Supplementary Materials: MCMC Algorithm for Posterior Sampling

Bradley J. Barney

(Kennesaw State University),

Federica Amici

(Max Planck Institute for Evolutionary Anthropology),

Filippo Aureli

(Universidad Veracruzana),

Josep Call

(Max Planck Institute for Evolutionary Anthropology), and

Valen E. Johnson

(Texas A&M University)

In each of the updating steps, the algorithm uses the current values of various parameters/latent variables. Any quantity that is conditioned on is implicitly using the most recent value of that quantity unless stated otherwise. When it is necessary to distinguish between the current value and the proposed value of an arbitrary quantity ξ, ξ^c will represent the current value and ξ^* will represent the proposed value. When it is important to identify the iteration number, $\xi^{(m)}$ represents the value of ξ at the end of the m^{th} iteration. At the end of each iteration, $\xi^{(m)}$ must equal ξ^c because ξ^c is updated whenever ξ changes.

Different techniques are used to update $\mathbf{z}$, depending on whether the latent variables correspond to binomial or rank data. First, consider binomial data. There is an intricate relationship between the cutpoint parameters $\boldsymbol{\tau}$ and the latent variables $\mathbf{z}$. It is possible to update each cutpoint parameter using its full conditional (a truncated Cauchy). Then the latent variables for binomial data can be updated using their full conditionals (truncated normals). These full conditionals are now stated; for convenience, parameters and latent variables not appearing in the functional form of the full conditionals are not explicitly stated.

$$\pi(\tau_j|\mathbf{z},\mathbf{y}) \propto Cauchy(m_j, s_j^2) 1(\tau_j \in (l_j^c, u_j^c)), \quad \forall j : B(j) = 1 \tag{1}$$

$$\begin{aligned}\pi(z_{ijt}|\tau_j, \mathbf{u}_1, \ldots, \mathbf{u}_P, \sigma_\epsilon^2, \mathbf{y}) \propto{}& N(u_{1,l(1,i,j,t)} + \cdots + u_{P,l(P,i,j,t)}, \sigma_\epsilon^2) \\ &\times 1(z_{ijt} \in (L_{ijt}^c, U_{ijt}^c)) \quad \forall i,t,j : B(j) = 1\end{aligned} \tag{2}$$

The truncation regions are dependent on the current values of $\mathbf{z}$ or $\boldsymbol{\tau}$.

$$l_j^c = \max(-\infty, \max_{i,t}\{z_{ijt}^c : y_{ijt} = 0\})$$

$$u_j^c = \min(\infty, \min_{i,t}\{z_{ijt}^c : y_{ijt} = 1\})$$

$$L_{ijt}^c = \begin{cases} -\infty & \text{if } y_{ijt} = 0 \text{ or is missing} \\ \tau_j^c & \text{if } y_{ijt} = 1 \end{cases}$$

$$U_{ijt}^c = \begin{cases} \tau_j^c & \text{if } y_{ijt} = 0 \\ \infty & \text{if } y_{ijt} = 1 \text{ or is missing} \end{cases}$$

It is conceptionally easy to update these cutpoint parameters and latent variables. However, the approach of updating $\boldsymbol{\tau}$ and $\mathbf{z}$ with Gibbs sampling in separate blocks is problematic because it tends to mix poorly (see Cowles 1996, p. 104).

Such recognized inefficiency led Cowles (1996) to explore an alternative strategy for MCMC sampling of ordinal probit models, which is equally applicable to the special case of binomial data when the cutpoint is not fixed. Cowles proposed that instead of updating the cutpoint parameters and the latent variables in separate blocks, they be updated in one block. However, instead of simultaneously sampling both $\mathbf{z}$ and $\boldsymbol{\tau}$ from $\pi(\mathbf{z}, \boldsymbol{\tau}|$ all else), this step was split into sampling from $\pi(\boldsymbol{\tau}|$all else except $\mathbf{z}$) and then from $\pi(\mathbf{z}|$ all else). The implication is that by marginalizing over $\mathbf{z}$ in the full conditional of $\boldsymbol{\tau}$, exact sampling is no longer practical. Cowles used a Metropolis-Hastings step to simultaneously update the multiple cutpoints in her ordinal probit setting, and then updated the latent variables only if the proposed cutpoint draws were accepted.

Unlike the scheme proposed by Cowles (1996), the following sampling scheme updates the latent variables regardless of whether or not the proposed cutpoint draws were accepted and thus matches the implementation of Cowles' algorithm explained by (Johnson and Albert, 1999, pp. 135–136).

1. Update $\boldsymbol{\tau}$ by individually updating each τ_j with a Metropolis step using a normal distribution as the proposal distribution.

2. Regardless of whether $\boldsymbol{\tau_j}$ was accepted in the previous step, update each $z_{ijt} : B(j) = 1$ using its complete conditional.

A potential advantage of updating z_{ijt} in every instance is better chain mixing. Although this mandatory update is not required for the algorithm to be valid, the consequences of less frequent updating of the latent variables might involve poorer mixing of other model unknowns, such as the random effects.

The details for updating each τ_j with a Metropolis step are adapted from Cowles' approach. Let $u_{ijt} \equiv \sum_p u_{p,l(p,i,j)}$ be the sum of the random effects, and let $f(\tau_j)$ be the unnormalized full conditional of τ_j after integrating over the latent variables $\mathbf{z}$.

$$\begin{aligned} f(\tau_j) &= (1 + (\tau_j - m_j)^2 / s_j^2)^{-1} \\ &\quad \times \prod_{i=1}^{I} \prod_{t=1}^{T_j} \left[\left(\Phi \left((\tau_j - u_{ijt}) / \sigma_\epsilon \right) \right)^{1 - y_{ijt}} \left(1 - \Phi \left((\tau_j - u_{ijt}) / \sigma_\epsilon \right) \right)^{y_{ijt}} \right]^{w_{ijt}} \end{aligned} \tag{3}$$

As usual, $\Phi(\cdot)$ represents the cdf of the standard normal. To set $\tau_j^{(m)}$, the proposed value τ_j^* is sampled from the $N(\tau_j^c, v_j^2)$ distribution, where v_j^2 is a tuning parameter. Note that this proposal distribution is symmetric; the transition kernel q has the property that $q(\tau_j^*|\tau_j^c) = q(\tau_j^c|\tau_j^*)$. Thus, the Metropolis ratio implies that the acceptance probability is $\min(1, f(\tau_j^*)/f(\tau_j^c))$. With this probability, set $\tau_j^{(m)} = \tau_j^*$, and otherwise set $\tau_j^{(m)} = \tau_j^c$.

After updating $\boldsymbol{\tau}$, each z_{ijt} for binomial data is updated using its full conditional (Equation 2). This is conceptionally straightforward because the full conditional is in each case a truncated normal distribution. Actual implementation might use rejection sampling or evaluation of the quantile function at a randomly chosen point.

An updating procedure for $\boldsymbol{\tau}$ and the latent variables for binomial data has been detailed; we now demonstrate how the latent variables for rank data may be updated. The first part of the procedure is given by Johnson et al. (2002). A Metropolis-Hastings update is used, with each $z_{ijt} : B(j) = 0$ being individually updated. Recall that the t is optional for rank data because then $t \equiv 1$. The proposal distribution for z_{ij} is the $N(u_{1,l(1,i,j)} + \cdots + u_{P,l(P,i,j)}, \sigma_\epsilon^2)$ distribution, truncated to the region

$$\left(\max(-\infty, \max_{i' : y_{i'j} < y_{ij}} z_{i'j}), \min(\infty, \min_{i' : y_{i'j} > y_{ij}} z_{i'j}) \right). \tag{4}$$

If y_{ij} is missing, z_{ij} does not affect the proposal distribution's truncation region for any of the latent variables, and also the truncation region for updating z_{ij} is defined as $(-\infty, \infty)$. To calculate the acceptance probability, let $\mathbf{z}^c$ be the collection of current values of z_{ijt}, and let $\mathbf{z}^*$ be the collection of candidate (proposed) values. Note that because the z_{ijt} are individually updated, $\mathbf{z}^c$ and $\mathbf{z}^*$ will differ by at most a single element. Furthermore, upon updating each z_{ij}, both $\mathbf{z}^c$ and $\mathbf{z}^*$ are also updated. Let $p_{(ij)}(\kappa_j)^c$ be $p_{(ij)}(\kappa_j)$ when using the values of κ_j^c and $\mathbf{z}^c$, and let $p_{(ij)}(\kappa_j)^*$ be $p_{(ij)}(\kappa_j)$ when using the values of κ_j^c and $\mathbf{z}^*$. The acceptance probability for $z_{ij}^* : B(j) = 0$ is

$$\min\left(1, \left[\prod_{i=1}^{C(j)} p_{(ij)}(\kappa_j)^*\right]\left[\prod_{i=1}^{C(j)} p_{(ij)}(\kappa_j)^c\right]^{-1}\right). \tag{5}$$

If y_{ij} is missing then the acceptance probability for z_{ij} is always one.

Because latent variables are individually updated, it might be difficult for a group of latent variables with the same observed response to effectively traverse the support. For example, consider the possibility that two observations, say y_{1j} and y_{2j}, are tied for being the worst in assessment j, but the proportion of ties is low because of a very small value of κ_j. In an overdispersed initialization, it is possible that $z_{(1)j}^{(0)}$ and $z_{(2)j}^{(0)}$ are both much lower than $z_{(3)j}^{(0)}$. Suppose they are -5.1, -5.09, and -1.8. If the proposal distribution for $z_{1j}^{(1)}$ is very concentrated around, say, -2.0, the proposed value might be unlikely to be accepted because a value near -2.0 would make $p_{(1j)}(\kappa_j^{(0)})$ very small. But likewise, if the proposal distribution for $z_{2j}^{(1)}$ is concentrated around -2.0, a proposed value near -2.0 would again cause $p_{(1j)}(\kappa_j^{(0)})$ to be very small. The conundrum, then, is that the proposal distributions might favor values of z_{1j} that are far from z_{2j} (and vice versa) but the likelihood might be dramatically smaller if z_{1j} and z_{2j} are not close. This would make it difficult for either z_{1j} or z_{2j} to move. This problem can persist throughout any finite run of the MCMC algorithm.

To circumvent this difficulty, we add an extra step to the procedure given by Johnson et al. (2002) for updating the z_{ij}'s associated with rank data. This step allows shifts in z_{ij} values from assessment j that have the same y_{ij} values. The extra step is not essential for the algorithm to be valid, but it is recommended to help with chain mixing. Let y_j be a unique observed value of the assessment j responses. After updating each individual z_{ij} for which $y_{ij} = y_j$ (and thus each such z^c_{ij}), an additive shift of δ is proposed for the collection of such z_{ij}'s.

$$\forall i, \quad z^*_{ij} = \begin{cases} z^c_{ij} + \delta, \text{ if } y_{ij} = y_j \\ z^c_{ij}, \text{ otherwise} \end{cases}$$

Care is taken in choosing δ's proposal distribution to prevent a proposed shift that is inconsistent with the observed rankings. The proposal distribution is a normal distribution with mean 0 and variance given by tuning parameter v^2, truncated to the region that ensures appropriate ordering on the latent variables. The lower and upper limits, given by Equation 6 and Equation 7, are denoted by LL^c_{yj} and UL^c_{yj} to emphasize that they are dependent on the current values of all z_{ij}'s with observed rankings below or above the unique

y_j value being considered.

$$LL_{yj}^c = \left(\max_{i' : y_{i'j} < y_j} z_{i'j}^c - \min_{i' : y_{i'j} = y_j} z_{i'j}^c \right) \tag{6}$$

$$UL_{yj}^c = \left(\min_{i' : y_{i'j} > y_j} z_{i'j}^c - \max_{i' : y_{i'j} = y_j} z_{i'j}^c \right) \tag{7}$$

The Metropolis-Hastings ratio used in the acceptance probability also depends on LL_{yj}^* and UL_{yj}^*, which are analagously defined. The acceptance probability for the collection $\{z_{ij}^* : y_{ij} = y_j\}$ is the minimum of one and the result of Equation 8.

$$\frac{\left(\prod_{i=1}^{C(j)-1} p_{(ij)}(\kappa_j)^*\right) \left(\prod_{i=1}^{I} \exp(-(z_{ij}^* - u_{1,l(1,i,j)} - \cdots - u_{P,l(P,i,j)})^2 / 2\sigma_\epsilon^2)\right)}{\left(\prod_{i=1}^{C(j)-1} p_{(ij)}(\kappa_j)^c\right) \left(\prod_{i=1}^{I} \exp(-(z_{ij}^c - u_{1,l(1,i,j)} - \cdots - u_{P,l(P,i,j)})^2 / 2\sigma_\epsilon^2)\right)} \times \frac{\left[\Phi(UL_{yj}^c / v_j) - \Phi(LL_{yj}^c / v_j)\right]}{\left[\Phi(UL_{yj}^* / v_j) - \Phi(LL_{yj}^* / v_j)\right]} \tag{8}$$

A new value of δ is proposed for each unique y_j of each rank assessment j.

Each κ_j can be updated using a Metropolis-Hastings step because it is not convenient to sample directly from its full conditional distribution. As recommended by Johnson et al. (2002), a lognormal distribution depending on a tuning parameter c_j^2 is used for the proposal distribution of κ_j^* given the current state κ_j^c.

$$q(\kappa_j^*|\kappa_j^c) = \frac{\exp(-(\log(\kappa_j^*) - \log(\kappa_j^c))^2/(2c_j^2))}{\kappa_j^*\sqrt{2\pi c_j^2}} 1(\kappa_j^* > 0)$$

The lognormal distribution is appealing as a proposal distribution because it is easy to sample from and obeys the restriction that κ_j be positive. The acceptance probability for setting $\kappa_j^{(m)} = \kappa_j^*$ is

$$\min\left(1, \quad \left[\prod_{i=1}^{C(j)-1} p_{(ij)}(\kappa_j^*)/p_{(ij)}(\kappa_j^c)\right] (\kappa_j^*/\kappa_j^{(m-1)})^{a_j} \exp(-b_j(\kappa_j^* - \kappa_j^c))\right). \tag{9}$$

Recall that $\boldsymbol{\sigma} \equiv (\sigma_\epsilon^2, \sigma_{u,1}^2, \ldots, \sigma_{u,P}^2)$. The prior for $\boldsymbol{\sigma}$ is a Dirichlet density, but the full conditional is much more complicated. A Metropolis-Hastings step is used to update these variance parameters. The Dirichlet family of distributions can be used for the proposal distribution. One advantage is that each proposal will satisfy the modeling constraint that the sum of these variance parameters must equal 1. The transition kernel depends on two tuning parameters, $a_{MH} > 0$ and $b_{MH} \geq 0$.

$$q(\boldsymbol{\sigma}^*|\boldsymbol{\sigma}^c) = Dirichlet[\boldsymbol{\sigma}^*; a_{MH}\boldsymbol{\sigma}^c + b_{MH}\mathbf{1}]$$

The shorthand notation $Dirichlet[\mathbf{x}; \boldsymbol{\alpha}]$ is used to represent the pdf of the Dirichlet distribution as a function of $\mathbf{x}$ and having parameter vector $\boldsymbol{\alpha}$. The general idea is for the proposal distribution to have a mean that is close to the current value. The larger the value of a_{MH} is, the tighter the proposal distribution is. Positive values of b_{MH} essentially shrink each proposed value towards $1/(P+1)$, with the shrinkage more pronounced as b_{MH} increases.

The acceptance probability for $\boldsymbol{\sigma}^*$ is quite involved, so the convention previously undertaken of explicitly stating the acceptance probability in a specific form is now interrupted. The acceptance probability is the minimum of one and the quantity given by Equation 10.

$$\frac{q(\boldsymbol{\sigma}^c|\boldsymbol{\sigma}^*)\pi(\boldsymbol{\sigma}^*)\pi(\mathbf{z}|\mathbf{u}_1,\ldots,\mathbf{u}_p,\boldsymbol{\sigma}^*)\prod_{p=1}^{P}\pi(\mathbf{u}_p|\boldsymbol{\sigma}^*)}{q(\boldsymbol{\sigma}^*|\boldsymbol{\sigma}^c)\pi(\boldsymbol{\sigma}^c)\pi(\mathbf{z}|\mathbf{u}_1,\ldots,\mathbf{u}_p,\boldsymbol{\sigma}^c)\prod_{p=1}^{P}\pi(\mathbf{u}_p|\boldsymbol{\sigma}^c)}. \tag{10}$$

The random effects can be individually updated using their complete conditional distributions. For each level of each random effect, $u_{p,l(p,i,j)}^{(m)}$ is sampled from the normal distribution with mean μ and variance σ^2, where

$$\sigma^2 = \left[1/\sigma_{u,p}^2 + \sum_{i',j',t':l(p,i',j')=l(p,i,j)} 1/\sigma_\epsilon^2\right]^{-1} \tag{11}$$

and

$$\mu = \sum_{i',j',t':l(p,i',j')=l(p,i,j)} (\sigma^2/\sigma_\epsilon^2)\left(z_{i'j't'} - \sum_{p'\neq p} u_{p',l(p',i',j')}\right). \tag{12}$$

An alternative is to update all random effects in a block using the complete conditional distribution, a multivariate normal. This approach might be preferrable if the complete conditional of the random effects block indicates substantial correlation among the random effects. On the other hand, a block update might not be advisable if the dimensionality of the random effects block is fairly large because of computational difficulties with using large covariance matrices.

Most of this model's Metropolis or Metropolis-Hastings update steps involve one or more tuning parameters, with $\mathbf{z}$ being the lone exception. Because it is difficult to specify at the outset which tuning parameter values will give the desired acceptance rates, we suggest that the tuning parameters be periodically updated through some part of the burn-in period and then left constant (see, e.g., Gelman et al., 2002, p. 307).

References

Cowles, M. K. (1996), "Accelerating Monte Carlo Markov Chain Convergence for Cumulative-link Generalized Linear Models," *Statistics and Computing*, 6, 101–111.

Johnson, V. E. and Albert, J. H. (1999), *Ordinal Data Modeling*, New York: Springer-Verlag.

Johnson, V. E., Deaner, R. O., and van Schaik, C. P. (2002), "Bayesian Analysis of Rank Data With Application to Primate Intelligence Experiments," *Journal of the American Statistical Association*, 97, 8–17.