CU-MSDSp Technical Underpinnings

John Chavis

December 3, 2020

1 Introduction

We begin with a description of problem setting for the Reversible Jump Markov Chain Monte Carlo (RJMCMC) algorithm. Let $\boldsymbol{x}$ be a vector of observed data, $\{\mathcal{M}_k, k \in \mathcal{K}\}$ with $\mathcal{K} = \{1, 2, 3....\}$ represent the countable collection of candidate models that describe $\boldsymbol{x}$, and that are indexed by k. Each such model is instantiated by a collection of n_k parameters, formed as a vector $\phi_k \in \mathbb{R}^{n_k}$. We note that the dimension, n_k, can vary for each model. Let a Markov state be $\theta_k = (k, \phi_k)$ over the state space $\Theta = \cup_{k \in \mathcal{K}}(\{k\} \times \mathbb{R}^{n_k})$. The target density, $\pi(\theta_k|\boldsymbol{x})$, resulting from application of RJMCMC is the joint posterior of $\theta_k = (k, \phi_k)$ conditioned on the data $\boldsymbol{x}$. Bayes's Theorem states that the joint posterior distribution is the product of the likelihood, $p(\boldsymbol{x} \mid k, \phi_k)$, and joint prior, $p(k, \phi_k) = k$:

$$\pi(\theta_k \mid \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid k, \phi_k)k}{\sum_{k' \in \mathcal{K}} \int_{\mathbb{R}^{n_{k'}}} \mathscr{L}(\boldsymbol{x}|k', \vec{\phi}_{k'}) p(\vec{\phi}_{k'} \mid k') p(k') d\vec{\phi}_{k'}} \tag{1}$$

Note that the denominator serves as a normalizing term to yield a unit measure across the entire probability space.

The RJMCMC Algorithm relies on satisfaction of the *detailed balance* condition, in order to construct a time-reversible Markov Chain. For all Borel sets $\mathcal{A} \times \mathcal{B} \subset \Theta$ and general Markov Transition Kernel, P, the detailed balance condition is expressed as:

$$\sum_{\mathcal{K}} \int_{(\theta_k, \theta_{k'}) \in \mathcal{A} \times \mathcal{B}} \pi(d\theta_k \mid \boldsymbol{x}) P(d\theta_{k'} \mid \theta_k) d\theta_k d\theta_{k'} = \sum_{\mathcal{K}} \int_{(\theta_k, \theta_{k'}) \in \mathcal{A} \times \mathcal{B}} \pi(d\theta_{k'} \mid \boldsymbol{x}) P(d\theta_k \mid \theta_{k'}) d\theta_k d\theta_{k'} \tag{2}$$

indicating that moves from θ_k to $\theta_{k'}$ are equally probable as moves from $\theta_{k'}$ to θ_k under the target density. One moves from state space $\theta_k = (k, \phi_k) \in \mathcal{A}$ to $\theta_{k'} = (k', \phi_{k'}) \in \mathcal{B}$ according to a *proposal distribution*, $P(\theta_{k'} \mid \theta_k)$.

This proposed move, under P, must either be accepted or rejected through an acceptance probability $\alpha(\theta_k, \theta_{k'})$, in order to ensure that detailed balance holds. The detailed balance condition can now be re-written as (Green (1995)):

$$\sum_{\mathcal{K}} \int_{(\theta_k, \theta_{k'}) \in \mathcal{A} \times \mathcal{B}} \pi(d\theta_k \mid \boldsymbol{x}) P(d\theta_{k'} \mid \theta_k) \alpha(\theta_k, \theta_{k'}) d\theta_k d\theta_{k'} = \sum_{\mathcal{K}} \int_{(\theta_k, \theta_{k'}) \in \mathcal{A} \times \mathcal{B}} \pi(d\theta_{k'} \mid \boldsymbol{x}) P(d\theta_k \mid \theta_{k'}) \alpha(\theta_{k'}, \theta_k) d\theta_k d\theta_{k'} \tag{3}$$

which can be shown to hold if the acceptance probability is given as:

$$\alpha(\theta_k, \theta_{k'}) = \min\left\{1, \frac{\pi(\theta_{k'} \mid \boldsymbol{x}) P(\theta_k \mid \theta_{k'})}{\pi(\theta_k \mid \boldsymbol{x}) P(\theta_{k'} \mid \theta_k)}\right\} \tag{4}$$

The full technical theoretical foundation of RJMCMC is beyond the scope of this paper but we refer users to several works (Ye et al. (2009); Green (1995); Sisson (2005)).

1.1 Designing Proposal Move

Designing the proposal move can be challenging due to the potentially varied dimensions that an RJMCMC sampler has to transverse in the state space. Green (1995) argued that this problem can be addressed by defining $\binom{|\mathcal{K}|}{2}$ bijections, $h_{k,k'}(\cdot)$, to describe the relationship between states θ_k and $\theta_{k'}$ with dimensions $n_k, n_{k'}$ respectively. The dimensions between these two states must match in order for these mappings to hold. Dimension matching can be accomplished by generating a random variable vector, $\vec{u} \sim g$ with dimension m_k and $\vec{u'} \sim g'$ with diemsions $m_{k'}$ such that the $n_k + m_k = n_{k'} + m_{k'}$. We redefine transition matrix be $P(\theta_{k'} \mid \theta_k) = j(k' \mid k)g(\vec{u})$ to represent the proposed model move and parameters generated to match dimensions. The acceptance probability now becomes:

$$\alpha(\theta_k, \theta_{k'}) = \min\left\{1, \frac{\pi(\theta_{k'} \mid \boldsymbol{x})j(k \mid k')g'(\vec{u'})}{\pi(\theta_k \mid \boldsymbol{x})j(k' \mid k)g(\vec{u})} \left|\frac{\partial_{g_{k\to k'}}(\theta_k, \boldsymbol{u})}{\partial(\theta_k, \boldsymbol{u})}\right|\right\} \tag{5}$$

where the last term represents the Jacobian to account for the change between states (Green (1995), Sisson (2005)).

2 CU-MSDSp Theoretical Foundation

At a high level CU-MSDSp differs from the standard RJMCMC algorithm by first independently approximating the model parameter posterior distribution, $\pi(\phi_k \mid k, \boldsymbol{x})$, for each of the models via the Stan Stan *application programming interface (API) for C++* (Carpenter et al. (2017)). These samples, or *gold standard chains* are then used to asses the model space, $\pi(k \mid \boldsymbol{x})$. The samples generated by Stan *API* and model space assessment, are realized via our modifications to Green's RJMCMC algorithm ((Green (1995)). In this modified case, we propose within-model parameter updates based on the gold standard chains, in lieu of a user-defined proposal distribution, as would be needed in the standard RJMCMC algorithm. As such we must derive a proper acceptance probability.

2.1 Acceptance Probability

When performing dependent sampling over the state space of our Markov chains of varying dimensions, it is vital to ensure that detailed balance is satisfied. A properly formulated acceptance probability guarantees this, and so we now outline the steps leading to the construction of the acceptance probability used in CU-MSDSp.

We begin by stating what is required for CU-MSDSp to estimate the model posterior distribution $\pi(k \mid \boldsymbol{x})$ given the samples generated from the within-model parameter posterior distribution, $\pi(\phi_k \mid k, \boldsymbol{x})$.

Applying Bayes' Theorem:

$$\pi(k \mid \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid k)p(k)}{\sum\limits_{\mathcal{K}} p(x|k)} \tag{6}$$

We then apply Bayes theorem on the within-model parameter posterior distribution for a fixed ϕ_k:

$$\pi(\phi_k \mid k, \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid \phi_k, k)p(\phi_k|k)}{p(\boldsymbol{x} \mid k)} \tag{7}$$

$$\implies p(\boldsymbol{x} \mid k) = \frac{p(\boldsymbol{x} \mid \phi_k, k)p(\phi_k|k)}{\pi(\phi_k \mid k, \boldsymbol{x})} \tag{8}$$

Substituting Equation 8 into Equation 6 gives:

$$\pi(k \mid \boldsymbol{x}) = \frac{p(\boldsymbol{x} \mid \phi_k, k)p(\phi_k|k)}{\pi(\phi_k \mid k, \boldsymbol{x})\sum\limits_{\mathcal{K}} p(x|k)} \tag{9}$$

The sought for acceptance probability emanates from the consideration of detailed balance:

$$\sum_{\mathcal{K}} \pi(k \mid \boldsymbol{x}) P(k' \mid k) \alpha(k, k') = \sum_{\mathcal{K}} \pi(k' \mid \boldsymbol{x}) P(k \mid k') \alpha(k', k) \tag{10}$$

Similar to before, our transition kernel is defined as $P(\theta_{k'} \mid \theta_k) = j(k' \mid k)$; representing a proposed model move to state k' from state k. The resulting acceptance probability is:

$$\alpha(k, k') = \min \left\{ 1, \frac{j(k \mid k') p(\boldsymbol{x} \mid \phi_{k'}, k') p(\phi_{k'} \mid k') \pi(\phi_k \mid k, \boldsymbol{x})}{j(k' \mid k) p(\boldsymbol{x} \mid \phi_k, k) p(\phi_k \mid k) \pi(\phi_{k'} \mid k', \boldsymbol{x})} \right\} \tag{11}$$

We observe that the acceptance probability is similar to the one described in Section 1.1, except for the addition of a within-model parameter posterior distribution: used as our proposal distribution $\vec{u} \sim g$. Drawing from the within-model parameter posterior distribution ensures dimension matching as $n_k + n_{k'} = n_{k'} + n_k$ (given that state k and k' have dimension $n_k, n_{k'}$, respectively). Furthermore, we observe that the Jacobian does not appear since it takes on the form of the identity, as a result of the convenient mapping $h_{k,k'}(\cdot)$, where the $\vec{u}$ corresponds to the exact parameters in ϕ_k and similarly $\vec{u'}$ maps exactly to the parameters in $\phi_{k'}$.

In light of the foregoing, our parallel RJMCMC approach focuses purely on fixed dimension MCMC techniques, for both estimating the within-model parameter posterior, as well as the model posterior distributions. Together, these form our joint posterior distribution, as $\pi(\theta_k \mid \boldsymbol{x}) = \pi(\phi_k \mid k, \boldsymbol{x}) \pi(k \mid \boldsymbol{x})$. Also, since the Jacobian of our bijective diffeomorphism between proablity spaces is the identity, we only need to focus on finding an appropriate approximation for the within model parameter posterior distribution given the samples from the gold standard chains.

Finally, instead of using pairwise model comparisons, as is common in the literature, we employ a special application of Green's original RJMCMC method to estimate the model space across all models. We also take advantage of the entire gold-standard chains and remove the need to select within-model samples with high probability. (Ye et al. (2009), Chib and Jeliazkov (2001), Green and Hastie (2009), Chib (1995), Carlin and Louis (2010)).

To summarize, the parallel RJMCMC algorithm has two stages: 1) generate gold standard chains that are drawn from the within-model posterior distribution, $\pi(\phi_k \mid k, \boldsymbol{x})$ and 2) use the pre-computed gold standard chains to ultimately estimate the joint model and within-model parameter joint posteiror distribution $\pi(\theta_k \mid \boldsymbol{x}) = \pi(\phi_k \mid k, \boldsymbol{x}) \pi(k \mid \boldsymbol{x})$ via CU-MSDSp.

References

Bradley P Carlin and Thomas A Louis. *Bayes and empirical Bayes methods for data analysis*. Chapman and Hall/CRC, 2010.

Bob Carpenter, Andrew Gelman, Matthew D Hoffman, Daniel Lee, Ben Goodrich, Michael Betancourt, Marcus Brubaker, Jiqiang Guo, Peter Li, and Allen Riddell. Stan: A probabilistic programming language. *Journal of statistical software*, 76(1), 2017.

Siddhartha Chib. Marginal likelihood from the gibbs output. *Journal of the american statistical association*, 90(432):1313–1321, 1995.

Siddhartha Chib and Ivan Jeliazkov. Marginal likelihood from the metropolis–hastings output. *Journal of the American Statistical Association*, 96(453):270–281, 2001.

Peter J Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

Peter J Green and David I Hastie. Reversible jump mcmc. *Genetics*, 155(3):1391–1403, 2009.

Scott A Sisson. Transdimensional markov chains: A decade of progress and future perspectives. *Journal of the American Statistical Association*, 100(471):1077–1089, 2005.

Jing Ye, Andrew Wallace, and John Thompson. Parallel markov chain monte carlo computation for varying-dimension signal analysis. In *2009 17th European Signal Processing Conference*, pages 2673–2677. IEEE, 2009.