Distributed Bayesian Inference in Linear Mixed-Effects Models

Sanvesh Srivastava [*1] and Yixiang Xu [†2]

[1]*Department of Statistics and Actuarial Science, The University of Iowa, Iowa City, Iowa, U.S.A.*

[2]*Marketing Group, Haas School of Business, University of California, Berkeley, California, U.S.A.*

December 17, 2020

Abstract

Linear mixed-effects models play a fundamental role in statistical methodology. A variety of Markov chain Monte Carlo (MCMC) algorithms exist for fitting these models, but they are inefficient in massive data settings because every iteration of any such MCMC algorithm passes through the full data. Many divide-and-conquer methods have been proposed to solve this problem, but they lack theoretical guarantees, impose restrictive assumptions, or have complex computational algorithms. Our focus is one such method called the Wasserstein Posterior (WASP), which has become popular due to its optimal theoretical properties under general assumptions. Unfortunately, practical implementation of the WASP either requires solving a complex linear program or is limited to one-dimensional parameters. The former method is inefficient and the latter method fails to capture the joint posterior dependence structure of multivariate parameters. We develop a new algorithm for computing the WASP of multivariate parameters that is easy to implement and is useful for computing the WASP in any model where the posterior distribution of parameter belongs to a location-scatter family of probability measures. The algorithm is introduced for linear mixed-effects models with both implementation details and theoretical properties. Our algorithm outperforms the current state-of-the-art method in inference on the functions of the covariance matrix of the random effects across diverse numerical comparisons.

[*] sanvesh-srivastava@uiowa.edu
[†] yixiang-xu@berkeley.edu

Key words: Data augmentation; Divide-and-Conquer; Geometric mean; Location-Scatter Family; Parameter Expansion; Monte Carlo Error; Wasserstein Distance; Wasserstein Barycenter.

1 Introduction

Fitting Bayesian linear mixed-effects models to massive data is challenging because posterior computations scale poorly with the sample size. MCMC algorithms are most commonly used for fitting such models, but existing algorithms require multiple passes through the full data in every iteration; therefore, their application in massive data settings is impractically slow. This limitation has motivated a rich literature on divide-and-conquer algorithms that scale sampling algorithms by dividing the full data into smaller subsets, running sampling algorithms in parallel on the subsets, and estimating a posterior distribution by combining MCMC draws from all the subsets. The posterior distribution estimated in the last step replaces the full data posterior distribution for inference. A major problem with these approaches is that they either rely on restrictive assumptions on the posterior distributions of parameters or have a complex combination algorithm. We develop a new combination algorithm for divide-and-conquer posterior computations in linear mixed-effects modeling that is computationally simple and has asymptotic Monte Carlo and statistical guarantees.

Linear mixed-effects models are widely used in extending linear regression to account for hierarchical dependence in the data and are very popular in Bayesian modeling (Gelman and Hill, 2007). There are numerous applications of mixed-effects models in the modern setting, including e-commerce (von Brzeski et al., 2015; Gao and Owen, 2017), recommendation systems (Perry, 2017), and semiparametric regression (Wand, 2017). Software exist to automate the fitting of Bayesian mixed-effects models (Stan Development Team, 2017), but they cannot be used in massive data applications due to memory and computational bottlenecks. We are motivated to scale existing Bayesian methodology and software to arbitrarily large data sets using the divide-and-conquer technique that allows us to bypass any restrictive asymptotic approximations.

Scalable Bayesian inference in massive data is an active area of research, but three main groups of methods stand out. The first group relies on analytic approximations such that the posterior distribution of parameters is estimated via optimization, including expectation propagation, variational Bayes, and Laplace approximation (Rue et al., 2009; Gelman et al., 2014; Tan and Nott, 2014; Kucukelbir et al., 2015; Lee and Wand, 2016; Ranganath et al., 2016). All these methods are very general and are easily applied for linear mixed-effects modeling; however, analytic approximations can be highly biased in the estimation of posterior dependence structure and most of these methods have no theoretical guarantees on posterior uncertainty quantification (Giordano et al., 2017). The second group relies on new MCMC or sequential Monte Carlo (SMC) algo-

rithms based on subsampling or approximate transition kernels that avoid passing through the full data (Welling and Teh, 2011; Ahn et al., 2012; Korattikara et al., 2014; Lan et al., 2014; Shahbaba et al., 2014; Maclaurin and Adams, 2015; Bardenet et al., 2017; Johndrow et al., 2015; Alquier et al., 2016; Campbell and Broderick, 2018; Quiroz et al., 2018). While these algorithms are very simple, none of them focus on posterior inference in linear mixed-effects models, and the current examples are limited to regression, classification, and simple time series models.

The third group consists of methods based on the divide-and-conquer technique and has two main advantages. First, any existing sampling algorithm requires a simple modification before it runs on the subsets. Second, if the number of subsets is chosen large enough, then any sampling algorithm can be used in massive data settings. This means that theoretical guarantees for the sampling algorithm and the associated software are also easily extended to the new setting. A variety of methods exist in this group, but they mainly differ in their third step that combines MCMC draws from the subsets (Neiswanger et al., 2014; Wang and Dunson, 2013; Minsker et al., 2014; Wang et al., 2015; Scott et al., 2016; Li et al., 2017; Minsker et al., 2017; Staib et al., 2017; Savitsky and Srivastava, 2018; Xue and Liang, 2019). One such method is the WASP that combines posterior distributions estimated across subsets through their Wasserstein barycenter, a notion of geometric center for probability measures (Agueh and Carlier, 2011; Srivastava et al., 2015, 2018). WASP is broadly applicable and has asymptotic statistical guarantees, but its combination algorithm requires solving a linear program, which becomes computationally demanding as the number of subsets increase. Resolving this limitation in linear mixed-effects modeling, we propose a new combination algorithm that only requires centering and scaling of subset MCMC draws.

The WASP has motivated divide-and-conquer methods with simpler combination algorithms. The Posterior Interval Estimation (PIE) algorithm is the first such method that computes quantiles of the WASP by averaging the quantiles estimated from the subset MCMC draws (Li et al., 2017). Compared to the WASP, PIE algorithm has stronger theoretical guarantees and a conceptually appealing combination algorithm, but its major restriction is that it is valid only for one-dimensional parameters. A more recent method in this thread is Double-Parallel Monte Carlo (DPMC), which is valid for multivariate parameters (Xue and Liang, 2019). Relying on the asymptotic normality of posterior distributions on the subsets, DPMC approximates the full data posterior by a mixture of appropriately centered subset MCMC draws. The asymptotic normality assumption in DPMC is difficult to justify in practice because it implies that covariance matrices of the subset posterior distributions are the same. Our experiments present cases where this assumption fails for inference on the covariance matrix of random effects.

Focusing on scalable linear mixed-effects modeling, the proposed combination algorithm for the WASP retains the computational simplicity of PIE and DPMC and has asymptotic statistical guarantees. We exploit the fact that the WASP is analytically tractable if the subset posterior distri-

butions belong to the same location-scatter family of probability measures (Álvarez-Esteban et al., 2016). Under additional theoretical assumptions, we show that the full data posterior distribution and the posterior distribution estimated by our combination algorithm have the same asymptotic means and covariance matrices. The location-scatter family includes non-Gaussian posterior distributions, especially the elliptical families, so our combination algorithm is applicable even if asymptotic normality of subset posterior distributions does not hold and reduces to the DPMC algorithm if this assumption is justified. Due to the importance of location-scatter family of probability distributions in our combination algorithm, we call our method the Location-Scatter WASP (LS-WASP).

LS-WASP has asymptotic Monte Carlo guarantees. In MCMC applications, intractable posterior expectations are approximated using Monte Carlo averages. The approximation error is called the Monte Carlo error, and it decays as $O(T^{-1/2})$ under some assumptions, where T is the number of MCMC iterations (Robert and Casella, 2004). The literature on quantifying the Monte Carlo error using a divide-and-conquer method is sparsely populated. DPMC's theoretical guarantees ignore Monte Carlo error and focus only on the statistical error. The PIE algorithm obtains an empirical approximation of the WASP through quantile estimates from subset MCMC draws and quantifies the approximation error; however, this error differs from the Monte Carlo error. The LS-WASP is applicable to MCMC draws from any sampling algorithm, but we quantify the Monte Carlo error in LS-WASP based on the conditional data augmentation algorithm of van Dyk (2000). Our choice is driven by the fact that developing theoretical Monte Carlo guarantees in Bayesian linear mixed-effects modeling is an active area of research, which is outside the scope of this work; see Román and Hobert (2015). If we assume that the error variance is known, then we are able to extend known results for van Dyk's algorithm to our setup and show that the Monte Carlo and statistical errors are of the same order if T is large, an assumption that is typically true in practice.

There is an increasing interest in using ideas from optimal transport to solve machine learning and statistical problems; see, for example, Arjovsky et al. (2017); Claici and Solomon (2018); Li and Zhang (2018). Wasserstein distance has been also used for developing new MCMC algorithms (Bernton et al., 2017). We emphasize that LS-WASP is not a new sampling algorithm and does not compete with any of these approaches. Our goal is to contribute to this growing literature and to show that we can obtain draws from the WASP if all the subset posterior distributions are completely specified by their mean vectors and covariance matrices up to the same but unknown probability measure with zero mean and identity covariance matrix.

2 Background on data augmentation in linear mixed-effects modeling

Our linear mixed-effects modeling setup is based on the classical data augmentation literature (van Dyk, 2000). Let n be the number of subjects, s be the total number of observations from n subjects (hereafter, s is called the sample size), s_i be the total number of observations specific to subject i ($i = 1, \ldots, n$), $s = \sum_{i=1}^{n} s_i$, and p, q be the number of fixed and random effects, respectively. If $y_i \in \mathbb{R}^{s_i}$ is the ith response, then the linear mixed-effects model assumes that

$$y_i = X_i\beta + Z_i c_i + e_i, \quad c_i \sim N_q(0, D), \quad e_i \sim N_{s_i}(0, \sigma^2 R_i), \quad i = 1, \ldots, n, \tag{1}$$

where R_i is a known s_i-by-s_i symmetric positive definite matrix, $X_i \in \mathbb{R}^{s_i \times p}$ and $Z_i \in \mathbb{R}^{s_i \times q}$ are known matrices of fixed and random effects covariates, and $c_i \in \mathbb{R}^q$ and $e_i \in \mathbb{R}^{s_i}$ are unknown vectors of random effects and idiosyncratic errors for subject i, respectively, $\beta \in \mathbb{R}^p$ is the fixed effects parameter vector, $N_a(m, V)$ represents an a-variate Gaussian distribution with mean vector m and covariance matrix V, and 0, I represent a zero vector and an identity matrix of appropriate dimensions. The q-by-q covariance matrix D and variance σ^2 are unknown. If c_i is missing data, then the observed data are $\mathcal{D}_{\text{obs}} = \{(y_i, X_i, Z_i, R_i), i = 1, \ldots, n\}$, augmented data are $\mathcal{D}_{\text{aug}} = \{(y_i, X_i, Z_i, c_i, R_i), i = 1, \ldots, n\}$, and the parameter vector is $\theta = \{\beta, \text{vech}(D), \sigma^2\}$, where $\text{vech}(D)$ stacks the lower triangular portion of D column-wise in a vector.

Efficient parameter estimation and inference in (1) using EM-type and data augmentation algorithms has been extensively studied. Our focus is on the conditional augmentation scheme in which Lb_i replaces c_i, where L is a lower triangular (Cholesky) matrix with positive diagonal elements such that $D = LL^T$ and $b_i \sim N_q(0, I)$ (Meng and Van Dyk, 1999; Van Dyk and Meng, 2001). This scheme modifies (1) to

$$y_i = X_i\beta + Z_i L b_i + e_i, \quad b_i \sim N_q(0, I), \quad e_i \sim N_{s_i}(0, \sigma^2 R_i), \quad i = 1, \ldots, n, \tag{2}$$

the augmented data to $\mathcal{D}_{\text{aug}} = \{(y_i, X_i, Z_i, b_i, R_i), i = 1, \ldots, n\}$, and the parameter vector to $\theta = \{\beta, \text{vech}(L), \sigma^2\}$. The parameter is assigned a prior with density $\pi(\theta) = \pi(\beta)\pi\{\text{vech}(L)\}\pi(\sigma^2)$, where

$$\pi(\beta) = N_p(\mu_\beta, \Sigma_\beta), \quad \pi\{\text{vech}(L)\} = N_{q(q+1)/2}(\mu_L, \Sigma_L), \quad \pi(\sigma^2) = \text{Inverse-Gamma}(a, b), \tag{3}$$

where μ_β, μ_L are mean vectors, Σ_β, Σ_L are covariance matrices, Inverse-Gamma distribution has mean $b/(a-1)$ and variance $b^2/\{(a-1)^2(a-2)\}$ for $a > 2$, $b > 0$. We have developed the imputation (I) and prediction (P) steps of the conditional augmentation algorithm based on the E

and M steps of ECME$_1$ algorithm in van Dyk (2000); see the supplementary material for details.

In massive data applications, conditional augmentation is impractically slow. Obtaining draws of β, vech(L), and σ^2 requires passing through data for all the n subjects in every iteration. This poses a major barrier to the use of conditional augmentation in applications with large number of subsets and offsets the benefits of easy implementation and theoretically guaranteed convergence to the stationary distribution. Our divide-and-conquer approach described next avoids passing through the full data in every iteration by conditioning on smaller subsets of the full data while retaining all the advantages of the conditional augmentation algorithm.

3 Distributed Bayesian inference in linear mixed-effects modeling

3.1 First step: partitioning of samples

The first step in distributed Bayesian inference divides the n subjects into k subsets. We use indices i and j for subjects and subsets, respectively. Define m_j to be the number of subjects on subset j, $\tilde{s}_j$ to be the total number of samples in subset j, and $\tilde{s}_{ji}$ to be the total number of samples on subject i in subset j so that $\tilde{s}_j = \sum_{i=1}^{m_j} \tilde{s}_{ji}$. Let $y_{ji}, X_{ji}, Z_{ji}, R_{ji}, b_{ji}, e_{ji}$ be the equivalents of $y_i, X_i, Z_i, R_i, b_i, e_i$ for subject i in subset j ($i = 1, \ldots, m_j$; $j = 1, \ldots, k$), and $\mathcal{D}_{j\,\text{obs}}$, $\mathcal{D}_{j\,\text{aug}}$ be the observed and augmented data on subset j, respectively. We adopt two partitioning schemes: with (default) and without overlap of subjects in the subsets. In the partitions with overlap, the same subject can belong to multiple subsets, so $\sum_{j=1}^{k} m_j \geq n$ and $\sum_{j=1}^{k} \tilde{s}_j \geq s$, whereas a subject belongs to only one data subset in the partitions with no overlap. Only one restriction is imposed in both the partitioning schemes: all observations specific to a subject lie in the same subset.

Consider the linear mixed-effects model in (2) for subset j. Let β_j, L_j, σ_j^2, and $\theta_j = \{\beta_j, \text{vech}(L_j), \sigma_j^2\}$ be the subset j versions of β, L, σ^2, and θ. The model (2) for subject i in subset j is

$$y_{ji} = X_{ji}\beta_j + Z_{ji}L_j b_{ji} + e_{ji}, \quad b_{ji} \sim N_q(0, I), \quad e_{ji} \sim N_{s_{ji}}(0, \sigma_j^2 R_{ji}), \quad i = 1, \ldots, m_j. \tag{4}$$

The likelihoods of L_j and σ_j^2, respectively, are obtained using (4). The likelihood of β_j is defined using the following model obtained by marginalizing (or collapsing) over b_{ji} in (4) for $i = 1, \ldots, m_j$

$$y_{ji} = X_{ji}\beta_j + \tilde{e}_{ji}, \quad \tilde{e}_{ji} \sim N_{s_{ji}}\left\{0, U_{ji}(\theta)^{-1}\right\}, \quad U_{ji}(\theta) = \left(\sigma_j^2 R_{ji} + Z_{ji}L_j L_j^T Z_{ji}^T\right)^{-1}. \tag{5}$$

The priors on L_j, σ_j^2, and β_j are assumed to be the same as in (3) and $\pi(\theta_j) = \pi(\beta_j)\pi\{\text{vech}(L_j)\}\pi(\sigma_j^2)$.

A direct application of the conditional augmentation algorithm based on van Dyk (2000) is problematic. The posterior distribution of θ_j conditions on the data in subset j that contains (m_j/n)-fraction of the full data; therefore, the Bernstein-von Mises theorem implies that the variance of the jth subset posterior distribution is inflated by a factor of n/m_j relative to that of the full data posterior distribution of θ (Minsker et al., 2017). The estimate of posterior distribution of θ obtained using θ_j draws from the conditional augmentation of van Dyk (2000) overestimates uncertainty in θ. The modified conditional augmentation algorithm developed next ensures that the asymptotic order of uncertainty estimates for θ obtained using the jth subset and full data posterior distributions are the same as $n, m_j \to \infty$.

3.2 Second step: data augmentation on the subsets

We modify the conditional augmentation algorithm of van Dyk (2000) to ensure that the posterior uncertainty of θ_j is properly calibrated relative to that of the full data posterior distribution for θ. Let $\ell\{\text{vech}(L_j)\}$, $\ell(\sigma_j^2)$, and $\ell(\beta_j)$ be the conditional likelihoods of L_j, σ_j^2, and β_j computed using (4) and (5). For ξ_j equalling β_j, vech(L_j), or σ_j^2, the conditional density of jth subset posterior distribution for ξ_j given $\mathcal{D}_{j\text{ aug}}$ and the remaining parameters is defined as

$$\pi(\xi_j \mid \theta_j \backslash \xi_j, \mathcal{D}_{j\text{ aug}}) = \frac{\ell^{n/m_j}(\xi_j)\pi(\xi_j)}{\int \ell^{n/m_j}(\xi_j)\pi(\xi_j)d\xi_j}, \quad j = 1, \ldots, k, \tag{6}$$

where $\theta_j \backslash \xi_j$ represents all the parameters in θ_j except ξ_j, $\pi(\xi_j)$ is the prior imposed on ξ_j, and we have assumed that $\int \ell^{n/m_j}(\xi_j)\pi(\xi_j)d\xi_j$ is finite for every n, ξ_j, m_j, and j. The modification of the likelihood by raising it to the power of n/m_j in (6) is called *stochastic approximation* because it is equivalent to computing the conditional likelihood of ξ_j after replicating the data for every subject (n/m_j)-times in the jth subset. The subset posterior density in (6) extends stochastic approximation to models with random effects and uncertainty estimates of the full and subset posterior distributions have the same asymptotic order as $n, m_j \to \infty$ (Minsker et al., 2014, 2017).

The conditional augmentation algorithm cycles through the I and P steps in parallel on the k subsets. On subset j, the I step imputes b_{ji} $(i = 1, \ldots, m_j)$ and the P step predicts θ_j using the full conditionals defined in (6). At the end of tth iteration on subset j, let $b_{ji}^{(t)}$ and $\theta_j^{(t)}$ be the imputed b_{ji} and predicted θ_j. Then, the I step in the $(t+1)$-th iteration on subset j is

(a) draw $b_{ji}^{(t+1)}$ given $\theta_j^{(t)}$ and $\mathcal{D}_{j\text{ aug}} \backslash b_{ji}$ from $N_q\{m_{b_{ji}}(\theta_j^{(t)}), V_{b_{ji}}(\theta_j^{(t)})\}$ for $i = 1, \ldots, m_j$, where

$$m_{b_{ji}}(\theta_j^{(t)}) = L_j^{(t)T} Z_{ji}^T \left(Z_{ji} L_j^{(t)} L_j^{(t)T} Z_{ji}^T + \sigma_j^{2(t)} R_{ji} \right)^{-1} (y_{ji} - X_{ji}\beta_j^{(t)}), \tag{7}$$

$$V_{b_{ji}}(\theta_j^{(t)}) = I - L_j^{(t)T} Z_{ji}^T \left(Z_{ji} L_j^{(t)} L_j^{(t)T} Z_{ji}^T + \sigma_j^{2(t)} R_{ji} \right)^{-1} Z_{ji} L_j^{(t)}. \tag{8}$$

Our I step is identical to the I step in the conditional augmentation algorithm of van Dyk (2000) because the missing data model for the random effects remains unchanged on subset j.

Our P step draws $\theta_j^{(t+1)}$ given $b_{ji}^{(t+1)}$ $(i = 1, \ldots, m_j)$ in a sequence of three steps. Let $\mathcal{D}_{j\text{ aug}}^{(t+1)}$ denote the augmented data in the $(t+1)$-th iteration. The P step draws $\beta_j^{(t+1)}$, $L_j^{(t+1)}$, and $\sigma_j^{2(t+1)}$ as follows:

(b) marginalize over b_{ji}s and draw β_j given $L_j^{(t)}, \sigma_j^{2(t)}$, and $\mathcal{D}_{j\text{ aug}}^{(t+1)}$ from $N_p\{m_{\beta_j}^{(t+1)}, V_{\beta_j}^{(t+1)}\}$, where

$$m_{\beta_j}^{(t+1)} = \left\{ \frac{n}{m_j} \sum_{i=1}^{m_j} X_{ji}^T U_{ji}(\theta_j^{(t)}) X_{ji} + \Sigma_\beta^{-1} \right\}^{-1} \left\{ \frac{n}{m_j} \sum_{i=1}^{m_j} X_{ji}^T U_{ji}(\theta_j^{(t)}) y_{ji} + \Sigma_\beta^{-1} \mu_\beta \right\},$$

$$V_{\beta_j}^{(t+1)} = \left\{ \frac{n}{m_j} \sum_{i=1}^{m_j} X_{ji}^T U_{ji}(\theta_j^{(t)}) X_{ji} + \Sigma_\beta^{-1} \right\}^{-1}; \tag{9}$$

(c) define $\tilde{b}_j^{(t+1)} = (b_{j1}^{(t+1)}, \ldots, b_{jm_j}^{(t+1)})^T$ and draw vech(L_j) given $\sigma_j^{2(t)}, \beta_j^{(t+1)}$, and $\mathcal{D}_{j\text{ aug}}^{(t+1)}$ from $N_{\frac{q(q+1)}{2}}\{m_{L_j}^{(t+1)}, V_{L_j}^{(t+1)}\}$, where

$$m_{L_j}^{(t+1)} = \left(\frac{n}{m_j} \sum_{i=1}^{m_j} \tilde{Z}_{ji}^{(t+1)T} R_{ji}^{-1} \tilde{Z}_{ji}^{(t+1)} + \sigma^{2(t)} \Sigma_L^{-1} \right)^{-1} \left\{ \frac{n}{m_j} \sum_{i=1}^{m_j} \tilde{Z}_{ji}^{(t+1)T} R_{ji}^{-1} (y_{ji} - X_{ji}\beta_j^{(t+1)}) + \sigma_j^{2(t)} \Sigma_L^{-1} \mu_L \right.$$

$$V_{L_j}^{(t+1)} = \sigma_j^{2(t)} \left(\frac{n}{m_j} \sum_{i=1}^{m_j} \tilde{Z}_{ji}^{(t+1)T} R_{ji}^{-1} \tilde{Z}_{ji}^{(t+1)} + \sigma_j^{2(t)} \Sigma_L^{-1} \right)^{-1},$$

$$\tilde{Z}_{ji} = \left(b_{ji}^{(t+1)T} \otimes Z_{ji} \right) E_q, \quad i = 1, \ldots, m_j, \tag{10}$$

where E_q is the $q^2 \times q(q+1)/2$ matrix satisfying vec$(L_j) = E_q$vech(L_j); and

(d) draw σ_j^2 given $\beta_j^{(t+1)}, L_j^{(t+1)}, \tilde{b}_j^{(t+1)}$, and $\mathcal{D}_j^{(t+1)}$ from Inverse-Gamma$(a_j^{(t+1)}, b_j^{(t+1)})$, where

$$a_j^{(t+1)} = \frac{n}{2m_j}\tilde{s}_j + a,$$
$$b_j^{(t+1)} = \frac{n}{2m_j}\sum_{i=1}^{m_j} f_{ji}^{(t+1)T} R_{ji}^{-1} f_{ji}^{(t+1)} + b, \quad f_{ji}^{(t+1)} = y_{ji} - X_{ji}\beta_j^{(t+1)} - Z_{ji}L_j^{(t+1)}b_{ji}^{(t+1)}. \tag{11}$$

Our conditional augmentation algorithm in parts (a)–(d) is a generalization of the ECME$_1$ algorithm in van Dyk (2000) for distributed Bayesian inference. The computational complexity of this algorithm is $O\{\max(m_1, \ldots, m_k) \times \max(q^6, p^3)\}$. If n is large, then k is chosen to be large enough so that $m_j \ll n$ for every j and parts (a)–(d) pass through that data for m_j subjects in every iteration. The algorithm in parts (a)–(d) draws θ from a modified likelihood that is raised to a power of n/m_j, but it is applicable more generally in situations where the likelihood of θ has been raised to any positive power. In particular, if $k = 1$ and $m_j = n$, then parts (a)–(d) reduce to van Dyk's conditional augmentation algorithm for sampling from the posterior distribution of θ, which also implies that the algorithm in parts (a)–(d) is easy to implement and numerically stable.

The post burn-in subset posterior draws for β and L are collected on every subset. Let T be the total number of post burn-in iterations, $\beta_j^{(t)}$ and $L_j^{(t)}$ be the marginal β and L draws obtained from $\theta_j^{(t)}$ on subset j at the tth post burn-in iteration, and $\Pi_\beta(\cdot \mid \mathcal{D}_{j\text{ obs}})$ and $\Pi_L(\cdot \mid \mathcal{D}_{j\text{ obs}})$ be the marginal posterior distributions of β and L, with densities $\pi(\beta \mid \mathcal{D}_{j\text{ obs}})$ and $\pi(L \mid \mathcal{D}_{j\text{ obs}})$, obtained from $\Pi_{\mathcal{D}_{\text{aug}},\theta_j}(\cdot \mid \mathcal{D}_{j\text{ obs}})$. Under certain assumptions discussed in Section 4, $\beta_j^{(t)}$ and $L_j^{(t)}$ marginally follow $\Pi_\beta(\cdot \mid \mathcal{D}_{j\text{ obs}})$ and $\Pi_L(\cdot \mid \mathcal{D}_{j\text{ obs}})$, respectively, for $j = 1, \ldots, k$ and $t = 1, \ldots, T$; however, the posterior draws from a subset cannot be used for inference on β or L because they are obtained by conditioning on a fraction of the full data.

The WASP combines the subset posterior draws to obtain draws with a distribution that conditions on the full data. The linear program for estimating the WASPs for β and L approximates $\Pi_\beta(\cdot \mid \mathcal{D}_{j\text{ obs}})$ and $\Pi_L(\cdot \mid \mathcal{D}_{j\text{ obs}})$ using empirical measures supported on the jth subset posterior draws for β and L, respectively. To achieve an approximation error of ϵ in Wasserstein distance of order 2, the number of subset posterior draws must satisfy $T \geq C\max\{(1/\epsilon)^{p/2}, (1/\epsilon)^{q(q+1)/4}\}$ when $p \geq 3$ or $q \geq 2$, where C is a positive constant; see Theorem 15 in (Fournier and Guillin, 2015) for a precise statement; therefore, the number of subset posterior draws grows exponentially with the dimensions for obtaining accurate empirical approximations of $\Pi_\beta(\cdot \mid \mathcal{D}_{j\text{ obs}})$ and $\Pi_L(\cdot \mid \mathcal{D}_{j\text{ obs}})$, resulting in the curse of dimensionality when $p \geq 3$ or $q \geq 2$. Furthermore, the computational cost of the linear program for estimating the WASP grows as $O(T^5)$, which is expensive when ϵ is small and $T \geq C\max\{(1/\epsilon)^{p/2}, (1/\epsilon)^{q(q+1)/4}\}$. Resolving these two problems in the next section, we exploit the analytic properties of the WASP under certain regularity assumptions

to develop a conceptually simple and easy to implement combination algorithm.

3.3 Third step: combining the subset posterior distributions for β and L

3.3.1 Optimal transportation and its application for distributed Bayesian inference

We introduce two definitions from optimal transportation that are required in the combination step. The space of probability measures on $\mathbb{R}^d$ is $\mathcal{P}(\mathbb{R}^d)$ and the *Wasserstein space of order 2* probability measures on $\mathbb{R}^d$ is $\mathcal{P}_2(\mathbb{R}^d) = \{\nu \in \mathcal{P}(\mathbb{R}^d) : \int_{\mathbb{R}^d} \|\xi - \xi_0\|_2^2 \, \nu(d\xi) < \infty\}$, where $\|\cdot\|_2^2$ is the Euclidean distance, $\xi_0 \in \mathbb{R}^d$, and the definition of $\mathcal{P}_2(\mathbb{R}^d)$ does not depend on ξ_0. Let $\mu, \nu \in \mathcal{P}_2(\mathbb{R}^d)$ and $\mathcal{L}(\mu, \nu)$ be the set of all probability measures on $\mathbb{R}^d \times \mathbb{R}^d$ with marginals μ and ν. The Wasserstein distance of order 2 between μ and ν is $W_2(\mu, \nu) = \left\{ \inf_{\pi \in \mathcal{L}(\mu,\nu)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|x - y\|_2^2 \, d\pi(x, y) \right\}^{1/2}$.

The first definition required for the combination step is that of the Wasserstein barycenter. If $\nu_1, \ldots, \nu_k \in \mathcal{P}_2(\mathbb{R}^d)$ are k subset posterior distributions, then their Wasserstein barycenter is defined as

$$\overline{\nu} = \underset{\nu \in \mathcal{P}_2(\mathbb{R}^d)}{\operatorname{argmin}} \sum_{j=1}^{k} \frac{w_j}{2} W_2^2(\nu, \nu_j), \quad \sum_{j=1}^{k} w_j = 1, \quad w_1, \ldots, w_k > 0, \tag{12}$$

where w_j is the weight assigned to ν_j. We fix w_j at $1/k$ in our numerical experiments because the values of $m_1, \ldots, m_k$ are very similar. If $m_1, \ldots, m_k$ values differ significantly across the subsets, then we suggest choosing $w_j = m_j/(m_1 + \ldots + m_k)$, which gives more importance to subsets containing more subjects. It is known that $\overline{\nu}$ exists uniquely under general regularity assumptions (Agueh and Carlier, 2011). In the context of distributed Bayesian inference, ν_j is the jth subset posterior distribution and $\overline{\nu}$ is the WASP, which replaces the full data posterior distribution for inference (Srivastava et al., 2015).

The second definition is that of the location-scatter family of probability distributions:

Definition 3.1 (Location-scatter family; Álvarez-Esteban et al. (2016)) *Let X_0 be a random vector with probability law $P_0 \in \mathcal{P}_2(\mathbb{R}^d)$ such that $E(X_0) = 0$ and $cov(X_0) = I$, where I is a $d \times d$ identity matrix, $\mathcal{L}(W)$ be the probability distribution of a random variable W, and $\mathcal{M}_+^{d\times d}$ be the set of $d \times d$ positive definite matrices. The family $\mathcal{F}(P_0) = \{\mathcal{L}(\Sigma^{1/2} X_0 + \mu) : \Sigma \in \mathcal{M}_+^{d\times d}, \mu \in \mathbb{R}^d\}$ of probability laws induced by positive definite affine transformations from P_0 is called a location-scatter family, where $\Sigma^{1/2}$ is the symmetric square-root of Σ.*

The family $\mathcal{F}(P_0)$ includes elliptical and non-elliptical distributions; see Definition 2.1 in Álvarez-Esteban et al. (2016) for greater details.

3.3.2 Combination of subset posterior samples via the LS-WASP

We start by stating a theorem about the WASP if subset posterior distributions belong to the same location-scatter family. Let Q_0 be a probability measure with zero mean and identity covariance matrix similar to P_0 in Definition 3.1. The following theorem restates Theorems 2.3, 2.4, and 4.2 in Álvarez-Esteban et al. (2016) adapted to our setup.

Theorem 3.2 (Álvarez-Esteban et al. (2016)) *Assume that $\nu_1, \ldots, \nu_k \in \mathcal{F}(Q_0)$ for some Q_0 with zero mean vector and identity covariance matrix. Let $a_1, \ldots, a_k \in \mathbb{R}^d$ and $V_1, \ldots, V_k \in M_+^{d\times d}$ be the mean vectors and covariance matrices of $\nu_1, \ldots, \nu_k$. The Wasserstein barycenter $\overline{\nu}$ of $\nu_1, \ldots, \nu_k$ with weights $w_1, \ldots, w_k$ defined in (12) belongs to $\mathcal{F}(Q_0)$ with mean $\overline{a}$ and covariance matrix $\overline{V}$ that satisfy*

$$\overline{a} = \sum_{j=1}^{k} w_j a_j, \quad \overline{V} = \sum_{j=1}^{k} w_j \left(\overline{V}^{1/2} V_j \overline{V}^{1/2}\right)^{1/2} \tag{13}$$

where $\overline{V}$ exists uniquely and $\overline{V}^{1/2}$ denotes the symmetric square root of $\overline{V}$. Define the sequence of matrices S_t ($t = 0, \ldots, \infty$) starting from a given $S_0 \in M_+^{d\times d}$ as

$$S_{t+1} = S_t^{-1/2} \left\{ \sum_{j=1}^{k} w_j (S_t^{1/2} V_j S_t^{1/2})^{1/2} \right\}^2 S_t^{-1/2}, \quad t = 0, 1, 2, \ldots, \infty. \tag{14}$$

Let $\nu^{(t)}$ be a probability measure with mean $\overline{m}$ and covariance matrix S_t and $\nu^{(t)} \in \mathcal{F}(Q_0)$. Then, $W_2(\overline{\nu}, \nu^{(t)}) \to 0$ as $t \to \infty$, implying that $S_t \to \overline{V}$ in every matrix norm.

In our simulations, we set $S_0 = I$ and perform the iterations in (14) until $|\text{trace}(S_{t+1} - S_t)| < 10^{-6}$.

We bypass solving the WASP linear program by exploiting the analytic form of the mean vector and covariance matrix of the WASP in Theorem 3.2. Let m_{β_j}, m_{L_j} and V_{β_j}, V_{L_j} be the mean vectors and covariance matrices of $\Pi_\beta(\cdot \mid \mathcal{D}_{j\,\text{obs}})$, $\Pi_L(\cdot \mid \mathcal{D}_{j\,\text{obs}})$, and $\overline{m}_\beta$, $\overline{m}_L$ and $\overline{V}_\beta$, $\overline{V}_L$ be the mean vectors and covariance matrices of the WASPs of subset posterior distributions for β, L. Based on Definition 3.1 and Theorem 3.2, if there are probability measures Q_1 and Q_2 such that $\Pi_\beta(\cdot \mid \mathcal{D}_{j\,\text{obs}}) \in \mathcal{F}(Q_1)$ and $\Pi_L(\cdot \mid \mathcal{D}_{j\,\text{obs}}) \in \mathcal{F}(Q_2)$ for every $j = 1, \ldots, k$, then for any $\beta_j \sim \Pi_\beta(\cdot \mid \mathcal{D}_{j\,\text{obs}})$ and $L_j \sim \Pi_L(\cdot \mid \mathcal{D}_{j\,\text{obs}})$,

$$U_{1j} = V_{\beta_j}^{-1/2}(\beta_j - m_{\beta_j}) \sim Q_1, \quad U_{2j} = V_{L_j}^{-1/2}\{\text{vech}(L_j) - m_{L_j}\}) \sim Q_2. \tag{15}$$

Theorem 3.2 also implies that the WASPs of β and L belong to $\mathcal{F}(Q_1)$ and $\mathcal{F}(Q_2)$, respectively, so $\overline{m}_\beta + \overline{V}_\beta^{1/2} U_{1j}$ and $\overline{m}_L + \overline{V}_L^{1/2} U_{2j}$ follow the WASPs of β and L, respectively, for every $j = 1, \ldots, k$.

Our combination algorithm replaces the unknown parameters and random variables in (15) by their MCMC counterparts. The Monte Carlo estimates of m_{β_j}s, m_{L_j}s, V_{β_j}s, and V_{L_j}s based on the subset MCMC draws are

$$\hat{m}_{\beta_j} = \frac{1}{T}\sum_{t=1}^{T}\beta_j^{(t)}, \quad \hat{m}_{L_j} = \frac{1}{T}\sum_{t=1}^{T}\text{vech}(L_j^{(t)}), \quad \hat{V}_{\beta_j} = \frac{1}{T}\sum_{t=1}^{T}(\beta_j^{(t)} - \hat{m}_{\beta_j})(\beta_j^{(t)} - \hat{m}_{\beta_j})^T,$$
$$\hat{V}_{L_j} = \frac{1}{k}\sum_{j=1}^{k}\left\{\text{vech}(L_j^{(t)}) - \hat{m}_{L_j}\right\}\left\{\text{vech}(L_j^{(t)}) - \hat{m}_{L_j}\right\}^T. \tag{16}$$

We plug-in $\hat{m}_{\beta_j}$ and $\hat{m}_{L_j}$ for m_{β_j} and m_{L_j} in (13) and $\hat{V}_{\beta_j}$s and $\hat{V}_{L_j}$ for V_{β_j} and V_{L_j} in (14) to estimate the means and covariance matrices for the WASPs of β and L, which are denoted as $\hat{\overline{m}}_\beta$, $\hat{\overline{m}}_L$, $\hat{\overline{V}}_\beta$, and $\hat{\overline{V}}_L$. The Monte Carlo realizations of U_{1j} and U_{2j} in (15) based on $\beta_j^{(t)}$ and $L_j^{(t)}$ are

$$\hat{u}_{j1}^{(t)} = \hat{V}_{\beta_j}^{-1/2}\left(\beta_j^{(t)} - \hat{m}_{\beta_j}\right), \; \hat{u}_{j2}^{(t)} = \hat{V}_{L_j}^{-1/2}\left\{\text{vech}(L_j^{(t)}) - \hat{m}_{L_j}\right\}, \quad j = 1,\ldots,k, \; t = 1,\ldots,T, \tag{17}$$

Finally, we obtain t'th WASP draws of β and L as

$$\overline{\beta}^{(t')} = \hat{\overline{m}}_\beta + \hat{\overline{V}}_\beta^{1/2}\hat{u}_{j1}^{(t)}, \quad \text{vech}(\overline{L}^{(t')}) = \hat{\overline{m}}_L + \hat{\overline{V}}_L^{1/2}\hat{u}_{j2}^{(t)}, \quad t' = t + (j-1)T, \tag{18}$$

for every j and t.

In our simulated and real data analyses, we have observed that computation of $\hat{\overline{V}}_\beta$ and $\hat{\overline{V}}_L$ using Theorem 3.2 fails numerically because S_t in (14) becomes rank deficient. This problem persists irrespective of the choices for S_0, and it motivates the development of a numerically stable version of (14). We accomplish this by first rewriting the fixed point equation for $\overline{V}$ in (13) as

$$I = \sum_{j=1}^{k}\overline{V}^{-1/2}\left(\overline{V}^{1/2}w_j^2V_j\overline{V}^{1/2}\right)^{1/2}\overline{V}^{-1/2} \equiv \sum_{j=1}^{k}A_j, \quad A_j = \overline{V}^{-1/2}\left(\overline{V}^{1/2}w_j^2V_j\overline{V}^{1/2}\right)^{1/2}\overline{V}^{-1/2}, \tag{19}$$

where A_j is known as the geometric mean of $\overline{V}^{-1}$ and $w_j^2V_j$ and is denoted as $\overline{V}^{-1}\sharp(w_j^2V_j)$; see Section 4.1 in Bhatia (2009) for greater details. Taking the transpose of both sides in (19) implies that $I = \sum_{j=1}^{k}A_j^T$, and averaging this with $I = \sum_{j=1}^{k}A_j$ gives $I = \sum_{j=1}^{k}(A_j + A_j^T)/2$. Second, following the definition of fixed point iterations in (14), we first square both sides of $I = \sum_{j=1}^{k}(A_j + A_j^T)/2$ and define the new fixed point algorithm after multiplying by $\overline{S}_t^{1/2}$ on both

1. **Input**:
   (a) Subset posterior draws for β and L, $\beta_j^{(t)}$ and $L_j^{(t)}$ ($j = 1, \ldots, k$; $t = 1, \ldots, T$), and a known function of β, L, denoted as $f(\beta, L)$; for example, $f(\beta, L) = LL^T = D$ or $f(\beta, L) = \rho$, where ρ is the correlation matrix obtained from D.
   (b) Monte Carlo estimates of mean vectors and covariance matrices of the subset posterior distributions and the WASPs of β and L, $\hat{m}_{\beta_j}$, $\hat{m}_{L_j}$, $\hat{\overline{m}}_\beta$, $\hat{\overline{m}}_L$, $\hat{V}_{\beta_j}$, $\hat{V}_{L_j}$, $\hat{\overline{V}}_\beta$, $\hat{\overline{V}}_L$ ($j = 1, \ldots, k$).
2. **Do**:
   (a) Center and scale the subset posterior draws for $j = 1, \ldots, k$ and $t = 1, \ldots, T$ to define
   $$\hat{u}_{j1}^{(t)} = \hat{V}_{j\beta}^{-1/2}(\beta_j^{(t)} - \hat{m}_{\beta_j}), \quad \hat{u}_{j2}^{(t)} = \hat{V}_{jL}^{-1/2}\{\text{vech}(L_j^{(t)}) - \hat{m}_{L_j}\}.$$
   (b) For $j = 1, \ldots, k$ and $t = 1, \ldots, T$, define the t'th β and L WASP draws using the tth draws from the jth subset posterior distribution for β and L as
   $$\overline{\beta}^{(t')} = \hat{\overline{m}}_\beta + \hat{\overline{V}}_\beta^{1/2}\hat{u}_{j1}^{(t)}, \quad \text{vech}(\overline{L}^{(t')}) = \hat{\overline{m}}_L + \hat{\overline{V}}_L^{1/2}\hat{u}_{j2}^{(t)}, \quad t' = (j-1)T + t.$$
   (c) For $t' = 1, \ldots, kT$, define the t'th WASP draw for $f(\beta, L)$ as $f(\overline{\beta}^{(t')}, \overline{L}^{(t')})$.
3. **Return**: $\overline{\beta}^{(1)}, \ldots, \overline{\beta}^{(kT)}$, $\overline{L}^{(1)}, \ldots, \overline{L}^{(kT)}$, and $f(\overline{\beta}^{(1)}, \overline{L}^{(1)}), \ldots, f(\overline{\beta}^{(kT)}, \overline{L}^{(kT)})$.

Algorithm 1: Sampling from the WASP of β, L, $f(\beta, L)$ using the LS-WASP algorithm.

sides as

$$\overline{S}_{t+1} = \overline{S}_t^{1/2}\left(\sum_{j=1}^{k}\left[\overline{S}_t^{-1}\sharp(w_j^2 V_j) + \{S_t^{-1}\sharp(w_j^2 V_j)\}^T\right]/2\right)^2 \overline{S}_t^{1/2}, \quad t = 0, 1, 2, \ldots, \infty, \tag{20}$$

and declare the convergence of $\overline{S}_t$ sequence when $|\text{trace}(\overline{S}_{t+1} - \overline{S}_t)| < 10^{-6}$. The computation of $\overline{S}_t^{-1}\sharp(w_j^2 V_j)$ in (20) is greatly simplified by using the following result from matrix analysis: $\overline{S}_t^{-1}\sharp(w_j^2 V_j) = w_j \overline{S}_t^{-1}\left(\overline{S}_t V_j\right)^{1/2}$; see Theorem 4.1.3 in Bhatia (2009) for the proof. The matrix A_j has been discovered earlier for exploring the Riemannian geometry of a Gaussian distribution (Skovgaard, 1984), but we use it to combine covariance matrices of the subset posterior distributions, where we use the fixed point iterations in (20) instead of (14) for computing $\hat{\overline{V}}_\beta$ and $\hat{\overline{V}}_L$. Algorithm 1 summarizes the steps for obtaining the WASP draws of β, L, and a known function $f(\beta, L)$ using the subset posterior draws. It is justified theoretically in the next section.

Algorithm 1 has several advantages, including conceptual and computational simplicity. First, an important advantage of Algorithm 1 is that it does not assume that the subsets are mutually independent or subset posterior distributions are Gaussian; therefore, our algorithm is applicable

even for dependent non-Gaussian subset posterior distributions that belong to the same location-scatter family. This feature differentiates Algorithm 1 from the DPMC algorithm. Second, unlike the PIE algorithm or the algorithm in Srivastava et al. (2015), Algorithm 1 generates draws from the WASP rather than approximating the WASP using an empirical measure. This feature allows it to bypass the curse of dimensionality when $p \geq 3$ or $q \geq 2$. Furthermore, the PIE algorithm is limited to computation of marginal barycenters and it is not clear how to compute the joint barycenter from a collection of marginal barycenters. Finally, if all the $V_{\beta_j}^{(t)}$s and $V_{L_j}^{(t)}$s are the same, then Algorithm 1 reduces to the DPMC algorithm.

4 Theoretical properties of the LS-WASP algorithm

4.1 Theoretical properties of the Markov chains generated on the subsets

The I and P steps in parts (a)–(d) form a partially collapsed Gibbs sampler (Van Dyk and Park, 2008; Park and Van Dyk, 2009). At the $(t+1)$-th iteration, we draw (a) $(b_{j1}, \ldots, b_{jm_j})$ given $\beta_j^{(t)}$, $L_j^{(t)}$, $\sigma_j^{2(t)}$, $\mathcal{D}_{j\text{ obs}}$, (b) β_j given $L_j^{(t)}$, $\sigma_j^{2(t)}$, $\mathcal{D}_{j\text{ obs}}$ after collapsing over b_{ji}s, (c) L_j given $\sigma_j^{2(t)}$, $\beta_j^{(t+1)}$, $\mathcal{D}_{j\text{ aug}}^{(t+1)}$, and (d) σ_j^2 given $\beta_j^{(t+1)}$, $L_j^{(t+1)}$, $\mathcal{D}_{j\text{ aug}}^{(t+1)}$ for $t = 0, \ldots, \infty$. A slower-mixing parent sampler of the sampler in (a)–(d) blocks over β and L and has three steps (i) same as (a), (ii) draw (β, L) given $\sigma_j^{2(t)}$, $\mathcal{D}_{j\text{ aug}}^{(t+1)}$, and (iii) same as (d). It is known that the Markov chain generated by (i)–(iii) is geometrically ergodic if L is diagonal (Román and Hobert, 2015), but new tools are required to develop similar theoretical guarantees for the Markov chain generated by parts (i)–(iii) and (a)–(d) for the general model in (4) after stochastic approximation.

Developing such tools is outside the scope of this work, but the MCMC theory is greatly simplified if σ^2 is known. Let β_0, L_0, and σ_0^2 be the true values of β, L, and σ^2. Given $\sigma^2 = \sigma_0^2$, the modified form of (4) after stochastic approximation is

$$y_{ji} = \tilde{X}_{ji}\theta_j + \epsilon_{ji}, \;\; \tilde{X}_{ji} = [X_{ji}\ \tilde{Z}_{ji}], \;\; \tilde{Z}_{ji} = b_{ji}^T \otimes Z_{ji}, \;\; e_{ji} \sim N_{s_{ji}}(0, \tfrac{m_j\sigma_0^2}{n} R_{ji}), \;\; b_{ji} \sim N_q(0, I), \tag{21}$$

where $\theta_j = \{\beta_j, \text{vech}(L_j)\}$. Under the setup of (21), parts (a)–(d) and (i)–(iii) are modified to new samplers with parts (a′)–(c′) and (i′)–(ii′) that fix $\sigma_j^{2(t)} = \sigma_0^2$ for every t and remove part (d) and (iii), respectively, which draw σ_j^2. The full conditionals in (i′) and (ii′) are Gaussian, which implies that $\{\mathcal{D}_{j\text{ aug}}, \beta_j, \text{vech}(L_j)\}$ given $\mathcal{D}_{j\text{ obs}}$ follows a Gaussian distribution. The last result, Section 2.2 of Meng and Van Dyk (1999), and Liu (1994) imply that the geometric rate of convergence of the Markov chain $\{\{\beta_j^{(t)}, \text{vech}(L_j^{(t)})\}, t \geq 0\}$ generated by parts (i′)–(ii′) is $\lambda_j^{\text{DA}} = \|I - I_{j\text{ obs}} I_{j\text{ aug}}^{-1}\|_{\text{op}}$,

where $\|A\|_{\text{op}}$ is the maximum eigenvalue of the symmetric positive definite matrix A,

$$I_{j\text{ obs}} = -\frac{\partial^2 \log \pi(\theta_j \mid \mathcal{D}_{j\text{ obs}})}{\partial\theta_j \partial\theta_j^T}\bigg|_{\theta_j=\theta_0}, \quad I_{j\text{ aug}} = E\left\{-\frac{\partial^2 \log \pi(\theta_j \mid \mathcal{D}_{j\text{ aug}})}{\partial\theta_j \partial\theta_j^T}\bigg| \mathcal{D}_{j\text{ obs}}, \theta_j\right\}\bigg|_{\theta_j=\theta_0}, \tag{22}$$

$\theta_0 = \{\beta_0, \text{vech}(L_0)\}$, and $\pi(\theta_j \mid \mathcal{D}_{j\text{ obs}})$, $\pi(\theta_j \mid \mathcal{D}_{j\text{ aug}})$ are the conditional densities of θ_j implied by (21). Because the sampler in parts (i′)–(ii′) is a slower mixing parent of the sampler in parts (a′)–(c′), the geometric ergodicity of the Markov chain generated by parts (i′)–(ii′) implies that the Markov chain generated by parts (a′)–(c′) is geometrically ergodic.

We justify replacing m_{β_j}s, m_{L_j}s, V_{β_j}s, and V_{L_j}s in Algorithm 1 by their MCMC estimates under the assumption that σ^2 is known and equals σ_0^2 in (4). The sampler with parts (a′)–(c′) is a faster-mixing variant of (i′)–(ii′) and is identical to the jth subset posterior sampler that fixes $\sigma_j^{2(t)} = \sigma_0^2$ for every t; therefore, central limit theorems for Monte Carlo averages based on jth subset posterior draws exist and $\hat{m}_{\beta_j} - m_{\beta_j} = O_{P_j}(T^{-1/2})$, $\hat{m}_{L_j} - m_{L_j} = O_{P_j}(T^{-1/2})$, $\hat{V}_{\beta_j} - V_{\beta_j} = O_{P_j}(T^{-1/2})$, and $\hat{V}_{L_j} - V_{L_j} = O_{P_j}(T^{-1/2})$ $(j = 1, \ldots, k)$, where every convergence is elementwise and P_j is the probability measure on $\{\beta_j^{(t)}, \text{vech}(L_j^{(t)}), t \geq 0\}$ Markov chain generated subset j; see supplementary material for derivations and the next section for assumptions under which the central limit theorems for Monte Carlo averages exist.

4.2 Monte Carlo and statistical errors in draws obtained using the LS-WASP algorithm

Our theoretical guarantees for the LS-WASP algorithm are based on (21). In Section 4.1, we have assumed that $\sigma^2 = \sigma_0^2$ for a simplified analysis of asymptotic Monte Carlo properties. We simplify the setup further by assigning flat priors on β and vech(L) and setting $m_j = m$, $w_j = 1/k$ for every j. The additional assumption still implies that the Markov chains generated across subsets are geometrically ergodic (Meng and Van Dyk, 1999), but it enables simpler analysis of asymptotic statistical properties. We introduce additional notations that are required to state our assumptions and theoretical results. Let $P_{\theta_0}^n$ be the true probability distribution of y implied by the full data version of (21), Π_n, Π_{jm} be the full data and jth subset posterior distributions of θ, and $\overline{\Pi}_n$ be the WASP of $\Pi_{1m}, \ldots, \Pi_{km}$. Define X_j and Z_j by stacking X_{ji}s and Z_{ji}s along the rows, R_j by stacking R_{ji}s into a block diagonal matrix, and X, Z, and R to be the full data versions of X_j, Z_j, and R_j.

The posterior inference using Algorithm 1 with known a σ^2 has two sources of errors. First, the Monte Carlo error arises from using the Monte Carlo estimates in (17) and (18). Second, we use $\overline{\Pi}_n$ for inference on θ instead of Π_n, which is the source of statistical error. This source of

error is named so because it is free of any Monte Carlo approximation. We impose the following assumptions in our theoretical setup so that the Monte Carlo and statistical errors decay to zero at the root-n rate as n, m, T tend to infinity:

1. There are probability distributions Q_1 and Q_2 specifying location-scatter families $\mathcal{F}(Q_1)$ and $\mathcal{F}(Q_2)$, respectively, such that the posterior distributions $\Pi_\beta(\cdot \mid \mathcal{D}_n)$ and $\Pi_\beta(\cdot \mid \mathcal{D}_j)$ $(j = 1, \ldots, k)$ belong to $\mathcal{F}(Q_1)$ and $\Pi_L(\cdot \mid \mathcal{D}_n)$ and $\Pi_L(\cdot \mid \mathcal{D}_j)$ $(j = 1, \ldots, k)$ belong to $\mathcal{F}(Q_2)$ with $P_{\theta_0}^n$-probability 1.

2. The number of fixed effects p and random effects q do not depend on n or m.

3. The number of subjects on a subset (m) and number of subsets (k) increase with n and satisfy $km = n\{1 + o_n(1)\}$, where $o_n(1) \to 0$ as $n, m \to \infty$.

4. There are $p \times p$, $q \times q$ symmetric positive definite matrices Ω_{XX}, Ω_{ZZ} that do not depend on n or m and that satisfy

$$\frac{1}{m} X_j^T R_j^{-1} X_j = \sigma_0^2 \Omega_{XX} + o_m(1), \quad \frac{1}{m} Z_j^T R_j^{-1} Z_j = \sigma_0^2 \Omega_{ZZ} + o_m(1), \quad j = 1, \ldots, k,$$

   where $o_m(1)$ is a matrix that converges to the zero matrix elementwise as $m \to \infty$.

5. There are $p \times p$, $q^2 \times q^2$ symmetric positive definite matrices $\Omega_{U_1U_1}$, $\Omega_{U_2U_2}$ that do not depend on n or m and that satisfy

$$\frac{1}{n} \sum_{i=1}^{n} X_i^T U_i(\theta_0) X_i = \Omega_{U_1U_1} + o_n(1), \quad \frac{1}{n} \sum_{i=1}^{n} \left[\{L_0^T Z_i^T Z_i U_i(\theta_0)\} \otimes \{U_i(\theta_0) Z_i^T Z_i L_0\} \right] = \Omega_{U_2U_2} + o_n(1),$$
$$\frac{1}{m} \sum_{i=1}^{m} X_{ji}^T U_{ji}(\theta_0) X_{ji} = \Omega_{U_1U_1} + o_m(1), \quad \frac{1}{m} \sum_{i=1}^{m} \left[\{L_0^T Z_{ji}^T Z_{ji} U_{ji}(\theta_0)\} \otimes \{U_{ji}(\theta_0) Z_{ji}^T Z_{ji} L_0\} \right] = \Omega_{U_2U_2} + o_m(1),$$

   for every $j = 1, \ldots, k$, where $U_i(\theta) = (Z_i L L^T Z_i^T + \sigma_0^2 R_i)^{-1}$, $U_{ji}(\theta) = (Z_{ji} L L^T Z_{ji}^T + \sigma_0^2 R_{ji})^{-1}$, and $o_m(1)$ is a matrix that converges to the zero matrix elementwise as $n, m \to \infty$.

Our assumptions are based on existing theoretical results in Meng and Van Dyk (1999) and Srivastava et al. (2018). Assumption 1 says that the full data and subset posterior distributions of β and L are members of the same location-scatter families, respectively, This assumption is used in deriving analytic expressions for means and covariance matrices of the WASPs for β and L. Assumption 2 ensures that the dimension of θ remains fixed as n, m tend to infinity and is used when we apply the strong law of large numbers and derive (22). Assumptions 3 and 4 are required for justifying that any subset posterior distribution is a noisy approximation of the full data posterior distribution. Assumption 3 is used to determine the limits of $\Pi_{1m}, \ldots, \Pi_{km}$ and Π_n as m and n tend to infinity. Assumption 4 is based on the strong law of large numbers and

imposes conditions on the partitions and full data such that the asymptotic orders of the means and covariance matrices of $\Pi_{1m}, \ldots, \Pi_{km}$ and Π_n are the same up to $o(1)$ terms. Similar to Assumption 4, Assumption 5 is used in proving that the geometric rate of convergences of the Markov chains generated on the k subsets and that generated by the full data sampler are identical up to $o(1)$ terms. Assumptions 1–4 and 1–5 are required for bounding the statistical and Monte Carlo errors, respectively.

Our assumptions are violated in applications where the full data or subset posterior distributions do not belong to the same location-scatter family. In these applications, Algorithm 1 provides an approximation of the true WASP based on a location-scatter family. Assumption 2 implies that Algorithm 1 is inapplicable for high-dimensional linear mixed-effects modeling. There are two main reason for imposing this restriction. First, theoretical results in Meng and Van Dyk (1999) assume that the parameter dimension is fixed, which rules out high-dimensional models. Second, in high-dimensional settings, the prior distribution impacts the rate of convergence of the posterior distribution of parameters, so it is critical to understand the theoretical properties of the prior distribution (Castillo et al., 2015). The literature on high-dimensional mixed-effects model is sparsely populated and frequentist results have become available only recently (Jiang et al., 2016; Dicker and Erdogdu, 2017). This implies that the prior distributions on L and β in (3) require a careful study as $p \to \infty$, $q \to \infty$ with $n \to \infty$, which is beyond the scope of this work. We demonstrate some of these difficulties later in Section 6.

We first quantify the rate of decay of the Monte Carlo error. Denote the means and covariance matrices of Π_n, Π_{jm}, $\overline{\Pi}_n$ as μ, μ_j, $\overline{\mu}$ and Σ, Σ_j, $\overline{\Sigma}$. If θ_j is distributed as Π_{jm}, then Assumption 1 implies that $\overline{\theta} = \overline{\mu} + \overline{\Sigma}^{1/2}\{\Sigma_j^{-1/2}(\theta_j - \mu_j)\}$ follows the WASP with $P_{\theta_0}^n$-probability 1 ($j = 1, \ldots, k$). In Algorithm 1, we replace $\overline{\mu}$, $\overline{\Sigma}$, Σ_j, and μ_j by their Monte Carlo estimates $\hat{\overline{\mu}}$, $\hat{\overline{\Sigma}}$, $\hat{\Sigma}_j$, and $\hat{\mu}_j$ to obtain $\overline{\theta}_T$, the Monte Carlo estimate of $\overline{\theta}$. Let $\overline{\Pi}_T$ be the distribution of $\overline{\theta}_T$ and $P^k = P_1 \otimes \cdots \otimes P_k$ be the probability measure on the Markov chains of θ generated by the k subset posterior distributions. Then, the following theorem defines the Monte Carlo error as $W_2(\overline{\Pi}_T, \overline{\Pi}_n)$ and proves that it decays at the root-n rate under certain assumptions.

Theorem 4.1 *If Assumptions 1–5 hold and $n = o(\sqrt{T})$, then as $n, T \to \infty$*

$$W_2(\overline{\Pi}_T, \overline{\Pi}_n) = o(n^{-1/2}) \text{ in } P^k\text{-probability}$$

with $P_{\theta_0}^n$-probability 1.

The proof of this theorem is in the supplementary materials along with other proofs. If $W_2(\overline{\Pi}_T, \overline{\Pi}_n) = o(n^{-1/2})$ in P^k-probability as $n, T \to \infty$, then $\overline{\Pi}_T$ and $\overline{\Pi}_n$ deliver the same inference in the sense that their posterior means and covariance matrices are the same if n is large

and $n^2 = o(T)$ on every subset. In practice, we run MCMC algorithms on all the subsets such that $T > n^2$.

We define the statistical error as $W_2(\overline{\Pi}_n, \Pi_n)$ and the following theorem shows that it decays to zero at the root-n rate under certain assumptions.

Theorem 4.2 *If Assumptions 1–4 hold, then as* $n \to \infty$

$$W_2(\Pi_n, \overline{\Pi}_n) = o(n^{-1/2}) \text{ in } P^n_{\theta_0}\text{-probability}.$$

The proof of this theorem is based on Theorem 1 in Srivastava et al. (2018). It is known that $\overline{\Sigma}$ and Σ are $O(n^{-1})$ in $P^n_{\theta_0}$-probability. If $W_2^2(\overline{\Pi}_n, \Pi_n)$ is $o(n^{-1})$ in $P^n_{\theta_0}$-probability and n is large, then this theorem implies that the posterior credible intervals obtained using $\overline{\Pi}_n$ and Π_n match up to $o(n^{-1/2})$ terms.

Extensions of Theorems 4.1 and 4.2 hold for functions of θ. In applications, the interest lies in the functions of β and L; for example, the covariance matrix of random effects $D = LL^T$ is a function of L. Consider any function f mapping $\theta \in \Theta$ to $\{f_1(\theta), \ldots, f_a(\theta)\} \in \mathbb{R}^a$ for some integer a; for example, if $f(\theta) = \text{vech}(LL^T)$, then $a = q(q+1)/2$. Let Π_{nf}, $\overline{\Pi}_{nf}$, and $\overline{\Pi}_{Tf}$ be the full data posterior distribution, the WASP, and the MCMC-based approximation of the WASP of $f(\theta)$. The following corollary states that $\overline{\Pi}_{Tf}$ provides an accurate alternative to Π_{nf} for inference on $f(\theta)$ if Assumptions 1–5 hold and the second moments of $f(\theta)$ with respect to $\Pi_{1m}, \ldots, \Pi_{km}$ and Π_n exist.

Corollary 4.3 *If the assumptions of Theorems 4.1 and 4.2 hold and* $\sum_{i=1}^a f_i^2(\theta) = \|f(\theta)\|_2^2 \leq C_f(1 + \|\theta\|_2^2)$*, where* C_f *is a universal constant free of* n *and* $\|\cdot\|_2$ *is the Euclidean norm, then the approximation error in using* $\overline{\Pi}_{Tf}$ *for inference on* $f(\theta)$ *instead of* Π_{nf} *as* $n, T \to \infty$ *is defined as*

$$W_2(\overline{\Pi}_{Tf}, \Pi_{nf}) = o_{P^k}(n^{-1/2}) + o_{P^n_{\theta_0}}(n^{-1/2}),$$

where the first and second terms on the right represent the asymptotic orders of Monte Carlo and statistical errors, respectively.

The proof follows from Corollary 5 in Srivastava et al. (2018) and noticing that $W_2(\overline{\Pi}_{Tf}, \Pi_{nf}) \leq W_2(\overline{\Pi}_{Tf}, \overline{\Pi}_{nf}) + W_2(\overline{\Pi}_{nf}, \Pi_{nf})$. The extra assumption on $\|f(\theta)\|_2^2$ is required to ensure that the posterior covariance matrix of $f(\theta)$ exists. Since $W_2(\overline{\Pi}_{Tf}, \overline{\Pi}_{nf})$ quantifies the approximation error in using $\overline{\Pi}_{Tf}$ instead of $\overline{\Pi}_{nf}$ for inference on $f(\theta)$, this corollary provides the theoretical basis for using L draws obtained using Algorithm 1 for inference on $D = LL^T$.

5 Experiments

5.1 Setup

We replicate the setup for linear mixed-effects modeling in Li et al. (2017). They show that the WASP performs better than its divide-and-conquer competitors in combining subset posterior draws; however, these empirical results do not compare the WASP with DPMC, so we use DPMC as our main competitor and focus on joint inference on multivariate parameters with dimensions greater than one. The MCMC based approximation of the full data posterior, denoted as $\hat{\Pi}_n$, is the benchmark in all our simulation comparisons. If $\tilde{\Pi}_n$ is the LS-WASP- or DPMC-based approximation of $\hat{\Pi}_n$, then the approximation error in using $\tilde{\Pi}_n$ instead of $\hat{\Pi}_n$ is

$$\text{Approximation Error} = \left[\|\mu - \tilde{\mu}\|_2^2 + \text{tr}\left\{\Sigma + \tilde{\Sigma} - 2(\Sigma^{1/2}\tilde{\Sigma}\Sigma^{1/2})^{1/2}\right\}\right]^{1/2}, \tag{23}$$

where μ, $\tilde{\mu}$ represent the means and Σ, $\tilde{\Sigma}$ represent the covariance matrices of $\hat{\Pi}_n$, $\tilde{\Pi}_n$, respectively. The error in (23) is defined based on $W_2(\hat{\Pi}_n, \tilde{\Pi}_n)$ and is small when the differences between the means and covariance matrices of $\hat{\Pi}_n$ and $\tilde{\Pi}_n$ are simultaneously small.

We compare the computational efficiencies of DPMC and LS-WASP based on their run-times and effective sample sizes (ESSs). If $\tilde{\text{ESS}}$ and ESS are the ESSs and $\tilde{t}$ and t are the run-times (in hours) of the DPMC or LS-WASP and full data posterior distributions, respectively, then the computational efficiency of DPMC or LS-WASP relative to the full data posterior distribution is defined based on Johndrow et al. (2019) as

$$\text{Computational Efficiency} = \log_2\{(\tilde{\text{ESS}}/\tilde{t})/(\text{ESS}/t)\}. \tag{24}$$

In our experiments, the run-times of DPMC and LS-WASP are very similar, so the differences between their computational efficiencies result mainly from the differences between their ESSs.

The full data and subset posterior draws are obtained using the data augmentation algorithm in Section 3.2. Every sampling algorithm is run for 10^4 iterations, with a burn-in time of 5×10^3 iterations. We collect every fifth MCMC draw after burn-in for inference on θ. DPMC and LS-WASP follow the generic three-step strategy for obtaining posterior draws for θ from the subset posterior draws and differ only in the combination step. All the three steps are implemented in R or R combined with C++ using Rcpp and run on an Oracle Grid Engine cluster with 56-core 128GB compute nodes running CentOS-7.4 Linux. The full data posterior computations are allocated four times more memory resources than those for subset posterior computations.

5.2 Simulated data analysis

The simulation setup follows the model in (1). We set $s = 10^6$, $n = 50,000$, and randomly assign the s samples to n subjects, with 20 being the average number of observations per subject (s_i). The entries in X_i and Z_i are set to 1 or -1 with equal probability for every i. We set $p = 4$, $q = 3$, $\beta = (-2, 2, -2, 2)$, $D_{11} = 1$, $D_{22} = 2$, $D_{33} = 3$, $D_{12} = -0.56$, $D_{31} = 0.52$, and $D_{23} = 0.0025$, $\sigma^2 = 0.01$, and $R_i = I$, where I is an identity matrix of appropriate dimension. The parameter values are the same as in Li et al. (2017) and responses are generated using (1). The simulation is replicated ten times.

The approximation error in (23) is evaluated for every two- and higher-dimensional joint distributions of parameters. We use β, D, and correlation matrix ρ obtained from D as the parameters for comparisons, where the last two parameters are functions of L. The approximation errors are computed for the posterior distributions of all a-dimensional parameters, where $1 < a \leq p$ for β, $1 < a \leq q(q+1)/2$ for D, and $1 < a \leq q(q-1)/2$ for ρ. The computational efficiency in (24) is evaluated using the draws for β and D. We also evaluate the effects of disjoint and overlapping data subsets on the approximation errors and computational efficiencies. We set k as $50, 100, 150, 200$ in both partitioning schemes. In the disjoint scheme, subset sample size decreases as k increases from 50 to 200. On the other hand, the subset sample sizes are fixed at 250 and 500, respectively, in the overlapping partitioning scheme for every k.

Our simulation setup highlights the generality of the assumptions of LS-WASP. Theoretical guarantees for DPMC rely on the asymptotic normality of the subset and full data posterior distributions, whereas LS-WASP assumes that subset and full data posterior distributions belong to the same location-scatter family. The asymptotic normality of the subset posterior distributions for β are easier to justify than those for L because L is a (lower triangular) Cholesky matrix; therefore, we expect LS-WASP to have a smaller approximation error than DPMC for inference on D and ρ whenever the subset sample size is small. This happens when k is large in the disjoint partitioning scheme. The LS-WASP scales the subset MCMC draws to have identity covariance in (17) before the re-scaling and re-centering step in (18). Because the scaling of subset MCMC draws is absent in DPMC, we expect the computational efficiency of LS-WASP to be slightly higher than that of DPMC for β, D, and ρ. The simulation results and conclusions for ρ are very similar to those for D, so we have moved them to the supplementary material.

The approximation errors of LS-WASP for inference on β and D are slightly smaller than those of DPMC, but the magnitudes of differences depend on the partitioning schemes (Tables 1–2). In both partitioning schemes, DPMC and LS-WASP have similar approximation errors for inference on β for every m, k and for inference on D when m is large. In all these cases, m is large enough to guarantee that the errors for DPMC and LS-WASP are of the same order. The approximation errors for DPMC and LS-WASP decrease with increasing k in the overlapping partitioning scheme for

Table 1: Average approximation errors (23) in estimating posterior distributions of two- and higher-dimensional parameters based on the entries of β and D under the disjoint partitioning scheme across ten simulation replications. The Monte Carlo errors across replications are in parentheses.

	β											
	2-dimensional				3-dimensional				4-dimensional			
	$k=50$	$k=100$	$k=150$	$k=200$	$k=50$	$k=100$	$k=150$	$k=200$	$k=50$	$k=100$	$k=150$	$k=200$
DPMC	0.005 (0.002)	0.005 (0.002)	0.007 (0.004)	0.008 (0.002)	0.01 (0.002)	0.011 (0.003)	0.013 (0.005)	0.014 (0.004)	0.046 (0.009)	0.051 (0.013)	0.062 (0.021)	0.072 (0.023)
LS-WASP	0.005 (0.002)	0.005 (0.002)	0.007 (0.004)	0.008 (0.002)	0.01 (0.002)	0.011 (0.003)	0.013 (0.005)	0.014 (0.004)	0.046 (0.009)	0.051 (0.013)	0.062 (0.021)	0.071 (0.023)
	D											
	2-dimensional				3-dimensional				4-dimensional			
	$k=50$	$k=100$	$k=150$	$k=200$	$k=50$	$k=100$	$k=150$	$k=200$	$k=50$	$k=100$	$k=150$	$k=200$
DPMC	1.243 (0.021)	1.911 (0.02)	2.396 (0.025)	2.834 (0.036)	1.047 (0.018)	1.606 (0.014)	2.02 (0.02)	2.392 (0.028)	1.848 (0.03)	2.817 (0.023)	3.574 (0.058)	4.229 (0.038)
LS-WASP	1.239 (0.014)	1.88 (0.011)	2.385 (0.016)	2.81 (0.019)	1.044 (0.01)	1.585 (0.008)	2.009 (0.014)	2.369 (0.015)	1.835 (0.018)	2.791 (0.019)	3.549 (0.023)	4.189 (0.026)
	5-dimensional				6-dimensional							
DPMC	5.043 (0.069)	7.682 (0.06)	9.787 (0.146)	11.566 (0.1)	30.929 (0.413)	47.135 (0.346)	60.015 (0.871)	70.986 (0.602)				
LS-WASP	5.012 (0.047)	7.627 (0.046)	9.702 (0.057)	11.46 (0.068)	30.734 (0.281)	46.781 (0.269)	59.504 (0.327)	70.298 (0.395)				

a given parameter dimension a and m because both methods combine a greater number of noisy approximations of the full data posterior distribution, resulting in smaller approximation errors for larger ks. On the other hand, in the disjoint partitioning scheme, the approximation errors of DPMC and LS-WASP increase with k and a, but the errors for LS-WASP are slightly smaller than those for DPMC when k is large.

We observe an interesting pattern in the approximation errors of D in the overlapping partitioning scheme. The errors for $m = 250$ are smaller than those for $m = 500$, and the difference increases with a. This pattern cannot be explain by the existing theoretical results in Li et al. (2017) and Srivastava et al. (2018), which focus on disjoint partitions. A possible explanation for this observation is that the second term in (23) (that is, $\mathrm{tr}\left\{\Sigma + \tilde{\Sigma} - 2(\Sigma^{1/2}\tilde{\Sigma}\Sigma^{1/2})^{1/2}\right\}$) increases with m in overlapping partitions due to the increasing dependencies among the subset posterior distributions. On the other hand, if we define the squared bias based on (23) as $\|\mu - \tilde{\mu}\|_2^2$, then the squared bias decreases with m. The second term in (23) increases more quickly with m than the decrease in squared bias, which results in an increase in the approximation error with m. This empirical result motivates a general strategy for choosing k in divide-and-conquer Bayesian inference with overlapping partitions based on the bias-variance trade-off, which we leave as future work.

The computational efficiencies of LS-WASP are slightly larger than that of DPMC in both partitioning schemes and for every parameter, m, and k (Tables 3–4). This happens mainly because the ESS of LS-WASP is higher than that of DPMC due to the additional step in LS-WASP that scales the subset MCMC draws. The number of subset MCMC draws increases with k in DPMC and LS-WASP, which leads to the increase in computational efficiencies with k. DPMC and LS-WASP are significantly faster than the full data posterior computations. Specifically, both methods require 1.39 and 0.74 hours on an average to finish when $k = 50$ and $k = 200$, respectively, whereas the full data posterior computations run for 55 hours on an average. Because LS-WASP

Table 2: Average approximation errors (23) in estimating posterior distributions of two- and higher-dimensional parameters based on the entries of β and D under the overlapping partitioning scheme with $m = 250$ and $m = 500$ across ten simulation replications. The Monte Carlo errors across replications are in parentheses. The '-' entry indicates that $km < n$; that is, the sum of the number of subjects on all the subsets is less than the total number of subjects.

	β											
	$m = 250$											
	2-dimensional				3-dimensional				4-dimensional			
	$k = 50$	$k = 100$	$k = 150$	$k = 200$	$k = 50$	$k = 100$	$k = 150$	$k = 200$	$k = 50$	$k = 100$	$k = 150$	$k = 200$
DPMC	-	-	-	0.045	-	-	-	0.087	-	-	-	0.421
	-	-	-	(0.021)	-	-	-	(0.037)	-	-	-	(0.13)
LS-WASP	-	-	-	0.045	-	-	-	0.087	-	-	-	0.421
	-	-	-	(0.021)	-	-	-	(0.037)	-	-	-	(0.13)
	$m = 500$											
	2-dimensional				3-dimensional				4-dimensional			
	$k = 50$	$k = 100$	$k = 150$	$k = 200$	$k = 50$	$k = 100$	$k = 150$	$k = 200$	$k = 50$	$k = 100$	$k = 150$	$k = 200$
DPMC	-	0.079	0.06	0.051	-	0.156	0.125	0.108	-	0.693	0.555	0.446
	-	(0.045)	(0.029)	(0.025)	-	(0.065)	(0.055)	(0.044)	-	(0.244)	(0.198)	(0.168)
LS-WASP	-	0.079	0.06	0.051		0.156	0.125	0.108	-	0.693	0.555	0.446
	-	(0.045)	(0.029)	(0.025)	-	(0.065)	(0.055)	(0.044)	-	(0.244)	(0.198)	(0.168)

	D											
	$m = 250$											
	2-dimensional				3-dimensional				4-dimensional			
	$k = 50$	$k = 100$	$k = 150$	$k = 200$	$k = 50$	$k = 100$	$k = 150$	$k = 200$	$k = 50$	$k = 100$	$k = 150$	$k = 200$
DPMC	-	-	-	1.264	-	-	-	1.063	-	-	-	1.872
	-	-	-	(0.012)	-	-	-	(0.009)	-	-	-	(0.021)
LS-WASP	-	-	-	1.258	-	-	-	1.058	-	-	-	1.863
	-	-	-	(0.012)	-	-	-	(0.009)	-	-	-	(0.021)
	5-dimensional				6-dimensional							
DPMC	-	-	-	5.108	-	-	-	31.325				
	-	-	-	(0.054)	-	-	-	(0.32)				
LS-WASP	-	-	-	5.084	-	-	-	31.177				
	-	-	-	(0.054)	-	-	-	(0.318)				
	$m = 500$											
	2-dimensional				3-dimensional				4-dimensional			
	$k = 50$	$k = 100$	$k = 150$	$k = 200$	$k = 50$	$k = 100$	$k = 150$	$k = 200$	$k = 50$	$k = 100$	$k = 150$	$k = 200$
DPMC	-	1.952	1.914	1.915	-	1.642	1.608	1.613	-	2.864	2.864	2.859
	-	(0.058)	(0.044)	(0.04)	-	(0.047)	(0.032)	(0.028)	-	(0.051)	(0.091)	(0.053)
LS-WASP	-	1.932	1.904	1.892	-	1.627	1.601	1.595	-	2.843	2.851	2.833
	-	(0.041)	(0.024)	(0.014)	-	(0.034)	(0.017)	(0.014)	-	(0.037)	(0.083)	(0.041)
	5-dimensional				6-dimensional							
DPMC	-	7.822	7.798	7.812	-	47.93	47.826	47.902				
	-	(0.117)	(0.198)	(0.115)	-	(0.717)	(1.172)	(0.714)				
LS-WASP	-	7.771	7.76	7.736	-	47.637	47.598	47.439				
	-	(0.084)	(0.188)	(0.102)	-	(0.528)	(1.112)	(0.631)				

has lower approximation errors and higher computational efficiencies than DPMC when $k < m$ and $m \ll n$, we conclude that LS-WASP is better than DPMC for inference on L and its functions, including D, in these cases.

5.3 MovieLens ratings data analysis

We evaluate the performance of DPMC and LS-WASP on the 10M MovieLens ratings data (`http://grouplens.org`). This data set contains about 10 million movie ratings from about 72 thousand users, where the ratings range from 0.5 to 5 in increments of 0.5. Following Perry (2017), we have used the MovieLens ratings data to define three new predictors capturing a movie's category, a movie's popularity, and a user's mood. Four movie categories are defined based on the genre of a movie, which is a non-empty subset of the 19 predefined genres in the MovieLens ratings data. A movie belongs to the *action* category if its genre is action, adventure, fantasy, horror, sci-fi, or thriller, to the *children* category if its genre is animation or children, to the *drama* category

Table 3: Average computational efficiencies (24) in estimating the posterior distributions of β and D under the disjoint partitioning scheme across ten simulation replications. The Monte Carlo errors across replications are in parentheses.

	β				D			
	$k = 50$	$k = 100$	$k = 150$	$k = 200$	$k = 50$	$k = 100$	$k = 150$	$k = 200$
DPMC	10.458 (0.088)	12.402 (0.093)	13.44 (0.101)	14.096 (0.092)	10.391 (0.085)	12.365 (0.087)	13.523 (0.097)	14.328 (0.083)
LS-WASP	10.459 (0.089)	12.404 (0.093)	13.448 (0.1)	14.117 (0.092)	10.438 (0.083)	12.413 (0.085)	13.575 (0.094)	14.389 (0.086)

Table 4: Average computational efficiencies (24) in estimating the posterior distributions of β and D under the overlapping partitioning scheme with $m = 250$ and $m = 500$ across ten simulation replications. The Monte Carlo errors across replications are in parentheses. The '-' entry indicates that $km < n$; that is, the sum of the number of subjects on all the subsets is less than the total number of subjects.

	β							
	$m = 250$				$m = 500$			
	$k = 50$	$k = 100$	$k = 150$	$k = 200$	$k = 50$	$k = 100$	$k = 150$	$k = 200$
DPMC	- -	- -	- -	13.323 (0.092)	- -	12.342 (0.107)	12.836 (0.099)	13.338 (0.099)
LS-WASP	- -	- -	- -	13.325 (0.093)	- -	12.344 (0.105)	12.84 (0.097)	13.34 (0.098)
	D							
	$m = 250$				$m = 500$			
	$k = 50$	$k = 100$	$k = 150$	$k = 200$	$k = 50$	$k = 100$	$k = 150$	$k = 200$
DPMC	- -	- -	- -	13.233 (0.082)	- -	12.3 (0.096)	12.799 (0.074)	13.3 (0.099)
LS-WASP	- -	- -	- -	13.275 (0.082)	- -	12.346 (0.096)	12.852 (0.077)	13.351 (0.099)

if its genre is crime, documentary, drama, film-noir, musical, mystery, romance, war, or western, and to the *comedy* category if its genre is comedy. A movie category predictor is a vector of length four representing the four categories. If a movie belongs to $C \in \{1, 2, 3, 4\}$ categories, then the entries in the movie's predictor vector for these categories is $1/C$ and the remaining entries are 0.

The movie popularity and user mood predictors are computed using the movie ratings. If a user's rating for a movie is greater than or equal to four, then we say that the user likes the movie. The popularity of a movie is defined as $\text{logit}\{(l + 0.5)/(r + 1)\}$, where l and r respectively are the number of users who liked and rated the movie among the 30 or fewer most recent reviewers of the movie. A user's mood is defined as 1 if the user liked the previously rated movie and 0 otherwise. The movie popularity and user mood are treated as numeric predictors, and the effect of movie category predictor is coded with *action* category as the baseline such that the category coefficients sum to zero.

We perform 10 replications of our experiment by randomly selecting 50,000 users (subjects) in each replication. The number of ratings (samples) in each replication is more than seven million. The total number of fixed effect predictors is 6, where four predictors are for the movie categories and the remaining two are movie popularity and user mood. The random effect predictors are the same as fixed effect predictors. The dimensions of β and D are 6×1 and 6×6, respectively. In the application of DPMC and LS-WASP, we vary k as 100,200 for the disjoint partitioning scheme

and $k = 100, m = 2000$ for the overlapping partitioning scheme. The application of conditional data augmentation based on van Dyk (2000) is impractically slow in that it failed to finish 5000 iterations in a week. On the other hand, the DPMC and LS-WASP algorithms finish 10^4 iterations within 2 days; therefore, we do not present results for the full data posterior distribution.

Due to the absence of the full data posterior results as the benchmark, we present results for a modified form of the computational efficiency only (Table 5). The approximation errors cannot be calculated due to the absence of benchmark results; for the same reason, we define a modified computational efficiency based on (24) as $\log_2 \text{ESS}/t$, where ESS and t are the effective sample size and run-time (in hours) of the DPMC or WASP. The modified computational efficiencies of the LS-WASP and DPMC for all the parameters and partitioning schemes agree with our simulation results in that LS-WASP has a slightly higher modified computational efficiency than DPMC due to higher ESSs for every parameter across all settings. The subset sample sizes in both partitioning schemes are relatively large to satisfy the theoretical assumptions of DPMC and LS-WASP, so the full data posterior approximations obtained using both methods are very similar across all settings; see supplementary material for the definition of metric for comparing DPMC and LS-WASP.

Table 5: Average *modified* computational efficiencies in estimating the posterior distributions of β and D under the overlapping and disjoint partitioning schemes across ten replications of MovieLens ratings data analysis. The Monte Carlo errors across replications are in parentheses.

	β		
	Overlapping Partitions ($m = 2000$)	Disjoint Partitions ($m = 500$)	Disjoint Partitions ($m = 250$)
	$k = 100$	$k = 100$	$k = 200$
DPMC	22.045 (0.068)	22.815 (0.371)	24.434 (0.142)
LS-WASP	22.036 (0.066)	22.811 (0.365)	24.391 (0.132)
	D		
	Overlapping Partitions ($m = 2000$)	Disjoint Partitions ($m = 500$)	Disjoint Partitions ($m = 250$)
	$k = 100$	$k = 100$	$k = 200$
DPMC	20.189 (0.054)	21.039 (0.232)	21.79 (0.194)
LS-WASP	20.257 (0.056)	21.167 (0.233)	22.074 (0.048)

5.4 US natality data analysis

We compare DPMC and LS-WASP on perinatal health data from the United States National Center for Health Statistics (US natality data) (Abrevaya, 2006). Li et al. (2017) have used this data to show that the WASP performs better than competing approaches even when the asymptotic normality fails to hold for the subset posterior distributions, but their results are restricted to one-dimensional parameters. Our empirical results show that the conclusions of Li et al. (2017) are also true for multivariate parameters. We have also chosen this example because the application of conditional data augmentation algorithm of van Dyk (2000) is feasible, which allows us to compare the approximation errors and computational efficiencies of DPMC and LS-WASP.

Table 6: Average approximation errors (23) in estimating posterior distributions of two- and higher-dimensional parameters based on the entries of β and D under the disjoint partitioning scheme across ten replications of the US natality data analysis. The Monte Carlo errors across replications are in parentheses.

	β									
	2-dimensional		3-dimensional		4-dimensional		5-dimensional		6-dimensional	
	$k = 20$	$k = 30$	$k = 20$	$k = 30$	$k = 20$	$k = 30$	$k = 20$	$k = 30$	$k = 20$	$k = 30$
DPMC	0.022 (0.009)	0.028 (0.012)	0.008 (0.003)	0.012 (0.004)	0.003 (0.001)	0.005 (0.002)	0.002 (0.001)	0.002 (0.001)	0.001 (0)	0.002 (0.001)
LS-WASP	0.021 (0.009)	0.027 (0.012)	0.008 (0.004)	0.011 (0.005)	0.003 (0.001)	0.004 (0.002)	0.002 (0.001)	0.002 (0.001)	0.001 (0)	0.002 (0.001)
	D									
	2-dimensional		3-dimensional		4-dimensional		5-dimensional		6-dimensional	
	$k = 20$	$k = 30$	$k = 20$	$k = 30$	$k = 20$	$k = 30$	$k = 20$	$k = 30$	$k = 20$	$k = 30$
DPMC	0.123 (0.023)	0.122 (0.014)	0.219 (0.036)	0.277 (0.03)	0.292 (0.048)	0.369 (0.04)	1.005 (0.099)	1.504 (0.21)	6.812 (0.593)	9.891 (1.157)
LS-WASP	0.113 (0.023)	0.101 (0.014)	0.197 (0.037)	0.234 (0.023)	0.263 (0.049)	0.312 (0.03)	0.88 (0.1)	1.202 (0.072)	6.05 (0.608)	8.12 (0.392)

The US natality data contains samples from 3809 mothers. There are two observation for every mother, so $n = 3809$ and $s = 7618$. The response is infant's birthweight, and the covariates include 13 birth-specific predictors. Following Li et al. (2017), we use 14 variables including all 13 covariates and one intercept as fixed effects and mother's age, gestation period, and number of living infants as random effects in (2). We partition the data for mothers into k disjoint and overlapping subsets. Choosing k greater than 30 for disjoint partitions results in unstable estimation of subset posterior distributions, so we set $k = 20, 30$ for disjoint partitions. Motivated from the simulations, we use $k = 20, 30, 50, 100$ and $m = 250, 500$ for overlapping partitions. This setup is replicated ten times for different selections of mothers.

The differences in the approximation errors of DPMC and LS-WASP for inference on D are larger than those in the simulations (Tables 6–7). Agreeing with our simulation results, LS-WASP and DPMC have similar approximation errors for all two- to six-dimensional parameters based on β. The number of subjects per subset is very small in the disjoint partitioning scheme to justify the asymptotic normality of subset posterior distributions for L, so DPMC's assumptions are violated and LS-WASP has a clear advantage over DPMC when $k = 30$. This results in smaller approximation errors of LS-WASP than DPMC for inference on two- to six-dimensional parameters based on D when $k = 30$ in the disjoint partitioning scheme. The same conclusions also holds for overlapping partitions when $m = 250$.

Agreeing with our simulation results, the computational efficiencies of LS-WASP are higher than those of DPMC in every setting (Table 8). The only change is that the differences are much higher than those observed in the simulations. The estimates of the posterior distributions of parameters based on LS-WASP are more stable than those based on DPMC. This happens mainly due to the two scaling of subset MCMC draws in LS-WASP, which is absent in DPMC; see (17) and (18). For example, despite the relatively large subset sample size, Figure 1 shows that bi-variate kernel density estimates of LS-WASP are more stable than those of DPMC as k changes from 20

Table 7: Average approximation errors (23) in estimating posterior distributions of two- and higher-dimensional parameters based on the entries of β and D under the overlapping partitioning scheme with $m = 250$ and $m = 500$ across ten replications of the US natality data analysis. The Monte Carlo errors across replications are in parentheses.

	β											
	$m = 250$											
	2-dimensional				3-dimensional				4-dimensional			
	$k = 20$	$k = 30$	$k = 50$	$k = 100$	$k = 20$	$k = 30$	$k = 50$	$k = 100$	$k = 20$	$k = 30$	$k = 50$	$k = 100$
DPMC	0.039 (0.018)	0.029 (0.021)	0.029 (0.019)	0.018 (0.008)	0.012 (0.006)	0.013 (0.005)	0.01 (0.005)	0.007 (0.004)	0.005 (0.002)	0.005 (0.002)	0.004 (0.002)	0.003 (0.001)
LS-WASP	0.039 (0.018)	0.029 (0.022)	0.029 (0.019)	0.018 (0.008)	0.012 (0.006)	0.013 (0.005)	0.01 (0.005)	0.007 (0.004)	0.005 (0.002)	0.005 (0.002)	0.004 (0.002)	0.003 (0.001)
	5-dimensional				6-dimensional							
DPMC	0.003 (0.001)	0.003 (0.001)	0.002 (0.001)	0.002 (0.001)	0.002 (0.001)	0.002 (0.001)	0.002 (0.001)	0.001 (0)				
LS-WASP	0.003 (0.001)	0.003 (0.001)	0.002 (0.001)	0.002 (0.001)	0.002 (0.001)	0.002 (0.001)	0.002 (0.001)	0.001 (0)				
	$m = 500$											
	2-dimensional				3-dimensional				4-dimensional			
	$k = 20$	$k = 30$	$k = 50$	$k = 100$	$k = 20$	$k = 30$	$k = 50$	$k = 100$	$k = 20$	$k = 30$	$k = 50$	$k = 100$
DPMC	0.026 (0.015)	0.02 (0.009)	0.018 (0.008)	0.013 (0.006)	0.01 (0.004)	0.007 (0.004)	0.007 (0.004)	0.004 (0.002)	0.004 (0.001)	0.003 (0.001)	0.003 (0.001)	0.002 (0.001)
LS-WASP	0.026 (0.015)	0.02 (0.009)	0.018 (0.008)	0.013 (0.006)	0.01 (0.004)	0.007 (0.004)	0.007 (0.004)	0.004 (0.002)	0.004 (0.001)	0.003 (0.001)	0.003 (0.001)	0.002 (0.001)
	5-dimensional				6-dimensional							
DPMC	0.002 (0.001)	0.002 (0.001)	0.002 (0.001)	0.001 (0)	0.002 (0)	0.001 (0)	0.001 (0)	0.001 (0)				
LS-WASP	0.002 (0.001)	0.002 (0.001)	0.002 (0.001)	0.001 (0)	0.002 (0)	0.001 (0)	0.001 (0)	0.001 (0)				

	D											
	$m = 250$											
	2-dimensional				3-dimensional				4-dimensional			
	$k = 20$	$k = 30$	$k = 50$	$k = 100$	$k = 20$	$k = 30$	$k = 50$	$k = 100$	$k = 20$	$k = 30$	$k = 50$	$k = 100$
DPMC	0.092 (0.033)	0.093 (0.016)	0.109 (0.017)	0.103 (0.011)	0.174 (0.032)	0.179 (0.031)	0.173 (0.023)	0.176 (0.022)	0.232 (0.042)	0.238 (0.041)	0.231 (0.031)	0.235 (0.03)
LS-WASP	0.083 (0.037)	0.085 (0.017)	0.103 (0.018)	0.096 (0.011)	0.152 (0.035)	0.156 (0.025)	0.154 (0.021)	0.155 (0.02)	0.203 (0.047)	0.208 (0.033)	0.206 (0.028)	0.207 (0.027)
	5-dimensional				6-dimensional							
DPMC	0.795 (0.091)	0.805 (0.151)	0.808 (0.14)	0.828 (0.117)	5.509 (0.437)	5.575 (0.866)	5.557 (0.743)	5.609 (0.657)				
LS-WASP	0.673 (0.09)	0.676 (0.081)	0.705 (0.124)	0.701 (0.079)	4.773 (0.486)	4.786 (0.528)	4.916 (0.643)	4.826 (0.453)				
	$m = 500$											
	2-dimensional				3-dimensional				4-dimensional			
	$k = 20$	$k = 30$	$k = 50$	$k = 100$	$k = 20$	$k = 30$	$k = 50$	$k = 100$	$k = 20$	$k = 30$	$k = 50$	$k = 100$
DPMC	0.103 (0.035)	0.083 (0.024)	0.088 (0.02)	0.09 (0.011)	0.133 (0.029)	0.134 (0.024)	0.124 (0.026)	0.131 (0.026)	0.178 (0.038)	0.178 (0.032)	0.166 (0.035)	0.175 (0.034)
LS-WASP	0.103 (0.035)	0.082 (0.025)	0.087 (0.021)	0.089 (0.011)	0.126 (0.029)	0.127 (0.025)	0.114 (0.027)	0.123 (0.025)	0.169 (0.038)	0.17 (0.034)	0.153 (0.035)	0.164 (0.033)
	5-dimensional				6-dimensional							
DPMC	0.614 (0.088)	0.602 (0.103)	0.623 (0.105)	0.629 (0.112)	4.355 (0.492)	4.401 (0.598)	4.291 (0.605)	4.405 (0.645)				
LS-WASP	0.579 (0.081)	0.566 (0.106)	0.571 (0.097)	0.578 (0.096)	4.097 (0.474)	4.142 (0.611)	3.918 (0.56)	4.051 (0.562)				

to 30. The full data posterior computations require 3 hours on an average to finish, whereas the DPMC and LS-WASP algorithms are significantly faster, requiring 0.42 and 0.41 hours to finish when $k = 20$ and $k = 100$, respectively. Because our US natality data results agree with our simulations results, we conclude that LS-WASP outperforms DPMC in inference on D and maintains the superior performance in both partitioning schemes.

Table 8: Average computational efficiencies (24) in estimating the posterior distributions of β and D under the disjoint scheme and overlapping partitioning schemes across ten replications of the US natality data analysis. The Monte Carlo errors across replications are in parentheses.

	β									
	Disjoint Partitions		Overlapping Partitions ($m = 250$)				Overlapping Partitions ($m = 500$)			
	$k = 20$	$k = 30$	$k = 20$	$k = 30$	$k = 50$	$k = 100$	$k = 20$	$k = 30$	$k = 50$	$k = 100$
DPMC	7.739 (0.13)	8.509 (0.07)	6.548 (0.098)	7.102 (0.122)	7.699 (0.105)	8.661 (0.12)	6.045 (0.146)	6.573 (0.126)	7.38 (0.138)	8.159 (0.081)
LS-WASP	8.028 (0.099)	9.046 (0.062)	6.746 (0.084)	7.293 (0.093)	7.895 (0.087)	8.871 (0.095)	6.084 (0.137)	6.636 (0.114)	7.431 (0.138)	8.219 (0.075)
	D									
	Disjoint Partitions		Overlapping Partitions ($m = 250$)				Overlapping Partitions ($m = 500$)			
	$k = 20$	$k = 30$	$k = 20$	$k = 30$	$k = 50$	$k = 100$	$k = 20$	$k = 30$	$k = 50$	$k = 100$
DPMC	5.598 (0.479)	6.458 (0.41)	4.108 (1.005)	4.499 (0.617)	5.081 (0.4)	5.904 (0.338)	3.723 (0.819)	4.478 (0.813)	4.682 (0.502)	5.381 (0.493)
LS-WASP	7.433 (0.485)	8.744 (0.331)	6.21 (0.433)	6.59 (0.364)	7.093 (0.27)	8.078 (0.254)	5.147 (0.409)	5.711 (0.38)	6.361 (0.404)	7.129 (0.251)

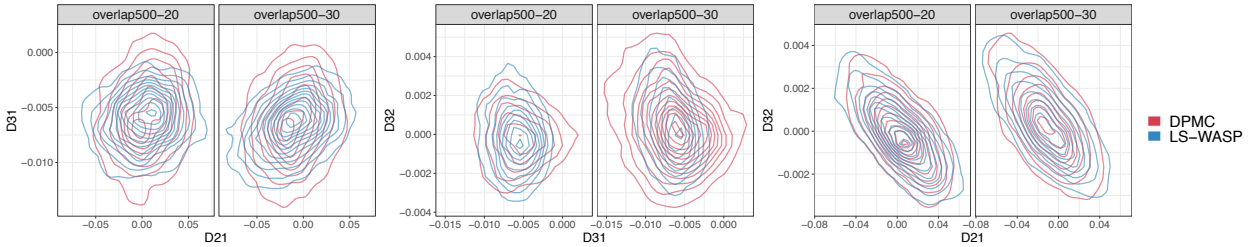

Figure 1: Contours of bi-variate kernel density estimates of the posterior distributions of (D_{21}, D_{31}), (D_{31}, D_{32}), and (D_{21}, D_{32}) obtained using DPMC and LS-WASP in the overlapping partitioning scheme with $m = 500$ and $k = 20, 30$ in a replication of US natality data analysis. In each of the three density plot, k increases from 20 to 30 across panels.

6 Discussion

We have presented the LS-WASP algorithm motivated from applications in linear mixed effects modeling, but it can be used for computing the WASP in other applications if the assumptions of Theorems 4.1 and 4.2 are justified. In many divide-and-conquer applications, it is reasonable to approximate the subset posterior distributions as members of a common location-scatter family. In these cases, Algorithm 1 provides an approximation to the true WASP based on a location-scatter family. One such application is divide-and-conquer nonparametric regression using Gaussian process priors (Guhaniyogi et al., 2017), where the PIE algorithm is used for simple and efficient marginal inference on the value of an unknown regression function at a given predictor value; however, joint inference on the values of unknown regression function at a collection of predictor values requires solving a computationally intensive linear program for computing the WASP. Algorithm 1 bypasses this problem while retaining the simplicity of the PIE algorithm. More research is required for developing analogues of Algorithm 1 for computing the WASP in time-series and network models that account for second and higher order dependencies.

We have used the LS-WASP algorithm for combining subset MCMC draws obtained using

conditional data augmentation, but it can be used with any type of sampling algorithm. The main reason for our choice is that we are able to prove Theorem 4.1 using a known result about geometric ergodicity of the Markov chain generated by conditional data augmentation. Similar results are unavailable for other sampling algorithms in mixed-effects modeling, especially for the Hamiltonian Monte Carlo (HMC) sampling algorithm used by Stan and the Stochastic Approximation Monte Carlo (SAMC) algorithm used by Xue and Liang (2019). Given a proof for geometric ergodicity of Stan's HMC or SAMC sampler is developed in the future, the proof of Theorem 4.1 demonstrates that such guarantees can be immediately extended to Markov chains generated by Stan's HMC or SAMC sampler on the subsets. We expect that Algorithm 1 is applicable for generating draws from the WASP if we have obtained subset posterior samples using Stan or SAMC.

We have remarked in Section 4.2 that Assumption 2 implies that the LS-WASP algorithm is inapplicable for high-dimensional linear mixed-effects modeling. The main reason for limiting our focus to finite dimensional models is that studying the impacts of priors on the posterior distributions of high-dimensional β and L is beyond the scope of this work. We present a simple example to illustrate some of the computational problems. Consider a linear random-effects model based on (2) with $p = 0$, $s_i = 1$, $y_i \in \mathbb{R}$, $z_i^T \in \mathbb{R}^{1\times q}$, and $R_i = 1$ for every i, $L = \tau I$, σ^2 is fixed at σ_0^2, and $q/n = c$, where c is a constant in $(0, 1)$ and τ is set to be positive for identifiability. In this set up, (2) and (3) reduce to

$$y_i = \tau z_i^T b_i + e_i, \quad e_i \sim N(0, \sigma_0^2), \quad b_i \sim N_q(0, I), \quad \tau \sim N(0, \psi^2), \quad i = 1, \ldots, n, \tag{25}$$

where τ is the parameter with prior variance ψ^2 and σ_0^2 is known. The data augmentation algorithm for sampling τ is recovered by setting $k = 1$ in parts (a)–(d) of the algorithm in Section 3.2:

(i) (I step) draw b_i given y_i and τ as $N_q(\frac{z_i y_i}{w_i \tau}, I - \frac{z_i z_i^T}{w_i})$ for $i = 1, \ldots, n$, where $w_i = \|z_i\|_2^2 + \sigma_0^2/\tau^2$; and

(ii) (P step) draw τ given $y_1, \ldots, y_n, b_1, \ldots, b_n$ as $N(\frac{\sum_{i=1}^n y_i (z_i^T b_i)}{v}, \frac{\sigma_0^2}{v})$, where $v = \sum_{i=1}^n (z_i^T b_i)^2 + \sigma_0^2/\psi^2$.

We evaluate the empirical performance of this algorithm through a simulation study. We set $\tau = 2$, $\sigma_0^2 = 0.01$, generate elements of z_i's as independent $N(0, 1)$ random variables, and use three different values of n and q: $n \in \{100, 500, 1000\}$ and $q = cn$, where $c \in \{0.1, 0.5, 0.9\}$. For every n and q, the data are simulated following (25) and posterior samples of τ are obtained by running the algorithm in steps (i)–(ii) for 20,000 iterations. After discarding the first 10,000 samples as burn-ins, the posterior samples of τ have extremely high serial autocorrelation for every n and q (Figure 2). The problem worsens as n increases because the fraction of augmented data (that is, $b_1, \ldots, b_n$) also increases with n. Furthermore, this problem is not resolved by the subset posterior

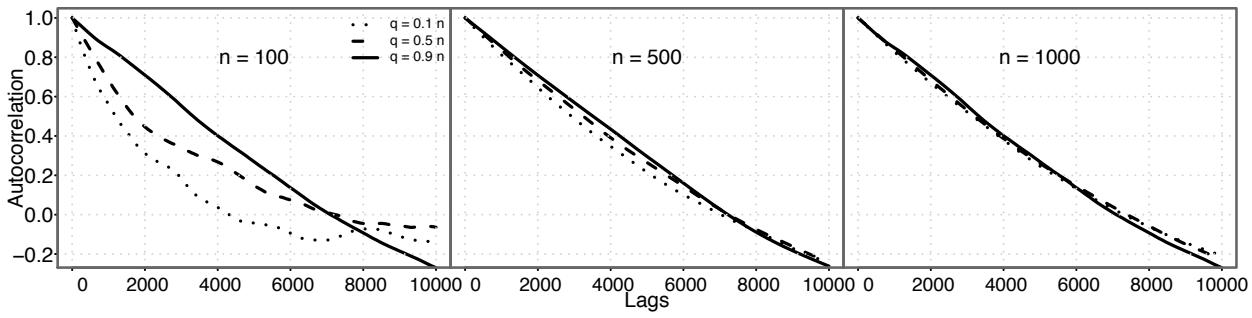

Figure 2: The auto-correlation function depending on n for the simulation example in (25). The x and y axes represent the lags and auto-correlation computed using the post burn-in MCMC draws obtained from the conditional data augmentation algorithm of van Dyk (2000) for inference on τ. The sample size (n) and number of random effects (q) increase along the panels from left to right.

sampling algorithms based on (i)–(ii). This example motivates study of the prior distributions on β and L, the rate at which p, q, k, and m increase to infinity with n, and data augmentation algorithms for posterior inference on β and L in high-dimensional linear mixed-effects models, which can be extended to massive data settings using a high-dimensional extension of the LS-WASP algorithm.

Supplementary Materials

The supplementary material is available online and contains proofs of the theoretical results and additional experimental results. The code used in the experiments is available at `https://github.com/blayes/LSWASP`.

Acknowledgment

We thank the Editor, Associate Editor, and two anonymous referees for their helpful comments and constructive criticisms. Yixiang Xu started working on this project when he was a graduate student at The University of Iowa. Cheng Li, Terrance Savitsky, N. D. Shyamalkumar, and Aixin Tan gave invaluable feedback on an initial draft of the manuscript. Sanvesh Srivastava's research is partially supported by grants from the Office of Naval Research (ONR-BAA N000141812741) and the National Science Foundation (DMS-1854667/1854662).

References

Abrevaya, J. (2006). Estimating the effect of smoking on birth outcomes using a matched panel data approach. *Journal of Applied Econometrics 21*, 489–519.

Agueh, M. and G. Carlier (2011). Barycenters in the Wasserstein space. *SIAM Journal on Mathematical Analysis 43*(2), 904–924.

Ahn, S., A. Korattikara, and M. Welling (2012). Bayesian posterior sampling via stochastic gradient Fisher scoring. *Proceedings of the 29th International Conference on Machine Learning*.

Alquier, P., N. Friel, R. Everitt, and A. Boland (2016). Noisy Monte Carlo: Convergence of Markov chains with approximate transition kernels. *Statistics and Computing 26*(1-2), 29–47.

Álvarez-Esteban, P. C., E. del Barrio, J. Cuesta-Albertos, and C. Matrán (2016). A fixed-point approach to barycenters in Wasserstein space. *Journal of Mathematical Analysis and Applications 441*(2), 744–762.

Arjovsky, M., S. Chintala, and L. Bottou (2017). Wasserstein gan. *arXiv preprint arXiv:1701.07875*.

Bardenet, R., A. Doucet, and C. Holmes (2017). On Markov chain Monte Carlo methods for tall data. *The Journal of Machine Learning Research 18*(1), 1515–1557.

Bernton, E., P. E. Jacob, M. Gerber, and C. P. Robert (2017). Inference in generative models using the Wasserstein distance. *arXiv preprint arXiv:1701.05146*.

Bhatia, R. (2009). *Positive definite matrices*. Princeton University Press.

Campbell, T. and T. Broderick (2018). Bayesian coreset construction via greedy iterative geodesic ascent. *arXiv preprint arXiv:1802.01737*.

Castillo, I., J. Schmidt-Hieber, A. Van der Vaart, et al. (2015). Bayesian linear regression with sparse priors. *The Annals of Statistics 43*(5), 1986–2018.

Claici, S. and J. Solomon (2018). Wasserstein Coresets for Lipschitz Costs. *arXiv preprint arXiv:1805.07412*.

Dicker, L. H. and M. A. Erdogdu (2017). Flexible results for quadratic forms with applications to variance components estimation. *The Annals of Statistics 45*(1), 386–414.

Fournier, N. and A. Guillin (2015). On the rate of convergence in Wasserstein distance of the empirical measure. *Probability Theory and Related Fields 162*(3-4), 707–738.

Gao, K. and A. Owen (2017). Efficient moment calculations for variance components in large unbalanced crossed random effects models. *Electronic Journal of Statistics 11*(1), 1235–1296.

Gelman, A. and J. Hill (2007). *Data analysis using regression and multilevel/hierarchical models*, Volume 3. Cambridge University Press New York, New York, USA.

Gelman, A., A. Vehtari, P. Jylänki, C. Robert, N. Chopin, and J. P. Cunningham (2014). Expectation propagation as a way of life. *arXiv preprint arXiv:1412.4869*.

Giordano, R., T. Broderick, and M. I. Jordan (2017). Covariances, robustness, and variational bayes. *arXiv preprint arXiv:1709.02536*.

Guhaniyogi, R., C. Li, T. D. Savitsky, and S. Srivastava (2017). Distributed Kriging for massive data. *arXiv preprint*.

Jiang, J., C. Li, D. Paul, C. Yang, and H. Zhao (2016). On high-dimensional misspecified mixed model analysis in genome-wide association study. *The Annals of Statistics 44*(5), 2127–2160.

Johndrow, J. E., J. C. Mattingly, S. Mukherjee, and D. B. Dunson (2015). Approximations of Markov Chains and High-Dimensional Bayesian Inference. *arXiv preprint arXiv:1508.03387v1*.

Johndrow, J. E., A. Smith, N. Pillai, and D. B. Dunson (2019). MCMC for imbalanced categorical data. *Journal of the American Statistical Association 114*(527), 1394–1403.

Korattikara, A., Y. Chen, and M. Welling (2014). Austerity in MCMC land: Cutting the Metropolis-Hastings budget. In *Proceedings of the 31st International Conference on Machine Learning*, pp. 181–189.

Kucukelbir, A., R. Ranganath, A. Gelman, and D. Blei (2015). Automatic variational inference in Stan. In *Advances in Neural Information Processing Systems*, pp. 568–576.

Lan, S., B. Zhou, and B. Shahbaba (2014). Spherical hamiltonian monte carlo for constrained target distributions. In *JMLR workshop and conference proceedings*, Volume 32, pp. 629. NIH Public Access.

Lee, C. Y. Y. and M. P. Wand (2016). Streamlined mean field variational Bayes for longitudinal and multilevel data analysis. *Biometrical Journal 58*(4), 868–895.

Li, C., S. Srivastava, and D. B. Dunson (2017). Simple, scalable and accurate posterior interval estimation. *Biometrika 104*, 665–680.

Li, J. and F. Zhang (2018). Geometry-Sensitive Ensemble Mean based on Wasserstein Barycenters: Proof-of-Concept on Cloud Simulations. *Journal of Computational and Graphical Statistics* (just-accepted), 1–30.

Liu, J. S. (1994). The fraction of missing information and convergence rate for data augmentation. *Computing Science and Statistics*, 490–490.

Maclaurin, D. and R. P. Adams (2015). Firefly Monte Carlo: Exact MCMC with Subsets of Data. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Meng, X.-L. and D. A. Van Dyk (1999). Seeking efficient data augmentation schemes via conditional and marginal augmentation. *Biometrika 86*(2), 301–320.

Minsker, S., S. Srivastava, L. Lin, and D. Dunson (2014). Scalable and robust Bayesian inference via the median posterior. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pp. 1656–1664.

Minsker, S., S. Srivastava, L. Lin, and D. B. Dunson ((2017)). Robust and scalable Bayes via a median of subset posterior measures. *Journal of Machine Learning Research*.

Neiswanger, W., C. Wang, and E. Xing (2014). Asymptotically exact, embarrassingly parallel MCMC. In *Proceedings of the 30th International Conference on Uncertainty in Artificial Intelligence*, pp. 623–632.

Park, T. and D. A. Van Dyk (2009). Partially collapsed Gibbs samplers: Illustrations and applications. *Journal of Computational and Graphical Statistics 18*(2), 283–305.

Perry, P. O. (2017). Fast moment-based estimation for hierarchical models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 79*(1), 267–291.

Quiroz, M., M. Villani, R. Kohn, M.-N. Tran, and K.-D. Dang (2018). Subsampling MCMC-A review for the survey statistician. *arXiv preprint arXiv:1807.08409*.

Ranganath, R., D. Tran, and D. Blei (2016). Hierarchical variational models. In *International Conference on Machine Learning*, pp. 324–333.

Robert, C. P. and G. Casella (2004). *Monte Carlo Statistical Methods*. Springer Verlag.

Román, J. C. and J. P. Hobert (2015). Geometric ergodicity of Gibbs samplers for Bayesian general linear mixed models with proper priors. *Linear Algebra and its Applications 473*, 54–77.

Rue, H., S. Martino, and N. Chopin (2009). Approximate Bayesian inference for latent Gaussian models by using integrated nested Laplace approximations. *Journal of the Royal Statistical Society: Series B (Statistical Methodology) 71*(2), 319–392.

Savitsky, T. D. and S. Srivastava (2018). Scalable Bayes under informative sampling. *Scandinavian Journal of Statistics*.

Scott, S. L., A. W. Blocker, F. V. Bonassi, H. A. Chipman, E. I. George, and R. E. McCulloch (2016). Bayes and big data: the consensus Monte Carlo algorithm. *International Journal of Management Science and Engineering Management 11*(2), 78–88.

Shahbaba, B., S. Lan, W. O. Johnson, and R. M. Neal (2014). Split Hamiltonian Monte Carlo. *Statistics and Computing 24*(3), 339–349.

Skovgaard, L. T. (1984). A Riemannian geometry of the multivariate normal model. *Scandinavian Journal of Statistics*, 211–223.

Srivastava, S., V. Cevher, Q. Dinh, and D. Dunson (2015). WASP: Scalable Bayes via barycenters of subset posteriors. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*, pp. 912–920.

Srivastava, S., C. Li, and D. B. Dunson (2018). Scalable Bayes via Barycenter in Wasserstein Space. *Journal of Machine Learning Research 19*, 312–346.

Staib, M., S. Claici, J. Solomon, and S. Jegelka (2017). Parallel Streaming Wasserstein Barycenters. *arXiv preprint arXiv:1705.07443*.

Stan Development Team (2017). Stan: A C++ library for probability and sampling, version 2.5.0.

Tan, L. S. and D. J. Nott (2014). A stochastic variational framework for fitting and diagnosing generalized linear mixed models. *Bayesian Analysis 9*(4), 963–1004.

van Dyk, D. A. (2000). Fitting mixed-effects models using efficient EM-type algorithms. *Journal of Computational and Graphical Statistics 9*(1), 78–98.

Van Dyk, D. A. and X.-L. Meng (2001). The art of data augmentation. *Journal of Computational and Graphical Statistics 10*(1), 1–50.

Van Dyk, D. A. and T. Park (2008). Partially collapsed Gibbs samplers: Theory and methods. *Journal of the American Statistical Association 103*(482), 790–796.

von Brzeski, V., M. Taddy, and D. Draper (2015). Causal inference in repeated observational studies: A case study of ebay product releases. *arXiv preprint arXiv:1509.03940*.

Wand, M. P. (2017). Fast approximate inference for arbitrarily large semiparametric regression models via message passing. *Journal of the American Statistical Association 112*(517), 137–168.

Wang, X. and D. B. Dunson (2013). Parallel MCMC via Weierstrass sampler. *arXiv preprint arXiv:1312.4605*.

Wang, X., F. Guo, K. A. Heller, and D. B. Dunson (2015). Parallelizing MCMC with random partition trees. In *Advances in Neural Information Processing Systems*, pp. 451–459.

Welling, M. and Y. W. Teh (2011). Bayesian learning via stochastic gradient Langevin dynamics. In *Proceedings of the 28th International Conference on Machine Learning*, pp. 681–688.

Xue, J. and F. Liang (2019). Double-Parallel Monte Carlo for Bayesian analysis of big data. *Statistics and computing 29*(1), 23–32.