Substitution-model-averaging (SISMID version)

David A. Rasmussen, Carsten Magnus and Remco Bouckaert (Adapted by Nicola Müller for SISMID)

1 Background

Before running any phylogenetic analysis in BEAST, we need to decide on a model of molecular evolution that describes how our sequence data evolved. In particular, we need to decide on a substitution model that describes the relative rates at which different types of substitutions occur. For nucleotide data, the substitution model is typically represented as a 4x4 symmetric rate matrix Q with the general form:

$$Q = \begin{pmatrix} - & r_{ac} & r_{ag} & r_{at} \\ r_{ca} & - & r_{cg} & r_{ct} \\ r_{ga} & r_{gc} & - & r_{gt} \\ r_{ta} & r_{tc} & r_{tg} & - \end{pmatrix} \tag{1}$$

Here, $r\_\{xy\}$ describes the rate of nucleotide change from $x \to y$. Due to mathematical reasons, only *time reversible* models are considered by BEAST. The substitution rate matrix in time reversible models takes the form:

$$Q = \begin{pmatrix} - & r_{ac}\pi_C & r_{ag}\pi_G & r_{at}\pi_T \\ r_{ac}\pi_A & - & r_{cg}\pi_G & r_{ct}\pi_T \\ r_{ag}\pi_A & r_{cg}\pi_C & - & r_{gt}\pi_T \\ r_{at}\pi_A & r_{ct}\pi_C & r_{gt}\pi_G & - \end{pmatrix} = \begin{pmatrix} - & r_{ac} & r_{ag} & r_{at} \\ r_{ac} & - & r_{cg} & r_{ct} \\ r_{ag} & r_{cg} & - & r_{gt} \\ r_{at} & r_{ct} & r_{gt} & - \end{pmatrix} \times \begin{pmatrix} \pi_A & 0 & 0 & 0 \\ 0 & \pi_C & 0 & 0 \\ 0 & 0 & \pi_G & 0 \\ 0 & 0 & 0 & \pi_T \end{pmatrix} \tag{2}$$

where $\pi\_\{X\}$ is the equilibrium frequency of nucleotide X. The latter decomposition comes in handy to understand the parameterisation used in **bModelTest**, here the first matrix is referred to as *rate matrix* and the second is a *diagonal matrix* with the equilibrium frequencies on the diagonal.

The different named substitution models (e.g. JC69, HKY, TN93 and GTR) group these rates into different categories. *Group* here means that the rates are the same. For example, the JC69 model groups all rates together into a single rate category, i.e. $r\_\{ac\} = r\_\{ag\} = r\_\{at\} = r\_\{cg\} = r\_\{ct\} = r\_\{gt\}$. In addition, this model assumes equal equilibrium frequencies. The GTR model, however, assigns each rate to a different category and assumes a different equilibrium frequency for each nucleotide. We are therefore faced with the difficult choice of deciding *a priori* which one of these substitution models is most appropriate for our data.

Topic for discussion: In terms of phylogenetic inference, what would the consequences be of picking a substitution model that is overparameterized (too complex) for a given data set? What would the consequences be of picking a model that is underparameterized?

In addition to the substitution model, we also need to decide whether to include rate heterogeneity across sites. We might also want to include a proportion of invariant sites. On top of all this, we need to decide whether to estimate nucleotide base frequencies (the $\pi\_\{X\}$ in the equation above) or fix them at their empirical frequencies. All of these choices leads to a bewildering number of different models to choose from. For this reason, researchers have often based their model choice on common conventions rather than on which model is most appropriate for their data.

Fortunately, nowadays we can be more sophisticated in our modeling choices and let the data inform us about which model is most appropriate using Bayesian model averaging. In this tutorial, we will use BEAST2's model averaging tool **bModelTest** (**Bouckaert2017**) to select the most appropriate substitution model for the primate mitochondrial data set we already saw in the introductory tutorial. **bModelTest** uses reversible jump MCMC (rjMCMC), which allows the Markov chain to jump between states representing different possible substitution models, much like we jump between different parameter states in standard Bayesian MCMC inference. This allows us to treat the substitution model as a nuisance parameter and integrate over all *available* (more on this later) substitution models while simultaneously estimating the phylogeny and other model parameters. Thus, parameter estimates are effectively averaged over different substitution models, weighted by the support of each model. A useful consequence is that as we are exploring the space of different substitution models we also log the proportion of time that the Markov chain spends in a particular model state. This can be interpreted as the posterior support of a model, which tells us how strongly the data and our prior beliefs support a model in comparison to other competing models.

Note that **bModelTest** is only able to average over a subset of substitution models that are (a) implemented in BEAST2 and (b) that it knows how to move between. Ideally we would want to integrate over all possible substitution models, but since non-reversible models are mathematically inconvenient we restrict ourselves to the set of time-reversible (symmetric) nucleotide substitution models, which leaves us with 203 possible models. In addition, we can jump between models with empirical/estimated base frequencies, with/without gamma distributed rate heterogeneity and with/without invariant sites, resulting in a total of 203 x 2 x 2 x 2 = 1,624 possible model combinations.

2 Programs used in this Exercise

2.0.1 BEAST2 - Bayesian Evolutionary Analysis Sampling Trees

BEAST2 (`http://www.beast2.org`) is a free software package for Bayesian evolutionary analysis of molecular sequences using MCMC and strictly oriented toward inference using rooted, time-measured phylogenetic trees. This tutorial is written for BEAST v{{ page.beastversion }} (**Bouckaert2014**).

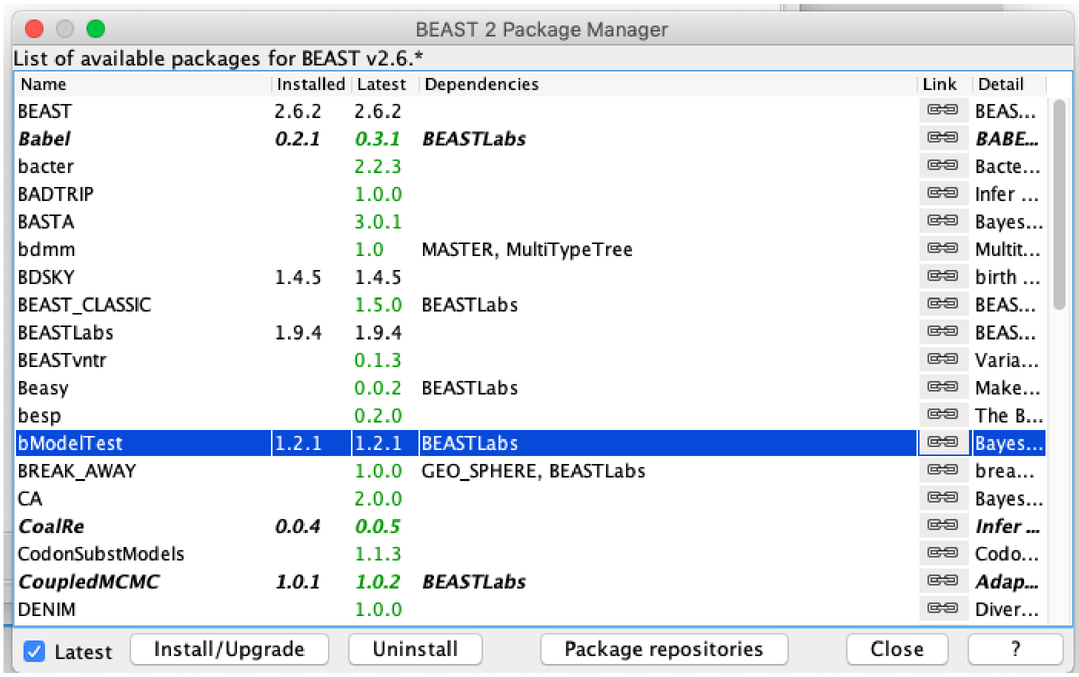
2.0.2 BEAUti2 - Bayesian Evolutionary Analysis Utility

BEAUti2 is a graphical user interface tool for generating BEAST2 XML configuration files.

Both BEAST2 and BEAUti2 are Java programs, which means that the exact same code runs on all platforms. For us it simply means that the interface will be the same on all platforms. The screenshots used in this tutorial are taken on a Mac OS X computer; however, both programs will have the same layout and functionality on both Windows and Linux. BEAUti2 is provided as a part of the BEAST2 package so you do not need to install it separately.

2.0.3 Tracer

Tracer (`http://tree.bio.ed.ac.uk/software/tracer`) is used to summarise the posterior estimates of the various parameters sampled by the Markov Chain. This program can be used for visual inspection and to assess convergence. It helps to quickly view median estimates and 95% highest posterior density intervals of the parameters, and calculates the effective sample sizes (ESS) of parameters. It can also be used to investigate potential parameter correlations. We will be using Tracer v{{ page.tracerversion }}.



BEAST 2 Package Manager

List of available packages for BEAST v2.6.*

Name	Installed	Latest	Dependencies	Link	Detail
BEAST	2.6.2	2.6.2		🔗	BEAS...
Babel	***0.2.1***	***0.3.1***	***BEASTLabs***	🔗	***BABE...***
bacter		2.2.3		🔗	Bacte...
BADTRIP		1.0.0		🔗	Infer ...
BASTA		3.0.1		🔗	Bayes...
bdmm		1.0	MASTER, MultiTypeTree	🔗	Multit...
BDSKY	1.4.5	1.4.5		🔗	birth ...
BEAST_CLASSIC		1.5.0	BEASTLabs	🔗	BEAS...
BEASTLabs	1.9.4	1.9.4		🔗	BEAS...
BEASTvntr		0.1.3		🔗	Varia...
Beasy		0.0.2	BEASTLabs	🔗	Make...
besp		0.2.0		🔗	The B...
bModelTest	1.2.1	1.2.1	BEASTLabs	🔗	Bayes...
BREAK_AWAY		1.0.0	GEO_SPHERE, BEASTLabs	🔗	brea...
CA		2.0.0		🔗	Bayes...
CoalRe	***0.0.4***	***0.0.5***		🔗	***Infer ...***
CodonSubstModels		1.1.3		🔗	Codo...
CoupledMCMC	***1.0.1***	***1.0.2***	***BEASTLabs***	🔗	***Adap...***
DENIM		1.0.0		🔗	Diver...

Figure 1: Installing bModelTest in the Manage Packages window in BEAUti

3 Practical: Selecting a substitution model

In this tutorial we will go through an analysis using bModelTest in BEAST v{{ page.beastversion }} and look into how to interpret the results. This tutorial assumes that you have already done some of the other tutorials and that you are familiar with the basics of using BEAUti, BEAST and Tracer.

3.1 Installing the bModelTest Package

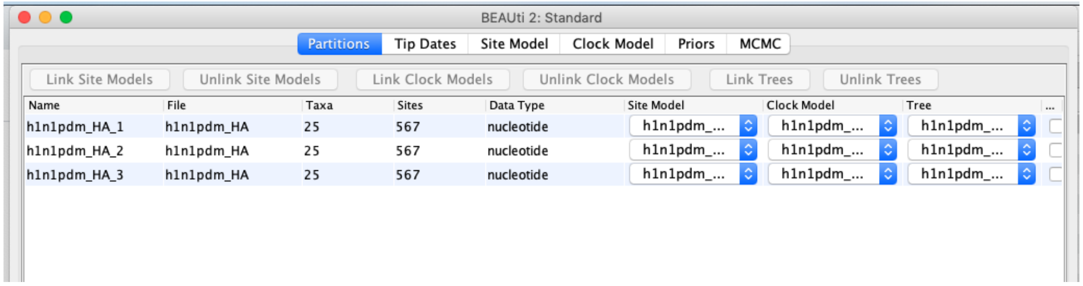
We first have to install the bModelTest (version {{ page.bmodeltestversion }} or above) package.

> Open BEAUti and navigate to **File > Manage Packages**. Select bModelTest and then click **Install/Upgrade** (Figure 1). Then ***restart BEAUti*** to load the package.

3.2 The Data

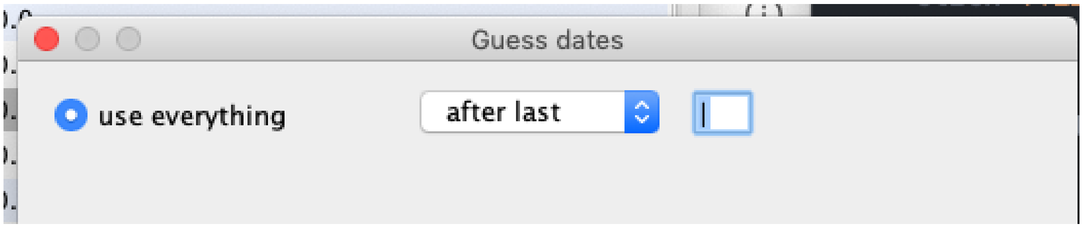
We will continue analyzing influenza A/H1N1 sequence data sampled over several years. All the sequences are from the hemagglutinin (HA) segment.

> Open BEAUti and navigate to **File > Import Alignment**. Select the file `h1n1pdm_HA.fasta` in the data directory.



Name	File	Taxa	Sites	Data Type	Site Model	Clock Model	Tree	...
h1n1pdm_HA_1	h1n1pdm_HA	25	567	nucleotide	h1n1pdm_...	h1n1pdm_...	h1n1pdm_...	□
h1n1pdm_HA_2	h1n1pdm_HA	25	567	nucleotide	h1n1pdm_...	h1n1pdm_...	h1n1pdm_...	□
h1n1pdm_HA_3	h1n1pdm_HA	25	567	nucleotide	h1n1pdm_...	h1n1pdm_...	h1n1pdm_...	□

Figure 2: Linking the Site Model across partitions in BEAUti.



● use everything after last

Figure 3: Select use everything after last to set the sampling dates.

3.3 Setting up the analysis in BEAUti

First, we will need to tell BEAST to consider the different codon positions separately. To do so, click the alignment and the split and choose 1+2+3. This will consider the first, second and third codon position separately.

> In the **Partitions** panel select all three partitions (w **shift+click**) and then click **Link Site Models**, **Link Clock Models** and **Link Trees**.

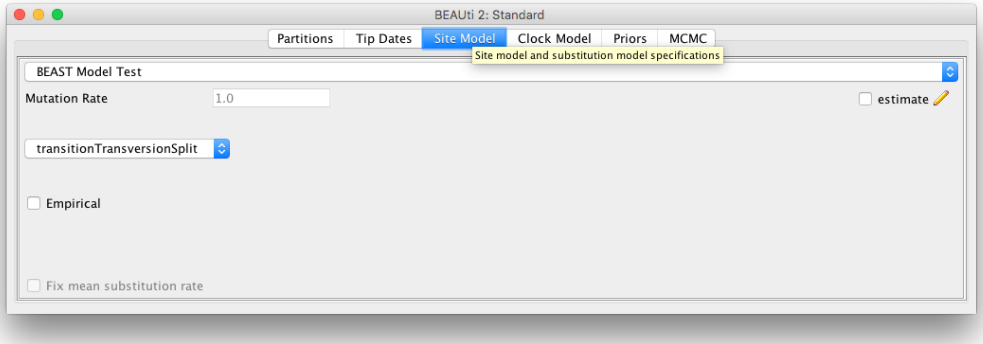
The Partition window should now look like Figure 3.

Next, go to Tip Dates and click use tip dates. Set `as dates with format` to `yyyy-M-dd` and click `Auto-configure`. Then, select use everything after Last as in the figure below

Now we want to set up our Site Model to run the model averaging analysis.

> Click the **Site Model** tab in BEAUti and then select the drop-down box at the top which says **Gamma Site Model** and change it to **BEAST Model Test** (Figure 4).

In the lower drop-down box we will keep **transitionTransversionSplit** selected. This tells bModelTest to only consider substitution models that differentiate between transitions (A ↔ G and C ↔ T) and transversions (all other substitutions). If all possibilities to group the rates in the substitution rate matrix, independent of whether substitutions are transitions or transersions, there are a total of 203 reversible models with symmetric rate matrices (**Bouckaert2017**). However, if we only consider models that group transitions together with transitions but not with transversions, there are only 31 models. Selecting **transitionTransversionSplit** therefore dramatically reduces the number of models that we need to explore.



BEAST Model Test

Mutation Rate 1.0 □ estimate

transitionTransversionSplit

□ Empirical

□ Fix mean substitution rate

Figure 4: Setting up the BEAST ModelTest.

Additionally, select estimate next to the **Mutation Rate**. This allows the different codon positions to evolve at different speeds, which are estimated during the MCMC.

Next, go back to **Partitions** and click **Unlick Site Models**. This will allow each codon position to have its own site model.

In the **Clock Model** we do not need to change any of the default settings for this tutorial.

In the **Priors Tab** change the tree prior to Coalescent Constant Population (We will discuss tree priors in more details on Tuesday).

Click the **MCMC** tab in BEAUti. Change the chain length to 5 000 000 and the sampling frequency to every 5 000 by changing **Log Every** under **tracelog** and **treelog**. This will help us to avoid autocorrelation by ensuring that we are not sampling too frequently. (You can also increase **Log Every** under **screenlog** to keep BEAST2 from producing too much screen output). Change the **tracelog** file name to `h1n1_pdm.log` and the **treelog** file name to `h1n1_pdm.trees`. Click **File > Save As** and save as `h1n1_pdm.xml`. You can now close BEAUTi.

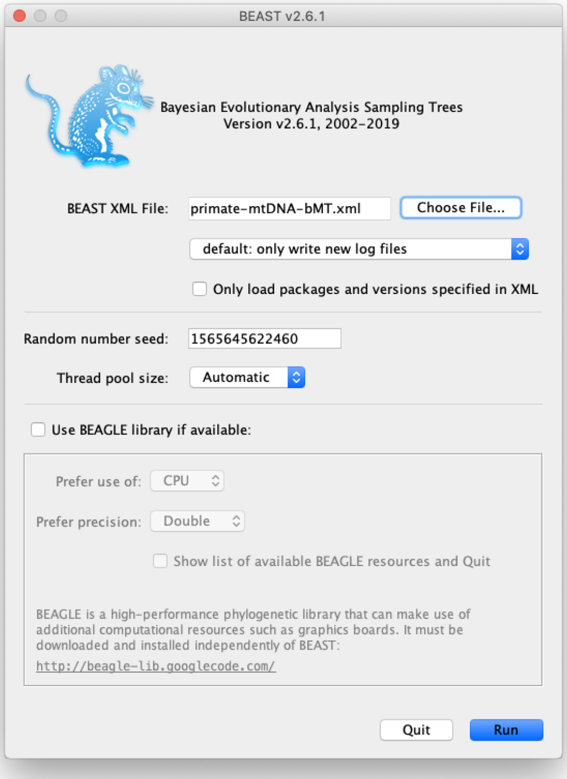
3.4 Run the analysis in BEAST

Open BEAST and choose `primate-mtDNA-bMT.xml` as the BEAST XML File (Figure 5). If BEAGLE is installed check the box to use it. Then click **run**.

BEAST will now run for a couple of minutes

3.5 Analyzing the output in Tracer

Open the `h1n1_pdm.log` file in Tracer. There should be a long list of entries in the window on the left hand side (Figure 6).



BEAST v2.6.1

Bayesian Evolutionary Analysis Sampling Trees
Version v2.6.1, 2002-2019

BEAST XML File: primate-mtDNA-bMT.xml

default: only write new log files

□ Only load packages and versions specified in XML

Random number seed: 1565645622460

Thread pool size: Automatic

□ Use BEAGLE library if available:

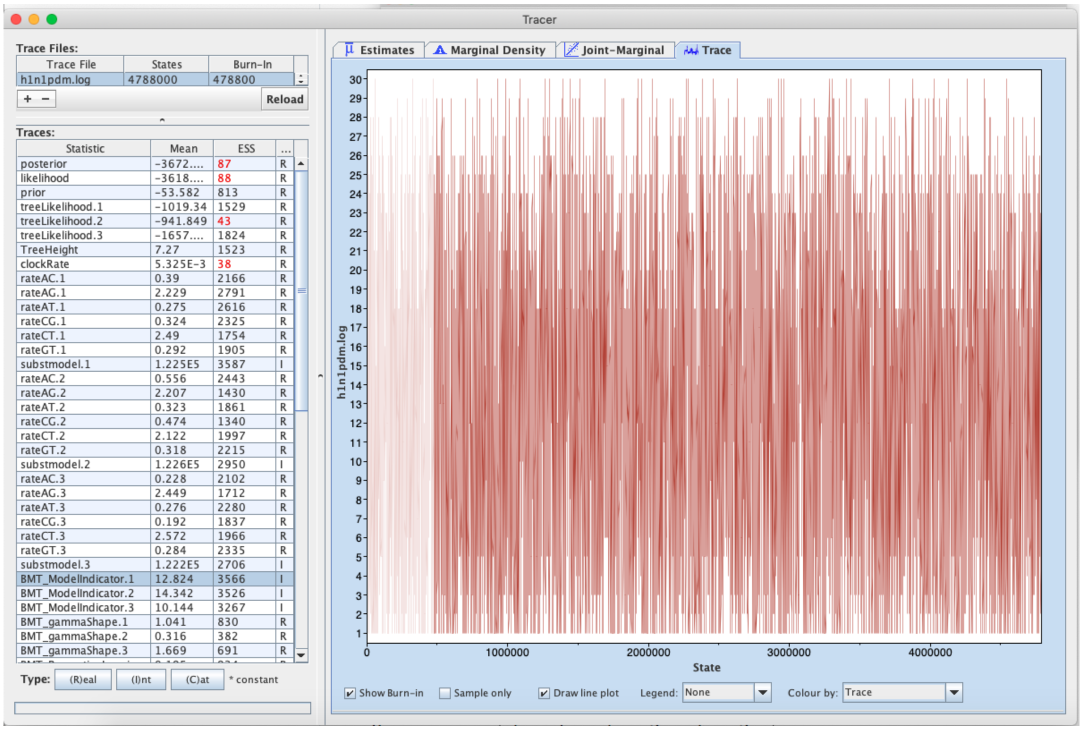
Prefer use of: CPU

Prefer precision: Double

□ Show list of available BEAGLE resources and Quit

BEAGLE is a high-performance phylogenetic library that can make use of additional computational resources such as graphics boards. It must be downloaded and installed independently of BEAST:
http://beagle-lib.googlecode.com/

Figure 5: Running the analysis in BEAST.



Trace Files:

Trace File	States	Burn-In
h1n1pdm.log	4788000	478800

Traces:

Statistic	Mean	ESS	...
posterior	-3672....	87	R
likelihood	-3618....	88	R
prior	-53.582	813	R
treeLikelihood.1	-1019.34	1529	R
treeLikelihood.2	-941.849	43	R
treeLikelihood.3	-1657....	1824	R
TreeHeight	7.27	1523	R
clockRate	5.325E-3	38	R
rateAC.1	0.39	2166	R
rateAG.1	2.229	2791	R
rateAT.1	0.275	2616	R
rateCG.1	0.324	2325	R
rateCT.1	2.49	1754	R
rateGT.1	0.292	1905	R
substmodel.1	1.225E5	3587	I
rateAC.2	0.556	2443	R
rateAG.2	2.207	1430	R
rateAT.2	0.323	1861	R
rateCG.2	0.474	1340	R
rateCT.2	2.122	1997	R
rateGT.2	0.318	2215	R
substmodel.2	1.226E5	2950	I
rateAC.3	0.228	2102	R
rateAG.3	2.449	1712	R
rateAT.3	0.276	2280	R
rateCG.3	0.192	1837	R
rateCT.3	2.572	1966	R
rateGT.3	0.284	2335	R
substmodel.3	1.222E5	2706	I
BMT_ModelIndicator.1	12.824	3566	I
BMT_ModelIndicator.2	14.342	3526	I
BMT_ModelIndicator.3	10.144	3267	I
BMT_gammaShape.1	1.041	830	R
BMT_gammaShape.2	0.316	382	R
BMT_gammaShape.3	1.669	691	R

Type: (R)eal (I)nt (C)at * constant

☑ Show Burn-in ☐ Sample only ☑ Draw line plot Legend: None Colour by: Trace

Figure 6: Visualizing how the Markov chain explored different models in Tracer.

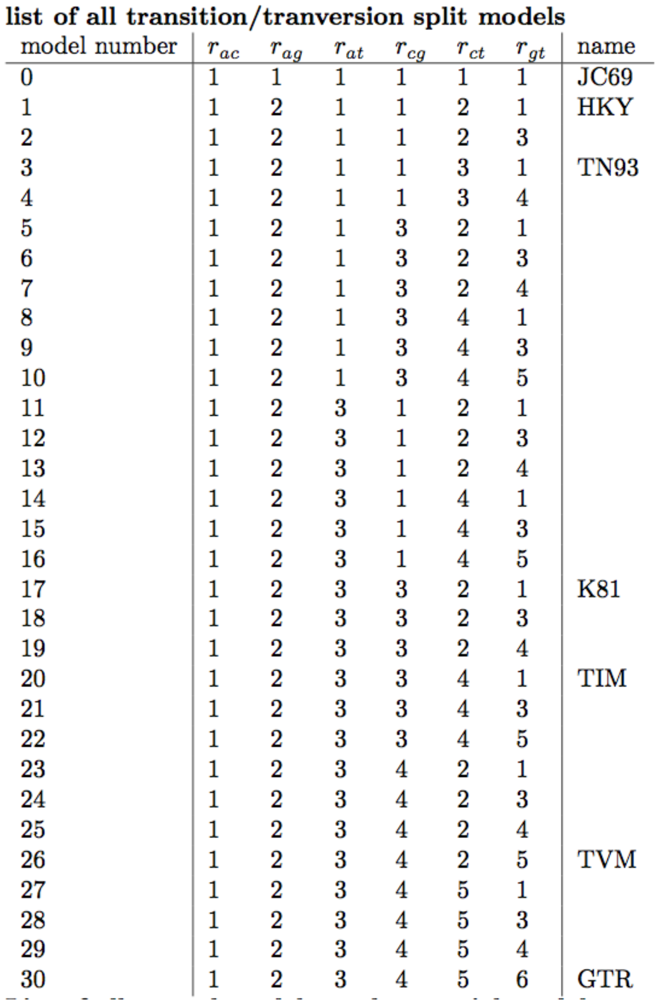
If we select **BMT_ModelIndicator.1** from the list of entries, we can see how the Markov chain explored the space of different models by jumping between substitution models. This is best seen if we click on the **Trace** tab (Figure 6). Here, the sampled integer values refer to the indexes of the different substitution models. To see which index corresponds to which model, refer to (Figure 7).

Now click on the **Estimates** tab above. This frequency histogram shows us how much time the Markov chain spent in each model state relative to other model states, and therefore reflects the posterior support for each model. We can see that the chain spent the most time in model number 1 (Figure 8), which by consulting Figure 7 we see corresponds to the HKY model.

> **Topic for discussion:** Did the chain ever visit the JC69 model? Why not?

We can also use the output of our analysis to see if a model with (gamma) rate heterogeneity and/or a proportion of invariant sites is supported. If we select **hasGammaRates** in the window on the left and then click **Estimates** we see the proportion of time the chain spent in a model state with rate heterogeneity on (1) versus off (0), and thus the posterior support for a model with rate heterogeneity. Here, the chain seems to remain in a state with rate heterogeneity on, indicating very strong support for heterogeneity (Figure 9). We can also select **hasInvariableSites** to see if a model with invariant sites is supported. Here we see that the model spends more time in a model state with invariant sites off (0) than on (1), indicating that the presence of invariant sites are not as strongly supported (Figure 10). Note that we can also look at the traces for **BMT_gammaShape** and **BMT_ProportionInvariant** to see which values of these two parameters the chain visited.

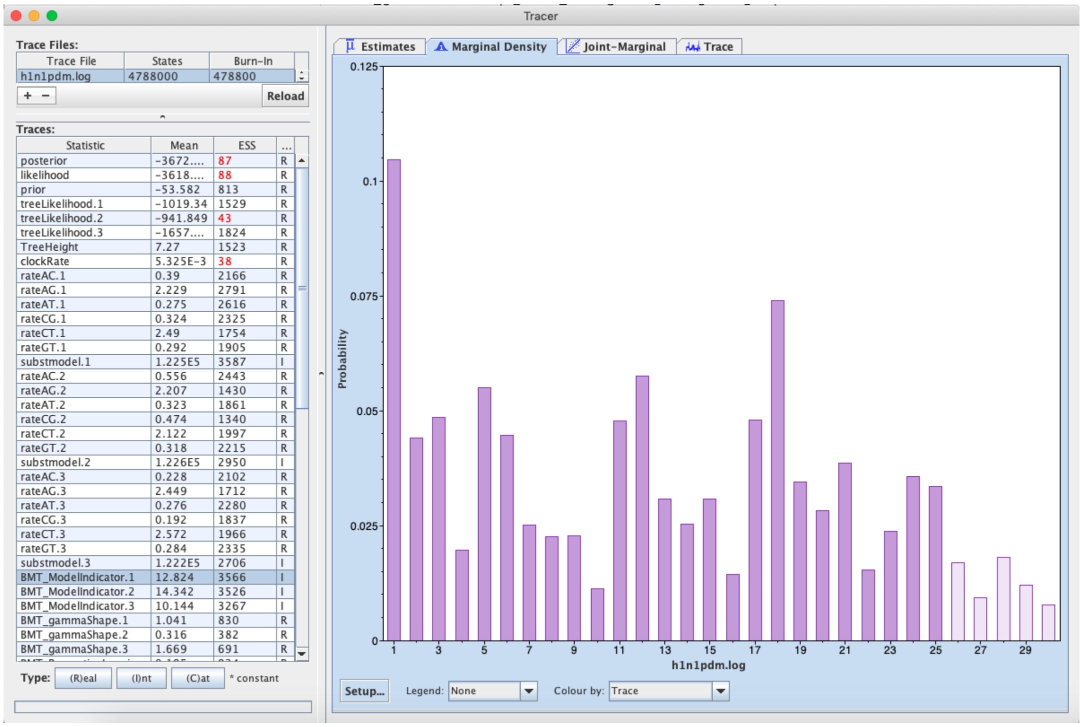
There are a few other things we can look at in Tracer as well:



list of all transition/tranversion split models

model number	r_{ac}	r_{ag}	r_{at}	r_{cg}	r_{ct}	r_{gt}	name
0	1	1	1	1	1	1	JC69
1	1	2	1	1	2	1	HKY
2	1	2	1	1	2	3	
3	1	2	1	1	3	1	TN93
4	1	2	1	1	3	4	
5	1	2	1	3	2	1	
6	1	2	1	3	2	3	
7	1	2	1	3	2	4	
8	1	2	1	3	4	1	
9	1	2	1	3	4	3	
10	1	2	1	3	4	5	
11	1	2	3	1	2	1	
12	1	2	3	1	2	3	
13	1	2	3	1	2	4	
14	1	2	3	1	4	1	
15	1	2	3	1	4	3	
16	1	2	3	1	4	5	
17	1	2	3	3	2	1	K81
18	1	2	3	3	2	3	
19	1	2	3	3	2	4	
20	1	2	3	3	4	1	TIM
21	1	2	3	3	4	3	
22	1	2	3	3	4	5	
23	1	2	3	4	2	1	
24	1	2	3	4	2	3	
25	1	2	3	4	2	4	
26	1	2	3	4	2	5	TVM
27	1	2	3	4	5	1	
28	1	2	3	4	5	3	
29	1	2	3	4	5	4	
30	1	2	3	4	5	6	GTR

Figure 7: A list of substitution models, their associated model number (index) and how the substitution rates are grouped.



Tracer

Trace Files:

Trace File	States	Burn-In
h1n1pdm.log	4788000	478800

Traces:

Statistic	Mean	ESS	...
posterior	-3672....	87	R
likelihood	-3618....	88	R
prior	-53.582	813	R
treeLikelihood.1	-1019.34	1529	R
treeLikelihood.2	-941.849	43	R
treeLikelihood.3	-1657....	1824	R
TreeHeight	7.27	1523	R
clockRate	5.325E-3	38	R
rateAC.1	0.39	2166	R
rateAG.1	2.229	2791	R
rateAT.1	0.275	2616	R
rateCG.1	0.324	2325	R
rateCT.1	2.49	1754	R
rateGT.1	0.292	1905	R
substmodel.1	1.225E5	3587	I
rateAC.2	0.556	2443	R
rateAG.2	2.207	1430	R
rateAT.2	0.323	1861	R
rateCG.2	0.474	1340	R
rateCT.2	2.122	1997	R
rateGT.2	0.318	2215	R
substmodel.2	1.226E5	2950	I
rateAC.3	0.228	2102	R
rateAG.3	2.449	1712	R
rateAT.3	0.276	2280	R
rateCG.3	0.192	1837	R
rateCT.3	2.572	1966	R
rateGT.3	0.284	2335	R
substmodel.3	1.222E5	2706	I
BMT_ModelIndicator.1	12.824	3566	I
BMT_ModelIndicator.2	14.342	3526	I
BMT_ModelIndicator.3	10.144	3267	I
BMT_gammaShape.1	1.041	830	R
BMT_gammaShape.2	0.316	382	R
BMT_gammaShape.3	1.669	691	R

Type: (R)eal (I)nt (C)at * constant

Legend: None

Colour by: Trace

Figure 8: Visualizing the posterior support for different substitution models in Tracer.

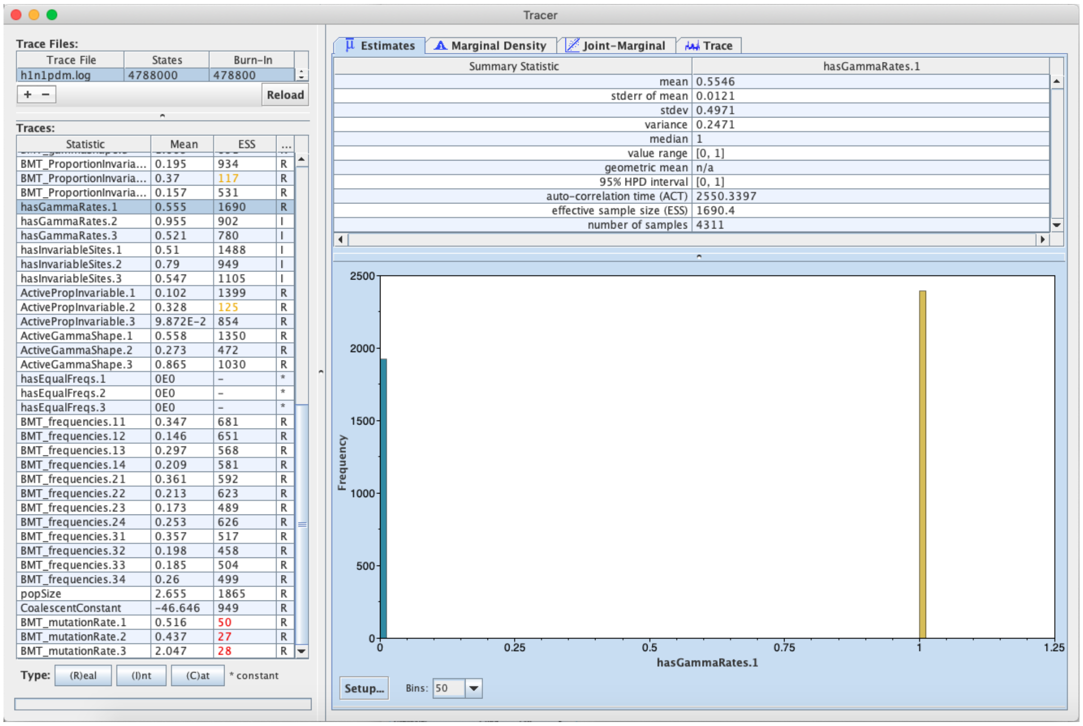
- **rateAC, . . . ,rateGT** are the substitution rates between pairs of nucleotides in the substitution matrix. Note that these rates are averaged over all the models, weighted by the time the Markov chain spent in each model state.
- **ActiveGammaShape/PropInvariable** are the gamma shape parameter and the proportion of variables sites when active, that is, when **hasGammaRates** and **hasInvariableSites** are selected. To get the estimate of the mean of the shape parameter, divide the mean **ActiveGammaShape** by the mean of **hasGammaRates**.
- **hasEqualFreqs** indicates if the chain is in a state with equal nucleotide base frequencies.

Select pairs of the **rateAC, . . . ,rateGT** parameters (using **shift+click**) and click on the **Joint-Marginal** tab to investigate parameter correlations (Figure 11). Try looking at **rateAT** vs. **rateCG** and **rateCG** vs **rateGT**).

Topic for discussion: It appears that some pairs of the rate parameters are highly correlated for some samples and uncorrelated for the rest. What is happening here? Should we be worried about these parameter correlations?

3.6 Analyzing the output using BModelAnalyzer

Another really nice feature of bModelTest is that we can graphically analyze the output using the **BModelAnalyser App**.



Trace Files:

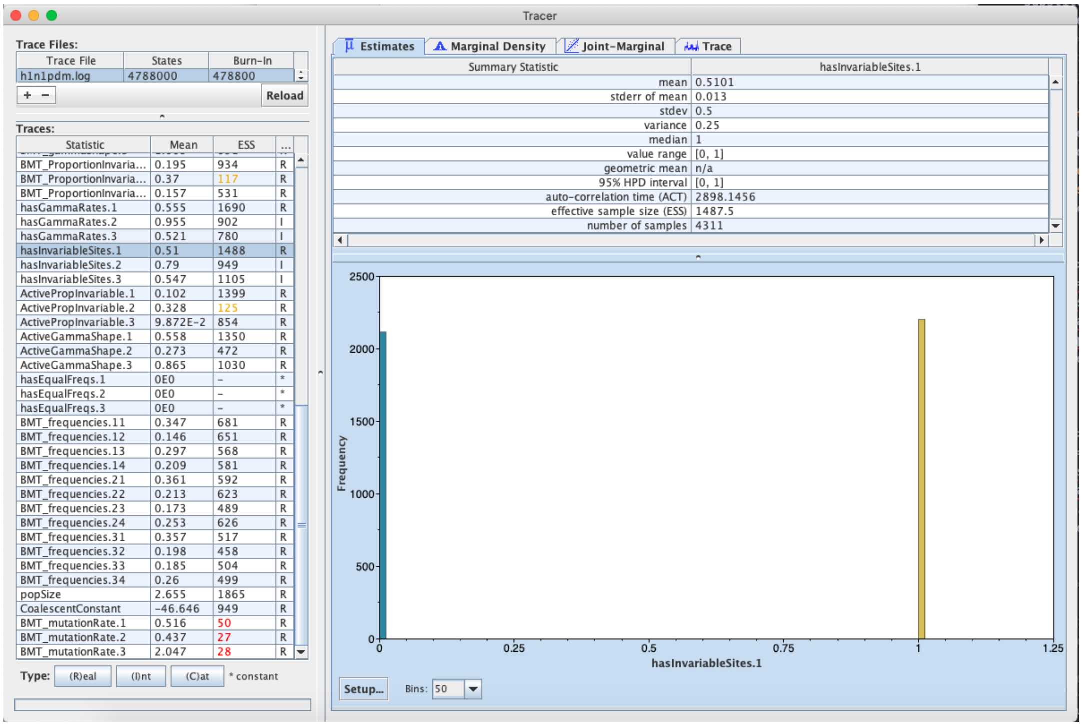
Trace File	States	Burn-In
h1n1pdm.log	4788000	478800

Traces:

Statistic	Mean	ESS	...
BMT_ProportionInvaria...	0.195	934	R
BMT_ProportionInvaria...	0.37	117	R
BMT_ProportionInvaria...	0.157	531	R
hasGammaRates.1	0.555	1690	R
hasGammaRates.2	0.955	902	I
hasGammaRates.3	0.521	780	I
hasInvariableSites.1	0.51	1488	I
hasInvariableSites.2	0.79	949	I
hasInvariableSites.3	0.547	1105	I
ActivePropInvariable.1	0.102	1399	R
ActivePropInvariable.2	0.328	125	R
ActivePropInvariable.3	9.872E-2	854	R
ActiveGammaShape.1	0.558	1350	R
ActiveGammaShape.2	0.273	472	R
ActiveGammaShape.3	0.865	1030	R
hasEqualFreqs.1	0E0	–	*
hasEqualFreqs.2	0E0	–	*
hasEqualFreqs.3	0E0	–	*
BMT_frequencies.11	0.347	681	R
BMT_frequencies.12	0.146	651	R
BMT_frequencies.13	0.297	568	R
BMT_frequencies.14	0.209	581	R
BMT_frequencies.21	0.361	592	R
BMT_frequencies.22	0.213	623	R
BMT_frequencies.23	0.173	489	R
BMT_frequencies.24	0.253	626	R
BMT_frequencies.31	0.357	517	R
BMT_frequencies.32	0.198	458	R
BMT_frequencies.33	0.185	504	R
BMT_frequencies.34	0.26	499	R
popSize	2.655	1865	R
CoalescentConstant	-46.646	949	R
BMT_mutationRate.1	0.516	50	R
BMT_mutationRate.2	0.437	27	R
BMT_mutationRate.3	2.047	28	R

Summary Statistic	hasGammaRates.1
mean	0.5546
stderr of mean	0.0121
stdev	0.4971
variance	0.2471
median	1
value range	[0, 1]
geometric mean	n/a
95% HPD interval	[0, 1]
auto-correlation time (ACT)	2550.3397
effective sample size (ESS)	1690.4
number of samples	4311

Figure 9: The posterior support for including gamma rate heterogeneity.



Trace Files:

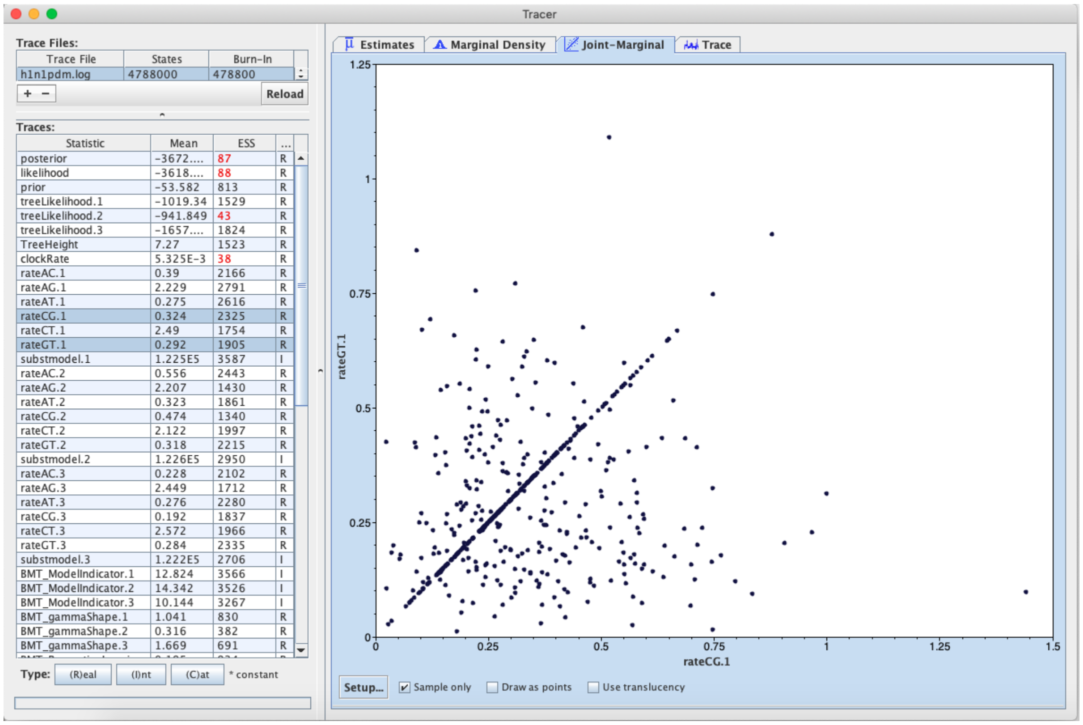
Trace File	States	Burn-In
h1n1pdm.log	4788000	478800

Traces:

Statistic	Mean	ESS	...
BMT_ProportionInvaria...	0.195	934	R
BMT_ProportionInvaria...	0.37	117	R
BMT_ProportionInvaria...	0.157	531	R
hasGammaRates.1	0.555	1690	R
hasGammaRates.2	0.955	902	I
hasGammaRates.3	0.521	780	I
hasInvariableSites.1	0.51	1488	R
hasInvariableSites.2	0.79	949	I
hasInvariableSites.3	0.547	1105	I
ActivePropInvariable.1	0.102	1399	R
ActivePropInvariable.2	0.328	125	R
ActivePropInvariable.3	9.872E-2	854	R
ActiveGammaShape.1	0.558	1350	R
ActiveGammaShape.2	0.273	472	R
ActiveGammaShape.3	0.865	1030	R
hasEqualFreqs.1	0E0	–	*
hasEqualFreqs.2	0E0	–	*
hasEqualFreqs.3	0E0	–	*
BMT_frequencies.11	0.347	681	R
BMT_frequencies.12	0.146	651	R
BMT_frequencies.13	0.297	568	R
BMT_frequencies.14	0.209	581	R
BMT_frequencies.21	0.361	592	R
BMT_frequencies.22	0.213	623	R
BMT_frequencies.23	0.173	489	R
BMT_frequencies.24	0.253	626	R
BMT_frequencies.31	0.357	517	R
BMT_frequencies.32	0.198	458	R
BMT_frequencies.33	0.185	504	R
BMT_frequencies.34	0.26	499	R
popSize	2.655	1865	R
CoalescentConstant	-46.646	949	R
BMT_mutationRate.1	0.516	50	R
BMT_mutationRate.2	0.437	27	R
BMT_mutationRate.3	2.047	28	R

Summary Statistic	hasInvariableSites.1
mean	0.5101
stderr of mean	0.013
stdev	0.5
variance	0.25
median	1
value range	[0, 1]
geometric mean	n/a
95% HPD interval	[0, 1]
auto-correlation time (ACT)	2898.1456
effective sample size (ESS)	1487.5
number of samples	4311

Figure 10: The posterior support for including a proportion of invariant sites.



Trace Files:

Trace File	States	Burn-In
h1n1pdm.log	4788000	478800

Traces:

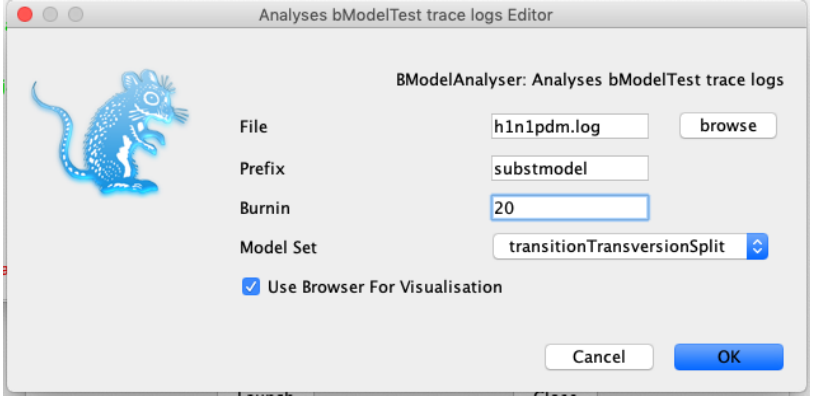
Statistic	Mean	ESS	...
posterior	-3672....	87	R
likelihood	-3618....	88	R
prior	-53.582	813	R
treeLikelihood.1	-1019.34	1529	R
treeLikelihood.2	-941.849	43	R
treeLikelihood.3	-1657....	1824	R
TreeHeight	7.27	1523	R
clockRate	5.325E-3	38	R
rateAC.1	0.39	2166	R
rateAG.1	2.229	2791	R
rateAT.1	0.275	2616	R
rateCG.1	0.324	2325	R
rateCT.1	2.49	1754	R
rateGT.1	0.292	1905	R
substmodel.1	1.225E5	3587	I
rateAC.2	0.556	2443	R
rateAG.2	2.207	1430	R
rateAT.2	0.323	1861	R
rateCG.2	0.474	1340	R
rateCT.2	2.122	1997	R
rateGT.2	0.318	2215	R
substmodel.2	1.226E5	2950	I
rateAC.3	0.228	2102	R
rateAG.3	2.449	1712	R
rateAT.3	0.276	2280	R
rateCG.3	0.192	1837	R
rateCT.3	2.572	1966	R
rateGT.3	0.284	2335	R
substmodel.3	1.222E5	2706	I
BMT_ModelIndicator.1	12.824	3566	I
BMT_ModelIndicator.2	14.342	3526	I
BMT_ModelIndicator.3	10.144	3267	I
BMT_gammaShape.1	1.041	830	R
BMT_gammaShape.2	0.316	382	R
BMT_gammaShape.3	1.669	691	R

Figure 11: Correlations between rate parameters.

In BEAUti, select **File > Launch Apps** and then launch the **BModelAnalyser App**. A dialogue window should pop up (Figure 12). Enter `h1n1_pdm.log` as the file to analyze. You can leave the other entries at their default settings but make sure **transitionTransversionSplit** is selected for the Model Set. Make sure that the Burnin is the same as we used before in Tracer and that the box next to **Use Browser For Visualization** is checked. Then click **OK**.

After BModelAnalyser runs, three new windows should appear in your default web browser that represents the model selection results graphically (Figure 13). Each window represents the results for one of the three codon positions. This graph depicts the nested relationship of the different substitution models: an arrow pointing from one model to another indicates that the model at the tail is nested within the model at the head of the arrow. As we can see, JC69 is nested within all other models and all other models are nested within GTR.

The area of the circle surrounding each model is proportional to the the posterior support for that model. The colours represent whether the model is contained within the 95% credible set (blue) or not (red). For the h1n1 data set, the HKY model has the highest posterior support (Figure 13). However, other models, also have fairly high posterior support. The six digit model code describes how the different substitution rates are grouped in the order of $r\_\{ac\}$, $r\_\{ag\}$, $r\_\{at\}$, $r\_\{cg\}$, $r\_\{ct\}$ and $r\_\{gt\}$. For instance, **121323** is a slight variant of the HKY model with an additional group for the rates $r\_\{ct\}$ and $r\_\{gt\}$. The six digit codes for all models are shown in Figure 7.



Analyses bModelTest trace logs Editor

BModelAnalyser: Analyses bModelTest trace logs

File h1n1pdm.log

Prefix substmodel

Burnin 20

Model Set transitionTransversionSplit

☑ Use Browser For Visualisation

Figure 12: Running the BModelAnalyser App.

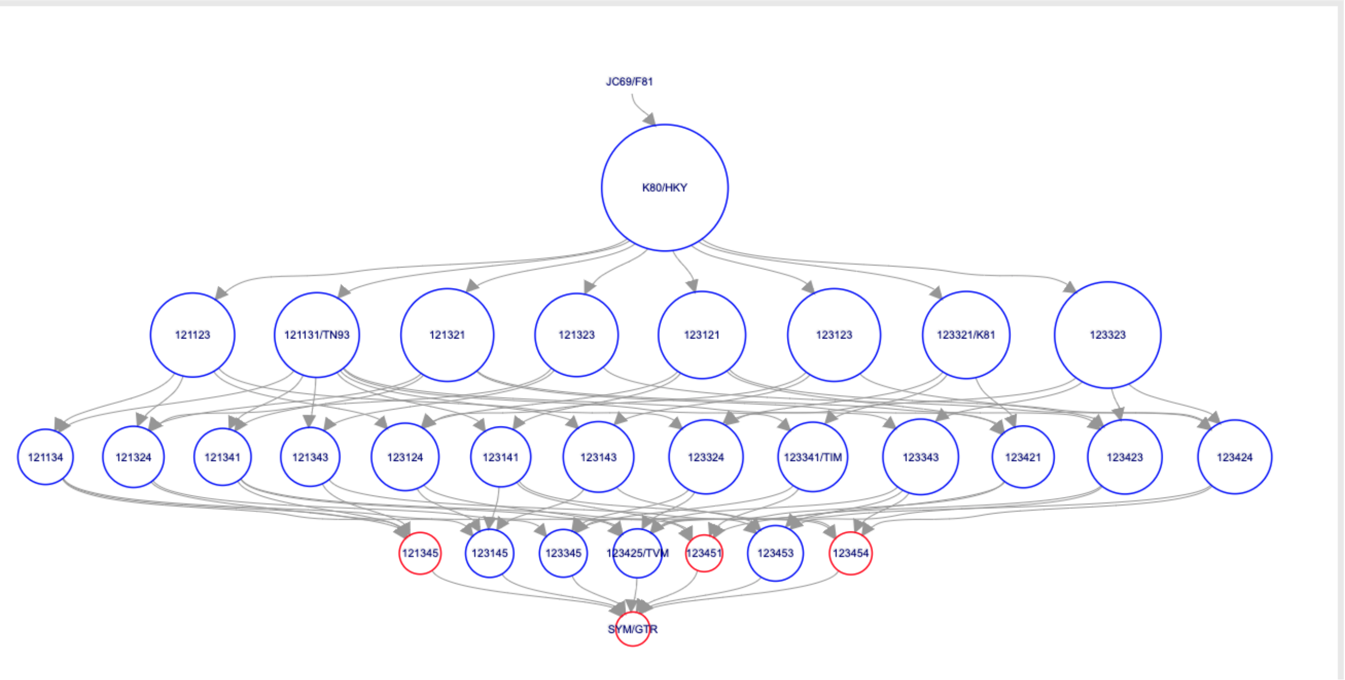


Figure 13: A graphical representation of the model selection results produced by BModelAnalyzer.

Topic for discussion: We have used bModelTest to explore a large set of substitution models. But how do we know that any of the substitution models actually fit the observed sequence data well?

4 Acknowledgment

This tutorial is based on the original bModelTest tutorial by Remco Bouckaert.

5 Useful Links

- Official bModelTest documentation: `https://github.com/BEAST2-Dev/bModelTest/wiki`
- The original bModelTest tutorial is available here: `https://github.com/BEAST2-Dev/bModelTest/releases/download/v0.3.0/bModelTestTutorial.pdf` and is also included in the source code.