Phylogenetic Inference using MrBayes v3.2

Overview

MrBayes is a software program for inferring phylogenetic parameters in a Bayesian statistical framework (Huelsenbeck and Ronquist, 2001; 2005; Ronquist et al., 2012). This program relies on the Markov chain Monte Carlo (MCMC) numerical method to approximate the joint posterior probability distribution of model parameters (Metropolis et al., 1953; Hastings, 1970). The current version, v3.2, allows you to perform several different types of analyses:

- Topological and branch-length inference of partitioned alignments comprising different data types under a range of site-evolution models:

  DNA: 4x4 nucleotide model, 16x16 doublet model, 61x61 codon model

  Restriction: 0/1 state model (used for any binary-type data including indel coding and 2-state morphological characters)

  Standard: 0–9 state model for standard, discrete morphological characters

  Protein: A–Y amino acid data under either fixed-rate or variable rate models

- Ancestral-state estimation
- Divergence-time estimation under strict and relaxed-clock models, including analysis of serially sampled taxa
- Gene-tree/species-tree inference under the multi-species coalescent model (based on the method described by Liu and Pearl, 2007)
- Model averaging over the GTR family of models using reversible-jump MCMC (rjMCMC)
- Model selection using Bayes factors (comparing marginal likelihoods for competing models)

This tutorial demonstrates two types of analyses that might be required for a typical empirical study. Specifically, we will demonstrate how to use MrBayes v3.2 to perform the following analyses:

1. Setting up and selecting among partition schemes (mixed models) using stepping-stone sampling
2. Performing model averaging over members of the GTR substitution model family using rjMCMC

Each exercise will work through the series of commands required to perform the analyses interactively from the command line, and will conclude with the equivalent batch of commands required to perform the analyses in batch mode. MrBayes relies on the `NEXUS` file format for specifying the data alignment and program-specific commands. The batch files and data alignment are in this format.

Getting Started

This tutorial assumes that you have already downloaded, compiled, and installed MrBayes v3.2 (Ronquist et al., 2012). We also recommend that—if you are working on a Unix machine—you put the `mb` binary in your path.

For the exercises outlined in this tutorial, we will use MrBayes interactively by typing commands in the command-line program. The format of this exercise uses text bullets (•) to delineate important steps. The various MrBayes commands are specified using **`typewriter text`**. And the specific commands that you should type (or copy/paste) into MrBayes are indicated by the text bullet and prompt. For example, after opening the MrBayes program, you can execute your data file:

- **`MrBayes > execute data-file.nex`**

For this command, type in the command and its options: **`execute data-file.nex`**. **DO NOT** type in "**`MrBayes >`**", the prompt is included simply included to replicate what you see on your screen.

This tutorial also includes hyperlinks: bibliographic citations are burnt orange and link to the full citation in the references, external URLs are cerulean, and internal references to figures and equations are purple.

The various exercises in this tutorial take you through the steps required to perform phylogenetic analyses of the example datasets. In addition, we have provided the output files for every exercise so you can verify your results. (Note that since the MCMC runs you perform will start from different random seeds, the output files resulting from your analyses *will not* be identical to the ones we provide you.)

- Download data and output files from: http://treethinkers.org/phylogenetic-inference-using-mrbayes-v3-2/

1 Model Selection & Partitioning using Bayes Factors

Variation in the evolutionary process across the sites of nucleotide sequence alignments is well established, and is an increasingly pervasive feature of datasets composed of gene regions sampled from multiple loci and/or different genomes. Inference of phylogeny from these data demands that we adequately model the underlying process heterogeneity; failure to do so can lead to biased estimates of phylogeny and other parameters (Brown and Lemmon, 2007). To accommodate process heterogeneity within and/or between various gene(omic) regions, we will evaluate the support for various partition schemes using Bayes factors to compare the marginal likelihoods of the candidate partition schemes.

Accounting for process heterogeneity involves adopting a 'mixed-model' approach, (Ronquist and Huelsenbeck, 2003) in which the sequence alignment is first parsed into a number of partitions that are intended to capture plausible process heterogeneity within the data. The determination of the partitioning scheme is guided by biological considerations regarding the dataset at hand. For example, we might wish to evaluate possible variation in the evolutionary process within a single gene region (*e.g.*, between stem and loop regions of ribosomal sequences), or among gene regions in a concatenated alignment (*e.g.*, comprising multiple nuclear loci and/or gene regions sampled from different genomes). The choice of partitioning scheme is up to the investigator and many possible partitions might be considered for a typical dataset.

Next, a substitution model is specified for each predefined process partition (using a given model-selection criterion, such as Bayes factors, the hierarchical likelihood ratio test, or the Akaike information criterion).

This results in a composite model, in which all sites are assumed to share a common tree topology, denoted τ, and proportional branch lengths, ν, but subsets of sites ('data partitions') are assumed to have independent substitution model parameters (*e.g.*, for the relative substitution rates, θ_{ij}, stationary frequencies, π_i, degree of gamma-distributed among-site rate variation, α, etc.). This composite model is referred to as a *mixed model*.

Finally, we perform a separate MCMC simulation to approximate the joint posterior probability density of the phylogeny and other parameters. Note that, in this approach, the mixed model is a fixed assumption of the inference (*i.e.*, the parameter estimates are conditioned on the specified mixed model), and the parameters for each process partition are independently estimated.

For most sequence alignments, several (possibly many) partition schemes of varying complexity are plausible *a priori*, which therefore requires a way to objectively identify the partition scheme that balances estimation bias and error variance associated with under- and over-parameterized mixed models, respectively. Increasingly, mixed-model selection is based on *Bayes factors* (*e.g.*, Suchard, Weiss and Sinsheimer, 2001), which involves first calculating the marginal likelihood under each candidate partition scheme and then comparing the ratio of the marginal likelihoods for the set of candidate partition schemes (Brandley, Schmitz and Reeder, 2005; Nylander et al., 2004; McGuire et al., 2007). The analysis pipeline that we will use in this tutorial is depicted in Figure 1.

Given two models, M_0 and M_1, the Bayes factor comparison assessing the relative plausibility of each model as an explanation of the data, $BF(M_0, M_1)$, is:

$$BF(M_0, M_1) = \frac{\text{posterior odds}}{\text{prior odds}}.$$

The posterior odds is the posterior probability of M_0 given the data, $\mathbf{X}$, divided by the posterior odds of M_1 given the data:

$$\text{posterior odds} = \frac{\mathbb{P}(M_0 \mid \mathbf{X})}{\mathbb{P}(M_1 \mid \mathbf{X})},$$

and the prior odds is the prior probability of M_0 divided by the prior probability of M_1:

$$\text{prior odds} = \frac{\mathbb{P}(M_0)}{\mathbb{P}(M_1)}.$$

Thus, the Bayes factor measures the degree to which the data alter our belief regarding the support for M_0 relative to M_1 (Lavine and Schervish, 1999):

$$BF(M_0, M_1) = \frac{\mathbb{P}(M_0 \mid \mathbf{X}, \theta_0)}{\mathbb{P}(M_1 \mid \mathbf{X}, \theta_1)} \div \frac{\mathbb{P}(M_0)}{\mathbb{P}(M_1)}. \tag{1}$$

This, somewhat vague, definition does not lead to clear-cut identification of the "best" model. Instead, you must decide the degree of your belief in M_0 relative to M_1. Despite the absence of any strict "rule-of-thumb", you can refer to the scale (outlined by Jeffreys, 1961) for interpreting these measures (Table 1).

Unfortunately, direct calculation of the posterior odds to prior odds ratio is unfeasible for most phylogenetic models. However, we can further define the posterior odds ratio as:

$$\frac{\mathbb{P}(M_0 \mid \mathbf{X})}{\mathbb{P}(M_1 \mid \mathbf{X})} = \frac{\mathbb{P}(M_0)}{\mathbb{P}(M_1)} \frac{\mathbb{P}(\mathbf{X} \mid M_0)}{\mathbb{P}(\mathbf{X} \mid M_1)},$$

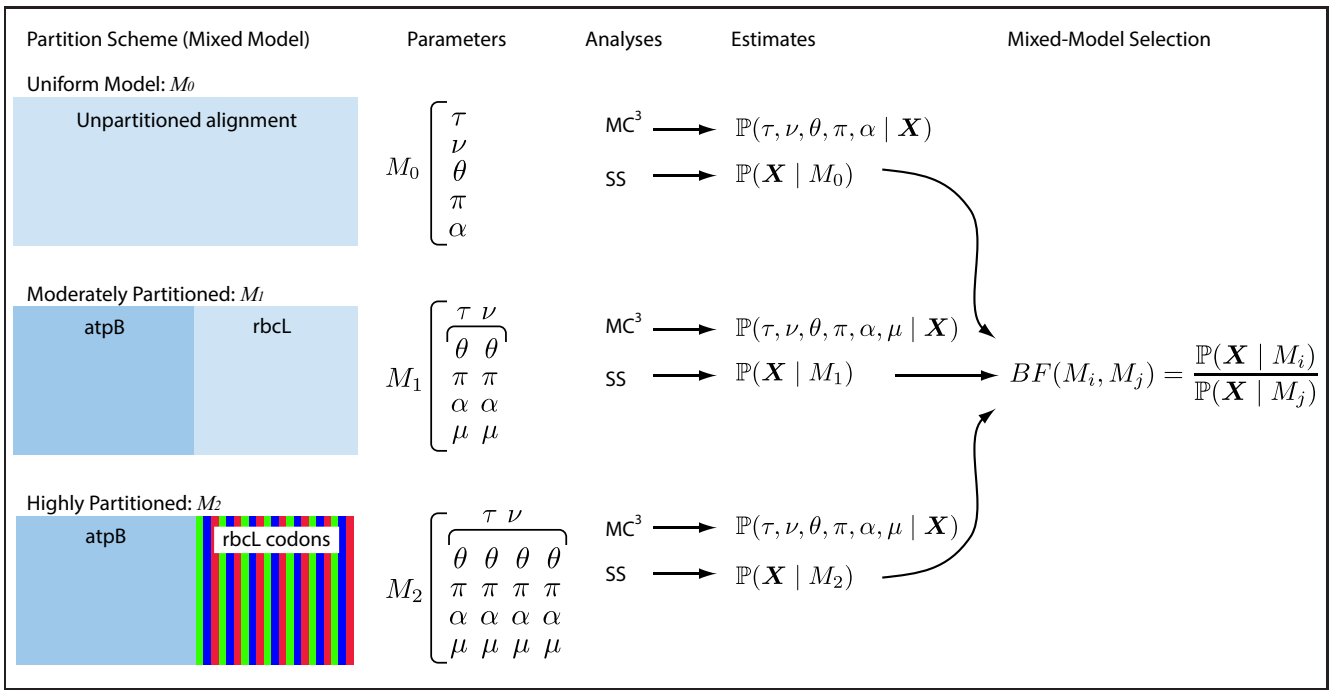

Figure 1: The analysis pipeline for Exercise 1. We will explore three partition schemes for the conifer dataset. The first model (the 'uniform model', M_0) assumes that all sites evolved under a common GTR+Γ substitution model. The second model (the 'moderately partitioned' model, M_1) invokes two data partitions corresponding to the two gene regions (atpB and rbcL), and assumes each subset of sites evolved under an independent GTR+Γ model, each with its own rate multiplier, μ. The final mixed model (the 'highly partitioned' model, M_2) invokes four data partitions—the first partition corresponds to the atpB gene region, and the remaining partitions correspond to the three codon positons of the rbcL gene region—and each data partition is assumed evolved under an independent GTR+Γ substitution model. Note that we assume that all sites share a common tree topology, τ, and branch-length proportions, ν, for each of the candidate partition schemes. We perform two separate sets of analyses for each mixed model—a Metropolis-coupled MCMC simulation to approximate the joint posterior probability density of the mixed-model parameters, and a 'stepping-stone' MCMC simulation to approximate the marginal likelihood for each mixed model. The resulting marginal-likelihood estimates are then evaluated using Bayes factors to assess the fit of the data to the three candidate mixed models.

where $\mathbb{P}(\mathbf{X} \mid M_i)$ is the *marginal likelihood* of the data marginalized over all parameters for M_i; it is also referred to as the *model evidence* or *integrated likelihood*. More explicitly, the marginal likelihood is the probability of the set of observed data ($\mathbf{X}$) under a given model (M_i), while averaging over all possible values of the parameters of the model (θ_i) with respect to the prior density on θ_i

$$\mathbb{P}(\mathbf{X} \mid M_i) = \int \mathbb{P}(\mathbf{X} \mid \theta_i)\mathbb{P}(\theta_i)dt. \tag{2}$$

If you refer back to equation 1, you can see that, with very little algebra, the ratio of marginal likelihoods is equal to the Bayes factor:

$$BF(M_0, M_1) = \frac{\mathbb{P}(\mathbf{X} \mid M_0)}{\mathbb{P}(\mathbf{X} \mid M_1)} = \frac{\mathbb{P}(M_0 \mid \mathbf{X}, \theta_0)}{\mathbb{P}(M_1 \mid \mathbf{X}, \theta_1)} \div \frac{\mathbb{P}(M_0)}{\mathbb{P}(M_1)}. \tag{3}$$

Therefore, we can perform a Bayes factor comparison of two models by calculating the marginal likelihood for each one. Alas, exact solutions for calculating marginal likelihoods are not known for phylogenetic models (see equation 2), thus we must resort to numerical integration methods to estimate or approximate these values. In this exercise, we will estimate the marginal likelihood for each partition scheme using both the harmonic-mean (unreliable) and stepping-stone (preferable) estimators.

Table 1: The scale for interpreting Bayes factors by Harold Jeffreys (1961).

$BF(M_0, M_1)$	Strength of evidence
$< 1:1$	Negative (supports M_1)
$1:1$ to $3:1$	Barely worth mentioning
$3:1$ to $10:1$	Substantial
$10:1$ to $30:1$	Strong
$30:1$ to $100:1$	Very strong
$> 100:1$	Decisive

For a detailed description of Bayes factors see Kass and Raftery (1995)

- Open the file **conifer_dna.nex** in your text editor. This file contains the sequences for 2 different genes sampled from 9 species (Box 1). The elements of the **DATA** block indicate the type of data, number of taxa, and length of the sequences.

```
#NEXUS
BEGIN DATA;
    DIMENSIONS NTAX=9 NCHAR=2659;
    FORMAT DATATYPE = DNA GAP = - MISSING = ?  INTERLEAVE;
    MATRIX
[ATPB]
    Ginkgo_biloba              TTATTGGTCCAGTACTGGATGTAGCTTTTCCCCCGGG...
    Araucaria_araucana         ---GGTCCGGTACTGGATGTATCTTTTCCTCCAGA...
    Cedrus_deodara             TCATTGGCCCAGTACTGGA?GTCTCTTTTCCTCCAGG...
    Cupressus_arizonica        ---------GATGTATCTTTCCCTCCAGG...
    Juniperus_communis         -------------------...
    Pinus_densiflora           TCATTGGCCCAGTACTGGATGTCTCTTTTCCTCCAGG...
    Podocarpus_chinensis       TCATCGGCCCTGTACTGGATGTATCTTTTCCTCCAGA...
    Sciadopitys_verticillata   TCATTGGTCCAGTACTAGATGTATCTTTCCCTCCAGG...
    Taxus_baccata              TTATCGGCCCAGTACTAGATGTCTCTTTTCCTCCAGG...

[RBCL]
    Ginkgo_biloba                ATGGATAAGTT---------AAAGAG...
    Araucaria_araucana           ATAGATTAACTACTCCGCAATATCAGACCAAAGAT...
    Cedrus_deodara               ACAGATTAACTACTCCTGAATATCAGACCAAAGAT...
    Cupressus_arizonica          --ATTAACTACTCCGGAATATCAGACCAAAGAT...
    Juniperus_communis           ACAGATTAACTACTCCGGAATATCAGACAAAAGAT...
    Pinus_densiflora             ACAGATTAACTACTCCTGAATATCAGACCAAAGAT...
    Podocarpus_chinensis         ACAGATTAACTACTCCGGAATATCAGACCAAAGAT...
    Sciadopitys_verticillata     ACAGATTAACTACCCCTGAATATCAGACCAAAGAC...
    Taxus_baccata                ACAGACTAACTACTCCACAATATCAGACCAAAGAT...
;
END;
```

Box 1: A fragment of the NEXUS file containing the sequences for this exercise.

- Open the batch file, **conifer_partn.nex**, in a text editor. This file contains all of the commands required to perform the necessary analyses to explore various partition schemes (unpartitioned, partitioned by gene region, and partitioned by gene region+codon position). The details of each command are described in adjacent comments, surrounded in brackets; *e.g.*, **[this is a comment]**.

Typically, we would perform these analyses by simply executing this batch file. For the purposes of this exercise, however, we will walk through the different steps interactively in the command line.

Load the Sequences & Specify the Outgroup

Execute the MrBayes binary. If this program is in your path, then you can simply type in your Unix terminal:

- `> mb`

When you execute the program, you will see the program information, including the current version number and commands that will provide information about the history and main authors of the program (**about** and **acknowledgments**) and a command reference **help**. Execute the **help** function by typing:

- `MrBayes > help`

This displays a list of the different elements and commands available in MrBayes. The **help** command also provides more detailed information about each of these items.

For example, we can view the **help** information about the **log** command:

- `MrBayes > help log`

The **log** command allows you to save all of the screen output from your analysis to a log file. The **help** information for this command displays all of the available options for specifying screen logging. We are going to log our screen output to a file called **conifer-partn-log.txt**.

- `MrBayes > log start filename=conifer-partn-log.txt`

Next, load the sequences into the program using the **execute** command.

- `MrBayes > execute conifer_dna.nex`

Now that MrBayes has read in our data, we can define our outgroup taxon. Unless a clock-based analysis is specified, MrBayes v3.2 infers *unrooted* trees, however trees are written to output files as rooted trees (unrooted trees are not phylogenies as they do not specify a temporal direction). Accordingly, this command specifies how we would like our trees written to file. If we do not specify an outgroup, the trees will be rooted on the first species in the data matrix by default.

- `MrBayes > outgroup Ginkgo_biloba`

1.1 An Unpartitioned Analysis

The first analysis in this exercise involves performing an analysis on our unpartitioned alignment. This corresponds to the assumption that the process that gave rise to our data was homogeneous across all sites of the alignment. Specifically, we will assume that both genes evolved under the same GTR+Γ model (Fig. 1). The **lset** command is used to specify the details of our sequence model.

- `MrBayes > lset nst=6 rates=gamma`

This command specifies a substitution matrix with six relative substitution rates (GTR) with gamma-distributed rate variation across sites. Because models are specified this way, it is apparent that some types of DNA models are not available in MrBayes. Thus, with the **nst** element of the **lset** command, we can specify the JC69 or F81 models (**nst=1**), the K2P or HKY models (**nst=2**), or the GTR model (**nst=6**).

The Bayesian perspective views parameters as random variables, which requires that we specify a prior probability density that describes the precise nature of that random variation. Accordingly, we need to specify priors for all of the parameters of the specified nucleotide substitution model. The command for modifying priors is the **prset** command.

Use the **help** command to view the list of priors available for modification:

- **MrBayes > help prset**

First, we will parameterize a flat Dirichlet prior on the 6 exchangeability parameters.

- **MrBayes > prset revmatpr=dirichlet(1,1,1,1,1,1)**

The Dirichlet distribution assigns probability densities to grouped parameters: *e.g.*, those that measure proportions and must sum to 1. Above, we specified a 6-parameter Dirichlet prior on the relative rates of the GTR model, where the placement of each value specified represents one of the 6 relative rates: (1) $A \leftrightarrows C$, (2) $A \leftrightarrows G$, (3) $A \leftrightarrows T$, (4) $C \leftrightarrows G$, (5) $C \leftrightarrows T$, (6) $G \leftrightarrows T$. The input parameters of a Dirichlet distribution are called shape parameters or concentration parameters and a value is specified for each of the 6 GTR rates. The expectation and variance for each variable are related to the sum of the shape parameters. The prior above is a 'flat' or symmetric Dirichlet since all of the shape parameters are equal (1,1,1,1,1,1), thus we are specifying a model that allows for equal rates of change between nucleotides, such that the expected rate for each is equal to $\frac{1}{6}$ (Zwickl and Holder, 2004). Figure 2a shows the probability density of each rate under this model. If we parameterized the Dirichlet distribution such that all of the parameters were equal to 100, this would also specify a prior with an expectation of equal exchangeability rates (Figure 2b). However, by increasing the shape parameters of the Dirichlet distribution, **dirichlet(100,100,100,100,100,100)**, would heavily restrict the MCMC from sampling sets of GTR rates in which the values were not equal or very nearly equal (*i.e.*, this is a very *informative* prior). We can consider a different Dirichlet parameterization if we had strong prior belief that transitions and transversions occurred at different rates. In this case, we could specify a more informative prior density: **dirichlet(4,8,4,4,8,4)**. Under this model, the expected rate for transversions would be $\frac{4}{32}$ and the expected rate for transitions would equal $\frac{8}{32}$, and there would be greater prior probability on sets of GTR rates that matched this configuration (Figure 2c). An alternative informative prior would be one where we assumed that each of the 6 GTR rates had a different value conforming to a Dirichlet(2,4,6,8,10,12). This would lead to a different prior probability density for each rate parameter (Figure 2d). Without strong prior knowledge about the pattern of relative rates, however, we can better capture our statistical uncertainty with a vague prior on the GTR rates. Notably, all patterns of relative rates have the same probability under **dirichlet(1,1,1,1,1,1)**.

We can use the same type of distribution as a prior on the 4 base frequencies ($\pi_A, \pi_C, \pi_G, \pi_T$) since these parameters also represent proportions. Specify a flat Dirichlet prior density on the base frequencies:

- **MrBayes > prset statefreqpr=dirichlet(1,1,1,1)**

When we specified our model with the **lset** command, we also indicated that the substitution rates varied among sites according to a gamma distribution, which has two parameters: the shape parameter, α, and the scale parameter, β. In order that we can interpret the branch lengths as the expected number of substitutions per site, this model assumes that the mean site rate is equal to 1. The mean of the gamma is equal to α/β, so a mean-one gamma is specified by setting the two parameters to be equal, $\alpha = \beta$. Therefore, we need only consider the single shape parameter, α (Yang, 1994). The degree of among-site substitution rate variation (ASRV) is inversely proportional to the value of the shape parameter—as the

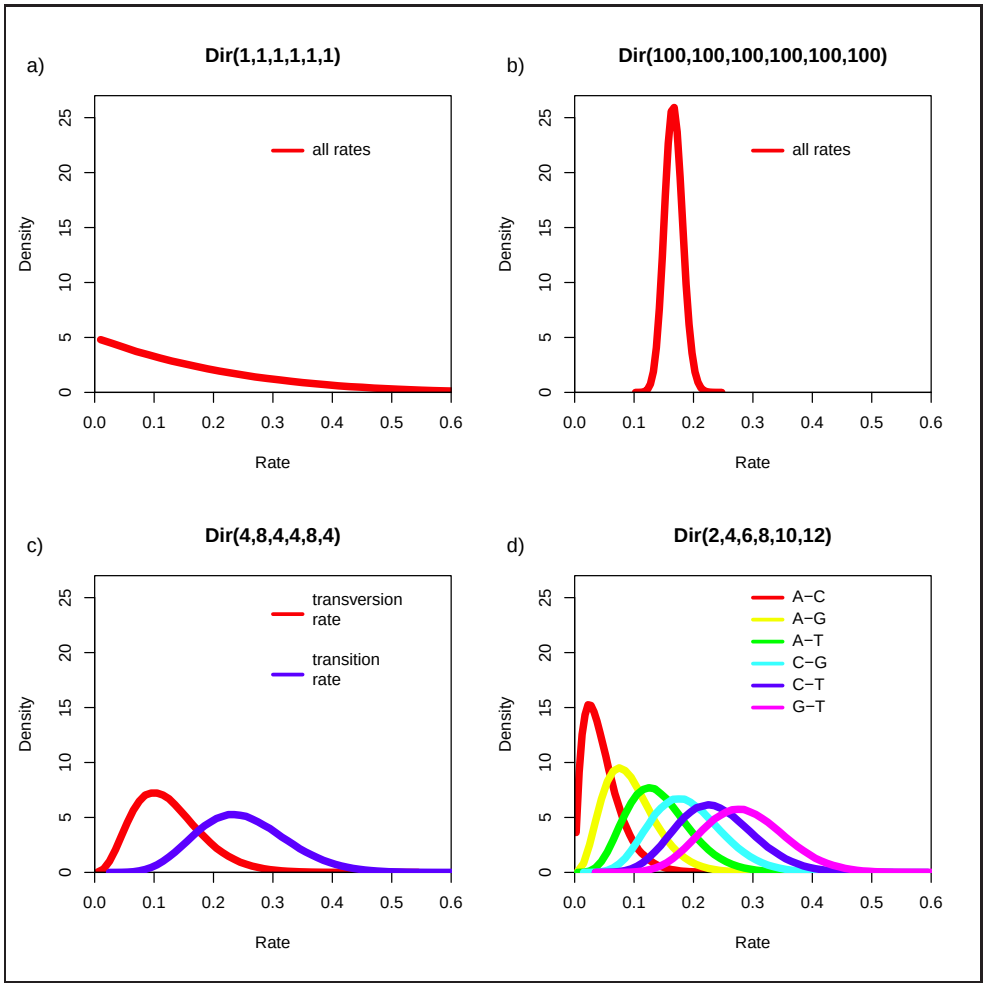

Figure 2: Four different examples of Dirichlet priors on exchangeability rates.

value of α-shape parameter increases, the gamma distribution increasingly resembles a normal distribution with decreasing variance, which corresponds to decreasing levels of ASRV (Figure 3). If $\alpha = 1$, then the gamma distribution collapses to an exponential distribution with a rate parameter equal to β. By contrast, when the value of the α-shape parameter is < 1, the gamma distribution assumes a concave distribution that places most of the prior density on low rates but allows some prior mass on sites with very high rates, which corresponds to high levels of ASRV (Figure 3).

Alternatively, we might not have good prior knowledge about the variance in site rates, thus we can place an uninformative, or diffuse prior on the shape parameter. For this analysis, we will use an exponential distribution with a rate parameter, λ, equal to `0.05`. Under an exponential prior, we are placing non-zero probability on values of α ranging from 0 to ∞. The rate parameter, λ, of the exponential distribution controls both the mean and variance of this prior such that the expected (or mean) value of α is:

$$\mathbb{E}[\alpha] = \frac{1}{\lambda}.$$

Thus, if we set $\lambda = 0.05$, then $\mathbb{E}[\alpha] = 20$. The gamma shape parameter is called **`shapepr`** in the **`prset`** command.

- `MrBayes > prset shapepr=exponential(0.05)`

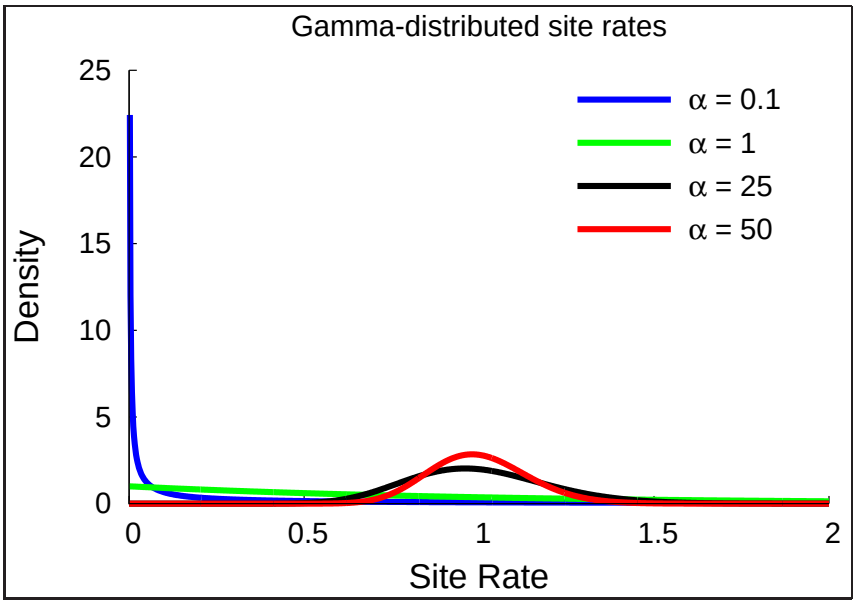

Figure 3: The probability density of mean-one gamma-distributed rates under different shape parameters.

Now we have specified a simple, single-partition analysis—each parameter of the model will be estimated from every site in our alignment. We can review the various model settings and parameters that we have specified for this dataset using the **showmodel** command.

- `MrBayes > showmodel`

This provides a summary of the parameters of your model and all of the priors and their hyperparameters that you have specified (in Bayesian inference the parameter characterizing a prior distribution is called a *hyperparameter* to distinguish them from parameters directly involved in the likelihood function).

Now we are ready to set up the details of the MCMC. In MrBayes, one can specify the details of the MCMC simulation using either the **mcmcp** or **mcmc** commands. The first command, **mcmcp**, reads in the settings—the number of iterations, file names, number of runs, number of chains, etc.—for the MCMC simulation, but does not initiate the MCMC analysis. By contrast, the **mcmc** command reads in the settings and then immediately initiates the run. Thus, when working interactively with MrBayes, it is recommended that you first set up the MCMC using **mcmcp**. Begin by setting the chain length (**ngen**) and the frequency that states are saved to file (**samplefreq**) and printed to screen (**printfreq**).

- `MrBayes > mcmcp ngen=300000 printfreq=100 samplefreq=100`

We will use Metropolis-coupled MCMC (often called MC^3; Geyer, 1991) for our analysis and we can set the number of chains using the **nchains** option. Also, since we wish to monitor branch-length information as well as the topological states sampled by the chain, indicate this in the **mcmcp** settings.

- `MrBayes > mcmcp nchains=4 savebrlens=yes`

During our run, we can monitor statistics that will help us assess whether the independent runs have converged on the same stationary distribution. For this analysis, use the maximum standard deviation of split frequencies (**maxstddev**) and perform that diagnostic every **diagnfreq=1000** steps.

- `MrBayes > mcmcp nruns=2 diagnfreq=1000 diagnstat=maxstddev`

Finally, we can set the output file-name stem (it is the 'base' to which various suffixes will be appended for the various types of output files). Set the file-name prefix to **conifer-uniform**, so that the output files will be saved as **conifer-uniform.run1.p**, **conifer-uniform.run2.p**, **conifer-uniform.run2.t**, **conifer-uniform.run2.t**, etc.

- **MrBayes > mcmcp filename=conifer-uniform**

Once you have specified all of the MCMC settings, execute the Markov chain with the **mcmc** command.

- **MrBayes > mcmc**

Depending on your computer, the runs should complete in a few minutes (it's not strictly a race, but feel free to wager with your neighbors). When running interactively, MrBayes will give you the option to continue sampling once the chain has reached **ngen** iterations:

- **Continue with analysis? (yes/no): no**

This is the million-dollar question! In theory, a properly constructed MCMC algorithm is guaranteed to provide a precise description of the joint posterior probability density **IF** it is run for an *infinite* number of cycles; and an adequate approximation of the posterior distribution can be generated after a *sufficient* number of iterations. The question is whether the length of this particular analysis—to infer parameters of this particular phylogenetic model from this particular dataset—is sufficient or insufficient.

We base this decision on diagnostic tools designed to detect when the MCMC simulation has failed to adequately approximate the joint posterior probability density. There are many diagnostics tools at our disposal, and you should use them judiciously if you are a good person (*e.g.*, Nylander et al., 2008; Rambaut and Drummond, 2009). One such tool is the maximum standard deviation of split frequencies diagnostic. Let's take a moment to understand how this diagnostic works.

Convergence is an important aspect of MCMC performance: essentially, we wish to know whether the chain has targeted the stationary distribution (the joint posterior probability density of parameters that we are attempting to approximate). One approach for assessing convergence is to compare samples from multiple, independent chains that have been initiated from random starting points in the joint posterior probability density. Initially, samples from these chains will be quite disparate (because the chains were initiated from different points in parameter space), but are expected to become increasingly similar as the chains converge to the stationary distribution. This is the gist of the maximum standard deviation of split frequencies diagnostic: it monitors the topological similarity of trees sampled by two (or more) independent chains.

The rationale of the **maxstddev** and ASDSF diagnostics is quite simple. A low standard deviation indicates that the data points (split frequencies) tend to be close to the mean, suggesting that the trees sampled by the independent chains are similar and presumably sampled from the same (stationary) distribution. By contrast, high values of the standard deviation indicates that the data points (split frequencies) tend to deviate greatly from the mean, suggesting that the trees sampled by the independent chains are quite different and presumably not sampled from the same (stationary) distribution.

When running remote jobs, you can always specify very large chain lengths since it is far easier to intermittently monitor your runs and terminate those that are mixing well and have converged. Since we don't have that much time, terminate the Markov chain after 300,000 iterations for this exercise.

We are interested in comparing the marginal likelihood of the unpartitioned analysis to other partition configurations of our alignment using Bayes factors. This step requires a second analysis using stepping-stone sampling (Xie et al., 2011; Fan et al., 2011). Specify the parameters of the stepping-stone run using the **ssp** command.

- **MrBayes > ssp ngen=100000 diagnfreq=1000 filename=conifer-uniform-ss**

Now execute the stepping-stone analysis.

- **MrBayes > ss**

MrBayes is now running an MCMC simulation that steps from the joint posterior probability density to the joint prior probability density of parameters in order to estimate the marginal likelihood, which we can use later to compare the unpartitioned mixed model to the alternative partition schemes.

- Once the stepping-stone sampling run has completed, the estimated *stepping-stone* marginal likelihood for the uniform partition is reported to the screen. **Record** the **Mean** marginal likelihood for the 2 runs.

1.2 Partitioning by Gene Region

The uniform model used in the previous section assumes that all sites in the alignment evolved under the same process described by a shared tree, branch length proportions, and parameters of the GTR+Γ substitution model. However, our alignment contains two distinct gene regions—atpB and rbcL—so we may wish to explore the possibility that the substitution process differs between these two gene regions. This requires that we first specify the data partitions corresponding to these two genes, then define an independent substitution model for each data partition.

In MrBayes, we can define the subset of sites belonging to each of the gene regions using the **charset** command. Indicate sites 1–1,394 belong to the atpB gene and sites 1,395–2,659 are from rbcL:

- **MrBayes > charset atpB = 1-1394**
- **MrBayes > charset rbcL = 1395-2659**

In the next step, we can define our partition configuration using the **partition** command.

- **MrBayes > partition partition-by-gene = 2: atpB, rbcL**

Here, we have created a data partition called **partition-by-gene** that contains two subsets of sites within our alignment. Next, set the partition scheme to **partition-by-gene**.

- **MrBayes > set partition=partition-by-gene**

For these two genes, we are assuming that both evolved under a GTR+Γ model. By separating them and partitioning by gene, we are specifying that the substitution-model parameters are independent for each gene. Thus, atpB evolved under a GTR+Γ model with its own set of base frequencies, exchangeability rates, and gamma shape parameter; and rbcL evolved under another GTR+Γ model with a different set of base frequencies, exchangeability rates, and gamma shape parameter. First, use the **lset** command to indicate the type of sequence model for each gene. The **applyto** option specifies that we are assuming the same model type for each gene.

- `MrBayes > lset applyto=(all) nst=6 rates=gamma`

As in the first part of this exercise, we can also specify the priors.

- `MrBayes > prset revmatpr=dirichlet(1,1,1,1,1,1) statefreqpr=dirichlet(1,1,1,1)`
- `MrBayes > prset shapepr=exponential(0.05)`

At this point, the parameters for each gene are linked, and if we ran the MCMC under these settings, we would be performing an unpartitioned analysis. We must now **unlink** the parameters for each gene so that they will be estimated independently.

- `MrBayes > unlink revmat=(all) statefreq=(all) shape = (all)`

Now, each gene has a set of partition-specific parameters. Next, we must allow the overall substitution rate to vary across the subsets of our alignment. This is done using the **prset** command.

- `MrBayes > prset applyto=(all) ratepr=variable`

We are now ready to set up the MCMC and run our chains. Start by using the **mcmcp** command to set the values of our MCMC options. (Notice that you can abbreviate commands, provided that they are still unique.)

- `MrBayes > mcmcp ng=300000 printf=100 samp=100 diagnf=1000 diagnst=maxstddev`
- `MrBayes > mcmcp nch=4 savebr=yes filename=conifer-partn`

Execute the Markov chain with the **mcmc** command:

- `MrBayes > mcmc`

Do not continue this Markov chain when it reaches 300,000 generations (for the sake of time).

- `Continue with analysis? (yes/no): no`

Since we will compare the partitioned analysis to the unpartitioned run, we need to approximate the marginal likelihood of this model specification. Set up and execute a stepping-stone analysis using the **ss** command.

- `MrBayes > ss ng=100000 diagnfr=1000 filename=conifer-partn-ss`

MrBayes will now run an MCMC simulation that steps from the joint posterior probability density to the joint prior probability density of parameters in order to estimate the marginal likelihood, which we can use later to compare the unpartitioned mixed model to the alternative partition schemes.

- Once the stepping-stone sampling run has completed, the estimated *stepping-stone* marginal likelihood for the partitioned-by-gene analysis is reported to the screen. **Record** the **Mean** marginal likelihood for the 2 runs.

1.3 Partitioning by Codon Position and by Gene

Because of the genetic code, we often find that different positions within a codon (first, second, and third) evolve at different rates. Thus, using our knowledge of biological data, we can devise a third approach that further partitions our alignment. For this exercise, we will partition sites within the rbcL gene by codon position.

The genes in our alignment each start at position 1 (*i.e.* they are in frame). When defining a **charset** we can specify the different positions using specific notation indicating that every third base-pair belongs in a given **charset**.

- `MrBayes > charset rbcL1stpos = 1395-2659\3`
- `MrBayes > charset rbcL2ndpos = 1396-2659\3`
- `MrBayes > charset rbcL3rdpos = 1397-2659\3`

Notice that each charset begins with a different character position in our alignment. This shifts the 3-base frame so that the proper sites are included in each character set.

Now we will specify four different subsets of the alignment: the entire atpB gene, rbcL 1st positions, rbcL 2nd positions, and rbcL 3rd positions.

- `MrBayes > partition sat-partition = 4: atpB, rbcL1stpos, rbcL2ndpos, rbcL3rdpos`
- `MrBayes > set partition=sat-partition`

For this exercise, we will again assume a GTR+Γ model for every character set, but unlink the parameters across our subsets.

- `MrBayes > lset applyto=(all) nst=6 rates=gamma`
- `MrBayes > prset revm=dir(1,1,1,1,1,1) statef=dir(1,1,1,1) shape=expon(0.05)`
- `MrBayes > unlink revmat=(all) statef=(all) shape=(all)`
- `MrBayes > prset applyto=(all) ratepr=variable`

We set up the MCMC and run our chains as described for the previous models. Start by using the **mcmcp** command to set the values of our MCMC parameters.

- `MrBayes > mcmcp ng=300000 printf=100 samp=100 diagnf=1000 diagnst=maxstddev`
- `MrBayes > mcmcp nch=4 savebr=yes filename=conifer-sat-partn`

Execute the Markov chain with the **mcmc** command:

- `MrBayes > mcmc`

Do not continue this Markov chain when it reaches 300,000 generations (ideally, we would run this much longer).

- `Continue with analysis? (yes/no): no`

Although there is evidence that the MCMC has not yet provided an adequate sample of the joint posterior probability density of parameters, because time is short, we will terminate terminate the analysis.

As for the previous mixed models, we will next approximate the marginal likelihood of this mixed model specification. Set up and execute a stepping-stone analysis using the **ss** command.

- **MrBayes > ss ng=100000 diagnfr=1000 filename=conifer-sat-partn-ss**

- Once the stepping-stone sampling run has completed, the estimated *stepping-stone* marginal likelihood for the highly partitioned analysis is reported to the screen. **Record** the **Mean** marginal likelihood for the 2 runs.

1.4 Summarize and Analyze

MrBayes has some built-in tools for summarizing MCMC samples of trees and other parameters. The commands for performing these analyses are **sumt** and **sump**. The **sumt** command summarizes the samples of the tree topology and branch lengths and **sump** summarizes all of the other model parameters (*e.g.* base frequencies, shape parameter, etc.).

We will begin by assessing the scalar parameters sampled by MCMC for our 3 different analyses using the **sump** command. By default this command will discard the first 25% of the samples as 'burn-in'. Begin with the uniform analysis:

- **MrBayes > sump filename=conifer-uniform**

Upon completion of the **sump** command, you will see a table listing the estimated marginal likelihoods of these analyses.

- Record the marginal likelihood estimated by the harmonic mean for the uniform partition analysis.

- Review the table summarizing the MCMC samples of the various parameters. Notice that this table contains the **Mean**, **Median**, **Variance**, etc.

This table also give the 95% credible interval of each parameter. This statistic approximates the 95% highest posterior density (HPD) and is a measure of uncertainty while *accounting for the data* (MrBayes labels this value as **95% HPD**). More specifically, the probability that the true value of the parameter lies within the credible interval is 0.95 *given the model and the data*.

- Continue summarizing the MC^3 runs for the moderately partitioned run and the highly partitioned run and record the **Harmonic mean** estimate of the marginal likelihood for each.

- **MrBayes > sump filename=conifer-partn**

- **MrBayes > sump filename=conifer-sat-partn**

Now that we have estimates of the marginal likelihood under each of our different models, we can evaluate their relative plausibility using Bayes factors. Use Table 2 to summarize the marginal log-likelihoods estimated using the harmonic mean and stepping-stone methods.

Table 2: Estimated marginal likelihoods for different partition configurations*.

	Marginal lnL estimates	
Partition	*Harmonic mean*	*Stepping-stone*
1.1 uniform (M_1)		
1.2 moderate (M_2)		
1.3 extreme (M_3)		

*you can edit this table

Phylogenetics software programs log-transform the likelihood to avoid underflow, because multiplying likelihoods results in numbers that are too small to be held in computer memory. Thus, we must use a different form of equation 3 to calculate the ln-Bayes factor (we will denote this value $\mathcal{K}$):

$$\mathcal{K} = \ln[BF(M_0, M_1)] = \ln[\mathbb{P}(\mathbf{X} \mid M_0)] - \ln[\mathbb{P}(\mathbf{X} \mid M_1)], \tag{4}$$

where $\ln[\mathbb{P}(\mathbf{X} \mid M_0)]$ is the *marginal lnL* estimate for model M_0. The value resulting from equation 4 can be converted to a raw Bayes factor by simply taking the exponent of $\mathcal{K}$

$$BF(M_0, M_1) = e^{\mathcal{K}}. \tag{5}$$

Alternatively, you can interpret the strength of evidence in favor of M_0 using the $\mathcal{K}$ and skip equation 5. In this case, we evaluate the $\mathcal{K}$ in favor of model M_0 against model M_1 so that:

if $\mathcal{K} > 1$, then model M_0 wins
if $\mathcal{K} < -1$, then model M_1 wins.

Thus, values of $\mathcal{K}$ around 0 indicate ambiguous support.

Using the values you entered in Table 2 and equation 4, calculate the ln-Bayes factors (using $\mathcal{K}$) for the different model comparisons. Enter your answers in Table 3 using the harmonic-mean and the stepping-stone estimates of the marginal log likelihoods.

Table 3: Bayes factor calculation*.

	ln-Bayes Factor ($\mathcal{K}$)	
Model comparison	*Harmonic mean*	*Stepping-stone*
M_1, M_2		
M_2, M_3		
M_1, M_3		
Supported model?		

*you can edit this table

Once you complete Table 3, you will notice that the Bayes factor comparison indicates strong evidence in support of the highly partitioned model using both the harmonic mean and stepping-stone estimates of

the marginal likelihoods. However, this does not mean that model M_3 is the *true* partition model. We only considered three out of the many, many possible partitions for 2,659 sites (the number of possible partitions can be viewed if you compute the 2659^{th} Bell number). Given the strength of support for the highly partitioned model, it is possible that further partitioning is warranted for these data. In particular, partitioning the dataset by codon position for both atpB *and* rbcL is an important next step for this exercise (consider taking some time on your own to test this model).

Because of the computational costs of computing marginal likelihoods and the vast number of possible partitioning strategies, it is not feasible to evaluate all of them. New methods based on nonparametric Bayesian models have recently been applied to address this problem (Lartillot and Philippe, 2004; Huelsenbeck and Suchard, 2007; Wu, Suchard and Drummond, 2013). These approaches use an infinite mixture model (the Dirichlet process; Ferguson, 1973; Antoniak, 1974) that places non-zero probability on *all* of the countably-infinite possible partitions for a set of sequences. Bayesian phylogenetic inference under these models is implemented in the program PhyloBayes (Lartillot, Lepage and Blanquart, 2009) and the subst-bma plug-in for BEAST2 (Wu, Suchard and Drummond, 2013).

Note that Bayes factors based on comparison of HM-based marginal likelihoods often *strongly* favor the most extremely partitioned mixed model. In fact, the harmonic mean estimator has been shown to provide unreliable estimates of marginal likelihoods, compared to more robust approaches (Lartillot and Philippe, 2006; Xie et al., 2011; Fan et al., 2011). Based on these studies, it is recommended that you avoid using HM-derived marginal likelihoods for Bayes factor comparisons. (The Canadian Bayesian Radford Neal says the harmonic mean is the "worst Monte Carlo method ever".)

Next, summarize the tree topologies and branch lengths sampled by MCMC for each of the different analyses. Begin with the uniform analysis:

- `MrBayes > sumt filename=conifer-uniform`

The primary summary performed by **sumt** calculates the clade credibility values (*i.e.*, bipartition posterior probabilities). These values are reported on an ASCII cladogram upon completion of the **sumt** command.

Continue summarizing the other two analyses:

- `MrBayes > sumt filename=conifer-partn`
- `MrBayes > sumt filename=conifer-sat-partn`

Then quit MrBayes:

- `MrBayes > quit`

The **sumt** command also writes the majority-rule consensus tree to a NEXUS tree file with the file-name extension `*.con.tre`. The trees in these files are also annotated with various branch- or node-specific parameters or statistics in an extended Newick format called NHX. We can use FigTree to visualize these summary trees.

- Open the summary trees in FigTree: `conifer-uniform.con.tre`, `conifer-partn.con.tre`, and `conifer-sat-partn.con.tre`.
- Use the tools on the side panel to display the posterior probabilities as node labels. An example is shown in Figure 4.

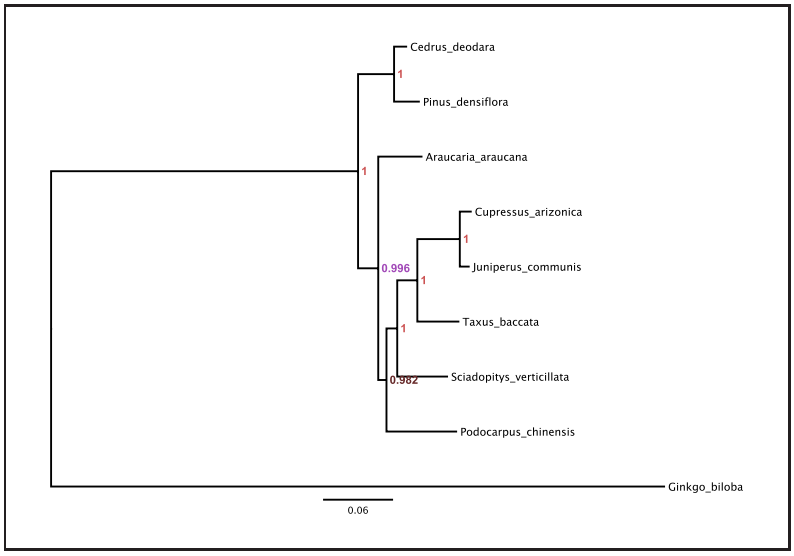

Figure 4: The summary tree from the uniform analysis, with posterior probabilities labeled at nodes.

Exercise 1 – Batch Mode

If you wish to run this exercise in batch mode, the files are provided for you. Open the file **conifer_partn.nex** in your text editor. This NEXUS-formatted file contains a **MrBayes** block with all of the commands you entered for exercise 1. The NEXUS command "**BEGIN**" specifies the type of block and all of the commands within a **MrBayes** block are directions for the analysis. The **MrBayes** block is then terminated by "**end;**". When including commands in a NEXUS block, these lines must be terminated by a semicolon "**;**". For example, the first command we give the program is to log the screen output to file:

- `log start filename=conifer-partn-log.txt;`

You will notice that several of the commands that you entered separately above are combined into a single command in the file:

- ```
  prset revmatpr=dirichlet(1,1,1,1,1,1)
  statefreqpr=dirichlet(1,1,1,1)
  shapepr=exponential(0.05);
  ```

The options for a single command, such as **prset** can span multiple lines as long as the last element is followed by a "**;**". Furthermore, we have annotated each element of the MrBayes commands with comments, which are contained within square braces: **[this is a comment]**.

The batch file allows you to save all of the commands for a single analysis and execute each command in succession without typing them in the command line. When executed in MrBayes, this file will provide the commands and MrBayes will execute each step of this exercise.

You can carry out these batch commands by providing the file name when you execute the **mb** binary in your unix terminal (this will overwrite all of your existing run files).

- `> mb conifer_partn.nex`
  ```

2 Averaging Over the GTR Family of Models

Model selection entails ranking a set of candidate models according to their relative fit to the data at hand. An emerging convention in our field is to first assess the *relative* fit of our dataset to a pool of candidate substitution models using maximum-likelihood based model-selection methods (*e.g.*, the likelihood-ratio test, AIC, BIC, etc.), and then proceed to estimate the phylogeny and other model parameters (via maximum likelihood or Bayesian inference methods) under the chosen model.

Even if we have exhaustively evaluated all possible candidate models and accurately assessed the relative fit of each model to a given dataset, it may nevertheless be unwise to condition our inference on the "best" model. This relates to the issue of model uncertainty. Imagine, for example, that there are several (possibly many) alternative models that provide a similarly good fit to a given dataset (Figure 5). In such scenarios, conditioning inference on *any single model* (even the 'best') ignores uncertainty in the chosen model and will cause estimates to be biased. This scenario is apt to become increasingly plausible as the size of the candidate model pool continues to grow.

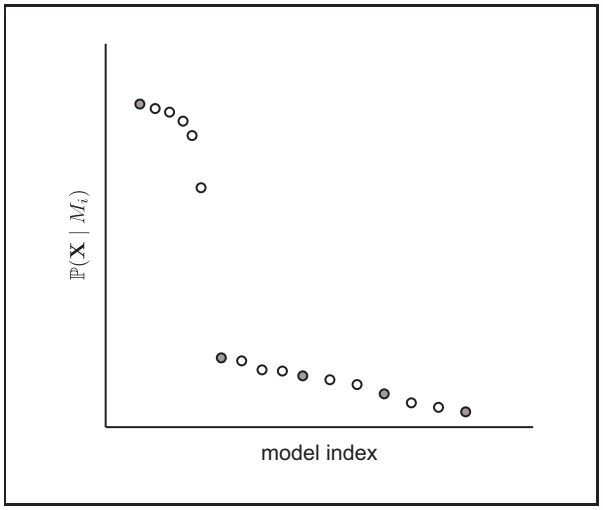

Figure 5: The relative fit of a dataset, $\mathbb{P}(\mathbf{X} \mid M_i)$, has been assessed for two sets of candidate models, M_i; the first set includes five models (filled circles), the second set includes an additional 12 models (filled and open circles). In the first scenario, it might seem reasonable to condition inference on M_1, which has a substantially better fit (*i.e.*, higher marginal likelihood) than the other four candidate models. In the second scenario, there are several models that provide a similarly good fit to the data. Failure to accommodate this uncertainty in the choice of model may lead to biased estimates.

Model uncertainty can be addressed by means of *model averaging*. The Bayesian framework provides a more natural approach for accommodating model uncertainty; we simply adopt the perspective that the models (like the parameters within each model) are themselves random variables.

Bayesian model averaging has been implemented for various phylogenetic problems using reversible-jump MCMC, where the chain integrates over the joint prior probability density of a given model in the usual manner, but also *jumps* between all possible candidate substitution models, visiting each model in proportion to its marginal probability. An approach for using rjMCMC to average over the pool of substitution models corresponding to all possible members of the GTR substitution model family was described by Huelsenbeck, Larget and Alfaro (2004). Analyses of empirical datasets using the approach demonstrate that the credible set typically contains many substitution models. Moreover, accommodating this source

of uncertainty has been shown to impact parameters of interest, providing less biased estimates of the topological variance and corresponding clade probabilities (Alfaro and Huelsenbeck, 2006). Here, we will demonstrate how to use this approach using MrBayes.

We can determine the pool of candidate models that can be specified by evaluating all possible combinations of the six exchangeability parameters, which is given by the Bell number (Bell, 1934). The Bell number for n elements is the sum of the Stirling numbers of the second kind:

$$\mathcal{B}(n) = \sum_{k=1}^{n} \mathcal{S}_2(n,k).$$

The Stirling number of the second kind, $\mathcal{S}_2(n,k)$, for n elements and k subsets (corresponding here to the number of relative rates in the GTR matrix and unique rate values, respectively) is given by the following equation:

$$\mathcal{S}_2(n,k) = \frac{1}{k!}\sum_{i=0}^{k-1}(-1)^i \binom{k}{i}(k-i)^n.$$

Thus, the number of ways we can partition the 6 parameters of the rate matrix equals: $\mathcal{B}(6) = 203$. Hence, the GTR family includes the 203 substitution models. Huelsenbeck, Larget and Alfaro (2004) defined and indexed all of the models possible under this parameterization. The list of models can be found in their table: http://mbe.oxfordjournals.org/content/21/6/1123/T1.expansion.html.

For this exercise, we will work in the MrBayes command line (batch files are also provided). Execute the **mb** binary in your Unix terminal:

- **> mb**

Save the screen output to file:

- **MrBayes > log start filename=conifer-rjmcmc-log.txt**

We will use the 9-taxon conifer dataset (Box 1). Load the sequences into MrBayes:

- **MrBayes > execute conifer_dna.nex**

And specify *Ginko biloba* as the outgroup.

- **MrBayes > outgroup Ginko_biloba**

Specify the mixed model

In MrBayes, model averaging over the family of GTR models is specified using the **lset** command. Read more details about this in the **help** documentation:

- **MrBayes > help lset**

Notice that the **nst** parameter can be set to **Mixed**. If you scroll up, this option is defined such that when you specify **lset nst=mixed**, this "*results in the Markov chain sampling over the space of all possible reversible substitution models, including the GTR model and all models that can be derived from it by grouping the six rates in various combinations. This includes all the named models above and a large number of others, with or without name.*" Thus when using reversible-jump MCMC, we are no longer restricted to choosing a single, named substitution model (F81, HKY85, GTR).

In the MrBayes software, the GTR family of models are given the parameter label **gtrsubmodel[xxxxxx]**, where the indices contained within the square braces represent the category assignment of each of the exchangeability rates. Under this notation, the different named models and many other models can be described. MrBayes uses rjMCMC to sample among the different models. This algorithm will average over all 203 models in the GTR family **in proportion to their marginal posterior probability**, including F81 (**gtrsubmodel[111111]**), HKY85 (**gtrsubmodel[121121]**), and GTR (**gtrsubmodel[123456]**).

Specify model averaging over GTR rates and gamma-distributed rate heterogeneity:

- **MrBayes > lset nst=mixed rates=gamma**

We will leave the remaining parameters and priors specified to the program default values. Evaluate your model specification:

- **MrBayes > showmodel**

Running under the prior

For **every** Bayesian analysis, it's critical to examine the various priors specified and identify induced priors that may result from interactions between parameters. This procedure is done by generating samples of the various parameters and hyperparameters under the prior, without accounting for the data. This is also often called "running on empty". Essentially, you use MCMC to simulate under the prior, and this allows you to inspect the marginal distributions of the parameters.

In MrBayes, running under the prior is specified in the **mcmc/mcmcp** command by the option **data=no**. When generating samples under the prior with MCMC, the only important concern is that you have a sufficient number. Therefore, it is not necessary to run multiple chains or multiple independent runs.

Use the **mcmcp** command to specify the details of the Markov chain.

- **MrBayes > mcmcp data=no nruns=1 nchains=1**

When you set **data=no**, the program simply disregards the sequence data by returning a constant (0.0) every time the likelihood function is called. Accordingly, running on empty takes very little time.

Set the number of generations equal to 2,000,000 using the **mcmcp** command.

- **MrBayes > mcmcp ngen=2000000 printfreq=100 samplefreq=100**

You are ready to run the analysis. Set the file name prefix to **conifer-rjmcmc-prior** when you execute the **mcmc** command:

- **MrBayes > mcmc filename=conifer-rjmcmc-prior**

Tracer (Rambaut and Drummond, 2009) is an excellent program for examining marginal prior and posterior densities for Bayesian phylogenetic analysis. Evaluating the parameter samples under the prior can often help to identify misspecified priors or errors in your analysis set-up. When examining prior densities, we are only concerned with the values reported to the parameter file (***.p**).

- Open the file called **conifer-rjmcmc-prior.p** in Tracer. You may have to open a new terminal window and execute the **tracer** binary:

  > tracer conifer-rjmcmc-prior.p

You will notice that the plots for the likelihood (**LnL**) are not informative. This is because, when running MCMC without data, the likelihood is 0. You can disregard this statistic since we are just interested in the marginal distributions of monitored numerical parameters. The goal when evaluating priors is to ensure that their shapes meet our expectations and that all of the model parameters are accounted for (often, if you accidentally specify the wrong model, you will discover it here–*e.g.* if you set **nst=2** instead of **nst=mixed**, you would have a different set of parameters).

- Look through each of the parameters, paying close attention to the shapes of the distributions in the ***Marginal Density*** pane.

- Inspect the marginal densities of the relative exchangeability rates. Under this prior, the rates are sampled from a mixture of distributions, thus these look unlike any obvious parametric density.

For this exercise, we are primarily interested in the variables relevant to the mixed model over GTR submodels. These include **gtrsubmodel** and **k_revmat**. The **k_revmat** statistic indicates the number of unique rate values in the GTR matrix.

- Select **k_revmat** in Tracer and observe the histogram of the prior distribution in the ***Estimates*** window.

- Refer back to **Table 1** of Huelsenbeck, Larget and Alfaro (2004) and Table 4 below, both show the distribution of the number of rates across the different GTR submodels. The prior mean number of rate categories is **k_revmat** = 3.320197. When using MCMC to sample without data, the **k_revmat** prior distribution in Tracer should (approximately) match the expected distribution.

Table 4: The distribution of the number of rate parameters across the 203 GTR submodels.

Number of rates	Number of models
k_revmat = 1	1
k_revmat = 2	31
k_revmat = 3	90
k_revmat = 4	65
k_revmat = 5	15
k_revmat = 6	1

The trace called **gtrsubmodel** gives the model in the GTR family sampled by MCMC. The value is determined by the indices in **Table 1** of Huelsenbeck, Larget and Alfaro (2004).

- Select **gtrsubmodel** in ***Traces*** and examine the prior distribution of this parameter in the ***Estimates*** window. The mixed model assumes that all of the 203 GTR submodels have equal probability. Therefore, the distribution of **gtrsubmodel** is aproximately uniform (Figure 6).

We have verified that the marginal prior densities of the relevant parameters match the expected densities. In addition, sampling under the prior provides a straightforward way to assess whether the data are informative for the numerous parameters and hyperparameters in our model. This is done by comparing the marginal *prior* densities to the marginal *posterior* densities (after running with data).

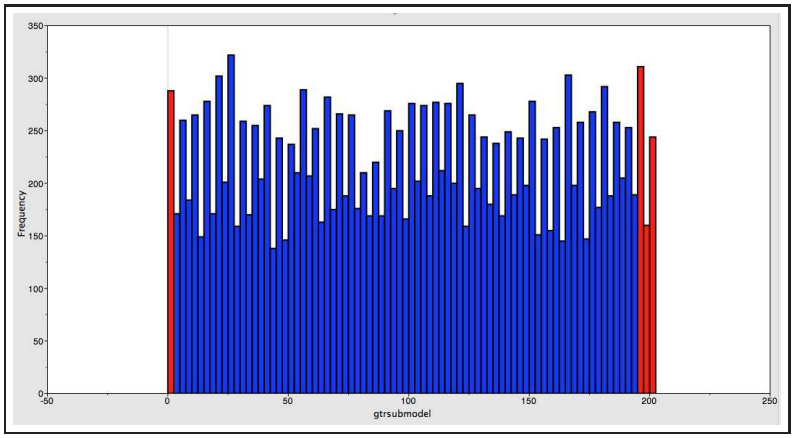

Figure 6: The prior distribution over all possible GTR submodels, simulated using MCMC.

Analyze the data

Since you have determined that the model was specified correctly, it's time to proceed with the full analysis. Return to MrBayes and run MCMC while accounting for the information in the data. For this analysis, perform 2 independent runs using Metropolis-coupled MCMC (MC^3) by setting **nruns=2** and **nchains=4**; and set **savebrlens=yes**.

- **MrBayes > mcmcp data=yes nruns=2 nchains=4 savebrlens=yes**

Needless to say, performing MCMC while accounting for the data is far more computationally intensive and takes a great deal longer than running on empty. Therefore, for the purposes of this exercise, specify fewer steps for the Markov chain. (This is simply for the sake of time, ideally we would run MCMC much longer.)

- **MrBayes > mcmcp ngen=100000 printfreq=100 samplefreq=100**

Set the diagnostics to **maxstddev** every 1,000 iterations.

- **MrBayes > mcmcp diagnfreq=1000 diagnstat=maxstddev**

Run the analysis with **mcmc** command and specify the output file name.

- **MrBayes > mcmc filename=conifer-rjmcmc**

Once the Markov chain is completed, summarize the MCMC samples of the scalar parameters using the **sump** command.

- **MrBayes > sump filename=conifer-rjmcmc**

The **sump** command will generate tables showing summary statistics of the model parameters.

- First, look at the table of model parameter summaries and find **k_revmat**. Notice that the mean is higher (**k_revmat** $\approx$ 4.67) than the mean value expected under the prior (**k_revmat** = 3.32). Furthermore, the 95% credible interval (also referred to as the 95% HPD interval) of **k_revmat** is [4,6] (this is the case for the provided output files). This range does not cover the expected number of rates under the prior (3.320197), indicating that there is a significant amount of information in the data for the number of unique values in the rate matrix.

- The next table generated by the **sump** command shows the different GTR submodels with posterior probability over 0.05. This table is reproduced for you (from the provided output files) in Table 5. For these data, the analysis shows that only 6 of the 203 GTR submodels have posterior probability over 0.05. Furthermore, no single model stands out as the "best" model with a significantly high probability. Thus, model averaging for these data provides a way to estimate under multiple GTR-model configurations.

Table 5: Models in the GTR family with posterior probability over 0.05.

Model	Posterior Probability	Standard Deviation	Min. Probability	Max. Probability
gtrsubmodel[123345]	0.326	0.009	0.320	0.333
gtrsubmodel[123343]	0.208	0.016	0.197	0.220
gtrsubmodel[123341]	0.163	0.005	0.160	0.166
gtrsubmodel[123454]	0.119	0.004	0.116	0.121
gtrsubmodel[123456]	0.067	0.009	0.060	0.073
gtrsubmodel[123453]	0.061	0.006	0.057	0.065

We can get a more detailed view of these features of our data when we evaluate the marginal densities of the **gtrsubmodel** and **k_revmat** in Tracer (open a new terminal window, do not quit MrBayes). Open all of the ***.p** files from this exercise in Tracer.

- **> tracer conifer-rjmcmc-prior.p conifer-rjmcmc.run1.p conifer-rjmcmc.run2.p**

- Select the file for **run1** in the ***Trace Files*** pane of the Tracer window. Evaluate the ***Marginal Density*** of **k_revmat** for each of the traces and each of your 3 ***.p** files. To select multiple files, hold down the control key while clicking on each file name. (Do not select the ***Combined*** file. This combines the samples for all of the trace files loaded into Tracer, including the samples from the prior. The combined trace file is intended only when the loaded files are from independent runs on the same dataset under the same model and prior parameterization. If **conifer-rjmcmc-prior.p** was not loaded *and* if the two independent runs (**run1** and **run2**) sampled the same stationary distribution, then it is appropriate to evaluate the ***Combined*** trace.)

- Color the marginal densities of each run by selecting ***Colour by: Trace File*** in the pull-down menu at the botom of the window. Also, choose ***Legend: Top-Left*** to place a key in the graph. Here, you are comparing the marginal densities of the **k_revmat** parameter for 2 independent runs and the prior.

- Take note of the fact that the 2 posterior marginal densities are almost completely overlapping, this indicates that the two runs have both converged on the same stationary distribution.

- Furthermore, the marginal posterior densities are distinctly different from the marginal prior distribution, which is uniform over all 203 models. The posterior samples show that the data support very few of the GTR-submodels. This remains the case, even when the Markov chains are run for 5,000,000 generations. This is shown in Figure 7.

Return to MrBayes and summarize the tree topology and branch lengths:

- **MrBayes > sumt filename=conifer-rjmcmc**

And quit the program:

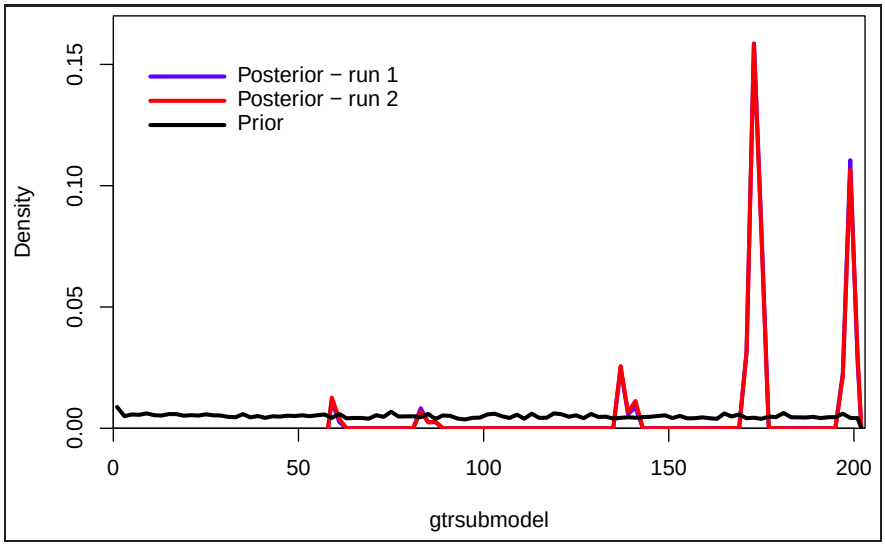

Figure 7: The marginal posterior and prior distributions of the GTR submodel. The posterior densities of the submodel are colored in red and blue. Because the two runs (**ngen=5000000**) have converged on the same stationary distribution, they are almost completely overlapping (*i.e.*, the blue line is barely visible).

- **MrBayes > quit**

Open the summary tree in FigTree and make a figure that shows the bipartition posterior probabilities (Figure 8).

- **figtree conifer-rjmcmc.con.tre**

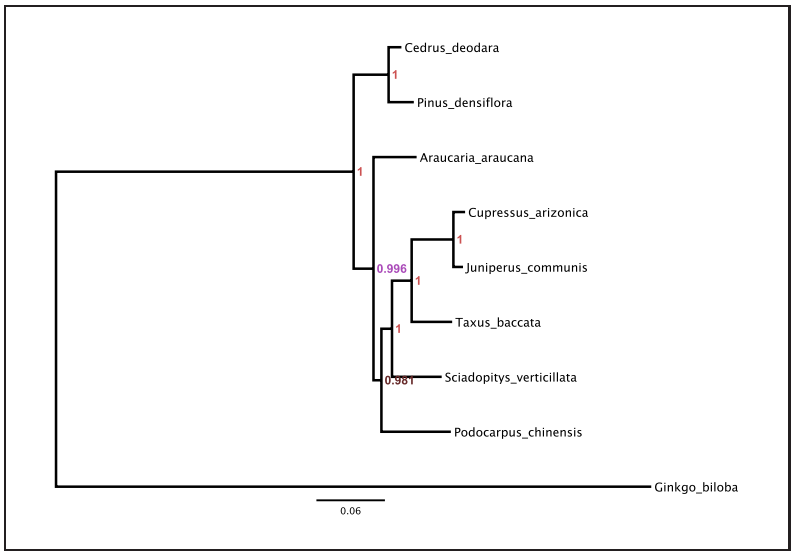

Figure 8: The summary tree from the rjMCMC analysis, with posterior probabilities labeled at nodes.

Exercise 2 – Batch Mode

All of the MrBayes commands are in the batch file: **conifer_rjMCMC.nex**

Useful Links

- MrBayes: http://mrbayes.sourceforge.net/
- PhyloBayes: www.phylobayes.org/
- BEAGLE: http://code.google.com/p/beagle-lib/
- Tree Thinkers: http://treethinkers.org/

Questions about this tutorial can be directed to:

- Tracy Heath (email: tracyh@berkeley.edu)
- Brian Moore (email: brianmoore@ucdavis.edu)
- Conor Meehan (email: conor.meehan@dal.ca)

Relevant References

Alfaro ME, Huelsenbeck JP. 2006. Comparative performance of Bayesian and AIC-based measures of phylogenetic model uncertainty. Systematic Biology. 55:89–96.

Antoniak CE. 1974. Mixtures of Dirichlet processes with applications to non-parametric problems. Annals of Statistics. 2:1152–1174.

Bell ET. 1934. Exponential numbers. American Mathematics Monthly. 41:411–419.

Brandley MC, Schmitz A, Reeder TW. 2005. Partitioned Bayesian analyses, partition choice, and the phylogenetic relationships of scincid lizards. Systematic Biology. 54:373–390.

Brown JM, Lemmon AR. 2007. The importance of data partitioning and the utility of Bayes factors in Bayesian phylogenetics. Systematic Biology. 56:643–655.

Fan Y, Wu R, Chen MH, Kuo L, Lewis PO. 2011. Choosing among partition models in Bayesian phylogenetics. Molecular Biology and Evolution. 28:523–532.

Ferguson TS. 1973. A Bayesian analysis of some nonparametric problems. Annals of Statistics. 1:209–230.

Geyer CJ. 1991. Markov chain Monte Carlo maximum likelihood. In: Kerimidas EM, editor, Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface. Fairfax Station: Interface Foundation, pp. 156–163.

Hastings WK. 1970. Monte carlo sampling methods using markov chains and their applications. Biometrika. 57:97–109.

Huelsenbeck JP, Larget B, Alfaro ME. 2004. Bayesian phylogenetic model selection using reversible jump Markov chain Monte Carlo. Molecular Biology and Evolution. 21:1123–1133.

Huelsenbeck JP, Ronquist F. 2001. MRBAYES: Bayesian inference of phylogenetic trees. Bioinformatics. 17:754–755.

Huelsenbeck JP, Ronquist F. 2005. Bayesian analysis of molecular evolution using MrBayes. In: Nielsen R, editor, Statistical Methods in Molecular Evolution. Springer-Verlag, pp. 183–232.

Huelsenbeck JP, Suchard M. 2007. A nonparametric method for accommodating and testing across-site rate variation. Systematic Biology. 56:975–987.

Jeffreys H. 1961. Theory of Probability. Oxford: Oxford University Press.

Kass RE, Raftery AE. 1995. Bayes factors. Journal of the American Statistical Association. 90:773–795.

Lartillot N, Lepage T, Blanquart S. 2009. Phylobayes 3: a bayesian software package for phylogenetic reconstruction and molecular dating. Bioinformatics. 25:2286.

Lartillot N, Philippe H. 2004. A Bayesian mixture model for across-site heterogeneities in the amino-acid replacement process. Molecular Biology and Evolution. 21:1095–1109.

Lartillot N, Philippe H. 2006. Computing Bayes factors using theromodynamic integration. Systematic Biology. 55:195–207.

Lavine M, Schervish MJ. 1999. Bayes factors: what they are and what they are not. The American Statistician. 53:119–122.

Liu L, Pearl DK. 2007. Species trees from gene trees: Reconstructing bayesian posterior distributions of a species phylogeny using estimated gene tree distributions. Systematic Biology. 56:504–514.

McGuire JA, Witt CC, Altshuler DL, Remsen JV. 2007. Phylogenetic systematics and biogeography of hummingbirds: Bayesian and maximum likelihood analyses of partitioned data and selection of an appropriate partitioning strategy. Systematic Biology. 56:837–856.

Metropolis N, Rosenbluth AW, Rosenbluth MN, Teller AH, Teller E. 1953. Equation of state calculations by fast computing machines. Journal of Chemical Physics. 21:1087–1092.

Nylander J, Wilgenbusch J, Warren D, Swofford D. 2008. Awty (are we there yet?): a system for graphical exploration of mcmc convergence in bayesian phylogenetics. Bioinformatics. 24:581.

Nylander JAA, Ronquist F, Huelsenbeck JP, Aldrey JLN. 2004. Bayesian phylogenetic analysis of combined data. Systematic Biology. 53:47–67.

Rambaut A, Drummond AJ. 2009. Tracer v1.5. Edinburgh (United Kingdom): Institute of Evolutionary Biology, University of Edinburgh. Available from: http://beast.bio.ed.ac.uk/Tracer.

Rannala B, Yang Z. 1996. Probability distribution of molecular evolutionary trees: A new method of phylogenetic inference. Journal of Molecular Evolution. 43:304–311.

Robert CP, Casella G. 2002. Monte Carlo Statistical Methods. New York: Springer.

Rodrigue N, Philippe H, Lartillot N. 2008. Uniformization for sampling realizations of Markov processes: Applications to Bayesian implementations of codon substitution models. Bioinformatics. 24:56–62.

Ronquist F, Huelsenbeck JP. 2003. Mrbayes 3: Bayesian phylogenetic inference under mixed models. Bioinformatics. 19:1572–1574.

Ronquist F, Teslenko M, van der Mark P, Ayres DL, Darling A, Höhna S, Larget B, Liu L, Suchard MA, Huelsenbeck JP. 2012. Mrbayes 3.2: Efficient bayesian phylogenetic inference and model choice across a large model space. Systematic Biology. 61:539–542.

Ronquist F, van der Mark P, Huelsenbeck JP. 2009. Bayesian analysis of molecular evolution using MrBayes. In: Lemey P, Salemi M, Vandamme AM, editors, The Phylogenetic Handbook: A Practical Approach to Phylogenetic Analysis and Hypothesis Testing, Second Edition. Cambridge University Press, pp. 1–1.

Rubinstein R. 1981. Simulation and the Monte Carlo Method. John Wiley & Sons, Inc. New York, NY, USA.

Simon D, Larget B. 2001. Bayesian analysis in molecular biology and evolution (BAMBE). http://www.mathcs.duq.edu/larget/bambe.html.

Smith A, Roberts G. 1993. Bayesian computation via the Gibbs sampler and related Markov chain Monte Carlo methods. Journal of the Royal Statistical Society. Series B (Methodological). 55:3–23.

Suchard M, Weiss R, Sinsheimer J. 2005. Models for estimating bayes factors with applications to phylogeny and tests of monophyly. Biometrics. 61:665–673.

Suchard MA, Weiss RE, Sinsheimer JS. 2001. Bayesian selection of continuous-time Markov chain evolutionary models. Molecular Biology and Evolution. 18:1001–1013.

Sukumaran J, Holder MT. 2010. DendroPy: A Python library for phylogenetic computing. Bioinformatics. 26:1569–1571.

Verdinelli I, Wasserman L. 1995. Computing Bayes factors using a generalization of the SavageÐDickey density ratio. Journal of the American Statistical Association. 90:614–618.

Wong KM, Suchard MA, Huelsenbeck JP. 2008. Alignment uncertainty and genomic analysis. Science. 319:473–476.

Wu CH, Suchard MA, Drummond AJ. 2013. Bayesian selection of nucleotide substitution models and their site assignments. Molecular Biology and Evolution. 30:669–688.

Xie W, Lewis PO, Fan Y, Kuo L, Chen MH. 2011. Improving marginal likelihood estimation for Bayesian phylogenetic model selection. Systematic Biology. 60:150–160.

Yang Z. 1994. Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: Approximate methods. Journal of Molecular Evolution. 39:306–314.

Yang Z. 2007. Fair-balance paradox, star-tree paradox, and Bayesian phylogenetics. Molecular biology and evolution. 24:1639.

Yang Z, Rannala B. 1997. Bayesian phylogenetic inference using DNA sequences: a Markov chain Monte Carlo method. Molecular Biology and Evolution. 14:717–724.

Yang Z, Rannala B. 2005. Branch-length prior influences Bayesian posterior probability of phylogeny. Systematic Biology. 54:455–470.

Zwickl DJ, Holder MT. 2004. Model parameterization, prior distributions, and the general time-reversible model in Bayesian phylogenetics. Systematic Biology. 53:877–888.