MCMC using data augmentation – BayesCode

Nicolas Lartillot
nicolas.lartillot@univ-lyon1.fr

December 7, 2017

1 Introduction

The aim of this document is to describe the MCMC strategies that are used in the context of BayesCode. These MCMC methods heavily rely on data augmentation and sufficient statistics. Codon models will be used as a running example.

In its simplest form, a codon model depends on a 4×4 nucleotide substitution matrix Q and a single positive parameter $\omega = dN/dS$, which stands for the ratio of non-synonymous over synonymous substitution rates. Here, the nucleotide matrix itself is a GTR matrix, parameterized in terms of relative exchange rates $\rho_{n_1 n_2}$, such that $\rho_{n_1 n_2} = \rho_{n_2 n_1}$ and equilibrium frequencies π_n, for $n = A, C, G, T$. Thus, $Q = Q(\rho, \pi)$, where ρ and π can be taken as 6-dimensional and 4-dimensional vectors of positive real numbers, in both cases summing to 1.

The 61×61 codon matrix $R = R(Q, \omega)$, which defines the instant rate of point substitutions between pairs of codons, is then as follows: the rate of substitution between codons c_1 and c_2, differing at only one position, and with nucleotides n_1 and n_2 at this position, is given by:

$$R_{c_1 c_2} = Q_{n_1 n_2} \quad \texttt{if synonymous} \tag{1}$$

$$R_{c_1 c_2} = Q_{n_1 n_2}\,\omega \quad \texttt{if non synonymous} \tag{2}$$

and, finally, the rate of substitution between two codons differing at more than one position is 0.

The codon process defined by rate matrix R has equilibrium frequencies Π_c over the 61 codons, which are proportional to the product of the equilibrium frequencies of the underlying three nucleotides. That is, if codon $c = (n_1, n_2, n_3)$, then $\Pi_c = \frac{1}{Z}\pi_{n_1}\pi_{n_2}\pi_{n_3}$ (the proportionality constant Z is not 1, since the stop codons are not allowed). Of note, the equilbrium frequencies of the codon process do not depend on ω.

Pruning- versus mapping-based likelihood computation

A sequence of length N evolves by point substitutions (according to the codon model given above), over a phylogenetic tree T, with P tips (P taxa). The tree topology will be considered as known and fixed. The branch lengths will be noted $l = (l_j)_{j=1..2P-3}$. Sites are independent, although in some cases, different sites might evolve according to different substitution processes (different values of ω). In the models considered below, the process is assumed homogeneous through time (over the tree), although this assumption could be relaxed in other models. The parameters of the process will be generically denoted by $\theta = (l, \rho, \pi, \omega..)$, with a detailed structure that will depend on the specific model. Dependence on the fixed tree topology is omitted.

A realization of this random process results in a detailed *substitution history* over the tree, which will be denoted by S. The substitution history at site i will be denoted by S_i. The probability of drawing a specific substitution history given the tree and the parameters of the model, is denoted by $p(S \mid \theta)$. Since the process is site-independent, this probability can be factorized as a product over sites:

$$p(S \mid \theta) \quad = \quad \prod_i p(S_i \mid \theta) \tag{3}$$

Sequences are observed at the tips of the phylogenetic tree, giving a data set $D = (D_i)_{1..N}$, of aligned sequences of length N, where D_i is the ith column of the alignment. For a given site i, one can sum the probability of all substitution histories that are compatible with the data at the tips:

$$p(D_i \mid \theta) \quad = \quad \int_{S_i \mid D_i} p(S_i \mid \theta) dS_i$$

using the so-called pruning algorithm (Felsenstein). The *pruning*-based likelihood is then a product over sites:

$$p(D \mid \theta) \quad = \quad \prod_i p(D_i \mid \theta) \tag{4}$$

This likelihood can then be combined with the prior over the model parameters, e.g:

$$p(\theta \mid D) \quad \propto \quad p(D \mid \theta)\, p(\theta). \tag{5}$$

Most phylogenetic MCMC samplers target the distribution over the model parameters given by equation 5 – which means that they have to repeatedly invoke the pruning algorithm to recalculate the pruning-based likelihood given by equation 4 – which is most often the limiting step of the MCMC.

An alternative, which is used here, is to do the MCMC conditionally on the detailed substitution history S, thus doing the MCMC over the augmented configuration (S, θ), under the target distribution obtained by combining the *mapping*-based likelihood given by equation 3 with the prior over model parameters:

$$p(\theta, S \mid D) \quad \propto \quad p(S \mid \theta)\, p(\theta) \tag{6}$$

The key idea that makes this strategy efficient is that the mapping-based likelihood depends on compact summary statistics of S (which in turn depend on the specific parameter component being resampled), leading to very fast evaluation of $p(S \mid \theta)$. On the other hand, this requires to implement more complex MCMC procedures, that have to alternate between

- sampling S conditionally on the data and the current parameter configuration: $S \mid \theta, D$
- re-sampling the parameters conditionally on S: $\theta \mid S$.

Sampling the substitution history conditionally on the current parameter configuration can be done in several ways (described in Nielsen, Lartillot, Rodrigue).

If this data-augmentation formalism is very efficient under a given tree topology, it raises the problem that the substitution history is highly dependent on the current tree topology, thus making topological updates difficult to implement. The compromise currently used in phylobayes is to switch between mapping-based sampling (for updating the continuous parameters of the model) and pruning-based sampling (for updating the tree topology). In BayesCode, where the topology is fixed, this problem does not exist: the entire MCMC is mapping-based. Finally, sufficient statistics can be used at other levels of the model (e.g. for summarizing the distribution of an array of iid gamma random variables, conditional on their shape and scale parameter, which will be considered below for branch lengths).

2 Model structure

In a first step, we consider a simple model assuming a global ω, for all sites and over all branches. This model will be used to illustrate how the mapping-based idea can be used to efficiently resample branch lengths, nucleotide rates ρ and π, and ω. The structure of the model is as follows:

- an unrooted phylogenetic tree topology T (fixed)
- a scale parameter λ for branch lengths, with an exponential prior of mean 10.
- a set of iid exponentially distributed branch lengths $l_j \mid \lambda$, $j = 1..2P-3$
- an ω parameter, with gamma prior (fixed shape parameter α and scale parameter β).
- a vector of nucleotide relative exchangeabilities ρ (dimension 6, uniform Dirichlet distribution)
- a vector of nucleotide equilibrium frequencies π (dimension 4, uniform Dirichlet distribution)
- a nucleotide GTR matrix $Q = Q(\rho, \pi)$
- an array of codon substitution processes (iid across sites) along the tree $S_{1:N} \mid Q, l, \omega$

3 MCMC sampling

To implement the mapping-based MCMC sampling strategy, we first sample the detailed substitution history S for all sites along the tree. Several methods exist for doing this (Nielsen, Lartillot and Rodrigue, refs).

Then, we write down the probability of S given the parameters, and finally, we collect all factors that depend on some parameter of interest and make some simplifications. This ultimately leads to relatively compact sufficient statistics. These suff stats depend on the specific parameter we want to consider: here, we have to consider two cases: branch lengths, on the one hand, and all parameters of the rate matrix R, on the other hand.

3.1 branch lengths l (PoissonSuffStat)

In the case of branch lengths, sufficient statistics take a very simple form:

$$p(S \mid l) \quad \propto \quad \prod_j l_j^{u_j} e^{-b_j l_j}$$

where u_j is the total number of substitutions over branch j (summed over all sites), and b_j is the mean rate away from current codon state (averaged over the entire substitution history). Thus, formally, the probability of the substitution mapping can be summarized by saying that the total number of substitutions along a given branch over all sites, u_j, is Poisson distributed, of mean $b_j l_j$.

In turn, this Poisson likelihood is conjugate with the gamma prior on l (in fact, exponential prior, which is gamma with shape parameter equal to 1), leading to analytical relation for resampling branch lengths. This can be seen as follows: the posterior is proportional to the product of prior and likelihood, which gives

$$\begin{aligned} p(l \mid S, \lambda) &\propto p(l \mid \lambda) p(S \mid l) \\ &\propto \prod_j e^{-\lambda l_j} \prod_j l_j^{u_j} e^{-b_j l_j} \\ &\propto \prod_j l_j^{u_j} e^{-(b_j + \lambda) l_j} \end{aligned}$$

where all factors not depending on the l_j's have been dropped. This shows that, conditional on the suff stats and on the prior hyperparameters, the l_j's are gamma distributed (of shape parameter $u_j + 1$ and scale parameter $b_j + \lambda$ for branch j) and thus can be directly resampled by Gibbs.

3.2 General sufficient statistics for a rate matrix (PathSuffStat)

If we express the probability of the substitution mapping as a function of the codon substitution process R, we get the following simpified expression:

$$p(S \mid l, R) \propto \prod_k \Pi_k^{n_k} \prod_{kl} R_{kl}^{m_{kl}} \prod_k e^{-|R_{kk}| a_k}. \qquad (7)$$

where we define the sufficient statistics:

- m_{kl}: the total number of substitutions from codon k to codon l
- n_k: the total number of sites starting with codon k at the root of the tree
- a_k: the total waiting time in codon k

Once these suff stats have been computed, the parameters of the rate matrix R (here, ρ, π and ω) can be resampled conditional on S, using equation 7 each time the likelihood needs to be recomputed. This leads to a first relatively fast MCMC strategy.

In fact, it turns out that this equation can be further simplified, depending on whether we want to resample ω or the nucleotide rate parameters ρ and π. We consider the two cases in turn.

3.3 $\omega = dN/dS$ (OmegaSuffStat)

Developing the codon matrix $R = R(Q, \omega)$ according to the equations given in the introduction and gathering all terms depending on ω in equation 7 leads to an expression of the probability of the mapping conditional on ω which also takes a Poisson-like functional form:

$$p(S \mid \omega) \propto \omega^v e^{-c\omega}$$

where v is the total number of non-synonymous substitutions over the whole tree (and across all sites) and c is the mean non-synonymous rate away from the current codon state (averaged over the whole tree and across all sites), multipled by the total tree length – in other words, c is the mean expected number of non-synonymous substitutions over the tree and for the complete alignment, under the current parameter values.

Since ω has a Gamma prior, the conjugate relation mentioned above in the case of branch lengths is also valid here:

- the prior over ω is Gamma of shape α and scale β;
- the probability of the mapping is formally equivalent to a Poisson variable v of mean $c\omega$
- therefore, the posterior over ω is Gamma, of shape $\alpha + v$ and scale $\beta + c$.

3.4 Nucleotide rates ρ and π (NucPathSuffStat)

Developing the codon matrix $R = R(Q, \omega)$ according to the equations given in the introduction and gathering all terms depending on the underlying nucleotide matrix Q in equation 7 leads to an expression of the probability of the mapping conditional on Q which can be simplified as follows:

$$p(S \mid Q) \quad \propto \quad \prod_i \left(\frac{\pi_i}{Z(Q)} \right)^{w_i} \prod_{ij} Q_{ij}^{x_{ij}} \prod_{ij} e^{-d_{ij} Q_{ij}} \qquad (8)$$

Here, $Z = Z(Q)$ is the normalization factor of the stationary probabilities of the codons (which depends only on Q, see introduction), w_i is the number of occurrences of nucleotide i at the root, x_{ij} is the number of codon substitutions implying a nucleotide substitution from i to j, and d_{ij} is some average nucleotide substitution rate from nucleotide i to nucleotide j, with a complicated expression.

When we want to resample ρ and π, we can therefore first compute these sufficient statistics, and then we can perform MH moves on ρ and π, each time recomputing Q and then relying on equation 8 to recalculate the probability before and after the move.

3.5 Branch length hyperparameters: λ

If we want to MH resample the λ parameter conditional on branch lengths l, the probability of branch lengths conditional on λ is:

$$\begin{aligned} p(l \mid \lambda) &= \prod_j p(l_j \mid \lambda) \\ &= \prod_j \lambda e^{-\lambda l_j}. \end{aligned}$$

Taking the log, this can be simplified as:

$$\begin{aligned} \ln p(l \mid \lambda) &= \sum_j \ln p(l_j \mid \lambda) \\ &= \sum_j \ln \lambda - \lambda l_j \\ &= (2P - 3)\lambda - \lambda L. \end{aligned}$$

where $L = \sum_j l_j$ is the total tree length. In the more general case where the branch lengths are gamma distributed, with arbitrary shape parameter μ:

$$\begin{aligned} \ln p(l \mid \lambda, \mu) &= \sum_j \ln p(l_j \mid \lambda, \mu) \\ &= \sum_j \ln \frac{\lambda^\mu}{\Gamma(\mu)} + (\mu - 1) l_j - \lambda l_j \\ &= (2P-3) \ln \frac{\lambda}{\Gamma(\mu)} + (\mu - 1) M - \lambda L. \end{aligned}$$

where $M = \sum_j \ln l_j$.

Thus, when we want to resample λ, we can first compute the following Gamma sufficient statistics: $(2P-3, L, M)$, and then perform MH resampling on λ, targeting a very compact posterior distribution on λ, given by the equation just above. This can lead to a substantial increase in computational efficiency for a large number of taxa. This idea works for any array of iid gamma random variables, and is formalized in BayesCode in the `GammaSuffStat` class.

Overall schedule

Altogether, the full sequence of moves under this model can be organized as follows:

- sample a substitution history $S \mid T, l, \rho, \pi, \omega$
- gather sufficient statistics for S, as a function of branch lengths (PoissonSuffStat)
- Gibbs resample branch lengths conditional on these suffstats and conditional on λ
- gather sufficient statistics for l, as a function of λ (GammaSuffStat)
- MH resample λ conditional on these suffstats
- gather general sufficient statistics for S, as a function of the rate matrix R (PathSuffStat)
- further simplify these PathSuffStat into a suff stat of S as a function of ω (OmegaSuffStat)
- Gibbs resample ω
- further simplify these PathSuffStat into a suff stat of S as a function of Q (NucPathSuffStat)
- MH resample ρ and π

The whole sequence of moves is conditional on S. It can be done multiple times, refreshing the substitution history once in a while.

This approach can easily be generalized to all sorts of models: for instance, with site-specific ω_i's, one would just gather the OmegaSuffStat separately for each site. If the ω_i's are iid Gamma, then they can be directly resampled by Gibbs. Otherwise, the conjugate relation is lost. However, even then, fast MH under the compact likelihood functions given above, as a function of the sufficient statistics, can still be used to make the MCMC more efficient.