Calibrated Data Augmentation for Scalable Markov Chain Monte Carlo

Leo L. Duan, James E. Johndrow, David B. Dunson

February 10, 2017

Abstract: Data augmentation is a common technique for building tuning-free Markov chain Monte Carlo algorithms. Although these algorithms are very popular, autocorrelations are often high in large samples, leading to poor computational efficiency. This phenomenon has been attributed to a discrepancy between Gibbs step sizes and the rate of posterior concentration. In this article, we propose a family of calibrated data augmentation algorithms, which adjust for this discrepancy by inflating step sizes. A Metropolis-Hastings step is included to account for the slight discrepancy between the stationary distribution of the resulting sampler and the exact posterior distribution. The approach is applicable to a broad variety of existing data augmentation algorithms, and we focus on three popular models: probit, logistic and Poisson log-linear. Theoretical support is provided and dramatic gains are shown in applications.

KEY WORDS: Bayesian probit; Bayesian logit; Big n; Data Augmentation; Maximal Correlation; Polya-Gamma.

1 Introduction

With the deluge of data in many modern application areas, there is pressing need for scalable computational algorithms for inference from such data, including uncertainty quantification (UQ). Somewhat surprisingly, even as the volume of data increases, uncertainty often remains sizable. Examples in which this phenomenon occurs include financial fraud detection (Ngai et al., 2011), disease mapping (Wakefield, 2007) and online click-through tracking (Wang et al., 2010). Bayesian approaches provide a useful paradigm for quantifying uncertainty in inferences and predictions in these and other settings.

The standard approach to Bayesian posterior computation is Markov chain Monte Carlo (MCMC) and related sampling algorithms. Non-sampling alternatives, such as variational Bayes, tend lack general accuracy guarantees. However, it is well known that conventional MCMC algorithms often scale poorly in problem size and complexity. Due to its sequential nature, the computational cost of MCMC is the product of two factors: the evaluation cost at each sampling iteration and the total number of iterations needed to obtain

an acceptably low Monte Carlo error. The latter is related to the properties of the Markov transition kernel; we will refer to this informally as the *mixing properties* of the Markov chain.

In recent years, a substantial literature has developed focusing on decreasing computational cost per iteration (Minsker et al. (2014); Srivastava et al. (2015); Conrad et al. (2015) among others), mainly through accelerating or parallelizing the sampling procedures at each iteration. Moreover, myriad strategies for improving mixing have been described in the literature. For Metropolis-Hastings (M-H) algorithms, improving mixing is usually a matter of contructing a better proposal distribution. An important difference between M-H and Gibbs is that one has direct control over step sizes in M-H through choice of the proposal, while Gibbs step sizes are generally not tunable; on the other hand, finding a good proposal for multi-dimensional parameters in M-H is significantly more challenging compared to Gibbs sampling. Thus, improving mixing for Gibbs has historically focused on decreasing autocorrelation by changing the update rule itself, for example by parameter expansion (PX), marginalization, or slice sampling.[1]

The theory literature on behavior of MCMC for large n and/or p is arguably somewhat limited. Many authors have focused on studying mixing properties by showing an ergodicity condition, such as geometric ergodicity (Roberts et al., 2004; Meyn and Tweedie, 2012). This generally yields bounds on the convergence rate and spectral gap of the Markov chain, but Rajaratnam and Sparks (2015) observe that in many cases, these bounds converge to one exponentially fast in p or n, so that no meaningful guarantee of performance for large problem sizes is provided by most existing bounds. In the probability literature, a series of papers have developed an analogue of Harris' theorem and ergodic theory for infinite-dimensional state spaces (Hairer et al., 2011). Recent work verifies the existence of MCMC algorithms for computation in differential equation models with dimension-independent spectral gap (Hairer et al., 2014). In this example, the algorithm under consideration is an M-H algorithm, and it is clear that the proposal must be tuned very carefully to achieve dimension independence. Other work has studied the properties of the limiting differential equation that describes infinite-dimensional dynamics of MCMC.

A recent paper (Johndrow et al. (2016)) studies popular data augmentation algorithms for posterior computation in probit (Albert and Chib, 1993) and logistic (Polson et al., 2013) models, showing that the algorithms fail to mix in large sample sizes when the data are imbalanced. An important insight is that the performance can be largely explained by a discrepancy between the rate at which Gibbs step sizes and the width of the high-probability region of the posterior converge to zero as the sample size increases. Thus, since Gibbs step sizes are generally not tunable, slow mixing is likely to occur as the sample size grows unless the order of the step size happens to match the order of the posterior width. This implies that if a way to directly control the step sizes of the Gibbs sampler could be devised, it would be possible to make the mixing

[1] Although strictly speaking, slice sampling is just an alternative approach to sampling from a full conditional distribution, in practice, it is often an alternative to data augmentation, so that using a slice sampling strategy results in the removal of a data augmentation step from an alternative Gibbs sampler.

properties of the sampler insensitive to sample size by scaling the step sizes appropriately. This is similar to the conclusion of Hairer et al. (2014), except in this case, we have growing n instead of growing p.

In this article, we propose a method for tuning step sizes by introducing auxiliary parameters that change the variance of full conditional distributions for one or more parameters. As these "calibrated" data augmentation algorithms alter the invariant measure, one can use the Gibbs step as a highly efficient M-H proposal, thereby recovering the correct invariant, or view the resulting algorithm as a perturbation of the original Markov chain. In this article, we focus on the former strategy, providing theoretical support and showing very substantial practical gains in computational efficiency attributed to our calibration approach.

2 Calibrated Data Augmentation

Data augmentation Gibbs samplers alternate between sampling latent data z from their conditional posterior distribution given model parameters θ and observed data y, and sampling parameters θ given z and y; either of these steps can be further broken down into a series of full conditional sampling steps but we focus for simplicity on algorithms of the form (Tanner and Wong, 1987):

$$\begin{aligned} z \mid \theta, y &\sim h(z; m(\theta), y) \\ \theta \mid z, y &\sim f(\theta; \mu, \Sigma), \end{aligned} \tag{1}$$

where f belongs to a location-scale family, such as the Gaussian, with μ as its conditional location and Σ its conditional scale; h is also a distribution that can be sampled directly. We use $\pi(.)$ to denote the density. Letting $i = 1, \ldots, n$ index the samples, with $y = (y_1, \ldots, y_n)$, often $\pi(z \mid \theta, y) = \prod_{i=1}^{n} \pi(z_i; m_i(\theta), y_i)$, so that the latent data for different subjects can be sampled independently. Popular data augmentation algorithms independently sample the z_i's and then draw θ simultaneously (or at least in blocks) from a multivariate Gaussian or other standard distribution. Data augmentation algorithms are particularly common for generalized linear models (GLMs), with $\mathbb{E}(y_i \mid x_i, \theta) = g^{-1}(x_i\theta)$ and a conditionally Gaussian prior distribution chosen for θ. We focus in particular on binomial probit, binomial logistic, and Poisson log-linear as motivating examples.

Particularly as n increases, a key problem can arise in implementing update rule (1); in particular, the step size of $\text{var}(\theta_t \mid \theta_{t-1})$ from the $(t-1)$th iteration to the tth iteration can be substantially smaller than the posterior $\text{var}(\theta \mid y)$. This leads to slow mixing. To solve this problem, we replace these sampling steps with two steps from a *calibrated* data augmentation (CDA); since the target is modified because of the calibration, we use an M-H step to preserve the original target.

We first present the general algorithm, then use examples to illustrate the details.

2.1 General Algorithm

The key problem with algorithm (1) is that the step size being too small compared to $\text{var}(\theta \mid y)$. Since $\text{var}(\theta_t \mid \theta_{t-1}) \geq \mathbb{E}_{z|\theta_{t-1}} \text{var}(\theta_t \mid z, y)$, we propose to increase the expected conditional variance $\mathbb{E}_{z|\theta_{t-1}} \text{var}(\theta_t \mid z, y)$ to adjust the step size and match the marginal variance $\text{var}(\theta \mid y)$. Through this, each sampling iteration from conditional posterior could explore the marginal posterior region efficiently. For simpler notation, we write the expected conditional variance as $\mathbb{E}_z \text{var}(\theta \mid z, y)$.

To start the calibration, note the two steps in (1) are associated with two equivalent factorizations of likelihood $L(\theta; y)$ with augmented data z:

$$L(\theta; y) = \int \pi(z \mid \theta, y)\, \pi(\theta \mid y) dz = \int \pi(\theta \mid z, y)\, \pi(z \mid y) dz. \tag{2}$$

Depending on the form of $\text{var}(\theta \mid z, y)$ in the second step, there are two different ways to induce an increase in $\mathbb{E}_z \text{var}(\theta \mid z, y)$. If $\text{var}(\theta \mid z, y)$ is free from z (hence $\mathbb{E}_z \text{var}(\theta \mid z, y) = \text{var}(\theta \mid z, y)$), we directly modify it via changing $\pi(\theta \mid z, y)$ in the second factorization of (2); if $\text{var}(\theta \mid z, y)$ is a function of z, we modify the first step by changing $\pi(z \mid \theta, y)$ in the first factorization of (2), in order to stochastically influence the value of z and $\mathbb{E}_z \text{var}(\theta \mid z, y)$.

By "change", we meant a *relaxation* of a fixed distribution parameter to a tunable working parameter r. Therefore, this minor change does not impact the posterior distribution families h and f in (1), nor the integrability for obtaining the marginal likelihood via (2). The two adjustment approaches will be concretely illustrated via examples in the next few subsections.

As the change alters the target distribution, we use an M-H step to correct the difference. As the modification in variance usually impacts the location parameter of θ as well, to facilitate higher M-H acceptance rate, we introduce another auxiliary parameter b to allow further adjusting of the location parameter of θ.

The CDA sampling algorithm involves generating a new proposal θ^* via two steps:

$$z \mid \theta, y \sim h_{r,b}(z; m(\theta), y)$$

$$\theta^* \mid z, y \sim f(\theta^* | \mu(r, b), \Sigma(r))$$

and accepting the proposal θ^* with probability (with its derivation deferred to the theory section):

$$1 \wedge \frac{L(\theta^*; y) L_{r,b}(\theta; y)}{L(\theta; y) L_{r,b}(\theta^*; y)},$$

where $L_{r,b}(\theta; y)$ is the integrated marginal likelihood after inclusion of r and b:

$$L_{r,b}(\theta; y) = \int \pi_{r,b}(z \mid \theta, y)\, \pi_{r,b}(\theta \mid y) dz. \tag{3}$$

There are three important properties about CDA. First, for any fixed (r, b) in suitable domain, $L_{r,b}(\theta; y)$ is a proper likelihood function, because the calibration does not alter the families of distribution in (2). Second, a constrained domain $\mathcal{R}$ can be specified for r so that the step size is always greater or equal to the un-calibrated one. Third, $L(\theta; y)$ is a special case of $L_{r,b}(\theta; y)$ for certain r and b (e.g. mostly $L(\theta; y) = L_{1,0}(\theta; y)$). When that happens, M-H step will always accept the proposal with probability 1 and the algorithm reverts to Gibbs sampling. Therefore, CDA sampling is a generalized algorithm of Gibbs sampling for data augmentation.

2.2 Initial example: probit with intercept only

We first use a simple model to illustrate the calibration procedure for one parameter and demonstrate the effects of step size on mixing. Consider a probit model with intercept only

$$y_i \sim \text{Bernoulli}(p_i), \quad p_i = \Phi(\theta) \quad i = 1, \ldots, n, \quad \pi(\theta) \propto 1,$$

where $\Phi(\theta)$ is the cumulative distribution function of standard normal; $\pi(\theta)$ is the flat prior for θ. The basic data augmentation algorithm (Albert and Chib, 1993) has the update rule

$$z_i \mid \theta, y_i \sim \begin{cases} \text{No}_{[0,\infty)}(\theta, 1) & \text{if } y_i = 1 \\ \text{No}_{(-\infty,0]}(\theta, 1) & \text{if } y_i = 0 \end{cases} \qquad i = 1, \ldots, n$$

$$\theta \mid z, y \sim \text{No}(\sum_i z_i/n, 1/n),$$

where $\text{No}_{[a,b]}(\mu, \sigma^2)$ is the normal distribution with mean μ and variance σ^2 truncated to the interval $[a, b]$. Johndrow et al. (2016) show that when $\sum_i y_i$ is fixed and much smaller than n, $\text{var}(\theta_t \mid \theta_{t-1})$ is approximately $n^{-1} \log n$, while the width of the high probability density region of the posterior is of order $(\log n)^{-1}$. The rate difference causes critical mixing issue in rare event modeling.

As the conditional variance $\text{var}(\theta \mid z, y)$ is free from z, we replace the fixed small variance $1/n$ with a tunable scalar r/n and adjust the location with another scalar parameter b. We further require $r \geq 1$ to ensure step size is unchanged or increased. Utilizing the equivalent factorizations in (2), this is the same as changing the scale of $z_i \mid \theta, y_i$ from 1 to r and the mean from θ to $\theta + b$. These adjustments lead to a new sampling algorithm with proposal θ^*:

$$z_i \mid \theta, y_i \sim \begin{cases} \text{No}_{[0,\infty)}(\theta + b, r) & \text{if } y_i = 1 \\ \text{No}_{(-\infty,0]}(\theta + b, r) & \text{if } y_i = 0 \end{cases} \qquad i = 1, \ldots, n \tag{4}$$

$$\theta^* \mid z, y \sim \text{No}(\sum_i (z_i - b)/n, r/n),$$

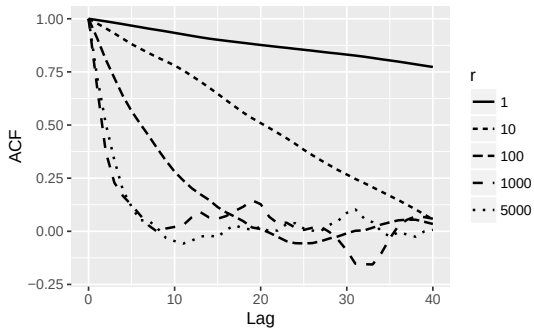
and accepting θ^* with probability

$$1 \wedge \prod_i \frac{L_{r,b}(\theta; y_i) L(\theta^*; y_i)}{L_{r,b}(\theta^*; y_i) L(\theta; y_i)}, \tag{5}$$

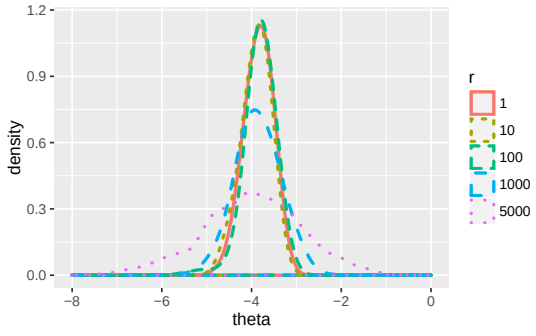
where $L_{r,b}(\theta; y_i) = \Phi\big(\frac{\theta+b}{\sqrt{r}}\big)^{y_i} \Phi\big(-\frac{\theta+b}{\sqrt{r}}\big)^{(1-y_i)}$ and $L(\theta; y_i) = L_{1,0}(\theta; y_i)$. This calibration increases the step size of the proposal to $\text{var}(\theta^* \mid \theta_{t-1}) \geq \mathbb{E}_z \,\text{var}(\theta^* \mid z) = r/n$. To match to the approximate width of marginal high probability density region $(\log n)^{-1}$, theoretically we need $r \approx n/\log n$.

To illustrate, we set $\sum_i y_i = 1$ and $n = 10^4$, and run the new algortihm for these data with different values of r ranging from $r = 1$ to $r = 5,000$, among which $r = 1,000 \approx n/\log n$ corresponding to the theoretically optimal value. In this simple example, it is easy to compute a "good" value of $b = -3.7(\sqrt{r} - 1)$, as it results in $\text{pr}(y_i = 1) = \Phi(-3.7) = n^{-1} \sum_i y_i \approx 10^{-4}$ in the proposal distribution. This choice ensures that each $L_{r,b}(\theta; y_i)$ for different r are centered around the center of $L(\theta; y_i)$, so that the proposal is around the high posterior density of the target, so that it has high acceptance rate.

Figure 1a plots autocorrelation functions (ACFs) for these different samplers. Autocorrelation is very high even at lag 40 for $r = 1$ (which corresponds to the basic Gibbs sampler), while increasing r leads to dramatic improvements in mixing. There are no further gains in increasing r from the theoretically optimal value of $r = 1,000$ to $r = 5,000$. Figure 1b shows the density estimates under $L_{r,b}(\theta; y)$; they are all centered about the same values due to the adjustment of b, and with variance increasing according to r. With M-H adjustment the differences are removed; the M-H step has average acceptance rate close to 1 for $r = 10$ and 100, 0.6 for $r = 1,000$, and 0.2 for $r = 5,000$.



(a) ACF for CDA with different r controlling the step size.



(b) Density estimates of the integrated marginal $L_{r,b}$ with different r.

Figure 1: Autocorrelation functions (ACFs) and density estimates for $L_{r,b}$ in intercept-only probit model.

2.3 Probit regression example

The intercept only probit model illustrates the adjustment of variance when θ is a single parameter. When θ is multi-dimensional, the calibration needs to be applied on the matrix of conditional covariance. To illustrate

this scenario, we consider a probit regression with flat prior:

$$y_i \sim \text{Bernoulli}(p_i), \quad p_i = \Phi(x_i\theta) \quad i = 1, \ldots, n, \quad \pi(\theta) \propto 1$$

and Gibbs sampling rule (Albert and Chib, 1993):

$$z_i \mid \theta, x_i, y_i \sim \begin{cases} \text{No}_{[0,\infty)}(x_i\theta, 1) & \text{if } y_i = 1 \\ \text{No}_{(-\infty,0]}(x_i\theta, 1) & \text{if } y_i = 0 \end{cases} \quad i = 1, \ldots, n$$

$$\theta \mid z, x, y \sim \text{No}((X'X)^{-1}X'z, (X'X)^{-1}),$$

where $X = \{x_1', \ldots, x_n'\}'$ is the predictor matrix.

Similar to the intercept only model, we can directly modify the $\text{var}(\theta|z, y)$ as it is free from z. we begin calibration by changing the variance of z_i from 1 to a parameter r_i; we add b_i to the predictor $x_i\theta$ to allow optimizing for M-H acceptance rate. This yields new update rule of sampling proposal from:

$$z_i \mid \theta, x_i, y_i \sim \begin{cases} \text{No}_{[0,\infty)}(x_i\theta + b_i, r_i) & \text{if } y_i = 1 \\ \text{No}_{(-\infty,0]}(x_i\theta + b_i, r_i) & \text{if } y_i = 0 \end{cases} \quad i = 1, \ldots, n \tag{6}$$

$$\theta^* \mid z, X \sim \text{No}\left((X'R^{-1}X)^{-1}X'R^{-1}(z - b), (X'R^{-1}X)^{-1}\right),$$

with $R = \text{diag}(r_1, \ldots, r_n)$, $b = \{b_1, \ldots, b_n\}'$, and accepting θ^* with probability:

$$1 \wedge \prod_i \frac{L_{r,b}(x_i\theta; y_i)L(x_i\theta^*; y_i)}{L_{r,b}(x_i\theta^*; y_i)L(x_i\theta; y_i)}, \tag{7}$$

where $L_{r,b}(x_i\theta; y_i) = \Phi\big(\frac{x_i\theta+b_i}{\sqrt{r}}\big)^{y_i}\Phi\big(-\frac{x_i\theta+b_i}{\sqrt{r}}\big)^{(1-y_i)}$ and $L(x_i\theta; y_i) = L_{1,0}(x_i\theta; y_i)$.

The calibration generates an adjustable covariance matrix $\text{var}(\theta^* \mid z, X) = (X'R^{-1}X)^{-1}$. We allow r and b to vary over index i so that the covariance can be flexibly tuned. In general, we can tune the covariance to be approximately close to the marginal covariance $\text{var}(\theta \mid y)$, so that the sampler can explore the high posterior density region efficiently. We also require all $r_i \geq 1$ so that the variance can only increase.

We will continue discussing the choice for r and b in section 2.5, after demonstrating the second calibration strategy when $\text{var}(\theta|z, y)$ is not free from z.

2.4 Logistic regression example

The probit examples illustrate the scenario when $\text{var}(\theta|z, y)$ is free from z, the step size can be directly calibrated through changing the second step of (1). When $\text{var}(\theta|z, y)$ does involve z, the calibration can be achieved by changing the first step of (1). We now demonstrate this approach.

Consider the logistic regression model with flat prior:

$$y_i \sim \text{Bernoulli}(p_i), \quad p_i = \frac{\exp(x_i\theta)}{1 + \exp(x_i\theta)} \quad i = 1, \ldots, n, \quad \pi(\theta) \propto 1$$

and a two-step update rule based on Polya-Gamma data augmentation (Polson et al., 2013):

$$z_i \mid \theta, x_i \sim \mathrm{PG}(1, |x_i\theta|) \quad i = 1, \ldots, n,$$

$$\theta \mid z, X, y \sim \mathrm{No}\left((X'ZX)^{-1}X'(y - 0.5), (X'ZX)^{-1}\right),$$

where $Z = \mathrm{diag}(z_1, \ldots, z_n)$ and PG is the Polya-Gamma distribution.

Since Z is random, it would be difficult to directly modify $(X'ZX)^{-1}$ in the second step. Instead, we can stochastically influence the value of Z by modifying the first step. This is achieved by replacing the first Polya-Gamma parameter 1 with a tunable r_i. We require $r_i \in (0, 1]$ to ensure step size is unchanged or increased. Intuitively, smaller r_i tends to produce smaller z_i hence larger z_i^{-1} since $\mathbb{E}z_i = \frac{r_i}{2a}\tanh(\frac{a}{2})$ for $z_i \sim \mathrm{PG}(r_i, a)$. More rigorously, its first negative moment can be computed as $\mathbb{E}z_i^{-1} = \int_0^\infty \prod_{k=1}^\infty (1 + d_k^{-1}t)^{-r_i} dt$ with $d_k = 2(k - \frac{1}{2})^2\pi^2 + \frac{a^2}{2}$ (combining Cressie et al. (1981) and Polson et al. (2013)). Therefore, smaller r_i's lead to larger expected conditional variance $\mathbb{E}_z(X'ZX)^{-1}$.

Similar to the probit example, we add the location adjusting parameter b_i to the linear predictor $x_i\theta$ and allow optimizing for better acceptance rate. The new algorithm is sampling from proposal:

$$z_i \mid \theta, x_i \sim \mathrm{PG}(r_i, |x_i\theta + b_i|) \quad i = 1, \ldots, n,$$

$$\theta^* \mid z, X, y \sim \mathrm{No}\left((X'ZX)^{-1}X'(y - r/2 - Zb), (X'ZX)^{-1}\right),$$

with acceptance probability:

$$1 \wedge \prod_i \frac{L_{r,b}(x_i\theta; y_i)L(x_i\theta^*; y_i)}{L_{r,b}(x_i\theta^*; y_i)L(x_i\theta; y_i)},$$

where

$$\begin{aligned} L_{r,b}(x_i\theta; y_i) = &\int_0^\infty \exp\{(x_i\theta + b_i)(y_i - r_i/2)\} \exp\left\{ - \frac{z_i(x_i\theta + b_i)^2}{2} \right\} \mathrm{PG}(z_i \mid r_i, 0) dz_i \\ = &\frac{\exp\{(x_i\theta + b_i)y_i\}}{\{1 + \exp(x_i\theta + b_i)\}^{r_i}} \end{aligned} \tag{8}$$

and $L(x_i\theta; y_i) = L_{1,0}(x_i\theta; y_i)$.

2.5 Choice of calibration parameters

We now illustrate a simple and efficient strategy for selecting calibration parameters $r = (r_1, \ldots, r_n)$ and $b = (b_1, \ldots, b_n)$, and use them for calibrating the above probit and logistic regression examples. Unlike the intercept probit example, in general it is difficult to analytically calculate the marginal or expected conditional variances. Therefore, we now propose an empirical approach to estimate the difference between the two. Despite being empirical, this method follows the guarantee that the mixing will be accelerated, since the tuned r is constrained in region $\mathcal{R}$ where step size can only increase.

For the marginal variance, we use inverse Fisher information for approximation. In large samples, the inverse of Fisher information evaluated at the "true value" θ_0 of the parameter is an asymptotic approximation to the posterior marginal covariance. When the expectation of conditional variance is intractable, we can consider the inverse of expected conditional precision as an approximate. Depending on which is convenient to compute, it is useful to choose $r \in \mathcal{R}$ to minimize the one of the distances below, :

$$\Delta_{\mathcal{I}^{-1}}(\theta_0) = \|c_0\mathcal{I}^{-1}(\theta_0) - \mathbb{E}_{z|\theta,r,b}\,\mathrm{var}(\theta^* \mid z)\Big|\theta_0\|_F$$

$$\Delta_{\mathcal{I}}(\theta_0) = \|\mathcal{I}(\theta_0)/c_0 - \mathbb{E}_{z|\theta,r,b}\big(\mathrm{var}(\theta^* \mid z)\big)^{-1}\Big|\theta_0\|_F$$

where $\mathcal{I}(\theta_0) = E_y\left[\left(\frac{\partial}{\partial\theta}\log L(y;\theta)\right)^2\right]\Big|\theta_0$ with expectation taken over the distribution of the data y under the target likelihood L, and $\|A\|_F$ is the Frobenius norm of A; $\mathbb{E}_{z|\theta,r,b}$ is taken with respect to the latent variable $z \mid \theta$ under $L_{r,b}(\theta; y)$. By default, we set $c_0 = 1$. The parameter c_0 is a scalar constant reserved for adjusting when this approximating approach over-estimated the needed step size. When that happens, it is shown by a very low acceptance rate, as the calibrated step size is too wide and most of proposals fall out of the high posterior region of L; it would then be useful to use smaller c_0 to reduce the step size.

Since there is no fixed θ_0 in Bayesian paradigm, we use the empirical maximum-a-posteriori (MAP) estimate $\hat{\theta}_t = \underset{\theta\in\{\theta_1,\ldots,\theta_t\}}{\arg\max}\ L(\theta; y)\Pi^0(\theta)$, based on posterior samples $\{\theta_1, \ldots, \theta_t\}$ collected in the first t iterations, and $\Pi^0(\theta)$ is the prior. We then dynamically update r_t, b_t based on $\hat{\theta}_t$. Specifically, we choose r_{t+1} to minimize $\Delta_{\mathcal{I}^{-1}}(\hat{\theta}_t)$ or $\Delta_{\mathcal{I}}(\hat{\theta}_t)$, and set b_{t+1} to minimize the difference between $L_{1,0}(\hat{\theta}_t; y)$ and $L_{r_{t+1},b_{t+1}}(\hat{\theta}_t; y)$. Thus, we use r to adjust the expected conditional variance based on $L_{r,b}$ to match the marginal variance based on L; and we use b to make $L_{r,b}$ close to $L_{1,0}$ in the neighborhood of $\hat{\theta}_t$. Intuitively, this will make the proposal distribution closer to the target, and correspondingly increase the M-H acceptance rate.

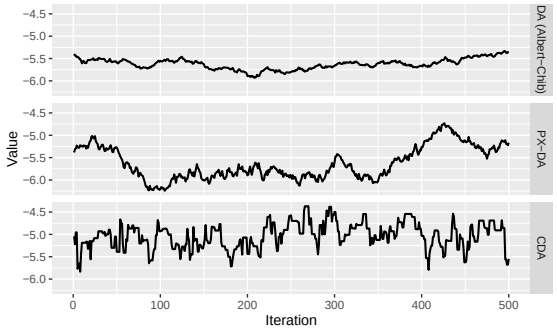
The proposal kernel we describe above is *adaptive*; that is, we have a collection of proposal kernels $\mathcal{Q} = \{Q_{r,b}\}_{(r,b)\in R\times\mathcal{B}}$, and we choose a different member of $\mathcal{Q}$ to create the proposal when $\hat{\theta}_t$ is changed. For every $Q_{r,b}$, the *target* Π for the resulting transition kernel is the same as the original Gibbs sampler, because of the Metropolis-Hastings rejection step. In general, ergodicity of adaptive algorithms requires a diminishing adaptation condition; a general condition of this sort is given in Roberts and Rosenthal (2007). Although the algorithm we describe is unlikely to satisfy diminishing adaptation, the condition is trivially satisfied by any algorithm that stops adaptation after a fixed number of iterations. Thus, for simplicity, we choose a tuning period length, after which we fix r, b at their current values. More sophisticated adaptation schemes could be devised; however, the fixed tuning period works well empirically.

For a concrete illustration, we now return to the example of the probit regression. Letting $\hat{\eta}_i = x_i\hat{\theta}_t$ denote the linear predictor corresponding to the empirical MAP, during the tuning period, the marginal approximate and the expected conditional variance are:

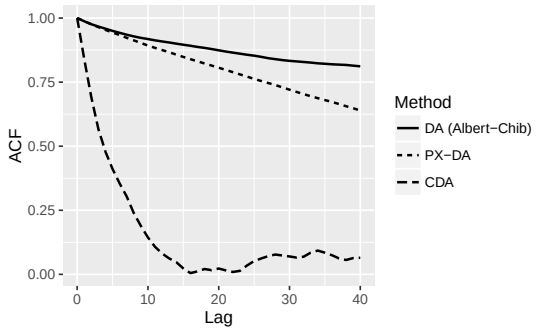
$$c_0\mathcal{I}^{-1}(\hat{\theta}_t) = c_0(X' \operatorname{diag}\left\{\frac{\phi(\hat{\eta}_i)^2}{\Phi(\hat{\eta}_i)(1-\Phi(\hat{\eta}_i))}\right\}_{i=1,\dots n} X)^{-1}, \qquad \mathbb{E}_{z|\theta,r,b} \operatorname{var}(\theta^* \mid z) = (X'R^{-1}X)^{-1}$$

respectively, where ϕ is the standard normal density. Therefore, we set $r_i = \hat{r}_i \vee 1$ with $\hat{r}_i = c_0 \frac{\Phi(\hat{\eta}_i)(1-\Phi(\hat{\eta}_i))}{\phi(\hat{\eta}_i)^2}$, as $\Delta_{\mathcal{I}^{-1}}(\hat{\theta}_t) = 0$ if all $\hat{r}_i \geq 1$. The r_i's can be calculated using this expression at low cost without calculating the full information matrix. We set $b_i = \hat{\eta}_i(\sqrt{r_i} - 1)$ to ensure that $|L_{r,b}(\hat{\eta}_i; y_i) - L(\hat{\eta}_i; y_i)| = 0$.

To illustrate, we use a probit regression with an intercept $x_{i,0} = 1$ and two predictors $x_{i,1}, x_{i,2} \sim \mathrm{No}(1, 1)$, with "true" $\theta = (-5, 1, -1)'$, generating $\sum y_i = 20$ among $n = 10,000$. The Albert and Chib (1993) DA algorithm mixes slowly (Figure 2a and 2b). We also consider the parameter expansion algorithm (PX-DA) (Liu and Wu, 1999; Meng and Van Dyk, 1999), which re-parameterizes θ with a redundant scale parameter. The key difference is that PX-DA does not increase the conditional variance of θ and the extra parameter is very constrained as θ-marginal likelihood needs to be unaltered, therefore it cannot directly solve the small step size problem. As the result, PX-DA only mildly reduces the correlation. Using CDA, we initially tuned r and b for 100 steps using the Fisher information at default $c_0 = 1$, and collected samples afterwards with fixed r and b. We obtain dramatically better mixing at an average acceptance rate of 0.6.



(a) Traceplot for the original DA, parameter expanded DA and CDA algorithms.



(b) ACF for original DA, parameter expanded DA and CDA algorithms.

Figure 2: Panel (a) demonstrates in traceplot and panel (b) in autocorrelation the substantial improvement in CDA by correcting the variance mis-match in probit regression with rare event data, compared with the original (Albert and Chib, 1993) and parameter-expanded methods (Liu and Wu, 1999).

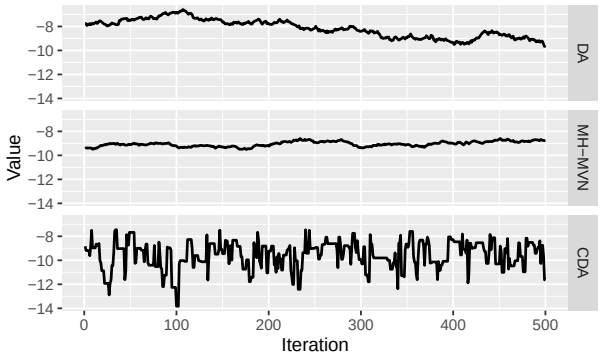
As for the second example of the logistic regression, since $\mathbb{E}z$ is more convenient to compute than $\mathbb{E}z^{-1}$, we minimize the distance $\Delta_{\mathcal{I}}(\hat{\theta}_t)$ based on precision matrices. Letting $\hat{\eta}_i = x_i\hat{\theta}_t$ based on empirical MAP during tuning, the marginal approximate and the conditional precision matrices are:

$$\mathcal{I}(\hat{\theta}_t)/c_0 = (X' \operatorname{diag}\left\{\frac{\exp(\hat{\eta}_i)}{\{1+\exp(\hat{\eta}_i)\}^2}\right\}_{i=1,\dots,n} X)/c_0,$$

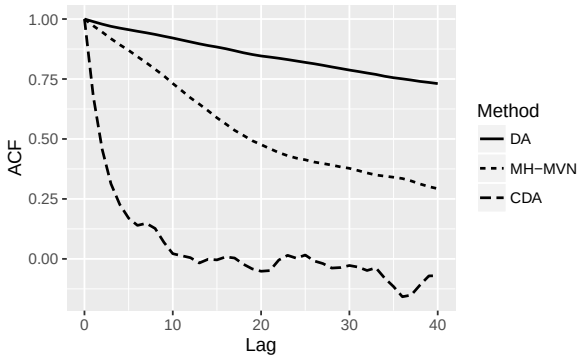
$$\mathbb{E}_{z|\theta,r,b}\big(\operatorname{var}(\theta^* \mid z)\big)^{-1}\Big|\hat{\theta}_t = X' \operatorname{diag}\left\{\frac{r_i}{2|\hat{\eta}_i + b_i|}\tanh\Big(\frac{|\hat{\eta}_i + b_i|}{2}\Big)\right\}_{i=1,\ldots,n} X$$

We set $r_i = \hat{r}_i \wedge 1$ with $\hat{r}_i = \frac{1}{c_0}\frac{\exp(\hat{\eta}_i)}{\{1+\exp(\hat{\eta}_i)\}^2} 2|\hat{\eta}_i + b_i|/\tanh(\frac{|\hat{\eta}_i+b_i|}{2})$, as $\Delta_{\mathcal{I}}(\hat{\theta}_t) = 0$ if all $\hat{r}_i \leq 1$; comparing $L_{r,b}(\theta \mid y)$ and $L(\theta \mid y)$, we set $b_i = \log[\{1 + \exp(\hat{\eta}_i)\}^{1/r_i} - 1] - \hat{\eta}_i$ to minimize the difference.

To illustrate, we use a two parameter intercept-slope model with $x_{i,0} = 1$, $x_{i,1} \sim \text{No}(0, 1)$ for $i = 1, \ldots, n$ and "true" $\theta = (-9, 1)'$. With $n = 10^5$, we obtain rare event data with $\sum y_i = 50$. Shown in Figure 3, the original DA algorithm (Polson et al., 2013) mixes slowly. We consider a simpler M-H algorithm with multivariate normal proposal with inverse Fisher information as the covariance (denoted as MH-MVN). Since the normal proposal and the logistic target are in different distribution families, the mixing does not get significantly improved. For CDA we tuned r and b for 100 steps using the Fisher information at default $c_0 = 1$, reaching an acceptance rate of 0.8, then collected samples. CDA has dramatically better mixing.



(a) Traceplots for DA, CDA and M-H with multivariate normal proposal.



(b) ACF for DA, CDA and M-H with multivariate normal proposal.

Figure 3: Panel (a) demonstrates in traceplot and panel (b) in autocorrelation the substantial improvement of CDA in logistic regression with rare event data, compared with the original DA (Polson et al., 2013) and the M-H algorithm with multivariate normal proposal (MH-MVN).

3 Theory

In this section, we provide theoretical support for CDA algorithms. Consider a Markov kernel $K((\theta, z); \cdot)$ with invariant measure Π and update rule of the form (1), and a Markov chain (θ_t, z_t) on a state space $\Theta \times \mathcal{Z}$ evolving according to K. We will abuse notation in writing $\Pi(d\theta) = \int_{z\in\mathcal{Z}} \Pi(d\theta, dz)$. The lag-1 autocorrelation for a function $g : \Theta \to \mathbb{R}$ at stationarity can be expressed as the Bayesian fraction of missing information (Papaspiliopoulos et al., 2007)

$$\gamma_g = 1 - \frac{\mathbb{E}[\operatorname{var}(g(\theta) \mid z)]}{\operatorname{var}(g(\theta))}, \tag{9}$$

where the integrals in the numerator are with respect to $\Pi(d\theta, dz)$ and in the denominator with respect to $\Pi(d\theta)$. Let

$$L_2(\Pi) = \left\{ g : \Theta \to \mathbb{R}, \int_{\theta \in \Theta} \{g(\theta)\}^2 \Pi(d\theta) < \infty \right\}$$

be the set of real-valued, Π square-integrable functions. The *maximal autocorrelation*

$$\gamma = \sup_{g \in L^2(\Pi)} \gamma_g = 1 - \inf_{g \in L^2(\Pi)} \frac{\mathbb{E}[\text{var}(g(\theta) \mid z)]}{\text{var}(g(\theta))}$$

is equal to the geometric convergence rate of the data augmentation Gibbs sampler (Liu, 2008). For $g(\theta) = \theta_j$ a coordinate projection, the numerator of the last term of (9) is, informally, the average squared step size for augmentation algorithm at stationarity in direction j, while the denominator is the squared width of the bulk of the posterior in direction j. Consequently, γ will be close to 1 whenever the average step size at stationarity is small relative to the width of the bulk of the posterior, leading to slow mixing.

The purpose of CDA is to introduce working parameters that allow us to control the step size – roughly speaking, the numerator of (9) – with greater flexibility than reparametrization or parameter expansion. The flexibility gains are achieved by allowing the invariant measure to change as a result of the introduced parameters. The working parameters (r, b) correspond to a collection of reparametrizations, each of which defines a proper (but distinct) likelihood $L_{r,b}(\theta; y)$, and for which there exists an update rule of the form (1). Without using M-H step, one can derive a Gibbs sampler associated with θ-marginal invariant measure $\Pi_{r,b}(\theta; y) \propto L_{r,b}(\theta; y)\Pi^0(\theta)$. We refer this sampler as calibrated Gibbs (C-Gibbs). But ultimately, we are interested in the original $\Pi(\theta; y)$, so we use one iteration of C-Gibbs as an efficient proposal for Metropolis-Hastings. That is, we propose θ^* from $Q(\theta; \cdot)$ where

$$Q_{r,b}(\theta; A) = \int_{(\theta^*, z) \in A \times \mathcal{Z}} \pi_{r,b}(z; \theta, y) f_{r,b}(\theta^*; z, y) dz d\theta^* \tag{10}$$

for $A \subseteq \Theta$, where $\pi_{r,b}$ and $f_{r,b}$ denote the conditional densities of z and θ in C-Gibbs sampler with invariant measure $\Pi_{r,b}$. Similarly, we denote the Markov kernel as $K_{r,b}((\theta, z); (\theta^*, z'))$ for the transition from (θ, z) to (θ^*, z'). By tuning working parameters during an adaptation phase to minimize the lag-1 autocorrelation for the identity function while optimizing the Metropolis-Hastings acceptance rate, we can select values of the working parameters that yield a computationally efficient algorithm.

First, we show that CDA is ergodic. This is basically a consequence of C-Gibbs being ergodic for fixed r, b and the fact that $\Pi_{r,b}$ and Π are absolutely continuous with respect to Lebesgue measure on $\mathbb{R}^p$.

Remark 1 (ergodicity)**.** *Assume that $\Pi(d\theta)$ and $\Pi_{r,b}(d\theta)$ have densities with respect to Lebesgue measure on $\mathbb{R}^p$, and that $K_{r,b}((\theta, z); (\theta^*, z')) > 0 \, \forall \, ((\theta, z), (\theta^*, z')) \in (\Theta \times \mathcal{Z}) \times (\Theta \times \mathcal{Z})$. Then, for fixed r, b, C-Gibbs is*

ergodic with invariant measure $\Pi_{r,b}(d\theta, dz)$*. Moreover, CDA, a Metropolis-Hastings algorithm with proposal kernel* $Q_{r,b}(\theta^*; \theta)$ *as defined in* (10) *with fixed* r, b*, is ergodic with invariant measure* $\Pi(d\theta)$*.*

Proof. For any r, b, the conditionals $\Pi_{r,b}(z \mid \theta)$ and $\Pi_{r,b}(\theta \mid z)$ are well-defined for all $z \in \mathcal{Z}, \theta \in \Theta$, and therefore the Gibbs transition kernel $K_{r,b}((\theta, z); \cdot)$ and corresponding marginal kernels $Q_{r,b}(\theta; \cdot)$ are well-defined. Moreover, for any $(z, \theta) \in \mathcal{Z} \times \Theta$, we have $\mathbb{P}[(\theta^*, z') \in A \mid (\theta, z)] > 0$ by assumption. Thus $K_{r,b}$ is aperiodic and $\Pi_{r,b}$-irreducible.

$Q_{r,b}(\theta^*; \theta)$ is aperiodic and $\Pi_{r,b}(\theta)$-irreducible, since it is the θ marginal transition kernel induced by $K_{r,b}((\theta, z); \cdot)$. Thus, it is also $\Pi(\theta)$-irreducible so long as $\Pi \gg \Pi_{r,b}$, where for two measure μ, ν, $\mu \gg \nu$ indicates absolute continuity. Since $\Pi, \Pi_{r,b}$ have densities with respect to Lebesgue measure, $\Pi_{r,b}$-irreducibility implies Π irreducibility. Moreover, $Q(\theta; \theta^*) > 0$ for all $\theta \in \Theta$. Thus, by Theorem 3 of Roberts and Smith (1994), CDA is Π-irreducible and aperiodic. □

Having established ergodicity of both C-Gibbs and CDA under weak assumptions that hold for all of the data augmentation strategies we consider here, we now provide a semi-rigorous argument for why our approach to tuning r and b results in both rapid convergence and closeness of $\Pi_{r,b}$ to Π. Suppose there exists r such that

$$\mathbb{E}_{\Pi_{r,b}}[\mathrm{var}(\theta \mid z)] = \mathrm{var}_{\Pi}(\theta)$$

for any value of b. This is a simplification, since our adaptation strategies only allow us to control the discrepancy between the approximate marginal and expected conditional variance at certain fixed state. By tuning r during the adaptation phase, we make the lag-1 autocorrelation for the identity function small.

This is obviously much weaker than minimizing the autocorrelation for worst-case functions. However, for the sake of exposition, we will proceed on the assumption that we can make the lag-1 autocorrelation for the identity function close to zero by appropriately tuning r. This makes the rationale for tuning b to increase the Metropolis acceptance probability much clearer. First, we note the form of the Metropolis acceptance ratios, which we have used previously without rigorous justification.

Remark 2. *The CDA acceptance ratio is given by*

$$\frac{L(\theta^*; y)\Pi^0(\theta^*)Q_{r,b}(\theta; \theta^*)}{L(\theta; y)\Pi^0(\theta)Q_{r,b}(\theta^*; \theta)} = \frac{L(\theta^*; y)L_{r,b}(\theta; y)}{L(\theta; y)L_{r,b}(\theta^*; y)} \tag{11}$$

Proof. Since $Q_{r,b}(\theta; \theta^*)$ is the θ marginal of a Gibbs transition kernel, and Gibbs is reversible on its margins, we have

$$Q(\theta; \theta^*)\Pi_{r,b}(\theta) = Q(\theta^*; \theta)\Pi_{r,b}(\theta),$$

and so

$$\frac{L(\theta^*; y)\Pi^0(\theta^*)Q(\theta; \theta^*)}{L(\theta; y)\Pi^0(\theta)Q(\theta^*; \theta)} = \frac{L(\theta^*; y)\Pi^0(\theta^*)L_{r,b}(\theta; y)\Pi^0(\theta)}{L(\theta; y)\Pi^0(\theta)L_{r,b}(\theta^*; y)\Pi^0(\theta^*)}$$

$$= \frac{L(\theta^*; y) L_{r,b}(\theta; y)}{L(\theta; y) L_{r,b}(\theta^*; y)}.$$

□

The expression in (11) will be near 1 at stationarity if

$$\int \log \left(\frac{L(\theta^*; y) L_{r,b}(\theta; y)}{L(\theta; y) L_{r,b}(\theta^*; y)} \right) Q_{r,b}(\theta^*; \theta) \Pi(d\theta) \approx 0.$$

Now, suppose that a Markov chain evolving according to $K_{r,b}$ is rapidly mixing, so that for starting measures satisfying a condition like

$$\sup_A \frac{\nu(A)}{\Pi_{r,b}(A)} < M$$

for M not too large we have

$$\mathrm{KL}\left(\Pi_{r,b} \parallel \int Q_{r,b}(\theta^*; \theta) \nu(d\theta) \right) \quad \text{small.}$$

Then the symmetric KL is

$$\begin{aligned}
\mathrm{KL}\left(\Pi_{r,b} \parallel \Pi\right) + \mathrm{KL}\left(\Pi \parallel \Pi_{r,b}\right) &= \int \Pi_{r,b}(d\theta) \log \frac{\Pi_{r,b}(\theta)}{\Pi(\theta)} + \int \Pi(d\theta) \log \frac{\Pi(\theta)}{\Pi_{r,b}(\theta)} \\
&= \int \Pi_{r,b}(d\theta) \log \frac{c_{r,b} L_{r,b}(\theta) \Pi_0(\theta)}{c L(\theta) \Pi_0(\theta)} + \int \Pi(d\theta) \log \frac{c L(\theta) \Pi_0(\theta)}{c_{r,b} L_{r,b}(\theta) \Pi_0(\theta)} \\
&\approx \int K_{r,b}(\theta^*; \theta) \Pi(d\theta) \log \frac{L_{r,b}(\theta^*)}{L(\theta^*)} + \int \Pi(d\theta) \log \frac{L(\theta)}{L_{r,b}(\theta)} \\
&= \mathbb{E}\left[\frac{L_{r,b}(\theta^*) L(\theta)}{L_{r,b}(\theta) L(\theta^*)} \right],
\end{aligned}$$

for $\theta \sim \Pi$ and $\theta^* \mid \theta \sim K_{r,b}(\theta^*; \theta)$, so that tuning b to make the M-H acceptance ratio larger will tend to make the symmetric KL between $\Pi_{r,b}$ and Π small. As the acceptance ratio approaches 1, CDA and C-Gibbs coincide, and the C-Gibbs invariant measure is identically Π, but the corresponding sampler converges rapidly.

4 Co-Browsing Behavior Application

We now apply CDA to an online browsing activity dataset, in order to illustrate the application of CDA under more complicated setting such as hierarchical model and random effects. The dataset contains a two-way table of visit count, made by internet users who browsed one of 96 client websites, and one of the $n = 59,792$ high-traffic sites during the same browsing session. We refer to the behavior of visiting more than one sites at the same session as "co-browsing". For each client site, it is of high commercial interests to identify the top

few high-traffic sites with high co-browsing rates, in order to place more ads on those sites. For advertising company, it is useful to predict the traffic patterns between the high-traffic sites and their clients, so that the data collection can be less frequent on those clients that can be well predicted. We consider two models for these data.

4.1 Hierarchical Binomial Model for Estimating Co-browsing Rates

We first focus on one client website and analyze its co-browsing rates with the high-traffic sites. With the total visit count N_i available for the ith high-traffic site, the count of co-browsing visit y_i with the client of interests can be considered as the result of a binomial trial. With y_i extremely small relative to N_i (with ratio (0.00011 ± 0.00093)), the maximum likelihood estimate y_i/N_i can have poor performance. For example, when $y_i = 0$, estimating the rate as exactly 0 is not ideal. Instead, it is useful to consider a hierarchical model that allows borrowing of information across high-traffic sites.

$$y_i \sim \text{Binomial}\left(N_i, p_i\right), \quad p_i = \frac{\exp(\theta_i)}{1+\exp(\theta_i)}, \quad \theta_i \overset{iid}{\sim} \text{No}(\theta_0, \sigma_0^2) \quad i = 1 \dots n$$

$$(\theta_0, \sigma_0^2) \sim \pi(\theta_0, \sigma_0^2)$$

Based on expert opinion in quantitative advertising, we use a weakly informative prior $\theta_0 \sim \text{No}(-12, 49)$ and non-informative uniform prior on σ_0^2.

Since $\theta_i \mid \theta_0, \sigma_0^2, y_i$ are conditionally independent, each θ_i can be sampled via CDA separately. For $i = 1, \dots, n$, we calibrate the binomial Polya-Gamma augmentation, leading to proposing from:

$$z_i \mid \theta_i, N_i \sim \text{PG}\left(N_i r_i, \theta_i + b_i\right)$$

$$\theta_i^* \mid z_i, y_i, N_i \sim \text{No}\left(\frac{y_i - r_i N_i/2 - z_i b_i + \theta_0/\sigma_0^2}{z_i + 1/\sigma_0^2}, \frac{1}{z_i + 1/\sigma_0^2}\right),$$

and accepting with probability

$$1 \wedge \frac{L_{r,b}(\theta_i; y_i, N_i) L(\theta_i^*; y_i, N_i)}{L_{r,b}(\theta_i^*; y_i, N_i) L(\theta_i; y_i, N_i)},$$

where

$$L_{r,b}(\theta_i; y_i, N_i) = \frac{\exp(\theta_i + b_i)_i^y}{\{1+\exp(\theta_i + b_i)\}^{N_i r_i}}$$

and $L(\theta_i; y_i, N_i) = L_{1,0}(\theta_i; y_i, N_i)$. We require $r_i \in (0, 1]$ to obtain increased or unchanged step size and $r_i > (y_i - 1)/N_i$ to have a proper $L_{r,b}(\theta_i; y_i, N_i)$.

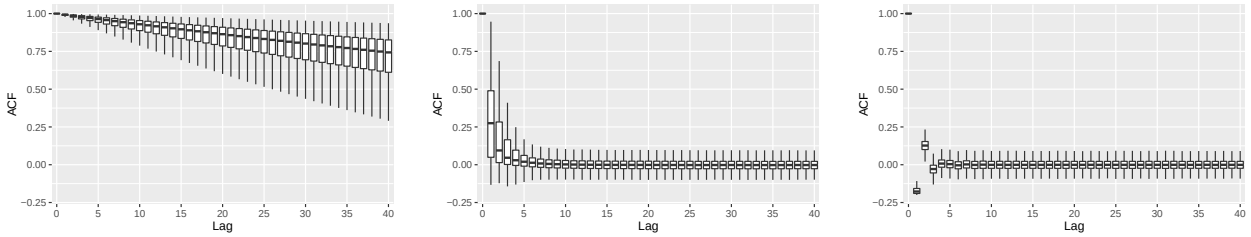
For adaptation of parameters in CDA, we use the empirical MAP $\hat{\theta}_i$ based on each $L(\theta_i; y_i, N_i)\pi(\theta_i)$ to minimize $\Delta_{\mathcal{I}}(\hat{\theta}_i)$. We set $r_i = (\hat{r}_i \vee ((y_i - 1)/N_i + \epsilon)) \wedge 1$ with $\hat{r}_i = \frac{1}{c_0} \frac{\exp(\hat{\theta}_i)}{\{1+\exp(\hat{\theta}_i)\}^2} / \left(\frac{1}{2|\hat{\theta}_i + b_i|} \tanh \frac{|\hat{\theta}_i + b_i|}{2}\right)$ and

$b_i = \log[\{1 + \exp(\hat{\theta}_i)\}^{1/r_i} - 1] - \hat{\theta}_i$, with ϵ as small positive number to ensure $L_{r,b}(\theta_i; y_i, N_i)$ is proper. The default $c_0 = 1$ leads to a high average acceptance of 0.9 for all θ_i.

After θ_i's are updated, other parameters are sampled from $\theta_0 \sim \text{No}\left((n/\sigma^2 + 1/49)^{-1}(\sum_i \theta_i/\sigma^2 - 12/49), (n/\sigma^2 + 1/49)^{-1}\right)$ and $\sigma_0^2 \sim \text{Inverse-Gamma}(n/2 - 1, \sum_i(\theta_i - \theta_0)^2/2)$.

Figure 4 shows the boxplots of the ACFs for all θ_i's. We compare the result with the original DA (Polson et al., 2013) and Hamiltonian Monte Carlo (HMC) provided by the `STAN` software (Carpenter et al., 2016). We run DA for $100,000$ steps, HMC for 2000 steps and CDA for $2,000$ steps, so that they have approximately the same effective sample size (calculated with the `CODA` package in `R`). All of the parameters mix poorly in DA; HMC and CDA leads to significant improvement with autocorrelation rapidly decaying to close to zero within 5 lags.

Shown in Table 1, CDA and HMC have very close estimates in posterior means and 95% credible intervals for the parameters; while DA has poor estimates due to critically slow mixing. The difference between HMC and CDA is that, although HMC is slightly more efficient in effective sample size per iteration (T_{eff}/T) for this model, it is much more computationally intensive and generate much less iterations than CDA, within a same budget of computing time. As the result, CDA has the most efficient computing time per effective sample.



(a) ACFs of the rate parameters θ_i using DA.

(b) ACFs of the rate parameters θ_i using CDA.

(c) ACFs of the rate parameters θ_i using HMC.

Figure 4: Boxplots of the ACFs show the mixing of the $59,792$ parameters in the hierarchical binomial model, for the original DA(Polson et al., 2013), CDA and HMC.

	DA	CDA	HMC
$\sum \theta_i/n$	-10.03 (-10.16, -9.87)	-12.05 (-12.09, -12.02)	-12.06 (-12.09, -12.01)
$\sum \theta_i^2/n$	102.25 (98.92, 105.23)	153.04 (152.06, 154.05)	153.17 (152.02, 154.29)
θ_0	-10.03 (-10.17, -9.87)	-12.05 (-12.09,-12.01)	-12.06 (-12.10, -12.01)
σ^2	1.60 (1.36, 1.82)	7.70 (7.49, 7.88)	7.71 (7.51, 7.91)
T_{eff}/T	0.0085 (0.0013 0.0188)	0.5013 (0.1101,1.0084)	0.8404 (0.5149, 1.2470)
Computing Time / T	1.2 sec	1.2 sec	6 sec
Computing Time / T_{eff}	140.4 sec	0.48 sec	1.3 sec

Table 1: Parameter estimates (with 95% credible intervals) and computing speed (ratios among computing time, effective sample sizes T_{eff} and total iterations T) of the DA, CDA and HMC in hierarchical binomial model. CDA provides parameter estimates as accurate as HMC, and is more computationally efficient than HMC.

4.2 Poisson Log-Normal Model for Web Traffic Prediction

As a second application, since one co-browsing record of each high-traffic and client pair sites usually indicates a link for user to click through to go from one to another, the total count of co-browsing can be treated as a surrogate for click-through traffic. For the advertising company with historic data, comparing the co-browsing records clients can reveal the predictive patterns among clients, hence reduce the frequency of tracking those clients that can be well predicted.

We consider a Poisson regression model. For illustration, we choose the co-browsing count of one client website as the outcome y_i, and the count of other 95 websites as the covariates $w_{i,j}$ for $i = 1 \dots n$ with $n = 59,792$ and $j = 1 \dots p$ with $p = 95$. To use them as predictors in Poisson log-normal model, we transform the count onto the log scale with $x_{i,j} = \log(w_{i,j} + 1)$. We use a normal random intercept τ_i for each i to allow over-dispersion. We the random intercepts have a common mean τ_0, so that we do not need a fixed effect intercept.

$$y_i \sim \text{Poisson}\left(\exp(x_i\beta + \tau_i)\right), \quad \tau_i \overset{iid}{\sim} \text{No}(\tau_0, \nu^2) \quad i = 1 \dots n$$

$$\beta \sim \text{No}(0, I\sigma_\beta^2), \quad \tau_0 \sim \text{No}(0, \sigma_\tau^2) \quad \nu^2 \sim \pi(\nu^2),$$

where $x_i\beta = \sum_j x_{i,j}\beta_j$. We assign a diffuse prior for β and τ_0 with $\sigma_\beta^2 = \sigma_\tau^2 = 100$. For the over-dispersion parameter ν^2, we assign a non-informative uniform prior.

We first exclude other factors that could contribute to slow mixing. In this case, when β and τ are sampled separately, the random effects $\tau = \{\tau_1, \dots, \tau_n\}$ can cause slow mixing. Instead, we sample β and τ jointly. Using $\tilde{X} = [I_n||X]$ as a $n \times (n+p)$ juxtaposed matrix, and $\eta_i = x_i\beta + \tau_i$ for the linear predictor, the model can be viewed as a linear predictor with $n + p$ coefficients, and $\theta = \{\tau, \beta\}'$ can be sampled jointly in a block. The reason for improved mixing with blocked sampling can be found in Liu (1994).

We now focus on the mixing behavior due to data augmentation. We first review the the data augmentation for Poisson log-normal model. Zhou et al. (2012) proposed to treat Poisson(η_i) as the limit of the

negative binomial NB($\lambda, \frac{\eta_i}{\lambda+\eta_i}$) with $\lambda \to \infty$, and used moderate $\lambda = 1,000$ for approximation. The limit can be simplified as (omitting constant):

$$L(\eta_i; y_i) = \frac{\exp(y_i\eta_i\}}{\exp\{\exp(\eta_i)\}} = \lim_{\lambda\to\infty} \frac{\exp(y_i\eta_i)}{\{1+\exp(\eta_i)/\lambda\}^\lambda}. \tag{12}$$

With finite λ approximation, the posterior can be sampled via Polya-Gamma augmented Gibbs sampling:

$$\begin{aligned}
z_i \mid \eta_i &\sim \mathrm{PG}(\lambda, \eta_i - \log\lambda) \quad i = 1\dots n \\
\theta \mid z, y &\sim \mathrm{No}\big((\tilde{X}'Z\tilde{X} + \begin{bmatrix} 1/\nu^2 \cdot I_n & 0 \\ 0 & 1/\sigma_\beta^2 \cdot I_p \end{bmatrix})^{-1}\{\tilde{X}'\big(y - \lambda/2 + Z\log\lambda\big) + \begin{bmatrix} \tau_0/\nu^2 1_n \\ 0_p \end{bmatrix}\}, \\
&\qquad (\tilde{X}'Z\tilde{X} + \begin{bmatrix} 1/\nu^2 \cdot I_n & 0 \\ 0 & 1/\sigma_\beta^2 \cdot I_p \end{bmatrix})^{-1}\big),
\end{aligned}$$

where $Z = \mathrm{diag}\{z_1, \dots, z_n\}$, $1_n = \{1, \dots 1\}'$ and $0_p = \{0, \dots 0\}'$.

Although this approximation enables closed-form posterior sampling, it is prone to cause slow mixing. To control approximation error, λ needs to be large, but would generate large Z and small conditional variance for $\theta \mid z, y$. In fact, for moderately large $\eta_i \approx 10$, λ needs to be at least 10^9 to make $\exp(2\eta_i)/\lambda$ close to 0, as the expansion in (12) has $(1 + \exp(\eta_i)/\lambda)^\lambda = \exp\{\exp(\eta_i) + \mathcal{O}(\exp(2\eta_i)/\lambda)\}$. This means a good control on approximation error would lead to an almost complete stop in the mixing.

We use CDA to solve this dilemma. We replace λ with $r_i\lambda$, where λ is set to 10^9 and r_i is a small fraction to allow calibration of the mixing behavior. With smaller $r_i\lambda$, we generate proposal that are much less correlated to the current state, then M-H step corrects the target to exact Poisson. Adding location adjusting parameter b, this leads to a proposal rule:

$$\begin{aligned}
z_i &\sim \mathrm{PG}(r_i\lambda, \eta_i - \log\lambda + b_i) \quad i = 1\dots n \\
\theta^* &\sim \mathrm{No}\big((\tilde{X}'Z\tilde{X} + \begin{bmatrix} 1/\nu^2 \cdot I_n & 0 \\ 0 & 1/\sigma_\beta^2 \cdot I_p \end{bmatrix})^{-1}\{\tilde{X}'\big(y - r\lambda/2 + Z\log(\lambda - b)\big) + \begin{bmatrix} \tau_0/\nu^2 1_n \\ 0_p \end{bmatrix}\}, \\
&\qquad (\tilde{X}'Z\tilde{X} + \begin{bmatrix} 1/\nu^2 \cdot I_n & 0 \\ 0 & 1/\sigma_\beta^2 \cdot I_p \end{bmatrix})^{-1}\big)
\end{aligned}$$

with acceptance probability

$$1 \wedge \prod_i \frac{L_{r,b}(\eta_i; y_i) L(\eta_i^*; y_i)}{L_{r,b}(\eta_i^*; y_i) L(\eta_i; y_i)},$$

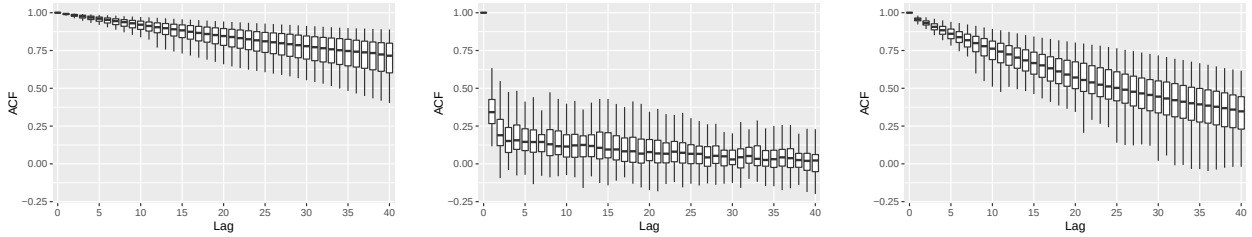
where

$$L_{r,b}(\eta_i; y_i) = \frac{\exp(y_i(\eta_i + b_i))}{\{1 + \exp(\eta_i + b_i)/r_i\lambda\}^{r_i\lambda}}, \quad L(\eta_i; y_i) = \frac{\exp(y_i\eta_i\}}{\exp\{\exp(\eta_i)\}},$$

where $L(\theta_i; y_i) = L_{\infty,0}(\eta_i; y_i)$. To ensure increased or unchanged step size compared to basic algorithm where $r = 1$, we require $r_i \in (0, 1]$; to have proper $L_{r,b}(\eta_i; y_i)$, we need $r_i > (y_i - 1)/\lambda$.

We use a tuning period to find the optimal values for calibration parameters. By minimizing $\Delta_{\mathcal{I}}(\hat{\eta})$, we set $r_i = (\hat{r}_i \vee ((y_i - 1)/\lambda + \epsilon)) \wedge 1$ with $\hat{r}_i = 1/c_0 \exp(\hat{\eta}_i)/\left(\frac{\lambda}{2|\hat{\eta}_i + b_i - \log\lambda|} \tanh\frac{|\hat{\eta}_i + b_i - \log\lambda|}{2}\right)$ where $\hat{\eta}_i$ is the empirical MAP during tuning, ϵ is a small positive number to ensure $L_{r,b}(\eta_i; y_i)$ is proper. We found that the default $c_0 = 1$ resulted in the acceptance being too low. This indicates the inverse Fisher information based approach over-estimated the step size in this finite sample. We then use $c_0 = 0.1$ so that the amount of variance increase is reduced. This increased the acceptance rate to 0.6. To reduce distance between $L_{r,b}(\eta_i; y_i)$ and $L(\eta_i; y_i)$, we used $b_i = \log[\exp\{\exp(\hat{\eta}_i - \log\lambda - \log r_i)\} - 1] - \hat{\eta}_i + \log\lambda$. After θ is updated, the other parameters can be sampled via $\tau_0 \sim \text{No}\left((n/\nu^2 + 1/\sigma_\tau^2)^{-1} \sum_i \tau_i/\nu^2, (n/\nu^2 + 1/\sigma_\tau^2)^{-1}\right)$ and $\nu^2 \sim \text{Inverse-Gamma}(n/2 - 1, \sum_i (\tau_i - \tau_0)^2/2)$.

We ran the basic DA with $\lambda = 1,000$ approximation, CDA with $\lambda = 10^9$ and HMC. We ran DA for $200,000$ steps, CDA for $2,000$ steps and HMC for $20,000$ steps so that they have approximately the same effective sample size. For CDA, we used the first $1,000$ steps for adapting r and b. Figure 5 shows the mixings of DA, CDA and HMC. Even with small $\lambda = 1,000$ in DA, all of the parameters mix poorly; HMC seemed to be affected by the presence of random effects, and most of parameters remain highly correlated within 40 lags; CDA substantially improves the mixing. Table 2 compares all three algorithms. CDA has the most efficient computing time per effective sample, and is about $30 - 300$ times more efficient than the other two algorithms.



(a) Autocorrelation of the parameters from DA.

(b) Autocorrelation of the parameters from CDA.

(c) Autocorrelation of the parameters from HMC.

Figure 5: CDA significantly improves the mixing of the parameters in the Poisson log-normal.

To evaluate the prediction performance, we use another co-browsing count table for the same high traffic and client sites, collected during a different time period. We use the high traffic co-browsing count of $p = 95$ clients $w_i^\dagger = \{w_{i,1}^\dagger, w_{i,2}^\dagger \dots w_{i,p}^\dagger\}$ with its log transform $x_i^\dagger = \log(w_i^\dagger + 1)$ to make prediction $\hat{y}_i^\dagger = \mathbb{E}_{\beta,\tau|y,x} y_i^\dagger = \mathbb{E}_{\beta,\tau|y,x} \exp(x_i^\dagger \beta + \tau_i)$ on the client site. The expectation is taken over posterior sample $\beta, \tau \mid y, x$ with training set $\{y, x\}$ discussed above. Cross-validation root-mean-squared error $\left(\sum_i (\hat{y}_i^\dagger - y_i^\dagger)^2/n\right)^{1/2}$ between

the prediction and actual count $y_i^\dagger$'s is computed. Shown in Table 2, slow mixing in DA and HMC cause poor estimation of the parameters and high prediction error, while CDA is significantly better.

	DA	CDA	HMC
$\sum \beta_j/95$	0.072 (0.071, 0.075)	-0.041 (-0.042, -0.038)	-0.010 (-0.042, -0.037)
$\sum \beta_j^2/95$	0.0034 (0.0033, 0.0035)	0.231 (0.219 0.244)	0.232 (0.216 0.244)
$\sum \tau_i/n$	-0.405 (-0.642, -0.155)	-1.292 (-2.351, -0.446)	-1.297 (-2.354, -0.451)
$\sum \tau_i^2/n$	1.126 (0.968, 1.339)	3.608 (0.696, 7.928)	3.589 (0.678, 8.011)
Prediction RMSE	33.21	8.52	13.18
T_{eff}/T	0.0037 (0.0011 0.0096)	0.3348 (0.0279, 0.699)	0.0173 (0.0065, 0.0655)
Computing Time / T	1.3 sec	1.3 sec	56 sec
Computing Time / T_{eff}	346.4 sec	11.5 sec	3240.6 sec

Table 2: Parameter estimates, prediction error and computing speed of the DA, CDA and HMC in Poisson regression model.

5 Discussion

Data augmentation is a technique routinely used to sample posterior in closed-form. Despite the convenience, it could slow down the mixing, as the step size becomes smaller than the marginal variance. At the age of "big n" data, any concentration rate difference between the two will lead to a near stop in posterior mixing as n increases. To our best knowledge, although the previous methods such as centered or non-centered re-parameterization (Papaspiliopoulos et al., 2007) and parameter-expansion (Liu and Wu, 1999) lead to some improvement in small dataset, they do not fundamentally solve the rate difference problem in large data.

Our proposed CDA method directly tackles this issue by adjusting the step size to the same order of the unconditional variance. The generated samples are used as proposals in M-H step to obtain correct posterior. As the original un-calibrated Gibbs sampler is a special case of CDA with certain r and b, CDA can be viewed as a generalized class of sampling algorithms with data augmentation. CDA adds a little cost during likelihood evaluation, but its burden is often negligible as dominated by other tasks, mostly random number generation. In this article, we demonstrate that calibration is generally applicable when the conditional $\theta \mid z$ given latent variable z belongs to the location-scale family. We expect it to be extensible to any conditional distribution with a variance or scale.

As both CDA and HMC involve M-H step, we draw some further comparison between the two. Both methods rely on finding good proposal through searching over a region far from the current state. The key difference lies in the computing efficiency in generating proposal. To generate one proposal, HMC often requires multiple iterations in Hamiltonian dynamics, which are computationally intensive; in contrast, CDA only requires one iteration of sampling. Therefore, CDA is likely to be much more efficient than HMC. Although we discuss calibration in the context of data augmentation, the idea can be more generally applicable to most Gibbs sampling to adjust its step size for faster mixing.

References

James H Albert and Siddhartha Chib. Bayesian analysis of binary and polychotomous response data. *Journal of the American statistical Association*, 88(422):669–679, 1993.

Bob Carpenter, Andrew Gelman, Matt Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Michael A Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *J Stat Softw*, 2016.

Patrick R Conrad, Youssef M Marzouk, Natesh S Pillai, and Aaron Smith. Accelerating asymptotically exact mcmc for computationally intensive models via local approximations. *Journal of the American Statistical Association*, ((to appear)), 2015.

Noel Cressie, Anne S Davis, J Leroy Folks, and J Leroy Folks. The moment-generating function and negative integer moments. *The American Statistician*, 35(3):148–150, 1981.

Martin Hairer, Jonathan C Mattingly, and Michael Scheutzow. Asymptotic coupling and a general form of harris theorem with applications to stochastic delay equations. *Probability Theory and Related Fields*, 149 (1-2):223–259, 2011.

Martin Hairer, Andrew M Stuart, Sebastian J Vollmer, et al. Spectral gaps for a Metropolis–Hastings algorithm in infinite dimensions. *The Annals of Applied Probability*, 24(6):2455–2490, 2014.

James E Johndrow, Aaron Smith, Natesh Pillai, and David B Dunson. Inefficiency of data augmentation for large sample imbalanced data. *arXiv preprint arXiv:1605.05798*, 2016.

Jun S Liu. The collapsed gibbs sampler in bayesian computations with applications to a gene regulation problem. *Journal of the American Statistical Association*, 89(427):958–966, 1994.

Jun S Liu. *Monte Carlo strategies in scientific computing*. Springer Science & Business Media, 2008.

Jun S Liu and Ying Nian Wu. Parameter expansion for data augmentation. *Journal of the American Statistical Association*, 94(448):1264–1274, 1999.

Xiao-Li Meng and David A Van Dyk. Seeking efficient data augmentation schemes via conditional and marginal augmentation. *Biometrika*, 86(2):301–320, 1999.

Sean P Meyn and Richard L Tweedie. *Markov chains and stochastic stability*. Springer Science & Business Media, 2012.

Stanislav Minsker, Sanvesh Srivastava, Lizhen Lin, and David B Dunson. Robust and scalable bayes via a median of subset posterior measures. *arXiv preprint arXiv:1403.2660*, 2014.

EWT Ngai, Yong Hu, YH Wong, Yijun Chen, and Xin Sun. The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decision Support Systems*, 50(3):559–569, 2011.

Omiros Papaspiliopoulos, Gareth O Roberts, and Martin Sköld. A general framework for the parametrization of hierarchical models. *Statistical Science*, pages 59–73, 2007.

Nicholas G Polson, James G Scott, and Jesse Windle. Bayesian inference for logistic models using pólya–gamma latent variables. *Journal of the American statistical Association*, 108(504):1339–1349, 2013.

Bala Rajaratnam and Doug Sparks. MCMC-based inference in the era of big data: A fundamental analysis of the convergence complexity of high-dimensional chains. *arXiv preprint arXiv:1508.00947*, 2015.

Gareth O Roberts and Jeffrey S Rosenthal. Coupling and ergodicity of adaptive markov chain monte carlo algorithms. *Journal of applied probability*, pages 458–475, 2007.

Gareth O Roberts and Adrian FM Smith. Simple conditions for the convergence of the Gibbs sampler and Metropolis-Hastings algorithms. *Stochastic processes and their applications*, 49(2):207–216, 1994.

Gareth O Roberts, Jeffrey S Rosenthal, et al. General state space Markov chains and MCMC algorithms. *Probability Surveys*, 1:20–71, 2004.

Sanvesh Srivastava, Volkan Cevher, Quoc Tran-Dinh, and David B Dunson. Wasp: Scalable bayes via barycenters of subset posteriors. In *AISTATS*, 2015.

Martin A Tanner and Wing Hung Wong. The calculation of posterior distributions by data augmentation. *Journal of the American statistical Association*, 82(398):528–540, 1987.

Jon Wakefield. Disease mapping and spatial regression with count data. *Biostatistics*, 8(2):158–183, 2007.

Xuerui Wang, Wei Li, Ying Cui, Ruofei Zhang, and Jianchang Mao. Click-through rate estimation for rare events in online advertising. *Online Multimedia Advertising: Techniques and Technologies*, pages 1–12, 2010.

Mingyuan Zhou, Lingbo Li, David Dunson, and Lawrence Carin. Lognormal and gamma mixed negative binomial regression. In *Machine learning: proceedings of the International Conference. International Conference on Machine Learning*, volume 2012, page 1343. NIH Public Access, 2012.