Generalizations of BART-BMA and BART-IS

Eoghan O'Neill*
Econometric Institute
Erasmus University Rotterdam

August 5, 2021

Abstract

This chapter outlines extensions of the BART-BMA and BART-IS algorithms to more general settings, including binary outcomes, treatment effects for binary outcomes, censored outcomes, categorical outcomes, and count data. BART-IS and BART-BMA are readily extendable to model frameworks for which the marginal likelihood and posterior can be efficiently calculated or approximated. The examples discussed in this chapter make use of standard Quasi-Newton methods in combination with Laplace approximations.

As examples of how to apply the general approach, Logit-BART-BMA and Logit-BART-IS are described and shown to be competitive with existing tree-based methods on real-world binary classification datasets. In addition, Logit-BCF-IS (and Logit-BCF-BMA) give treatment effect estimates and intervals with accuracy comparable to the best performing methods on simulated datasets from a data analysis challenge. As a further example, Tobit-BART is introduced and implemented using the general BART-IS framework.

Contents

*Email: oneill@ese.eur.nl

1 Introduction

1.1 BART for Generalized Linear Models

This chapter outlines how BART-BMA and BART-IS can be generalized to a variety of data settings, including binary outcomes, censored outcomes, count data, and multinomial response data. The general approach

is applicable in settings in which a linear combination of variables can be replaced by a sum-of-tree model. As explained in chapter 2 of this thesis, a sum-of-trees is itself a representation of a linear combination of indicator variables for terminal nodes with coefficients equal to the terminal node mean parameters. This approach allows for non-linearity and complex interactions between variables, while also accounting for model uncertainty.

Recent advances in Bayesian methods have allowed for a large class of models to be approximated efficiently. The methods introduced in this paper are averages over generalized linear models with the linear combinations of covariates replaced by sums-of-trees. The approach may be applicable to a wider class of models, but this chapter will restrict attention to generalized linear models.

While the general algorithms are not restricted to a particular approximation method, a key candidate method that will be focused on in this chapter is Laplace approximation. Rue et al. (2009) introduce Integrated Nested Laplace Approximations (INLA), which are applicable to latent Gaussian models. Most structured Bayesian models take the form of latent Gaussian models, which are a special case of structured additive regression models.

In structured additive regression models, the outcome y_i is assumed to belong to an exponential family, where the mean μ_i is linked to a structured additive predictor η_i through a link-function $g(.)$, so that $g(\mu_i) = \eta_i$. The structured additive predictor η_i takes the form:

$$\eta_i = \alpha + \sum_{j=1}^{n_f} f^{(j)}(u_{ji}) + \sum_{k=1}^{n_\beta} \beta_k z_{ki} + \varepsilon_i \tag{1}$$

where the $\{f^{(j)}(.)\}$'s are unknown functions of the covariates u, the $\{\beta_k\}$'s represent the linear effects of covariates z and the ε_i's are unstructured terms. Latent Gaussian models apply a Gaussian prior to $\{f^{(j)}(.)\}$, $\{\beta_k\}$, and ε_i.

This paper will focus on averages of generalized linear models, which are of the form $\eta_i = \alpha + \sum_{k=1}^{n_\beta} \beta_k z_{ki}$. The models being averaged over all use the same link function, and the linear combination of covariates $\sum_{k=1}^{n_\beta} \beta_k z_{ki}$ (or part of the linear combination) is replaced by a linear combination of indicator variables for inclusion in terminal nodes of sums-of-trees (as described in chapter 2).[1] Models that do not involve Gaussian priors may also be included in the general framework outlined in this paper, provided an efficient approximation is available.

The discussion above outlines the general applicability of the approach through the use of INLA (Rue et al. 2009). However, for simplicity of demonstration, this paper will focus on examples in which standard Laplace approximations are feasible, and therefore further description of INLA is omitted. Furthermore, the feasibility of generalization to a particular model depends on the computational speed of the approximation, and performance will depend on the accuracy of the approximation and the appropriateness of the link function.[2] The limited existing literature on the combination of INLA and Bayesian Model Averaging involves averaging over parameters in spatial econometric models (Gómez-Rubio et al. 2020, Gómez-Rubio & Rue 2018, Bivand et al. 2014, 2015). However, this literature does not discuss averaging over different sets of (non-linear functions of) covariates.

The General BART-IS algorithm involves random, data-independent draws of sum-of-tree models, and a marginal likelihood weighted average of these models. The General BART-BMA algorithm involves a model search algorithm that beings by constructing single tree models, and then appends trees to these models, and averages over the set of searched models that have highest posterior probability. The BART-BMA approach requires construction of residuals representing the unexplained part of η_i, which are used to construct trees to be appended to the models. However, the calculation of residuals, while straightforward in the case of Logit, is not always possible. In this sense BART-IS is more generalizable than BART-BMA, as BART-BMA requires model-specific adjustments to the model search algorithm.

[1] Models that include $f(.)$ terms such as random effects models $f(u_i) = f_i$, dynamic models $f(u_t) = f_t$, and spatial models $f(u_s) = f_s$, may also be included in the general approach described in this chapter, but these models are not the focus of this chapter.

[2] It could be argued that the link function imposes a strong assumption, and therefore a moment-condition based approach such as Generalized Random Forests (Athey et al. 2019) or Orthogonal Random Forests (Oprescu et al. 2018) is more appropriate in some contexts.

A key requirement for this approach to be feasible is that the marginal likelihood can be efficiently calculated and the posterior distribution has a closed form or has a very efficient sampler. This requirement is satisfied in the case of Bayesian logistic regression with a standard Laplace approximation, which is used in this chapter as an illustrative example. Logit-BART-BMA and Logit-BART-IS involve averaging over models in which the binary outcome has success probability equal to the logistic function of a sum-of-tree function.

1.2 Binary Classification Example and Literature Review

Single tree methods can readily be applied to binary outcome data. Trees in a random forest applied to binary outcomes produce predictions between zero and one because leaf estimates are averages of binary variables. However, sum-of-tree based methods such as BART are less directly applicable to binary outcome data because sums-of-trees can produce predictions outside the range $[0, 1]$ and ideally the statistical framework of BART should account for the fact that the outcomes are binary. Therefore BART-based models for binary outcomes (and other generalized linear models for different forms of outcome variable), rely on a choice of link function.

Sum-of-tree models, such as AdaBoost with decision trees as weak learners, often produce excellent results when applied to binary classification problems (Freund & Schapire 1995, Freund et al. 1996). An early example of a sum-of-tree model for binary outcomes placed in a statistical framework is the LogitBoost algorithm (Friedman et al. 2000). BART can be extended to binary outcome prediction by applying a probit or logit link function to a sum-of-tree model. Chipman et al. (2010) implement Probit-BART-MCMC using the data augmentation Markov Chain Monte Carlo approach of Albert & Chib (1993). Zhang & Härdle (2010) independently applied Probit-BART to credit risk modelling and found that it is competitive with other machine learning methods. Abu-Nimeh et al. (2008) also applied this approach to spam email detection. The **R** package **BART** implements Logit-BART using a computationally intensive MCMC algorithm based on the approach of Gramacy et al. (2012).

The performance of MCMC implementations of BART has been noted to be less impressive for binary outcomes than for continuous outcomes (Hill et al. 2020, Carnegie et al. 2015).[3] The algorithms in this paper provide alternatives to the MCMC implementations of BART for binary outcomes.

A number of recent papers have extended the applicability of BART. Examples include BART variations of multinomial Probit (Kindo, Wang & Peña 2016), quantile regression (Kindo, Wang, Hanson & Peña 2016), survival analysis (Sparapani et al. 2016), recurrent event analysis (Sparapani et al. 2018), and competing risks models (Sparapani et al. 2019). See Hill et al. (2020), Tan & Roy (2019) and Linero (2017) for review articles.

Murray (2017) proposes new priors and a data augmentation scheme that allow for an efficient MCMC sampler for BART-based methods outside the context of Gaussian models. The approach of Murray (2017) (Log-linear BART) is to model the log of the regression function as a sum-of-trees and apply a generalized inverse Gaussian prior distribution to the terminal node parameters. Log-linear BART is applicable to logistic regression, multinomial logistic regression, and Poisson regression among other models.

This paper provides alternatives to Log-linear BART that retain the standard BART priors and do not rely on MCMC. [4] [5] A BART-BMA framework provides efficient greedy algorithms that outputs a relatively small number of parsimonious models. A BART-IS framework is straightforward to implement and trivially parallelizable.[6] The simple BART-BMA and BART-IS approaches provide readily implementable benchmarks for more complicated schemes such as the MCMC-based methods.

The methods discussed in this chapter are relevant to a range of economic applications. Binary classification algorithms can be applied to propensity score estimation, and also prediction problems such as credit

[3] Dorie et al. (2019) note that performance can be improved by using cross-validation to choose hyperparameters.

[4] While the focus of this paper is implementation algorithms, I also provide options for alternative model priors on the tree structures, including the prior proposed by Quadrianto & Ghahramani (2014) and the spike and tree prior Rockova & van der Pas (2017), in the **R** packages **logitbartBMA** and **safeBart** . Code is available at https://github.com/EoghanONeill

[5] The approach introduced in this paper can be combined with alternative parameter priors, e.g. different terminal node priors and hierarchical priors, provided the marginal likelihood can be efficiently calculated and it is possible to sample efficiently from a given sum-of-tree model (in the set of models being averaged). This possibility is a topic for future research.

[6] See chapter 2 of this thesis for further discussion of the usefulness of the BART-BMA and BART-IS algorithms.

default prediction and prediction of consumer purchases. Multinomial regression methods are extensively used in modelling discrete choice problems in econometrics. The methods introduced in this chapter provide a flexible machine learning approach that accounts for model uncertainty and potentially complex functional forms.

The remainder of the paper is structured as follows. Section 2 provides a brief review of BART. Section 3 outlines the general framework for extending BART-BMA and BART-IS to a wide range of model settings. Section 4 describes the binary classification methods Logit-BART-IS and Logit-BART-BMA and compares the performance of these algorithms to other methods using publicly available datasets. Section 5 describes methods for treatment effect estimation with binary outcomes Logit-BCF-IS and Logit-BCF-BMA, and compares these algorithms to other methods using data from the ACIC 2019 data challenge. Section 6 discusses further model settings to which the generalized BART-BMA and BART-IS algorithms can be applied, with Tobit-BART-IS as an illustrative example.[7] Section 7 concludes the paper.

2 Review of BART and BART-BMA

In this section, we describe BART (Chipman et al. 2010), BART-BMA (Hernández et al. 2018), and an approximate, sub-optimal approach to implementation of Probit-BART-BMA and Probit-BART-IS that can be as a benchmark for the more principled approach introduced later in this paper.

This section repeats the overview from chapter 2, and is included for completeness so that this chapter is self-contained.

2.1 Overview of BART

2.1.1 Description of BART Model and Priors

Suppose there are n observations, and the $n \times p$ matrix of explanatory variables, X, has i^{th} row $x_i = [x_{i1}, ..., x_{ip}]$. For the standard BART model $Y_i = \sum_{j=1}^{m} g(x_i; T_j, M_j) + \varepsilon_i$, where $g(x_i; T_j, M_j)$ is the output of a decision tree. T_j refers to decision tree $j = 1, ..., m$, where m is the total number of trees in the model. M_j are the terminal node parameters of T_j, and $\varepsilon_i \overset{i.i.d}{\sim} N(0, \sigma^2)$.

For BART (Chipman et al. 2010), prior independence is assumed across trees T_j and across terminal node means $M_j = (\mu_{1j} ... \mu_{b_j j})$ (where $1, ..., b_j$ indexes the terminal nodes of tree j). The form of the prior used by Chipman et al. (2010) is:

$$p(M_1, ..., M_m, T_1, ..., T_m, \sigma) \propto \left[\prod_j \left[\prod_k p(\mu_{kj}|T_j) \right] p(T_j) \right] p(\sigma)$$

In standard BART, $\mu_{kj}|T_j \overset{i.i.d}{\sim} N(0, \sigma_0^2)$ where $\sigma_0 = \frac{0.5}{e\sqrt{m}}$ and e is a user-specified hyper-parameter.

Chipman et al. (2010) set a regularization prior on the tree size and shape $p(T_j)$ to discourage any one tree from having undue influence over the sum of trees. The probability that a given node within a tree T_j is split into two child nodes is $\alpha(1 + d_h)^{-\beta}$, where d_h is the depth of (internal) node h and α and β are parameters which determine the size and shape of T_j respectively. Thus $p(T_j) = \prod_{h=1}^{b_j - 1} \alpha(1+d_h)^{-\beta} \prod_{k=1}^{b_j} (1-\alpha(1+d_k)^{-\beta})$, where h indexes the internal nodes of the tree T_j, and k indexes the terminal nodes.

Chipman et al. (2010) assume that the model precision σ^{-2} has a conjugate prior distribution $\sigma^{-2} \sim Ga(\frac{v}{2}, \frac{v\lambda}{2})$ with degrees of freedom v and scale λ. There are also priors on the splitting variables and splitting points in each tree. Chipman et al. (2010) use the uniform prior on available splitting variables, and the uniform prior on the discrete set of available splitting variables.

[7] To the best of my knowledge, this is the first example of a Tobit-BART regardless of the implementation. An interesting topic for future research would be an MCMC based implementation of Tobit-BART.

2.2 Overview of BART-BMA

BART-BMA applies the same priors as standard BART (section 2.1.1), except the variance of the terminal node parameters is proportional to the variance of the error term, $\mu_{ij}|T,\sigma \sim N(0,\frac{\sigma^2}{a})$, as suggested by Chipman et al. (1998).[8] Integration of the likelihood with respect to the μ parameters and σ results in a closed form expression proportional to the marginal likelihood.

The marginal likelihood can be derived as follows. Let $Y = (Y_1, ..., Y_n)$ be the outcome vector. For a given sum of trees model $\mathcal{T}$, the posterior distribution of Y is:

$$Y|\mathcal{T}, M, \sigma^{-2} \sim N(\sum_{j=1}^{m} J_j M_j, \sigma^2 I)$$

where J_j (which depends on the original matrix of covariates X) is an $n \times b_j$ binary matrix with the element in position (i, j) indicating the inclusion of observation $i = 1, ..., n$ in terminal node $k = 1, ..., b_j$ of tree j.

Let $W = [J_1 ... J_m]$ be an $n \times b$ matrix , where $b = \sum_{j=1}^{m} b_j$ and $\underline{\boldsymbol{\mu}} = (M_1^T ... M_m^T)^T$ be a vector of size b of terminal nodes assigned to trees $T_1, ..., T_m$. We can then write $W\underline{\boldsymbol{\mu}} = \sum_{j=1}^{m} J_j M_j$, [9] and therefore

$$Y|\underline{\boldsymbol{\mu}}, \sigma^{-2} \sim N(W\underline{\boldsymbol{\mu}}, \sigma^2 I)$$

which, with $\underline{\boldsymbol{\mu}} \sim N(0, \frac{\sigma^2}{a} I_b)$, where I_b is a $b \times b$ identity matrix, implies

$$p(Y) = MVST_v(0, \lambda(I_n + \frac{1}{a} W W^T))$$

$$= \frac{\Gamma(\frac{\nu+n}{2})(\lambda v)^{\frac{\nu+n}{2}}}{\Gamma(\frac{v}{2}) v^{\frac{n}{2}} \pi^{\frac{n}{2}} \lambda^{\frac{n}{2}} (\frac{1}{a})^{\frac{b}{2}} det\,(aI_b + W^T W)^{\frac{1}{2}}} \left[\lambda v + Y^T Y - Y^T W (aI_b + W^T W)^{-1} W^T Y\right]^{-\frac{\nu+n}{2}}$$

Then, noting that anything that does not depend on W or b will cancel out when calculating the model weights, we can calculate:

$$\propto \frac{1}{(\frac{1}{a})^{\frac{b}{2}} det\,(aI_b + W^T W)^{\frac{1}{2}}} \left[\lambda v + Y^T Y - Y^T W (aI_b + W^T W)^{-1} W^T Y\right]^{-\frac{\nu+n}{2}}$$

And the log of this expression is: $\frac{b}{2}\log(a) - \frac{1}{2}\log(det(M)) - \frac{\nu+n}{2}\log(\lambda v + Y^T Y - Y^T W M^{-1} W^T Y)$ where $M = aI_b + W^T W$.

A deterministic model search algorithm first reduces the set of potential splitting variables by a change-point detection algorithm, and then recursively adds splits to trees that are potentially to be appended to models in the set of currently selected sum of tree models. After a set of single tree models are selected, changepoints in the residuals are used as potential splitting variables for constructing the next set of trees to potentially append to the selected models. Then a new set of residuals is constructed for the new set of sum-of-two-tree models, changepoints are detected, and trees are appended to create a set of sum-of-three-tree models, and so on.

The set of models to be averaged over are those with posterior probability within some distance of the highest probability model found by the model search algorithm. i.e. For all proposed models, $\mathcal{T}_\ell$, indexed by ℓ, the algorithm obtains

$$p(Y|\mathcal{T}_\ell, X)p(\mathcal{T}_\ell) \propto p(\mathcal{T}_\ell|Y, X) = \frac{p(Y|\mathcal{T}_\ell, X)p(\mathcal{T}_\ell)}{p(\mathbf{y})}$$

And keeps the models such that

$$\arg\max_{\ell'}(\log(p(\mathcal{T}_{\ell'}|Y, X))) - \log(p(\mathcal{T}_\ell|Y, X)) \leq \log(o)$$

where o is Occam's window, and the minimum is over the set of all proposed models.

[8] Moran et al. (2018) argue against the use conjugate priors in Bayesian linear regression. However, this issue will not be discussed in further detail in this paper. Nonetheless, it is worth noting that the methods introduced in this paper can be improved further by careful calibration of the a parameter, e.g. by cross-validation.

[9] $W\underline{\boldsymbol{\mu}} = \sum_{j=1}^{m} J_j M_j$ is analogous to $X\boldsymbol{\beta}$ in standard linear regression notation.

3 Framework for Generalization of BART-BMA and BART-IS

BART-BMA and BART-IS are applicable to a wide range of model settings in which a linear combination of covariates can be replaced by a sum-of-tree model. For example, for Logit-BART, the latent outcome can be modelled as a sum-of-trees instead of a standard linear model.

A key requirement for the computational feasibility of this general framework is that there should exist efficient methods for calculating the marginal likelihood and posterior predictions of the model of interest. A closed form for the posterior distribution or an efficient method for sampling from the posterior distribution is required for sampling any quantities of interest or producing credible intervals.

For models such as Logit and Tobit, it is possible to obtain an approximation to the marginal likelihood by a Laplace approximation about the Maximum a Posteriori parameter estimates, which can be obtained efficiently from a Quasi-Newton algorithm (Murphy 2012, Chib 1992). Similar approaches can be used for other models. Recently developed methods, including Integrated Nested Laplace Approximations (Rue et al. 2009), are applicable to a wide range of models including multinomial logit, Poisson regression, and models with hierarchical priors, e.g. mixed logit.[10]

Algorithm 1 and Algorithm 2 outline the general BART-BMA and BART-IS algorithms. The General BART-BMA algorithm begins by constructing latent outcome variable values for the training data (this is necessarily model-specific and arbitrary) and then applying a changepoint detection algorithm to obtain a set of potential splitting rules.[11] Single tree models are constructed using these splitting rules, as in standard BART-BMA (see chapter 2), but with marginal likelihood calculations that are specific to the generalized linear model and approximation method. Then a new set of residuals are calculated for each single-tree model in Occam's window, and changepoints are found for these new residuals. The new changepoints are used to construct new trees to be appended to the existing models, creating a set of sum-of-two tree models. Residuals are calculated for the sum-of-two tree models and a set of sum-of-three tree models are created, and so on. The General BART-IS algorithm is essentially the same as the algorithm presented in chapter 2, except the marginal likelihood and sampling from the mixture of posterior distributions are specific to the generalized linear model and approximation method. The random draws of trees are the same as in chapter 2,[12] and the General BART-IS is highly parallelizable as each iteration of the for-loop can be assigned to a different processor.

A limitation of the BART-BMA approach is that it requires the calculation of residuals and application of a changepoint detection algorithm to the residuals for the purpose of reducing the set of potential splitting variables. For some models, the residuals are of a latent outcome, and it is not clear how to proceed. In the case of Logit-BART-BMA, section 4 outlines how it is possible to make use of existing ideas for logit boosted tree methods (Friedman et al. 2000). However, there might not exist a straightforward and effective method for calculation of residuals for some models, and therefore the BART-IS approach, for which there is no calculation of residuals nor data-dependent search for splitting points, is more general. For General BART-BMA, the latent outcome is unknown for any observations in the training data, and the initialization is entirely arbitrary and model-specific. It is not guaranteed that there exists an initialization that leads to an effective model search for all models, and an entirely different model search algorithm without construction of latent outcome values or residuals may be more effective.

For averages of models with multiple latent outcomes (each modelled by a sum-of-trees) per model, the BART-BMA approach is infeasible[13] and the BART-IS approach remains feasible (although a larger number of models should be sampled). An example of such a model would be multinomial logistic regression with different sums-of-trees for the latent utility of each alternative.[14] However, the discussion in this paper will be restricted to settings where the same underlying variables (or sum-of-trees) are used for all latent variables.

A word of caution is required here. The performance of these methods is highly dependent on the

[10] However, there is a trade-off between accuracy of approximations and computational speed. In some cases it might not be computationally feasible to place a model in the BART-IS framework. This would require a level of experimentation with different approximation methods.

[11] Changepoint detection algorithms include Pruned Exact Linear Time Killick et al. (2012) and a simple grid-search.

[12] The samples of models can be made "offline", i.e. before any data is obtained, as in BART-IS and safe-Bayesian Random Forests (Quadrianto & Ghahramani 2014).

[13] A form of BART-BMA with considerable changes to the model search algorithm might be possible. This is beyond the scope of this paper.

[14] It is possible to share the same sum-of-tree structure, e.g. Linero et al. (2019), or sample separate sums-of-trees, e.g. Murray (2017).

appropriateness of the overall model specification (e.g. logit link function), the accuracy of the approximations of the marginal likelihood and posteriors. In the case of BART-BMA, the model search algorithm might not perform as well as for a simple linear model, and parameters such as the size of Occam's window and changepoint detection parameters may have to be tuned to control the trade-off between computational feasibility and breadth of the model search. BART-IS generally requires a large number of draws of models, and the feasibility of the approach is inversely related to the size of the model space and the computational time required to calculate the marginal likelihood.

Input: $n \times p$ matrix X
Response Y. Vector of binary, censored, categorical, or count data, or perhaps a more complicated (e.g. multivariate) outcome.
Output: Depends on the model. e.g. predicted outcomes or probabilities, parameter estimates.

Initialize: *Residuals*. Details depend on the model setting, e.g. for standard BART-BMA, begin with the vector of outcomes, and for Logit-BART-BMA begin with a transformation to the scale of the latent outcome (η_i in equation (1)). In general the residuals should be on the scale of the (possibly latent) variable that is directly modelled by a sum-of-trees.
Initialize *lowest_model_prob*, the minimum posterior probability of all models found so far.
Initialize: $L = 1$, Set the list of models $List\_ST$ to include a single tree model with no splits.

[Each round in the outer loop searches over possible additions of one tree to existing sum-of-tree models. (First round begins with single tree models)]

for $j \leftarrow 1$ *to num_trees* **do**
  [For each model ℓ in OW from the previous round, search for trees to add] **for** $\ell \leftarrow 1$ *to* L **do**
    if $count\_mu\_trees_\ell \leq m_\mu$ **then**
      1. **Find Good Splitting Rules**.

         Apply a changepoint detection algorithm to the residuals to reduce the number of potential splitting rules. This is model-specific, and may involve first applying some function to the residuals. See Logit-BART-BMA for an example.

      2. **Grow trees to append to sum-of-tree model**.

         Begin with a tree stump and grow trees recursively using splitting rules from step 1. Each time a split is considered, calculate the posterior model probability and check if the model is in OW. [This requires efficient calculation of the marginal likelihood].
         Add new models to temporary list $temp\_OW$ if in OW.
    end
    Make sum of trees models and update residuals
    Reset list of models in OW $List\_ST = temp\_OW$.
    Update *lowest_model_prob* to minimum posterior probability of models in $List\_ST$.
    Set $L = length(temp\_OW)$. Reset $temp_{OW}$ to list of length zero.
  end
end
Delete models in $list\_ST$ with log posterior probability more than $log(o)$ from *lowest_model_prob*.

The output is a model averaged prediction of an outcome/probability or parameter estimate.

Intervals can be obtained from either a closed form expression or probability-weighted sampling from each model in OW.

Algorithm 1: BART-BMA General Algorithm

Input: $n \times p$ matrix X
Response Y. Vector of binary, censored, categorical, or count data, or perhaps a more complicated (e.g. multivariate) outcome.
Output: Depends on the model. e.g. predicted outcomes or probabilities, parameter estimates.

Each round in the outer loop involves drawing a model from a model sampler. This loop is trivially parallelizable.

for $m \leftarrow 1$ *to num_models* **do**

1. Draw a model from the model sampler. This can be the sampler used by Quadrianto & Ghahramani (2014), the BART prior, or the spike and tree prior (Rockova & van der Pas 2017).

2. Obtain the model predictions and/or parameters that summarize the (possibly approximate) posterior distribution.

3. Obtain model weights. This requires efficient calculation of the marginal likelihood. If the model sampler is not the model prior, then multiply the marginal likelihood by the ratio of the model prior probability to the model sampler probability. For a safe-Bayesian approach, use the marginal likelihood to the power of a number between 0 and 1 (Quadrianto & Ghahramani 2014).

end
The output depends on the model and object of interest.
e.g. The predicted outcome or probability is a marginal likelihood weighted average of model predictions.
Parameter distributions and credible intervals can be obtained from model weighted samples from (possibly approximate) posterior distributions. In some cases a closed form gives an efficient alternative.

Algorithm 2: BART-IS General Algorithm

4 Example of General Algorithms Applied to Binary Outcome Data: Logit-BART-BMA and Logit-BART-IS

In this section, the general algorithms introduced in section 3 are applied to Logit-sum-of-tree models for binary outcome data. First, an outline is given for a simpler benchmark approach that does not make use of the more principled algorithm. Second, the binary outcome model and Laplace approximation method are summarized. The Logit-BART-BMA and Logit-BART-IS algorithms are detailed as specific examples of the general framework. Finally, the methods are applied to binary classification datasets.

4.1 A Benchmark Probit Approximation for BART-BMA and BART-IS

Single tree methods can readily be applied to binary outcome data. Trees in a random forest applied to binary outcomes produce predictions between zero and one because leaf estimates are averages of binary variables. However, sum-of-tree based methods such as BART are less directly applicable to binary outcome data because sums-of-trees can produce predictions outside the range $[0, 1]$ and ideally the statistical framework of BART should account for the fact that the outcomes are binary.

A simple extension of BART to Probit involves first converting the binary outcomes to the scale of the latent variable, replacing observations $y_i = 1$ with $y_i^* = 3.1$ and replacing observations $y_i = 0$ with $y_i^* = -3.1$. These latent variable values correspond to very high and very low probabilities of $y_i = 1$. Then standard BART-MCMC, BART-BMA, or BART-IS is applied to the data with y_i^* as the dependent variable. Finally,

the normal CDF function is applied to the latent outcome predictions to obtain predicted probabilities, and applied to the latent outcome prediction intervals to obtain prediction intervals for the probabilities. This is the approach adopted by Hernández et al. (2018) and will be referred to in the remainder of this document as Approximate Probit-BART-BMA.[15] Similarly, this approach in combination with the BART-IS algorithm will be referred to as Approximate Probit-BART-IS.

However, the approach outlined above does not truly apply a binary outcome model to the data. A more rigorous approach would involve a likelihood that accounts for the probability that the actual outcome equals to zero or one, and not begin with arbitrary values for the latent outcome. The framework outlined in section 3 provides one possible method for implementing the more rigorous approach.

4.2 Model, Priors, and Notation for Logit-BART

Throughout this chapter, the notation is chosen to be similar to that used by Hernández et al. (2018). The prior for the terminal node parameters is $\mu_{k,j}|T,\sigma \sim N(0,\frac{1}{a})$ (where j,k denotes the j^{th} terminal node of the k^{th} tree in the sum-of-tree model), and unlike in BART-BMA for continuous outcomes, there is not a separate parameter for the variance of the error term (the variance of the error term is not separately identified).

Let $W = [J_1...J_m]$ be an $n \times b$ matrix , where $b = \sum_{j=1}^{m} b_j$, J_j is a binary matrix of size $n \times b_j$ with the element in the i^{th} row and j^{th} column denoting the inclusion of observation $i = 1, ..., n$ in terminal node $k = 1, ..., b_j$ of tree j. Let M_j be a vector $(\mu_{1,j}, ..., \mu_{b_j,j})$ of terminal node means for the j^{th} tree, and let $\underline{\boldsymbol{\mu}} = (M_1^T ... M_m^T)^T$ be a vector of size b of terminal node means assigned to trees $T_1, ..., T_m$. We can then write $W\underline{\boldsymbol{\mu}} = \sum_{j=1}^{m} J_j M_j$. The product $W\underline{\boldsymbol{\mu}} = \sum_{j=1}^{m} J_j M_j$ is analogous to $X\boldsymbol{\beta}$ in standard linear regression notation. Let W_i denote the i^{th} row of W.

The outcomes are binary, $y_i \in \{0, 1\}$. The probability of the outcome $y_i = 1$ is given by the logistic function, and will be denoted by p_i for convenience:

$$p_i = \Pr(y_i = 1|W_i, \underline{\boldsymbol{\mu}}) = \frac{1}{1 + e^{-\underline{\boldsymbol{\mu}}^T W_i^T}} = \frac{e^{W_i \underline{\boldsymbol{\mu}}}}{1 + e^{W_i \underline{\boldsymbol{\mu}}}}$$

where W_i denotes the i^{th} row of W. The likelihood is: $p(\boldsymbol{y}|W, \underline{\boldsymbol{\mu}}) = \prod_{i=1}^{N} p_i^{y_i}(1-p_i)^{1-y_i}$.[16] The log-likelihood is $\sum_{i=1}^{N}[y_i \log p_i + (1 - y_i) \log(1 - p_i)] = \boldsymbol{y}^T W\underline{\boldsymbol{\mu}} - \sum_{i=1}^{N} \log(1 + e^{-W_i\underline{\boldsymbol{\mu}}})$.

4.3 Laplace Approximation

The prior $\underline{\boldsymbol{\mu}} \sim \mathcal{N}(0, \frac{1}{a} I_b)$ and the likelihood give an intractable posterior distribution. However, a Laplace approximation gives a normal posterior distribution for the terminal node parameters. An approximation of the posterior can be obtained by s second order Taylor expansion about the Maximum A Posteriori (MAP) estimate:

$$\underline{\boldsymbol{\mu}}_{MAP} = \arg\min_{\underline{\boldsymbol{\mu}}} -(\log p(\boldsymbol{y}|W, \underline{\boldsymbol{\mu}}) + \log p(\underline{\boldsymbol{\mu}}))$$

$$= \arg\min_{\underline{\boldsymbol{\mu}}} - \left[\boldsymbol{y}^T W\underline{\boldsymbol{\mu}} - \sum_{i=1}^{N} \log(1 + e^{-W_i\underline{\boldsymbol{\mu}}}) - \frac{1}{2} b \log(2\pi) + \frac{1}{2} b \log(a) - \frac{a}{2} \underline{\boldsymbol{\mu}}^T \underline{\boldsymbol{\mu}} \right]$$

The approximate distribution is:

$$p(\underline{\boldsymbol{\mu}}|\boldsymbol{y}, W) \approx \mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP}, H^{-1})$$

where H is the Hessian matrix of the negative log posterior (evaluated at the MAP).

$$H = W^T S W + a I_b$$

where $S = diag(p_i(1-p_i))$ is an $n \times n$ diagonal matrix with diagonal elements determined by the probabilities p_i obtained from the logistic function. The Hessian and the gradient of the negative posterior probability can be used to obtain an approximation of the MAP. The gradient is $\boldsymbol{g} = W^T(\boldsymbol{p} - \boldsymbol{y}) + a\underline{\boldsymbol{\mu}}$ where $\boldsymbol{p} = (p_1, ..., p_n)^T$.

[15] The improvements to the BART-BMA algorithm described in chapter 2 of this thesis also apply to this approximate Probit-BART implementation.

[16] Note that W is defined by the sum-of-tree model $\mathcal{T}$. Conditioning on the model is excluded here for brevity.

The MAP can be found by Newton's method or more efficient Quasi-Newton methods such as the limited memory BFGS (L-BFGS) algorithm.[17]

When averaging over the set of sum-of-tree models $\mathcal{T}_1, ..., \mathcal{T}_M$, the approximate distribution of the parameters is:

$$\underline{\boldsymbol{\mu}}|\boldsymbol{y} \sim \sum_{m=1}^{M} \mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)}, H_{(m)}^{-1}) p(\mathcal{T}_m|\boldsymbol{y})$$

where $p(\mathcal{T}_m|\boldsymbol{y})$ is the posterior model probability,

$$p(\mathcal{T}_m|\boldsymbol{y}) \propto p(\boldsymbol{y}|\mathcal{T}_m) p(\mathcal{T}_m)$$

where $p(\boldsymbol{y}|\mathcal{T}_m)$ is the marginal likelihood, which can be approximated using the Laplace approximation, as outlined in Appendix B, and $p(\mathcal{T}_m)$ is the prior model probability. The prior probability is the same as for BART-BMA for continuous outcomes and straightforward to calculate or, in the case of BART-IS, it does not need to be calculated.

The subsections below include details for estimating the posterior mean, calculating credible intervals, and calculating the marginal likelihood.

4.3.1 Estimation of Posterior Predictive Mean Probability

The model averaged approximate posterior for coefficients is $\underline{\boldsymbol{\mu}}|\boldsymbol{y} \sim \sum_{m=1}^{M} \mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)}, H_{(m)}^{-1}) p(\mathcal{T}_m|\boldsymbol{y})$. Using the logistic (sigmoid) function probability $\frac{e^{W_i \underline{\boldsymbol{\mu}}}}{1+e^{W_i \underline{\boldsymbol{\mu}}}}$, the model averaged posterior (predictive) probability is:

$$p(y_* = 1|\boldsymbol{x}_*, X, \boldsymbol{y}) = \sum_{m=1}^{M} \left(\int \frac{e^{W_{*,(m)} \underline{\boldsymbol{\mu}}_{(m)}}}{1 + e^{W_{*,(m)} \underline{\boldsymbol{\mu}}_{(m)}}} p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)}, H_{(m)}^{-1}) d\underline{\boldsymbol{\mu}}_{(m)} \right) p(\mathcal{T}_m|\boldsymbol{y})$$

where y_* is the outcome for a new observation, X is the matrix of variables in the training data, $\boldsymbol{y}$ is the vector of outcomes in the training data, $\boldsymbol{x}_*$ is the covariate vector for the new observation, which is input to the sum-of-tree models to obtain row vectors for each model, $W_{*,(m)}$, $m = 1, ..., M$, consisting of binary variables to indicate inclusion in terminal nodes.

The integral in the above expression for $p(y_* = 1|\boldsymbol{x}_*, X, \boldsymbol{y})$ is intractable. Numerous approaches are possible for estimation of predictive probabilities, and calculation of the marginal likelihood and credible intervals.[18] The example below outlines a standard Laplace approximation with the probit function (normal CDF) used as an approximation to the logistic (sigmoid) function because this approach fast, straightforward to implement, and can be used to benchmark other approaches. Appendix D outlines simple Monte Carlo alternatives.

4.3.2 Probit Approximation of Posterior Predictive Mean Probability

Machine learning methods often combine the Laplace approximation for logistic regression with a normal CDF approximation (Spiegelhalter & Lauritzen 1990, Bishop 2006, Murphy 2012). The logistic (sigmoid) function can be approximated by the normal CDF:

$$p(y_* = 1|\boldsymbol{x}_*, X, \boldsymbol{y}) = \sum_{m=1}^{M} \left(\int \frac{e^{W_{*,(m)} \underline{\boldsymbol{\mu}}_{(m)}}}{1 + e^{W_{*,(m)} \underline{\boldsymbol{\mu}}_{(m)}}} p(\underline{\boldsymbol{\mu}}_{(m)}|O_{MAP,(m)}, H_{(m)}^{-1}) d\underline{\boldsymbol{\mu}}_{(m)} \right) p(\mathcal{T}_m|\boldsymbol{y})$$

$$\approx \sum_{m=1}^{M} \left(\int \Phi(e^{W_{*,(m)} \underline{\boldsymbol{\mu}}_{(m)}}) p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)}, H_{(m)}^{-1}) d\underline{\boldsymbol{\mu}}_{(m)} \right) p(\mathcal{T}_m|\boldsymbol{y})$$

[17] See appendix A for the standard Newton method for finding the minimum of the negative log of the posterior distribution. The implementations provided in the **R** packages **safeBart** and **logitbartbma** use L-BFGS.

[18] See Chopin et al. (2017) for a discussion

The integrals in the above expression can be rewritten as one-dimensional integrals:

$$p(y_* = 1|\boldsymbol{x}_*, X, \boldsymbol{y}) \approx \sum_{m=1}^{M} \left(\int \Phi(\alpha_{(m)}) p(\alpha_{(m)}|\psi_{\alpha,(m)}, \sigma^2_{\alpha,(m)}) d\alpha_{(m)} \right) p(\mathcal{T}_m|\boldsymbol{y}) = \sum_{m=1}^{M} \Phi \left(\frac{\psi_{\alpha,(m)}}{\sqrt{1+\sigma^2_{\alpha,(m)}}} \right) p(\mathcal{T}_m|\boldsymbol{y})$$

where $\psi_{\alpha,(m)} = W_{*,(m)} \underline{\boldsymbol{\mu}}_{MAP,(m)}$ and $\sigma^2_{\alpha,(m)} = W_{*,(m)} H^{-1}_{(m)} W^T_{*,(m)}$. For each model, the distribution of $\alpha_{(m)} = W_{*,(m)} \underline{\boldsymbol{\mu}}_{(m)}$ is $\mathcal{N}(\psi_{\alpha,(m)}, \sigma^2_{\alpha,(m)})$.

Often 1 is replaced by t^{-2} where $t^2 = \frac{\pi}{8}$ to give a closer approximation to the probability that would have been obtained from the logistic function (Spiegelhalter & Lauritzen 1990, Bishop 2006, Murphy 2012):

$$p(y_* = 1|\boldsymbol{x}_*, X, \boldsymbol{y}) \approx \sum_{m=1}^{M} \Phi \left(\frac{\psi_{\alpha,(m)}}{\sqrt{\frac{8}{\pi}+\sigma^2_{\alpha,(m)}}} \right) p(\mathcal{T}_m|\boldsymbol{y})$$

A number of alternative approaches exist for calculating the marginal likelihood and posterior mean. Appendix C describes an approach for estimating the posterior mean that involves applying Laplace's method twice (Tierney & Kadane 1986). Calculation of credible intervals by root-finding or Monte Carlo draws from the posterior is straightforward and detailed in Appendices E and D.2.

4.3.3 Alternative to Probit Approximation: Gibbs Sampler for Final Inference (Laplace Approximation for Marginal Likelihoods)

As in the original BART-BMA paper (Hernández et al. 2018), after the models are selected (or sampled in the case of BART-IS) it is possible to use a Gibbs sampler to take draws from each model, and draw from each model with probability equal to the posterior model probability. In the case of Logit-BART-BMA, this can be implemented by estimating the posterior model probability using a Laplace approximation (as outlined above), or some other method[19] and then taking "exact" draws (from the true model rather than an approximation) using a Gibbs sampler (Albert & Chib 1993). For each draw of model parameters, it is possible to calculate the quantity of interest. e.g. $\frac{e^{W_{*,(m)}\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{W_{*,(m)}\underline{\boldsymbol{\mu}}_{(m),s}}}$ (or differences in probabilities for treatment effects). Then the mean and quantiles of the values across samples can be used for predictions and credible intervals.

The Gibbs sampler described by Polson et al. (2013) is potentially well-suited to this purpose because it is fast and uniformly ergodic (Choi et al. 2013). Polson et al. (2013) note that their sampler "opens the door for exact Bayesian treatments of many modern-day machine-learning classification methods based on mixtures of logits ".

4.4 Logit-BART-BMA

The prior over the model space is the same as for standard BART-BMA and BART-IS.[20] The Logit-BART-BMA model search algorithm is a special case of Algorithm 1 and only differs from the standard BART-BMA algorithm in the calculation of residuals and application of a changepoint detection algorithm to the residuals to reduce the number of potential splitting rules. This section discusses a number of possible approaches to the calculation of residuals and the changepoint detection algorithm for Logit-BART-BMA.

There are a few potential methods for suggesting potential splitting rules in each round of the model search algorithm. The approach presented here is inspired by the LogitBoost algorithm (Friedman et al. 2000). Alternative approaches are detailed in Appendix F.

A variant of AdaBoost, LogitBoost (Friedman et al. 2000), involves fitting a base learner to be added to a sum of models (i.e. boosted models), to which the logistic function is then applied to obtain the probability.

[19] See Friel & Wyse (2012) for a review of possible methods.

[20] Alternative priors on the tree structures, provided in the **R** packages **logitbartBMA** and **safeBart** include the prior proposed by Quadrianto & Ghahramani (2014) and the spike and tree prior Rockova & van der Pas (2017). Code is available at https://github.com/EoghanONeill

Each base learner minimizes the weighted sum of squares, i.e. applies weighted least squares, to the following variable:

$$z_i = \frac{y_i - p(x_i)}{p(x_i)(1 - p(x_i))}$$

with weights $w_i = p(x_i)(1-p(x_i))$, where $p(x_i)$ is the individual-specific probability estimated in the previous round, initialized at $p(x_i) = 0.5$. [21]

Logit BART-BMA estimates the whole logit model at each step, and therefore z_i is only really relevant to the initial stage in each round that involves applying the changepoint detection algorithm. The proposed approach here is to apply the changepoint detection algorithm to z_i with a weighted (sum of squares) cost function with weights w_i. [22]

The key idea, as with AdaBoost, is that the set of changepoints used in constructing new trees to be appended to the model, should place more weight on observations misclassified by the current model. However, unlike AdaBoost, there is no such adjustment made in the final criterion for the acceptance of the new trees because the entire sum-of-tree model is re-estimated when a new tree is appended to a model and the marginal likelihood based criterion is applied to the entire model.

This adjustment to changepoint detection is also applicable to the naive approximation to Probit-BART-BMA and Probit-BART-IS discussed in section 4.1. In approximate Probit-BART-BMA, it is also possible to fit the new tree using z_i as in LogitBoost. However, this is a topic for future research. [23]

4.5 Logit-BART-IS

Logit-BART-IS is a special case of the general framework given in Algorithm 2. Algorithm 3 outlines how to apply the efficient logit approximation methods described in section 4.3 in the general BART-IS framework. Logit-BART-BMA does not involve model search, and therefore does not involve initialization of latent outcome values or calculation of residuals.

[21] It is possible to apply the restriction $z_i \in [-3, 3]$ and also apply trimming or another method to avoid numerical instability issues when dividing by $p(x_i)(1 - p(x_i))$ when $p(x_i)$ is close to zero or one.

[22] The weights w_i essentially account for second-order information. Other methods such as MART (Friedman & Meulman 2003) only use $y_i - p(x_i)$ in the tree building step (but use second order information when estimating terminal nodes values).

[23] This is more applicable to the original BART-BMA implementation of Hernández et al. (2018) that estimated each new tree separately using only residuals, and less applicable to the new BART-BMA implementation presented in chapter 2 of this thesis which estimates the whole model at each step. In this sense the new implementation of BART-BMA is more analogous to variations on AdaBoost algorithms that perform backfitting at each step.

Input: $n \times p$ matrix X
Response binary vector Y.
Output: Predictive probabilities, intervals for predictive probabilities.

Each round in the outer loop involves drawing a model from a model sampler. This loop is trivially parallelizable.

for $m \leftarrow 1$ *to num_models* **do**

1. Draw a model from the model sampler. This can be the sampler used by Quadrianto & Ghahramani (2014), the BART prior (Chipman et al. 2010), or the spike and tree prior (Rockova & van der Pas 2017).
2. Obtain MAP parameter values for a Laplace approximation as outlined in section 4.3. Obtain predicted probabilities as outlined in section 4.3.2.
3. Obtain model weights. The marginal likelihood is efficiently calculated as outlined in section B.

end

Model averaged predictions are calculated as outlined in section 4.3.2.
Credible intervals for the model averaged distribution are obtained as outlined in Appendix E.

Algorithm 3: Logit BART-IS Algorithm

4.6 Application to UCI Datasets

This section contains a comparison of Logit-BART-BMA and Logit-BART-IS against other methods using publicly available datasets from the widely used UCI Machine Learning Repository (Dua & Graff 2017). The chosen datasets are binary classification datasets relevant to economic applications. Table 1 contains a description of the data. Missing observations are removed from all datasets. The number of variables is the number remaining after removal of some variables (e.g. unique text strings), and transformation of some categorical variables into multiple binary variables so that tree-based methods are applicable with available software.

The algorithms compared are Logit-BART-IS,[24] Logit-BART-BMA, Approximate Probit BART-IS,[25] Approximate Probit BART-BMA, Probit-BART-MCMC, Logit-BART-MCMC,[26] linear logistic regression, and Random Forests.[27] Methods are evaluated using the Brier Score and Area Under the Curve (AUC).[28] The data is randomly divided into training and hold-out test data, and all methods are applied without parameter tuning.

Tables 1 to 6 show the binary classification results for a range of training sample sizes. Across many examples, the Logit-BART-IS and Logit-BART-BMA implementations are surprisingly competitive with the MCMC implementations given the small number of trees in each model and relatively small number of sampled models for BART-IS and very small number of models in Occam's window in BART-BMA. For a

[24] Logit-BART-IS was implemented with only 5 trees per model, and a total of 20,000 sampled models. This is a small number of draws relative to the number of models drawn for BART-IS with continuous outcomes in the second chapter of this thesis. Each model takes more computational time than a linear model, therefore some compromise must be made on computational speed. However, the results are surprisingly competitive with the MCMC implementations considering the small number of samples. Therefore a topic for future research would be whether the results are more accurate with a larger set of samples, perhaps using parallelization over a larger number of cores for computational feasibility.

[25] Approximate Probit BART-IS was implemented with only 10 trees per model and a total of only 1000 sampled models. The results are surprisingly competitive given the small number of samples.

[26] Probit-BART-MCMC and Logit-BART-MCMC were both implemented using the **R** package **BART** with 5000 burn-in draws and 10,000 post-burn-in draws. Each model sampled by Probit-BART-MCMC has the default number of 50 trees, and each model sampled by Logit-BART-MCMC has the default number of trees of 200.

[27] Random Forests were implemented using the **R** package **ranger** and 10,000 trees. All other parameters were set to the default values.

[28] The Brier score is defined as $\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{p}_i)^2$ where N is the number of samples in the holdout data and $\hat{p}_i$ is the predicted probability that $y_i = 1$. The area under the Receiver Operating Characteristic curve is calculated using the **R** package **ROCR**.

number of datasets, the more principled general framework with Laplace approximations provides a notable improvement over the probit transformation approach described in section 4.1. The results demonstrate that the general BART approach introduced in this paper produces the intended result of estimates that are similar to those produced by MCMC BART implementations.[29] Logit-BART-BMA results are only presented for the training samples of up to 2000 observations due to the computational time required for some datasets under default parameters.[30]

The similar performance of Logit-BART-MCMC and Probit-BART-MCMC is unsurprising. The fact that there is no consistently best performing model across all sample sizes and datasets (although Probit-BART-MCMC is the method that slightly outperforms other methods across the most datasets) indicates that all methods produce similar estimates and the ranking of methods may be influenced by random variation in the data and splitting into test and training data. It is possible that for some datasets, MCMC does not deliver notable improvements over simple BMA or IS based approaches, while for other datasets there may be potential for more substantial gains from MCMC.[31]

The results for the `Census Income` dataset indicate that the IS based approach does not perform as well as the MCMC approach, and no approaches perform markedly better than standard logistic regression. It is possible that there is a strong linear relationship between one of the covariates and the dependent variable, and therefore the logistic regression model performs well. This may also explain the poor results for Logit-BART-IS when applied to this dataset because an implementation that makes use of fewer trees per sum-of-tree model is likely to be less precise at capturing linear functions of covariates.[32]

[29] It is not expected that the Logit-BART-BMA or Logit-BART-IS results give notably more accurate estimates than the MCMC based implementations as ultimately these are alternative implementations for essentially the same model framework.

[30] The computational requirements for the Logit-BART-BMA search algorithm are likely to be sensitive to model search parameters, model prior parameters, and choice of changepoint detection algorithm. The optimal choice of parameters may differ across datasets, and differ to standard BART-BMA for continuous outcomes.

[31] See Chopin et al. (2017) for a similar discussion regarding sampling of parameters in a single logistic regression model.

[32] Probit-BART-MCMC was implemented with 50 trees per model, Logit-BART-MCMC was implemented with 200 trees per model, and Logit-BART-IS and Logit-BART-BMA were implemented with 5 trees per model.

Dataset name	Description (from UCI repository)	Number of variables	Number of Observations	Reference
`Shopper`	Online Shoppers Purchasing Intention Dataset Data Set.	74	12,330	Sakar et al. (2019)
`Bank Marketing`	The data is related with direct marketing campaigns (phone calls) of a Portuguese banking institution. The classification goal is to predict if the client will subscribe a term deposit (variable y).	51	4521	Moro et al. (2014)
`Insurance`	This data set used in the CoIL 2000 Challenge contains information on customers of an insurance company (caravan insurance in the Netherlands). The data consists of 86 variables and includes product usage data and socio-demographic data.	133	5821	Van Der Putten & van Someren (2000)
`Credit Cards`	Prediction of customer default in Taiwan.	33	30,000	Yeh & Lien (2009)
`Credit Screening`	Examples represent positive and negative instances of people who were and were not granted credit by a Japanese company that grants credit.	46	653	None
`German Credit`	Statlog (German Credit) Data Set. This dataset classifies people described by a set of attributes as good or bad credit risks.	61	1000	None
`Australian Credit`	Statlog (Australian Credit Approval) Data Set. This file concerns credit card applications.	42	690	Quinlan (1987)
`Census Income`	Predict whether income exceeds \$50K/yr based on census data. Extraction was done by Barry Becker from the 1994 US Census database.	64	30,162 training, 15,060 testing	Kohavi (1996)

Table 1: UCI Dataset descriptions

4.6.1 UCI Binary Outcome Data Results

500 Training Observations

	Shopper		Bank Marketing		Insurance		Credit Cards	
Method	Brier	AUC	Brier	AUC	Brier	AUC	Brier	AUC
Logit-BART-IS	**0.079**	0.903	0.081	0.865	0.057	0.661	0.145	0.751
Logit-BART-BMA	0.083*	0.868*	0.084	0.841	0.058	0.653	0.143	0.727
Approx-Probit-BART-IS	0.145	0.661	0.102	0.715	0.060	0.591	0.175	0.698
Approx-Probit-BART-BMA	0.091	0.866	0.090	0.861	0.065	0.603	0.172	0.719
Probit-BART-MCMC	**0.079**	**0.906**	0.080	0.870	**0.056**	**0.688**	0.141	0.751
Logit-BART-MCMC	0.081	0.905	0.082	0.854	**0.056**	0.679	**0.140**	**0.753**
Logistic Regression	0.149	0.692	0.559	0.485	0.104	0.580	0.164	0.688
RF	0.085	0.897	**0.077**	**0.883**	0.059	0.591	0.145	0.745
Holdout sample size	11,830		4021		5321		29,500	

* indicates where the PELT algorithm with unweighted residuals was used for reducing the number of splitting points.

Table 2: UCI Binary Classification Datasets, training sample size = 500

500 Training Observations

	Credit Screening		German Credit		Australian Credit		Census Income	
Method	Brier	AUC	Brier	AUC	Brier	AUC	Brier	AUC
Logit-BART-IS	0.097	0.930	0.178	0.737	0.147	0.873	0.128	0.859
Logit-BART-BMA	0.095	0.930	0.187	0.713	**0.101**	**0.927**	0.124	0.862
Approx-Probit-BART-IS	0.210	0.811	0.188	0.736	0.224	0.749	0.155	0.768
Approx-Probit-BART-BMA	0.102	0.912	0.214	0.712	0.163	0.888	0.135	0.855
Probit-BART-MCMC	0.096	0.924	0.164	0.777	0.142	0.875	**0.113**	**0.888**
Logit-BART-MCMC	0.097	0.919	**0.162**	**0.778**	0.147	0.871	0.118	0.879
Logistic Regression	0.129	0.857	0.168	0.775	0.158	0.858	0.140	0.845
RF	**0.092**	**0.942**	0.167	0.772	0.131	0.887	**0.113**	0.886
Holdout sample size	153		500		190		15,060	

Table 3: UCI Binary Classification Datasets, training sample size = 500

1000 Training Observations

	Shopper		Bank Marketing		Insurance		Credit Cards		Census Income	
Method	Brier	AUC	Brier	AUC	Brier	AUC	Brier	AUC	Brier	AUC
Logit-BART-IS	**0.077**	**0.915**	0.080	0.849	0.055	**0.751**	**0.140**	**0.756**	0.123	0.864
Logit-BART-BMA	0.100	0.791	0.080	0.842	0.056	0.711	0.142	0.735	0.119	0.871
Approx-Probit-BART-IS	0.144	0.680	0.108	0.752	0.060	0.659	0.178	0.714	0.197	0.763
Approx-Probit-BART-BMA	0.088	0.898	0.085	0.846	0.060	0.635	0.182	0.692	0.118	0.888
Probit-BART-MCMC	0.078	0.909	**0.075**	0.879	**0.054**	0.744	0.141	0.751	**0.106**	**0.899**
Logit-BART-MCMC	0.079	0.908	0.077	0.874	0.055	0.733	**0.140**	0.754	0.109	0.895
Logistic Regression	0.160	0.749	0.080	0.848	0.078	0.583	0.150	0.706	0.220	0.641
RF	0.080	0.908	0.076	**0.885**	0.058	0.630	0.144	0.740	0.109	0.896
Holdout sample size	11,330		3521		4821		29,000		15,060	

Table 4: UCI Binary Classification Datasets, training sample size = 1000

2000 Training Observations

	Shopper		Bank Marketing		Insurance		Credit Cards		Census Income	
Method	Brier	AUC	Brier	AUC	Brier	AUC	Brier	AUC	Brier	AUC
Logit-BART-IS	0.075	0.912	0.085	0.848	**0.050**	0.717	0.141	0.744	0.124	0.871
Logit-BART-BMA	0.097	0.793	0.076	0.860	0.052	0.722	0.140	0.739	0.116	0.879
Approx-Probit-BART-IS	0.144	0.698	0.103	0.689	0.055	0.641	0.172	0.701	0.185	0.758
Approx-Probit-BART-BMA	0.085	0.906	0.081	0.846	0.054	0.707	0.165	0.711	0.121	0.885
Probit-BART-MCMC	**0.075**	**0.917**	**0.072**	0.896	**0.050**	**0.753**	**0.137**	**0.766**	**0.102**	**0.909**
Logit-BART-MCMC	0.076	0.915	0.073	0.895	**0.050**	0.751	**0.137**	**0.766**	**0.102**	0.908
Logistic Regression	0.135	0.705	0.080	0.864	0.058	0.673	0.148	0.711	0.111	0.888
RF	0.076	**0.917**	**0.072**	**0.899**	0.052	0.718	0.140	0.764	0.104	0.905
Holdout sample size	10,330		2521		2821		28,000		15,060	

Table 5: UCI Binary Classification Datasets, training sample size = 2000

10,000 Training Observations

	Credit Cards		Census Income	
Method	Brier	AUC	Brier	AUC
Logit-BART-IS	0.138	0.751	0.125	0.854
Approx-Probit-BART-IS	0.171	0.682	0.193	0.711
Probit-BART-MCMC	**0.135**	**0.776**	**0.098**	**0.914**
Logit-BART-MCMC	0.136	0.778	0.100	0.912
Logistic Regression	0.144	0.721	0.105	0.902
RF	0.136	0.770	0.100	0.910
Holdout sample size	20,000		15,060	

Table 6: UCI Binary Classification Datasets, training sample size = 10,000

5 Example Application of General Algorithms to Treatment Effect Estimation For Binary Outcomes

Subsection 5.1 outlines how the methods introduced in section 4 can be applied to treatment effect estimation for binary outcomes. Subsection 5.2 outlines BMA and an alternative parameterization of Logit-BART for treatment effect estimation. Subsection 5.3 compares the accuracy of these methods to existing implementations using simulated data.

5.1 Treatment Effect Estimation with Logit-BART-BMA and Logit-BART-IS

Often a policy maker is interested not only in prediction, but in the effect of the allocation of an individual or other unit of interest to "treatment" (Kleinberg et al. 2015). The object of interest in such a scenario is the treatment effect, which is defined as the difference in potential outcomes $Y_i(1) - Y_i(0)$, where $Y_i(1)$ is the potential outcome if individual i is allocated to treatment and $Y_i(0)$ is the potential outcome if individual i is allocated to the control group (Neyman 1923, Rubin 1974). The fundamental problem of causal inference is that we do not observe the causal effect for any individual, i (Holland 1986).

The estimand of interest is the Individual Treatment Effect (ITE)

$$\tau(x) = \mathbb{E}[Y_i(1) - Y_i(0)|X_i = x]. \tag{2}$$

Whereas the ATE can be estimated by a difference in means $\bar{y}_t - \bar{y}_c$, where $\bar{y}_t$ ($\bar{y}_c$) is the mean of the outcome variable for the treated (control) group, the CATE can be thought of as a subpopulation average treatment effect.[33] [34] The CATE is identified under unconfoundedness, i.e. $Y_i(1), Y_i(0) \perp T_i|X_i$, and overlap, i.e. $0 < \Pr(T_i = 1|X_i = x) < 1 \; \forall \; x$, where T_i denotes the treatment indicator variable.

BART has been shown to be a highly effective method for treatment effect estimation (Hill 2011, Green & Kern 2012, Dorie et al. 2019, Hahn et al. 2019, Wendling et al. 2018). The standard approach to treatment effect estimation using BART is in the S-Learner framework of treatment effect meta-algorithms (Künzel et al. 2019). The treatment variable is included as a potential splitting variable in the same way as all the other covariates. Treatment effect estimates are obtained from the difference in predictions from the trained model when treatment is set to 1 and set to 0, i.e. the estimates ITE is $\hat{f}(1, x_i) - \hat{f}(0, x_i)$ where $\hat{f}$ is the prediction function obtained from an average of sum-of-tree models and the arguments are the treatment dummy variable and all other covariates, x_i. Confounding can be mitigated by including the estimated propensity score as a potential splitting variable (Hahn et al. 2017). See the second chapter of this thesis for further details.

Logit-BART-BMA and Logit-BART-IS can be applied to treatment effect estimation for binary outcomes using the usual S-Learner approach. The technical details for calculation of predictions and prediction intervals are included in appendix G.

5.2 Logit-BCF-BMA and Logit-BCF-IS

Bayesian Causal Forests (BCF) is a method for treatment effect estimation (Hahn et al. 2020). See section 5.1 for an overview of treatment effects and the potential outcome framework. BCF is re-parameterization of BART that allows for an independent prior to be placed on τ and also include the estimated propensity score, $\hat{\pi}_i$, as a potential splitting variable.

$$f(x_i, z_i) = \mu(x_i, \hat{\pi}_i) + \tau(x_i)z_i$$

where $\mu(x_i, \hat{\pi}_i)$ and $\tau(x_i)$ are both sums of trees. See the second chapter of this thesis for further details. This section outlines how the general BART-BMA and BART-IS frameworks can be used to implement Bayesian Causal Forests for binary dependent variables with a logistic function of the sum-of-tree model:

$$\Pr(y_i = 1|x_i, \hat{\pi}_i, z_i) = sig(\mu(x_i, \hat{\pi}_i) + \tau(x_i)z_i)$$

[33] In instances where we condition on x being in some subset of the covariate space, i.e. $x \in \mathbb{A} \subset \mathbb{X}$, and $\tau_{\mathbb{A}} = \mathbb{E}[Y_i(1) - Y_i(0)|x \in \mathbb{A}]$, we also refer to this as the CATE (with suitably re-defined covariates).

[34] Another estimand is the average treatment effect conditional upon observed covariates $\bar{\tau} = \frac{1}{N}\sum_{i=1}^{N} \tau(x_i) = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}[Y_i(1) - Y_i(0)|X_i = x_i]$. Imbens & Rubin (2015) refer to this as the conditional average treatment effect, but we shall use the above definition of the CATE.

where sig is a sigmoid (logistic) function. The vectors of μ and τ parameters, $\underline{\boldsymbol{\mu}}$ and $\underline{\boldsymbol{\tau}}$ have prior distributions $\underline{\boldsymbol{\mu}} \sim N(0, \frac{1}{a_\mu} I_{b_\mu})$ and $\underline{\boldsymbol{\tau}} \sim N(0, \frac{1}{a_\tau} I_{b_\tau})$ respectively. A similar formulation of BCF for binary outcomes is used by Starling et al. (2020) with a Probit link function and targeted smoothing.[35] Starling et al. (2020) estimate relative risk, however the focus here will be difference in probabilities for comparability with existing treatment effect estimation methods.

However, a limitation of this approach, relative to standard Bayesian Causal Forests for continuous outcomes, is that the treatment effect, $sig(\mu(x_i, \hat{\pi}_i) + \tau(x_i)) - sig(\mu(x_i, \hat{\pi}_i))$, depends not only on the sum-of-trees $\tau(x_i)$, but also on $\mu(x_i, \hat{\pi}_i)$, and therefore this re-parameterization does not provide a framework in which the regularization of the treatment effect estimates is wholly specified through the prior on $\tau(x_i)$. A similar issue has previously been noted by Starling et al. (2020) in the estimation of relative risk. Shrinkage of τ does not imply shrinkage to homogeneous relative risk. Starling et al. (2020) refer to heterogeneity in relative risk arising due to heterogeneity in baseline risk as *structural heterogeneity*. Therefore, ideally the scale of the priors for μ and τ should be set by careful prior elicitation (Starling et al. 2020).

For Logit-BCF-BMA and Logit-BCF-IS the algorithm for obtaining the MAP and Laplace approximations is slightly different to Logit-BART-BMA and Logit-BART-IS because the μ and τ terminal nodes are regularized by different parameters a_μ and a_τ respectively. The posterior mean and interval calculations are the same as for BART-BMA and BART-IS ITEs and the CATE (means and intervals), except $W^{tr}_{(m)}$ and $W^{c}_{(m)}$ are replaced by $[W_{(\mu,m)} W_{(\tau,m)}]$ and $[W_{(\mu,m)} \mathbf{0}]$ respectively.[36] [37] The details for the calculation of the MAP by standard quasi-Newton methods are given in Appendix H.

5.3 Application to ACIC Data Challenge

The annual Atlantic Causal Inference Conference (ACIC) has run a data analysis competition for treatment effect estimation methods. BART and BCF have performed well in this competition (Dorie et al. 2019, Hahn et al. 2019).

Table 7 presents a comparison between BCF-MCMC,[38], BART-MCMC,[39], Causal Forests,[40] Probit-BART-MCMC,[41] Probit-BART-cause,[42] Logit-BART-IS,[43] and Logit-BCF-IS[44] applied to the publicly available data from the 2019 ACIC Data Challenge.[45] The results are restricted to the 1200 datasets in the low-dimensional category with less than 1000 observations and a binary dependent variable.[46] The RMSE and coverage are calculated using the true population ATE.

[35] Starling et al. (2020) implement Probit BCF using MCMC.

[36] $\mathbf{0}$ is a matrix of zeros of the same dimensions as $W_{(\tau,m)}$

[37] As in the case of BART-BMA and BART-IS, a viable alternative may be to apply a Gibbs sampler for draws from each model in the mixture.

[38] BCF was implemented using the **R** package **bcf** function for continuous outcomes because currently the software does not provide options for logit or probit based implementations. The number of burn-in draws was set to 2000 and the number of post-burn-in draws aws set to 2000. Each model contained the default number of 200 μ trees and 50 τ trees. All other parameters were set t ot he default values.

[39] BART-MCMC was implemented with 100 burn-in draws and 1000 post-burn-in draws. Each model contained the default number of 200 trees. All other parameters were set to their default value.

[40] Causal forests were implemented with the **R** package **grf**. The number of trees was set to 4000.

[41] Probit-BART was implemented using the **BART** package in **R**. The number of model draws was set to the default value of 1000 post-burn-in draws with 100 burn-in draws. The number of tree in each model was set to the default number of 50, and all other parameters were set to default values.

[42] BART-cause is an alternative MCMC implementation of BART for average treatment effect estimation available in the **R** package **bartCause**. This was implemented with 4,000 post-burn-in samples, 1000 burn-in samples, and 1 separate chains. The the rest of the parameters are set to the defaults (see the **dbarts** package function **bart2** for more details), with 75 trees per model.

[43] Logit-BART-IS was implemented with 5,000 sampled models, only 5 trees per model and 10,000 CATE samples (from the mixture of sampled models) for calculation of CATE intervals.

[44] Logit-BCF-IS was implemented with 5,000 sampled models, 5 μ trees and 5 τ trees per model and 10,000 CATE samples (from the mixture of sampled models) for calculation of CATE intervals.

[45] Results are not presented for BCF-BMA or BART-BMA, because the current implementations can require a large quantity of RAM, and this can lead to errors/crashes.

[46] The current implementations of BART-IS and BCF-IS are slow when applied to datasets with many observations. The methods presented in this chapter are designed for data with a binary dependent variable. See chapter 3 of this thesis for the results for ACIC 2019 datasets with continuous outcomes.

The standard Probit-BART-MCMC implementation produces the most accurate ATE estimates, however this method involved a large number of draws and was quite slow. Logit-BCF-IS produces impressive results given that each model contains only a small number of trees. The confidence intervals produced by Logit-BCF-IS are much wider than those produced by other methods. This may suggest that a larger number CATE samples is required from the mixture of 5,000 models.

	ATE		
Method	RMSE	Coverage	Length
BCF-MCMC	0.0486	0.850	0.174
BART-MCMC	0.0465	0.821	0.153
CF	0.0477	0.863	0.175
Probit-BART-MCMC	0.0427	0.879	0.169
BART-cause	0.0423	0.935	0.199
Logit-BART-IS	0.0555	0.813	0.199
Logit-BCF-IS	0.0452	0.913	0.292

Table 7: Results for ACIC Data Challenge low-dimensional datasets with less than 1000 observations and a binary dependent variable.

6 Example of General-BART-BMA and General-BART-IS for Censored Outcome Data

6.1 Tobit BART-BMA and Tobit BART-IS

The example in this subsection is an average of Bayesian Tobit models, with variables described by sums-of-trees. The tree structures have the standard BART prior.[47] The terminal node parameters have a normal prior distribution and there is an inverse gamma prior on the variance of the error term.[48]

$$\tau^2 = \sigma^{-2} \sim \Gamma(\frac{\nu}{2}, \frac{\nu\lambda}{2}) \ , \ \underline{\boldsymbol{\mu}} \sim N(0, \frac{\sigma^2}{a}) \ , \ or \ \underline{\boldsymbol{\mu}} \sim N(0, \frac{1}{a}\tau^{-2})$$

and the convenient Tobin reparameterization is $(\underline{\boldsymbol{\mu}}, \tau^2) \rightarrow (\boldsymbol{\alpha} = \underline{\boldsymbol{\mu}}\tau, \tau = (\tau^2)^{\frac{1}{2}})$. This gives

$$\alpha = \tau\beta \sim N(0, \frac{1}{a}I)$$

Let the covariate matrix, W be the set of binary variables indicating inclusion of observations in terminal nodes. The standard Tobit model framework is

$$y_i^* = row_i(W)\underline{\boldsymbol{\mu}} + \varepsilon_i \ , \ \varepsilon \sim i.i.d. \ N(0, \tau^{-2})$$

$$y_i = \max\{y_i^*, 0\} \ , \ i = 1, ..., n$$

See appendix I for details on how to implement the Tobit model using standard Laplace approximations. Chib (1992) outlines a number of approaches for implementation of Bayesian Tobit models, including Laplace approximations (fully exponential Laplace approximations, as outlined by Tierney & Kadane (1986)) and a Gibbs sampler.

An average of Tobit sum-of-tree models is obtainable by application of the general BART-BMA or BART-IS algorithms outlined in section 3 in combination with one of a number of potenrial Tobit approximation methods, including:

1. Use a Laplace approximation for the marginal likelihood and posterior distributions.

[47] I have provided options in the **safeBart** package for Tobit-BART-IS with draws from the Quadrianto & Ghahramani (2014) prior and the spike and tree prior (Rockova & van der Pas 2017).

[48] Chib (1992) used an uninformative prior for Bayesian Tobit. The normal prior is preferred here for the terminal nodes because this is the prior used by standard BART, and it is desirable to regularize the terminal node parameters.

2. Use a Laplace approximation for the marginal likelihood, and then apply a Gibbs sampler for each model in the mixture. [49]

A sum-of-tree Tobit model, based on gradient boosting, is used by Sigrist (2018) to predict defaults on loans made to Small and Medium Sized enterprises. Gradient-boosted Tobit outperforms Logit, Tobit, and a number of machine learning methods. BART can be viewed as a Bayesian alternative to gradient boosted trees as it involves sum-of-tree models. Therefore it is desirable to investigate the performance of a Tobit-BART implementation at predicting censored outcomes.

The example below is based on the simulations described by Sigrist (2018). The goal is prediction of censored outcomes out-of-sample. As in Sigrist (2018), the competing methods are be Tobit and binary classification methods logistic regression, Logit-BART-IS,[50] Probit-BART-MCMC, Logit-BART-MCMC,[51] and Random Forests.[52] The performance measures are the Brier Score and Area Under the Curve (AUC) for out of sample predictive probabilities of censored outcomes.

There are 30 uniformly distributed covariates, $X_1, ..., X_{30} \sim Unif(-1, 1)$, the latent outcome Y^*, and observed outcome Y are determined by the following data generating process:

$$Y^* = \sum_{k=1}^{f} 0.3(X_k)_+ + \sum_{k=1}^{3} \sum_{j=k+1}^{4} (X_k X_j)_+ + \varepsilon \ , \ \varepsilon \sim N(0, \sigma_\varepsilon^2)$$

$$Y = min(2.84, Y^*)$$

where $(x)_+ = max(x, 0)$, and $\sigma_\varepsilon = 0.7$.[53] The results are presented in Table 8. While Tobit-BART-IS outperforms the other methods, the results are somewhat underwhelming. One possible explanation for this is that the outcome should be transformed and the prior on the terminal node parameters should be carefully calibrated so that coefficients are regularized towards zero or predictive probabilities are regularized towards the training sample proportion of censored outcomes.[54]

Method	Brier	AUC
Tobit-BART-IS	**0.046**	**0.776**
Tobit	0.049	0.613
Logit-BART-IS	0.047	0.740
Probit-BART-MCMC	**0.046**	0.750
Logit-BART-MCMC	0.047	0.721
Logistic Regression	0.051	0.610
RF	0.047	0.758

Table 8: Results for Tobit-BART-IS simulation study.

7 Conclusion

7.1 Summary

This chapter outlines a generalization of BART to a wide range of model settings. This approach builds on the algorithms introduced in chapter 2 and existing methods for approximate inference and calculation of model

[49] Chib (1992) describes a Gibbs sampler for Tobit. Perhaps an alternative based on the sampler of Polson et al. (2013) is applicable.

[50] Logit-BART-IS and Tobit-BART-IS were implemented with 20,000 draws and 5 trees per model.

[51] Probit-BART-MCMC and Logit-BART-MCMC were implemented using the **BART** package in **R** using 5000 burn-in draws and 10,000 post burn-in draws.

[52] Random Forests was implemented using the **ranger** package in **R** with 10,000 trees.

[53] The upper bound and standard deviation are those chosen by Sigrist (2018). Unlike the simulations presented by Sigrist (2018), the simulations presented here do not involve data censoring being determined by a latent variable that has a different error term to the error term for the observed outcome.

[54] One possibility would be to add an intercept to the sum-of-tree model, demean the outcome, and set a, ν and λ such that the prior predicts observations to lie in the training data range with high probability.

evidence. As an example, the approach is applied to the implementation of Logit-BART. The approach is validated by the fact that Logit-BART-IS and Logit-BART-BMA produce similar results to existing MCMC implementations of Probit-BART and Logit-BART.

Depending on computational resources and the speed of approximate inference methods such as Laplace approximations, the new methods may provide fast alternatives to MCMC-based approaches. The general BART-IS algorithm is highly parallelizable.

The methods outlined in this chapter have some limitations. There is no guarantee that the BART-BMA search algorithm will be particularly effective in searching the model space, and in some Logit-BART-BMA examples the current implementation is prohibitively slow if model search parameters are not appropriately adjusted. The BART-IS model sampler only takes a small sample from the large model space, and does not adapt to find models with higher posterior probability.[55] Therefore it is possible that none of the sampled models are similar to the "true" model.

Despite the potential limitations, the algorithms described in this chapter, particularly BART-IS, are of practical use to researchers seeking a quick and dirty approach to implementation of generalizations of BART. Simple BART-IS implementations can provide useful benchmarks for testing the accuracy of new MCMC-based implementations of similar BART model frameworks.

7.2 Future Research: Multinomial Logit, Poisson Regression, and Other Generalizations

Poisson regression and multinomial logit can be implemented with standard Laplace approximations (Madigan et al. 2005, Cawley et al. 2007, Silverman et al. 2019). Integrated Nested Laplace Approximations (Rue et al. 2009) can be used to implement a range of models including multinomial logit[56] and Poisson regression (and allows for hierarchical priors, e.g. mixed logit) and provides accurate calculations of the marginal likelihood. The general framework introduced in section 3 can therefore be extended to a wide variety of settings.[57]

However, a key requirement is that the calculation of the marginal likelihood and posterior inference are computationally efficient. The speed of the BART-IS or BART-BMA based implementations will depend on the choice of methods. The construction of residuals for the changepoint detection algorithm in BART-BMA is not straightforward for all model settings, and the arbitrary model search algorithm is not guaranteed to perform well outside of the linear regression context for which it was originally designed. Therefore the BART-IS framework is more generalizable and is recommended above the BART-BMA framework for application of BART to a wider class of models.

A possible approach for multinomial logit BART is to use Integrated Nested Laplace Approximations to calculate the marginal likelihoods, and then use a variation of the sampler described by Polson et al. (2013).[58]

References

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

Albert, J. H. & Chib, S. (1993), 'Bayesian analysis of binary and polychotomous response data', *Journal of the American statistical Association* **88**(422), 669–679.

Athey, S., Tibshirani, J., Wager, S. et al. (2019), 'Generalized random forests', *The Annals of Statistics* **47**(2), 1148–1178.

[55] However, as noted in chapter 2, a Bayesian Adaptive Sampling approach to sampling of BART models is an interesting topic for future research (Clyde et al. 2011). However, it is not obvious how to proceed in constructing such a sampler.

[56] Multinomial logit is implementable using the multinomial-Poisson transform (Baker 1994).

[57] See Barber et al. (2016) for some general asymptotic results on the use of the marginal likelihood for model selection.

[58] Linderman et al. (2015) describe this sampler for multinomial logit within a larger model. Similarly multinomial-Logit-BCF-BMA and multinomial-Logit-BCF-IS are possible extensions and these methods would produce estimates of treatment effects on probabilities of categories.

Baker, S. G. (1994), 'The multinomial-poisson transformation', *Journal of the Royal Statistical Society: Series D (The Statistician)* **43**(4), 495–504.

Barber, R. F., Drton, M. & Tan, K. M. (2016), Laplace approximation in high-dimensional bayesian regression, *in* 'Statistical Analysis for High-Dimensional Data', Springer, pp. 15–36.

Bishop, C. M. (2006), *Pattern recognition and machine learning*, springer.

Bivand, R., Gómez-Rubio, V., Rue, H. et al. (2015), 'Spatial data analysis with r-inla with some extensions', *Journal of Statistical Software* **63**(i20).

Bivand, R. S., Gómez-Rubio, V. & Rue, H. (2014), 'Approximate bayesian inference for spatial econometrics models', *Spatial Statistics* **9**, 146–165.

Carnegie, N., Harada, M., Dorie, V. & Hill, J. (2015), 'treatsens: sensitivity analysis for causal inference', *R package* .

Cawley, G. C., Talbot, N. L. & Girolami, M. (2007), Sparse multinomial logistic regression via bayesian l1 regularisation, *in* 'Advances in neural information processing systems', pp. 209–216.

Chib, S. (1992), 'Bayes inference in the tobit censored regression model', *Journal of Econometrics* **51**(1-2), 79–99.

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

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

Choi, H. M., Hobert, J. P. et al. (2013), 'The polya-gamma gibbs sampler for bayesian logistic regression is uniformly ergodic', *Electronic Journal of Statistics* **7**, 2054–2064.

Chopin, N., Ridgway, J. et al. (2017), 'Leave pima indians alone: binary regression as a benchmark for bayesian computation', *Statistical Science* **32**(1), 64–87.

Clyde, M. A., Ghosh, J. & Littman, M. L. (2011), 'Bayesian adaptive sampling for variable selection and model averaging', *Journal of Computational and Graphical Statistics* **20**(1), 80–101.

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

Dua, D. & Graff, C. (2017), 'UCI machine learning repository'.
URL: *http://archive.ics.uci.edu/ml*

Freund, Y. & Schapire, R. E. (1995), A desicion-theoretic generalization of on-line learning and an application to boosting, *in* 'European conference on computational learning theory', Springer, pp. 23–37.

Freund, Y., Schapire, R. E. et al. (1996), Experiments with a new boosting algorithm, Citeseer.

Friedman, J. H. & Meulman, J. J. (2003), 'Multiple additive regression trees with application in epidemiology', *Statistics in medicine* **22**(9), 1365–1381.

Friedman, J., Hastie, T., Tibshirani, R. et al. (2000), 'Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)', *The annals of statistics* **28**(2), 337–407.

Friel, N. & Wyse, J. (2012), 'Estimating the evidence–a review', *Statistica Neerlandica* **66**(3), 288–308.

Gómez-Rubio, V., Bivand, R. S. & Rue, H. (2020), 'Bayesian model averaging with the integrated nested laplace approximation', *Econometrics* **8**(2), 23.

Gómez-Rubio, V. & Rue, H. (2018), 'Markov chain monte carlo with the integrated nested laplace approximation', *Statistics and Computing* **28**(5), 1033–1051.

Gramacy, R. B., Polson, N. G. et al. (2012), 'Simulation-based regularized logistic regression', *Bayesian Analysis* **7**(3), 567–590.

Green, D. P. & Kern, H. L. (2012), 'Modeling heterogeneous treatment effects in survey experiments with bayesian additive regression trees', *Public opinion quarterly* **76**(3), 491–511.

Hahn, P. R., Dorie, V. & Murray, J. S. (2019), 'Atlantic causal inference conference (acic) data analysis challenge 2017', *arXiv preprint arXiv:1905.09515* .

Hahn, P. R., Murray, J. S. & Carvalho, C. M. (2017), 'Bayesian regression tree models for causal inference: regularization, confounding, and heterogeneous effects'.

Hahn, P. R., Murray, J. S., Carvalho, C. M. et al. (2020), 'Bayesian regression tree models for causal inference: regularization, confounding, and heterogeneous effects', *Bayesian Analysis* .

Hernández, B., Raftery, A. E., Pennington, S. R. & Parnell, A. C. (2018), 'Bayesian additive regression trees using bayesian model averaging', *Statistics and Computing* **28**(4), 869–890.

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

Hill, J., Linero, A. & Murray, J. (2020), 'Bayesian additive regression trees: A review and look forward', *Annual Review of Statistics and Its Application* **7**.

Holland, P. W. (1986), 'Statistics and causal inference', *Journal of the American statistical Association* **81**(396), 945–960.

Imbens, G. W. & Rubin, D. B. (2015), *Causal inference in statistics, social, and biomedical sciences*, Cambridge University Press.

Killick, R., Fearnhead, P. & Eckley, I. A. (2012), 'Optimal detection of changepoints with a linear computational cost', *Journal of the American Statistical Association* **107**(500), 1590–1598.

Kindo, B. P., Wang, H., Hanson, T. & Peña, E. A. (2016), 'Bayesian quantile additive regression trees', *arXiv preprint arXiv:1607.02676* .

Kindo, B. P., Wang, H. & Peña, E. A. (2016), 'Multinomial probit bayesian additive regression trees', *Stat* **5**(1), 119–131.

Kleinberg, J., Ludwig, J., Mullainathan, S. & Obermeyer, Z. (2015), 'Prediction policy problems', *American Economic Review* **105**(5), 491–95.

Kohavi, R. (1996), Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid., *in* 'Kdd', Vol. 96, pp. 202–207.

Künzel, S. R., Sekhon, J. S., Bickel, P. J. & Yu, B. (2019), 'Metalearners for estimating heterogeneous treatment effects using machine learning', *Proceedings of the national academy of sciences* **116**(10), 4156–4165.

Linderman, S., Johnson, M. J. & Adams, R. P. (2015), Dependent multinomial models made easy: Stick-breaking with the pólya-gamma augmentation, *in* 'Advances in Neural Information Processing Systems', pp. 3456–3464.

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

Linero, A. R., Sinha, D. & Lipsitz, S. R. (2019), 'Semiparametric mixed-scale models using shared bayesian forests', *Biometrics* .

Madigan, D., Genkin, A., Lewis, D. D. & Fradkin, D. (2005), Bayesian multinomial logistic regression for author identification, *in* 'AIP conference proceedings', Vol. 803, American Institute of Physics, pp. 509–516.

Moran, G. E., Ročková, V., George, E. I. et al. (2018), 'Variance prior forms for high-dimensional bayesian variable selection', *Bayesian Analysis* pp. 1091–1119.

Moro, S., Cortez, P. & Rita, P. (2014), 'A data-driven approach to predict the success of bank telemarketing', *Decision Support Systems* **62**, 22–31.

Murphy, K. P. (2012), *Machine learning: a probabilistic perspective*, MIT press.

Murray, J. S. (2017), 'Log-linear bayesian additive regression trees for categorical and count responses', *arXiv preprint arXiv:1701.01503* .

Neyman, J. (1923), 'Sur les applications de la theorie des probabilites aux experiences agricoles: essai des principes (masters thesis); justification of applications of the calculus of probabilities to the solutions of certain questions in agricultural experimentation. excerpts english translation (reprinted)', *Stat Sci* **5**, 463–472.

Oprescu, M., Syrgkanis, V. & Wu, Z. S. (2018), 'Orthogonal random forest for causal inference', *arXiv preprint arXiv:1806.03467* .

Polson, N. G., Scott, J. G. & Windle, J. (2013), 'Bayesian inference for logistic models using pólya–gamma latent variables', *Journal of the American statistical Association* **108**(504), 1339–1349.

Quadrianto, N. & Ghahramani, Z. (2014), 'A very simple safe-bayesian random forest', *IEEE transactions on pattern analysis and machine intelligence* **37**(6), 1297–1303.

Quinlan, J. R. (1987), 'Simplifying decision trees', *International journal of man-machine studies* **27**(3), 221–234.

Rockova, V. & van der Pas, S. (2017), 'Posterior concentration for bayesian regression trees and their ensembles', *arXiv preprint arXiv:1708.08734* .

Rubin, D. B. (1974), 'Estimating causal effects of treatments in randomized and nonrandomized studies.', *Journal of educational Psychology* **66**(5), 688.

Rue, H., Martino, S. & Chopin, N. (2009), 'Approximate bayesian inference for latent gaussian models by using integrated nested laplace approximations', *Journal of the royal statistical society: Series b (statistical methodology)* **71**(2), 319–392.

Sakar, C. O., Polat, S. O., Katircioglu, M. & Kastro, Y. (2019), 'Real-time prediction of online shoppers' purchasing intention using multilayer perceptron and lstm recurrent neural networks', *Neural Computing and Applications* **31**(10), 6893–6908.

Sigrist, F. (2018), 'Gradient and newton boosting for classification and regression', *arXiv preprint arXiv:1808.03064* .

Silverman, J. D., Roche, K., Holmes, Z. C., David, L. A. & Mukherjee, S. (2019), 'Bayesian multinomial logistic normal models through marginally latent matrix-t processes', *arXiv preprint arXiv:1903.11695* .

Sparapani, R. A., Logan, B. R., McCulloch, R. E. & Laud, P. W. (2016), 'Nonparametric survival analysis using bayesian additive regression trees (bart)', *Statistics in medicine* **35**(16), 2741–2753.

Sparapani, R. A., Rein, L. E., Tarima, S. S., Jackson, T. A. & Meurer, J. R. (2018), 'Non-parametric recurrent events analysis with bart and an application to the hospital admissions of patients with diabetes', *Biostatistics* .

Sparapani, R., Logan, B. R., McCulloch, R. E. & Laud, P. W. (2019), 'Nonparametric competing risks analysis using bayesian additive regression trees', *Statistical methods in medical research* p. 0962280218822140.

Spiegelhalter, D. J. & Lauritzen, S. L. (1990), 'Sequential updating of conditional probabilities on directed graphical structures', *Networks* **20**(5), 579–605.

Starling, J. E., Murray, J. S., Lohr, P. A., Aiken, A. R., Carvalho, C. M. & Scott, J. G. (2020), 'Targeted smooth bayesian causal forests: An analysis of heterogeneous treatment effects for simultaneous versus interval medical abortion regimens over gestation', *arXiv preprint arXiv:1905.09405* .

Tan, Y. V. & Roy, J. (2019), 'Bayesian additive regression trees and the general bart model', *arXiv preprint arXiv:1901.07504* .

Tierney, L. & Kadane, J. B. (1986), 'Accurate approximations for posterior moments and marginal densities', *Journal of the american statistical association* **81**(393), 82–86.

Van Der Putten, P. & van Someren, M. (2000), Coil challenge 2000: The insurance company case, Technical report, Technical Report 2000–09, Leiden Institute of Advanced Computer Science ....

Wendling, T., Jung, K., Callahan, A., Schuler, A., Shah, N. & Gallego, B. (2018), 'Comparing methods for estimation of heterogeneous treatment effects using observational data from health care databases', *Statistics in medicine* **37**(23), 3309–3324.

Yeh, I.-C. & Lien, C.-h. (2009), 'The comparisons of data mining techniques for the predictive accuracy of probability of default of credit card clients', *Expert Systems with Applications* **36**(2), 2473–2480.

Zhang, J. L. & Härdle, W. K. (2010), 'The bayesian additive classification tree applied to credit risk modelling', *Computational Statistics & Data Analysis* **54**(5), 1197–1205.

A Standard Newton-Raphson algorithm for finding the MAP of Bayesian Logistic Regression

Require parameter value, e.g. $a = 0.01$
Initialize $\underline{\boldsymbol{\mu}} = \mathbf{0}$
repeat
  $p_i = \frac{1}{1+e^{-W_i\underline{\boldsymbol{\mu}}}}$ for $i = 1, ..., n$
  $S = diag(p_i(1-p_i))$
  $\boldsymbol{g} = W^T(\boldsymbol{p} - \boldsymbol{y}) + a\underline{\boldsymbol{\mu}}$
  $H = W^T SW + aI_b$
  $\underline{\boldsymbol{\mu}}_{new} = \underline{\boldsymbol{\mu}}_{old} - H^{-1}\boldsymbol{g}$
until *convergence*;

Algorithm 4: Newton's method for obtaining the mode (MAP) of the posterior approximation

B Marginal Likelihood Approximation

The Laplace Approximation gives the following approximation for the marginal likelihood (using the normalization constant of the multivariate Gaussian distribution)

$$p(\boldsymbol{y}|\mathcal{T}^{(m)}) \approx e^{\log(p(\underline{\boldsymbol{\mu}}_{MAP,(m)}, \boldsymbol{y}))}(2\pi)^{b^{(m)}/2}|H_{MAP,(m)}|^{-1/2}$$

$$= (\frac{1}{a})^{-\frac{b^{(m)}}{2}} \prod_{i=1}^{N} \left[\left(\frac{e^{row_i(W_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)}}}{1+e^{row_i(W_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)}}} \right)^{y_i} \left(\frac{1}{1+e^{row_i(W_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)}}} \right)^{1-y_i} \right] |H_{MAP,(m)}|^{-1/2}$$

and the log of the marginal likelihood is approximated by:

$$\frac{b^{(m)}}{2}\log(a) + \sum_{i=1}^{N} \left[y_i row_i(W_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)} - \log\left(1+e^{row_i(W_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)}}\right)\right] - \frac{1}{2}log(|H_{MAP,(m)}|)$$

C Applying Laplace's Method Approximation Twice to Approximate Posterior Mean Probability

Tierney & Kadane (1986) describe an approach for approximating the posterior mean of any smooth unimodal function of the parameters, $g(\theta)$.[59] This involves the observation that the posterior mean can be approximated by first applying Laplace's method to find the mode of the integral in the numerator of the posterior mean of the function.

$$\mathbb{E}[g] = \mathbb{E}[g(\theta|X)] = \frac{\int g(\theta)e^{\mathcal{L}(\theta)}\pi(\theta)d\theta}{\int e^{\mathcal{L}(\theta)}\pi(\theta)d\theta}$$

where $\mathcal{L}$ is the log likelihood function.

First, the MAP of the posterior for θ is obtained by Newton's method. Then Laplace's method is used to obtain an approximation for the denominator integral.

Then this is combined with a Laplace approximation for the integral in the numerator.

Let $L = \log(\pi(\theta)) + \frac{\mathcal{L}(\theta)}{n}$ and $L^* = \log(g(\theta)) + \log(\pi(\theta)) + \frac{\mathcal{L}(\theta)}{n}$. Then

$$\mathbb{E}[g] = \mathbb{E}[g(\theta|X)] = \frac{\int e^{nL^*}d\theta}{\int e^{nL}d\theta}$$

Let $\hat{\theta} = \theta_{MAP}$ be the mode of L. Similarly let $\hat{\theta}^*$ be the mode of L^*. Then, taking the ratio of the two Laplace approximations gives:

$$\hat{E}_n[g] = \left(\frac{\det(H^{*-1})}{\det(H^{-1})}\right)^{1/2} \exp\{n(L^*(\hat{\theta}^*) - L^*(\hat{\theta}))\}$$

where H and H^* are the negatives of the Hessians of L and L^* respectively (i.e. the Hessians of the negative log likelihood). The error is of order $\mathcal{O}(n^{-2})$.

This can in principle be applied to the logit model with $g(\underline{\boldsymbol{\mu}}_{(m)}) = \frac{e^{W_{*,(m)}\underline{\boldsymbol{\mu}}_{(m)}}}{1+e^{W_{*,(m)}\underline{\boldsymbol{\mu}}_{(m)}}}$

D Outline of Monte Carlo Approximation for Logit-BART-BMA and Logit-BART-IS

D.1 Monte Carlo Approximation of Posterior Predictive Mean Probability

Two possible Monte Carlo approximation approaches are:

1. Approximate each integral separately, and then average by the model posterior probability. i.e. For each model, obtain a large number S of samples of $\underline{\boldsymbol{\mu}}_{(m),1}, ..., \underline{\boldsymbol{\mu}}_{(m),S}$ from the approximate distribution $\mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)})$ and estimate the posterior predicted probability of $y_* = 1$ for model m as $\frac{1}{S}\sum_{s=1}^{S} \frac{e^{W_{*,(m)}\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{W_{*,(m)}\underline{\boldsymbol{\mu}}_{(m),s}}}$ and then the model averaged probability is:

$$\sum_{m=1}^{M} p(\mathcal{T}_m|\boldsymbol{y})\frac{1}{S}\sum_{s=1}^{S}\frac{e^{W_{*,(m)}\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{W_{*,(m)}\underline{\boldsymbol{\mu}}_{(m),s}}}$$

2. Take a large number, S, of samples from the mixture of multivariate normal distributions $\underline{\boldsymbol{\mu}}|\boldsymbol{y} \sim \sum_{m=1}^{M}\mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)})p(\mathcal{T}_m|\boldsymbol{y})$. Note that this involves sampling from each model's normal approximation with probability $p(\mathcal{T}_m|\boldsymbol{y})$, and for each model the sampled vector $\underline{\boldsymbol{\mu}}$ has a different interpretation and can have different dimensions because the sum-of-tree structures differ across models. Then the estimate is

$$\frac{1}{S}\sum_{s=1}^{S}\frac{e^{W_{*,(m)}\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{W_{*,(m)}\underline{\boldsymbol{\mu}}_{(m),s}}}$$

[59] The function is also required to be nonzero, and preferably positive, but it is possible to add a large constant or take the negative

D.2 Monte Carlo Approximation of Credible Intervals for Posterior Predictive Probability

Take a large number, S, samples from the mixture of multivariate normal distributions

$$\underline{\boldsymbol{\mu}}|\boldsymbol{y} \sim \sum_{m=1}^{M} \mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)}, H_{(m)}^{-1}) p(\mathcal{T}_m|\boldsymbol{y})$$

Note that this involves sampling from each model's normal approximation with probability $p(\mathcal{T}_m|\boldsymbol{y})$, and for each model the sampled vector $\underline{\boldsymbol{\mu}}$ has a different interpretation and can have different dimensions because the sum-of-tree structures differ across models.

For each draw, calculate the probability $\frac{e^{W_{*,(m)}\underline{\mu}_{(m),s}}}{1+e^{W_{*,(m)}\underline{\mu}_{(m),s}}}$. Then obtain the desired quantiles of the S probabilities. e.g. For a 95% interval, find L_b and U_b such that $\frac{1}{S}\sum_{s=1}^{S} \mathbb{I}\left(\frac{e^{W_{*,(m)}\underline{\mu}_{(m),s}}}{1+e^{W_{*,(m)}\underline{\mu}_{(m),s}}} < L_b\right) = 0.025$ and $\frac{1}{S}\sum_{s=1}^{S} \mathbb{I}\left(\frac{e^{W_{*,(m)}\underline{\mu}_{(m),s}}}{1+e^{W_{*,(m)}\underline{\mu}_{(m),s}}} < U_b\right) = 0.975$.

E Root-finding Approximation of Credible Intervals for Posterior Predictive Probability

Using the sigmoid (logistic) function, $sig()$, we require L_b such that

$$\sum_{m=1}^{M} \left(\int_{-\infty}^{\infty} \mathbb{I}\left(sig(\alpha_{(m)}) < L_b\right) p(\alpha_{(m)}|\psi_{\alpha,(m)}, \sigma^2_{\alpha,(m)}) d\alpha_{(m)} \right) p(\mathcal{T}_m|\boldsymbol{y})$$

$$= \sum_{m=1}^{M} \Phi\left(\frac{sig^{-1}(L_b) - \psi_{\alpha,(m)}}{\sigma_{\alpha,(m)}}\right) p(\mathcal{T}_m|\boldsymbol{y}) = 0.025$$

A simple root finding algorithm (e.g. bisection) can be used to find $c = sig^{-1}(L_b)$. Then L_b is obtained from $sig(c)$. Similarly, for the upper bound, we require U_b such that $\sum_{m=1}^{M} \Phi\left(\frac{sig^{-1}(U_b) - \psi_{\alpha,(m)}}{\sigma_{\alpha,(m)}}\right) p(\mathcal{T}_m|\boldsymbol{y}) = 0.975$, which can be obtained by a root-finding algorithm.

F Alternative methods for constructing Logit-BART-BMA Residuals

F.1 Constructing Residuals using Predicted Probabilities or MAP Estimates

Three simple methods for calculation of residuals are::

- A naive approach resulting from an unedited model search algorithm using the residuals $y_i - \widehat{Pr}(y_i = 1)$ where $\widehat{Pr}(y_i = 1)$ is the estimated probability from the model. However, the tree will be appended to a sum-of-trees modelling the latent outcome, which is a continuous variable that is not restricted to be between 0 and 1.

- Residuals can be calculated for the latent outcome U by beginning with $U_i = 3.1$ if $y_i = 1$ and $U = -3.1$ if $y_i = 0$ (or a similar number that gives a probability close to 1 or 0) and for each model obtaining $U_i - row_i(W)\underline{\boldsymbol{\mu}}_{MAP}$ as the residual to be used in the changepoint detection algorithm in the next round, where $row_i(W)\underline{\boldsymbol{\mu}}_{MAP}$ is the MAP prediction of the latent outcome y^*.

- An even less computationally burdensome approach would be to only search for potential splits before the first round of the algorithm. This involves applying the changepoint detection algorithm to the latent outcome U defined by $U_i = 3.1$ if $y_i = 1$ and $U = -3.1$ if $y_i = 0$. Then keep these potential split points for all future rounds of the algorithm (i.e. do not apply the changepoint algorithm again).

F.2 Arbitrary fixed grid of splits, without residuals

The changepoint detection algorithm can be replaced by an alternative method for reducing the number of potential splitting points.

- Propose an arbitrary deterministic grid of splitting points, possibly after applying a Probability Integral Transform using the Empirical Distribution Function of the residuals, and proceed to use these splits in the rest of the algorithm without applying a changepoint detection algorithm. This is likely to be very slow, particularly for high-dimensional data, unless the set of potential splits for each variable is severely restricted, which may compromise the ability of these methods to find models close to the true data generating process.
- Alternatively, the grid of points for each variable can be found by first applying some other tree-based method or search algorithm. For example, one could use standard BART-BMA or the simple Probit-BART-BMA and save the splitting points to use in Logit-BART-BMA. [60]

G Technical Details for Logit-BART-BMA and Logit-BART-IS Treatment Effect Estimation

G.1 Estimation of Mean of Posterior Distribution of Individual Treatment Effects

Beginning with the Laplace approximations of posterior distributions of the terminal node coefficients outlined in section 4.3,
$\underline{\boldsymbol{\mu}}_{(m)}|\boldsymbol{y}, \mathcal{T}_m \sim \mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)})$, the goal is to estimate the expected difference in the the probability $y_* = 1$ for an individual with and without treatment, i.e. $T_* = 1$ and $T_* = 0$, conditioning on the same set of values for other covariates x_* in both cases. i.e. Estimate $\mathbb{E}[y_*|x_*, T_* = 1] - \mathbb{E}[y_*|x_*, T_* = 0]$.

When treatment is a splitting variable in the sum-of-tree model, the terminal nodes that individual i is allocated to when we set $T_i = 1$ can be different to the terminal nodes for $T_i = 0$. Therefore the variables indicating inclusion in terminal nodes will be different in these two scenarios. Denote these two sets of indicator variables as $row_i(W^{tr})$ and $row_i(W^c)$ for allocation to treatment and control respectively. Then for a new observation, with original covariate vector x_*, we estimate the expected difference in probabilities for $row_*(W^{tr})$ and $row_*(W^c)$. Therefore the expected treatment effect is:

$$\sum_{m=1}^{M} \left[\int \left(sig(row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m)}) - sig(row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m)}) \right) p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)}) d\underline{\boldsymbol{\mu}}_{(m)} \right] p(\mathcal{T}_m|\boldsymbol{y})$$

this can also be separated into two integrals

$$\sum_{m=1}^{M} \left[\int sig(row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m)}) p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)}) d\underline{\boldsymbol{\mu}}_{(m)} - \int sig(row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m)}) p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)}) d\underline{\boldsymbol{\mu}}_{(m)} \right] p(\mathcal{T}_m|\boldsymbol{y})$$

where *sig* denotes the sigmoid function (i.e. logistic). We consider two possible approaches: Monte Carlo and Probit approximation.

G.1.1 Monte Carlo Approximation of Expected ITE

Two possible approaches to Monte Carlo Approximation of the Expected ITE are:

1. Approximate each integral and then average by the model posterior probability. i.e. For each model, obtain a large number S of samples of $\underline{\boldsymbol{\mu}}_{(m),1}, ..., \underline{\boldsymbol{\mu}}_{(m),S}$ from the approximate distribution $\mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)})$

[60] i.e. Save all the splits that were used or suggested in a first step BART-BMA (as if the outcome were continuous) or approximate Probit-BART-BMA

and estimate the difference in probabilities for model m. Then the model averaged difference in probabilities (treated minus untreated) is:

$$\sum_{m=1}^{M} p(\mathcal{T}_m|\boldsymbol{y}) \frac{1}{S} \sum_{s=1}^{S} \left[\frac{e^{row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}} - \frac{e^{row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}} \right]$$

2. Take a large number, S, samples from the mixture of multivariate normal distributions $\underline{\boldsymbol{\mu}}|\boldsymbol{y} \sim \sum_{m=1}^{M} \mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)}) p(\mathcal{T}_m|\boldsymbol{y})$. Note that this involves sampling from each model's normal approximation with probability $p(\mathcal{T}_m|\boldsymbol{y})$, and for each model the sampled vector $\underline{\boldsymbol{\mu}}$ has a different interpretation and can have different dimensions because the sum-of-tree structures differ across models. Then the estimate is

$$\frac{1}{S} \sum_{s=1}^{S} \left[\frac{e^{row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}} - \frac{e^{row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}} \right]$$

G.1.2 Probit Approximation of Expected ITE

The sigmoid (logistic) function can be approximated by a normal CDF:

$$\sum_{m=1}^{M} \left[\int sig(row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m)}) p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)}) d\underline{\boldsymbol{\mu}}_{(m)} - \int sig(row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m)}) p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)}) d\underline{\boldsymbol{\mu}}_{(m)} \right] p(\mathcal{T}_m|\boldsymbol{y})$$

$$\approx \sum_{m=1}^{M} \left[\int \Phi(row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m)}) p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)}) d\underline{\boldsymbol{\mu}}_{(m)} - \int \Phi(row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m)}) p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)}) d\underline{\boldsymbol{\mu}}_{(m)} \right] p(\mathcal{T}_m|\boldsymbol{y})$$

Let $\psi_{\alpha,(m,tr)} = row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)}$ and $\sigma^2_{\alpha,(m,tr)} = row_*(W^{tr}_{(m)})H^{-1}_{(m)} row_*(W^{tr}_{(m)})^T$ and $\psi_{\alpha,(m,c)} = row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)}$ and $\sigma^2_{\alpha,(m,c)} = row_*(W^{c}_{(m)})H^{-1}_{(m)} row_*(W^{c}_{(m)})^T$. Then $\alpha_{(m,tr)} = row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m)} \sim \mathcal{N}(\psi_{\alpha,(m,tr)}, \sigma^2_{\alpha,(m,tr)})$ and $\alpha_{(m,c)} = row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m)} \sim \mathcal{N}(\psi_{\alpha,(m,c)}, \sigma^2_{\alpha,(m,c)})$

Then the integrals can be rewritten as one dimensional integrals, and the expected ITE is:[61]

$$\sum_{m=1}^{M} \left[\int \Phi(\alpha_{(m,tr)}) p(\alpha_{(m,tr)}|\psi_{\alpha,(m,tr)}, \sigma^2_{\alpha,(m,tr)}) d\alpha_{(m,tr)} - \int \Phi(\alpha_{(m,c)}) p(\alpha_{(m,c)}|\psi_{\alpha,(m,c)}, \sigma^2_{\alpha,(m,c)}) d\alpha_{(m,c)} \right] p(\mathcal{T}_m|\boldsymbol{y})$$

$$= \sum_{m=1}^{M} \left[\Phi\left(\frac{\psi_{\alpha,(m,tr)}}{\sqrt{1+\sigma^2_{\alpha,(m,tr)}}} \right) - \Phi\left(\frac{\psi_{\alpha,(m,c)}}{\sqrt{1+\sigma^2_{\alpha,(m,c)}}} \right) \right] p(\mathcal{T}_m|\boldsymbol{y})$$

G.1.3 Monte Carlo Approximation of ITE Intervals

Take a large number, S, of samples from the mixture of multivariate normal distributions $\underline{\boldsymbol{\mu}}|\boldsymbol{y} \sim \sum_{m=1}^{M} \mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)}) p(\mathcal{T}_m|\boldsymbol{y})$. Note that this involves sampling from each model's normal approximation with probability $p(\mathcal{T}_m|\boldsymbol{y})$, and for each model the sampled vector $\underline{\boldsymbol{\mu}}$ has a different interpretation and can have different dimensions because the sum-of-tree structures differ across models.

For each sample, s calculate the difference in probabilities under treatment and control group allocation (i.e. for $W^{tr}_{(m)}$ and $W^{c}_{(m)}$), and then find the relevant quantiles.

For example, for a 95% interval, find L_b such that

[61] For closer approximations to logistic probabilities, this can be replaced by $\sum_{m=1}^{M} \left[\Phi\left(\frac{\psi_{\alpha,(m,tr)}}{\sqrt{\frac{8}{\pi}+\sigma^2_{\alpha,(m,tr)}}} \right) - \Phi\left(\frac{\psi_{\alpha,(m,c)}}{\sqrt{\frac{8}{\pi}+\sigma^2_{\alpha,(m,c)}}} \right) \right] p(\mathcal{T}_m|\boldsymbol{y})$.

$$\frac{1}{S}\sum_{s=1}^{S}\mathbb{I}\Big\{\frac{e^{row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}-\frac{e^{row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}<L_b\Big\}=0.025$$

and find U_b such that $\frac{1}{S}\sum_{s=1}^{S}\mathbb{I}\Big\{\frac{e^{row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}-\frac{e^{row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}<U_b\Big\}=0.975.$

G.1.4 Monte Carlo Approximation of ITE Interval, reducing the dimension of the integral

Unlike in section E, the interval for the ITE does not have an obvious closed form obtainable from a Probit approximation. However, it is still possible to reduce the dimension of the integral, such that when the integral is approximated by Monte Carlo methods, draws can be made from univariate or bivariate normal distributions (instead of potentially much higher dimensional draws of $\underline{\boldsymbol{\mu}}_{(m)}$.

The goal is to find L_b defined in the following formula

$$\sum_{m=1}^{M}\left[\int_{-\infty}^{\infty}\mathbb{I}\left(sig(\alpha_{(m,tr)})-sig(\alpha_{(m,c)})<L_b\right)p(\alpha_{(m,tr)},\alpha_{(m,c)}|\psi_{\alpha,(m,tr)},\sigma^2_{\alpha,(m,tr)},\psi_{\alpha,(m,c)},\sigma^2_{\alpha,(m,c)})d\alpha_{(m,tr)}d\alpha_{(m,c)}\right]p(\mathcal{T}_m|\boldsymbol{y})$$

where the variables and parameters are defined as in section E. Note that $(\alpha_{(m,tr)},\alpha_{(m,tr)})$ has the following bivariate normal distribution:

$$\begin{bmatrix}\alpha_{(m,tr)}\\ \alpha_{(m,c)}\end{bmatrix}=\begin{bmatrix}row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m)}\\ row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m)}\end{bmatrix}$$
$$\sim\mathcal{N}\left(\begin{bmatrix}row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)}\\ row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)}\end{bmatrix},\begin{bmatrix}row_*(W^{tr}_{(m)})\\ row_*(W^{c}_{(m)})\end{bmatrix}H^{-1}\begin{bmatrix}row_*(W^{tr}_{(m)})^T & row_*(W^{c}_{(m)})^T\end{bmatrix}\right)$$

It is possible to take S draws from the model weighted average of bivariate normal distributions (i.e. draw from each model's bivariate normal distribution with probability equal to the posterior model probability), and for each draw, s, calculate $sig(\alpha_{(m,tr),s})-sig(\alpha_{(m,c),s})$ and then take obtain the desired quantiles of the draws.

However, it is also possible to reduce the integrals to one-dimensional integrals.

Note that the conditional distribution of $\alpha_{(m,tr)}|\alpha_{(m,c)}$ is

$$\alpha_{(m,tr)}|\alpha_{(m,c)}\sim$$
$$\mathcal{N}\left(\mathbb{E}[\alpha_{(m,tr)}|\alpha_{(m,c)}],(1-\rho^2)\sigma^2_{\alpha,(m,tr)}\right)$$

where $\rho=(row_*(W^{c}_{(m)})H^{-1}row_*(W^{tr}_{(m)})^T$

$\mathbb{E}[\alpha_{(m,tr)}|\alpha_{(m,c)}]=row_*(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)}+\rho\sqrt{\frac{\sigma_{\alpha,(m,tr)}}{\sigma_{\alpha,(m,c)}}}(\alpha_{(m,c)}-row_*(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)})$

Then the integral of interest can be re-written as

$$\sum_{m=1}^{M}\left[\int_{-\infty}^{\infty}\Phi\left(\frac{sig^{-1}(L_b+sig(\alpha_{(m,c)}))-\mathbb{E}[\alpha_{(m,tr)}|\alpha_{(m,c)}]}{\sqrt{(1-\rho^2)\sigma^2_{\alpha,(m,tr)}}}\right)\phi(\frac{\alpha_{(m,c)}-\psi_{\alpha,(m,c)}}{\sigma_{\alpha,(m,c)}})d\alpha_{(m,c)}\right]p(\mathcal{T}_m|\boldsymbol{y})$$

The *sig* function in the above integrals can be replaced by the normal CDF of a probit approximation if the computation is faster.

If an entirely deterministic algorithm is desired, deterministic numerical methods can probably be used to evaluate the univariate integrals in the above expression, however, this would have to be used in combination with a root finding algorithm, and in each iteration of the algorithm the integrals will have to be re-calculated. The integrals could probably be calculated using Monte Carlo methods, but again would have to be recalculated for each iteration of the root finding algorithm.

Therefore, the optimal approach may be to draw from the mixture of bivariate normal distributions, and obtain quantiles of calculated quantiles (the standard Monte Carlo approach, albeit with the dimension of the draws reduced to 2).

G.2 Estimation of Mean of Posterior Distribution of Conditional Average Treatment Effects

Now consider the Conditional Average Treatment Effect, i.e. $\frac{1}{N}\sum_{i=1}^{N}\left[\mathbb{E}[y_i|x_i,T_i=1]-\mathbb{E}[y_i|x_i,T_i=1]\right]$.

$$\sum_{m=1}^{M}\left[\int\frac{1}{N}\sum_{i=1}^{N}\left[\left(sig(row_i(W_{(m)}^{tr})\underline{\boldsymbol{\mu}}_{(m)})-sig(row_i(W_{(m)}^{c})\underline{\boldsymbol{\mu}}_{(m)})\right)\right]p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)},H_{(m)}^{-1})d\underline{\boldsymbol{\mu}}_{(m)}\right]p(\mathcal{T}_m|\boldsymbol{y})$$

this can also be separated into two integrals

$$\sum_{m=1}^{M}\left[\int\frac{1}{N}\sum_{i=1}^{N}sig(row_i(W_{(m)}^{tr})\underline{\boldsymbol{\mu}}_{(m)})p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)},H_{(m)}^{-1})d\underline{\boldsymbol{\mu}}_{(m)}-\int\frac{1}{N}\sum_{i=1}^{N}sig(row_i(W_{(m)}^{c})\underline{\boldsymbol{\mu}}_{(m)})p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)},H_{(m)}^{-1})d\underline{\boldsymbol{\mu}}_{(m)}\right]p(\mathcal{T}_m|\boldsymbol{y})$$

where *sig* denotes the sigmoid function (i.e. logistic). Note that $row_i(W_{(m)}^{tr})\underline{\boldsymbol{\mu}}_{(m)}$ can be estimated for $i=1,...,N$ in one matrix calculation $W_{(m)}^{tr}\underline{\boldsymbol{\mu}}_{(m)}$.

G.2.1 Monte Carlo Approximation of Expected CATE

[This is essentially the same as for ITEs]

Two possible approaches to Monte Carlo Approximation of the Expected CATE are:

1. It is possible to approximate each integral and then average by the model posterior probability. i.e. For each model, obtain a large number S of samples of $\underline{\boldsymbol{\mu}}_{(m),1},...,\underline{\boldsymbol{\mu}}_{(m),S}$ from the approximate distribution $\mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)},H_{(m)}^{-1})$ and estimate the difference in probabilities for model m .Then model averaged difference in probabilities (treated minus untreated) is:

$$\sum_{m=1}^{M}p(\mathcal{T}_m|\boldsymbol{y})\frac{1}{S}\sum_{s=1}^{S}\frac{1}{N}\sum_{i=1}^{N}\left[\frac{e^{row_i(W_{(m)}^{tr})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_i(W_{(m)}^{tr})\underline{\boldsymbol{\mu}}_{(m),s}}}-\frac{e^{row_i(W_{(m)}^{c})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_i(W_{(m)}^{c})\underline{\boldsymbol{\mu}}_{(m),s}}}\right]$$

2. Take a large number, S, samples from the mixture of multivariate normal distributions $\underline{\boldsymbol{\mu}}|\boldsymbol{y}\sim\sum_{m=1}^{M}\mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)},H_{(m)}^{-1})p(\mathcal{T}_m|\boldsymbol{y})$. Note that this involves sampling from each model's normal approximation with probability $p(\mathcal{T}_m|\boldsymbol{y})$, and for each model the sampled vector $\underline{\boldsymbol{\mu}}$ has a different interpretation and can have different dimensions because the sum-of-tree structures differ across models. Then the estimate is

$$\frac{1}{S}\sum_{s=1}^{S}\frac{1}{N}\sum_{i=1}^{N}\left[\frac{e^{row_i(W_{(m)}^{tr})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_i(W_{(m)}^{tr})\underline{\boldsymbol{\mu}}_{(m),s}}}-\frac{e^{row_i(W_{(m)}^{c})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_i(W_{(m)}^{c})\underline{\boldsymbol{\mu}}_{(m),s}}}\right]$$

G.2.2 Probit Approximation of Expected CATE

The sigmoid (logistic) function can be approximated by a normal CDF:

$$\begin{aligned}&\sum_{m=1}^{M}\frac{1}{N}\sum_{i=1}^{N}\left[\int sig(row_i(W_{(m)}^{tr})\underline{\boldsymbol{\mu}}_{(m)})p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)},H_{(m)}^{-1})d\underline{\boldsymbol{\mu}}_{(m)}-\right.\\&\qquad\left.\int sig(row_i(W_{(m)}^{c})\underline{\boldsymbol{\mu}}_{(m)})p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)},H_{(m)}^{-1})d\underline{\boldsymbol{\mu}}_{(m)}\right]p(\mathcal{T}_m|\boldsymbol{y})\\\approx&\sum_{m=1}^{M}\frac{1}{N}\sum_{i=1}^{N}\left[\int\Phi(row_i(W_{(m)}^{tr})\underline{\boldsymbol{\mu}}_{(m)})p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)},H_{(m)}^{-1})d\underline{\boldsymbol{\mu}}_{(m)}-\right.\\&\qquad\left.\int\Phi(row_i(W_{(m)}^{c})\underline{\boldsymbol{\mu}}_{(m)})p(\underline{\boldsymbol{\mu}}_{(m)}|\underline{\boldsymbol{\mu}}_{MAP,(m)},H_{(m)}^{-1})d\underline{\boldsymbol{\mu}}_{(m)}\right]p(\mathcal{T}_m|\boldsymbol{y})\end{aligned}$$

Let $\psi_{\alpha,i,(m,tr)} = row_i(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)}$ and $\sigma^2_{\alpha,i,(m,tr)} = row_i(W^{tr}_{(m)})H^{-1}_{(m)}row_i(W^{tr}_{(m)})^T$ and $\psi_{\alpha,i,(m,c)} = row_i(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{MAP,(m)}$ and $\sigma^2_{\alpha,i,(m,c)} = row_i(W^{tr}_{(m)})H^{-1}_{(m)}row_i(W^{c}_{(m)})^T$. Then $\alpha_{i,(m,tr)} = row_i(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m)} \sim \mathcal{N}(\psi_{\alpha,i,(m,tr)}, \sigma^2_{\alpha,i,(m,tr)})$ and $\alpha_{i,(m,c)} = row_i(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m)} \sim \mathcal{N}(\psi_{\alpha,i,(m,c)}, \sigma^2_{\alpha,i,(m,c)})$ Then the integrals can be rewritten as one dimensional integrals, and the expected ITE is:

$$\sum_{m=1}^{M}\frac{1}{N}\sum_{i=1}^{N}\left[\int \Phi(\alpha_{i,(m,tr)})p(\alpha_{i,(m,tr)}|\psi_{\alpha,i,(m,tr)},\sigma^2_{\alpha,i,(m,tr)})d\alpha_{i,(m,tr)} - \int \Phi(\alpha_{i,(m,c)})p(\alpha_{i,(m,c)}|\psi_{\alpha,i,(m,c)},\sigma^2_{\alpha,i,(m,c)})d\alpha_{i,(m,c)}\right]p(\mathcal{T}_m|\boldsymbol{y})$$

$$=\sum_{m=1}^{M}\frac{1}{N}\sum_{i=1}^{N}\left[\Phi\left(\frac{\psi_{\alpha,i,(m,tr)}}{\sqrt{1+\sigma^2_{\alpha,i,(m,tr)}}}\right)-\Phi\left(\frac{\psi_{\alpha,i,(m,c)}}{\sqrt{1+\sigma^2_{\alpha,i,(m,c)}}}\right)\right]p(\mathcal{T}_m|\boldsymbol{y})$$

This is equal to the arithmetic average of the ITE estimates.[62]

G.3 Credible Intervals for CATE Posterior Distribution

G.3.1 Monte Carlo Approximation of CATE Intervals

Take a large number, S, samples from the mixture of multivariate normal distributions $\underline{\boldsymbol{\mu}}|\boldsymbol{y} \sim \sum_{m=1}^{M}\mathcal{N}(\underline{\boldsymbol{\mu}}_{MAP,(m)}, H^{-1}_{(m)})p(\mathcal{T}_m|\boldsymbol{y})$. Note that this involves sampling from each model's normal approximation with probability $p(\mathcal{T}_m|\boldsymbol{y})$, and for each model the sampled vector $\underline{\boldsymbol{\mu}}$ has a different interpretation and can have different dimensions because the sum-of-tree structures differ across models.

For each sample, s calculate the average (over $i = 1, ..., N$) difference in probabilities under treatment and control group allocation (i.e. for $W^{tr}_{(m)}$ and $W^{c}_{(m)}$), and then find the relevant quantiles. i.e. calculate $\frac{1}{N}\sum_{i=1}^{N}\left(\frac{e^{row_i(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_i(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}} - \frac{e^{row_i(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_i(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}\right)$ for each draw and find the quantiles.

For example, for a 95% interval, find L_b such that

$$\frac{1}{S}\sum_{s=1}^{S}\mathbb{I}\left[\frac{1}{N}\sum_{i=1}^{N}\left(\frac{e^{row_i(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_i(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}} - \frac{e^{row_i(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_i(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}\right) < L_b\right] = 0.025$$

and find U_b such that

$$\frac{1}{S}\sum_{s=1}^{S}\mathbb{I}\left[\frac{1}{N}\sum_{i=1}^{N}\left(\frac{e^{row_i(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_i(W^{tr}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}} - \frac{e^{row_i(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}{1+e^{row_i(W^{c}_{(m)})\underline{\boldsymbol{\mu}}_{(m),s}}}\right) < U_b\right] = 0.975$$

G.3.2 Approximation of CATE Intervals, reducing the dimension of the integral

The dimension reduction can not be applied to the same extent as in the ITE case because $\frac{1}{N}\sum_{i=1}^{N} sig(\alpha_{i,(m,tr)}) - sig(\alpha_{i,(m,c)})$ depends on $2N$ parameters given by $\alpha_{i,(m,tr)}$ and $\alpha_{i,(m,c)})$ for $i = 1, ..., N$.

$$\sum_{m=1}^{M}\left[\int_{-\infty}^{\infty}\mathbb{I}\left(\frac{1}{N}\sum_{i=1}^{N}sig(\alpha_{i,(m,tr)}) - sig(\alpha_{i,(m,c)}) < L_b\right)p(\boldsymbol{\alpha}_{(m)}|\boldsymbol{\psi}_{(m)},\boldsymbol{\sigma}^{\boldsymbol{2}}_{(m)})d\boldsymbol{\alpha}_{(m)}\right]p(\mathcal{T}_m|\boldsymbol{y})$$

where $\boldsymbol{\alpha}_{(m)}$ is a $2N \times 1$ vector if the $\alpha_{i,(m,tr)}$ and $\alpha_{i,(m,c)})$ for $i = 1, ..., N$ and similarly $\boldsymbol{\psi}_{(m)}$ and $\boldsymbol{\sigma}^{\boldsymbol{2}}_{(m)}$ are vectors of the (approximate) means and variances of the elements of $\boldsymbol{\alpha}_{(m)}$. $\boldsymbol{\alpha}_{(m)}$ is multivariate normal, and

[62] For closer approximations to logistic probabilities, this can be replaced by

$$=\sum_{m=1}^{M}\frac{1}{N}\sum_{i=1}^{N}\left[\Phi\left(\frac{\psi_{\alpha,i,(m,tr)}}{\sqrt{\frac{8}{\pi}+\sigma^2_{\alpha,i,(m,tr)}}}\right)-\Phi\left(\frac{\psi_{\alpha,i,(m,c)}}{\sqrt{\frac{8}{\pi}+\sigma^2_{\alpha,i,(m,c)}}}\right)\right]p(\mathcal{T}_m|\boldsymbol{y})$$

it is possible to draw from each $\boldsymbol{\alpha}_{(m)}$ to evaluate all M integrals by Monte Carlo, or to draw from the model weighted mixture distribution of the $\boldsymbol{\alpha}_{(m)}$ (i.e. the mixture of multivariate normals). However, this may be generally of a higher dimension than $\underline{\boldsymbol{\mu}}_{(m)}$, depending on the data and selected models. Furthermore, extra calculations are required to obtain the means, variances, and covariances of the elements of $\boldsymbol{\alpha}_{(m)}$. Therefore this might not be computationally more efficient.

H Finding the MAP for Logit BCF

Let the vector of all terminal node parameters be denoted by $\underline{\boldsymbol{\theta}} = [\underline{\boldsymbol{\mu}}^T \underline{\boldsymbol{\tau}}^T]^T$. The Laplace approximation involves a second order Taylor expansion about the Maximum A Posteriori (MAP) estimate

$$\underline{\boldsymbol{\theta}}_{MAP} = \arg\min_{\underline{\boldsymbol{\theta}}} -(\log p(\boldsymbol{y}|W, \underline{\boldsymbol{\theta}}) + \log p(\underline{\boldsymbol{\theta}}))$$

$$= \arg\min_{\underline{\boldsymbol{\theta}}} - \left[\boldsymbol{y}^T W \underline{\boldsymbol{\theta}} - \sum_{i=1}^{N} \log(1 + e^{-W_i \underline{\boldsymbol{\theta}}}) - 0.5b \log(2\pi) + \frac{1}{2} b_\mu \log(a_\mu) + \frac{1}{2} b_\tau \log(a_\tau) - \frac{a_\mu}{2} \underline{\boldsymbol{\mu}}^T \underline{\boldsymbol{\mu}} - \frac{a_\tau}{2} \underline{\boldsymbol{\tau}}^T \underline{\boldsymbol{\tau}} \right]$$

(where b_μ and b_τ are the numbers of terminal nodes in the sums-of-trees represented by $\mu(x)$ and $\tau(x)$ respectively) gives the approximation of the posterior:

$$p(\underline{\boldsymbol{\theta}}|\boldsymbol{y}, W) \approx \mathcal{N}(\underline{\boldsymbol{\theta}}_{MAP}, H^{-1})$$

where H is the Hessian matrix of the negative log posterior (evaluated at the MAP).

$$H = W^T S W + A$$

where A is a diagonal matrix with the first b_μ diagonal elements equal to a_μ and the final b_τ elements equal to a_τ, and $S = diag(p_i(p_i))$ is an $n \times n$ diagonal matrix with diagonal elements determined by the probabilities p_i obtained from the logistic function.

The Hessian and the gradient of the negative posterior probability can be used to obtain an approximation of the MAP. The gradient is:

$$\boldsymbol{g} = W^T(\boldsymbol{p} - \boldsymbol{y}) + \begin{bmatrix} a_\mu \underline{\boldsymbol{\mu}} \\ a_\tau \underline{\boldsymbol{\tau}} \end{bmatrix}$$

where $\boldsymbol{p} = (p_1, ..., p_n)^T$, and $\underline{\boldsymbol{\mu}}$ and $\underline{\boldsymbol{\tau}}$ are the terminal nodes of the sums-of-trees $\mu(x)$ and $\tau(x)$ respectively.

I Tobit-BART-IS Implementation Details

I.1 Tobit Posterior and gradients with standard semi-conjugate priors

Chib (1992) used an uninformative prior for Bayesian Tobit. However, here we use the standard BART prior on the terminal node parameters and inverse gamma prior on the variance of the error term.

$$\tau^2 = \sigma^{-2} \sim \Gamma(\frac{\nu}{2}, \frac{\nu\lambda}{2})$$

$$\underline{\boldsymbol{\mu}} \sim N(0, \frac{\sigma^2}{a}) \ , \ or \ \underline{\boldsymbol{\mu}} \sim N(0, \frac{1}{a}\tau^{-2})$$

and the convenient Tobin reparameterization is $(\underline{\boldsymbol{\mu}}, \tau^2) \rightarrow (\boldsymbol{\alpha} = \underline{\boldsymbol{\mu}}\tau, \tau = (\tau^2)^{\frac{1}{2}})$. This gives

$$\alpha = \tau\beta \sim N(0, \frac{1}{a} I)$$

The standard Tobit model framework is

$$y_i^* = row_i(W)\underline{\boldsymbol{\mu}} + \varepsilon_i \ , \ \varepsilon \sim i.i.dN(0, \tau^{-2})$$

$$y_i = \max\{y_i^*, 0\} \ , \ i = 1, ..., n$$

The likelihood is:

$$\ell(\underline{\boldsymbol{\mu}}, \tau^2) = \left[\prod_{i \in C} 1 - \Phi(W_i \underline{\boldsymbol{\mu}} \tau)\right] (2\pi)^{-\frac{n_1}{2}} (\tau^2)^{\frac{n_1}{2}} e^{-\tau^2 \|y_1 - X_1 \underline{\boldsymbol{\mu}}\|^2/2} = \ell_0(\underline{\boldsymbol{\mu}}, \tau^2)\ell_1(\underline{\boldsymbol{\mu}}, \tau^2)$$

or, reparameterized, the likelihood is

$$\ell(\boldsymbol{\alpha}, \tau) = \left[\prod_{i \in C} 1 - \Phi(W_i \boldsymbol{\alpha})\right] (2\pi)^{-\frac{n_1}{2}} (\tau^2)^{\frac{n_1}{2}} e^{-\|\tau y_1 - W_1 \boldsymbol{\alpha}\|^2/2} = \ell_0(\boldsymbol{\alpha}, \tau)\ell_1(\boldsymbol{\alpha}, \tau)$$

where $c = \{j \ : \ y_j = 0, j = 1, ..., n\}$ (i.e. the set of observations for which the outcome is zero), n_1 is the number of observations for which the outcome is nonzero, y_1 is an $n_1 \times 1$ vector of nonzero outcomes, W_1 is an $n_1 \times b$ matrix of terminal node indicator variables corresponding to nonzero outcomes (y_1). $\|.\|$ is the Euclidean norm.

The log posterior is:

$$\tilde{L}(\underline{\boldsymbol{\mu}}, \tau^2) = \sum_{i \in C} \log[1 - \Phi(row_i(W\underline{\boldsymbol{\mu}}\tau))] - \frac{n_1}{2}\log(2\pi) + \frac{n_1}{2}\log(\tau^2) - \frac{\tau^2}{2}\|y_1 - W_1\underline{\boldsymbol{\mu}}\|^2$$

$$-\frac{b}{2}\log(2\pi) + \frac{b}{2}\log(a\tau) - \frac{a\tau}{2}\underline{\boldsymbol{\mu}}^T\underline{\boldsymbol{\mu}} + \frac{\nu}{2}[\log(2) - \log(\nu\lambda)] - \log\left(\Gamma\left(\frac{\nu}{2}\right)\right) + \left(\frac{\nu}{2} - 1\right)\log(\tau^2) - \frac{\tau^2 2}{\nu\lambda}$$

the reparameterized log posterior is

$$\tilde{L}(\boldsymbol{\alpha}, \tau) = \sum_{i \in C} \log[1 - \Phi(row_i(W\boldsymbol{\alpha}))] - \frac{n_1}{2}\log(2\pi) + \frac{n_1}{2}\log(\tau^2) - \frac{1}{2}(\tau y_1 - W_1\boldsymbol{\alpha})^T(\tau y_1 - W_1\boldsymbol{\alpha})$$

$$-\frac{b}{2}\log(2\pi) + \frac{b}{2}\log(a) - \frac{a}{2}\boldsymbol{\alpha}^T\boldsymbol{\alpha} + \frac{\nu}{2}[\log(2) - \log(\nu\lambda)] - \log\left(\Gamma\left(\frac{\nu}{2}\right)\right) + \left(\frac{\nu}{2} - 1\right)\log(\tau^2) - \frac{\tau^2 2}{\nu\lambda}$$

$$\tilde{L}_\alpha = -W_0^T A_0 + W_1^T(\tau Y_1 - W_1\boldsymbol{\alpha}) - a\boldsymbol{\alpha}$$

$$\tilde{L}_\tau = \frac{n_1}{\tau} - Y_1^T(\tau Y_1 - W_1\boldsymbol{\alpha}) + \frac{2(\frac{\nu}{2} - 1)}{\tau} - \frac{4\tau}{\nu\lambda}$$

And the Hessian matrix is:

$$\begin{bmatrix} -W_0^T B_0 W_0 - W_1^T W_1 - aI_b & W_1^T Y_1 \\ Y_1^T W_1 & -\frac{n_1}{\tau^2} - Y_1^T Y_1 - \frac{2(\frac{\nu}{2}-1)}{\tau^2} - \frac{4}{\nu\lambda} \end{bmatrix}$$

where $A_0 = vec(\lambda_i)$, $B_0 = diag(\lambda_i(\lambda_i - W_i\boldsymbol{\alpha}))$, $\lambda_i = \frac{\phi(W_i\boldsymbol{\alpha})}{1 - \Phi(W_i\boldsymbol{\alpha})}$

The negative of the gradient and the negative of the Hessian above can be used to obtain the MAP by Newton's algorithm (minimizing the negative of the log posterior). Algorithm 5 outline's Newton's method for minimizing the negative log-likelihood

Require parameter value, e.g. $a = 0.01$

Initialize $\begin{bmatrix}\boldsymbol{\alpha}\\ \tau\end{bmatrix} = \begin{bmatrix}\mathbf{0}_b\\ 1\end{bmatrix}$, where $\mathbf{0}_b$ is a zero vector of length b.

repeat

$\lambda_i = \frac{\phi(W_i\boldsymbol{\alpha})}{1-\Phi(W_i\boldsymbol{\alpha})}$ for $i \in C$

$A_0 = diag(\lambda_i)$

$B_0 = diag(\lambda_i(\lambda_i - W_i\boldsymbol{\alpha}))$

$$\boldsymbol{g} = -\begin{bmatrix} W_0^T A_0 + W_1^T(\tau Y_1 - W_1\boldsymbol{\alpha}) - a\boldsymbol{\alpha} \\ \frac{n_1}{\tau} - Y_1^T(\tau Y_1 - W_1\boldsymbol{\alpha}) + \frac{2(\frac{\nu}{2}-1)}{\tau} - \frac{4\tau}{\nu\lambda}\end{bmatrix}$$

$$H = -\begin{bmatrix} -W_0^T B_0 W_0 - W_1^T W_1 - aI_b & W_1^T Y_1 \\ Y_1^T W_1 & -\frac{n_1}{\tau^2} - Y_1^T Y_1 - \frac{2(\frac{\nu}{2}-1)}{\tau^2} - \frac{4}{\nu\lambda}\end{bmatrix}$$

$\underline{\boldsymbol{\mu}}_{new} = \underline{\boldsymbol{\mu}}_{old} - H^{-1}\boldsymbol{g}$

until *convergence*;

Algorithm 5: Newton's method for obtaining the mode (MAP) of the Tobit parameters

Alternatively, a quasi-Newton algorithm, such as the L-BFGS algorithm can be applied. The standard Laplace approximation for the marginal likelihood is:

$$p(\boldsymbol{y}|W_m, \mathcal{T}_{(m)}) = e^{\tilde{L}(\boldsymbol{\alpha}_{MAP}, \tau_{MAP})}(2\pi)^{b/2}|H_{MAP}|^{-1/2}$$

where H_{MAP} is the Hessian matrix of the negative log likelihood evaluated at the MAP parameter values. The log of the marginal likelihood approximation is:

$$log(p(\boldsymbol{y}|W_m, \mathcal{T}_{(m)})) = \tilde{L}(\boldsymbol{\alpha}_{MAP}, \tau_{MAP}) + \frac{b}{2}\log(2\pi) - (\frac{1}{2})log(|H_{MAP}|)$$

A more accurate approximation can be obtained using the double Laplace approximation methods of Tierney & Kadane (1986), as outlined by Chib (1992).

The Laplace approximation gives a multivariate normal approximation for the posterior distribution of the parameters:

$$\begin{bmatrix}\boldsymbol{\alpha}\\ \tau\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\boldsymbol{\alpha}_{MAP}\\ \tau_{MAP}\end{bmatrix}\ ,\ H_{MAP}^{-1}\right)$$

and the approximate marginal posterior distribution for $\boldsymbol{\alpha}$ is:

$$\boldsymbol{\alpha} \sim \mathcal{N}(\boldsymbol{\alpha}_{MAP}, H_{\boldsymbol{\alpha},MAP})$$

where $H_{\boldsymbol{\alpha},MAP} = W_0^T B_0 W_0 + W_1^T W_1 + aI_b$ is the submatrix of the Hessian of the negative log likelihood corresponding to $\boldsymbol{\alpha}$ evaluated at the MAP parameter values.

The posterior predictive mean probability that the outcome $y_\star$ is equal to one is:

$$p(y_\star = 1|row_\star(W), \mathcal{T}_{(m)}) = \int [1 - \Phi(row_\star(W)\boldsymbol{\alpha})]p(\boldsymbol{\alpha}|row_\star(W), \mathcal{T}_{(m)})d\boldsymbol{\alpha}$$

where $row_\star(W)$ is the row vector of terminal node indicator variables for the new observation. The integral can be re-written as a one-dimensional integral by considering $\psi = row_\star(W)\boldsymbol{\alpha}$, $\psi_{MAP} = row_\star(W)\boldsymbol{\alpha}_{MAP}$, and $\sigma_\psi^2 = row_\star(W)H_{\boldsymbol{\alpha},MAP}row_\star(W)^T$, which is approximately normally distributed $\psi \sim \mathcal{N}(\psi_{MAP}, \sigma_\psi^2)$.

$$p(y_\star = 1|row_\star(W), \mathcal{T}_{(m)}) = \int [1 - \Phi(\psi)]p(\psi|\psi_{MAP}, \sigma_\psi^2)d\psi$$

$$= 1 - \int \Phi(\psi)p(\psi|\psi_{MAP}, \sigma_\psi^2)d\psi = 1 - \Phi\left(\frac{\psi_{MAP}}{1+\sigma_\psi^2}\right)$$

and the average over models $\mathcal{T}_{(1)}, ..., \mathcal{T}_{(M)}$ is:

$$p(y_\star = 1) = 1 - \frac{1}{M}\sum_{m=1}^{M} \Phi\left(\frac{\psi_{MAP,(m)}}{1+\sigma^2_{\psi,(m)}}\right) p(\mathcal{T}_{(m)}|\boldsymbol{y}, \boldsymbol{X})$$

where $\psi_{MAP,(m)}$ and $\sigma_{\psi,(m)}$ are calculated using $\boldsymbol{\alpha}_{MAP,(m)}$ and $H_{\boldsymbol{\alpha},MAP,(m)}$, i.e. the MAP parameter values and Hessian evaluated at the MAP values for model (m).

Intervals for the predictive probability that $y_\star = 1$ can be obtained as follows. If the lower confidence probability is $lower\_prob = 0.025$ (i.e. for a 85% interval), then the lower bound for the predictive probability L_b satisfies:

$$\frac{1}{M}\sum_{i=1}^{M}\int_{-\infty}^{\infty} \mathbb{I}\{1-\Phi(\psi) < L_b\} p(\psi|\psi_{MAP}, \sigma^2_\psi)d\psi \; p(\mathcal{T}_{(m)}|\boldsymbol{y}, \boldsymbol{X}) = \frac{1}{M}\sum_{i=1}^{M}\int_{\Phi^{-1}(1-L_b)}^{\infty} p(\psi|\psi_{MAP}, \sigma^2_\psi)d\psi \; p(\mathcal{T}_{(m)}|\boldsymbol{y}, \boldsymbol{X})$$

$$= 1 - \frac{1}{M}\sum_{i=1}^{M}\int_{-\infty}^{\Phi^{-1}(1-L_b)} p(\psi|\psi_{MAP}, \sigma^2_\psi)d\psi \; p(\mathcal{T}_{(m)}|\boldsymbol{y}, \boldsymbol{X})$$

$$= 1 - \frac{1}{M}\sum_{i=1}^{M} \Phi\left(\frac{\Phi^{-1}(1-L_b) - \psi_{MAP}}{\sigma_\psi}\right) \; p(\mathcal{T}_{(m)}|\boldsymbol{y}, \boldsymbol{X}) = lower\_prob$$

or equivalently

$$\frac{1}{M}\sum_{i=1}^{M} \Phi\left(\frac{\Phi^{-1}(1-L_b) - \psi_{MAP}}{\sigma_\psi}\right) \; p(\mathcal{T}_{(m)}|\boldsymbol{y}, \boldsymbol{X}) = 1 - lower\_prob$$

and similarly the upper bound U_b is the number such that $1 - \frac{1}{M}\sum_{i=1}^{M} \Phi\left(\frac{\Phi^{-1}(1-U_b) - \psi_{MAP}}{\sigma_\psi}\right) \; p(\mathcal{T}_{(m)}|\boldsymbol{y}, \boldsymbol{X}) = 1 - upper\_prob$. Therefore L_b and U_b can be obtained by a root-finding algorithm (e.g. bisection).

Alternatively, Monte Carlo draws can be made from the mixture $\boldsymbol{\alpha} \sim \frac{1}{M}\sum_{i=1}^{M} \mathcal{N}(\boldsymbol{\alpha}_{MAP,(m)}, H_{\boldsymbol{\alpha},MAP,(m)})$, and for each draw the probability $[1 - \Phi(row_\star(W)\boldsymbol{\alpha})]$ can be calculated. Then the mean and quantiles across many Monte Carlo draws can be used for the predictive probability and interval for the predictive probability.