated. Thereafter, candidate trees ($\mathcal{T}_t^\star$) are sequentially proposed, one at a time — via one of the four standard moves employed by SSP-BART, or one of the novel 'double grow' and 'double prune' moves — and compared with their previous versions ($\mathcal{T}_t$) via a Metropolis-Hastings step. Later, the node-level parameters $\mu_{t\ell}$ are generated. Finally, the variance σ^2 is updated. For sufficiently large M, samples from the posterior distribution of the trees are obtained upon convergence.

Algorithm 4 describes CSP-BART considering only fixed effects. However, we recall that the model can be extended to also incorporate random effects, such that

$$y_i \mid \mathbf{x}_{1i}, \mathbf{z}_i, \mathbf{x}_{2i}, \boldsymbol{\beta}, \boldsymbol{\gamma}, \mathcal{M}, \mathcal{T}, \sigma^2 \sim \mathrm{N}\left(\mathbf{x}_{1i}\boldsymbol{\beta} + \mathbf{z}_i\boldsymbol{\gamma} + \sum_{t=1}^{T} g\left(\mathbf{x}_{2i}, \mathcal{M}_t, \mathcal{T}_t\right), \sigma^2\right),$$

where $\boldsymbol{\gamma}$ is the random effects vector of dimension q and $\mathbf{z}_i$ denotes the i-th row of the associated design matrix $\mathbf{Z}$. For completeness, we reiterate that the same algorithm can be directly used to fit such a model, following the scheme outlined in Section 5.3.3.

Algorithm 4 CSP-BART model

```
Input: $\mathbf{y}$, $\mathbf{X}_1$, $\mathbf{X}_2$, number of trees $T$, and number of MCMC iterations $M$.
Initialise: $\{\mathcal{T}_t\}_1^T$ and set all hyperparameters of the prior distributions.
for $(m = 1$ to $M)$ do
    Update the parameter vector $\boldsymbol{\beta}$ via (5.2).
    Update the covariance matrix $\boldsymbol{\Omega}_\beta$ via (5.3).
    for $(t = 1$ to $T)$ do
        Compute $\mathbf{R}_t = \mathbf{y} - \mathbf{X}_1\boldsymbol{\beta} - \sum_{j \neq t}^{T} g(\mathbf{X}_2, \mathcal{M}_j, \mathcal{T}_j)$.
        Propose a new tree $\mathcal{T}_t^\star$ by a grow, double grow, prune, double prune,
        change, or swap move; iterate until a valid tree structure is obtained.
        Compare the current $(\mathcal{T}_t)$ and proposed $(\mathcal{T}_t^\star)$ trees via Metropolis-
        Hastings, with
            $\alpha\left(\mathcal{T}_t, \mathcal{T}_t^\star\right) = \min\left\{1, \frac{p\left(\mathcal{T}_t^\star \mid \mathbf{R}_t, \sigma^2\right) q(\mathcal{T}_t^\star \to \mathcal{T}_t)}{p(\mathcal{T}_t \mid \mathbf{R}_t, \sigma^2) q(\mathcal{T}_t \to \mathcal{T}_t^\star)}\right\}$.
        Sample $u \sim \text{Uniform}(0,1)$: if $\alpha\left(\mathcal{T}_t, \mathcal{T}_t^\star\right) < u$, set $\mathcal{T}_t = \mathcal{T}_t$, otherwise
        set $\mathcal{T}_t = \mathcal{T}_t^\star$.
        Update all node-level parameters $\mu_{t\ell}$ via
            $\mu_{t\ell} \mid \mathcal{T}_t, \mathbf{R}_t, \sigma^2 \sim \text{N}\left(\frac{\sigma^{-2}\sum_{i \in \mathcal{P}_{t\ell}} r_i}{n_{t\ell}/\sigma^2 + \sigma_\mu^{-2}}, \frac{1}{n_{t\ell}/\sigma^2 + \sigma_\mu^{-2}}\right)$
         for $\ell = 1, \ldots, b_t$.
    end for
    Update $\sigma^2$ via (5.4).
    Update the predicted response $\hat{\mathbf{y}}$.
end for
Output: samples of the posterior distribution of $\mathcal{T}$.
```

5.B TIMSS dataset

In Section 5.5.1, we analysed data on Irish students at eighth grade level. To illustrate the novel CSP-BART, only one plausible value of the students' mathematics scores was used, and sampling weights were not accounted for. Nonetheless, it would be necessary to consider all five mathematics scores along with sampling weights for a more complete analysis; see Rutkowski et al. (2010) and Foy (2017) for details. In Table B.1, we present the 20 covariates that were pre-selected to demonstrate CSP-BART. These covariates were selected by identifying the 20 most-used variables in a standard BART model fit to the TIMSS dataset. In the comparisons with BCF, SSP-BART, and VCBART, all 20 covariates were included in the $\mathbf{X}_2$ matrix under CSP-BART, which is the matrix used by the BART component. In the first comparison with BCF, $\mathbf{X}_1$ contained only the binarised version of the covariate 'BCDGDAS', which was in turn excluded from $\mathbf{X}_2$ under SSP-BART. When comparing CSP-BART with VCBART, $\mathbf{X}_1$ contained the covariates 'BSDGEDUP', 'BSBM42BA', and 'BCDGDAS', which were in turn excluded from $\mathbf{X}_2$ under SSP-BART.

Table B.1: Covariates pre-selected for the analysis of the TIMSS 2019 dataset, listed according to how often they were used by a standard BART model fit to the same data, such that the first row gives the most-used covariate. Those marked with an asterisk ($\star$) were identified as being of primary interpretational interest prior to this screening, though they are also used by the standard BART model. The "Source" column indicates whether the covariate arises from a student questionnaire or from a questionnaire completed by the school principal. Notably, no covariates sourced from surveys of teachers are included.

Covariate	**Label**	**Source**
BSBG10	How often student is absent from school	Student
BSBM26AA	How often teacher gives homework	Student
BSBM17A	Does the student know what the teacher expects them to do?	Student
BSBG11B	How often student feels hungry	Student
BSBM20I	Does the student think it is important to do well in mathematics?	Student
BSBG05A	Does the student have a tablet or a computer at home?	Student
BCBG13BC	Does the school have library resources?	Principal
BSBG13D	Does the student think that teachers are fair in their school?	Student
BCBG13AD	Does the school have heating systems?	Principal
BCBG06C	How many instructional days in one calendar week does the school have?	Principal
BSBG05I	Country-specific indicator of wealth	Student
BSBM19A	Does the student do usually well in mathematics?	Student
BSBM42BA$^{(\star)}$	Minutes spent on homework	Student
BSBM43BA	For how many of the last 12 months has the student attended extra lessons or tutoring in mathematics?	Student
BCDGDAS$^{(\star)}$	Does the school have discipline problems?	Principal
BSBG11A	How often does the student feel tired?	Student
BCBG13AB	Does the school have shortage of supplies?	Principal
BCDGMRS	Are the instructions affected by the material resources shortage?	Principal
BSDGEDUP$^{(\star)}$	Parents' highest education level	Student
ITSEX	Sex of student	Student

CHAPTER 6

Conclusions

In this thesis, we have introduced a set of extensions for overcoming some key limitations of Bayesian additive regression trees (BART) and some of its semi-parametric versions thereof. BART is a non-parametric Bayesian tree-based model that automatically deals with linear, non-linear, and low-order interaction effects without requiring pre-specification. Specifically, such a model utilises a constant as the node parameter, which brings difficulty in approximating smooth effects. Unlike BART where the focus is more on the predictive accuracy, semi-parametric BART models combine a linear component along with a BART model in order to provide a greater level of interpretability for covariates of primary interpretational interest. Previously proposed models assumed that the parametric and non-parametric components cannot share covariates, which prevents important interactions from being estimated. In this final Chapter, we briefly revisit the models proposed in Chapters 3–5 by discussing some of their limitations and also pointing out some similarities.

In Chapter 3, we extended BART by considering observation-specific predictions at the terminal node level within the BART framework to address the issue related to the difficulty of approximating smooth effects. Instead of estimating one constant as the node parameter for each terminal node, a local Bayesian linear regression is considered where the covariates in the linear predictors are chosen based on the tree structure. Under the new formulation, smooth effects are captured more efficiently, while the recommended number of trees required to predict the response variable is drastically reduced. Through a simulation study based on the Friedman data (Friedman, 1991), which contains linear and non-linear effects, we show that MOTR-BART consistently outperforms BART and other tree-based competitors, except SBART, regardless of the sample size and number of noise covariates. We also explore the performance of MOTR-BART on well-known real-world applications, with MOTR-BART presenting the lowest or the second lowest out-of-sample root mean squared errors on almost all datasets considered.

One of the key points of MOTR-BART is related to the specification of the local linear predictors. We specified them based on the structure of each tree, either considering all covariates used to form the splitting rules or only those that are ancestors of the terminal nodes. In the current framework and implementation, covariates with coefficients close to zero are kept in the linear predictors since there is no procedure to actually select the variables; i.e., the linear predictors are specified given the topology of the trees but no variable selection is performed. In this sense, a future work would be to consider some variable selection procedures, such as spike-and-slab priors (George and McCulloch, 1993; Ishwaran and Rao, 2005), in order to keep only the covariates that are important and zero out coefficients associated with variables with little or no importance.

In Chapter 4, we proposed a new class of models for the estimation of genotype by environment (GxE) interactions in plant-based genetics. In particular, we extended the AMMI model by replacing its bilinear component, which is responsible for estimating the interactions between genotypes and environments, with BART. Furthermore, we modified the tree-generation process by introducing the double grow and double prune moves in the BART model so that it exclusively induces interactions between genotypes and environments while their marginal effects are estimated in a linear predictor. Through simulation experiments and a novel dataset from value of cultivation and usage experiments from the Irish Department of Agriculture, we showed that the performance of AMBARTI is competitive or superior when compared to AMMI, Bayesian AMMI, and other interaction detection methods based on out-of-sample root mean square error.

Along with the new model, we presented new visualisations that ease the interpretation of the marginal and interaction effects, which are suitable not only for AMBARTI, but for AMMI and Bayesian AMMI. The motivation for the new plots was the commonly used biplots. These plots are largely utilised in the determination of GxE interactions based on the results of the AMMI model, but they disregard the marginal effects since they display the interaction effects only. To this end, we created a heatmap and a bipartite network-style plot that take into account both the marginal and interaction effects, thus providing the data analyst with an additional and yet important source of information that can lead to better cultivar recommendation.

One of the challenges of AMBARTI is due to the high number of possible combinations between genotypes and between environments. In terms of computational implementation, all combinations must be available so that the BART component can select one combination of genotypes out of 2^{I-1} and one combination of 2^{J-1} to form the splitting rules, where I and J denote the number of genotypes and environments, respectively. In the simulation experiments in Chapter 4, we explored scenarios where $I = J = 10$ and $I = J = 25$, with the latter scenario posing a special difficulty as the number of possible combinations is very large and combinations containing high number of genotypes and/or environments tended to be rejected. Although it is not possible to reduce the number 2^{I-1}, an interesting research avenue would be to explore ways to propose combinations favouring low-order combinations in an attempt to improve the AMBARTI results when the number of genotypes/environments is large.

In Chapter 5, we proposed some extensions to semi-parametric models based on BART. The rationale of these models is to combine a linear predictor, where covariates of primary interpretational interest are specified, and a BART model, which deals with interactions and non-linearities among covariates of non-primary interpretational interest, an in attempt to elucidate the marginal effect that the covariates of primary interest have on the response. However, these models assume that the covariates in the linear predictor cannot be part of the covariates used by the BART component in order to avoid some undesirable properties for the estimates in the linear predictor, such as poor coverage, bias, and non-identifiability of the marginal effects.

To circumvent this assumption, we changed the tree-generation moves in BART to include the aforementioned double grow and double prune moves along with some stringent checks on the tree structure validity. In doing so, the BART component of our combined semi-parametric BART (CSP-BART) no longer assumes that the subsets used by the parametric (linear predictor) and non-parametric (BART) components are mutually exclusive, which in turn allows CSP-BART to capture interactions among the covariates of primary interest and between the covariates of primary and non-primary interest. In the analysis of the Trends in International Mathematics and Science Study (TIMSS) data, the benefits of sharing covariates across the components are highlighted by CSP-BART's predictive performance over its competitors. Furthermore, through two simulation experiments, we showed that the CSP-BART is able to estimate marginal effects of interest with low bias, while non-specified interactions and non-linearities are automatically accounted for.

The CSP-BART proposed in Chapter 5 can be used in regression and binary classification settings. However, it could be further generalised to deal with polychotomous response data by using the data augmentation strategies in Kindo et al. (2016b) and Albert and Chib (1993), or even extended to deal with multivariate skewed data (Um, 2021). Inspired by the recent theoretical developments on Bayesian forests (Ročková and Saha, 2019; Linero and Yang, 2018; Jeong and Ročková, 2020), another future work would be to study the optimal posterior concentration of CSP-BART. In addition, other BART extensions, such as SBART (Linero and Yang, 2018), log-linear BART (Murray, 2021), and BART for hurdle data, could be extended to a semi-parametric framework similar to the one presented in this work.

The main difference between the AMBARTI model proposed in Chapter 4 and the CSP-BART introduced in Chapter 5 is the cut in the feedback between the parametric and non-parametric components. In AMBARTI, the marginal effects of genotypes and environments are estimated taking into account the response variable only. In addition, the only marginal effects are those associated with genotypes and environments represented via dummy variables, while the subset used by the BART component contains dummy variables also associated with genotypes and environments only. In CSP-BART, however, the marginal effects of the covariates of primary interpretational interest are quantified considering the full residual obtained from the difference between the response variable and the BART prediction. Also, the covariates can be of any type, whether they be of primary or non-primary interest.

The main reason to cut the feedback between the model's components in AMBARTI was to keep the structure of the original AMMI model, which estimates the marginal and interaction effects separately without feeding back one into the other. For instance, the Bayesian version of the AMMI model (B-AMMI) proposed by Josse et al. (2014) estimates the interaction and marginal effects of genotypes and environments by feeding back one component into the other and then applying some orthonormalities constraints after the model is fitted. However, the simulation experiments and real-world data analysis carried out in Sections 4.3 and 4.4 indicated that B-AMMI struggles to estimate the marginal and interaction effects throughout.

In addition to the work presented in this thesis, we explored other venues in terms of extensions to the BART model that were not successfully satisfactory. For instance, we explored the normal inverse Gaussian (NIG; Barndorff-Nielsen, 1978, 1997) as the underlying distribution associated with the response variable. The NIG distribution is defined on the real line and can be skewed and heavy-tailed, which motivated applications in risk management (Eriksson et al., 2009), stock market (Karlis, 2002), model-based clustering (O'Hagan et al., 2016), to list a few. The distribution was initially proposed in the context of Brownian motion and is obtained by assuming that the variance of the normal distribution follows an inverse Gaussian distribution in the form of $y_i \sim \text{N}(\mu + \beta z_i)$, where $\beta \in \mathbb{R}$ and $z_i \sim \text{Inverse Gaussian}(\delta, \gamma)$, with $\delta, \gamma > 0$.

This work would be in line with the works of Linero et al. (2020) and Murray (2021) who explored other continuous and count distributions for the response variable for BART. However, the NIG extension was not possible as our implemented version of the approach failed to converge satisfactorily to the posterior distribution. In particular, we noticed that the β parameter — which controls the skewness — struggled to converge due to some large values sampled from the posterior distribution of the z_i's.

Finally, we remark that the implementation of the proposed methods presented in this work are freely available at `https://github.com/ebprado` in the repositories named `MOTR-BART`, `AMBARTI`, and `CSP-BART`, for Chapters 3, 4 and 5, respectively. Thus, all analyses in this thesis are reproducible and methodologies are available to interested practitioners. We emphasise that our implementations are fully in R and have not been optimised, as per some highly-optimised R packages like `dbarts` and `BART` which are both implemented in C++. Hence, improvements such as caching some summary statistics to avoid repeated computations and some minor design changes could significantly speed up our implementations and encourage wider use of our BART extensions proposed in this thesis.

Bibliography

Abu-Nimeh, S., Nappa, D., Wang, X., and Nair, S. (2008). Bayesian additive regression trees-based spam detection for enhanced email privacy. In *2008 Third International Conference on Availability, Reliability and Security*, pages 1044–1051. 2

Albert, J. H. and Chib, S. (1993). Bayesian analysis of binary and polychotomous response data. *Journal of the American Statistical Association*, 88(422):669–679. 19, 42, 69, 134, 146

Allard, R. and Bradshaw, A. (1992). Implications of genotype environmental interactions in applied plant breeding. *Crop Science*, 4:503–508. 62

Anbessa, Y., Juskiw, P., Good, A., Nyachiro, J., and Helm, J. (2009). Genetic variability in nitrogen use efficiency of spring barley. *Crop Science*, 49(4):1259–1269. 67

Athey, S., Tibshirani, J., and Wager, S. (2019). Generalized random forests. *The Annals of Statistics*, 47(2):1148–1178. 33

Badu-Apraku, B., Oyekunle, M., Obeng-Antwi, K., Osuman, A., Ado, S., Coulibay, N., Yallou, C., Abdulai, M., Boakyewaa, G., and Didjeira, A. (2012). Performance of extra-early maize cultivars based on GGE biplot and AMMI analysis. *The Journal of Agricultural Science*, 150(4):473. 67

Barndorff-Nielsen, O. (1978). Hyperbolic distributions and distributions on hyperbolae. *Scandinavian Journal of statistics*, pages 151–157. 147

Barndorff-Nielsen, O. (1997). Normal inverse Gaussian distributions and stochastic volatility modelling. *Scandinavian Journal of statistics*, 24(1):1–13. 147

Basak, P., Linero, A., Sinha, D., and Lipsitz, S. (2021). Semiparametric analysis of clustered interval-censored survival data using soft Bayesian additive regression trees (SBART). *Biometrics*. 69

Basford, K., Kroonenberg, P., and DeLacy, I. (1991). Three-way methods for multiattribute genotype-by-environment data: an illustrated partial survey. *Field Crops Research*, 27(1-2):131–157. 94

Bates, D., Mächler, M., Bolker, B., and Walker, S. (2015). Fitting linear mixed-effects models using lme4. *Journal of Statistical Software*, 67(1):1–48. 122

Bayarri, M., Berger, J., and Liu, F. (2009). Modularization in Bayesian analysis, with emphasis on analysis of computer models. *Bayesian Analysis*, 4(1):119–150. 63, 70, 73, 75

Blake, C. (1998). UCI repository of machine learning databases. *http://www.ics.uci.edu/m̃learn/MLRepository.html*. 112, 133

Bonato, V., Baladandayuthapani, V., Broom, B. M., Sulman, E. P., Aldape, K. D., and Do, K.-A. (2011). Bayesian ensemble methods for survival prediction in gene expression data. *Bioinformatics*, 27(3):359–367. 2

Brancourt-Hulmel, M. and Lecomte, C. (2003). Effect of environmental variates on genotype-by-environment interaction of winter wheat: A comparison of bi-additive factorial regression to AMMI. *Crop Science*, 43(2):608–617. 67

Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140. 2

Breiman, L. (2001). Random forests. *Machine learning*, 45(1):5–32. 1, 30, 109, 113

Breiman, L., Friedman, J., Stone, C., and Olshen, R. (1984). *Classification and Regression Trees*. Chapman and Hall/CRC. 1, 9

Brooks, S., Gelman, A., Jones, G., and Meng, X. (2011). *Handbook of Markov Chain Monte Carlo*. CRC press, New York, USA. 2, 115

Carvalho, C. M., Polson, N. G., and Scott, J. G. (2010). The horseshoe estimator for sparse signals. *Biometrika*, 97(2):465–480. 42

Castillo, I. and Ročková, V. (2019). Multiscale analysis of Bayesian CART. *University of Chicago, Becker Friedman Institute for Economics Working Paper*, 1(2019-127). 16

Chipman, H. A., George, E. I., and McCulloch, R. E. (1998). Bayesian CART model search. *Journal of the American Statistical Association*, 93(443):935–948. 1, 3, 4, 10, 11, 13, 14, 15, 17, 34, 35

Chipman, H. A., George, E. I., and McCulloch, R. E. (2002). Bayesian treed models. *Machine Learning*, 48(1):299–320. 4, 21, 34

Chipman, H. A., George, E. I., and McCulloch, R. E. (2010). BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298. 2, 15, 17, 19, 21, 30, 34, 36, 42, 44, 45, 51, 63, 67, 68, 70, 109, 113, 114, 117, 118, 123

Chipman, H. A., George, E. I., and McCulloch, R. E. (2013). Bayesian regression structure recovery. In Damien, P., Dellaportas, P., Polson, N. G., and Stephens, D. A., editors, *Bayesian Theory and Applications*, chapter 22, pages 451–465. Oxford University Press. 133

Chipman, H. A., George, E. I., McCulloch, R. E., and Shively, T. S. (2021). mBART: Multidimensional Monotone BART. *Bayesian Analysis*, 1(1):1–30. 113

Clark, T. E., Huber, F., Koop, G., Marcellino, M., and Pfarrhofer, M. (2021). Tail forecasting with multivariate Bayesian additive regression trees. *Federal Reserve Bank of Cleveland, Working Paper No. 21-08.* 2

Crossa, J., Perez-Elizalde, S., Jarquin, D., Cotes, J. M., Viele, K., Liu, G., and Cornelius, P. L. (2011). Bayesian estimation of the additive main effects and multiplicative interaction model. *Crop Science*, 51(4):1458–1469. 74, 84

de Mendiburu, F. (2019). Package 'agricolae'. *R Package, Version*, pages 1–2. 66

Denison, D. G., Mallick, B. K., and Smith, A. F. (1998). Bayesian MARS. *Statistics and Computing*, 8(4):337–346. 1, 4, 10, 14, 82

Deshpande, S. K., Bai, R., Balocchi, C., Starling, J. E., and Weiss, J. (2020). VCBART: Bayesian trees for varying coefficients. *arXiv preprint arXiv:2003.06416*. 2, 32, 33, 109, 112, 122, 123

Dias, C. (2005). *Métodos para escolha de componentes em modelo de efeito principal aditivo e interação multiplicativa (AMMI). 2005. 73p.* PhD thesis, Tese (Livre Docência)–Escola Superior de Agricultura Luiz de Queiroz, Piracicaba. 64

Dias, C. T. d. S. and Krzanowski, W. J. (2006). Choosing components in the additive main effect and multiplicative interaction (AMMI) models. *Scientia Agricola*, 63(2):169–175. 67

Dorie, V. (2020). dbarts: Discrete Bayesian additive regression trees sampler. R package version 0.9-19. 3, 30, 43, 72, 87, 116

Eriksson, A., Ghysels, E., and Wang, F. (2009). The Normal inverse Gaussian distribution and the pricing of derivatives. *The Journal of Derivatives*, 16(3):23–37. 147

Falconer, D. and Mackay, T. (1996). Introduction to quantitative genetics. 1996. *Harlow, Essex, UK: Longmans Green*, 3. 64

Farshadfar, E. and Sutka, J. (2003). Locating QTLs controlling adaptation in wheat using AMMI model. *Cereal Research Communications*, 31(3):249–256. 67

Fishbein, B., Foy, P., and Yin, L. (2021). TIMSS 2019 user guide for the international database. 110, 127

Foy, P. (2017). *TIMSS 2015 User Guide for the International Database*. TIMSS & PIRLS International Study Center, Lynch School of Education, Boston College and International Association for the Evaluation of Educational Achievement (IEA). 140

Francom, D. and Sansó, B. (2020). BASS: An R package for fitting and performing sensitivity analysis of Bayesian adaptive spline surfaces. *Journal of Statistical Software*, 94(8):1–36. 82

Freund, Y. and Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of computer and system sciences*, 55(1):119–139. 2

Friedberg, R., Tibshirani, J., Athey, S., and Wager, S. (2020). Local linear forests. *Journal of Computational and Graphical Statistics*, 30(2):503–517. 3, 33

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1. 43

Friedman, J. H. (1991). Multivariate adaptive regression splines. *The Annals of statistics*, pages 1–67. 5, 44, 123, 144

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232. 1, 30, 113

Gabriel, K. R. (1971). The biplot graphic display of matrices with application to principal component analysis. *Biometrika*, 58(3):453–467. 25, 87

Gamerman, D. and Lopes, H. F. (2006). *Markov chain Monte Carlo: stochastic simulation for Bayesian inference.* CRC Press. 68

Gauch Jr, H. G. (2013). A simple protocol for AMMI analysis of yield trials. *Crop Science*, 53(5):1860–1869. 66

Gelfand, A. E. and Smith, A. F. (1990). Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85(410):398–409. 30

George, E. I. and McCulloch, R. E. (1993). Variable selection via gibbs sampling. *Journal of the American Statistical Association*, 88(423):881–889. 144

George, E. I. and McCulloch, R. E. (1997). Approaches for Bayesian variable selection. *Statistica sinica*, pages 339–373. 51, 94

Gollob, H. F. (1968). A statistical model which combines features of factor analytic and analysis of variance techniques. *Psychometrika*, 33(1):73–115. 65

Good, I. J. (1969). Some applications of the singular decomposition of a matrix. *Technometrics*, 11(4):823–831. 66

Goodman, L. A. and Haberman, S. J. (1990). The analysis of nonadditivity in two-way analysis of variance. *Journal of the American Statistical Association*, 85(409):139–145. 84, 85, 86

Green, D. P. and Kern, H. L. (2012). Modeling heterogeneous treatment effects in survey experiments with Bayesian additive regression trees. *Public opinion quarterly*, 76(3):491–511. 2, 31, 113

Green, P. J. (1995). Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika*, 82(4):711–732. 14

Greenwell, B., Boehmke, B., Cunningham, J., and Developers, G. (2019). *gbm: Generalized Boosted Regression Models*. R package version 2.1.5. 43

Grilli, L., Pennoni, F., Rampichini, C., and Romeo, I. (2016). Exploiting TIMSS and PIRLS combined data: multivariate multilevel modelling of student achievement. *The Annals of Applied Statistics*, 10(4):2405–2426. 128, 136

Gu, C. (2014). Smoothing spline ANOVA models: R package gss. *Journal of Statistical Software*, 58(5):1–25. 82

Guhaniyogi, R., Qamar, S., and Dunson, D. B. (2017). Bayesian tensor regression. *The Journal of Machine Learning Research*, 18(1):2733–2763. 94

Hahn, P. R., Murray, J. S., and Carvalho, C. M. (2020). Bayesian regression tree models for causal inference: regularization, confounding, and heterogeneous effects. *Bayesian Analysis*. 31, 109, 110, 112, 113, 128

Harezlak, J., Ruppert, D., and Wand, M. P. (2018). *Semiparametric regression with R*. Springer. 109

Harshman, R. A. and Lundy, M. E. (1994). PARAFAC: Parallel factor analysis. *Computational Statistics & Data Analysis*, 18(1):39–72. 94

Hastie, T. and Tibshirani, R. (1990). *Generalized additive models*. CRC press. 25, 109

Hastie, T. and Tibshirani, R. (1993). Varying-coefficient models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 55(4):757–779. 112

Hastie, T. and Tibshirani, R. (2000). Bayesian backfitting (with comments and a rejoinder by the authors). *Statistical Science*, 15(3):196–223. 2, 17, 68

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer. 3, 9, 10

Hastings, W. K. (1970). Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109. 12, 30

He, J., Yalov, S., and Hahn, P. R. (2019). XBART: Accelerated Bayesian additive regression trees. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1130–1138. 2, 32

Hernández, B., Pennington, S. R., and Parnell, A. C. (2015). Bayesian methods for proteomic biomarker development. *EuPA Open Proteomics*, 9:54–64. 31, 69, 113

Hernández, B., Raftery, A. E., Pennington, S. R., and Parnell, A. C. (2018). Bayesian additive regression trees using Bayesian model averaging. *Statistics and computing*, 28(4):869–890. 2, 32, 63, 69, 113

Hill, J. and Su, Y.-S. (2013). Assessing lack of common support in causal inference using Bayesian nonparametrics: Implications for evaluating the effect of breastfeeding on children's cognitive outcomes. *The Annals of Applied Statistics*, pages 1386–1420. 2

Hill, J. L. (2011). Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240. 2, 31, 110, 113

Ishwaran, H. and Rao, J. S. (2005). Spike and slab variable selection: frequentist and Bayesian strategies. *The Annals of Statistics*, 33(2):730–773. 51, 94, 144

Isik, F., Holland, J., and Maltecca, C. (2017). Multi environmental trials. In *Genetic data analysis for plant and animal breeding*, pages 227–262. Springer. 64

Jeong, S. and Ročková, V. (2020). The art of BART: On flexibility of Bayesian forests. *arXiv preprint arXiv:2008.06620*. 3, 69, 146

Josse, J., van Eeuwijk, F., Piepho, H.-P., and Denis, J.-B. (2014). Another look at Bayesian analysis of AMMI models for genotype-environment data. *Journal of Agricultural, Biological, and Environmental Statistics*, 19(2):240–257. 74, 78, 84, 102, 147

Kapelner, A. and Bleich, J. (2016). bartMachine: Machine learning with Bayesian additive regression trees. *Journal of Statistical Software, Articles*, 70(4):1–40. 3, 13, 30, 48, 72, 78, 116, 133

Karlis, D. (2002). An EM type algorithm for maximum likelihood estimation of the Normal–inverse gaussian distribution. *Statistics & probability letters*, 57(1):43–52. 147

Kindo, B. P., Wang, H., Hanson, T., and Peña, E. A. (2016a). Bayesian quantile additive regression trees. *arXiv preprint arXiv:1607.02676*. 2, 32

Kindo, B. P., Wang, H., and Peña, E. A. (2016b). Multinomial probit Bayesian additive regression trees. *Stat*, 5(1):119–131. 2, 19, 32, 69, 113, 146

Künzel, S. R., Saarinen, T. F., Liu, E. W., and Sekhon, J. S. (2022). Linear aggregation in tree-based estimators. *Journal of Computational and Graphical Statistics*, pages 1–18. 3, 33, 44

Lal, R., Chanotiya, C., Dhawan, S., Gupta, P., Mishra, A., Srivastava, S., Shukla, S., and Maurya, R. (2020). Estimation of intra-specific genetic variability and half-sib family selection using AMMI model in menthol mint (*Mentha arvensis L.*). *J. Med. Arom. Plant Sci*, 42(1-2):102–113. 67

Landwehr, N., Hall, M., and Frank, E. (2005). Logistic model trees. *Machine learning*, 59(1-2):161–205. 33, 51

Leisch, F., Dimitriadou, E., Leisch, M. F., and No, Z. (2009). Package 'mlbench'. *CRAN*. 133

Linero, A. (2017a). *SoftBart: A package for implementing the SoftBart algorithm.* R package version 1.0. 43

Linero, A. R. (2017b). A review of tree-based bayesian methods. *Communications for Statistical Applications and Methods*, 24(6). 32

Linero, A. R. (2018). Bayesian regression trees for high-dimensional prediction and variable selection. *Journal of the American Statistical Association*, 113(522):626–636. 20, 21, 32, 42, 44, 69, 113, 118, 123, 136

Linero, A. R., Basak, P., Li, Y., and Sinha, D. (2021). Bayesian survival tree ensembles with submodel shrinkage. *Bayesian Analysis*, 1(1):1–24. 69, 113

Linero, A. R., Sinha, D., and Lipsitz, S. R. (2020). Semiparametric mixed-scale models using shared Bayesian forests. *Biometrics*, 76(1):131–144. 2, 32, 52, 113, 136, 147

Linero, A. R. and Yang, Y. (2018). Bayesian regression tree ensembles that adapt to smoothness and sparsity. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(5):1087–1110. 2, 3, 32, 51, 113, 146

Liu, Y., Traskin, M., Lorch, S. A., George, E. I., and Small, D. (2015). Ensemble of trees approaches to risk adjustment for evaluating a hospital's performance. *Health care management science*, 18(1):58–66. 2, 63

Love, S. L., Salaiz, T., Shafii, B., Price, W. J., Mosley, A. R., and Thornton, R. E. (2004). Stability of expression and concentration of ascorbic acid in North American potato germplasm. *HortScience*, 39(1):156–160. 67

Mahalingam, L., Mahendran, S., Babu, R. C., and Atlin, G. (2006). AMMI analysis for stability of grain yield in rice (*Oryza sativa L.*). *International Journal of Botany*. 67

Mandel, J. (1971). A new analysis of variance model for non-additive data. *Technometrics*, 13(1):1–18. 5, 23, 63, 65

Martin, M. O., Mullis, I. V., Gregory, K. D., Hoyle, C., and Shen, C. (2000). Effective schools in science and mathematics. *IEA's third international mathematics and science study, IEA: Chestnut Hill, MA*. 128

McCullagh, P. and Nelder, J. (1989). *Generalized Linear Models, Second Edition*. Chapman and Hall/CRC Monographs on Statistics and Applied Probability Series. Chapman & Hall. 6, 109

McCulloch, R., Sparapani, R., Spanbauer, C., Gramacy, R., and Pratola, M. (2020). *BART: Bayesian Additive Regression Trees*. R package version 2.8. 3, 30, 72, 87, 116

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092. 12, 30

Mitroviaã, B., Treski, S., Stojakkovã, M., Ivanoviã, M., and Bekavac, G. (2012). Evaluation of experimental maize hybrids tested in multi-location trials using AMMI and GGE biplot analyses. *Turkish Journal of Field Crops*, 17(1):35–40. 67

Mohammadpour, E., Shekarchizadeh, A., and Kalantarrashidi, S. A. (2015). Multilevel modeling of science achievement in the TIMSS participating countries. *The Journal of Educational Research*, 108(6):449–464. 128, 136

Müller, P. (1991). *A generic approach to posterior integration and Gibbs sampling*. Purdue University, Department of Statistics. 17

Müller, P. (1992). *Alternatives to the Gibbs sampling scheme*. Citeseer. 17

Mullis, I., Martin, M., Foy, P., Kelly, D., and Fishbein, B. (2020). TIMSS 2019 international results in mathematics and science. 110, 127

Murray, J. S. (2021). Log-linear Bayesian additive regression trees for multinomial logistic and count regression models. *Journal of the American Statistical Association*, 116(534):756–769. 2, 19, 32, 51, 113, 136, 146, 147

Nachit, M., Nachit, G., Ketata, H., Gauch, H., and Zobel, R. (1992). Use of AMMI and linear regression models to analyze genotype-environment interaction in durum wheat. *Theoretical and Applied genetics*, 83(5):597–601. 67

Nelder, J. A. and Wedderburn, R. W. (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, 135(3):370–384. 6, 109

Onofri, A. and Ciriciofolo, E. (2007). Using R to perform the AMMI analysis on agriculture variety trials. *R News*, 7(1):14–19. 66

Orlandi, V., Murray, J., Linero, A., and Volfovsky, A. (2021). Density regression with Bayesian additive regression trees. *arXiv preprint arXiv:2112.12259*. 2

O'Hagan, A., Murphy, T. B., Gormley, I. C., McNicholas, P. D., and Karlis, D. (2016). Clustering with the multivariate Normal inverse Gaussian distribution. *Computational Statistics & Data Analysis*, 93:18–30. 147

Plummer, M. (2003). JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling. In *Proceedings of the 3rd international workshop on distributed statistical computing*, volume 124. Vienna, Austria. 75

Plummer, M. (2015). Cuts in Bayesian graphical models. *Statistics and Computing*, 25(1):37–43. 24, 63, 70, 73, 75, 76, 77, 99, 100, 102

Prado, E. B., Moral, R. A., and Parnell, A. C. (2021). Bayesian additive regression trees with model trees. *Statistics and Computing*, 31(3):1–13. 69, 113, 116

Pratola, M. T., Chipman, H. A., George, E. I., and McCulloch, R. E. (2020). Heteroscedastic BART via multiplicative regression trees. *Journal of Computational and Graphical Statistics*, 29(2):405–417. 2, 32, 113

Prüser, J. (2019). Forecasting with many predictors using Bayesian additive regression trees. *Journal of Forecasting*, 38(7):621–631. 2

Quinlan, J. R. (1992). Learning with continuous classes. In *5th Australian joint conference on artificial intelligence*, volume 92, pages 343–348. World Scientific. 30, 33

R Core Team (2020). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. 3, 6, 43, 66, 122

Rad, M. N., Kadir, M. A., Rafii, M., Jaafar, H. Z., Naghavi, M., and Ahmadi, F. (2013). Genotype-by-environment interaction by AMMI and GGE biplot analysis in three consecutive generations of wheat (*Triticum aestivum*) under normal and drought stress conditions. *Australian Journal of Crop Science*, 7(7):956. 67

Robert, C. and Casella, G. (2013). *Monte Carlo statistical methods.* Springer Science & Business Media. 68

Ročková, V. and Saha, E. (2019). On theory for BART. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2839–2848. PMLR. 3, 32, 69, 113, 146

Ročková, V. and van der Pas, S. (2020). Posterior concentration for Bayesian regression trees and forests. *Annals of Statistics*, 48(4):2108–2131. 3, 32, 69, 113

Rodrigues, P. C., Monteiro, A., and Lourenço, V. M. (2016). A robust AMMI model for the analysis of genotype-by-environment data. *Bioinformatics*, 32(1):58–66. 63

Romagosa, I., Ullrich, S. E., Han, F., and Hayes, P. M. (1996). Use of the additive main effects and multiplicative interaction model in QTL mapping for adaptation in barley. *Theoretical and Applied Genetics*, 93(1-2):30–37. 67

Rutkowski, L., Gonzalez, E., Joncas, M., and von Davier, M. (2010). International large-scale assessment data: Issues in secondary analysis and reporting. *Educational researcher*, 39(2):142–151. 140

Sarti, D. A. (2013). *Gerenciamento de incertezas por análise de decisões: aplicações à otimização da produção e demandas incertas.* PhD thesis, Universidade de São Paulo. 62, 65

Sarti, D. A. (2019). *The statistical paradigm: probabilistic and multivariate analysis applied through computational simulation in the interaction between genotype x environment.* PhD thesis, Universidade de São Paulo. 63, 65, 67

Sarti, D. A., Prado, E. B., Inglis, A., dos Santos, A. A. L., Hurley, C., de Andrade Moral, R., and Parnell, A. (2023). Bayesian additive regression trees for genotype by environment interaction models. *The Annals of Applied Statistics*, 17(1). 113

Sato, K. and Takeda, K. (1993). Pathogenic variation of *Pyrenophora teres* isolates collected from Japanese and Canadian spring barley. *Report by the Institute of Resource Biological Sciences, Okayama University*, 1(2):147–158. 67

Schnell, P. M., Tang, Q., Offen, W. W., and Carlin, B. P. (2016). A Bayesian credible subgroups approach to identifying patient subgroups with positive treatment effects. *Biometrics*, 72(4):1026–1036. 31

Shafii, B. and Price, W. J. (1998). Analysis of genotype-by-environment interaction using the additive main effects and multiplicative interaction model and stability estimates. *Journal of Agricultural, Biological, and Environmental Statistics*, pages 335–345. 67

Silveira, L. C. I. d., Kist, V., Paula, T. O. M. d., Barbosa, M. H. P., Peternelli, L. A., and Daros, E. (2013). AMMI analysis to evaluate the adaptability and phenotypic stability of sugarcane genotypes. *Scientia Agricola*, 70(1):27–32. 67

Sivaganesan, S., Müller, P., and Huang, B. (2017). Subgroup finding via Bayesian additive regression trees. *Statistics in medicine*, 36(15):2391–2403. 31

Sparapani, R., Logan, B. R., McCulloch, R. E., and Laud, P. W. (2019). Nonparametric competing risks analysis using Bayesian additive regression trees. *Statistical methods in medical research*, page 0962280218822140. 31, 113

Sparapani, R. A., Logan, B. R., McCulloch, R. E., and Laud, P. W. (2016). Nonparametric survival analysis using Bayesian additive regression trees (BART). *Statistics in medicine*, 35(16):2741–2753. 31, 69, 113

Starling, J. E., Aiken, C. E., Murray, J. S., Nakimuli, A., and Scott, J. G. (2019). Monotone function estimation in the presence of extreme data coarsening: Analysis of preeclampsia and birth weight in urban Uganda. *arXiv preprint arXiv:1912.06946*. 2, 32

Starling, J. E., Murray, J. S., Carvalho, C. M., Bukowski, R. K., and Scott, J. G. (2020). BART with targeted smoothing: An analysis of patient-specific stillbirth risk. *Annals of Applied Statistics*, 14(1):28–50. 2, 32, 113

Suk, Y., Kim, J.-S., and Kang, H. (2021). Hybridizing machine learning methods and finite mixture models for estimating heterogeneous treatment effects in latent classes. *Journal of Educational and Behavioral Statistics*, 46(3):323–347. 113

Tan, Y. V. and Roy, J. (2019). Bayesian additive regression trees and the General BART model. *Statistics in medicine*, 38(25):5048–5069. 2, 3, 25, 69, 109

Tibshirani, J., Athey, S., and Wager, S. (2020). *grf: Generalized Random Forests*. R package version 1.2.0. 43

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, 58(1):267–288. 42

Tyagi, B., Singh, M., Singh, G., Kumar, R., Verma, A., and Sharma, I. (2016). Genetic variability and AMMI bi-plot analysis in bread wheat based on multi-location trials conducted under drought conditions across agro-climatic zones of India. *Triticeae Genomics and Genetics*, 7. 67

Um, S. (2021). *Bayesian Additive Regression Trees for Multivariate Responses*. PhD thesis, The Florida State University. 2, 146

Wang, Y., Witten, I., van Someren, M., and Widmer, G. (1997). Inducing models trees for continuous classes. In *Proceedings of the Poster Papers of the European Conference on Machine Learning, Department of Computer Science, University of Waikato, New Zeland*. 33

Wood, S. N. (2017). *Generalized additive models: an introduction with R*. CRC press. 25, 109, 122

Wright, M. N. and König, I. R. (2019). Splitting on categorical predictors in random forests. *PeerJ*, 7:e6339. 72

Wright, M. N. and Ziegler, A. (2017). ranger: A fast implementation of random forests for high dimensional data in C++ and R. *Journal of Statistical Software*, 77(1):1–17. 43

Zeldow, B., Re III, V. L., and Roy, J. (2019). A semiparametric modeling approach using Bayesian additive regression trees with an application to evaluate heterogeneous treatment effects. *The Annals of Applied Statistics*, 13(3):1989. 2, 3, 25, 70, 109, 110, 111, 116, 117, 122, 130, 138

Zhang, J. L. and Härdle, W. K. (2010). The Bayesian additive classification tree applied to credit risk modelling. *Computational Statistics & Data Analysis*, 54(5):1197–1205. 2, 31, 63, 69, 113