Multi-parameter MCMC notes by Mark Holder

Review

In the last lecture we justified the Metropolis-Hastings algorithm as a means of constructing a Markov chain with a stationary distribution that is identical to the posterior probability distribution. We found that if you propose a new state from a proposal distribution with probability of proposal denote $q(j,k)$ then you could use the following rule to calculate an acceptance probability:

$$\alpha(j,k) = \min\left[1, \left(\frac{\mathbb{P}(D|\theta=k)}{\mathbb{P}(D|\theta=j)}\right)\left(\frac{\mathbb{P}(\theta=k)}{\mathbb{P}(\theta=j)}\right)\left(\frac{q(k,j)}{q(j,k)}\right)\right]$$

To get the probability of moving, we have to multiple the proposal probability by the acceptance probability:

$$\begin{aligned} q(j,k) &= \mathbb{P}(x^*_{i+1}=k|x_i=j) \\ \alpha(j,k) &= \mathbb{P}(x_{i+1}=k|x_i=j,x^*_{i+1}) \\ m_{j,k} &= \mathbb{P}(x_{i+1}=k|x_i=j) \\ &= q(j,k)\alpha(j,k) \end{aligned}$$

If $\alpha(j,k) < 1$ then $\alpha(k,j) = 1$. In this case:

$$\begin{aligned} \frac{\alpha(j,k)}{\alpha(k,j)} &= \left[\left(\frac{\mathbb{P}(D|\theta=k)}{\mathbb{P}(D|\theta=j)}\right)\left(\frac{\mathbb{P}(\theta=k)}{\mathbb{P}(\theta=j)}\right)\left(\frac{q(k,j)}{q(j,k)}\right)\right] \Big/ 1 \\ &= \left(\frac{\mathbb{P}(D|\theta=k)}{\mathbb{P}(D|\theta=j)}\right)\left(\frac{\mathbb{P}(\theta=k)}{\mathbb{P}(\theta=j)}\right)\left(\frac{q(k,j)}{q(j,k)}\right) \end{aligned}$$

Thus, the ratio of these two transition probabilities for the Markov chain are:

$$\begin{aligned} \frac{m_{j,k}}{m_{k,j}} &= \frac{q(j,k)\alpha(j,k)}{q(k,j)\alpha(k,j)} \\ &= \left(\frac{q(j,k)}{q(k,j)}\right)\left(\frac{\mathbb{P}(D|\theta=k)}{\mathbb{P}(D|\theta=j)}\right)\left(\frac{\mathbb{P}(\theta=k)}{\mathbb{P}(\theta=j)}\right)\left(\frac{q(k,j)}{q(j,k)}\right) \\ &= \left(\frac{\mathbb{P}(D|\theta=k)}{\mathbb{P}(D|\theta=j)}\right)\left(\frac{\mathbb{P}(\theta=k)}{\mathbb{P}(\theta=j)}\right) \end{aligned}$$

If we recall that, under detailed balance, we have:

$$\frac{\boldsymbol{\pi}_k}{\boldsymbol{\pi}_j} = \frac{m_{j,k}}{m_{k,j}}$$

we see that we have constructed a chain in which the stationary distribution is proportional to the posterior probability distribution.

Convergence

We can (sometimes) diagnose failure-to-converge by comparing the results of separate MCMC simulations.

If all seems to be working, then we would like to treat our sampled points from the MCMC simulation as if they were draws from the posterior probability distribution over parameters. Unfortunately, our samples from our MCMC approximation to the posterior will display autocorrelation. We can calculate an effective sample size by diving the number of sampled points by the *autocorrelation time*. The `CODA` package in R provides several useful tools for diagnosing problems with MCMC convergence.

Multi-parameter inference

In the simple example discussed in the last lecture, θ, could only take one of 5 values. In general, our models have multiple, continuous parameters.

We can adopt the acceptance rules to continuous parameters by using a Hastings ratio which is the ratio of proposal densities (and we'd also have a ratio of prior probability densities).

Adapting MCMC to multi-parameter problems is also simple. Because of co-linearity between parameters, it may be most effective to design proposals that change multiple parameters in one step. But this is not necessary. If we update each parameter, we can still construct a valid chain. In doing this, we are effectively sampling the m-th parameter from $\mathbb{P}(\theta_m|\text{Data}, \theta_{-m})$ where θ_{-m} denote the vector of parameters without parameter m included.

Having a marginal distribution is not enough to reconstruct a joint distribution. But we have the distribution on θ_m for every possible value of the other parameters. So we are effectively sampling from the joint distribution, we are just updating one parameter at a time.

GLM in Bayesian world

Consider a data set of different diets many full sibs reared under two diets (normal=0, and unlimited=1). We measure snout-vent length for a bunch of gekkos. Our model is:

- There is an unknown mean SVL under the normal diet, α_0.
- α_1 is the mean SVL of an infinitely-large independent sample under the unlimited diet.
- Each family, j, will have a mean effect, B_j. This effect gets added to the mean based on the diet, regardless of diet.
- Each family, j, will have a mean response to unlimited diet, C_{1j}. This effect is only added to individuals on the unlimited diet. For notational convenience, we can simply define $C_{0j} = 0$ for all families.

- The SVL for each individual is expected to normally-distributed around the expected value; the difference between a response and the expected value is ϵ_{ijk}.

To complete the likelihood model, we have to say something about the probability distributions that govern the random effects:

- $B_j \sim \mathcal{N}(0, \sigma_B)$
- $C_{1j} \sim \mathcal{N}(0, \sigma_C)$
- $\epsilon_{ijk} \sim \mathcal{N}(0, \sigma_e)$

In our previous approach, we could do a hypothesis test such as $H_0 : \alpha_0 = \alpha_1$, or we could generate point estimates. That is OK, but what if we want to answer questions such as "What is the probability that $\alpha_1 - \alpha_0 > 0.5$mm?"

Could we:

- reparameterize to $\delta_1 = \alpha_1 - \alpha_0$,
- construct a $x\%$ confidence interval,
- search for the largest value of $x^\dagger$ such that 0 is not included in the confidence interval?
- do something like $\mathbb{P}(\alpha_1 - \alpha_0 > 0.5) = (1 - x^\dagger)/2$

or something like that? **No!** That is not a correct interpretation of a P-value, or confidence interval!

If we conduct Bayesian inference on this model, we can estimate the joint posterior probability distribution over all parameters. From this distribution we can calculate the $\mathbb{P}(\alpha_1 - \alpha_0 > 0.5|X)$ by integrating

$$\mathbb{P}(\alpha_1 - \alpha_0 > 0.5|X) = \int_{-\infty}^{\infty} \left(\int_{\alpha_0 + 0.5}^{\infty} p(\alpha_0, \alpha_1|X) d\alpha_1 \right) d\alpha_0$$

From MCMC output we can count the fraction of the sampled points for which $\alpha_1 - \alpha_0 > 0.5$, and use this as an estimate of the probability. Note that it is hard to estimate very small probabilities accurately using a simulation-based approach. But you can often get a very reasonable estimate of the probability of some complicated, joint statement about the parameters by simply counting the fraction of MCMC samples that satisfy the statement.

Definitions of probability

In the *frequency* or *frequentist* definition, the probability of an event is the fraction of times the event would occur if you were able to repeat the trial an infinitely large number of times. The probability is defined in terms of long-run behavior and "repeated experiments."

Bayesians accept that this is correct (if we say that the probability of heads is 0.6 for a coin, then a Bayesian would expect the fraction heads in a very large experiment to approach 60%), but Bayesians also use probability to quantify uncertainty – even in circumstances in which it is does

not seem reasonable to think of repeated trials. The classic example is the prior probability assigned to a parameter value. A parameter in a model can be a fixed, but unknown constant in nature. But even if there can only be one true value, Bayesians will use probability statements to represent the *degree of belief*. Low probabilities are assigned to values that seem unlikely. Probability statements can be made based on vague beliefs (as is often the case with priors), or they can be informed by previous data.

Fixed vs random effects in the GLM model

In order to perform Bayesian inference on the GLM model sketched out above, we would need to specify a prior distribution on α_0 and α_1. If we new that the gekkos tended to be around 5cm long (SVL), then we might use priors like this:

$$\alpha_0 \sim \text{Gamma}(\alpha = 10, \beta = 2)$$

$$\delta_1 \sim \text{Normal}(\mu = 1, \sigma = 10)$$

Where $\text{Gamma}(\alpha, \beta)$ is a Gamma-distribution with mean α/β and variance α/β^2.

Now that we have a prior for all of the parameters, we may notice that the distinction between random effects and a vector of parameters becomes blurry. Often we can implement a sampler more easily in a Bayesian context if we simply model the outcomes of random processes that we don't observe. A variable that is the unobserved result of the "action" of our model are referred to as "latent variables." When we conduct inference without imputing values for the latent variables, we are effectively integrating them out of the calculations. If we choose, we may prefer to use MCMC to integrate over some of the latent variables. When we do this, the latent variables "look" like parameters in the model, we calculate a likelihood ratio for them and a prior ratio for them whenever they need to be updated[1].

In the ML approach to the GLM, we did not try to estimate the random effects. We simply recognized that the presence of a "family" effect, for example, would cause a non-zero covariance. In essence we were integrating over possible values of each families effect, and just picking up on the degree to which within-family comparison were non-independent because they shared some unknown parameters.

In MCMC, it would also be easy to simply treat possible values for each B_j and C_{1j} element as a latent variable. The priors would be the Normal distributions that we outlined above, and

$$y_{ijk} = \alpha_0 + i\delta_1 + B_j + C_{ij} + \epsilon_{ijk}$$

or

$$y_{ijk} \sim \mathcal{N}(\alpha_0 + i\delta_1 + B_j + C_{ij}, \sigma_e)$$

[1]The distinction between a parameter and a latent variable can be subtle: when you reduce the model to the minimal statement about unknown quantities, you are dealing with the parameters and their priors. You can add latent variables to a model specification, but when you do this the prior for the latent variables comes from the parameters, existing priors, and other latent variables. So you don't have to specify a new prior for a latent variable - it falls out of the model.

So

$$f(y_{ijk}|\alpha_0, \delta_1, B_j, C_{ij}) = f(\epsilon_{ijk})$$

and we can calculate this density by:

$$f(y_{ijk}|\alpha_0, \delta_1, B_j, C_{ij}) = \frac{1}{\sqrt{2\pi\sigma_e^2}} \exp\left[\frac{-\left(y_{ijk} - \alpha_0 - i\delta_1 - B_j - C_{ij}\right)^2}{2\sigma_e^2}\right]$$

Note that δ_1 only appears in the likelihood for gekkos on diet 1, thus updating δ_1 will not change many of the terms in the log-likelihood. We only need to calculate the log of the likelihood ratio, so we can just ignore terms in which δ_1 does not occur when we want to update this parameter. This type of optimization can speed-up the likelihood calculations dramatically. Specifically, we have terms like:

$$\begin{aligned}
\ln f(y_{0jk}|\alpha_0, B_j) &= -\frac{1}{2}\ln\left[2\pi\sigma_e^2\right] - \left[\frac{\left(y_{0jk} - \alpha_0 - B_j\right)^2}{2\sigma_e^2}\right] \\
\ln f(y_{1jk}|\alpha_0, B_j, C_{1j}) &= -\frac{1}{2}\ln\left[2\pi\sigma_e^2\right] - \left[\frac{\left(y_{1jk} - \alpha_0 - \delta_1 - B_j - C_{1j}\right)^2}{2\sigma_e^2}\right]
\end{aligned}$$

in the log-likelihoods.

If we are updating α_0:

$$\begin{aligned}
\ln LR(y_{0jk}) &= \ln f(y_{0jk}|\alpha_0^*, \delta_1, B_j, C_{1j}, \sigma_e) - \ln f(y_{0jk}|\alpha_0, B_j, \delta_1, B_j, C_{1j}, \sigma_e) \\
&= \frac{1}{2\sigma_e^2}\left[\left(y_{0jk} - \alpha_0 - B_j\right)^2 - \left(y_{0jk} - \alpha_0^* - B_j\right)^2\right] \\
&= \frac{1}{2\sigma_e^2}\left[\left(y_{0jk} - B_j\right)^2 - 2\left(y_{0jk} - B_j\right)\alpha_0 + \alpha_0^2 - \left(y_{0jk} - B_j\right)^2 + 2\left(y_{0jk} - B_j\right)\alpha_0^* - \left(\alpha_0^*\right)^2\right] \\
&= \frac{1}{2\sigma_e^2}\left[2\left(y_{0jk} - B_j\right)\left(\alpha_0^* - \alpha_0\right) + \alpha_0^2 - \left(\alpha_0^*\right)^2\right]
\end{aligned}$$

Similar algebra leads to:

$$\begin{aligned}
\ln LR(y_{1jk}) &= \ln f(y_{1jk}|\alpha_0^*, \delta_1, B_j, C_{1j}, \sigma_e) - \ln f(y_{1jk}|\alpha_0, B_j, \delta_1, B_j, C_{1j}, \sigma_e) \\
&= \frac{1}{2\sigma_e^2}\left[2\left(y_{1jk} - \delta_1 - B_j - C_{1j}\right)\left(\alpha_0^* - \alpha_0\right) + \alpha_0^2 - \left(\alpha_0^*\right)^2\right]
\end{aligned}$$

Each of the data points is independent, conditional on all of the latent variables, so:

$$\ln LR = \left(\sum_j \sum_k \ln LR(y_{0jk})\right) + \left(\sum_j \sum_k \ln LR(y_{1jk})\right)$$

It is helpful to introduce some named variables that are simply convenient functions of the data or parameters:

$$n = \text{total \# individuals}$$

$$
\begin{aligned}
n_0 &= \#\text{ individuals in treatment 0} \\
n_1 &= \#\text{ individuals in treatment 1} \\
f &= \#\text{ families} \\
B_\star &= \sum_j \sum_k B_j \\
D_\star &= \sum_j B_j^2 \\
B_{1\star} &= \sum_j \sum_k B_j \text{ (summing only over treatment 1)} \\
C_{1\star} &= \sum_j \sum_k C_{1j} \\
y_{0\star\star} &= \sum_j \sum_k y_{0jk} \\
y_{1\star\star} &= \sum_j \sum_k y_{1jk} \\
y_{\star\star\star} &= y_{0\star\star} + y_{1\star\star} \\
R_\star &= \left(\sum_j \sum_k^{n_{0j}} \left[y_{0jk} - \alpha_0 - B_j \right]^2 \right) + \left(\sum_j \sum_k^{n_{1j}} \left[y_{0jk} - \alpha_0 - \delta_1 - B_j - C_{1j} \right]^2 \right)
\end{aligned}
$$

Updating α_0

Thus, the log-likelihood ratio for $\alpha_0 \to \alpha_0^*$ is:

$$
\ln LR = \frac{n \left[\alpha_0^2 - (\alpha_0^*)^2 \right] + 2 \left[y_{\star\star\star} - n_1 \delta_1 - B_\star - C_{1\star} \right] (\alpha_0^* - \alpha_0)}{2\sigma_e^2}
$$

Note that calculating this log-likelihood is very fast. It is easy to keep the various starred-sums up to date when B_j and C_{1j} elements change. So evaluating the likelihood ratio for proposals to α_0 does not even involve iterating through every data point.

Updating δ_1

If we are just updating δ_1 then the log-likelihood ratio have fewer terms, because constants drop out in the subtraction:

$$
\begin{aligned}
\ln LR(y_{1jk}) &= \ln f(y_{1jk} | \alpha_0, \delta_1^*, B_j, C_{1j}, \sigma_e) - \ln f(y_{1jk} | \alpha_0, B_j, \delta_1, B_j, C_{1j}, \sigma_e) \\
&= \frac{1}{2\sigma_e^2} \left[2 \left(y_{1jk} - \alpha_0 - B_j - C_{1j} \right) (\delta_1^* - \delta_1) + \delta_1^2 - (\delta_1^*)^2 \right] \\
\ln LR(Y) &= \frac{n_1 \delta_1^2 - n_1 (\delta_1^*)^2 + 2 \left[y_{1\star\star} - n_1 \alpha_0 - B_{1\star} - C_{1\star} \right] (\delta_1^* - \delta_1)}{2\sigma_e^2}
\end{aligned}
$$

Updating σ_B or σ_C

If we want to update σ_B, and we have a set of latent variables then we only have to consider the portion of the likelihood that comes from the latent variables (we don't have to consider the data):

$$\begin{aligned}
\ln LR &= \sum_j \left(\frac{1}{2} \ln \left[2\pi(\sigma_B)^2 \right] + \left[\frac{B_j^2}{2(\sigma_B)^2} \right] - \frac{1}{2} \ln \left[2\pi(\sigma_B^*)^2 \right] - \left[\frac{B_j^2}{2(\sigma_B^*)^2} \right] \right) \\
&= f \ln \left[\frac{\sigma_B}{\sigma_B^*} \right] + \sum_j \left(\left[\frac{B_j^2}{2(\sigma_B)^2} \right] - \left[\frac{B_j^2}{2(\sigma_B^*)^2} \right] \right) \\
&= f \ln \left[\frac{\sigma_B}{\sigma_B^*} \right] + \frac{D_\star}{2} \left[\frac{1}{(\sigma_B)^2} - \frac{1}{(\sigma_B^*)^2} \right]
\end{aligned}$$

The corresponding formula for updating σ_C would use C_{1j} as the variates that depend on the variance parameter.

Updating σ_E

Updating σ_e would entail summing the effect of all of the residual, but this is the only parameter that would require iterating over all of the data points:

$$\ln LR = n \ln \left[\frac{\sigma_E}{\sigma_E^*} \right] + \frac{R_\star}{2} \left[\frac{1}{(\sigma_E)^2} - \frac{1}{(\sigma_E o^*)^2} \right]$$

Updating B_j

Updating a family effect is much like updating another mean effect, except (of course) it only affects the likelihood of one family (denoted family j) but effects both treatments:

$$\ln LR = \frac{(n_{0j} + n_{1j}) \left[B_j^2 - (B_j^*)^2 \right] + 2 \left[y_{\star j \star} - (n_{0j} + n_{1j})\alpha_0 - n_{1j}(\delta_1 + C_{1j}) \right] (B_j^* - B_j)}{2\sigma_e^2}$$

Updating C_{1j}

Updating a C_{1j} variable only affects the treatment-1 individuals in one family (denoted family j):

$$\ln LR = \frac{n_{1j} \left[C_{1j}^2 - (C_{1j}^*)^2 \right] + 2 \left[y_{1j\star} - n_{1j}(\alpha_0 + \delta_1 + B_j) \right] (C_{1j}^* - C_{1j})}{2\sigma_e^2}$$

Latent variable mixing

The downside of introducing latent variables, is that our chain would have to sample over them (update them). The equation for udpating B_j looks a lot like the update of δ_1, but we only have to perform the summation over family j.

In essence, introducing latent variables lets us do calculations in this form:

$$\mathbb{P}(\alpha_0, \delta_1, \boldsymbol{B}_j, \boldsymbol{C}_{1j}, \sigma_B, \sigma_C, \sigma_e | \boldsymbol{Y}) = \frac{\mathbb{P}(\boldsymbol{Y} | \alpha_0, \delta_1, \boldsymbol{B}_j, \boldsymbol{C}_{1j}, \sigma_e) \mathbb{P}(\boldsymbol{B}_j | \sigma_B) \mathbb{P}(\boldsymbol{C}_{1j} | \sigma_C) \mathbb{P}(\alpha_0, \delta_1, \sigma_B, \sigma_C, \sigma_e)}{\mathbb{P}(\boldsymbol{Y})}$$

and use MCMC to integrate over $\boldsymbol{B}_j$ and $\boldsymbol{C}_{1j}$ to obtain:

$$\mathbb{P}(\alpha_0, \delta_1, \sigma_B, \sigma_C, \sigma_e | \boldsymbol{Y}) = \int \int \mathbb{P}(\alpha_0, \delta_1, \boldsymbol{B}_j, \boldsymbol{C}_{1j}, \sigma_B, \sigma_C, \sigma_e | \boldsymbol{Y}) d\boldsymbol{B}_j d\boldsymbol{C}_{1j}$$

Marginalizing over a parameter is easy – you simply ignore it when summarizing the MCMC output.

References