Assignment II: Advanced simulation techniques.

Computational Statistics
Henrique Ennes
Instructor: Luiz Max de Carvalho

December 5, 2021

Hand-in date: 02/12/2021.

General guidance

- State and prove all non-trivial mathematical results necessary to substantiate your arguments;
- Do not forget to add appropriate scholarly references *at the end* of the document;
- Mathematical expressions also receive punctuation;
- All computational implementations must be "from scratch", i.e., you may not employ a ready-made package to implement the technique in question. You may, however (a) employ pre-packaged routines for things like random variate generation and MCMC diagnostics and (b) use a package implementation against which to check your own.
- Please hand in a single PDF file as your final main document. Code appendices are welcome, *in addition* to the main PDF document.

Background

We have by now hopefully acquired a solid theoretical understanding of simulation techniques, including Markov chain Monte Carlo (MCMC). In this assigment, we shall re-visit some of the main techniques in the field of Simulation. The goal is to broaden your knowledge of the field by implementing one of the many variations on the general theme of simulation algorithms.

Each method/paper brings its own advantages and pitfalls, and each explores a slightly different aspect of Computational Statistics. You should pick **one** of the listed papers and answer the associated questions.

In what follows, ESS stands for effective sample size, and is similar to n_{eff} we have encountered before: it measures the number of effectively uncorrelated samples in a given collection of random variates.

Paper 3: Blocked Gibbs sampling (Tan and Hobert, 2009)

The so-called Gibbs sampler is a work horse of Computational Statistics. It depends on decomposing a target distribution into conditional densities from which new values of a given coordinate can be drawn.

One of the difficulties one might encounter with the Gibbs sampler is that it might be slow to converge, specially in highly-correlated targets. In Statistics, multilevel models (also called hierarchical or random effects) are extremely useful in modelling data coming from stratified structures (e.g. individuals within a city and cities within a state) and typically present highly correlated posterior distributions.

One way to counteract the correlation between coordinates in the Gibbs sampler is to **block** them together, and sample correlated coordinates jointly.

For this assigment you are referred to the 2009 *Journal of Computational and Graphical Statistics* paper by Tan and Hobert (Tan and Hobert, 2009).

1. Precisely describe the so-called blocked Gibbs sampler; *Hint:* you do not need to describe theoretical properties of the algorithm given in this paper; a general description of the algorithm should suffice.

2. Explain the advantages – both theoretical and practical – of a clever blocking scheme;

3. Would it be possible to apply the "simple" Gibbs sampler in this example? Why?

4. Implementation:

   (a) Implement the blocked Gibbs sampler discussed in the paper in order to fit the model of Section 1 of Tan and Hobert (2009) to the data described in Section 5 therein.

   (b) Assess convergence (or lack thereof) and mixing of the resulting chain.

   (c) Confirm your results agree with those given by the original authors up to Monte Carlo error.

5. Comment on the significance of geometric ergodicity for the blocked Gibbs sampler proposed by Tan and Hobert (2009).

Problem 1

Let us start by looking at a simple version of Gibbs sampler for computing the expectation of some variable $t(\theta)$ under a target distribution $\pi(\theta)$, $\mathbb{E}_\pi t$, assumed here to exist, where $\theta = \theta_1, ..., \theta_d$ is a multidimensional parameter vector coming from the distribution π. For such, we may use a Monte Carlo estimator $\bar{t}_N$, i.e. $\bar{t}_N = \frac{1}{N}\sum_{i=1} t(\theta^{(i)})$, where $\theta^{(i)}$ are i.i.d. realizations of π. In general, we do not know how to sample directly from π and this is where Gibbs sampler comes in: if, given the vector of parameters $\theta = (\theta_1, ..., \theta_d)$, we know how to sample directly from the conditional distributions

$$\pi(\theta_i|\theta_1, ..., \theta_{i-1}, \theta_{i+1}, ..., \theta_d) \equiv \pi(\theta_i|\theta_{-i}) \quad \text{for } i \in [1, d], \tag{1}$$

then the algorithm to create a chain of samples $\theta^{(0)}, \theta^{(1)}, ...$ works as following[1]:

1. randomly initiate $\theta^{(0)} = (\theta_1^{(0)}, ..., \theta_d^{(0)})$
2. for all $t = 1, 2, ...$
   (a) draw $\theta_1^{(t)}$ from $\pi(\theta_1|\theta_2^{(t-1)}, ..., \theta_d^{(t-1)})$
   (b) draw $\theta_i^{(t)}$ from $\pi(\theta_i|\theta_1^{(t)}, ..., \theta_{i-1}^{(t)}, \theta_{i+1}^{(t-1)}, ..., \theta_d^{(t-1)})$
   (c) draw $\theta_d^{(t)}$ from $\pi(\theta_d|\theta_1^{(t)}, ..., \theta_{d-1}^{(t)})$.

It was shown in our lectures [Rebeschini (2018)] that such method yields a Markov chain whose stationary distribution is, in fact, $\pi(\theta)$, so we might expect, under some suitable conditions which we will later explore, that this Gibbs sampler approach does converge to the target distribution π, therefore allowing us, through $\bar{t}_N$, to compute $\mathbb{E}_\pi f$. This means that the Markov chain modification to the Monte Carlo estimator allows one to define the MCMC ergodic mean estimator for $t(\theta)$, still given through $\bar{t}_N = \frac{1}{N}\sum_{i=1} t(\theta^{(i)})$, where now, $\theta^{(i)}$) are now the realizations of the Markov chain,

Nonetheless, the traditional Gibbs sampler suffers from a few severe problems. First, we need to know how to directly sample from all the conditionals $\pi(\theta_i|\theta_{-i})$, which is in several scenarios not a reasonable assumption. Indeed, we can, sometimes, mitigate this particular issue by introducing auxiliary variables, but finding good candidates to introduce is often easier said than done. Second, when some pair of components of θ are strongly correlated, it is known that the traditional Gibbs sampler approach will tend to be slow, something we will explore deeper in the next Problem. Nonetheless, we will say for now that convergence can sometimes boosted if we group the two, strongly correlated, parameters together. Such idea is exactly *blocking*: for sampling $\pi(\theta)$, instead of drawing, at each step, from $\pi(\theta_i|\theta_{i-1})$, where all θ_i are one-dimensional parameters (i.e. scalars), we join the variables θ_i together in multidimensional vectors $\eta_k = (\theta_{k_1}, ...\theta_{k_j})$, where each θ_{k_i} is a scalar parameter θ_i. Fortunately, the dimensional leap from the variables θ to η does little change to the Gibbs sampler's structure and allows us to devise the following Algorithm 1, which we shall denote by blocked Gibbs.

It is not hard to believe that, whenever the scalar Gibbs chain has $\pi(\theta)$ as stationary and is reducible – namely the MCMC estimator $\bar{t}_N(\theta)$ is known to

[1] We will be describing here only the so-called systematic sampling approach.

Algorithm 1 Blocked Gibbs Sampler

Let $\pi(\theta)$ be the distribution of interest, where the vector $\theta = (\theta_1, ..., \theta_d)$ may be written as the list $\eta = (\eta_1, ...\eta_e)$, in which each η_i corresponds to vectors of components $\theta_1, ..., \theta_d$, so that each θ_i only shows up in exactly one η_i.
Randomly assign $\eta_1^{(0)}, ..., \eta_e^{(0)}$.
for $t = 1, 2, ...$ **do**

1. draw $\eta_1^{(t)}$ from $\pi(\eta_1|\eta_2^{(t-1)}, ..., \eta_e^{(t-1)})$
2. draw $\eta_i^{(t)}$ from $\pi(\eta_i|\eta_1^{(t)}, ..., \eta_{i-1}^{(t)}, \eta_{i+1}^{(t-1)}, ..., \eta_e^{(t-1)})$
3. draw $\eta_e^{(t)}$ from $\pi(\eta_e|\eta_1^{(t)}, ..., \eta_{e-1}^{(t)})$.

end for

be strongly consistent for $t(\theta)$, the blocked version's associated estimator should converge to $t(\theta)$ as well. [Liu et al. (1994)] gives some theoretical justification for why this is, in fact, the case. We shall, in the next problem, discuss the advantages of the blocked approach. However, it is importance to notice that we still must face the constraint of knowing how to sample from all full conditionals $\pi(\eta_i|\eta_{-i})$, just as we required sampling from $\pi(\theta_i|\theta_{-i})$. Therefore, it is not the case that the blocked approach is always preferable to the unblocked version, as sometimes, we know the full-conditionals $\pi(\theta_i|\theta_{-i})$, but not $\pi(\eta_i|\eta_{-i})$.

Problem 2

Since its initial proposal in [Liu et al. (1994)], blocked Gibbs sampler has been widely used exactly in cases when some of the components of $\theta = (\theta_1, ..., \theta_d)$ have strong correlation. Figure 1 brings an informal justification for this behavior, where we analyze Gibbs sampling on a multivariate normal. If the normal's coordinate x and y have small correlation, the steps taken when sampling (black lines connecting the black little squares) are very independent of each other (left image). Nonetheless, if the relation between x and y is strong, when the x step brings the algorithm to a position x_0 in which $y|x_0$ dies quickly, the sampler can only take a small step (middle image). If, on the other hand we sample together (x, y) the "geometric" information of the target is already taken into account by the joint distribution, so steps can be extremely long and in arbitrary directions.

More theoretical work behind convergence rates of Gibbs sampler, with notable results found both in the seminal paper of [Liu et al. (1994)], and on the more recent (and, perhaps, accessible) work of [Roberts and Sahu (1997)], may be used to further elucidate this idea. We start by formally defining the rate of convergence of a chain output at time N, $P^N t(\theta^{(0)}) = t(\theta)|\theta^{(0)}$, for some square integrable function $t(\theta)$ in the target distribution π, through the minimum $\rho \in \mathbb{R}^+$ such that, for all $r \geq \rho$

$$\lim_{N\to\infty} \frac{\mathbb{E}_\pi[\mathbb{E}_\pi P^N t(\theta^{(0)}) - \mathbb{E}_\pi t(\theta)]^2}{r^N} = 0.$$

Consequently, the smaller the convergence rate ρ, the "faster" the MCMC (ergodic mean) estimator $\bar{t}_N(\theta) = \frac{1}{N}\sum_{i=1}^N t(\theta^{(i)}$, if strongly consistent[2], will ap-

[2]Recall that, for a MCMC estimator to be strongly consistent for some integrable func-

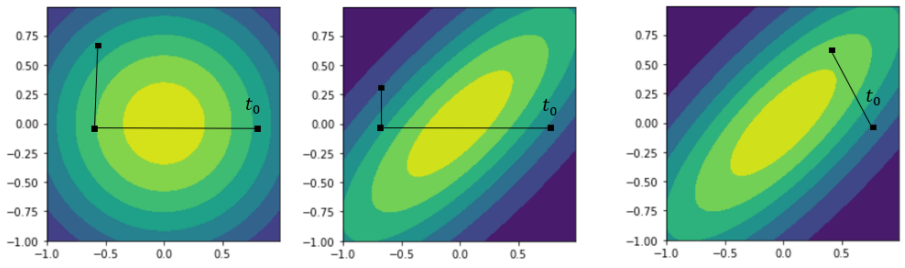

Figure 1: Images of Gibbs sampling of a multivariate normal distribution on the horizontal coordinate x and vertical coordinate y. The leftmost distribution has independent x and y coordinates, whereas the other two have $\text{cor}(x, y) = 0.8$. All three images indicate a Gibbs sampling process starting at the little black square t_0, with the first two showing the behavior of simple samplers and the right one of blocking x and y together. This illustration is intended to be interpreted as an informal suggestion of the impact of the covariance of the target's coordinates in the step size (mixing rate) of Gibbs samplers.

proach the expectation $\mathbb{E}_\pi t(\theta)$. In practical Markov chains notation, this is translated in the decaying speed of autocorrelation factors (ACF) for consecutive lags of samples of the chain, which implies faster mixing of the chain.

Fortunately, [Liu et al. (1994), Liu (2001)] establish that, for *reversible* chains, the convergence rate ρ is related to the spectral radius – i.e. largest magnitude of eigenvalues – of a matrix proportional to the targets' dispersion Σ; unfortunately, systematic scan Gibbs sampler is *not* reversible Rebeschini (2018). Nonetheless, [Roberts and Sahu (1997)] discusses some special cases in which the converge rate of a Gibbs chain may be determined for some target distributions. In particular, they were able to show for a multidimensional normal that, when the covariances magnitudes for two coordinates are large, biding them together will often accelerate convergence.

This is better illustrated with the example of a 3-dimensional normal $N(0, \Sigma)$, where

$$\Sigma = \begin{pmatrix} 1 & a & 0.5 \\ a & 1 & 0.5 \\ 0.5 & 0.5 & 1 \end{pmatrix}.$$

[Roberts and Sahu (1997)] showed, theoretically, that blocking the first two coordinates will only boost convergence if $a > 0.25$. If we extrapolate on this, dropping the normality assumption, we shall only take blocking as useful if the coordinates to be tied have strong correlation. Furthermore, as [Venugopal and Gogate (2012)] points out, since the blocked components η_i often require more expensive computations than the separated θ_i, we may, in practice, end up with slower estimations, even if these chain mixes faster.

tion, it is enough for the chain used to calculate it to be π irreducible and have stationary distribution π [Gamerman and Lopes (2006)].

Problem 3

We are finally in position to understand the example problem that [Tan and Hobert (2009)] discuss. The model analyzed by the authors is the so called one-way random effect, given by

$$y_{ij} = \theta_i + \epsilon_{ij} = \mu + u_i + \epsilon_{ij} \quad i = 1, ..., q \; j = 1, ..., m_i, \tag{2}$$

where θ_i are i.i.d. from $N(\mu, \sigma_\theta^2)$, u_i i.i.d from $N(0, \sigma_\theta^2)$ (i.e. $u_i = \theta_i - \mu$), and ϵ_{ij} are i.i.d. from $N(0, \sigma_e^2)$. This kind of model is often applied when measurements may be grouped in q sets, each of size m_i, and, besides being influenced by the whole group behavior, expressed through θ_i, they are also affected by some measurement specific random effect, encoded in ϵ_{ij}. Of particular interest is the comparison between the between groups variance, σ_θ^2 and the within group variance, σ_e^2: the bigger σ_e is compared to σ_θ, the less confident we are in explaining some measurement based on group division. In their paper, Tan and Hobert explore a particular application of the one-way random effect in describing the level of styrene measured $m_i = m = 3$ times among $q = 13$ workers of a factory, where each worker represents a group described θ_i and every measurement uncertainty is predicted by ϵ_{ij}.

From a Bayesian perspective, to estimate the parameters $(\theta_1, ..., \theta_q, \mu, \sigma_\theta^2, \sigma_e^2) \equiv (\theta, \mu, \sigma_\theta^2, \sigma_e^2)$, we introduce some prior $p(\theta, \mu, \sigma_\theta^2, \sigma_e^2)$ and write the posterior as

$$\pi(\theta, \mu, \sigma_\theta^2, \sigma_e^2 | y_j) \equiv \pi(\theta, \mu, \sigma_\theta^2, \sigma_e^2) \propto p(\theta, \mu, \sigma_\theta^2, \sigma_e^2) f(y_j | \theta, \mu, \sigma_\theta^2, \sigma_e^2), \tag{3}$$

then reducing the inference problem to sampling from the distribution above. The authors, inspired by [Hobert and Casella (1996)], choose to use the prior

$$p(\theta, \mu, \sigma_\theta^2, \sigma_e^2) = p(\theta | \mu, \sigma_\theta^2, \sigma_e^2) \pi_{a,b}(\mu, \sigma_\theta^2, \sigma_e^2),$$

where $p(\theta | mu, \sigma_\theta^2, \sigma_e^2) = N(\mu, I \times \sigma_\theta^2)$ and $\pi_{a,b}$ is the family of improper distributions

$$\pi_{a,b} = (\sigma_\theta^2)^{-(a+1)} (\sigma_e^2)^{-(b+1)},$$

where a and b are some known hyperparameters. Particularly, they advocate for setting $a = -1/2$ and $b = 0$, calling $\pi_{-1/2,0}$ the standard diffuse prior. Fortunately, even the choice of improper priors $\pi_{a,b}$ does not spoil the "properness" of the target $\pi(\theta, \mu, \sigma_\theta, \sigma_e)$ for the standard diffuse prior as, given $q \geq 3$ and $m_i = m$ for all i, the posterior becomes

$$\pi(\theta, \mu, \sigma_\theta, \sigma_e) \propto \prod_{i=1}^{q} \prod_{j=1}^{m} \sigma_\theta^{-1} \sigma_e^{-3/2} \exp\left(-\frac{1}{2\sigma_e^2} (y_{ij} - \theta_i)^2 \right) \exp\left(-\frac{1}{2\sigma_\theta^2} (\theta_i - \mu)^2 \right),$$

which *is* a distribution [Hobert and Casella (1996)]. Simple calculations allow

Table 1		
Function	MCMC Estimation	MCSE
$\mathbb{E}_\pi \sigma_\theta^2$	0.189	4.21×10^{-7}
$\mathbb{E}_\pi \sigma_e^2$	0.619	1.53×10^{-7}
$\mathbb{E}_\pi \frac{\sigma_\theta^2}{\sigma_\theta^2+\sigma_e^2}$	0.212	4.35×10^{-7}

for writing the full conditionals[3]

$$\begin{aligned}
\pi(u|\mu,\sigma_\theta,\sigma_e,y_{i,j}) &\sim N\left(\frac{q\sigma_\theta^2}{m\sigma_t^2+\sigma_e^2}\sum_i \overline{y}_i - u, \frac{\sigma_\theta^2\sigma_e^2}{m\sigma_\theta^2+\sigma_e^2}\right)\\
\pi(\mu|u,\sigma_\theta,\sigma_e,y_{i,j}) &\sim N\left(\frac{1}{q}\sum_i \overline{y}_i - \mu, \frac{\sigma_e^2}{mq}\right)\\
\pi(\sigma_\theta|u,\mu,\sigma_e,y_{i,j}) &\sim IG\left(\frac{q}{2}-\frac{1}{2}, \frac{1}{2}\sum_i(\theta_i-\mu)^2\right)\\
\pi(\sigma_\theta|u,\mu,\sigma_e,y_{i,j}) &\sim IG\left(\frac{q\times m}{2}, \frac{1}{2}\sum_{ij}(y_{ij}-\theta_i)^2\right),
\end{aligned} \tag{4}$$

where $\overline{y}_i = \frac{1}{q}\sum_j y_{ij}$ [Hobert and Casella (1996)].

Table 1 shows MCMCM estimates for $\mathbb{E}\sigma_\theta^2$, $\mathbb{E}\sigma_e^2$, and $\mathbb{E}\frac{\sigma_\theta^2}{\sigma_\theta^2+\sigma_e^2}$ from 697,869 iterations of the algorithm defined by equation (4), with code available at `https://github.com/HLovisiEnnes/Practice_problems/tree/main/Assignment_2_Computational%20Statistics`. We have not showed confidence intervals, for reasons that will be made clear in Problem 5. Nevertheless, notice that all obtained values fall within the expected 95% regions obtained by [Tan and Hobert (2009)], Section 5. The estimated Gelman-Rubin $\hat{R}$ in σ_θ, was of $\hat{R} = 1.053$ for 100 parallel chains of 1000 points each, bellow the literature suggestion of 1.10 as an indication of convergence of the estimator. Other diagnostic techniques will also be discussed in the next Problem.

Notwithstanding the initial good results, such a naive approach may suffer from a large autocorrelation between the variables θ and μ, and it is possible that we might improve convergence by blocking some variables according to what we have discussed. In the next problem, we will describe this blocking, initially proposed by Tan and Hobert (2009), as well as compare both techniques.

Problem 4

Tan and Hobert (2009) derive the following blocked version for sampling the parameters[4] of the random effect distribution with improper priors, where $\zeta =$

[3]Technically, the distribution for u is not in full conditional form, as that would be $\pi(u_i|u_{-i}\mu,\sigma_\theta,\sigma_e,y_{i,j})$ for all i. However, as each u_i is independent of the others, we may sample as a multidimensional normal distribution of scalar covariance.

[4]Notice that compared to equation (4), equation (5) samples in terms of u, not θ. This is simply to make computations easier.

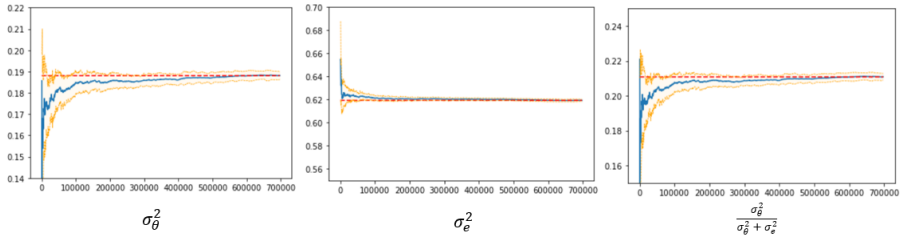

Figure 2: Traceplots of ergodic averages of MCMC estimators of $\mathbb{E}\sigma_\theta^2$, $\mathbb{E}\sigma_e^2$ and $\mathbb{E}\frac{\sigma_\theta^2}{\sigma_\theta^2+\sigma_e^2}$ for 697,869 sampled points using Gibbs blocking.

(η, μ) and $\sigma = (\sigma_\theta^2, \sigma_e^2)$,

$$\begin{aligned}
\pi(\zeta|\sigma, y_{i,j}) &\sim N\Big(\mathbb{E}\zeta, \mathrm{Var}\zeta\Big) \\
\pi(\sigma_\theta|u, \mu, \sigma_e, y_{i,j}) &\sim IG\bigg(\frac{q}{2} + a, \frac{1}{2}\sum_i(\theta_i - \mu)^2\bigg) \\
\pi(\sigma_\theta|u, \mu, \sigma_e, y_{i,j}) &\sim IG\bigg(\frac{q \times m}{2} + b, \frac{1}{2}\sum_{ij}(y_{ij} - \theta_i)^2\bigg),
\end{aligned} \tag{5}$$

with

$$\begin{aligned}
\mathbb{E}\theta_i &= \frac{\sigma_e^2}{\sigma_e^2 + m\sigma_\theta^2} \times \frac{1}{q}\sum_i \overline{y}_i + \frac{\sigma_\theta^2 m \overline{y}_i}{\sigma_e^2 + m\sigma_\theta^2} \\
\mathbb{E}\mu &= \frac{1}{q}\sum_i \overline{y}_i \\
\mathrm{Var}\theta_i &= \frac{\sigma_e}{\sigma_e^2 + m\sigma_\theta^2} \times \bigg(\sigma_\theta + \frac{\sigma_e}{mq}\bigg) \\
\mathrm{Var}\mu &= \frac{\sigma_e^2 + m\sigma_\theta^2}{mq} \\
\mathrm{Cov}(\theta_i, \mu) &= \frac{\sigma_e^2}{mq} \\
\mathrm{Cov}(\theta_i, \theta_j) &= \frac{\sigma_e^2}{mq(\sigma_e^2 + m\sigma_\theta^2)}.
\end{aligned}$$

Again, an implementation of this process can be found[5] at `https://github.com/HLovisiEnnes/Practice_problems/tree/main/Assignment_2_Computational%20Statistics`. Figure 2 shows traceplots of the ergodic averages as a function

[5]I have implemented up from scratch a blocked Gibbs sampler according to [Tan and Hobert (2009)] description. However, I was curious to see how the matrix-oriented code they have supplemented online in R would perform in relation to my implementation, so I translated it to Python. Shockingly, even Python was able to profit much from matrix operations: the translated code performed in 42 seconds what my own took 122. The translated version is also available at the Jupyter Notebook.

Table 2			
Function	MCMC Estimation	MCSE	95% Confidence Interval
$\mathbb{E}_\pi \sigma_\theta^2$	0.189	8.39×10^{-7}	$(0.186, 0.190)$
$\mathbb{E}_\pi \sigma_e^2$	0.619	2.19×10^{-7}	$(0.618, 0.620)$
$\mathbb{E}_\pi \frac{\sigma_\theta^2}{\sigma_\theta^2+\sigma_e^2}$	0.211	9.50×10^{-7}	$(0.209, 0.213)$

of the number of iterations for estimators of $\mathbb{E}\sigma_t, \mathbb{E}\sigma_e$, and $\mathbb{E}\frac{\sigma_\theta^2}{\sigma_\theta^2+\sigma_e^2}$, respectively, together with 95% confidence regions. In making these plots, we used only a single chain run of 697,869 sampled points – the same that [Tan and Hobert (2009)] used in their computations – with a 30 iterations warm-up. From the graph, we are able to identify very good mixing – confidence level within the 0.16-0.20 range – from 300,000 sampled points, which, even though might sound slow, given my experience with HMC from the last assignment, is still comparable to what [Tan and Hobert (2009)] saw[6]. Notice that, as far as we can tell from visual techniques, the sampler seems to converge for all parameters. In fact, the value estimated for Gelman- Rubin $\hat{R}$ in σ_θ, was of $\hat{R} = 1.041$ for 100 parallel chains of 1000 points each, testifying the so called convergence [Gelman and Rubin (1992)]. Finally, the bar plot at bottom right of Figure 3 shows the autocorrelation factors for a 10,000 iterations-long chain, which do seem to decrease faster than the hoped $1/lag^2$ ratio.

Table 2 shows the final ergodic avarage, 95% confidence interval and MCSE for each of the estimated parameters for [Tan and Hobert (2009)] configuration. Our estimations turned out to be a little shifted from the authors', nonetheless still within their confidence level.

Surprisingly, the results of the blocked version turned out not to be so much different from the simple Gibbs', as it can be seen in Figure 3. Although both the egordic mean traceplot and the autocorrelation indicate slightly better performance for the blocked case, the improvements are not significant enough to justify the use of the second in detriment of the first. This seems to suggest that the autocorrelations between θ and μ, are small. In fact, these could be directly calculated from the simple sampler and the correlation factors all turned out to be about -0.22, not large enough to indicate good evidence of beneficial sampling. In fact, it is not hard to admit that these two samplers work, empirically speaking, within same convergence levels.

However, if the blocked Gibbs achieve only slightly better results than the original formulation, why do [Tan and Hobert (2009)] even bother investigating it? I was myself unsure on how to answer this question, until I realized that their proof of geometric ergodicity is based on considering the *joint* distribution η for the Gibbs chain, not the scalars μ and θ. Therefore, the whole justification for the blocking scheme in this scenario comes not exactly from some speeding in convergence, but from the theoretical guarantee of its rate, guarantee which is crucial on its own, as we will now see.

[6]Their traceplots of the estimator is shown as a function of number of "torus" R, rather than number of iterations. In the next problem, we will shortly describe what these torus mean.

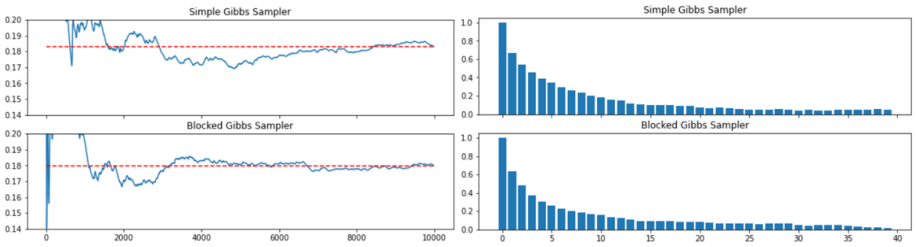

Figure 3: The plots on the left are ergodic average traceplots for 10,000 samples with no burnout time for both simple and blocked Gibbs, with the final empirical value in red. The images on the right are bar plots for the autocorrelation factors as a function of the number of lags, again for both simple and blocked Gibbs.

Problem 5

In the last Problem, we saw that even though, empirically, the simple version of the Gibbs sampler seems to converge at rate similar to blocked's, we cannot prove geometric ergodicity in the weaker scenario. Is such a property useful enough to justify the extra computational work? It turns out that it, in fact, is. [Gamerman and Lopes (2006)] establish a Central Limit Theorem type for an ergodic mean MCMC estimator $\bar{t}_N(\theta)$ of some function of the parameters $t(\theta)$, for which $t^{2+\alpha}$, where $\alpha > 0$, is integrable on the target distribution $\pi(\theta)$, namely

$$\sqrt{n}\frac{\bar{t}_N - \mathbb{E}_\pi(t)}{\sqrt{\mathrm{Var}_\pi(t)}} \xrightarrow{d} N(0,1). \tag{6}$$

Equation (6) above is the main tool which allows establishing a confidence interval for $\bar{t}_N$: we may consistently estimate $\mathrm{Var}_\pi(t)/n$ through the root of the MCSE, and, by small modifications in the assymptotic behavior to take into account the extra uncertainty (this boils down in substituting the normal for a t-distribution), we find an assymptotic 95% region. Unfortunately, the theorem which establishes (6) requires the chain which generates $\bar{t}_N$ to be *geometric ergodic*. Consequently, if we ever want to be certain of our confidence intervals as valid, we either require geometric ergodicity, or take the more complicated route of showing some CLT-type theorem for the particular chain we are using. This means that, even though empirical results seem to indicate the simple Gibbs chain to be geometric ergodic – as it apparently converges as fast as the blocked version, the lack of a proof elevates the blocked approach to a more useful position. This truly is the reason why we showed no confidence intervals in Table 1, but only in Table 2.

Before we finish, it is noteworthy that even though [Tan and Hobert (2009)] also use the geometric ergodicity of the blocked chain to establish a confidence interval to their parameters, the technique for actually estimating these intervals is quite different. Instead of using the MCSE, as suggested by [Gamerman and Lopes (2006)], they choose to use a new statistic γ^2, which makes most sense in their sampling process, called *regenerative sampling*. This method is somewhat related to thickening, but instead of ignoring some of the samples, we consider groups ("torus") of estimations that are replicas of each other.

However, since this approach, version, in my opinion[7], makes interpretation of the parameters somewhat more involved than using the full chain, we opted for the latter technique.

[7]And, as far as I can tell from its smaller usage, in literature's as well.

Bibliography

Gamerman, D. and Lopes, H. F. (2006). *Markov chain Monte Carlo: stochastic simulation for Bayesian inference*. Taylor & Francis.

Gelman, A. and Rubin, D. B. (1992). A single series from the gibbs sampler provides a false sense of security. *Bayesian statistics*, 4:625–631.

Hobert, J. P. and Casella, G. (1996). The effect of improper priors on gibbs sampling in hierarchical linear mixed models. *Journal of the American Statistical Association*, 91(436):1461–1473.

Liu, J. S. (2001). *Monte Carlo strategies in scientific computing*, volume 10. Springer.

Liu, J. S., Wong, W. H., and Kong, A. (1994). Covariance structure of the gibbs sampler with applications to the comparisons of estimators and augmentation schemes. *Biometrika*, 81(1):27–40.

Rebeschini, P. (2018). Advanced simulation methods. http://www.stats.ox.ac.uk/ rebeschi/teaching/AdvSim/18/index.html.

Roberts, G. O. and Sahu, S. K. (1997). Updating schemes, correlation structure, blocking and parameterization for the gibbs sampler. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 59(2):291–317.

Tan, A. and Hobert, J. P. (2009). Block Gibbs sampling for Bayesian random effects models with improper priors: Convergence and regeneration. *Journal of Computational and Graphical Statistics*, 18(4):861–878.

Venugopal, D. and Gogate, V. (2012). On lifting the gibbs sampling algorithm. *Advances in Neural Information Processing Systems*, 25:1655–1663.