Bayesian Analysis (2023) **18**, Number 3, pp. 889–907

Fast Methods for Posterior Inference of Two-Group Normal-Normal Models[*]

Philip Greengard[†], Jeremy Hoskins[‡], Charles C. Margossian[§], Jonah Gabry[¶], Andrew Gelman[‖], and Aki Vehtari[**]

Abstract. We describe a class of algorithms for evaluating posterior moments of certain Bayesian linear regression models with a normal likelihood and a normal prior on the regression coefficients. The proposed methods can be used for hierarchical mixed effects models with partial pooling over one group of predictors, as well as random effects models with partial pooling over two groups of predictors. We demonstrate the performance of the methods on two applications, one involving U.S. opinion polls and one involving the modeling of COVID-19 outbreaks in Israel using survey data. The algorithms involve analytical marginalization of regression coefficients followed by numerical integration of the remaining low-dimensional density. The dominant cost of the algorithms is an eigendecomposition computed once for each value of the outside parameter of integration. Our approach drastically reduces run times compared to state-of-the-art Markov chain Monte Carlo (MCMC) algorithms. The latter, in addition to being computationally expensive, can also be difficult to tune when applied to hierarchical models.

Keywords: hierarchical modeling, linear regression, mixed effects models, fast algorithms.

1 Introduction

Advances over the last decade in statistical methods and their implementation in open-source, user-friendly software have drastically simplified statistical modeling for applied researchers. For example, with probabilistic programming languages such as Stan (Carpenter et al., 2017) a user can specify and sample from a general choice of posterior density with flexible language and an easy-to-use interface. For its primary tool of inference, Stan (as well as other probabilistic programming languages) samples from the posterior distribution via dynamic Hamiltonian Monte Carlo sampler (HMC) (Betancourt, 2018; Hoffman and Gelman, 2014). HMC is a gradient-based sampling method

arXiv: 2110.03055

[*] Philip Greengard is supported by the Alfred P. Sloan Foundation. The authors thank the U.S. Office of Naval Research, Institute for Education Sciences, National Science Foundation, National Institutes of Health, and the Academy of Finland Flagship Programme: Finnish Center for Artificial Intelligence (FCAI) for partial support of this work.

[†] Columbia University, New York, USA, pg2118@columbia.edu
[‡] University of Chicago, Illinois, USA
[§] Columbia University, New York, USA
[¶] Columbia University, New York, USA
[‖] Columbia University, New York, USA
[**] Aalto University, Finland

https://doi.org/10.1214/22-BA1329

that has become ubiquitous in statistics over the last decade due to its being flexible, reliable, and general.

Despite its widespread use, HMC, as well as other Markov chain Monte Carlo (MCMC) methods, can have two drawbacks in statistical problems with large amounts of data—they can be prohibitively slow and difficult to tune (e.g. Betancourt et al., 2015). For example, in the case of a linear regression with n observations and k predictors, evaluation of the posterior density requires $O(nk)$ operations with straightforward implementation. To make matters worse, MCMC methods require large numbers of evaluations of the posterior density, and in the case of HMC, the posterior's gradient.

Alternative methods for inference have been proposed for problems where MCMC is impractical. These approaches typically involve a suitable approximation of the posterior density with a function with desirable properties. Laplace approximation (e.g. Margossian et al., 2020) and variational inference (Blei et al., 2017) are two examples. More generally, there is extensive literature on efficient computational tools and analysis of posterior densities, and there are various software packages devoted to their implementation (see, e.g. Rue et al., 2017; Kristensen et al., 2016).

While these packages, and indeed most of the literature, are devoted to general tools for a wide range of posterior densities, in this paper we introduce an efficient algorithm for computing posterior expectations for two particular classes of Bayesian regression models—two-group normal-normal models and mixed-effects models. These classes of models find a broad range of applications in, for example, social sciences, epidemiology, biochemistry, and environmental sciences (Gelman et al., 2013; Gelman and Hill, 2006; Greenland, 2000; Merlo et al., 2005; Bardini et al., 2017). Furthermore, in the broader context of statistical workflow, these regressions can serve as template models (Gelman et al., 2020).

Using general MCMC methods for sampling from these posteriors can be exceedingly slow for problems with large amounts of data. By specializing on this particular family of models, we leverage their structure to create customized algorithms for fast and accurate inference. We provide a publicly available implementation of the algorithms in R.

The two Bayesian linear regression models we consider are:

1. **Two group normal-normal:** We define the two-group normal-normal model by

$$\begin{aligned} y_i &\sim \mathrm{N}(X_1\beta_1 + X_2\beta_2, \sigma_y^2 I), \\ \beta_1 &\sim \mathrm{N}(0, \sigma_1^2 I), \\ \beta_2 &\sim \mathrm{N}(0, \sigma_2^2 I), \end{aligned} \tag{1.1}$$

   where X_1 is an $n \times k_1$ data matrix, $\beta_1 \in \mathbb{R}^{k_1}$ is a vector of regression coefficients, X_2 is an $n \times k_2$ data matrix, and $\beta_2 \in \mathbb{R}^{k_2}$ is a vector of regression coefficients. For Bayesian inference, we assume priors on the scale parameters $\sigma_y, \sigma_1, \sigma_2$. The performance of the algorithm is largely independent of the choice of these priors. In the models that we use in this paper, we assign independent weakly informative normal$^+(0,1)$ priors on $\sigma_y, \sigma_1, \sigma_2$ (assuming y and the columns of X have been normalized to have standard deviation 1). The normal$^+(0,1)$ distribution denotes the standard normal restricted to the non-negative reals.

2. **Mixed effects:** The mixed-effects model differs slightly from the two-group normal-normal model. Instead of modeling the scale parameter σ_2, fixed scale parameters are assigned to the normal priors on β_2. The mixed-effects model is defined by

$$\begin{aligned} y &\sim \mathrm{N}(X_1\beta_1 + X_2\beta_2, \sigma_y^2 I), \\ \beta_1 &\sim \mathrm{N}(0, \sigma_1^2 I), \\ \beta_{2,i} &\sim \mathrm{N}(0, \sigma_{2,i}^2), \end{aligned} \tag{1.2}$$

where $\sigma_{2,i}$ is the fixed scale parameter prior on each regression coefficient $\beta_{2,i}$ for $i = 1, \dots, k_2$ where $\beta_2 \in \mathbb{R}^{k_2}$. We will assume priors on the scale parameters σ_y, σ_1.

The models we discuss in this paper are standard models of Bayesian statistics and appear when seeking to model an outcome, y, as a linear combination of two (or more) distinct groups of predictors. In our notation, the data matrices of the two groups of predictors are X_1, X_2 with corresponding regression coefficients β_1, β_2. The Gaussian prior on the predictors enable various strategies commonly used in statistical modeling and machine learning; notably regularization and partial pooling between various sources of data. We demonstrate these models on three applications.

1. **COVID-19:** Due to a lack of reliable, fast, and widespread testing, an online survey initiative was created in Israel (Rossman et al., 2020) for tracking and predicting outbreaks of Coronavirus disease 2019 (COVID-19). We constructed a mixed-effects model for estimating geographic and age effects on the spread of the virus. With tens of thousands of responses, straightforward implementation of MCMC methods takes hours. Using the methods of this paper, we obtain accurate posterior inference in seconds.

2. **Rat growth:** We demonstrate the efficiency of our two-group algorithm on the classical two-group model for rat growth (Gelfand et al., 1990), which estimates the growth rates of a population of rats over the first few weeks of life.

3. **Public opinion on abortion:** We use 2018 results of the annual Cooperative Congressional Election Study (CCES) to estimate geographic and demographic effects on attitudes towards abortion. The CCES contains nearly 100,000 responses, and performing inference via MCMC sampling can be prohibitively slow. We use the mixed-effects algorithm introduced in this paper to perform posterior inference in seconds.

The computational methods we introduce for the two-group normal-normal model and the mixed-effects models are closely related. In fact, the mixed-effects model is, from a computational standpoint, a special case of the two-group model. We organize this paper by first describing an overview of the algorithms used in both the two-group normal-normal and mixed-effects models. In supplemental material, we provide a full description of both algorithms. We start by deriving the two-group normal-normal algorithm in detail, and then outline the minor modifications that allow for efficient evaluation of posterior moments of mixed-effects models.

The unnormalized density corresponding to the two-group model is given by

$$q(\beta, \sigma_1, \sigma_2, \sigma_y) = \frac{p(\sigma_1, \sigma_2, \sigma_y)}{\sigma_y^n \sigma_1^{k_1} \sigma_2^{k_2}} e^{-\frac{1}{2\sigma_y^2}\|X\beta - y\|^2} e^{-\frac{1}{2\sigma_1^2}\|\beta_1\|^2} e^{-\frac{1}{2\sigma_2^2}\|\beta_2\|^2}, \tag{1.3}$$

where $p(\sigma_1, \sigma_2, \sigma_y)$ denotes a prior probability density function for $(\sigma_1, \sigma_2, \sigma_y)$ and $\beta = (\beta_1, \beta_2)$ with $\beta_1 \in \mathbb{R}^{k_1}, \beta_2 \in \mathbb{R}^{k_2}, \beta \in \mathbb{R}^k$, and $y \in \mathbb{R}^n$. For convenience, we denote by σ the vector of scale parameters $(\sigma_y, \sigma_1, \sigma_2) \in \mathbb{R}^3$.

In the methods of this paper, we obtain high-order approximations of the posterior moments of β, σ by first analytically reducing the calculation of moments from integrals over $k + 3$ dimensions to 3-dimensional integrals. The 3-dimensional integrals are then approximated with spectral quadrature rules. The source of approximation of these methods is entirely in the quadrature and errors decay super-algebraically in the number of quadrature nodes. The total computational cost of evaluation of posterior means is $O(mk^3 + m^2k + m^3 + nk^2)$ operations, while posterior covariance requires $O(mk^3 + m^2k^2 + m^3 + nk^2)$ operations where n is the number of observations, k is the number of predictors, and m is the number of quadrature nodes in each dimension of the numerical integration. These computational costs assume n is larger than k.

The tools we use are a generalization of the approach proposed by Greengard et al. (2021) and generalize to higher-dimensional multilevel and higher-dimensional multi-group posterior distributions. Since we integrate the marginal density using a tensor product of Gaussian nodes, the cost of the integration scales like $O(m^d)$ where m is the number of discretization nodes in each direction and d is the dimension of the marginalized integral (where $d = 3$ in the models of this paper). As a result, higher dimensional problems require evaluation of marginal integrals via sampling-based algorithms and cannot rely solely on Gaussian quadrature. We leave the analysis and description of numerical tools for such models to a subsequent publication.

For the applications we consider in this paper, we assign half-normal priors (normal distributions restricted to the positive reals) to the scale parameters σ. This choice of prior is unrelated to the computational costs of our algorithm. After analytically integrating the regression coefficients, numerical integration via quadrature is performed on the remaining low-dimensional density. That density contains a multiplicative factor of $p(\sigma)$ where p denotes the prior on scale parameters σ. In that sense, the only conditions on p are that p is not highly singular and can be evaluated relatively cheaply. Both of these conditions are easily met by any reasonable choice of p.

We implemented the two-group normal-normal and mixed effects algorithms of this paper in an R package `fastNoNo`, which is publicly available on GitHub at `https://github.com/pgree/fastNoNo`.

The structure of this paper is as follows. In the following section we provide background on the quadrature rules we use for the numerical integration stage of the algorithms. In Section 3 we provide a summary of the numerical methods used in this paper as well as intuition behind the computational efficiency. In Section 5, Section 4 and Section 6 we apply the algorithms of this paper to applications. Conclusions and generalizations of the algorithm of this paper are presented in Section 7. Lastly, the supplemental material includes a detailed description of our algorithms as well as numerical implementation details.

2 Mathematical preliminaries

The algorithms of this paper rely heavily on numerical integration using Gaussian quadrature (or Gauss-Legendre quadrature). In this section, we provide a brief overview of Legendre polynomials and Gaussian quadrature (Abramowitz and Stegun, 1964) that will be used throughout the remainder of the paper. The contents of this section are well-known and a more in-depth discussion can be found in, for example, a book by Trefethen (2020).

In accordance with standard practice, we denote by $P_i : [-1, 1] \to \mathbb{R}$ the i-th Legendre polynomial for all $i = 0, 1, \ldots$. Legendre polynomials satisfy the three-term recursion

$$P_{i+1}(x) = \frac{2i+1}{i+1}\, x\, P_i(x) - \frac{i}{i+1} P_{i-1}(x)$$

with

$$\begin{aligned} P_0(x) &= 1, \\ P_1(x) &= x. \end{aligned}$$

Legendre polynomials are orthogonal with respect to $L^2[-1, 1]$. That is, for all $i, j = 0, 1, \ldots$

$$\int_{-1}^{1} P_i(x)\, P_j(x)\, dx = \begin{cases} 0 & i \neq j, \\ \frac{2}{2i+1} & i = j. \end{cases}$$

For all n, P_n has n unique roots which we denote $x_1, \ldots, x_n$. Additionally, for all n, there exist n positive reals $w_1, \ldots, w_n$ such that for any polynomial p of degree $\leq 2n-1$,

$$\int_{-1}^{1} p(x)dx = \sum_{i=1}^{n} w_i p(x_i).$$

The roots, $x_1, \ldots, x_n$ are known as the order-n Gaussian quadrature nodes (or Gaussian nodes) and $w_1, \ldots, w_n$ are known as Gaussian weights. Efficient algorithms for calculating Gaussian nodes and weights can be found in standard software packages such as Chebfun (Driscoll et al., 2014).

In dimensions more than one, integrals can be approximated using tensor-product Gaussian quadrature rules. Specifically, suppose that $f(x, y)$ is a real-valued function defined on the square, $[0,1]^2$. Then, a tensor-product Gaussian quadrature rule is used to obtain the approximation

$$\sum_{i=1}^{n} \sum_{i=1}^{n} f(x_i, x_j) w_i w_j \approx \int_0^1 \int_0^1 f(x, y) dx dy. \tag{2.1}$$

In the algorithms of this paper, we use Gaussian quadrature in large part due to its desirable convergence properties. In particular, for any smooth function, $f : [a, b] \to \mathbb{R}$, the error of the order-n Gaussian quadrature approximation

$$\left| \int_a^b f(x) dx - \sum_{i=1}^{n} f(x_i) w_i \right|$$

decays super-algebraically, that is, faster than $O(n^{-j})$ for any $j \in \mathbb{N}$ (naturally the constant on the decay rate grows as a function of j). Since f is defined on $[a, b]$, Gaussian nodes and weights must be appropriately shifted and scaled.

As with functions defined on $\mathbb{R}$, for smooth functions defined on a region of $\mathbb{R}^d$, tensor-product quadratures also possess super-algebraic accuracy – errors decay faster than $O(n^{-j/d})$ for any $j \in \mathbb{N}$ where again, the constant on the decay rate grows with d and j. Broadly, suppose that the integral of a function $f : [a, b] \to \mathbb{R}$, requires an n-degree Gaussian quadrature for some desired level of accuracy. Then for a function $g : [a, b]^d \to \mathbb{R}$ with similar smoothness properties to f, approximating its integral with tensor-product Gaussian quadrature will require roughly n^d points to achieve the same level of accuracy.

3 Overview of algorithm

Greengard et al. (2021) introduced a numerical method for computing with posterior unnormalized densities such as the one-group Bayesian regression model posterior

$$q_0(\beta, \sigma_1, \sigma_y) = \frac{p(\sigma_y, \sigma_1)}{\sigma_y^n \sigma_1^k} e^{-\frac{1}{2\sigma_y^2}\|X\beta - y\|^2} e^{-\frac{1}{2\sigma_1^2}\|\beta\|^2}, \tag{3.1}$$

where $p(\sigma_y, \sigma_1)$ denotes a prior probability density function on σ_y, σ_1. The approach of Greengard et al. (2021) uses a change of variables in β to the singular vectors of the data matrix X, that results in the conditional densities $q_0(\beta \,|\, \sigma_y, \sigma_1)$ being Gaussian with diagonal covariance, for all $\sigma_y, \sigma_1 > 0$. Moments of $\beta, \sigma_y, \sigma_1$ with respect to q_0 can then be computed via numerical integration with minimal work.

Unfortunately, for the class of models we consider in this paper, the approach of Greengard et al. (2021) cannot be directly applied – there is no change of variables over β such that the Gaussian $q(\beta \,|\, \sigma)$ has diagonal covariance for all σ. For the remainder of this section, we summarize the strategy used by the class of algorithms of this paper and provide intuition for the computational speed-ups obtained. In the supplemental material, we include a detailed derivation of the algorithms and associated numerical implementation for computing moments of β and σ with respect to q. The algorithms described in the supplemental material, for two-group normal-normal models and mixed effects models, are closely related. They both rely on a change of variables that facilitates an analytical integration of regression coefficients followed by numerical integration of the remaining low-dimensional density.

In order to summarize the algorithm, we demonstrate its use in evaluation of the normalizing constant of q (see (1.3)). Evaluating the first and second moments is straightforward using the same approach as the one outlined in this section.

The normalizing constant of q, which we denote by C, is defined by

$$C = \int_{\mathbb{R}^+} \int_{\mathbb{R}^+} \int_{\mathbb{R}^+} \int_{\mathbb{R}^k} q(\beta, \sigma_y, \sigma_1, \sigma_2) d\beta \, d\sigma_y \, d\sigma_1 \, d\sigma_2.$$

It is well-known that due to conjugacy of normal-normal models the inner integral can be evaluated in $O(k^3)$ operations after $O(nk^2)$ precomputation. This can be done in various equivalent ways, for example, using the determinants of Lindley and Smith (1972) or the singular value decomposition of Greengard et al. (2021). Since the innermost integral of C is smooth and low-dimensional, one natural approach for evaluating C is to compute the outer integrals, those with respect to σ, with quadrature (e.g. see (2.1)), that is, a sum of the form

$$C \approx \sum_{i}^{m} \sum_{j}^{m} \sum_{\ell}^{m} \int_{\mathbb{R}^k} q(\beta, \sigma_{y,i}, \sigma_{1,j}, \sigma_{2,\ell}) d\beta \, w_i w_j w_\ell, \tag{3.2}$$

where $\sigma_i, w_i \in \mathbb{R}^+$ for $i = 1, \ldots, m$. Such an approximation would result in a computational cost of $O(m^3 k^3)$ operations ($O(m^3)$ evaluations of a function that requires $O(k^3)$ operations to evaluate). For many modern problems, with large numbers of predictors (large k), this cost can result in prohibitively slow inference.

In order to improve the computational burden of this approach, we start with a change of variables of the scale parameters $\sigma_y, \sigma_1, \sigma_2$ to spherical coordinates:

$$C = \int_0^{\pi/2} \int_0^{\pi/2} \int_0^{\infty} \int_{\mathbb{R}^k} f(\beta, \rho, \theta, \phi) \rho^2 \sin(\phi) d\beta \, d\rho \, d\phi \, d\theta, \tag{3.3}$$

where $f : \mathbb{R} \times \mathbb{R}^+ \times (0, \pi/2)^2 \to \mathbb{R}$ is defined by

$$f(\beta, \rho, \phi, \theta) = \frac{e^{-\rho^2/2}}{\rho^{n+k} \cos^n(\phi) \sin^k \phi \cos^{k_1} \theta \sin^{k_2} \theta}$$
$$\exp\left[-\frac{1}{2\rho^2} \left(\frac{1}{\cos^2 \phi} \|X(\beta - \tilde{\beta})\|^2 + \frac{\|d\|^2}{\cos^2 \phi} + \frac{\beta^t \left(\frac{I_1}{\cos^2 \theta} + \frac{I_2}{\sin^2 \theta} \right) \beta}{\sin^2 \phi} \right) \right],$$

where $\tilde{\beta} \in \mathbb{R}^k$ is computed once in precomputation in $O(k^3)$ operations, $d = X\beta - y$, and I_1, I_2 are identity matrices.

The new conditional density $f(\beta \,|\, \rho, \phi, \theta)$ is still Gaussian in β, but now has a covariance matrix that depends on ρ only up to a multiplicative constant. That is, the covariance can be expressed as $\frac{1}{\rho^2}\Sigma(\theta,\phi)$ where $\Sigma(\theta,\phi)$ is a covariance matrix that depends only on θ,ϕ, and X. Using this property and identities of Section 2 of the supplemental material (Greengard et al., 2022), the normalizing constant (and moments) of $f(\beta\,|\,\rho,\phi,\theta)$ can be efficiently calculated in $O(1)$ operations given the normalizing constant of $f(\beta\,|\,\phi,\theta)$.

Using a quadrature-based scheme, this reduces the number of times the $O(k^3)$ cost of evaluating the inner integral of (3.3) needs to be performed. Now, instead of being performed over a three dimensional space, (θ,ϕ,ρ), the $O(k^3)$ operations needs to be performed over a 2-dimensional space (θ,ϕ). This reduces the cost of the quadrature approach of (3.2) from $O(m^3k^3)$ to $O(m^2k^3+m^3)$ operations.

We next perform a further change of variables of the regression coefficients, $w = M_\theta\beta$, where M_θ is a $k\times k$ matrix that depends only on θ (see supplemental material for details on M_θ). The matrix M_θ requires $O(k^3)$ operations to evaluate and converts the Gaussian $f(\beta\,|\,\rho,\phi,\theta)$ to a form that allows for rapid updating of the normalizing constant (and moments) for different values of ϕ and ρ. Specifically, after the change of variables $w = M_\theta\beta$, we obtain

$$\begin{aligned} C = \int_0^{\pi/2}\int_0^{\pi/2}\int_0^{\infty}\int_{\mathbb{R}^k} & \frac{e^{-\frac{\rho^2}{2}-\frac{\|d\|^2}{2\rho^2\cos^2\phi}}}{\rho^{n+k}\cos^n\phi\,\sin^k\phi} \\ & \exp\left[-\frac{1}{2\rho^2}\sum_{i=1}^k\left(\frac{\lambda_i(w_i-\tilde{w}_i)^2}{\cos^2\phi}+\frac{w_i^2}{\sin^2\phi}\right)\right]\,\mathrm{d}w\,d\rho\,d\phi\,d\theta. \end{aligned}$$

The inner integral (in addition to moments of w) are available using standard identities (e.g. Abramowitz and Stegun (1964)). In particular, we have the following formula for the normalizing constant.

$$C = \int_0^{\frac{\pi}{2}}\int_0^{\frac{\pi}{2}}\frac{\alpha(\phi,\theta)\sin\phi}{\cos^{n-k}\phi}\int_0^{\infty}\frac{e^{-\frac{\rho^2}{2}-\frac{1}{2\rho^2}\left(\frac{\|d\|^2}{\cos^2\phi}+\beta(\phi,\theta)\right)}}{\rho^n}\rho^2\,\mathrm{d}\rho\,\mathrm{d}\phi\,\mathrm{d}\theta,$$

where $\alpha:(0,\pi/2)^2\to\mathbb{R}^+$ and $\beta:(0,\pi/2)^2\to\mathbb{R}^+$ require $O(k)$ operations to compute and are defined in Section 2 of the supplemental material (Greengard et al., 2022). Now, the inner integrals of C,

$$\int_0^{\frac{\pi}{2}}\frac{\alpha(\phi,\theta)\sin\phi}{\cos^{n-k}\phi}\int_0^{\infty}\frac{e^{-\frac{\rho^2}{2}-\frac{1}{2\rho^2}\left(\frac{\|d\|^2}{\cos^2\phi}+\beta(\phi,\theta)\right)}}{\rho^n}\rho^2\,\mathrm{d}\rho\,\mathrm{d}\phi,$$

can be evaluated in $O(k^2)$ operations using Gaussian quadrature.

Finally, we now evaluate C using a 3-dimensional tensor product of Gaussian nodes. That is, we compute C with a sum of the form

$$C = \sum_i^m\sum_j^m\frac{\alpha(\phi_j,\theta_i)\sin\phi_j}{\cos^{n-k}\phi_j}\sum_\ell^m e^{-\frac{\rho_\ell^2}{2}-\frac{1}{2\rho_\ell^2}\left(\frac{\|d\|^2}{\cos^2\phi_j}+\beta(\phi_j,\theta_i)\right)}\rho_\ell^{2-n}w_{\rho,\ell}w_{\phi,j}w_{\theta,i}, \tag{3.4}$$

where $\theta_i, \phi_j, \rho_\ell$ and $w_{\rho,\ell}, w_{\phi,j}, w_{\theta,i}$ are appropriately scaled and shifted Gaussian nodes and weights. We've reduced the approximation of C from a sum requiring $O(k^3m^3)$ operations to a sum that requires $O(mk^3 + m^3)$ operations – the $O(mk^3)$ cost arising from the $O(k^3)$ construction of M_θ for m different values of θ, while $O(m^3)$ is required to sum each term in the tensor-product quadrature.

The change of variables described in this section allows for efficient evaluation of posterior means and covariances of β and σ in addition to the normalizing constant. We leave details to the supplemental material.

The supplemental material (Greengard et al., 2022) includes a detailed description of the algorithms of this paper and associated numerical implementation. We collectively refer to this class of algorithms as "fastNoNo." Additionally, we have included a publicly available code in R of the two-group normal-normal and mixed effects algorithms, which can be found at `https://github.com/pgree/fastNoNo` or installed from the command line in RStudio via `devtools::install_github(‘‘pgree/fastNoNo’’)`.

4 A simple example: Hierarchical linear model

We demonstrate the two-group normal-normal algorithm on a hierarchical linear model describing the growth of a group of young rats over a period of several weeks; this is a small example that has been used in the statistical literature (Gelfand et al., 1990). In the experiment, the weight of each rat is measured at regular time intervals. Regression coefficients are computed for each rat; that is, for the j^{th} rat, we estimate an intercept α_j and a linear coefficient β_j. We assign a normal prior on both parameters and estimate the prior scale. The full model is as follows:

$$\begin{aligned} y_i &\sim \mathrm{N}(X_1^i\alpha + X_2^i\beta, \sigma_y^2), \\ \alpha_j &\sim \mathrm{N}(0, \sigma_1^2), \\ \beta_j &\sim \mathrm{N}(0, \sigma_2^2), \\ \sigma_y &\sim \mathrm{N}^+(0, 10^2), \\ \sigma_k &\sim \mathrm{N}^+(0, 10^2) \text{ for } k = 1, 2, \end{aligned} \tag{4.1}$$

where X_1 is an indicator matrix indicating to which rat each observation (weighing) corresponds; X_2 is that same indicator matrix multiplied by $w - \bar{w}$, where w is the observation week and $\bar{w}$ the mean observation week. In other words, we have an intercept and a slope parameter for each rat. The data is centered at 0 and the priors on the scale parameters are weakly informative.

We demonstrate the efficiency of the two-group normal-normal algorithm on evaluating posterior means and standard deviations of the rats model on simulated data. We assume an experiment with 100 rats and 20 weighing times and randomly generated data for each weighing. As a result, matrix X_1 and X_2 of model (4.1) are 2000×100 matrices.

Because the data size is relatively small and the data matrices have a friendly, sparse structure, running MCMC with Stan (4 chains in parallel, each with 1,000 warmup it-

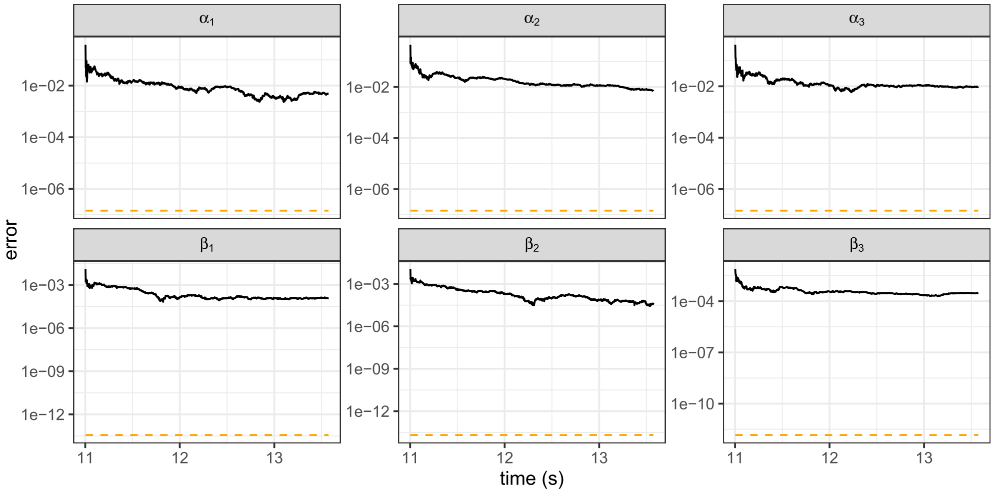

Figure 1: Absolute error of MCMC estimates via Stan as a function of run time. The horizontal orange line is the absolute error of fastNoNo.

erations and 1,000 sampling iterations) only takes 13.9s. This timing reflects a Stan implementation that takes advantage of the sparsity of the data matrices. Our algorithm takes 1.6s and achieves significantly smaller errors than MCMC estimates. For problems with larger data, the difference in time scale becomes important. Figure 1 shows the error of posterior mean approximations via MCMC in Stan as a function of time as well as the error achieved by our algorithm. Errors are defined to be the absolute difference between the true posterior mean and the approximation. Accurate approximations of true posterior moments were obtained via our algorithm with a large number of quadrature nodes.

5 Application: COVID-19 symptom survey

As of the writing of this article, the coronavirus pandemic is still raging in many countries and stressing healthcare systems around the world. A challenge at the start of the pandemic was tracking its spread, especially in locations where reliable testing was not widely available. Having accurate estimates of infection rates across geographical regions can be extremely helpful. For example, reliable estimates allow hospital systems to allocate resources efficiently, they can alert residents of the need to take extra precaution in their daily routines, and they can facilitate better policy from local governments. In order to get improved estimates of infection rates in the absence of widespread testing, initiatives were deployed in early 2020 in several countries that allowed individuals to report symptoms via publicly available surveys (e.g., Segal et al., 2020).

One country where these surveys provided valuable information was Israel (Rossman et al., 2020), where demographic and health data was provided by tens of thousands of respondents across the country. The large amount of data collected from survey respondents provided data scientists and policy-makers with a great resource, however at the same time, large amounts of data turns the computational aspect of statistical modeling into a substantial challenge.

In this section, we present an exploratory model used to analyze data from the COVID-19 survey conducted in Israel (Rossman et al., 2020). Using straightforward MCMC with Stan (Carpenter et al., 2017) was inconvenient; using the full data set resulted in run times of several hours. Using the algorithms of this paper, we were able to evaluate posterior moments to high accuracy in seconds.

Multilevel regression and poststratification procedure

The respondents are anonymous, but several of their features are recorded, including their age and the city in which they live. We can use the data to identify regions in which the average symptom score seems unusually high.

A first exploratory model uses an intercept, age group, and population density in the respondent's city, as covariates, X, and an indicator matrix Z for city:

$$y \sim \mathrm{N}(X\beta + Zu, \sigma_y^2 I),$$

with a hierarchical prior on the city parameters,

$$u \sim \mathrm{N}(0, \sigma_1 I),$$

and weakly informative priors on the other coefficients,

$$\beta \sim \mathrm{N}(0, I).$$

This unit prior is weakly informative if the outcome y_i has been standardized and the continuous predictors (in this case, population density) has also been standardized to be on unit scale.

In addition, we put weakly informative half-normal priors (standard normal distributions restricted to the non-negative reals) on the hyperparameters σ_y and σ_1:

$$\begin{aligned} \sigma_y &\sim \mathrm{N}^+(0,1), \\ \sigma_1 &\sim \mathrm{N}^+(0,1). \end{aligned}$$

This corresponds to a two-group normal-normal model with an additional covariate. In cities where u cannot be well estimated due to a low response rate, we can rely on the rest of the model, that is a regression model based on age and population density.

Only a fraction of the population responds to the survey, which raises questions about biases. This is notably a concern because different age groups behave differently: not only do their chances of contracting and spreading the disease vary, their susceptibility to the disease also changes. In multilevel regression and poststratification (MRP), we adjust for these biases by using estimates of the proportion of people in each city that belong to each age group. For this model, the proportions are estimated using census data. This leads to a corrected estimate for the expected symptom score of an individual in city i:

$$\tilde{u}_i = u_i + \beta_0 + \beta_{\text{density}} d_i + \sum_{j=1}^{n} a_j^i \beta_{\text{age},j},$$

where β_0 is the intercept, $\beta_{\text{density},i}$ is the regression coefficient of the population density covariate, d_i denotes the density of city i, a_j^i is the proportion of individuals in the j^{th} age group in the i^{th} city, and $\beta_{\text{age},j}$ is the regression coefficient of age group j.

Using the means and covariances of $u, \beta_0, \beta_{\text{density}}$, and β_{age} we compute the posterior mean and variance for $\tilde{u}$, per the following formulas. Given a linear combination of random variables, $Y = \sum_i \delta_i Z_i$, we have

$$EY = \sum_i \delta_i E Z_i,$$

and

$$\text{Var} Y = \sum_i \delta_i^2 \text{Var} Z_i + 2 \sum_{i<j} \delta_i \delta_j \text{Cov}(Z_i, Z_j).$$

Moreover, variance and thence standard deviations of $\tilde{u}$ can be computed, provided we also evaluate the relevant posterior covariances.

Comparison of our algorithm to MCMC

We analyze the data collected over the two weeks between April 15th and 30th, 2020, across 351 cities. These are cities for which we know, through census data, the population density and the age distribution. The total number of responses is 135,501.

Our proposed algorithm returns the posterior mean and standard deviation for all variables of interest and takes ∼7s to run.

We next fit the model in Stan using the default dynamic HMC sampler exploiting the sparsity of the data matrices for efficient sampling. After warming up the sampler for 500 iterations, we compute another 500 draws, using 4 chains computed in parallel, for a total of 2,000 sampling iterations. The wall time for this procedure is ∼12,000s (>3 hours). For each city, we computed the Monte Carlo mean. Figure 2 plots the posterior mean and standard deviation of $\tilde{u}$ for all cities, computed by both methods. Figure 3 shows the difference between our algorithm and the Monte Carlo estimate, as a function of computation time. While it takes on the order of hours to get accurate results with MCMC, our algorithm achieves better results within seconds. Errors presented are absolute differences between the true posterior means and approximations using MCMC and fastNoNo. As a benchmark, we use fastNoNo with a large number of quadrature nodes. Table 1 provides the accuracy of posterior mean estimates for several regression coefficients using both fastNoNo and MCMC.

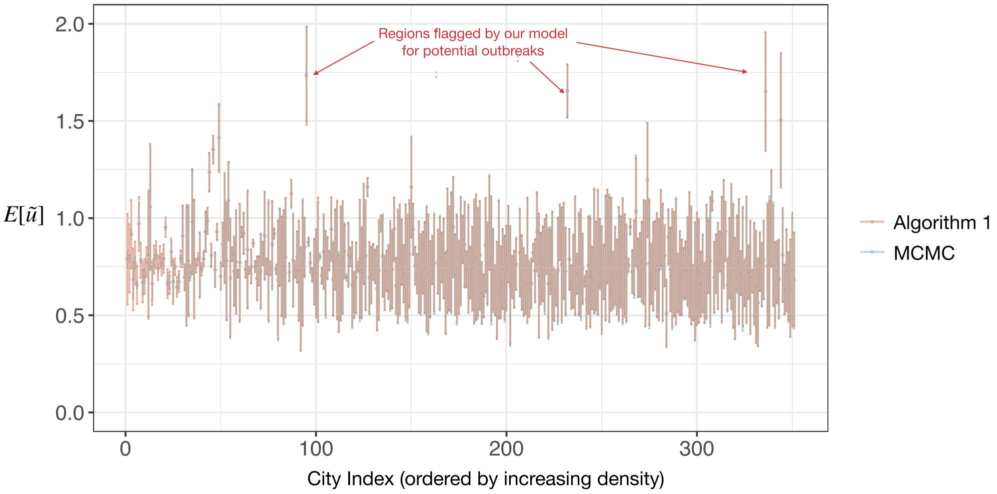

Figure 2: Posterior mean and standard deviation for $\tilde{u}$ computed using Algorithm 1 and MCMC. The points represent the estimated mean and the "error bars" span two standard deviations.

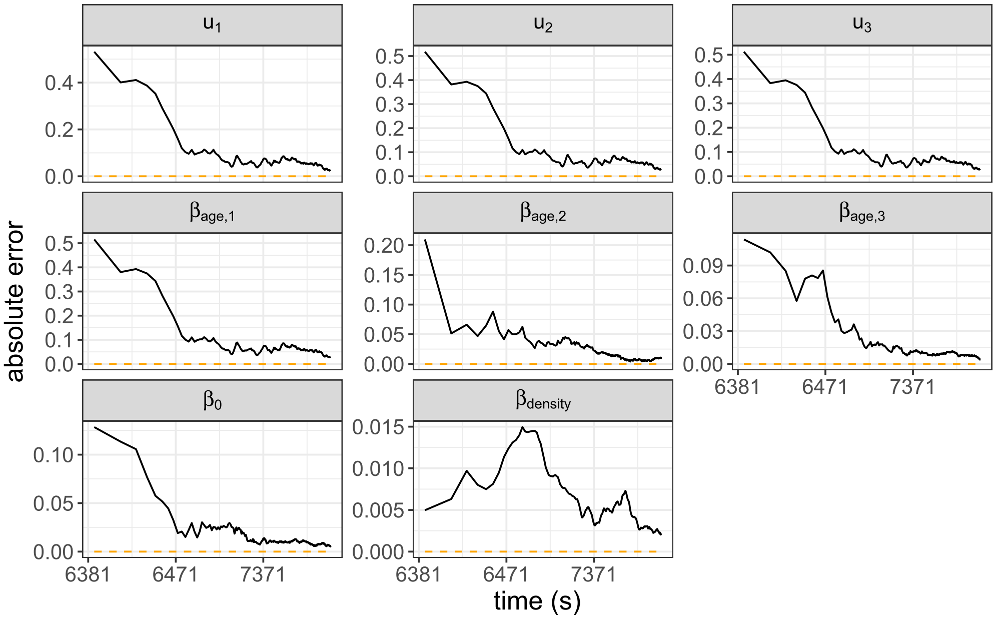

Figure 3: Absolute error of MCMC estimates via Stan as a function of run time. The horizontal orange line is the absolute error of our method, which took 7 seconds to run. MCMC in Stan required over 6,000 seconds of warmup before error can be measured. Total run time for Stan was ∼12,000 seconds.

Regression coefficient	MCMC Error (∼12,000 s)	fastNoNo Error (7 s)
β_0	3e-2	1e-5
u_1	1e-2	4e-4
u_2	1e-2	3e-4
u_3	2e-2	6e-5
$\beta_{\text{age},1}$	3e-2	7e-6
$\beta_{\text{age},2}$	3e-2	8e-6
$\beta_{\text{age},3}$	3e-2	9e-8
β_{density}	2e-3	2e-5

Table 1: Absolute error of the approximation of posterior means for several regression coefficients with both MCMC and fastNoNo. For MCMC, the total time for the approximation was ∼12,000s. Total time for fastNoNo was 7s.

Limitations of the model and our numerical method

We believe the presented model offers an improvement on the analysis conducted on the survey data (Rossman et al., 2020), because (i) it uses full Bayesian inference to quantify uncertainty and (ii) it corrects sampling biases using a poststratification step. A more careful quantification of uncertainty would use posterior intervals, rather than posterior variance. Such an interval can be estimated using MCMC draws. Extending our numerical scheme into a sampling scheme to estimate such intervals is a direction we are actively pursuing.

For the model of this paper we only used a fraction of the available covariates, that is, the data collected in survey responses. As a result, the model can be extended to include more than two groups. Estimates tend to be noisy because the studied covariates can be strongly correlated with the outcome. For example, age is correlated with intensity of symptoms. The marginal correlation however is weak. This, and other considerations, suggest that it might be beneficial from a modeling standpoint to build a more sophisticated model, which might be outside of the scope of application of the methods of this paper. Nevertheless, the model considered here is an important step in the development of a better model.

6 Application: Public opinion on abortion policies

We next apply our method to a hierarchical linear regression used to model attitudes on abortion policies as they vary across states, ethnicity, age groups, and education levels. Modeling this heterogeneity requires partitioning an initially large data set into small groups. Furthermore, we must address biases that can arise in our survey and correct them using more comprehensive surveys, such as census data. As in Section 5, we use MRP to do inference for small slices of big data and correct biases in our survey.

We analyze data from the 2018 Cooperative Congressional Election Study (CCES) using, as in the case study of (Lopez-Martin et al., 2022), a random subset of 5,000 respondents. Respondents express support or opposition on six abortion policies, for example "Ban abortion after the 20th week of pregnancy" or "Allow employers to decline coverage of abortion in insurance plan." These policies are intended to restrict access to abortion. Each respondent is given a support score, y, ranging from 0 to 6, indicating the number of supported policies.

We use a normal likelihood with the following covariates, recorded for each respondent: state, ethnicity, age group, education level, and sex. We use the proportion of votes for the Republican party in the state in 2016 as an additional predictor, denoted as `repvote`. The model also admits an intercept term. The statistical formulation of the model is the following:

$$\begin{aligned} y_i \quad \sim \quad & \mathrm{N}(\beta_0 + X_i^{\text{state}}\beta_{\text{state}} + X_i^{\text{ethnicity}}\beta_{\text{ethnicity}} + X^{\text{age}}\beta_{\text{age}} \\ & \quad + X^{\text{sex}}\beta_{\text{sex}} + X^{\text{education}}\beta_{\text{education}} + X^{\text{repvote}}\beta_{\text{repvote}}, \sigma_y^2). \end{aligned}$$

The difficult parameters to estimate here are the state coefficients, to which we give normal$(0, \sigma_1)$ priors. Because the model includes `repvote`, the partial pooling is done toward the prediction of the state based on its previous vote, not toward the national mean.

Table 2 summarizes the performance of the algorithm on this model. The posterior mean and standard deviation of the MRP estimates for each state can be computed as in Section 5 and are plotted in Figure 4.

n	k_1	k_2	max error	total time (s)
5000	50	19	1.2×10^{-8}	0.05

Table 2: Computation time and accuracy of fastNoNo applied to a model of support/opposition for abortion policies. The column "max error" shows the maximum error of posterior means and standard deviations of regression coefficients and scale parameters.

Figure 4 shows that the expected support score increases with the level of support for the Republican party, barring some fluctuations. The large posterior standard deviations indicate there is quite a bit of heterogeneity within each state. For further insight, we may examine how groups other than states, e.g. ethnic groups, age groups, etc. behave.

The present model has certain limitations. First, one could consider interaction terms. This seems sensible since, for instance, white males with no college education likely behave differently than white males with a college degree. The numerical method presented in this paper can handle interaction terms. Computing the posterior standard deviation of the MRP estimate however requires some data wrangling. We plan to create an R package with routines that seamlessly implement these MRP calculations, making it straightforward for modelers to experiment with different covariates and interaction terms.

There is also interest in nonlinear models with non-normal likelihoods. Lopez-Martin et al. (2022) consider an item-response or ideal-point logistic regression. This sort of model can better capture certain characteristics of the data, such as dependence among different survey responses. For such models, we cannot use the proposed integration scheme. This presents us with a tradeoff: the proposed algorithm takes a fraction of a

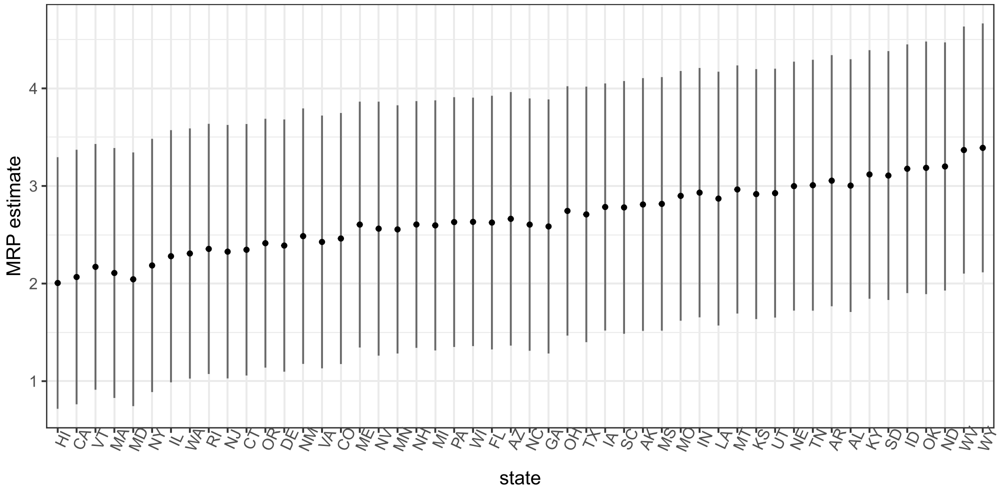

Figure 4: MRP estimate of the expected level of support for anti-abortion policies in each state. The point represents the posterior mean, and the bars span two posterior standard deviations. The states are ordered based on Republican vote share in the 2016 presidential election.

second to run, while fitting the ideal point model with Stan's MCMC takes hundreds of seconds (after exploiting sparsity of the data matrices). The difference is more severe if, rather than fitting a subset of 5,000 respondents, we use all 60,000 respondents in the survey. The modeler then needs to assess how useful it is to use a non-normal likelihood. Even then, the normal likelihood model can be a fast way to do model exploration, by for example examining various covariates and interaction terms.

7 Conclusions and generalizations

In this paper we describe a class of fast algorithms for evaluating the posterior moments of two Bayesian linear regression models, the two-group normal-normal model and the mixed effects model.

The algorithms of this paper allow for assigning a general choice of priors on the scale parameters. We demonstrated the performance of our algorithm for posterior inference on two applications. In Section 5 we used COVID-19 symptom survey data to model geographic and age effects. We also used the mixed-effects model with public opinion survey data to estimate geographic and demographic impacts on attitudes towards abortion. These are both existing applications that have been fit with MCMC; by allowing these models to be fit much faster, our algorithm can facilitate a workflow in which users can fit and explore many more models in real time.

The algorithms of this paper provide substantial improvements over standard MCMC methods in both computation time and accuracy in approximating posterior moments.

These improvements rely on analytically integrating the regression coefficients, which make up the bulk of the posterior dimensions, and then numerically integrating the remaining low-dimensional density with Gaussian quadrature.

Many of the techniques and analysis used in this paper generalize to multilevel and multigroup models with more than two-groups. For an m group model, the numerical integration of our algorithm is computed over a $m+1$ dimensional density, m scale parameters each corresponding to one group of predictors and the residual standard deviation. For models with large m (large number of groups) the analytic marginalization of this paper can still be applied, although integration via a tensor product of Gaussian nodes will not be feasible. On the other hand, using MCMC or other integration schemes can be used on the $m+1$ dimensional marginal density.

Another example of a natural extension of the models we consider in this paper are normal-normal models with priors on the mean of a group of regression coefficients. Consider, for example, the posterior unnormalized density q_μ defined by

$$q_\mu(\beta,\sigma_1,\sigma_2,\sigma_y,\mu) = \frac{p(\sigma_1,\sigma_2,\sigma_y,\mu)}{\sigma_y^n \sigma_1^{k_1} \sigma_2^{k_2}} e^{-\frac{1}{2\sigma_y^2}\|X\beta-y\|^2} e^{-\frac{1}{2\sigma_1^2}\|\beta_1\|^2} e^{-\frac{1}{2\sigma_2^2}\|\beta_2-\mu\|^2}. \tag{7.1}$$

This posterior differs from q (see (1.3)) in one respect. The regression coefficients β_2 are given a prior with non-zero mean, μ, which is itself given a prior. The numerical methods of this paper are applicable to this density with one modification. The low-dimensional density obtained after analytically integrating the regression coefficients will now be a 4-dimensional density (as opposed to 3-dimensional) over $\sigma_y, \sigma_1, \sigma_2, \mu$. As a result, computational costs of the quadrature stage of the algorithm will increase by a factor of m where m is the number of quadrature nodes in each direction. While quadrature of a 4-dimensional function can still be efficient, modeling further features of q_μ may call for a sampling-based approach.

Bayesian models such as q_μ, in addition to those with more than two groups and non-Gaussian likelihoods, are directions of future research.

Supplementary Material

Fast methods for posterior inference of two-group normal-normal models. Supplementary Material. (DOI: 10.1214/22-BA1329SUPP; .pdf).

References

Abramowitz, M. and Stegun, I. A. (eds.) (1964). *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables*. Washington: U.S. Govt. Print. Off. MR0167642. 893, 896

Bardini, R., Politano, G., Benso, A., and Di Carlo, S. (2017). "Multi-level and hybrid modelling approaches for systems biology." *Computational and Structural Biotechnology Journal*, 15: 396–402. 890

Betancourt, M. (2018). "A conceptual introduction to Hamiltonian Monte Carlo." 889

Betancourt, M., Byrne, S., and Girolami, M. (2015). "Optimizing the integrator step size for Hamiltonian Monte Carlo." *arXiv*, 1411.6669. MR3644666. 890

Blei, D. M., Kucukelbir, A., and McAuliffe, J. D. (2017). "Variational inference: A review for statisticians." *Journal of the American Statistical Association*, 112(518): 859–877. MR3671776. doi: https://doi.org/10.1080/01621459.2017.1285773. 890

Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M., Guo, J., Li, P., and Riddell, A. (2017). "Stan: A probabilistic programming language." *Journal of Statistical Software*, 76(1): 1–32. 889, 899

Driscoll, T. A., Hale, N., and Trefethen, L. N. (2014). *Chebfun Guide*. Pafnuty Publications. URL http://www.chebfun.org/docs/guide/. MR2960859. doi: https://doi.org/10.1007/s11425-012-4474-z. 893

Gelfand, A. E., Hills, S. E., Racine-Poon, A., and Smith, A. F. M. (1990). "Illustration of Bayesian inference in normal data Models Using Gibbs sampling." *Journal of the American Statistical Association*, 85(412): 972–985. URL http://www.jstor.org/stable/2289594. MR1825281. doi: https://doi.org/10.2307/2669775. 891, 897

Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., and Rubin, D. B. (2013). *Bayesian Data Analysis*. London: CRC Press, 3rd edition. MR3235677. 890

Gelman, A. and Hill, J. (2006). *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Cambridge University Press. 890

Gelman, A., Vehtari, A., Simpson, D., Margossian, C. C., Carpenter, B., Yao, Y., Kennedy, L., Gabry, J., Bürkner, P.-C., and Modrák, M. (2020). "Bayesian workflow." *arXiv*, 2011.01808. 890

Greengard, P., Gelman, A., and Vehtari, A. (2021). "A fast linear regression via SVD and marginalization." *Computational Statistics*. MR4400219. doi: https://doi.org/10.1007/s00180-021-01135-x. 892, 894, 895

Greengard, P., Hoskins, J., Margossian, C., Gabry, J., Gelman, A., and Vehtari, A. (2022). "Fast methods for posterior inference of two-group normal-normal models. Supplementary Material." *Bayesian Analysis.* doi: https://doi.org/10.1214/22-BA1329SUPP. 896, 897

Greenland, S. (2000). "Principles of multilevel modelling." *International Journal of Epidemiology*, 29(1): 158–167. doi: https://doi.org/10.1093/ije/29.1.158. 890

Hoffman, M. D. and Gelman, A. (2014). "The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo." *Journal of Machine Learning Research*, 15(47): 1593–1623. URL http://jmlr.org/papers/v15/hoffman14a.html. MR3214779. 889

Kristensen, K., Nielsen, A., Berg, C. W., Skaug, H., and Bell, B. M. (2016). "TMB: Automatic differentiation and Laplace approximation." *Journal of Statistical Software*, 70(5): 1–21. URL https://www.jstatsoft.org/v070/i05. 890

Lindley, D. and Smith, A. (1972). "Bayes estimates for the linear model." *Journal of the Royal Statistical Society. Series B (Methodological)*, 34(1): 1–41. URL http://www.jstor.org/stable/2985048. MR0415861. 895

Lopez-Martin, J., Phillips, J. H., and Gelman, A. (2022). "Multilevel regression and poststratification case studies." URL https://juanlopezmartin.github.io/. 902, 903

Margossian, C. C., Vehtari, A., Simpson, D., and Agrawal, R. (2020). "Hamiltonian Monte Carlo using an adjoint-differentiated Laplace approximation: Bayesian inference for latent Gaussian models and beyond." In *Advances in Neural Information Processing Systems*. MR4361132. 890

Merlo, J., Chaix, B., Yang, M., Lynch, J., and Rastam, L. (2005). "A brief conceptual tutorial of multilevel analysis in social epidemiology: Linking the statistical concept of clustering to the idea of contextual phenomenon." *Journal of Epidemiology and Community Health*, 59(6): 443–449. doi: https://doi.org/10.1136/jech.2004.023473. 890

Rossman, H., Keshet, A., Shilo, S., Gavrieli, A., Bauman, T., Cohen, O., Shelly, E., Balicer, R., Geiger, B., Dor, Y., and Segal, E. (2020). "A framework for identifying regional outbreak and spread of COVID-19 from one-minute population-wide surveys." *Nature Medicine*, 26(5): 634–638. 891, 898, 899, 902

Rue, H., Riebler, A., Sørbye, S. H., Illian, J. B., Simpson, D. P., and Lindgren, F. K. (2017). "Bayesian computing with INLA: A review." *Annual Review of Statistics and Its Application*, 4(1): 395–421. doi: https://doi.org/10.1146/annurev-statistics-060116-054045. 890

Segal, E., Zhang, F., Lin, X., King, G., Shalem, O., Shilo, S., Allen, W. E., Alquaddoomi, F., Altae-Tran, H., Anders, S., Balicer, R., Bauman, T., Bonilla, X., Booman, G., Chan, A. T., Cohen, O., Coletti, S., Davidson, N., Dor, Y., Drew, D. A., Elemento, O., Evans, G., Ewels, P., Gale, J., Gavrieli, A., Geiger, B., Grad, Y. H., Greene, C. S., Hajirasouliha, I., Jerala, R., Kahles, A., Kallioniemi, O., Keshet, A., Kocarev, L., Landua, G., Meir, T., Muller, A., Nguyen, L. H., Oresic, M., Ovchinnikova, S., Peterson, H., Prodanova, J., Rajagopal, J., Rätsch, G., Rossman, H., Rung, J., Sboner, A., Sigaras, A., Spector, T., Steinherz, R., Stevens, I., Vilo, J., and Wilmes, P. (2020). "Building an international consortium for tracking coronavirus health status." *Nature Medicine*, 26(8): 1161–1165. 898

Trefethen, L. N. (2020). *Approximation Theory and Approximation Practice: Extended Edition*. Philadelphia: SIAM. MR4050406. 893

Acknowledgments

The authors are grateful to Hagai Rossman and Ayya Keshet for useful discussions and their contribution to the COVID-19 model.