

Table of contents

- Overview
- Getting started
- Exercise
 - 1: Tree inference:
 - ◆ 1.1: Basic analysis
 - ◆ 1.2: Check convergence
 - ◆ 1.3: Alternative models
 - ◆ 1.4: PhyloBayes-MPI
 - ◆ 1.5: ML analysis using mixture models in IQ-TREE
 - 2: Model comparison (optional):
 - ◆ 2.1: Cross-validation
 - ◆ 2.2: Posterior predictive test

Overview

The objectivity of this exercise is to help you understand how to perform Bayesian phylogenetic inferences using PhyloBayes. You will learn how to set models, check for convergence, and perform model comparison in PhyloBayes. You will also learn how to run Maximum-likelihood analysis using mixture models in IQ-TREE.

Getting started

In this exercise, we will use PhyloBayes to analyze ten small protein sequence alignments simulated under LG+C60+F model using 4-taxon trees with various levels of long-branch attraction artefacts (Wang et al., 2017). The alignment `simuLBA.C60.7.5.phy` will be used as example throughout this tutorial, but you can use any of the 10 alignments.

The original alignments have >20,000 sites, which will be too long for this practical session. Therefore, we will first create a smaller alignment of 1,000 sites:

```
goalign sample sites -l 1000 -p -i simuLBA.C60.7.5.phy -o simuLBA.C60.7.5.1k.phy
```

Exercise 1: Tree inference

1.1: Basic analysis

To conduct a basic tree inference in PhyloBayes, run:

```
pb -s -cat -gtr -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.CAT_GTR.chain0 &
```

The “-d” option allows you to specify the data file. The “-s” option instructs the program to save all parameters sampled in the MCMC run in addition to tree topologies, so that we can later perform model comparison analyses. The “-cat” option turns on the CAT mixture model for equilibrium frequency profiles. The “-gtr” option sets the exchange rate matrix to GTR (all exchangeabilities to be estimated). The name of the run (“simuLBA.C60.7.5.1k.CAT_GTR.chain0”) is and is specified as the last parameter. The “&” at the end of command put the process into background so you can continue operating in the same terminal.

To start another analysis under the same model, just change the name of the run:

```
pb -s -cat -poisson -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.CAT_GTR.chain1 &
```

During a PhyloBayes run, summary information (e.g., cycle number, log-likelihood score, no. of categories) will be written continuously to the “RUN_NAME.trace” file (e.g., simuLBA.C60.7.5.1k.CAT_GTR.chain0.trace), whereas the trees will be written continuously to the “RUN_NAME.treelist” file (e.g., simuLBA.C60.7.5.1k.CAT_GTR.chain0.treelist). You can monitor the progress of the run by regularly checking the “RUN_NAME.trace” file. For example, you can find out how

many cycles the run “simuLBA.C60.7.5.1k.CAT_GTR.chain0” has reached by showing the last line of the “simuLBA.C60.7.5.1k.CAT_GTR.chain0.trace” file (the first number in the line is the current cycle number):

```
tail -n 1 simuLBA.C60.7.5.1k.CAT_GTR.chain0.trace
```

1.2: Check convergence

We can check if the two runs have converged by using the *tracecomp* and *bpcomp* programs provided by PhyloBayes:

```
tracecomp -x 500 1 -o CAT_GTR simuLBA.C60.7.5.1k.CAT_GTR.chain0  
simuLBA.C60.7.5.1k.CAT_GTR.chain1
```

The *tracecomp* program will compare the two runs for a number of summary statistics and report the discrepancies and effective sizes in the “CAT_GTR.contdiff” file (filename specified using the “-o” option). The “-x” option specifies the length of burn-in and the sampling frequency. In this case, the first 500 cycles will be discarded as burn-in and after that, every cycle will be sampled. Max. difference < 0.1 and effective size > 300 is considered good. Max. difference < 0.3 and effective size > 50 is considered acceptable.

```
bpcomp -x 500 1 -o CAT_GTR simuLBA.C60.7.5.1k.CAT_GTR.chain0  
simuLBA.C60.7.5.1k.CAT_GTR.chain1
```

The *bpcomp* program will compare the two runs for tree topologies, **generate a posterior consensus tree** in the “CAT_GTR.con.tre” file, and report the max. and mean differences in bipartition frequencies in the “CAT_GTR.bpdiff” file. Max. difference < 0.1 is considered good and max. difference < 0.3 is be considered acceptable.

You can also download the trace files to your computer to check convergence using Tracer.

Once a PhyloBayes analysis has converged, you can stop it by using the *stoppb* program. For example, to stop the two analyses above:

```
stoppb simuLBA.C60.7.5.1k.CAT_GTR.chain0
```

```
stoppb simuLBA.C60.7.5.1k.CAT_GTR.chain1
```

Stopped analyses can also be resumed by using the name of the run as the single input for the *pb* program:

```
pb simuLBA.C60.7.5.1k.CAT_GTR.chain0 &
```

```
pb simuLBA.C60.7.5.1k.CAT_GTR.chain1 &
```

1.3: Alternative models

There is a number of models that you can try:

Equilibrium frequency	Exchange rate matrix	PhyloBayes option	Command examples
CAT	GTR	-cat -gtr	<pre>pb -s -cat -gtr -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.CAT_GTR.chain0</pre>
CAT	LG	-cat -lg	<pre>pb -s -cat -lg -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.CAT_LG.chain0</pre>
CAT	Poisson	-cat -poisson	<pre>pb -s -cat -poisson -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.CAT_F81.chain0</pre>
homogenous	GTR	-ncat 1 -gtr	<pre>pb -s -ncat 1 -gtr -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.GTR.chain0</pre>
homogenous	LG	-ncat 1 -lg	<pre>pb -s -ncat 1 -lg -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.LG.chain0</pre>
homogenous	Poisson	-ncat -poisson	<pre>pb -s -ncat 1 -poisson -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.F81.chain0</pre>
C60	GTR	-catfix c60 -gtr	<pre>pb -s -catfix c60 -gtr -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.C60_GTR.chain0</pre>
C10	GTR	-catfix c10 -gtr	<pre>pb -s -catfix c10 -gtr -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.C10_GTR.chain0</pre>
C60	Poisson	-catfix c60	<pre>pb -s -catfix c60 -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.C60.chain0</pre>
C10	Poisson	-catfix c10	<pre>pb -s -catfix c10 -d simuLBA.C60.7.5.1k.phy simuLBA.C60.7.5.1k.C10.chain0</pre>

To run a PhyloBayes analysis with one of these models, just replace the “-cat -gtr” in the command we used in 1.1 with one of the options listed in the third column in the table above, and remember to change the name of the run (see command examples in the last column of the table).

Visualize and compare the trees inferred under different models using phylo.io. Are they the same?

1.4: PhyloBayes-MPI

PhyloBayes analyses can be really slow on empirical phylogenomic data sets, particularly when the CAT+GTR model is being used. To accelerate the analyses, we can use PhyloBayes-MPI to parallelize the computation:

```
mpirun -np 2 pb_mpi -cat -gtr -d simuLBA.C60.7.5.1k.phy  
simuLBA.C60.7.5.1k.CAT_GTR.mpi.chain0
```

Here, pb_mpi is used instead of pb, and we asked for 2 threads with “mpirun -np 2”. Since we can only use two threads at most, you probably will not observe much improvements in the runtime.

1.5: ML analysis using mixture models in IQ-TREE

Recently, several ML alternatives of the PhyloBayes-CAT model have been implemented in IQ-TREE. Here is how we can run the ML analysis using mixture models:

```
iqtree -nt 2 -m LG+C60+F -bb 1000 -bnni -s simuLBA.C60.7.5.1k.phy -pre  
simuLBA.C60.7.5.1k.LG+C60+F
```

The LG+C60+F model was the model used to simulate these alignments. You can also use other model configurations, such as LG+C60, C60+F, and C60 (do you know the

difference between them?).

You can also do a tree search using the homogenous model:

```
iqtree -nt 2 -m LG+G+F -bb 1000 -bnni -s simuLBA.C60.7.5.1k.phy -pre  
simuLBA.C60.7.5.1k.LG+G+F
```

Again, you can compare the ML trees inferred under different models, and also the trees inferred by PhyloBayes in previous steps.

Exercise 2: Model comparison

2.1 Cross-Validation

To run cross-validation, we first have to generate pairs of learning and test data sets:

```
cvrep -nrep 1 -nfold 10 -d simuLBA.C60.7.5.1k.phy cvb
```

Cross-Validation analyses typically consist of 10 replicates, but due to the time limits we will do just one replicate (-nrep 1). The “-nfold 10” option indicates that 90% and 10% of the original data set will be assigned to the learning and test data set, respectively. The name of the run is “cvb”.

Then we will run phylobayes analyses on the learning data set under different models using a fixed tree topology. Here we will use the “CAT_GTR.con.tre” generated in exercise 1.2 as the fixed tree, and compare the following model configurations:

CAT+GTR:

```
pb -T CAT_GTR.con.tre -x 1 1100 -cat -gtr -d cvb0_learn.ali CAT_GTRcvb0_learn.ali
```

CAT+POISSON:

```
pb -T CAT_GTR.con.tre -x 1 1100 -cat -poisson -d cvb0_learn.ali CAT_F81cvb0_learn.ali
```

GTR:

```
pb -T CAT_GTR.con.tre -x 1 1100 -ncat 1 -gtr -d cvb0_learn.ali GTRcvb0_learn.ali
```

POISSON:

```
pb -T CAT_GTR.con.tre -x 1 1100 -ncat 1 -lg -d cvb0_learn.ali F81cvb0_learn.ali
```

C60:

```
pb -T CAT_GTR.con.tre -x 1 1100 -catfix c60 -d cvb0_learn.ali C60cvb0_learn.ali
```

C10:

```
pb -T CAT_GTR.con.tre -x 1 1100 -catfix c10 -d cvb0_learn.ali C10cvb0_learn.ali
```

The “-x 1 1100” indicate that we would like to run 1100 cycles and save parameters per cycle. When the runs are done, we can calculate likelihood values for the test data set under different model configurations:

CAT+GTR:

```
readcv -rep 0 -x 100 1 CAT_GTR cvb
```

CAT+POISSON:

```
readcv -rep 0 -x 100 1 CAT_F81 cvb
```

GTR:

```
readcv -rep 0 -x 100 1 GTR cvb
```

F81:

```
readcv -rep 0 -x 100 1 F81 cvb
```

C60:

```
readcv -rep 0 -x 100 1 C60 cvb
```

C10:

```
readcv -rep 0 -x 100 1 C10 cvb
```

Here, the first 100 cycles will be discarded as burn-in (“-x 100 1”).

Lastly, we can compare between models:

```
sumcv -nrep 1 F81 GTR C10 C60 CAT_F81 CAT_GTR cvb
```

Note that the left most model will be used as the reference in the comparison.

2.2: Posterior predictive analysis

The PPA can be done using the ppred program provided by PhyloBayes:

CAT+GTR:

```
ppred -x 500 1 1500 -div -comp -sat simuLBA.C60.7.5.1k.CAT_GTR.chain0
```

CAT+POISSON:

```
ppred -x 500 1 1500 -div -comp -sat simuLBA.C60.7.5.1k.CAT_F81.chain0
```

GTR:

```
ppred -x 500 1 1500 -div -comp -sat simuLBA.C60.7.5.1k.GTR.chain0
```

POISSON:

```
ppred -x 500 1 1500 -div -comp -sat simuLBA.C60.7.5.1k.F81.chain0
```

C60:

```
ppred -x 500 1 1500 -div -comp -sat simuLBA.C60.7.5.1k.C60.chain0
```

C10:

```
ppred -x 500 1 1500 -div -comp -sat simuLBA.C60.7.5.1k.C10.chain0
```

The options “-div”, “-comp”, and “-sat” tell the program to compute statistics for the following features: “biochemical specificity”, “compositional homogeneity”, and “saturation”, respectively. Check the p-value for each test, and identify cases where the simulated data (i.e., data generated by *ppred* for the PPA) and the original data (i.e., *simuLBA.C60.7.5.1k.phy*) are significantly different (p-value < 0.05).