biphy manual

Walker Pett

May 2, 2016

1 Introduction

`biphy` is a tool optimized for Bayesian phylogenetic analysis of binary character data. Several models are available, as well as functions for doing posterior predictive simulations and cross-validation checks.

1.1 Markovian Substitution Process

`biphy` assumes that the gain and loss of binary characters occurs following a continuous time Poisson process along each branch, where the rates of gain and loss on a branch of length t are indicated by λ and μ respectively. The stochastic kernel of this substitution process, denoted $\mathbf{Q}$, can be expressed as follows:

$$\mathbf{Q} = \begin{bmatrix} -\lambda & \lambda \\ \mu & -\mu \end{bmatrix}$$

where the stationary probability of a character being present is $\pi = \frac{\lambda}{\lambda+\mu}$. When not constrained to ultrametric trees, it is natural to scale the branch length t so that it is expressed as the expected number of substitutions. This is acheived by scaling $\mathbf{Q}$ by the expected rate of character replacement $\rho = \lambda(1-\pi)+\mu\pi$ to give the scaled matrix

$$\mathbf{R} = \mathbf{Q}/\rho = \begin{bmatrix} -\pi & \pi \\ (1-\pi) & -(1-\pi) \end{bmatrix} /2\pi(1-\pi)$$

A new process defined using $\mathbf{R}$ can thus be parameterized using only π, and results in branch lengths being expressed as the expected number of substitutions. This is the parameterization that `biphy` uses.

Note that we may allow the substitution matrix to differ among branches, and therefore the entire substitution process along the tree is not time-reversible. As a consequence, the placement of the root is not trivial. Furthermore, because the process is not stationary, we specify an independent parameter ϕ representing the probability of a character being present in the ancestral sequence at the root.

2 Basic usage

2.1 Input Data Format

`biphy` accepts Phylip, Fasta, Pasta, or Nexus format data files with `datatype=restriction` or `datatype=standard`

2.2 Starting a run

A run is initiated by providing the `biphy` command with a data file and a run name

```
biphy [options...] -d <data file> <run name>
```

2.3 Summarizing the output

`biphy` produces a trace file `<run name>.trace` and a treelist file `<run name>.treelist` if the tree topology was not fixed. These files are suitable as inputs respectively to the `tracecomp` and `bpcomp` programs in the PhyloBayes package [3].

2.4 Restarting a run

In the event of a power failure or other error, a `biphy` run can be restarted using the run name with no other options

```
biphy <run name>
```

2.5 Stream-reading mode

Stream-reading mode is activated when a run name is specified with some options, but a data file is not specified via the `-d` option. This mode is used only for performing posterior predictive simulations, cross-validation checks, or simulating ancestral states via the `-ppred`, `-cv`, and `-a` options

```
biphy [-ppred -cv <test file> -a] <run name>
```

3 Detailed options

3.1 General options

```
-d <data file>
```

Option for specifying the name of the file containing a character matrix to be analyzed

```
-x <every> [<until>]
```

Specifies the saving frequency and (optional) the number of points after which the chain should stop. If this number is not specified, the chain runs forever. By definition, `-x 1000` corresponds to the default saving frequency.

```
-t <tree file>
```

Specifies a Newick format tree file containing a tree which will be used to fix the tree topology during the analysis

```
-o <outgroup file>
```

Specifies a file containing one line of tab-separated taxon names which will be constrained as a monophyletic outgroup during the analysis

```
-f
```

Forces the program to overwrite an already existing run with same name.

3.2 Model options

`-h`

Default time-homogeneous asymmetric reversible model. A single stationary frequency $\pi = \phi$ is assumed across the entire tree.

`-nh`

Time-heterogeneous asymmetric reversible model. The root frequency ϕ and branch-wise stationary frequencies π_j are drawn iid from a beta distribution with hyperparameters β_1 and β_2, which are both iid from an exponential distribution of mean 1.

`-dollo`

The irreversible stochastic Dollo model of Nicholls and Gray [5]. In this case, $\pi = 0$ and $\phi = 1$, and the likelihood is computed by integrating over all possible origination points for each character with respect to a homogeneous origination rate λ with prior density $f(\lambda) \propto \lambda^{-1}$, which is integrated out of the likelihood computations. Under this model, branch lengths are expressed in units of expected numbers of losses.

`-mk`

The symmetric reversible Mk model of Lewis [4]. In this case, $\pi = \phi = 1/2$.

3.3 Additional priors

`-dgam <int>`

This option specifies that site-specific rates should be drawn from a Gamma(α, α) distribution, discretized on `<int>` quantiles, where the prior for α is an exponential distribution of mean 1. The options `-dgam 0` or `-dgam 1` specify a constant rates model. The default is `-dgam 4`

`-dbeta <int>`

This option specifies that site-specific stationary frequencies should be drawn from a Beta(β, β) distribution, discretized on `<int>` quantiles, where the prior for β is an exponential distribution of mean 1. This option is only used with the Mk model. The options `-dbeta 0` or `-dbeta 1` specify the standard Mk model.

`-asymmbeta`

Specifies an asymmetric Beta(β_1, β_2) distribution for site-specific stationary frequencies under the Mk model, where β_1 and β_2 are both iid from an exponential distribution of mean 1.

`-lexp`

Hierarchical exponential prior on branch lengths. Branch lengths are drawn from an exponential distribution of mean μ, which is itself drawn from an exponential distribution of mean 1/10.

`-lfixed`

Branch lengths are fixed at the values in the input tree provided with the `-t` option.

`-lstrict`

Branch lengths are fixed at the values in the input tree provided with the `-t` option, and a homogeneous clock rate μ is drawn from an exponential distribution of mean 1.

`-rp <min> <max>`

The root frequency ϕ is truncated on the (unnormalized) interval `(min,max)`. Applies only to the time homogeneous model, and the hierarchical beta model.

`-rigid`

The root frequency ϕ and the stationary frequencies on the branches subtending the root node of the tree are constrained to be equal. This can improve mixing due to the poor identifiability of these parameters in the unconstrained model [1]. Applies only to time-heterogeneous models.

3.4 Correction for unobservable site-patterns

`-u <int>`

Use this option to specify site-patterns that cannot be observed or which have been omitted from the dataset. This results in a correction to the likelihood that accounts for the absence of these patterns [2].
The option `<int>` is one of:

0 = no correction, all site-patterns are observable (default)
1 = characters absent in all species cannot be observed
2 = characters present in all species cannot be observed
4 = characters present in a single species cannot be observed
8 = characters absent in a single species cannot be observed

Combinations of the above site-patterns are achieved by adding the indicated values. For example, `-u 3` indicates that constant sites have been omitted, and `-u 15` indicates that parsimony uninformative sites have been omitted.

3.5 MCMCMC options

`-n <int>`

This option specifies the number of Metropolis-coupled chains to use, with one chain per available thread. The default is 1 chain.

`-delta <float>`

This options specifies the δ tuning parameters for the calculation of chain heats during a parallel Metropolis-coupled run. The heat of chain k is computed as follows

$$\beta_k = \frac{1}{1+\delta k}$$

The default value is $\delta = 0.05$. This should be appropriate for small values of n (less than 10), but should be adjusted on a case by case basis to improve mixing.

`-si <int>`

This option specifies the number of generations between proposals to swap Metropolis-coupled chains. The default value is 1.

3.6 Output options

`-e`

In addition to the Newick treelist output, a Nexus format tree file with additional branch parameters is produced. This is useful for mapping branch-wise stationary frequencies to branches in tree visualization software.

`-s`

By default, the entire output of all chains is saved to the file `<run name>.stream`, but this behavior can be disabled using this option. This output is required if later posterior predictive simulations or cross-validation checks are desired. Note that the saving frequency of this stream is the single-chain saving frequency specified by the `-x` option, multiplied by the swap interval of the Metropolis-coupled chain specified by the `-si` option.

3.7 Stream-reading options

The following options are to be used in stream-reading mode, which is activated by running `biphy` without specifying a data file via the `-d` option. In stream-reading mode, the parameter configuration of each sample is read sequentially from the stream of the indicated run, and computations are performed by conditioning on this parameter configuration

`-ppred <int>`

An alignment is simulated at each sample point, and one of two statistics is computed based on that alignment.

`<int>` $= 0$ the proportion of characters present in each species is saved to the file `<run name>.ppred0`
`<int>` $= 1$ the number of characters per number of species present is saved to the file `<run name>.ppred1`

`-cv <test file>`

The likelihood of the specified test alignment is calculated at each sample point, and saved to the file `<run name>.cv`

`-a`

Ancestral states for each node are reconstructed at each sample point and saved to the file `<run name>.mapping`

3.8 Marginal likelihood estimation

The marginal likelihood is useful for comparing the relative fit of different substitution models, and can be accurately estimated using the steppingstone method [6]. The marginal likelihood is defined as the the

likelihood of the data integrated over the joint prior distribution of all model parameters. Specifically, for data D and model parameters θ, the marginal likelihood is defined as follows:

$$f(D) = \int_{\Theta} f(D \mid \theta) f(\theta) d\theta$$

The marginal likelihood can be computed by restarting an already existing `biphy` run in steppingstone mode, as follows

```
biphy -ss <int> <run name>
```

This option initializes $K =$ `<int>` chains with the last sampled state of the indicated run. Together, these chains form a steppingstone sampler used to estimate the marginal likelihood of the model. The target distribution of chain $k \in 0..K$ is $f(\theta \mid D)^{\beta_k}$, where $\beta_k = (k/K)^{0.3}$. Then, at each generation i of the steppingstone sampler, the log steppingstone factor $\hat{ss}_{k,i}$ contributed by chain k to the total marginal likelihood estimate is saved to the file `<run name>.ss`. Each $\hat{ss}_{k,i}$ is computed as follows:

$$\hat{ss}_{k,i} = (\beta_k - \beta_{k-1}) \ln f(D \mid \theta_{k-1,i})$$

where $\theta_{k-1,i}$ is the parameter state of chain $k-1$ at generation i. When the $\hat{ss}_{k,i}$ estimates converge, the final estimate of the marginal log likelihood is obtained as the product of the mean exponential of the sampled $\hat{f}_{k,i}$ values

$$\ln f(D) \approx \sum_{k=1}^{K} \ln \left(\frac{1}{N} \sum_{i=b}^{N} \exp(\hat{ss}_{k,i}) \right)$$

where b is the burn-in and N is the total number of samples. At each generation, an estimate using $b = N/5$ is saved to the same output file as the $\hat{ss}_{k,i}$ values. The steppingstone sampler works most efficiently if it is initialized with a sample from the posterior distribution $f(\theta \mid D)$. For this reason, it is recommended that you first perform a normal `biphy` run to obtain a sample from the posterior, and then initialize a steppingstone sampler from this run.

Finally, the variance of the marginal likelihood estimate decreases as the number of chains is increased in the steppingstone sampler. Thus, the optimal number of chains K is determined by a trade-off between increased computational requirements and increased precision.

References

[1] Joseph T Chang. Full Reconstruction of Markov Models on Evolutionary Trees : Identifiability and Consistency. *Mathematical Biosciences*, 13:51–73, 1996.

[2] Joseph Felsenstein. Phylogenies from restriction sites: A maximum-likelihood approach. *Evolution*, 46(1):159–173, 1992.

[3] Nicolas Lartillot, Nicolas Rodrigue, Daniel Stubbs, and Jacque Richer. PhyloBayes MPI : Phylogenetic Reconstruction with Infinite Mixtures of Profiles in a Parallel Environment. *Systematic Biology*, 62(4):611–615, 2013.

[4] PO Lewis. A likelihood approach to estimating phylogeny from discrete morphological character data. *Systematic Biology*, 50(6):913–925, 2001.

[5] Geoff K. Nicholls and Russell D. Gray. Dated ancestral trees from binary trait data and their application to the diversification of languages. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(3):545–566, jul 2008.

[6] Wangang Xie, Paul O Lewis, Yu Fan, Lynn Kuo, and Ming-Hui Chen. Improving marginal likelihood estimation for Bayesian phylogenetic model selection. *Systematic biology*, 60(2):150–60, mar 2011.