# Augmentation Samplers for Multinomial Probit Bayesian Additive Regression Trees

Yizhen Xu*
Department of Biostatistics, Johns Hopkins University
Joseph Hogan
Department of Biostatistics, Brown University
Michael Daniels
Department of Statistics, University of Florida
Rami Kantor
Division of Infectious Diseases, Brown University
Ann Mwangi
College of Health Sciences, School of Medicine, Moi University

December 9, 2022

**Abstract**

The multinomial probit (MNP)[7] framework is based on a multivariate Gaussian latent structure, allowing for natural extensions to multilevel modeling. Unlike multinomial logistic models, MNP does not assume independent alternatives. Kindo et al.[9] proposed multinomial probit BART (MPBART) to accommodate Bayesian additive regression trees (BART) formulation in MNP. The posterior sampling algorithms for MNP and MPBART are collapsed Gibbs samplers. Because the collapsing augmentation strategy yields a geometric rate of convergence no greater than that of a standard Gibbs sampling step, it is recommended whenever computationally feasible[7,12]. While this strategy necessitates simple sampling steps and a reasonably fast converging Markov chain, the complexity of stochastic search for posterior trees may undermine its benefit. We address this problem by sampling posterior trees conditional on the constrained parameter space and compare our proposals to that of Kindo et al.[9], who sample posterior trees based on an augmented parameter space. We also compare to the approach by Sparapani et al.[23] that specified the multinomial model in terms of conditional probabilities. In terms of MCMC convergence and posterior predictive accuracy, our proposals are comparable to the conditional probability approach and outperform the augmented tree sampling approach. We also show that the theoretical mixing rates of our proposals are guaranteed to be no greater than the augmented tree sampling approach.

*Keywords: keyword: Additive Regression Trees, Bayesian Data Augmentation, Categorical Outcomes, Latent Models*

*The authors gratefully acknowledge that *funding of this work is provided by the US National Institutes of Health (NIH) under R01 AI136664, R01 AI108441, R01 AI167694, and P30 AI 42853.*
Disclosure Statement: The authors report here are no competing interests to declare.

# 1 Introduction

Bayesian additive regression trees (BART)[4] is a flexible semiparametric Bayesian approach for regression on a recursively binary-partitioned predictor space; it uses sum-of-trees to model the mean function such that nonlinearities and interactions along with additive effects are naturally accounted for, and regularization priors are imposed to favor shallow trees to reduce over-fitting. There has been considerable literature on extending BART to various types of outcome variables[16,24,28]. We consider the extension of BART to multinomial probit models[7] (MNP). Existing BART-related work has developed efficient Markov chain Monte Carlo (MCMC) algorithms for Gaussian likelihoods, which naturally adapt to frameworks with Gaussian-distributed latent variables. However, careful consideration of data augmentation (DA) schemes is needed to ensure computational efficiency of implementing BART under the multinomial probit framework. The main contributions of this paper are to provide a detailed review of sampling algorithms for parameter expansion that are based on DA schemes and to introduce a set of new MCMC algorithms for multinomial probit BART (MPBART).

Our work is motivated by the need for accurate predictive modeling of patient engagement in HIV care[6,29], while accounting for death and transfer out of care as competing endpoints[10]. These models are used to characterize patient transition through the HIV cascade, which describes essential stages of the HIV care continuum: (a) HIV diagnosis through testing, (b) linkage to care, (c) engagement in care, (d) initiation of antiviral therapy (ART) through retention, and (e) sustained suppression of viral load. The care cascade framework has been widely used as a monitoring and evaluation tool for improving and managing HIV health care systems. We will demonstrate and compare different algorithms for using multinomial BART models to characterize engagement and retention in HIV care in Section 4.

MNP[7] and multinomial logistic[19] (MNL) regression models are widely used tools for predicting and describing the relationships of explanatory variables to multinomial outcomes. Kindo et al.[9] proposed the MPBART framework that fits BART to the multivariate Gaussian latent

variables in the MNP. Related work incorporating BART into categorical response models is introduced by Murray [21], where BART is extended to log-linear models that include multinomial logistic BART (MLBART). Both MNP and MNL regression can be derived from a latent variable framework, where each outcome category is a manifestation of a latent utility that depends on covariates. The observed categorical outcome is the utility-maximizing category. MNP and MNL regression assume the latent utility distribution to be multivariate Gaussian and independent extreme-value distribution, respectively. The MNP formulation is appealing because it incorporates between-category dependence, a feature that extends naturally to MPBART. We will show that allowing non-zero correlations between latent variables can have a substantial impact on predictive accuracy.

There are two difficulties in sampling from posterior distributions of MNP. First, a closed-form expression for the multinomial outcome's marginal distribution is not available; second, identifiability of the MNP model requires constraints on the covariance matrix of the latent variables, hindering specification of conjugate distributions and making posterior sampling challenging. There has been considerable work on Bayesian sampling techniques to address these computational issues based on DA-related methods [1,7,17,18,22]. The original DA algorithm [25] is a stochastic generalization of the EM algorithm [5]. Marginal data augmentation (MDA) [15,20,26] generalizes and accelerates the DA algorithm via parameter expansion such that full conditionals are easier to sample from and expansion parameter(s) are subsequently marginalized over. Heuristically, the MDA Gibbs sampler can traverse the parameter space more efficiently with the extra variation induced by the expansion parameter(s), resulting in possible computational gains, including a faster mixing rate [15,20]. Li et al. [11] provided an example for posterior sampling of a correlation matrix via parameter expansion. By contrast, sampling from the constrained model parameter space is difficult because the full conditionals do not have a simple closed form; the MDA scheme circumvents the difficulty and allows an easier and more efficient joint sampling of expansion parameter and transformed model parameters. Imai and van Dyk [7] unified several previous proposals under

the umbrella of MDA, examined different prior specifications of the model parameters, and outlined two adaptations of the MDA scheme for posterior sampling of the MNP based on parameter expansion.

Building upon the work of Imai and van Dyk[7], Kindo et al.[9] proposed an algorithm, which we refer to as KD, for fitting the MPBART. Our own implementation of KD yielded oversized posterior trees from overfitting and difficulty in posterior convergence.We therefore propose two alternative procedures for fitting the MPBART that have simpler algorithmic structure, improved convergence in the sum-of-trees and the covariance matrix, and a mixing rate at least as good as the original procedure when the Markov chain reaches equilibrium. Our algorithms show better out-of-sample accuracy and stability in predictive tasks under various settings when evaluated in terms of posterior predictive distribution and posterior mode. The posterior mode accuracy is commonly used as an evaluation metric in supervised learning literature[9]. Our proposals are based on the idea of fitting the sum-of-trees in a normalized parameter space to reduce disruptions to the stochastic search of posterior trees, resulting in a less difficult convergence of the Markov chain.

In every step of the Gibbs sampler, the MDA scheme requires (1) the joint sampling of expansion parameter(s) and transformed model parameters, and (2) the marginalization over the expansion parameter. However, the two actions are not always feasible for complicated Gibbs sampling problems. For example, sampling the functional mean component jointly with an expansion parameter in an MPBART algorithm is difficult because posterior trees are sampled by stochastic search. Algorithms for MNP and MPBART generally fall under the category of partially marginalized augmentation (PMA) samplers[27], which relaxes the fully marginalized structure of the MDA and can lead to improvements in convergence rate when more steps involve joint sampling and marginalization of the expansion parameter(s)' components.

We will show that KD and one of our proposals are, respectively, the MPBART-generalization of the Schemes 1 and 2 for estimating MNP proposed in Imai and van Dyk[7]. The primary distinc-

tion between the two MNP schemes is that the former uses augmentation in the sampling of model coefficients for the mean of latent variables, while the latter does not. Imai and van Dyk [7] recommended Scheme 1 over Scheme 2 because its geometric rate of convergence is at least as good as Scheme 2. One of our key contributions is to demonstrate that the same recommendation does not apply to MPBART. Contrary to the intuition regarding PMA samplers that more augmented posterior sampling steps are associated with improved posterior convergence, we illustrate that when sophisticated Metropolis-Hastings or stochastic search is involved in complex samplers, certain steps may be sensitive to or undermined by the incorporation of expansion parameters. This motivates the need for new algorithm design considerations.

This paper is structured as follows. Section 2.1 describes the formulation of MNP and MPBART frameworks; Section 2.2 reviews sampling schemes for the MNP, including DA and MDA; Section 2.4 describes the existing algorithms and introduces our new proposals for fitting the MPBART; and Section 2.5 provides a theoretical evaluation of different MPBART algorithms in terms of the mixing rate under stationarity. Section 3 compares multiple BART-related multinomial outcome models, including our proposals, on simulated datasets under different settings, and Section 4 demonstrates the comparison on a real-world dataset from a large HIV care program in Kenya. Section 5 summarizes the conclusions.

# 2 Method

## 2.1 General Background

For the categorical outcome $S$, which takes value in $\{0, \ldots, C\}$, the general latent variable framework for multinomial models assumes that $S$ is a manifestation of unobserved latent utilities $Z = (Z_0, \ldots, Z_C)^T \in \mathbb{R}^{C+1}$ , where $S = S(Z) = \text{argmax}_l Z_l$, i.e. $S = k$ if $Z_k \geq Z_l$ for all $l \neq k$. In general, $C$ is the number of outcome categories minus one. The framework requires normalization for identifiability because $S$ is invariant to a translation or a scaling (by a positive constant) of $Z$. Without loss of generality, we assume that the reference outcome cate-

gory is 0; the normalization is achieved by first characterizing $S$ as a function of latent variables $W = (W_1, \ldots, W_C)^T \in \mathbb{R}^C$, such that $W_l = Z_l - Z_0$ and

$$S(W) = \begin{cases} l & \text{if } \max(W) = W_l \geq 0 \\ 0 & \text{if } \max(W) < 0. \end{cases} \tag{1}$$

The MNP models $W$ in terms of covariates $X$ and accounts for correlation across outcome levels by assuming $W$ follows a multivariate normal model

$$W(X) \sim \text{MVN}(G(X;\theta), \Sigma), \tag{2}$$

where $G(X;\theta) = (G_1(X;\theta_1), \ldots, G_C(X;\theta_C))^T, \theta = (\theta_1, \ldots, \theta_C)^T$ and $\Sigma = \{\sigma_{ij}\}$ is a $C \times C$ positive definite symmetric matrix.

Identifiability of the model requires normalizing the scale of $W$ because by definition the outcome $S$ is invariant to a multiplication of $W$ by any positive constant. From (2), the normalization for scale occurs by imposing a constraint on the covariance matrix $\Sigma$, such as $\text{trace}(\Sigma) = C^2$. To illustrate, suppose there are latent variables $\widetilde{W}$ such that

$$\widetilde{W}(X) \sim \text{MVN}(G(X;\widetilde{\theta}), \widetilde{\Sigma}), \tag{3}$$

where $\widetilde{W}(X) = \alpha W(X)$, $G(X;\widetilde{\theta}) = \alpha G(X;\theta)$, $\widetilde{\Sigma} = \alpha^2\Sigma$, and $\alpha > 0$. By (1), $\widetilde{W}$ and $W$ yield the same $S$. However, if $\Sigma$ satisfies the trace constraint, $W$ is the normalized counterpart of $\widetilde{W}$ and $\alpha^2 = \text{trace}(\widetilde{\Sigma})/C$ is a positive scalar that ensures a one-to-one mapping from $W$ to $\widetilde{W}$.

Direct posterior sampling of parameters in (2) is difficult due to the constraint on $\Sigma$. A technique for easier sampling is to augment the parameter space such that it is possible to specify a conjugate prior so that target parameters can be obtained by converting samples back to the normalized scale. The obvious choice of augmented parameter space is the one without the normalization for scale, i.e. $(\widetilde{W}, \widetilde{\theta}, \widetilde{\Sigma})$ in (3). Imai and van Dyk [7] suggested a constrained inverse Wishart

prior for $\Sigma$ such that its joint distribution with $\alpha^2$ is equivalent to the unconstrained covariance matrix having prior distribution $\widetilde{\Sigma} \sim$ inv-Wishart$(\nu, \Psi)$. This makes it possible to sample easily from the conditional posterior of $\widetilde{\Sigma}$. Setting $\nu = C+1$ and $\Psi$ to be an identity matrix is equivalent to sampling the corresponding correlations of $\widetilde{\Sigma}$ from a uniform distribution. When $\nu > C + 1$, the expectation of $\widetilde{\Sigma}$ has a closed form $E(\widetilde{\Sigma}) = \Psi/(\nu - C - 1)$.

The standard framework for MNP regression assumes a linear model specification for each $W_l(X)$, i.e. $G_l(X; \theta_l) = X\theta_l$ for $l = 1, \ldots, C$. Kindo et al.[9] proposed MPBART to increase the predictive power and the flexibility in dealing with complicated nonlinear and interaction effects. The innovative idea is to approximate each mean component of $W(X)$ using a sum of $m$ trees, $G_l(X; \theta_l) = \sum_{k=1}^{m} g(X; \theta_{lk})$, where $l = 1, \ldots, C$ and $\theta_{lk}$ is the set of parameters corresponding to the $k$th binary tree for the $l$th latent variable, $W_l(X)$. MPBART uses the same Bayesian regularization prior on the trees to restrict over-fitting as in Chipman et al.[4]. An important contribution of Kindo et al.[9] is deriving from (2) the conditional distribution for Gibbs sampling of each individual tree, and embedding it into the backfitting procedure of BART. See Chipman et al.[3] and Chipman et al.[4] for details on the BART backfitting procedure.

## 2.2 Review of Data Augmentation

The goal of data augmentation (DA) schemes is to draw samples from $(y, \phi)$, where $y$ and $\phi$ represent the augmented data and model parameters, respectively. The sampling algorithm Kindo et al.[9] have for MPBART heavily relies on Imai & van Dyk's[7] work on fitting the MNP, which explores different Gibbs samplers of $(W, \theta, \Sigma)$ under the umbrella of marginal data augmentation (MDA)[15,20], an extension and improvement of the DA algorithm[25]. This section provides a brief overview of relevant developments on the DA algorithm for fitting the MPBART in Section 2.4.

*Basic data augmentation.* To begin with, we illustrate the simple task of sampling $(y, \phi)$ under the DA algorithm of Tanner and Wong[25]:

**Scheme [DA]**

1. Draw $y \sim f(y|\phi)$.

2. Draw $\phi \sim f(\phi|y)$.

*Marginalized data augmentation (MDA).* The basic idea of MDA versus DA is to expand the model and overparameterize $f(y, \phi)$ to $f(y, \phi, \alpha)$; the expansion parameter $\alpha$ often corresponds to a transformation of $y$ and/or $\phi$. For example, $\alpha$ may index a transformation of $y$ to $\widetilde{y} = t_\alpha(y)$ where $t_\alpha$ is one-to-one and differentiable, thereby expanding the model from $f(y, \phi)$ to $f(\widetilde{y}, \phi, \alpha)$. The choice to sample from $f(y, \phi, \alpha)$ or $f(\widetilde{y}, \phi, \alpha)$ depends on the specific model, and they are usually interchangeable. This approach is appealing when sampling from $f(y, \alpha|\phi)$ or $f(\tilde{y}, \alpha|\phi)$ is easier than the sampling of $y$ alone. Liu and Wu [15] and Meng and Van Dyk [20] simultaneously developed MDA. Liu and Wu [15] provided theoretical results on the convergence rate of the MDA. Meng and Van Dyk [20] introduced the MDA under two augmentation schemes, *grouping* and *collapsing* [12,14]; both procedures lead to the same distribution of $(y, \phi)$ as Scheme [DA].

*MDA with grouping.* The grouping scheme samples conditionally on the expansion parameter $\alpha$, while the collapsing scheme integrates $\alpha$ out from the joint distribution. MDA under the grouping scheme is preferred when the sampling of $y$ or $\phi$ jointly with $\alpha$ is easier than that in Scheme [DA]. For example, when $f(\phi|y, \alpha)$ is easier to sample than $f(\phi|y)$, and $f(y, \alpha|\phi)$ is easy to sample, the sampler can "group" $y$ and $\alpha$ together and treats them as a single component,

**Scheme [MDA-G]**

1. Draw $(y, \alpha) \sim f(y, \alpha|\phi)$.

2. Draw $\phi \sim f(\phi|y, \alpha)$.

*MDA with collapsing.* MDA under the collapsing scheme "collapses down" $\alpha$ by integrating it out from the joint distributions, i.e. $y \sim f(y|\phi) = \int f(y|\phi, \alpha) f(\alpha|\phi) d\alpha$ and $\phi \sim f(\phi|y) = \int f(\phi|y, \alpha) f(\alpha|y) d\alpha$. The implementation is as follows:

**Scheme [MDA-C]**

1. Draw $(y, \alpha) \sim f(y, \alpha|\phi)$ by $\alpha \sim f(\alpha|\phi)$ and $y \sim f(y|\phi, \alpha)$.

2. Draw $(\phi, \alpha) \sim f(\phi, \alpha|y)$ by $\alpha \sim f(\alpha|y)$ and $\phi \sim f(\phi|y, \alpha)$.

Notice that the newly sampled $\alpha$ is discarded in each step of the Scheme [MDA-C]. In practice, it may be reasonable to assume a priori independence between $\phi$ and $\alpha$ because $\phi$ are parameters identified from the observed data, which does not contain information on $\alpha$. Furthermore, given that transforming the augmented data $y$ is of interest, it may be true that the conditional sampling of model parameters $\phi$ is more plausible under $\widetilde{y}$ than $y$. Accordingly, Scheme [MDA-C] can be rewritten as:

**Scheme [MDA-C']**

1. Draw $(\widetilde{y}, \alpha)$ by drawing $\alpha \sim f(\alpha)$ and then $y \sim f(y|\phi, \alpha)$, and compute $\widetilde{y} = t_\alpha(y)$.

2. Draw $(\phi, \alpha)$ by drawing $\alpha \sim f(\alpha|\widetilde{y})$ and then $\phi \sim f(\phi|\widetilde{y}, \alpha)$.

The $f(\alpha)$ and $f(\alpha|\widetilde{y})$ are the prior and posterior (under the transformed augmented data) of $\alpha$, respectively. The optimality of MDA under the collapsing scheme (Scheme [MDA-C]) over the DA algorithm (Scheme [DA]) in terms of convergence rate is proven in Meng and Van Dyk [20] and Liu and Wu [15]. Liu and Wu [15] also introduced Scheme [MDA-LW], which is equivalent to Scheme [MDA-C'] in terms of the sampling distribution and rate of convergence. This scheme is implicitly applied in the algorithms for fitting the MNP and MPBART, typically in the normalization of model parameters after each round of Gibbs sampling. Structurally, Scheme [MDA-LW] is in the form of Scheme [DA] with an additional intermediate step, which makes more clear the connection between the MDA and the DA algorithm:

**Scheme [MDA-LW]**

1. Draw $y \sim f(y|\phi)$.

2. Draw $\alpha_1 \sim f(\alpha)$, compute $\widetilde{y} = t_{\alpha_1}(y)$; draw $\alpha_2 \sim f(\alpha|\widetilde{y})$, compute $y' = t^{-1}_{\alpha_2}(\widetilde{y})$.

3. Draw $\phi \sim f(\phi|y')$.

Note that $y$ and $y'$ follow the same distribution. The intuition behind the improvement of Scheme [MDA-LW] compared to the DA algorithm is that the intermediate step of sampling from $y'$ allows the sampler for $\phi$ to explore the expanded model space with more freedom.

## 2.3 Data Augmentation for the MNP

For fitting the MNP, Imai and van Dyk[7] introduced two algorithms for the Gibbs sampling of $(W, \theta, \Sigma)$, which we refer as IvD1 and IvD2. The IvD1 modifies Scheme [MDA-C'] by expanding the model to $(\widetilde{W}, \widetilde{\theta}, \widetilde{\Sigma}, \alpha)$ such that $\widetilde{W}$ and $(\widetilde{\theta}, \widetilde{\Sigma})$ correspond to $\widetilde{y}$ and $\phi$, respectively, and $\alpha = (\alpha_1, \alpha_2, \alpha_3)$:

**Scheme [IvD1]**

1. Draw $(\widetilde{W}, \alpha_1)$ by drawing $\alpha_1 \sim f(\alpha|\Sigma)$ and then $W \sim f(W|\theta, \Sigma)$, and compute $\widetilde{W} = \alpha_1 W$.

2. Draw $(\widetilde{\theta}, \alpha_2)$ by drawing $\alpha_2 \sim f(\alpha|\widetilde{W}, \Sigma)$ and then $\widetilde{\theta} \sim f(\widetilde{\theta}|\alpha_2, \widetilde{W}, \Sigma)$, and compute $\theta = \widetilde{\theta}/\alpha_2$.

3. Draw $(\widetilde{\Sigma}, \alpha_3)$ by $\widetilde{\Sigma} \sim f(\widetilde{\Sigma}|\widetilde{W} - X\widetilde{\theta})$ and compute $\alpha_3 = \sqrt{\text{trace}(\widetilde{\Sigma})/C}$.

Using $\widetilde{\Sigma}$ and $\alpha_3$ from Step 3, we can compute the normalized covariance matrix $\Sigma = \widetilde{\Sigma}/\alpha_3^2$ and use it in Steps 1 and 2 of the next round of posterior sampling; this is analogous to having $\alpha_3$ index a one-to-one mapping from the expanded model space $(\widetilde{\Sigma})$ to the normalized space $(\Sigma)$. Steps 1 and 3 in Scheme [IvD1] collapse down $\alpha_1$ and $\alpha_3$, but Scheme [IvD1] is not a direct implementation of the MDA as in Scheme [MDA-C'] because Step 1 is conditional on $\theta$, or equivalently $(\widetilde{\theta}, \alpha_2)$ where $\theta = \widetilde{\theta}/\alpha_2$. Hence, Step 2 does not integrate out (collapse down) $\alpha_2$.

Standard MDA (Schemes [MDA-C] and [MDA-C']) are collapsed Gibbs samplers that integrate out expansion parameter(s) by redrawing and discarding $\alpha$ in every step. Scheme [IvD1] is a partially marginalized augmentation (PMA)[27] procedure that relaxes the restrictive structure of full marginalization in MDA. PMA allows the conditional distribution in a $k$th step of the Gibbs sampler to depend on expansion parameter(s) drawn in other steps. Algorithms for fitting the MPBART in Section 2.4 are also PMA procedures.

IvD1 can also be viewed from a different perspective. Due to the linearity in model specification of the MNP, i.e. $G_l(X; \theta_l) = X\theta_l$ for $l = 1, \ldots, C$, the linear relationship between $\theta$ and $\widetilde{\theta}$

holds in Step 2 of Scheme [IvD1], and it is equivalent to direct sampling of $\theta$ from $f(\theta|\widetilde{W}/\alpha_2, \Sigma)$. Hence, IvD1 can be rearranged as follows:

**Scheme [IvD1']**

1. Draw $W \sim f(W|\theta, \Sigma)$.
2. Draw $\alpha_1 \sim f(\alpha|\Sigma)$, compute $\widetilde{W} = \alpha_1 W$; draw $\alpha_2 \sim f(\alpha|\widetilde{W}, \Sigma)$, compute $W' = \widetilde{W}/\alpha_2$.
3. Draw $\theta \sim f(\theta|W', \Sigma)$.
4. Draw $\Sigma$ by $\widetilde{\Sigma} \sim f(\widetilde{\Sigma}|\widetilde{W} - X\widetilde{\theta})$, compute $\alpha_3 = \sqrt{\text{trace}(\widetilde{\Sigma})/C}$, and $\Sigma = \widetilde{\Sigma}/\alpha_3^2$, where $\widetilde{\theta} = \alpha_2\theta$.

The first three steps are equivalent to sampling $f(W, \theta|\Sigma)$ in Scheme [MDA-LW]. Step 4 collapses down $\alpha_3$, but the fact that Step 4 is conditional on $(\alpha_1, \alpha_2)$ through $(\widetilde{W}, \widetilde{\theta})$ makes IvD1 not a collapsed Gibbs sampler collectively. IvD2 is given as follows:

**Scheme [IvD2]**

1. Draw $(\widetilde{\epsilon}, \alpha_1)$ by $\alpha_1 \sim f(\alpha|\Sigma)$ and $W \sim f(W|\theta, \Sigma)$, compute $\widetilde{\epsilon} = \alpha_1[W - G(X; \theta)]$.
2. Draw $(\Sigma, \alpha_3)$ by $\widetilde{\Sigma} \sim f(\widetilde{\Sigma}|\widetilde{\epsilon})$, compute $\alpha_3 = \sqrt{\text{trace}(\widetilde{\Sigma})/C}$, and $\Sigma = \widetilde{\Sigma}/\alpha_3^2$.
3. Draw $\theta \sim f(\theta|W, \Sigma)$.

IvD2 separates the sampling into two parts, $(\widetilde{\epsilon}, \Sigma)$ and $\theta$; the first part utilizes the MDA under Scheme [MDA-C] and the second part is a standard Gibbs sampling draw. Theoretically, as stated in Imai and van Dyk[7], IvD1 and IvD2 have the same lag-one autocorrelation when the MCMC chain is stationary. However, they showed through numerical experiments that IvD1 is better than IvD2 in estimating the MNP in terms of being less sensitive to the starting values of $(\theta, \Sigma)$.

In the next section, we describe Kindo et al.'s[9] algorithm (KD) and our two new proposals and connect them to the schemes reviewed here.

## 2.4 Algorithms for Posterior Sampling Algorithms of MPBART

For ease of notation, let $W_{i,-j} = (W_{i1}, \ldots, W_{i,j-1}, W_{i,j+1}, \ldots, W_{iC})$ and let $\mu = G(X;\theta) \in \mathbb{R}^C$ be the sum-of-trees component under the normalization of scale. Kindo et al.'s algorithm for fitting the MPBART can be summarized as the following augmented Gibbs sampler:

**Algorithm [KD]**

1. Sample $(\widetilde{W}, \alpha_1^2)|(\mu, \Sigma, S)$.
   (a) Draw $\alpha_1^2$ from its conditional prior $f(\alpha^2|\Sigma) = \text{trace}[\Psi\Sigma^{-1}]/\chi^2_{\nu C}$;
   (b) for each $j$, update $W_{ij}$ conditional on $W_{i,-j}$, $\mu$, $\Sigma$, and the observed outcome $S_i$, from a truncated normal distribution; and
   (c) transform $W_i$ and $\Sigma$ to $\widetilde{W}_i = \alpha_1 W_i$ and $\widetilde{\Sigma}^* = \alpha_1^2\Sigma$.
2. Sample $\widetilde{\theta}|(\widetilde{W}, \alpha_1^2, \Sigma)$.
   (a) Draw $\widetilde{\theta} \sim f(\widetilde{\theta}|\widetilde{W}, \widetilde{\Sigma}^*)$; and
   (b) set $\widetilde{\mu} = G(X;\widetilde{\theta})$ and $\mu = \widetilde{\mu}/\alpha_1$.
3. Sample $(\Sigma, \alpha_3^2)|(\widetilde{W}, \widetilde{\theta})$.
   (a) Draw $\widetilde{\Sigma} \sim$ Inv-Wishart$(N + \nu, \Psi + \sum_{i=1}^N \widetilde{\epsilon}_i\widetilde{\epsilon}_i^T)$, where $\widetilde{\epsilon}_i = \widetilde{W}_i - \widetilde{\mu}_i$;
   (b) set $\alpha_3^2 = \text{trace}(\widetilde{\Sigma})/C$; and
   (c) set $\Sigma = \widetilde{\Sigma}/\alpha_3^2$ and $W = \mu + \widetilde{\epsilon}/\alpha_3$.

Step 1 jointly samples from $f(\widetilde{W}, \alpha_1^2|\mu, \Sigma, S)$ by first drawing the expansion parameter $\alpha_1^2$ from its prior distribution $f(\alpha^2|\Sigma)$, and then computing $\widetilde{W} = \alpha^2 W$ where $W$ is sampled from $f(W|\mu, \Sigma, S)$. Step 1(a) samples $\alpha_1^2$ such that $\alpha_1^2/\text{trace}[\Psi\Sigma^{-1}]$ follows an inverse-chi-squared distribution with $\nu C$ degrees of freedom. Step 1(b) samples each $W_{ij}$ from a truncated normal distribution described in Appendix D.1 based on (1), as the observed outcome $S_i$ imposes an interval constraint on $W_i$, e.g. if $S_i$ equals the reference level 0, then $W_{ij}$'s are right truncated at

0. Step 2 samples posterior trees across multivariate mean components by Gibbs sampling and each posterior tree is sampled as in regular BART. Step 3 computes $\alpha_3$ using the sampled $\widetilde{\Sigma}$ and then normalizes the scale of the model by Step 3(c).

Notice that the sampling of model parameters $\widetilde{\theta}$ is conditional on $(\widetilde{W}, \widetilde{\Sigma}^*)$, which is equivalent to conditioning on $(\widetilde{W}, \alpha_1^2, \Sigma)$ or $(W, \alpha_1^2, \Sigma)$; this observation is essential to the analysis of Algorithm [KD] in Section 2.5. Algorithm [KD] is closely related to IvD1 (Scheme [IvD1]) but different in that it does not update the expansion parameter $\alpha_2$ as in Step (b) of IvD1. This is analogous to having $\alpha_2$ in IvD1 set to the sampled value of $\alpha_1$ from Step (a). The reason for this modification is that the posterior tree parameters in BART, denoted by $\theta$, are drawn via stochastic search; it would be extremely challenging to derive an analytical expression for $f(\alpha|\widetilde{W}, \Sigma)$ from $\int f(\alpha, \theta|\widetilde{W}, \Sigma)d\theta$ as in MNP because the specification is no longer linear in $\theta$.

In the first step, $\widetilde{W}$ is a scaled version of $W$ through $\widetilde{W} = \alpha_1 W$. From (3), fitting the sum-of-trees component to $\widetilde{W}$ is analogous to sampling the parameters in an un-normalized space. Posterior tree sampling in BART makes a one-step update on each tree from its previous state, by one of the following four types of proposals: GROW, PRUNE, CHANGE, and SWAP. Stochastic search in a massive space of possible tree structures can be challenging when $\widetilde{W}$, the quantity to which the sum-of-trees is fitting, is unstable. Heuristically, we would expect fitting the sum-of-trees component to $W$, which is a normalized quantity, instead of $\widetilde{W}$ to be more stable, induce better posterior convergence, and improve the prediction accuracy. Given these considerations, we modify Algorithm [KD] and propose the following:

**Algorithm [P1]**

1. Sample $(W, \alpha_1^2)|(\mu, \Sigma, S)$.

   (a) Draw $\alpha_1^2$ from its conditional prior $f(\alpha^2|\Sigma) = \text{trace}[\Psi\Sigma^{-1}]/\chi^2_{\nu C}$;

   (b) for each $j$, update $W_{ij}$ conditional on $W_{i,-j}$, $\mu$, $\Sigma$, and $S_i$, from a truncated normal distribution; and

   (c) transform $W_i$ to $\widetilde{W}_i = \alpha_1 W_i$.

2. Sample $\theta|(W, \Sigma)$. Draw $\theta \sim f(\theta|W, \Sigma)$ and then set $\mu = G(X; \theta)$.

3. Sample $(\Sigma, \alpha_3^2)|(W, \alpha_1, \theta)$.

   (a) Draw $\widetilde{\Sigma} \sim$ Inv-Wishart$(N + \nu, \Psi + \sum_{i=1}^{N} \widetilde{\epsilon}_i \widetilde{\epsilon}_i^T)$, where $\widetilde{\epsilon}_i = \widetilde{W}_i - \alpha_1 \mu_i$;

   (b) set $\alpha_3^2$ to trace$(\widetilde{\Sigma})/C$; and

   (c) set $\Sigma = \widetilde{\Sigma}/\alpha_3^2$ and $W = \mu + \widetilde{\epsilon}/\alpha_3$.

In the first proposal (Algorithm [P1]), the expansion parameters $(\alpha_1, \alpha_3)$ do not affect the sampling of the trees in Step 2. If the order of Step 2 and 3 are swapped, it becomes Scheme [IvD2] in Section 2.2. Algorithms [KD] and [P1] are the MPBART analogues of IvD1 and IvD2 for the MNP. Imai and van Dyk[7] expected IvD1 to outperform IvD2 for the MNP and demonstrated through simulations. While for MPBART, we find Algorithm [P1] to be equal or superior to Algorithm [KD] theoretically (Section 2.5) and computationally (Sections 3 and 4).

As an alternative to Algorithm [P1], we introduce another proposal, Algorithm [P2], which "abandons" the MDA framework. The only augmentation involved in Algorithm [P2] is Step 3, which adopts a Scheme [MDA-LW]-like strategy in the constrained parameter space. If we fix $\alpha_1$ to be 1, both Algorithms [KD] and [P1] simplify to Algorithm [P2]. We show in Appendix B that Algorithms [P1] and [P2] draw $\Sigma$ from approximately the same sampling distribution under certain conditions.

**Algorithm [P2]**

1. Sample $W|(\mu, \Sigma, S)$. For each $j$, update $W_{ij}$ conditional on $W_{i,-j}$, $\mu$, $\Sigma$, and $S_i$ from a truncated normal distribution.

2. Sample $\theta|(W, \Sigma)$. Draw $\theta \sim f(\theta|W, \Sigma)$ and then set $\mu = G(X; \theta)$.

3. Sample $(\Sigma, \alpha_3^2)|(W, \theta)$.

   (a) Draw $\widetilde{\Sigma} \sim$ Inv-Wishart$(N + \nu, \Psi + \sum_{i=1}^{N} \epsilon_i \epsilon_i^T)$, where $\epsilon_i = W_i - \mu_i$;

   (b) set $\alpha_3^2$ to trace$(\widetilde{\Sigma})/C$; and

(c) set $\Sigma = \widetilde{\Sigma}/\alpha_3^2$ and $W = \mu + \epsilon/\alpha_3$.

Appendix D provides more details on the implementation of the algorithms. Software for fitting all three algorithms is available at `https://github.com/yizhenxu/GcompBART`.

## 2.5 Theoretical Comparison of Algorithms for MPBART

In what follows, we assume the Markov chain of $(W, \theta, \Sigma)$ has reached stationary. Liu[13] introduced the usage of diagrams that show dependency structures between two consecutive iterations for analyzing Bayesian algorithms. We do this for Algorithms [KD], [P1], and [P2], and derive their mixing rate in terms of autocovariances. We restate the algorithms under the expanded model $(W, \mu, \Sigma, \alpha)$, where $W$ is the normalized latent variables with distribution MVN$(\mu, \Sigma)$ and $\alpha$ is the expansion parameter:

Algorithm [KD]:

$$(W, \alpha_1)|\mu, \Sigma \quad \Rightarrow \quad \mu|(W, \alpha_1), \Sigma \quad \Rightarrow \quad (\Sigma, \alpha_3)|(W, \alpha_1), \mu,$$

Algorithm [P1]:

$$(W, \alpha_1)|\mu, \Sigma \quad \Rightarrow \quad \mu|W, \Sigma \quad \Rightarrow \quad (\Sigma, \alpha_3)|(W, \alpha_1), \mu,$$

Algorithm [P2]:

$$W|\mu, \Sigma \quad \Rightarrow \quad \mu|W, \Sigma \quad \Rightarrow \quad (\Sigma, \alpha_3)|W, \mu,$$

where $\alpha$'s are indexed as in Scheme [IvD1].

We make a few observations about these three algorithms: (a) Algorithm [KD] groups $W$ and $\alpha_1$ together, as in Scheme [MDA-G]; (b) Algorithm [P1] is structurally equivalent to Scheme [IvD2]; and (c) the sampling of the normalized covariance matrix in all three algorithms integrates out $\alpha_3$ as in Scheme [MDA-C], i.e. $\Sigma \sim \int f(\Sigma, \alpha|W, \mu)d\alpha$ in Algorithm [P2], and $\Sigma \sim \int f(\Sigma, \alpha|W, \alpha_1, \mu)d\alpha$ in Algorithms [KD] and [P1]. Based on these observations, we prove the dependency structure as diagrams in Figure 1.

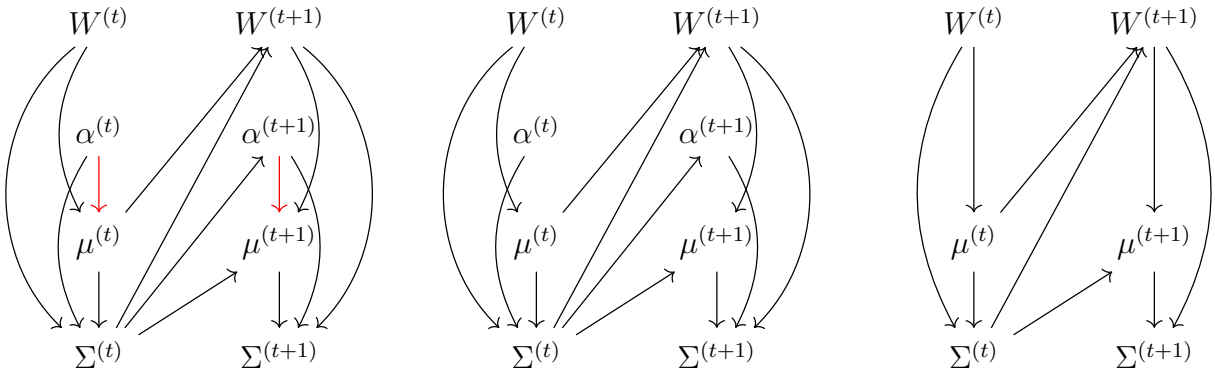



Figure 1: Above diagrams from left to right correspond to Algorithms [KD], [P1], and [P2], respectively.

A common measure for quantifying the mixing rate of a Markov chain is the lag-1 autocorrelation; lower autocorrelation indicates a better mixing rate. Using the dependency diagrams, we argue that Algorithm [P2] has the best mixing rate when the Markov chain is stationary.

**Theorem 1.** *Assuming the chain of MPBART parameters* $(W, \mu, \Sigma, \alpha)$ *has reached equilibrium,*

1. *For* $\mu$*, Algorithms [P1] and [P2] have the same lag-1 autocorrelation, which is no larger than that from Algorithm [KD];*

2. *For* $\Sigma$*, Algorithms [KD] and [P1] have the same lag-1 autocorrelation, which is no less than that from Algorithm [P2].*

Proof: Appendix A.

# 3 Simulation

This simulation study will compare the prediction accuracy of four algorithms, Algorithms [KD], [P1], [P2], and the multinomial BART via conditional probabilities, which is implemented by the $mbart$ function in the **BART** package[23] and denoted by Algorithm [CP] in the rest of the paper. Algorithm [CP] adopts a conditional probit framework and represents the categorical outcome with mutually exclusive binary indicators, to which a sequence of binary BART models are fitted to estimate the conditional probabilities across outcome levels, i.e. $p_{il} = P(S_i = l | S_i > l - 1)$

for $l = 0, \ldots, C-1$, such that the marginal outcome distribution has the form $P(S_i = l) = p_{il} \prod_{v=0}^{l-1}(1 - p_{iv})$ for $l < C$ and $P(S_i = C) = \prod_{v=0}^{C-1}(1 - p_{iv})$.

We consider two different metrics of predictive accuracy which we define as follows. Denote the posterior sample of model parameters by $\{\theta^{(j)}, \Sigma^{(j)} | j = 1, \ldots, J\}$. The posterior predictive distribution for $S_i$ can be represented by its $J$ posterior predictions, $\{\hat{S}_i^{(1)}, \ldots, \hat{S}_i^{(J)}\}$, where

$$\hat{S}_i^{(j)} = \begin{cases} l & \text{if } \max(\hat{W}_i^{(j)}) = \hat{W}_{il}^{(j)} \geq 0 \\ C & \text{if } \max(\hat{W}_i^{(j)}) < 0, \end{cases} \tag{4}$$

$\hat{W}_i^{(j)} = (\hat{W}_{i1}^{(j)}, \ldots, \hat{W}_{i,C}^{(j)})$ is the vector of latent variables, $\hat{W}_i^{(j)} \sim \text{MVN}(G(X_i; \theta^{(j)}), \Sigma^{(j)})$, and

$$G(X_i; \theta^{(j)}) = (G_1(X_i; \theta_1^{(j)}), \ldots, G_C(X_i; \theta_C^{(j)})).$$

Recall that each mean component is parameterized as sum of trees, $G_l(X_i; \theta_l^{(j)}) = \sum_{k=1}^{m} g(X_i; \theta_{lk}^{(j)})$, where $l = 1, \ldots, C$.

We use posterior percent agreement and posterior mode to assess prediction accuracy. While posterior mode accuracy compares the observed outcome $s_i$ and the maximum a posteriori (MAP) estimate of the outcome, posterior percent agreement measures the concordance between $s_i$ and the sampled posterior predictive distribution. Under the multinomial probit framework, Algorithms [KD], [P1] and [P2] directly sample posterior predicted outcomes, $\{\hat{S}_i^{(1)}, \ldots, \hat{S}_i^{(J)}\}$, e.g. the $j$th posterior draw is $\hat{S}_i^{(j)}$; the posterior percent agreement is averaged over $N$ subjects as follows,

$$\frac{1}{N} \sum_{i=1}^{N} \left\{ \frac{1}{J} \sum_{j=1}^{J} \mathbb{1}\{\hat{S}_i^{(j)} = s_i\} \right\}. \tag{5}$$

Algorithm [CP] generates the posterior predicted marginal probabilities, $\{\hat{p}_i^{(1)}, \ldots, \hat{p}_i^{(J)}\}$, where the $j$th posterior draw is $\hat{p}_i^{(j)} = \{\hat{P}^{(j)}(S_i = l); l = 0, \ldots, C\}$. The corresponding posterior

percent agreement is then written as $\frac{1}{N}\sum_{i=1}^{N}\left\{\frac{1}{J}\sum_{j=1}^{J}\hat{P}^{(j)}(S_i = s_i)\right\}$.

Posterior mode accuracy summarizes the agreement between the observed $s_i$ and the posterior mode prediction, $\check{S}_i$, via

$$\frac{1}{N}\sum_{i=1}^{N}\mathbb{1}\{\check{S}_i = s_i\}. \tag{6}$$

For Algorithms [KD], [P1], and [P2], the posterior mode prediction is the most frequent posterior outcome prediction, $\check{S}_i = \text{argmax}_{l\in\{0,\ldots,C\}}\sum_{j=1}^{J}\mathbb{1}\{\hat{S}_i^{(j)} = l\}$. For Algorithm [CP], it is the mode of the average posterior predictive marginal probability, $\text{argmax}_{l\in\{0,\ldots,C\}}\sum_{j=1}^{J}\hat{P}^{(j)}(S_i = l)$. The accuracy measures are different in that the posterior mode accuracy ignores the infrequent categories in MCMC sampling, whereas the posterior percent agreement accounts for all posterior predictive draws.

Numerical experiments for all simulations use 30,000 posterior draws after a burn-in of 50,000 for each model, and parameterize the mean component of each latent variable as the sum of 100 trees. We adopt default setting for estimating Algorithm [CP] using the BART package. The tree priors for the other three algorithms are specified as recommended in Chipman et al.[4], where the prior probabilities for the posterior tree search are 0.25, 0.25, 0.4, and 0.1 for tree GROWTH, PRUNE, CHANGE, and SWAP, respectively. Prior specification of the latent variable covariance matrix assumes the scale matrix $\Psi$ has diagonal elements equal to 1.

For each simulation, we create a training set $\mathcal{D}_1$ and test set $\mathcal{D}_2$, each of size 5000. Under different specifications of the reference level and prior on the covariance matrix, we use the algorithms on $\mathcal{D}_1$. For each set of posterior samples for each algorithm, the corresponding out-of-sample performance is evaluated by calculating the two accuracy metrics on $\mathcal{D}_2$.

We simulate $\mathcal{D}_1$ and $\mathcal{D}_2$ similar to Kang and Schafer[8]. We set $C = 2$ and assume a set of covariates $(U, V)$ where $U = (U_1, \ldots, U_5) \overset{\text{iid}}{\sim} \text{Uniform}(0, 1)$ and $V \sim \text{Uniform}(0, 2)$, and set $G_1 = 15\sin(\pi U_1 U_2) + (U_3 - 0.5)^2 - 10U_4 - 5U_5$. We set $G_2 = (U_3 - 0.5)^3 - 20U_4U_5 + 4V$ in Setting 1 for a relatively balanced distribution of the outcome categories and $G_2 = (U_3 - 0.5)^2 - U_4U_5 + 4V$

in Setting 2 for highly unbalanced outcomes. The covariance matrix is given by $\Sigma = \begin{pmatrix} 1 & 0.5 \\ 0.5 & 1 \end{pmatrix}$ for both settings.

We simulate 50 replicates for each setting. Averaged across simulation replicates, the distribution of the outcome alternatives is ( 0.45, 0.25, 0.30) and (0.32, 0.65, 0.03) for Settings 1 and 2, respectively. Table 1 compares the out-of-sample posterior predictive accuracy of the algorithms, under different priors for the augmented latent covariance, $\tilde{\Sigma} \sim$ Inv-Wishart$(\nu, \Psi)$. Assuming $\Psi_{11} = \Psi_{22} = 1$, we consider uniform ($\nu = C + 1, \Psi_{12} = 0$), negatively tilted ($\nu = C + 3, \Psi_{12} = -0.5$), and positively tilted ($\nu = C + 3, \Psi_{12} = 0.5$) priors. Our proposals and Algorithm [CP] have similar predictive performance based on both accuracy measures. Algorithm [KD] performs well under the posterior mode accuracy, but is relatively more sensitive to the prior specifications and tends to have large variation across posterior predictions, resulting in a sub-optimal performance under the posterior agreement accuracy, which reflects the posterior predictive distribution.

We also investigate how the multinomial probit algorithms behave in estimating $\Sigma$ under different prior specifications, with the same reference level as in data generation. Table 2 summarizes the posterior mean of the normalized covariance matrix $\Sigma$. For $\sigma_{11}$ and $\sigma_{12}$, $E[\cdot|D]$ is the posterior mean based on a simulation replicate $D$; $E\{E[\cdot|D]\}$ and $S\{E[\cdot|D]\}$ are the mean and standard deviation of $E[\cdot|D]$ across the 50 replicates. Note that $\sigma_{22}$ is not displayed in the Table 2 because $\Sigma$ is normalized, satisfying $\sigma_{22} = \text{trace}(\Sigma) - \sigma_{11}$. The true conditional correlation, $\text{corr}(W_1, W_2|G)$, equals 0.5; for the posterior mean of the covariance, $\sigma_{12}$, Algorithm [KD] returns negative estimates while our proposals generate positive estimates, agreeing with the true correlation in sign. Appendix C shows how $\sigma_{12}$ affects the outcome distribution, given $\sigma_{11} = \sigma_{22} = 1$. Conditional on the additive trees, $\sigma_{12}$ has a substantial effect on the outcome predictive distribution, for example, Appendix C illustrated that a negative $\sigma_{12}$ induces smaller reference level outcome probability $P(S = 3)$. Having a negative estimated posterior mean of $\sigma_{12}$ may lead to posterior tree estimates that are systematically different from the simulation truth, where $\sigma_{12}$ is set to be positive. In Ap-

Setting 1

Posterior Agreement Accuracy

| | | Train | | | | Test | | | |
|---|---|---|---|---|---|---|---|---|---|
| $\nu - C$ | $\Psi_{12}$ | KD | P1 | P2 | CP | KD | P1 | P2 | CP |
| 1 | 0 | 0.603 (0.003) | 0.900 (0.004) | 0.900 (0.004) | | 0.580 (0.004) | 0.881 (0.003) | 0.882 (0.003) | |
| 3 | -0.5 | 0.650 (0.004) | 0.900 (0.004) | 0.900 (0.004) | | 0.618 (0.004) | 0.881 (0.003) | 0.882 (0.003) | |
| 3 | 0.5 | 0.647 (0.003) | 0.900 (0.004) | 0.900 (0.004) | | 0.617 (0.004) | 0.881 (0.003) | 0.882 (0.003) | |
| | | | | | 0.869 (0.003) | | | | 0.855 (0.003) |

Posterior Mode Accuracy

| | | Train | | | | Test | | | |
|---|---|---|---|---|---|---|---|---|---|
| $\nu - C$ | $\Psi_{12}$ | KD | P1 | P2 | CP | KD | P1 | P2 | CP |
| 1 | 0 | 0.921 (0.005) | 0.946 (0.003) | 0.946 (0.003) | | 0.872 (0.006) | 0.919 (0.004) | 0.919 (0.004) | |
| 3 | -0.5 | 0.932 (0.004) | 0.946 (0.003) | 0.946 (0.003) | | 0.872 (0.006) | 0.919 (0.003) | 0.919 (0.004) | |
| 3 | 0.5 | 0.928 (0.004) | 0.946 (0.004) | 0.946 (0.003) | | 0.873 (0.006) | 0.919 (0.004) | 0.919 (0.003) | |
| | | | | | 0.937 (0.004) | | | | 0.916 (0.004) |

Setting 2

Posterior Agreement Accuracy

| | | Train | | | | Test | | | |
|---|---|---|---|---|---|---|---|---|---|
| $\nu - C$ | $\Psi_{12}$ | KD | P1 | P2 | CP | KD | P1 | P2 | CP |
| 1 | 0 | 0.651 (0.003) | 0.905 (0.003) | 0.905 (0.003) | | 0.632 (0.005) | 0.881 (0.003) | 0.882 (0.003) | |
| 3 | -0.5 | 0.701 (0.003) | 0.905 (0.003) | 0.905 (0.003) | | 0.676 (0.005) | 0.882 (0.003) | 0.882 (0.003) | |
| 3 | 0.5 | 0.700 (0.003) | 0.906 (0.003) | 0.906 (0.004) | | 0.679 (0.005) | 0.882 (0.003) | 0.882 (0.004) | |
| | | | | | 0.882 (0.003) | | | | 0.87 (0.003) |

Posterior Mode Accuracy

| | | Train | | | | Test | | | |
|---|---|---|---|---|---|---|---|---|---|
| $\nu - C$ | $\Psi_{12}$ | KD | P1 | P2 | CP | KD | P1 | P2 | CP |
| 1 | 0 | 0.924 (0.004) | 0.953 (0.003) | 0.952 (0.003) | | 0.896 (0.005) | 0.918 (0.004) | 0.918 (0.004) | |
| 3 | -0.5 | 0.930 (0.003) | 0.952 (0.003) | 0.952 (0.003) | | 0.893 (0.006) | 0.918 (0.004) | 0.918 (0.004) | |
| 3 | 0.5 | 0.927 (0.003) | 0.952 (0.003) | 0.952 (0.003) | | 0.895 (0.005) | 0.918 (0.004) | 0.918 (0.004) | |
| | | | | | 0.939 (0.003) | | | | 0.920 (0.004) |

Table 1: Accuracy comparison of Algorithms [KD], [P1], and [P2] under reference level 3 with 50 replications. Training and test datasets each with 5000 observations are generated under Settings 1 and 2 with reference level 3. Average (standard deviation) of accuracy across the 50 rounds are reported. The prior of $\widetilde{\Sigma}$ is Inv-Wishart$(\nu, \Psi)$, where $\Psi_{11} = \Psi_{22} = 1$. Posterior predictive accuracy measured by (5) and (6) are reported under $(\nu - C, \Psi_{12})$ being $(1, 0)$, $(3, -0.5)$, and $(3, 0.5)$.

pendix E, we provide diagnostic plots of the MCMC convergence in the first simulation replicate of the two settings for the sum-of-trees and the covariance components. Our proposals converge faster than Algorithm [KD] because the latter updates the sum-of-trees component conditional on latent utilities that are augmented / not normalized, which makes posterior convergence more challenging. When the outcome is unbalanced, posterior convergence is more difficult.

Setting 1

| | | $E\{E[\sigma_{11}\|D]\}(S\{E[\sigma_{11}\|D]\})$ | | |
|---|---|---|---|---|
| $\nu - C$ | $\Psi_{12}$ | KD | P1 | P2 |
| 1 | 0 | 1.311 (0.032) | 1.035 (0.041) | 1.039 (0.039) |
| 3 | -0.5 | 1.325 (0.057) | 1.035 (0.036) | 1.034 (0.036) |
| 3 | 0.5 | 1.296 (0.059) | 1.039 (0.042) | 1.038 (0.042) |

| | | $E\{E[\sigma_{12}\|D]\}(S\{E[\sigma_{12}\|D]\})$ | | |
|---|---|---|---|---|
| $\nu - C$ | $\Psi_{12}$ | KD | P1 | P2 |
| 1 | 0 | -0.108 (0.007) | 0.344 (0.053) | 0.354 (0.056) |
| 3 | -0.5 | -0.118 (0.009) | 0.321 (0.057) | 0.348 (0.062) |
| 3 | 0.5 | -0.122 (0.010) | 0.365 (0.054) | 0.359 (0.059) |

Setting 2

| | | $E\{E[\sigma_{11}\|D]\}(S\{E[\sigma_{11}\|D]\})$ | | |
|---|---|---|---|---|
| $\nu - C$ | $\Psi_{12}$ | KD | P1 | P2 |
| 1 | 0 | 0.848 (0.051) | 0.769 (0.046) | 0.770 (0.041) |
| 3 | -0.5 | 0.599 (0.059) | 0.783 (0.047) | 0.774 (0.041) |
| 3 | 0.5 | 0.691 (0.067) | 0.779 (0.036) | 0.758 (0.049) |

| | | $E\{E[\sigma_{12}\|D]\}\quad(S\{E[\sigma_{12}\|D]\})$ | | |
|---|---|---|---|---|
| $\nu - C$ | $\Psi_{12}$ | KD | P1 | P2 |
| 1 | 0 | -0.321 (0.009) | 0.801 (0.029) | 0.797 (0.025) |
| 3 | -0.5 | -0.349 (0.011) | 0.782 (0.028) | 0.791 (0.026) |
| 3 | 0.5 | -0.354 (0.011) | 0.801 (0.026) | 0.802 (0.028) |

Table 2: Comparison of Algorithms [KD], [P1], and [P2] under reference level 3 on $\Sigma$ with 50 replications. Training and test datasets each with 5000 observations are generated under Settings 1 and 2, setting reference level to 3. $E[\cdot|D]$ indicates sample mean on one simulated data $D$. $E\{E[\cdot|D]\}$ and $S\{E[\cdot|D]\}$ are the mean and sd of $E[\cdot|D]$ across the 50 simulations of $D$. The prior of $\widetilde{\Sigma}$ is Inv-Wishart$(\nu, \Psi)$, where $\Psi_{11} = \Psi_{22} = 1$. Posterior predictive accuracy measured by (5) and (6) are reported under $(\nu - C, \Psi_{12})$ being $(1, 0)$, $(3, -0.5)$, and $(3, 0.5)$.

# 4 Application

In this application, we investigate patients' retention in HIV care after enrollment as a function of their baseline characteristics and treatment status. The data were extracted from electronic health records of adults enrolled in HIV care between June 1st 2008 and August 23rd 2016 in AMPATH, an HIV care program in Kenya. We look at a 200-days window after the initial care encounter and split the data into training and test sets of sample sizes 49,942 and 26,714, respectively. We define the outcome as disengagement, engagement, and reported death, where engagement in care means there was at least one visit to the clinic for HIV care during the first 200 days after a patient's initial encounter, and disengagement otherwise if the person was not reported dead. The outcome distribution is extremely imbalanced, such that the frequency of disengagement, engagement, and death is $16\%$, $80\%$, and $4\%$, respectively. Covariates include baseline age, gender, year of enrollment, travel time to clinic, marriage status, weight, height, baseline treatment status, indication of CD4 measurement at or post baseline, and the most recent CD4. Table 3 summarizes the observed distribution of each covariate stratified by outcome level.

We use 10,000 posterior draws after a burn-in of 10,000 and keep other settings the same as in simulations. Table 4 compares the posterior accuracy for Algorithms [KD], [P1], [P2], and [CP]. Algorithm [KD] has posterior mode accuracy comparative to, but not as good as, that from our proposals and Algorithm [CP]. Algorithm [KD] is not separating the latent utilities of the true outcome level and those for the other outcome alternatives well, resulting in a less ideal posterior agreement accuracy. In terms of the stability in accuracy measures with respect to the choice of reference level, the performance of the proposals is similar to Algorithm [CP] and better than Algorithm [KD].

Under the reference level being disengagement, the first row of Figure 2 presents the MCMC convergence plots of the average tree depth corresponding to latent variables $W_1 = Z_{\text{eng}} - Z_{\text{diseng}}$ and $W_2 = Z_{\text{death}} - Z_{\text{diseng}}$, and the histogram of the posteriors of $\sigma_{12} = \text{Cov}(W_1, W_2)$, where $(Z_{\text{eng}}, Z_{\text{diseng}}, Z_{\text{death}})$ are latent utilities corresponding to each of the outcome levels and $\sigma_{12}$ is the

| | | Disengaged (6497) | Engaged (67462) | Died (2697) |
|---|---|---|---|---|
| Male | | 22.5 | 34 | 51.3 |
| Year of Enrolment | 2008 | 5.1 | 9.7 | 11.2 |
| | 2009 | 8.3 | 18.7 | 17.1 |
| | 2010 | 9.3 | 17.3 | 17.6 |
| | 2011 | 9.2 | 15.8 | 17 |
| | 2012 | 17.9 | 11.5 | 14 |
| | 2013 | 18.5 | 8.9 | 11.3 |
| | 2014 | 18.8 | 9.0 | 8.2 |
| | 2015 | 12.8 | 8.3 | 3.3 |
| | 2016 | 0.3 | 0.8 | 0.3 |
| Travel Time | <30 min | 17.4 | 24 | 23.6 |
| | 30 min - 1 h | 19.4 | 26.9 | 29.4 |
| | 1 h - 2 h | 8.2 | 14.6 | 16.5 |
| | >2 h | 5.2 | 7.7 | 7.8 |
| | Missing | 49.9 | 26.8 | 22.6 |
| WHO Stage | 1 | 13.7 | 4.7 | 1.0 |
| | 2 | 1.8 | 2.0 | 1.1 |
| | 3 | 2.3 | 2.2 | 4.3 |
| | 4 | 0.6 | 0.3 | 0.7 |
| | Missing | 81.7 | 90.7 | 92.9 |
| Married | | 57.2 | 52.3 | 49.7 |
| | Missing | 13.6 | 8.3 | 6.2 |
| On ART | | 39.9 | 14.1 | 14.1 |
| CD4 at/post baseline | | 64.8 | 80.9 | 74.7 |
| Post-baseline CD4 update | | 6.6 | 26 | 7.6 |
| Most recent CD4 | | 327 (144, 525) | 279.77 (137, 462) | 59 (18, 152) |
| Age | | 29.91 (24.66, 36.51) | 35.56 (28.93, 43.65) | 37.97 (31.7, 45.7) |
| Height | | 163 (158, 169) | 165 (159.1, 171) | 167 (160, 173) |
| | Missing | 24.6 | 16.2 | 17.7 |
| Weight | | 57.5 (51, 65) | 56 (50, 63) | 50 (44, 57) |
| | Missing | 7.9 | 3.9 | 6.9 |

Table 3: Summary table of covariates stratified by outcome. The table reports "median (25th percentile, 75th percentile)" for continuous variables and percentage of true for binary variables or each level of categorical variables.

Posterior Agreement Accuracy

| | Train | | | | Test | | | |
|---|---|---|---|---|---|---|---|---|
| Ref Level | KD | P1 | P2 | CP | KD | P1 | P2 | CP |
| 1 | 0.67 | 0.82 | 0.82 | | 0.67 | 0.81 | 0.81 | |
| 2 | 0.55 | 0.82 | 0.82 | | 0.54 | 0.81 | 0.81 | |
| 3 | 0.66 | 0.82 | 0.82 | | 0.66 | 0.81 | 0.81 | |
| | | | | 0.81 | | | | 0.81 |

Posterior Mode Accuracy

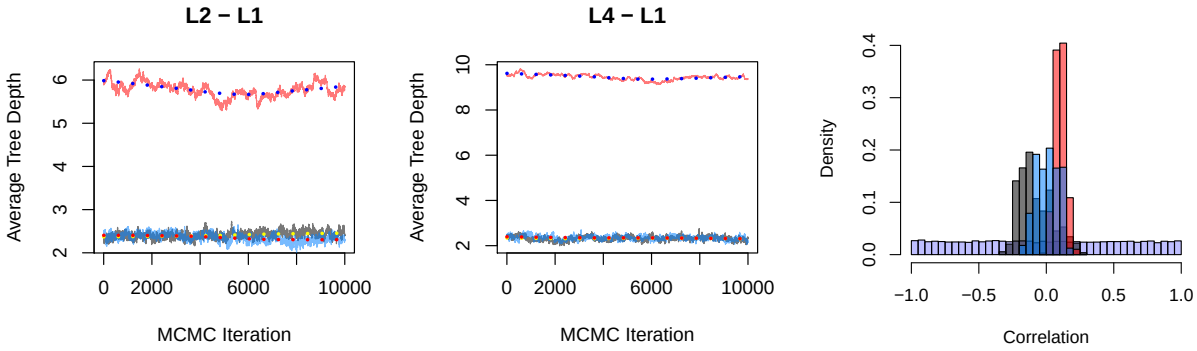
| | Train | | | | Test | | | |
|---|---|---|---|---|---|---|---|---|
| Ref Level | KD | P1 | P2 | CP | KD | P1 | P2 | CP |
| 1 | 0.88 | 0.89 | 0.89 | | 0.88 | 0.89 | 0.89 | |
| 2 | 0.85 | 0.89 | 0.89 | | 0.84 | 0.89 | 0.89 | |
| 3 | 0.88 | 0.89 | 0.89 | | 0.88 | 0.89 | 0.89 | |
| | | | | 0.89 | | | | 0.89 |

Table 4: Accuracy comparison of Algorithms [KD], [P1], and [P2] on the AMPATH data. Posterior predictive accuracy measured by (5) and (6) are reported under reference levels 1, 2, and 3. The prior of $\widetilde{\Sigma}$ is Inv-Wishart$(3, I_3)$.

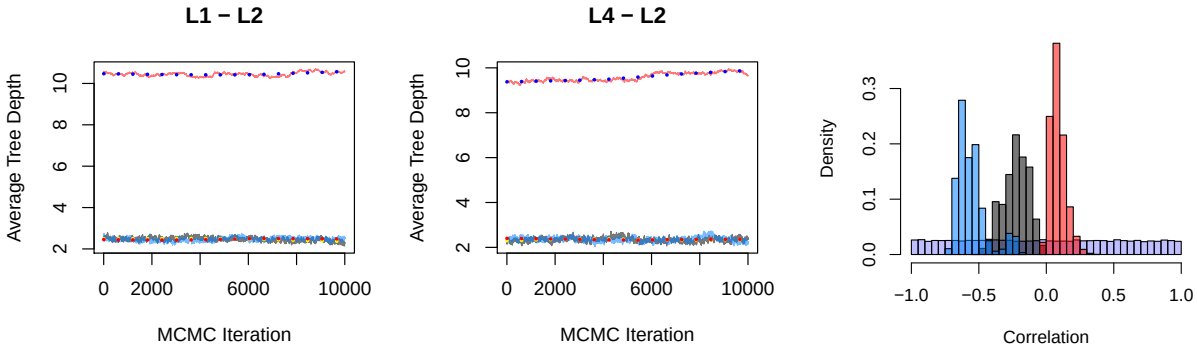
normalized conditional covariance of $W_1$ and $W_2$. The plots show that the average tree depths are around 6 and 9 respectively for $W_1$ and $W_2$ under Algorithm [KD], and approximately 2 for those under Algorithms [P1] and [P2]. The Bayesian regularization priors that favor shallow trees do not work well for Algorithm [KD], as a tree depth of 6 allows up to $2^6$ leaves, which increases the risk of over-fitting and makes the stochastic tree search inefficient. The second and third rows of Figure 2 set engagement and reported death as the reference level, respectively, and the latent variables are defined accordingly. Similar conclusions are observed for tree depth. Under all choices of the reference level, the histogram of $\sigma_{12}$ from Algorithms [P1] and [P2] agree on the sign of $\sigma_{12}$, which was demonstrated in previous simulations to match the sign of the true value of the underlying $\sigma_{12}$.

# 5 Concluding Remarks

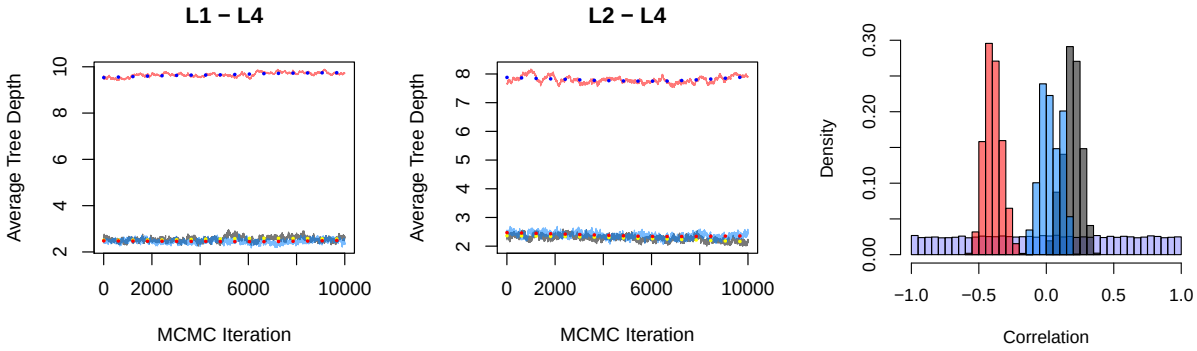
While computational performance is an important criterion in building Gibbs sampler for complicated models, the dependency structure and sampling schemes are as crucial for devising an algorithm that generates a Markov chain with computational efficiency and fast mixing rates. We




(a) Reference level 1 (disengagement)




(b) Reference level 2 (engagement)




(c) Reference level 4 (death)

Figure 2: Traceplot of posterior average tree depth for each latent utility in the application to AMPATH data (left), and histogram of the $\sigma_{12}$ (right) under its prior (purple), posterior from Algorithms [KD] (red), [P1] (black), and [P2] (blue); same color specification applies to the left plot. Posterior inference is under $\nu = C + 1$, $\Psi_{12} = 0$, with reference level as indicated in the plot labels.

explore the data augmentation schemes involved in the Bayesian estimation of multinomial probit models and propose two alternative algorithms that improve the computational and theoretical properties of the estimating procedure of MPBART proposed in Kindo et al.[9]. Theoretically, we prove that the mixing rate of our proposals is at least as good as Algorithm [KD] for the mean and covariance matrix of the latent variables.

We evaluate the algorithms' computational performance under the same parameter specifications using two accuracy measures: posterior percent agreement and posterior mode accuracy. Posterior mode accuracy, which compares observed categorical outcomes to the mode in posterior predictions, is widely used in machine learning literature, particularly in cross-sectional supervised learning studies such as Kindo et al.[9]. Alternatively, posterior percent agreement accounts for the posterior predictive probabilities of the outcome labels, so the estimated distribution of the non-dominant levels also influences the metric. In applications where multinomial models are used as generative components, posterior predictive distribution is more relevant than posterior mode predictions and it is crucial to examine the posterior predictive distribution of the categorical outcomes.

Through simulations and application, we compare our proposals to the estimating procedure in Kindo et al.[9] (Algorithm [KD]) and the $mbart$ function in the **BART** package (Algorithm [CP]). We find that our proposals and Algorithm [CP] have similar predictive performances; however, while Algorithm [KD] performs well in terms of posterior mode, its posterior percent agreement is less ideal. One possible explanation is that Algorithm [KD] samples posterior trees conditional on augmented latent variables, making posterior convergence more challenging; this may undermine the Bayesian regularization priors in BART, resulting in larger trees and higher computational costs, and lead to exploration of the latent correlation structure in a parameter space different from the truth. In Appendix C we further explore how the correlation of the latent variables affects the outcome distribution, demonstrating that an estimated covariance of the wrong sign may be associated with a sum-of-trees component with values that are systematically different

from the true data generating mechanism.

Even though our proposals are similar to Algorithm [CP] in terms of predictive performance, we point out that Sparapani et al.'s[23] approach is fundamentally different from MPBART because it models the multinomial outcome sequentially based on an ordering of the categories, and thus the approach is not invariant to the ordering. On the other hand, MPBART jointly models the latent utilities using a multivariate Gaussian BART such that the predictions do not depend on the ordering.

The two approaches have a subtle yet important distinction in actual application. Suppose in an application where we jointly model patients' transfer out, engagement, disengagement, and death using Algorithm [CP] with the following sequential models: $P(\text{transfer} = 1)$, $P(\text{death} = 1|\text{transfer} = 1)$, and $P(\text{engaged} = 1|\text{transfer} = 0, \text{death} = 0)$. In this case, the probability of death is independent of the choice between engagement and disengagement. This is not preferable in practice because we would expect either a higher disengagement rate to be associated with a lower death reporting rate or a higher engagement rate to be associated with a healthier status and thus a lower death rate. Additionally, the key feature separating the multinomial probit framework from other choice models, e.g. multinomial logit or conditional probit, is the assumption that the stochastic terms have a multivariate Gaussian distribution with a covariance matrix. This formulation enables natural extensions to multilevel data models by incorporating dependence structures with additional complexity, such as including random effects in the model to account for clinic clustering effect in the joint modeling of HIV care engagement and mortality using electronic health records.

# References

[1] Albert, J. H. and S. Chib (1993). Bayesian Analysis of Binary and Polychotomous Response Data. *Journal of the American Statistical Association 88*(422), 669–679.

[2] Burgette, L. F. and E. V. Nordheim (2012, July). The Trace Restriction: An Alternative

Identification Strategy for the Bayesian Multinomial Probit Model. *Journal of Business & Economic Statistics 30*(3), 404–410.

[3] Chipman, H. A., E. I. George, and R. E. McCulloch (1998, September). Bayesian CART Model Search. *Journal of the American Statistical Association 93*(443), 935–948.

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

[5] Dempster, A. P., N. M. Laird, and D. B. Rubin (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological) 39*(1), 1–38.

[6] Gardner, E. M., M. P. McLees, J. F. Steiner, C. Del Rio, and W. J. Burman (2011, March). The spectrum of engagement in HIV care and its relevance to test-and-treat strategies for prevention of HIV infection. *Clinical Infectious Diseases: An Official Publication of the Infectious Diseases Society of America 52*(6), 793–800.

[7] Imai, K. and D. A. van Dyk (2005). A Bayesian analysis of the multinomial probit model using marginal data augmentation. *Journal of Econometrics*, 311 – 334.

[8] Kang, J. D. and J. L. Schafer (2007). Demystifying double robustness: A comparison of alternative strategies for estimating a population mean from incomplete data. *Statistical Science*, 523–539.

[9] Kindo, B. P., H. Wang, and E. A. Peña (2016). Multinomial probit Bayesian additive regression trees. *Stat (International Statistical Institute) 5*(1), 119–131.

[10] Lee, H., B. L. Genberg, M. Nyambura, J. Hogan, P. Braitstein, and E. Sang (2017). State-Space Models for Engagement, Retention, and Reentry in the HIV Care Cascade. *CROI Conference*.

[11] Li, Z. R., T. H. McComick, and S. J. Clark (2018). Using Bayesian Latent Gaussian Graphical Models to Infer Symptom Associations in Verbal Autopsies. *Bayesian Analysis*.

[12] Liu, J. S. (1994). The Collapsed Gibbs Sampler in Bayesian Computations with Applications to a Gene Regulation Problem. *Journal of the American Statistical Association 89*(427), 958–966.

[13] Liu, J. S. (1994a). Fraction of Missing Information and Convergence Rate of Data Augmentation. Research Triangle Park, North Carolina.

[14] Liu, J. S., W. H. Wong, and A. Kong (1994). Covariance Structure of the Gibbs Sampler with Applications to the Comparisons of Estimators and Augmentation Schemes. *Biometrika 81*(1), 27–40.

[15] Liu, J. S. and Y. N. Wu (1999, December). Parameter Expansion for Data Augmentation. *Journal of the American Statistical Association 94*(448), 1264–1274.

[16] Low-Kam, C., D. Telesca, Z. Ji, H. Zhang, T. Xia, J. I. Zink, and A. E. Nel (2015, March). A Bayesian regression tree approach to identify the effect of nanoparticles' properties on toxicity profiles. *Annals of Applied Statistics 9*(1), 383–401. Publisher: Institute of Mathematical Statistics.

[17] McCulloch, R. and P. Rossi (1994). An exact likelihood analysis of the multinomial probit model. *Journal of Econometrics 64*(1-2), 207–240.

[18] McCulloch, R. E., N. G. Polson, and P. E. Rossi (2000, November). A Bayesian analysis of the multinomial probit model with fully identified parameters. *Journal of Econometrics 99*(1), 173–193.

[19] McFadden, D. (1974). Conditional logit analysis of qualitative choice behavior. *Frontiers in econometrics*.

[20] Meng, X.-L. and D. A. Van Dyk (1999). Seeking Efficient Data Augmentation Schemes via Conditional and Marginal Augmentation. *Biometrika 86*(2), 301–320.

[21] Murray, J. S. (2020, August). Log-Linear Bayesian Additive Regression Trees for Multinomial Logistic and Count Regression Models. *Journal of the American Statistical Association 116*(534).

[22] Nobile, A. (1998, August). A hybrid Markov chain for the Bayesian analysis of the multinomial probit model. *Statistics and Computing 8*(3), 229–242.

[23] Sparapani, R., C. Spanbauer, and R. McCulloch (2021). Nonparametric machine learning and efficient computation with bayesian additive regression trees: the bart r package. *Journal of Statistical Software 97*, 1–66.

[24] Sparapani, R. A., B. R. Logan, R. E. McCulloch, and P. W. Laud (2016, July). Nonparametric survival analysis using Bayesian Additive Regression Trees (BART). *Statistics in Medicine 35*(16), 2741–2753.

[25] Tanner, M. A. and W. H. Wong (1987). The Calculation of Posterior Distributions by Data Augmentation. *Journal of the American Statistical Association 82*(398), 528–540.

[26] Van Dyk, D. and X.-L. Meng (2001). The Art of Data Augmentation. *The Journal of Computational and Graphical Statistics 10*, 1–111.

[27] van Dyk, D. A. (2010). MARGINAL MARKOV CHAIN MONTE CARLO METHODS. *Statistica Sinica 20*(4), 1423–1454.

[28] Waldmann, P. (2016, June). Genome-wide prediction using Bayesian additive regression trees. *Genetics, Selection, Evolution : GSE 48*.

[29] WHO (2012). Meeting report on Framework for metrics to support effective treatment as prevention.

# Appendix

## A Proof of Theorem 1

The lag-1 autocorrelation of $\mu$ is defined as $\text{corr}(\mu^{(t)}, \mu^{(t+1)}) = \frac{\text{cov}(\mu^{(t)}, \mu^{(t+1)})}{\sqrt{\text{var}(\mu^{(t)})\text{var}(\mu^{(t+1)})}}$, where $t$ indexes posterior draws. Under the condition that the chain has reached its stationary distribution $f(W, \mu, \Sigma, \alpha|S, X)$, where $S$ and $X$ are observed outcome and covariates, $\text{var}(\mu^{(t)}) = \text{var}(\mu^{(t+1)}) = \text{var}(\mu)$. Hence, we only need to look at the covariance for comparing the autocorrelation.

Consider two consecutive draws of $\mu$ from the Algorithm [P1]. We find that

$$\begin{aligned}
E(\mu^{(t)}\mu^{(t+1)}) =& E[E(\mu^{(t)}\mu^{(t+1)}|\Sigma^{(t)}, W^{(t+1)}, \alpha^{(t+1)})] \\
=& E[E(\mu^{(t)}|\Sigma^{(t)}, W^{(t+1)}, \alpha^{(t+1)})] \\
& E[E(\mu^{(t+1)}|\Sigma^{(t)}, W^{(t+1)}, \alpha^{(t+1)})] \\
=& E[E(\mu|\Sigma, W, \alpha)^2],
\end{aligned}$$

where the first equality follows from the law of total expectation; the second and the third equalities follow from the fact that $\mu^{(t)}$ and $\mu^{(t+1)}$ are conditionally independent and identically distributed given $(\Sigma^{(t)}, W^{(t+1)}, \alpha^{(t+1)})$. This can be seen from the diagram for Algorithm [KD] in Figure 1; in particular, $\mu^{(t)}$ connects with $\mu^{(t+1)}$ only through $(\Sigma^{(t)}, W^{(t+1)}, \alpha^{(t+1)})$. As a result, the covariance is given by

$$\begin{aligned}
\text{cov}(\mu^{(t)}, \mu^{(t+1)}) &= E(\mu^{(t)}\mu^{(t+1)}) - E(\mu^{(t)})E(\mu^{(t+1)}) \\
&= E[E(\mu|\Sigma, W, \alpha)^2] - E[E(\mu|\Sigma, W, \alpha)]^2 \\
&= \text{var}[E(\mu|\Sigma, W, \alpha)].
\end{aligned}$$

Similarly, the covariance under Algorithms [P1] and [P2] is derived to be $\text{var}[E(\mu|\Sigma, W)]$. The key to this calculation is the fact that $\mu^{(t)}$ connects with $\mu^{(t+1)}$ only through $(\Sigma^{(t)}, W^{(t+1)})$ (see

Figure 1). We now compare the two variances,

$$\begin{aligned}
\text{var}[E(\mu|\Sigma, W, \alpha)] =& \text{var}\{E[E(\mu|\Sigma, W, \alpha)|\Sigma, W]\} \\
&+ E\{\text{var}[E(\mu|\Sigma, W, \alpha)|\Sigma, W]\} \\
\geq& \text{var}\{E[E(\mu|\Sigma, W, \alpha)|\Sigma, W]\} \\
=& \text{var}[E(\mu|\Sigma, W)].
\end{aligned}$$

The first equality comes from the law of total conditional variance and the last equality results from the law of total expectation. Therefore, we can conclude that the lag-1 autocorrelation of $\mu$, $\text{corr}(\mu^{(t)}, \mu^{(t+1)})$ is closer to zero in Algorithms [P1] and [P2] than in Algorithm [KD].

Recall from Figure 1 that, $\Sigma^{(t)}$ connects with $\Sigma^{(t+1)}$ only through $(W^{(t+1)}, \alpha^{(t+1)}, \mu^{(t+1)})$ in Algorithms [KD] and [P1], and through $(W^{(t+1)}, \mu^{(t+1)})$ in Algorithm [P2]. Hence, with a similar argument as above, we can show that the the lag-1 autocorrelation of $\Sigma$, $\text{corr}(\Sigma^{(t)}, \Sigma^{(t+1)})$, for Algorithm [P2] is no larger than in Algorithms [KD] and [P1].

# B Equivalence in the Sampling Distribution between Algorithms [P1] and [P2]

The purpose of this section is to show that Algorithms [P1] and [P2] have the same sampling distribution of $\Sigma$ when the sample size of the observed data, $N$, is sufficiently large and the scale matrix in the prior distribution of $\widetilde{\Sigma}$ is relatively small compared to the sample estimate of the covariance matrix for the latent variables. Note that Algorithms [P1] and [P2] draw $\theta$ and $W$ from the same conditional distributions. Hence, the conclusion here implies that the two procedures sample from the same joint distribution of $(\theta, W, \Sigma)$.

Given the prior on the unconstrained covariance matrix $\widetilde{\Sigma} \sim \text{Inv-Wishart}(\nu, \Psi)$, we can calculate in closed form the conditional posterior distributions of $\widetilde{\Sigma}$ and $\Sigma$ under Algorithms [P1] and [P2], respectively, where $\Sigma = \widetilde{\Sigma}\frac{C}{\text{trace}(\widetilde{\Sigma})}$.

In Algorithm [P1], Step 1 samples the working parameter $\alpha_1^2$ from its conditional prior distri-

bution,

$$\alpha^2|\Sigma \sim \text{Inv-Gamma}(\nu C/2, \text{trace}(\Psi\Sigma^{-1})/2),$$

which is equivalent to $\text{trace}(\Psi\Sigma^{-1})/\chi^2_{\nu C}$[7] and has expectation

$$E[\alpha_1^2|\Sigma] = \text{trace}(\Psi\Sigma^{-1})/(\nu C - 2).$$

Since inverse-Wishart is conditionally conjugate to the covariance matrix in a Gaussian model, the conditional posterior distribution of $\widetilde{\Sigma}$ under Algorithm [P1] is

$$\widetilde{\Sigma}|\alpha_1^2, W, \mu \sim \text{Inv-Wishart}(N + \nu, \Psi + \alpha_1^2 \sum_{i=1}^{N}(W_i - \mu_i)(W_i - \mu_i)^T),$$

and this leads to the conditional posterior distribution of the corresponding restricted covariance matrix $\Sigma$,

$$f(\Sigma|\alpha_1^2, W, \mu) \propto |\Sigma|^{-(N+\nu+C+1)/2} \times$$
$$\text{trace}\left\{[\Psi + \alpha_1^2 \textstyle\sum_{i=1}^{N}(W_i - \mu_i)(W_i - \mu_i)^T]\Sigma^{-1}\right\}^{-\nu C/2}.$$

Similarly, the conditional posterior distributions under Algorithm [P2] for $\widetilde{\Sigma}$ and $\Sigma$, respectively, are

$$\widetilde{\Sigma}|W, \mu \sim \text{Inv-Wishart}(N + \nu, \Psi + \textstyle\sum_{i=1}^{N}(W_i - \mu_i)(W_i - \mu_i)^T),$$
$$f(\Sigma|W, \mu) \propto |\Sigma|^{-(N+\nu+C+1)/2} \times$$
$$\text{trace}\left\{[\Psi + \textstyle\sum_{i=1}^{N}(W_i - \mu_i)(W_i - \mu_i)^T]\Sigma^{-1}\right\}^{-\nu C/2}.$$

In order to compare the posterior conditional of $\Sigma$ under Algorithms [P1] and [P2], we look at the posterior mean and variance using a first-order Taylor series expansion. The posterior mean

under Algorithm [P1] is

$$
\begin{aligned}
E(\Sigma|\alpha_1^2, W, \mu) &= E\left(\frac{\widetilde{\Sigma}}{\alpha^2}\middle|\alpha_1^2, W, \mu\right) \approx \frac{E(\widetilde{\Sigma}|\alpha_1^2, W, \mu)}{E(\alpha^2|\alpha_1^2, W, \mu)} \\
&= \frac{E(\widetilde{\Sigma}|\alpha_1^2, W, \mu)}{E[\text{trace}(\widetilde{\Sigma})|\alpha_1^2, W, \mu]} \times C \\
&= \frac{E(\widetilde{\Sigma}|\alpha_1^2, W, \mu)}{\text{trace}[E(\widetilde{\Sigma}|\alpha_1^2, W, \mu)]} \times C \\
&= \frac{\Psi + \alpha_1^2 \sum_{i=1}^N (W_i - \mu_i)(W_i - \mu_i)^T}{\text{trace}[\Psi + \alpha_1^2 \sum_{i=1}^N (W_i - \mu_i)(W_i - \mu_i)^T)]} \times C \\
&= \frac{\Psi + \alpha_1^2 \sum_{i=1}^N (W_i - \mu_i)(W_i - \mu_i)^T}{\text{trace}(\Psi) + \alpha_1^2 \sum_{i=1}^N \sum_{j=1}^C (W_{ij} - \mu_{ij})^2} \times C
\end{aligned}
$$

Similarly, the posterior mean of $\Sigma$ under Algorithm [P2] is

$$
E(\Sigma|W, \mu) = \frac{\Psi + \sum_{i=1}^N (W_i - \mu_i)(W_i - \mu_i)^T}{\text{trace}(\Psi) + \sum_{i=1}^N \sum_{j=1}^C (W_{ij} - \mu_{ij})^2} \times C.
$$

For the posterior variance of $\Sigma$, we simplify the notation by writing the posterior conditional distribution of $\widetilde{\Sigma}$ as Inv-Wishart$(\widetilde{\nu}, \widetilde{\Psi})$, where $\widetilde{\nu} = N + \nu$ and the scale matrix $\widetilde{\Psi}$ is

$$
\begin{cases}
\Psi + \alpha_1^2 \sum_{i=1}^N (W_i - \mu_i)(W_i - \mu_i)^T & \text{under Algorithm [P1]} \\
\Psi + \sum_{i=1}^N (W_i - \mu_i)(W_i - \mu_i)^T & \text{under Algorithm [P2]}
\end{cases}
$$

Then, the posterior variance has the following form,

$$
\begin{aligned}
&\text{var}(\sigma_{ij}) = \text{var}(\frac{\widetilde{\sigma}_{ij}}{\alpha^2}) \\
&\approx \frac{E(\widetilde{\sigma}_{ij})^2}{E(\alpha^2)^2} \times \left\{ \frac{\text{var}(\widetilde{\sigma}_{ij})}{E(\widetilde{\sigma}_{ij})^2} - 2\frac{\text{cov}(\widetilde{\sigma}_{ij}, \alpha^2)}{E(\widetilde{\sigma}_{ij})E(\alpha^2)} + \frac{\text{var}(\alpha^2)}{E(\alpha^2)^2} \right\}
\end{aligned}
$$

where

$$
\begin{aligned}
E(\widetilde{\sigma}_{ij}) &= \widetilde{\Psi}_{ij}/(\widetilde{\nu}-C-1) \\
E(\alpha^2) &= E(\widetilde{\sigma}_{11}+\ldots+\widetilde{\sigma}_{CC}) = \text{trace}(\widetilde{\Psi})/(\widetilde{\nu}-C-1) \\
\text{var}(\widetilde{\sigma}_{ij}) &= \begin{cases} \frac{(\widetilde{\nu}-C+1)\widetilde{\Psi}_{ij}^2+(\widetilde{\nu}-C-1)\widetilde{\Psi}_{ii}\widetilde{\Psi}_{jj}}{(\widetilde{\nu}-C)(\widetilde{\nu}-C-1)^2(\widetilde{\nu}-C-3)} & \text{if } i \neq j \\ \frac{2\widetilde{\Psi}_{ii}^2}{(\widetilde{\nu}-C-1)^2(\widetilde{\nu}-C-3)} & \text{if } i = j \end{cases} \\
\text{var}(\alpha^2) &= \frac{2}{(\widetilde{\nu}-C-1)^2(\widetilde{\nu}-C-3)}\sum_{i=1}^{C}\widetilde{\Psi}_{ii}^2 \\
\text{cov}(\widetilde{\sigma}_{ij},\alpha^2) &= \sum_{k=1}^{C}\text{cov}(\widetilde{\sigma}_{ij},\widetilde{\sigma}_{kk}) \\
&= 2\frac{\widetilde{\Psi}_{ij}\widetilde{\Psi}_{kk}+(\widetilde{\nu}-C-1)\widetilde{\Psi}_{ik}\widetilde{\Psi}_{jk}}{(\widetilde{\nu}-C)(\widetilde{\nu}-C-1)^2(\widetilde{\nu}-C-3)}.
\end{aligned}
$$

Specifying the inverse-Wishart Prior for the covariance matrix is analogous to assuming a prior knowledge of $\nu$ Gaussian samples having covariance matrix $\Psi$. When the number of observations $N$ gets larger, the posterior concentrates more on the empirical covariance matrix. When $N$ is sufficiently large and $\Psi_{kj} << \alpha_1^2\sum_{i=1}^{N}\sum_{j=1}^{C}(W_{ik}-\mu_{ik})(W_{ij}-\mu_{ij})$ for all $k,j=1,\ldots,C$, the posterior mean of $\Sigma$ under Algorithms [P1] and [P2] are approximately the same,

$$
\begin{aligned}
E(\Sigma|\alpha_1^2,W,\mu) &\approx \frac{\alpha_1^2\sum_{i=1}^{N}(W_i-\mu_i)(W_i-\mu_i)^T}{\alpha_1^2\sum_{i=1}^{N}\sum_{j=1}^{C}(W_{ij}-\mu_{ij})^2}\times C \\
&= \frac{\sum_{i=1}^{N}(W_i-\mu_i)(W_i-\mu_i)^T}{\sum_{i=1}^{N}\sum_{j=1}^{C}(W_{ij}-\mu_{ij})^2}\times C \\
&\approx E(\Sigma|W,\mu).
\end{aligned}
$$

In the same way, we can show that $\text{var}(\Sigma|\alpha_1^2,W,\mu) \approx \text{var}(\Sigma|W,\mu)$.

## C Outcome Distribution as a Function of Correlation

This section discusses how the outcome distribution is connected to the correlation of latent utilities for $C = 2$. We assume the outcome is determined as follows:

$$W = (W_1, W_2)^T \sim \textbf{MVN}\Bigg( \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix}, \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix} \Bigg),$$

$$S = \begin{cases} 1 & \text{if } W_1 \geq \max\{0, W_2\} \\ 2 & \text{if } W_2 \geq \max\{0, W_1\} \\ 3 & \text{if } W_1 < 0 \text{ and } W_2 < 0. \end{cases}$$

We start with the outcome probability at the reference level,

$$\begin{aligned}
&P(S = 3) = P(W_1 < 0, W_2 < 0) \\
&= \int_{-\infty}^{0} \int_{-\infty}^{0} \frac{1}{\sqrt{|2\pi\Sigma|}} \\
&\exp\Bigg\{ -\frac{1}{2}(w_1 - \mu_1, w_2 - \mu_2)\Sigma^{-1} \begin{pmatrix} w_1 - \mu_1 \\ w_2 - \mu_2 \end{pmatrix} \Bigg\} dw_1 dw_2 \\
&= \int_{-\infty}^{0} \int_{-\infty}^{0} \frac{1}{2\pi\sqrt{1-\rho^2}} \\
&\exp\Bigg\{ -\frac{1}{2}\frac{[w_1 - \mu_1 - \rho(w_2 - \mu_2)]^2}{1-\rho^2} - \frac{(w_2 - \mu_2)^2}{2} \Bigg\} dw_1 dw_2 \\
&= \int_{-\infty}^{-\mu_2} \int_{-\infty}^{0} \frac{1}{2\pi\sqrt{1-\rho^2}} \\
&\exp\Bigg\{ -\frac{1}{2}\frac{[w_1 - \mu_1 - \rho\widetilde{w}_2]^2}{1-\rho^2} \Bigg\} \exp\Bigg\{ -\frac{\widetilde{w}_2^2}{2} \Bigg\} dw_1 d\widetilde{w}_2 \\
&= \int_{-\infty}^{-\mu_2} \int_{-\infty}^{\frac{-\mu_1 - \rho\widetilde{w}_2}{\sqrt{1-\rho^2}}} \frac{1}{2\pi} \exp\Bigg\{ -\frac{\widetilde{w}_1^2}{2} \Bigg\} \exp\Bigg\{ -\frac{\widetilde{w}_2^2}{2} \Bigg\} d\widetilde{w}_1 d\widetilde{w}_2 \\
&= \int_{-\infty}^{-\mu_2} \frac{1}{2\pi} \tau\Bigg( \frac{-\mu_1 - \rho\widetilde{w}_2}{\sqrt{1-\rho^2}} \Bigg) \exp\Bigg\{ -\frac{\widetilde{w}_2^2}{2} \Bigg\} d\widetilde{w}_2,
\end{aligned}$$

where the second equality comes from the inversion and determinant lemma of matrices, and $\tau(u) = \int_{-\infty}^{u} \exp\{-\frac{s^2}{2}\} ds$ in the last equality.

Next, we write $P(S=3)$ as a function of $\rho$,

$f(\rho) = \int_{-\infty}^{-\mu_2} \frac{1}{2\pi}\tau(\frac{-\mu_1-\rho t}{\sqrt{1-\rho^2}})\exp\{-\frac{t^2}{2}\}dt$. The corresponding derivative w.r.t $\rho$ has the form,

$$
\begin{aligned}
&\frac{d}{d\rho}f(\rho) \\
=&\int_{-\infty}^{-\mu_2} \frac{1}{2\pi\sqrt{1-\rho^2}} \\
&\exp\Big\{-\frac{1}{2}\frac{(\mu_1+\rho t)^2}{1-\rho^2}\Big\}\exp\Big\{-\frac{t^2}{2}\Big\}\Big(-\frac{t+\rho\mu_1}{1-\rho^2}\Big)dt \\
=&\int_{-\infty}^{-\mu_2} \frac{1}{2\pi\sqrt{1-\rho^2}} \\
&\exp\Big\{-\frac{1}{2}\frac{(t+\rho\mu_1)^2+\mu_1^2(1-\rho^2)}{1-\rho^2}\Big\}\Big(-\frac{t+\rho\mu_1}{1-\rho^2}\Big)dt \\
=&\int_{-\infty}^{-\mu_2+\rho\mu_1} \frac{1}{2\pi\sqrt{1-\rho^2}} \\
&\exp\Big\{-\frac{1}{2}\frac{\widetilde{t}^2}{1-\rho^2}\Big\}\exp\Big\{-\frac{1}{2}\mu_1^2\Big\}\Big(-\frac{\widetilde{t}}{1-\rho^2}\Big)dt \\
=&\frac{1}{2\pi\sqrt{1-\rho^2}}\exp\Big\{-\frac{1}{2}\mu_1^2\Big\}\exp\Big\{-\frac{1}{2}\frac{(-\mu_2+\rho\mu_1)^2}{1-\rho^2}\Big\} \\
\Rightarrow \quad & \frac{d}{d\rho}f(\rho) > 0, \qquad \rho\in(-1,1).
\end{aligned}
$$

As a result, for every possible combination of $(\mu_1,\mu_2)$, $P(S=3)$ is always a strictly increasing function of $\rho$. Next, we show that how the non-reference-level outcome probabilities change w.r.t $\rho$ depends on $(\mu_1,\mu_2)$. For outcome level 1, we have

$$
\begin{aligned}
P(S=1) &= P(W_1 \geq W_2, W_1 \geq 0) = P(W_2 - W_1 \leq 0, -W_1 \leq 0) \\
&= P(Z_1 \leq 0, Z_2 \leq 0)
\end{aligned}
$$

with $\begin{pmatrix} Z_1 \\ Z_2 \end{pmatrix} \sim N\Big(\begin{pmatrix} \mu_2-\mu_1 \\ -\mu_1 \end{pmatrix}, \begin{pmatrix} 2(1-\rho) & 1-\rho \\ 1-\rho & 1 \end{pmatrix}\Big)$.

The outcome probability can be written as

$$
\begin{aligned}
&P(S=1) \\
&= \int_{-\infty}^{0}\int_{-\infty}^{0} \tfrac{1}{2\pi\sqrt{1-\rho^2}} \exp\left\{ -\tfrac{1}{2}\tfrac{[z_1-(\mu_2-\mu_1)-(1-\rho)(z_2+\mu_1)]^2}{1-\rho^2} \right\} \\
&\exp\left\{ -\frac{(z_2+\mu_1)^2}{2} \right\} dw_1 dw_2 \\
&= \int_{-\infty}^{\mu_1} \frac{1}{2\pi}\tau\left( \frac{-(\mu_2-\mu_1)-(1-\rho)t}{\sqrt{1-\rho^2}} \right) \exp\left\{ -\frac{t^2}{2} \right\} dt
\end{aligned}
$$

Similar to the procedure for $f(\rho)$, we write $P(S=1)$ as $g(\rho)$. The derivative w.r.t. $\rho$ is

$$
\begin{aligned}
&\frac{d}{d\rho} g(\rho) = \\
&-\tfrac{1}{2}\exp\left\{ -\tfrac{1}{1+\rho}\left[\tfrac{\mu_1+\mu_2}{2}\right]^2 \right\} - \tfrac{(\mu_2-\mu_1)\sqrt{1+\rho}}{2(1-\rho)}\tau\left( \tfrac{\mu_1+\mu_2}{2\sqrt{1+\rho}} \right) \\
&\in \left( -\tfrac{1}{2}\exp\left\{ -\tfrac{1}{1+\rho}\left[\tfrac{\mu_1+\mu_2}{2}\right]^2 \right\} \pm \tfrac{|\mu_1-\mu_2|}{1-\rho}\sqrt{\tfrac{\pi(1+\rho)}{2}} \right).
\end{aligned}
$$

The last interval comes from $\tau(u) \in (0, \sqrt{2\pi})$ by definition. In fact, from the way $S$ depends on $(W_1, W_2)$ for the non-reference levels, we can easily see that the derivative of $P(S=2)$ w.r.t $\rho$ falls into the same interval in the above derivation. Clearly, the center and width of the interval depend on $(\frac{\mu_1+\mu_2}{2}, \rho)$ and $(|\mu_1-\mu_2|, \rho)$, respectively. So how $P(S=1)$ and $P(S=2)$ vary with $\rho$ are heavily influenced by the position of and the distance between the latent variables.

The following plots display how the outcome distribution changes with $\rho$ under three different pairs, $(\mu_1, \mu_2)$. We can see that the reference level outcome probability $P(S=3)$ increases with $\rho$ in all three settings, and the relative positions of $P(S=1)$ and $P(S=2)$ are closely related to the values of and the difference between $\mu_1$ and $\mu_2$.

# D Algorithms' Pseudo Code

Using the notation in Section 2.1, we provide details on the algorithms in Section 2.4.

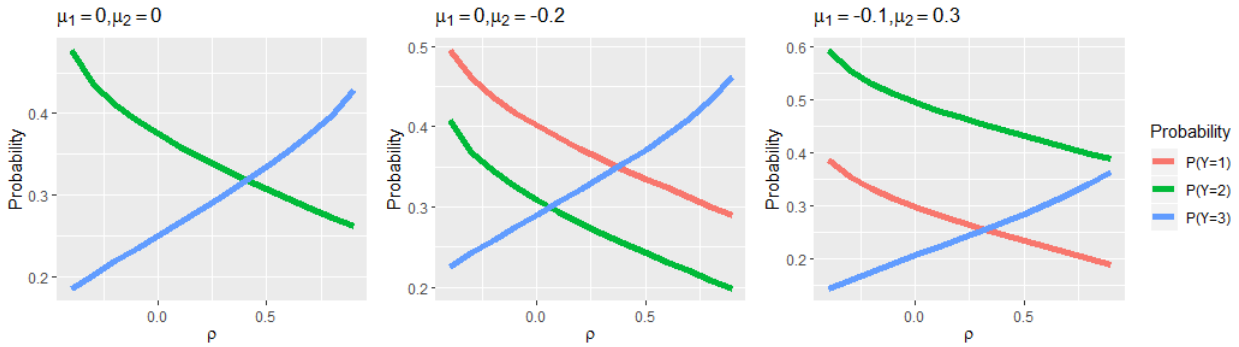



Figure 3: Outcome distribution as a function of $\rho$ under $(\mu_1, \mu_2)$ being (0,0), (0,-0.2), and (-0.1, 0.3).

**Algorithm [KD]**

**Step 0**: Initialize parameters $l = 0, \alpha^{(0)}, W^{(0)}, \theta^{(0)}, \Sigma^{(0)}$

**while** $l < L$ do **do**

**Step 1**: Update $(\widetilde{W}^{(l+1)}, (\alpha_1^{(l+1)})^2)$ via $P(\widetilde{W}, \alpha^2 | \mu^{(l)}, \Sigma^{(l)}, S)$

(a) Draw $(\alpha_1^{(l+1)})^2 \sim \text{trace}[\Psi(\Sigma^{(l)})^{-1}]/\chi^2_{\nu C}$;

(b) Draw $W^{(l+1)} = (W_1^{(l+1)}, \ldots, W_C^{(l+1)}) \sim MVN(\mu^{(l)}, \Sigma^{(l)})$ by

**for** $i \in 1, \ldots, N$ **do**

**for** $j \in 1, \ldots, C$ **do**

$W_{ij}^{(l+1)} | W_{i(-j)}^{(l+1)} \sim TN(m_{ij}^{(l)}, (\tau_j^{(l)})^2)$ ▷ Appendix D.1

where $W_{i(-j)}^{(l+1)} = (W_{i1}^{(l+1)}, \ldots, W_{i,j-1}^{(l+1)}, W_{i,j+1}^{(l)}, \ldots, W_{i,C}^{(l)})$

**end for**

**end for**;

(c) Set $\widetilde{W}^{(l+1)} = \alpha_1^{(l+1)} W^{(l+1)}$.

**Step 2**: Update $\widetilde{\theta}^{(l+1)}$ via $P(\widetilde{\theta} | \widetilde{W}^{(l+1)}, \alpha_1^{(l+1)}, \Sigma^{(l)})$

(a) Gibbs sampling of binary trees:

**for** $b \in 1, \ldots, m$ **do**

**for** $j \in 1, \ldots, C$ **do**

Update $\widetilde{W}_{jb}^{\dagger}$ and draw $\widetilde{\theta}_{jb}^{(l+1)} \sim P(\widetilde{\theta}_{jb} | \widetilde{W}_{jb}^{\dagger}, (\alpha_1^{(l+1)} \tau_j^{(l)})^2)$ ▷ Appendix D.2

**end for**

**end for**;

(b) Set $\widetilde{\mu}_{ij}^{(l+1)} = G_j(X_i; \widetilde{\theta}_j^{(l+1)})$ and $\mu_{ij}^{(l+1)} = \widetilde{\mu}_{ij}^{(l+1)} / \alpha_1^{(l+1)}$.

**Step 3**: Update $(\Sigma^{(l+1)}, (\alpha_3^{(l+1)})^2)$ via $P(\Sigma, \alpha^2 | \widetilde{W}^{(l+1)}, \widetilde{\theta}^{(l+1)})$

(a) Draw $\widetilde{\Sigma} \sim \text{Inv-Wishart}(n + \nu, \Psi + \sum_{i=1}^n \widetilde{\epsilon}_i \widetilde{\epsilon}_i^T)$,

where $\widetilde{\epsilon}_i = (\widetilde{\epsilon}_{i1}, \ldots, \widetilde{\epsilon}_{i,C})$ and $\widetilde{\epsilon}_{ij} = \widetilde{W}_{ij}^{(l+1)} - \widetilde{\mu}_{ij}^{(l+1)}$ for $j = 1, \ldots, C$;

(b) Setting $(\alpha_3^{(l+1)})^2 = \text{trace}(\widetilde{\Sigma}/C)$;

(c) Re-scaling model parameters based on $\alpha_3^{(l+1)}$:

$\Sigma^{(l+1)} = \widetilde{\Sigma}/(\alpha_3^{(l+1)})^2$ and $W^{(l+1)} = \mu^{(l+1)} + \widetilde{\epsilon}/\alpha_3^{(l+1)}$;

**end while**

**Step 4**: Prediction given new input ▷ Appendix D.3

**Algorithm [P1]**

**Step 0**: Initialize parameters $l = 0, \alpha^{(0)}, W^{(0)}, \theta^{(0)}, \Sigma^{(0)}$

**while** $l < L$ do **do**

**Step 1**: Update $(\widetilde{W}^{(l+1)}, (\alpha_1^{(l+1)})^2)$ via $P(\widetilde{W}, \alpha^2 | \mu^{(l)}, \Sigma^{(l)}, S)$

(a) Draw $(\alpha_1^{(l+1)})^2 \sim \text{trace}[\Psi(\Sigma^{(l)})^{-1}]/\chi^2_{\nu C}$;

(b) Draw $W^{(l+1)} = (W_1^{(l+1)}, \ldots, W_C^{(l+1)}) \sim MVN(\mu^{(l)}, \Sigma^{(l)})$ by

**for** $i \in 1, \ldots, N$ **do**

**for** $j \in 1, \ldots, C$ **do**

$W_{ij}^{(l+1)} | W_{i(-j)}^{(l+1)} \sim TN(m_{ij}^{(l)}, (\tau_j^{(l)})^2)$ ▷ Appendix D.1

where $W_{i(-j)}^{(l+1)} = (W_{i1}^{(l+1)}, \ldots, W_{i,j-1}^{(l+1)}, W_{i,j+1}^{(l)}, \ldots, W_{i,C}^{(l)})$

**end for**

**end for**;

(c) Set $\widetilde{W}^{(l+1)} = \alpha_1^{(l+1)} W^{(l+1)}$.

**Step 2**: Update $\theta^{(l+1)}$ via $P(\theta | W^{(l+1)}, \Sigma^{(l)})$

(a) Gibbs sampling of binary trees:

**for** $b \in 1, \ldots, m$ **do**

**for** $j \in 1, \ldots, C$ **do**

Update $W_{jb}^{\dagger}$ and draw $\theta_{jb}^{(l+1)} \sim P(\theta_{jb} | W_{jb}^{\dagger}, (\tau_j^{(l)})^2)$ ▷ Appendix D.2

**end for**

**end for**;

(b) Set $\mu_{ij}^{(l+1)} = G_j(X_i; \theta_j^{(l+1)})$.

**Step 3**: Update $(\Sigma^{(l+1)}, (\alpha_3^{(l+1)})^2)$ via $P(\Sigma, \alpha^2 | \widetilde{W}^{(l+1)}, \alpha_1^{(l+1)}, \theta^{(l+1)})$

(a) Draw $\widetilde{\Sigma} \sim$ Inv-Wishart$(n + \nu, \Psi + \sum_{i=1}^n \widetilde{\epsilon}_i \widetilde{\epsilon}_i^T)$,

where $\widetilde{\epsilon}_i = (\widetilde{\epsilon}_{i1}, \ldots, \widetilde{\epsilon}_{i,C})$ and $\widetilde{\epsilon}_{ij} = \widetilde{W}_{ij}^{(l+1)} - \alpha_1^{(l+1)} \mu_{ij}^{(l+1)}$ for $j = 1, \ldots, C$;

(b) Setting $(\alpha_3^{(l+1)})^2 = \text{trace}(\widetilde{\Sigma}/C)$;

(c) Re-scaling model parameters based on $\alpha_3^{(l+1)}$:

$\Sigma^{(l+1)} = \widetilde{\Sigma}/(\alpha_3^{(l+1)})^2$ and $W^{(l+1)} = \mu^{(l+1)} + \widetilde{\epsilon}/\alpha_3^{(l+1)}$.

**end while**

**Step 4**: Prediction given new input ▷ Appendix D.3

**Algorithm [P2]**

**Step 0**: Initialize parameters $l = 0, \alpha^{(0)}, W^{(0)}, \theta^{(0)}, \Sigma^{(0)}$
**while** $l < L$ do **do**
  **Step 1**: Update $W^{(l+1)}$ via $P(W|\mu^{(l)}, \Sigma^{(l)}, S)$
    (a) Draw $W^{(l+1)} = (W_1^{(l+1)}, \ldots, W_C^{(l+1)}) \sim MVN(\mu^{(l)}, \Sigma^{(l)})$ by
      **for** $i \in 1, \ldots, N$ **do**
        **for** $j \in 1, \ldots, C$ **do**
          $W_{ij}^{(l+1)}|W_{i(-j)}^{(l+1)} \sim TN(m_{ij}^{(l)}, (\tau_j^{(l)})^2)$ ▷ Appendix D.1
          where $W_{i(-j)}^{(l+1)} = (W_{i1}^{(l+1)}, \ldots, W_{i,j-1}^{(l+1)}, W_{i,j+1}^{(l)}, \ldots, W_{i,C}^{(l)})$
        **end for**
      **end for**.
  **Step 2**: Update $\theta^{(l+1)}$ via $P(\theta|W^{(l+1)}, \Sigma^{(l)})$
    (a) Gibbs sampling of binary trees:
      **for** $b \in 1, \ldots, m$ **do**
        **for** $j \in 1, \ldots, C$ **do**
          Update $W_{jb}^{\dagger}$ and draw $\theta_{jb}^{(l+1)} \sim P(\theta_{jb}|W_{jb}^{\dagger}, (\tau_j^{(l)})^2)$ ▷ Appendix D.2
        **end for**
      **end for**;
    (b) Set $\mu_{ij}^{(l+1)} = G_j(X_i; \theta_j^{(l+1)})$.
  **Step 3**: Update $(\Sigma^{(l+1)}, (\alpha_3^{(l+1)})^2)$ via $P(\Sigma, \alpha^2|W^{(l+1)}, \theta^{(l+1)})$
    (a) Draw $\widetilde{\Sigma} \sim$ Inv-Wishart$(n + \nu, \Psi + \sum_{i=1}^{n} \epsilon_i \epsilon_i^T)$,
      where $\epsilon_i = (\epsilon_{i1}, \ldots, \epsilon_{iC})$ and $\epsilon_{ij} = W_{ij}^{(l+1)} - \mu_{ij}^{(l+1)}$ for $j = 1, \ldots, C$;
    (b) Setting $(\alpha_3^{(l+1)})^2 =$ trace$(\widetilde{\Sigma}/C)$;
    (c) Re-scaling model parameters based on $\alpha_3^{(l+1)}$:
      $\Sigma^{(l+1)} = \widetilde{\Sigma}/(\alpha_3^{(l+1)})^2$ and $W^{(l+1)} = \mu^{(l+1)} + \widetilde{\epsilon}/\alpha_3^{(l+1)}$.
**end while**
**Step 4**: Prediction given new input ▷ Appendix D.3

### D.1 Gibbs Sampling of the Latent Utilities

Gibbs sampling of the latent utilities from univariate truncated normal distributions is described in the Section 3 of[17]. In the pseudo-code,

$$m_{ij} = \mu_{ij} + \Sigma_{j(-j)}(\Sigma_{(-j)(-j)})^{-1}[W_{i(-j)} - \mu_{i(-j)})]$$

$$\tau_j^2 = \Sigma_{jj} - \Sigma_{j(-j)}(\Sigma_{(-j)(-j)})^{-1}\Sigma_{(-j)j}$$

for the $i = 1, \ldots, N$, $j = 1, \ldots, C$, where

$\mu_{ij} = \sum_{d=1}^{m} g(X_i; \theta_{jd})$, $\Sigma_{jj}$ is the element at the $j$th row and $j$th column of $\Sigma$, $\Sigma_{(-j)(-j)}$ is the remaining of $\Sigma$ excluding its $j$th row and $j$th column, $\Sigma_{j(-j)}$ is the $j$th row of $\Sigma$ excluding its $j$th

element $\Sigma_{jj}$, and $\Sigma_{(-j)j}$ is similarly derived.

### D.2 Tree Sampling in MPBART

Using Algorithms [P1] and [P2] as an example,we follow Section 3.2 of[9] and provide the details on the conditional distributions used to update each individual tree. For simplicity, we exclude the subscript $i$. Given $W \sim MVN(\mu, \Sigma)$, we have $W_j|(W_{(-j)}, \mu, \Sigma) \sim N(m_j, \tau_j^2)$ where $m_j$ and $\tau_j^2$ are defined in Appendix D.1. Based on the fact that $\mu_j = \sum_{d=1}^{m} g(X; \theta_{jd})$, define

$$\begin{aligned} W_{jb}^{\dagger} =& W_j - \sum_{d=1}^{b-1} g(X; \theta_{jd}) - \sum_{d'=b+1}^{m} g(X; \theta_{jd'}) \\ & - \Sigma_{j(-j)}(\Sigma_{(-j)(-j)})^{-1}[W_{(-j)} - \mu_{(-j)})]. \end{aligned}$$

Conditional on $(W_{(-j)}, \mu, \Sigma)$,

$$\begin{aligned} & W_{jb}^{\dagger} - g(X; \theta_{jb}) = W_j - m_j \sim MVN(0, \tau_j^2) \\ \Rightarrow \quad & W_{jb}^{\dagger} \sim N(g(X; \theta_{jb}), \tau_j^2). \end{aligned}$$

Consequently, the $b$th binary tree of the $j$th latent variable, $\theta_{jb}^{(l+1)}$, is updated to estimate the mean of

$$\begin{aligned} W_{jb}^{\dagger} =& W_j^{(l+1)} - \sum_{d=1}^{b-1} g(X; \theta_{jd}^{(l+1)}) - \sum_{d'=b+1}^{m} g(X; \theta_{jd'}^{(l)}) \\ & - \Sigma_{j(-j)}^{(l)}(\Sigma_{(-j)(-j)}^{(l)})^{-1}[W_{(-j)}^{(l+1)} - \mu_{(-j)}^{(l+1)}] \end{aligned}$$

where $\mu_{(-j)}^{(l+1)} = \{G_k(X;\theta_k^{(l+1)}); k \neq j\}$ and $G_k(X;\theta_k^{(l+1)}) = \sum_{d=1}^{b-1} g(X;\theta_{kd}^{(l+1)}) + \sum_{d'=b}^{m} g(X;\theta_{kd'}^{(l)})$. Similarly, in Algorithm [KD],

$$\widetilde{W}_{jb}^{\dagger} = \widetilde{W}_j^{(l+1)} - \sum_{d=1}^{b-1} g(X\widetilde{\theta}_{jd}^{(l+1)}) - \sum_{d'=b+1}^{m} g(X;\widetilde{\theta}_{jd'}^{(l)}) - \Sigma_{j(-j)}^{(l)}(\Sigma_{(-j)(-j)}^{(l)})^{-1}[\widetilde{W}_{(-j)}^{(l+1)} - \widetilde{\mu}_{(-j)}^{(l+1)}]$$

where $\widetilde{\mu}_{(-j)}^{(l+1)} = \{G_k(X;\widetilde{\theta}_k^{(l+1)}); k \neq j\}$ and $G_k(X;\widetilde{\theta}_k^{(l+1)}) = \sum_{d=1}^{b-1} g(X;\widetilde{\theta}_{kd}^{(l+1)}) + \sum_{d'=b}^{m} g(X;\widetilde{\theta}_{kd'}^{(l)})$.

### D.3 Predictions from MPBART

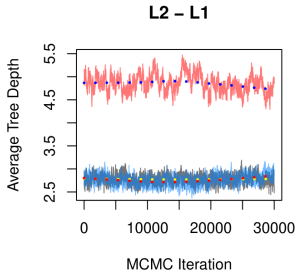
Given fitted model parameters from the $L$th round of posterior sampling, $(\theta^{(L)}, \Sigma^{(L)})$, we obtain outcome prediction for a new input $x$ as follows:

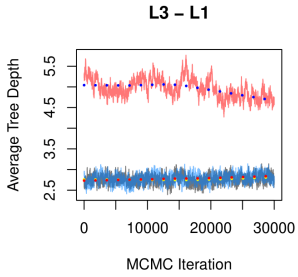
$$S(x) = \begin{cases} \text{reference level } 0 & \text{if } \max\{W_1(x), \ldots, W_C(x)\} < 0 \\ j & \text{if } \max\{0, W_1(x), \ldots, W_C(x)\} = W_j(x) \end{cases}$$

by drawing

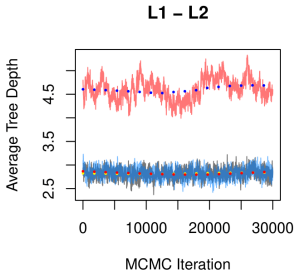
$$W(x) \sim MVN(G(x;\theta^{(L)}), \Sigma^{(L)}).$$

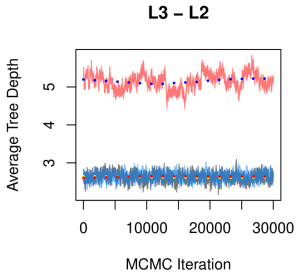
# E Convergence Plots



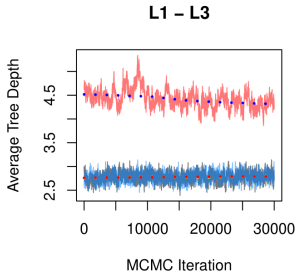



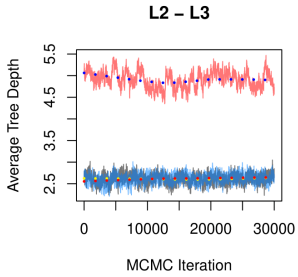

(a) $\nu = C + 1$, $\Psi_{12} = 0$, reference level 1



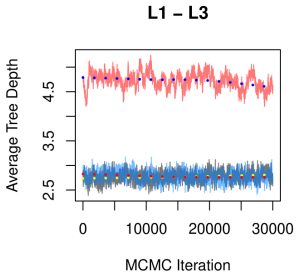



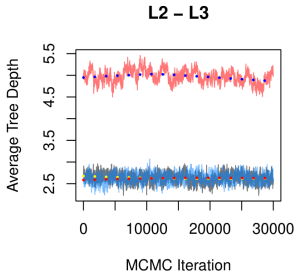

(b) $\nu = C + 1$, $\Psi_{12} = 0$, reference level 2



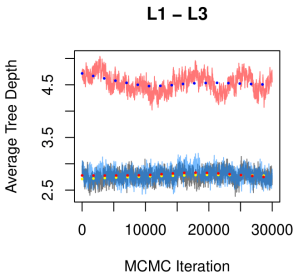



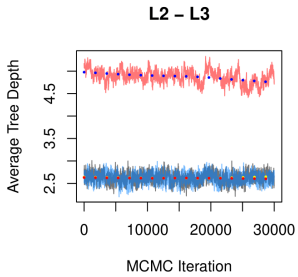

(c) $\nu = C + 1$, $\Psi_{12} = 0$, reference level 3



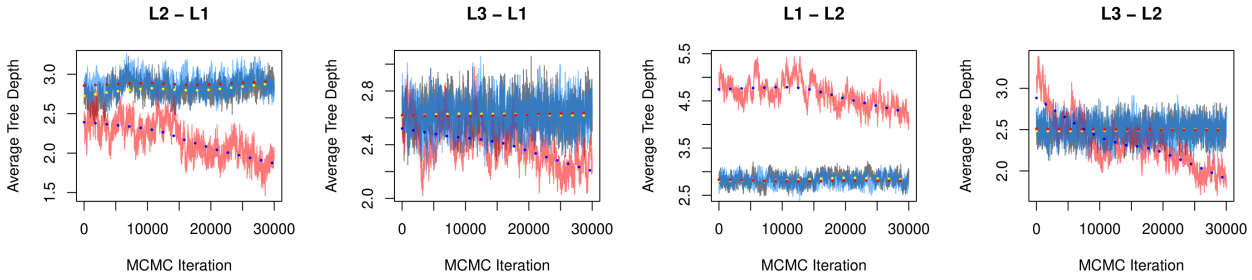

(d) $\nu = C + 3$, $\Psi_{12} = -0.5$, reference level 3



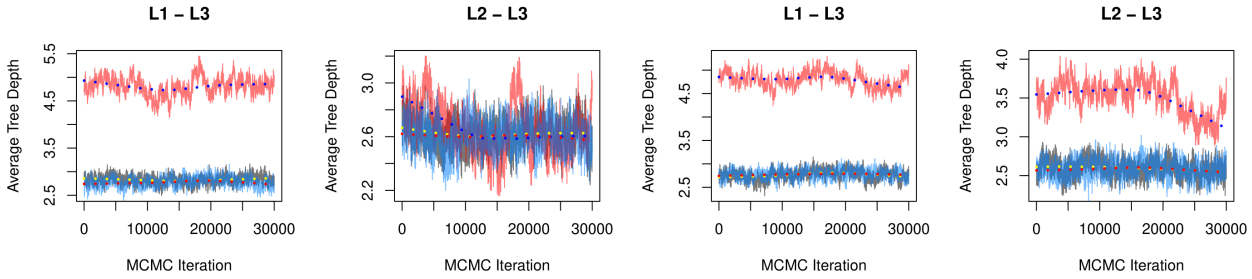

(e) $\nu = C + 3$, $\Psi_{12} = 0.5$, reference level 3

Figure 4: Plot of posterior average tree depth for each latent utility as time series, under simulation Setting 1 and hyperparameters described in plot labels. Red, black, and blue correspond to Algorithms [KD], [P1], and [P2], respectively.



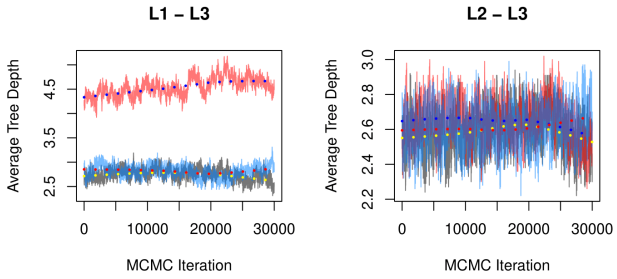

(a) $\nu = C + 1$, $\Psi_{12} = 0$, reference level 1

(b) $\nu = C + 1$, $\Psi_{12} = 0$, reference level 2



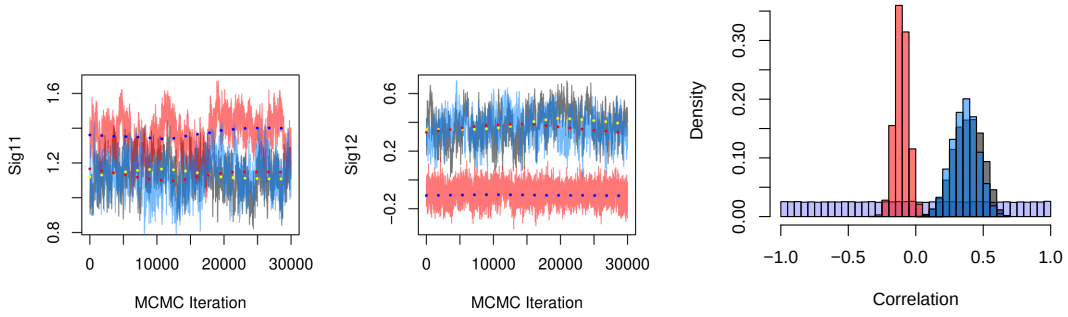

(c) $\nu = C + 1$, $\Psi_{12} = 0$, reference level 3

(d) $\nu = C + 3$, $\Psi_{12} = -0.5$, reference level 3

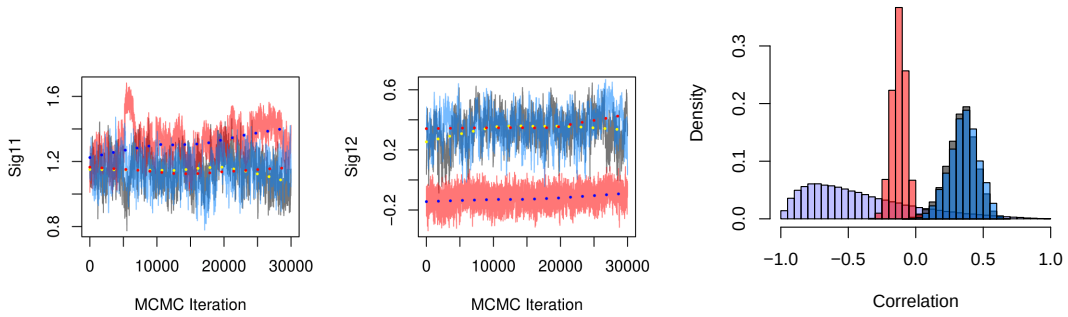



(e) $\nu = C + 3$, $\Psi_{12} = 0.5$, reference level 3

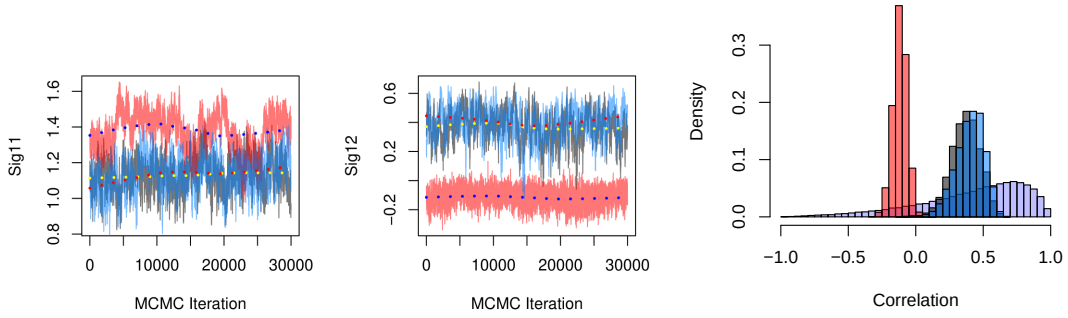
Figure 5: Plot of posterior average tree depth for each latent utility as time series, under simulation Setting 2 and hyperparameters described in plot labels. Red, black, and blue correspond to Algorithms [KD], [P1], and [P2], respectively.




(a) $\nu = C + 1$, $\Psi_{12} = 0$

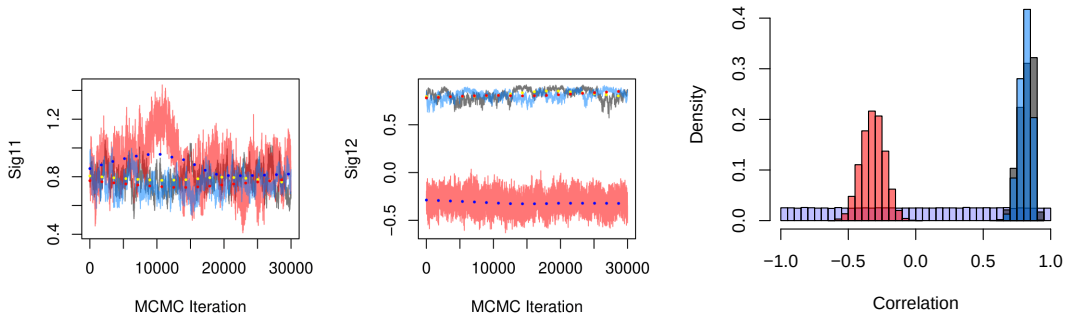



(b) $\nu = C + 3$, $\Psi_{12} = -0.5$

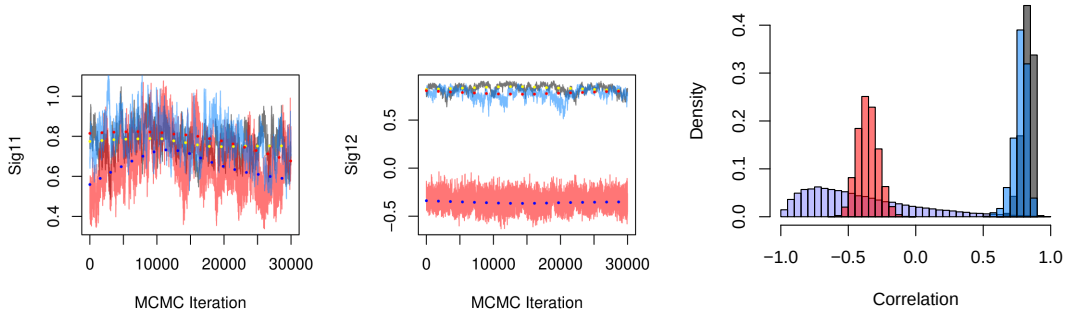



(c) $\nu = C + 3$, $\Psi_{12} = 0.5$

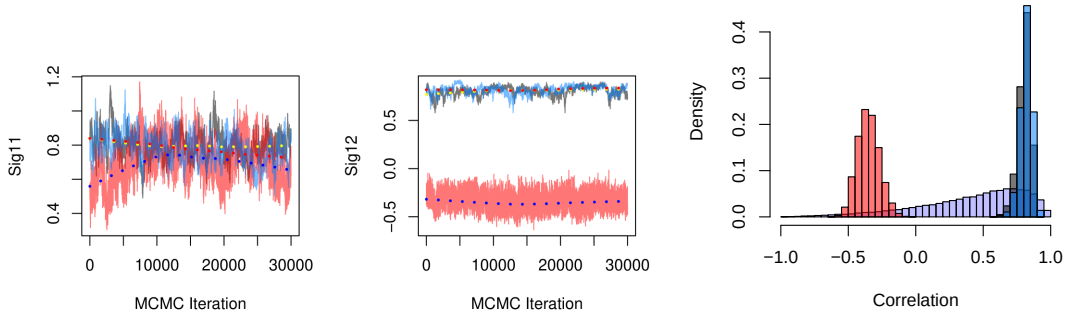
Figure 6: Plot of posterior $\sigma_{11}$ and $\sigma_{12}$ as time series on the left, and histogram of the $\sigma_{12}$ under its prior (purple), posterior from Algorithms [KD] (red), [P1] (black), and [P2] (blue); same color specification applies to the left plot. Posterior inference is conducted under Setting 1, reference level 3, and hyperparameters described in plot labels.




(a) $\nu = C + 1$, $\Psi_{12} = 0$




(b) $\nu = C + 3$, $\Psi_{12} = -0.5$




(c) $\nu = C + 3$, $\Psi_{12} = 0.5$

Figure 7: Plot of posterior $\sigma_{11}$ and $\sigma_{12}$ as time series on the left, and histogram of the $\sigma_{12}$ under its prior (blue), posterior from Algorithms [KD] (red), [P1] (black), and [P2] (blue). Posterior inference is conducted under simulation Setting 2, reference level 3, and hyperparameters described in plot labels. Red, black, and blue correspond to Algorithms [KD], [P1], and [P2], respectively.