Tutorial using BEAST v2.6.0

Prior-selection

Veronika Bošková, Venelin Mitov and Louis du Plessis
Adapted by Joëlle Barido-Sottani

Prior selection and clock calibration using SARS-CoV-2 data.

1 Background

In the Bayesian analysis of sequence data, priors play an important role. When priors are not specified correctly, it may cause runs to take a long time to converge, not converge at all or cause a bias in the inferred trees and model parameters. Specifying proper priors and starting values is crucial and can be a difficult exercise in the beginning. It is not always easy to pick a proper model of tree generation (tree prior), substitution model, molecular clock model or the prior distribution for an unknown parameter.

In this tutorial we will explore how priors are selected and how molecular clock calibration works using SARS-CoV-2 data from the current epidemic. The molecular clock model aims to estimate the substitution rate of the data. It is important to understand under which circumstances to use which model and when molecular calibration works. This will help the investigator determine which estimates of parameters can be trusted and which cannot.

2 Programs used in this exercise

2.0.1 BEAST2 - Bayesian Evolutionary Analysis Sampling Trees

BEAST2 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 v2.5.2 (Bouckaert et al. 2014), (Bouckaert et al. 2019).

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 TreeAnnotator

TreeAnnotator is used to summarise the posterior sample of trees to produce a maximum clade credibility tree. It can also be used to summarise and visualise the posterior estimates of other tree parameters (e.g. node height).

TreeAnnotator is provided as a part of the BEAST2 package so you do not need to install it separately.

2.0.4 Tracer

Tracer (`http://beast.community/tracer`) is used to summarize 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 v1.7.0.

2.0.5 FigTree

FigTree (`http://beast.community/figtree`) is a program for viewing trees and producing publication-quality figures. It can interpret the node-annotations created on the summary trees by TreeAnnotator, allowing the user to display node-based statistics (e.g. posterior probabilities). We will be using FigTree v1.4.4.

3 Practical: SARS-CoV-2 dynamics

In this tutorial, we will estimate the rate of evolution from a set of SARS-CoV-2 sequences from different points in time (heterochronous or time-stamped data). The aim of this tutorial is to obtain estimates for the:

- rate of molecular evolution
- phylogenetic relationships with measures of internal node heights
- date of the most recent common ancestor of the sampled virus sequences.

The more general aim of this tutorial is to:

- understand how to set priors and why this is important
- understand why and when the rate of evolution can be estimated from the data.

After completing this tutorial you should be able to:

- set up and run a BEAST2 XML file with heterochronous data
- install and use a BEAST2 package
- decide if an MCMC chain has converged or not
- identify parameter correlations

3.1 Creating the analysis file with BEAUti

We will use BEAUti to select the priors and starting values for our analysis and save these settings into a BEAST2 XML file.

Begin by starting **BEAUti2**.

3.1.1 Installing BEAST2 packages

Since we will be using the birth-death skyline model (**BDSKY**) (Stadler et al. 2013), we need to make sure it is available in BEAUti. It is not one of the default models but rather a package (also sometimes called a plug-in or an add-on). You only need to install a BEAST2 package once. After installing, if you close BEAUti, you do not have to load **BDSKY** again the next time you open the program. However, it is worth checking the package manager for updates to packages, particularly if you update your version of BEAST2. For this tutorial we need to ensure that we have at least BDSKY v1.4.5 installed.

Open the **BEAST2 Package Manager** by navigating to **File > Manage Packages**. (Figure 1)

Install the **BDSKY** package by selecting it and clicking the **Install/Upgrade** button. (Figure 2)

After the installation of a package, the program is on your computer, but BEAUti is unable to load the template files for the newly installed model unless it is restarted. So, let's restart BEAUti to make sure we have the **BDSKY** model at hand.

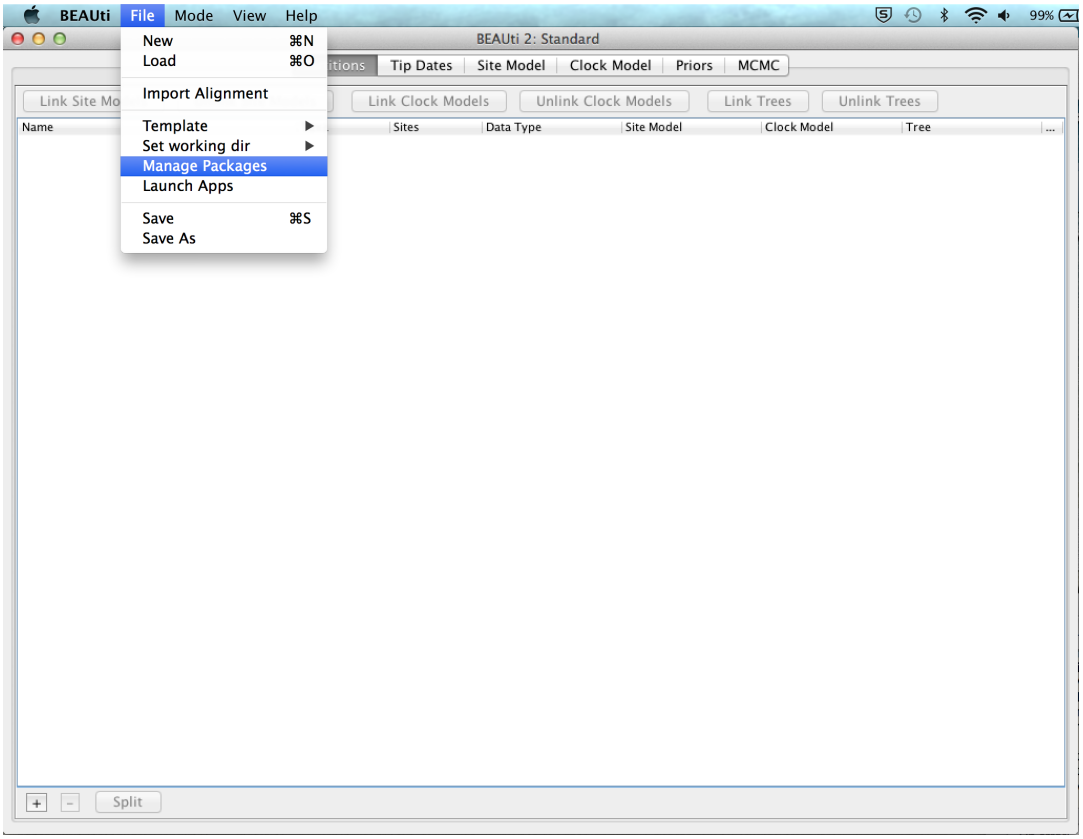

Figure 1: Finding the BEAST2 Package Manager.

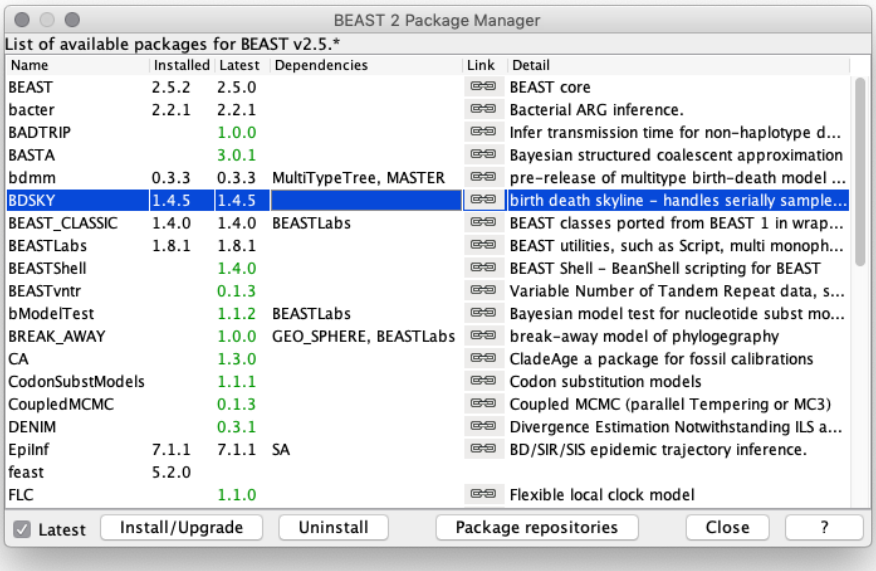

Figure 2: The BEAST2 Package Manager.

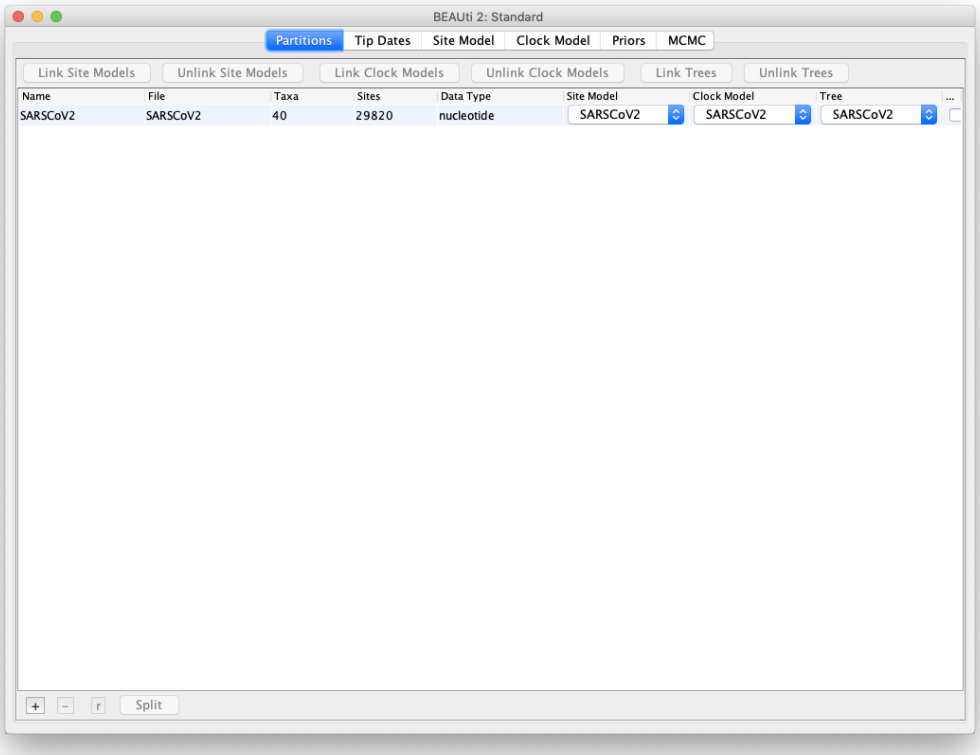

Figure 3: Data imported into BEAUti2.

Close the **BEAST2 Package Manager** and ***restart*** BEAUti to fully load the **BDSKY** package.

3.1.2 Importing the alignment

Note that the dataset used in this tutorial is the same as the one used in the introduction tutorial, so some of the sections will be identical.

In the **Partitions** panel, import the nexus file with the alignment by navigating to **File > Import Alignment** in the menu (Figure 3) and then finding the `SARSCoV2.fas`. file on your computer **or** simply drag and drop the file into the **BEAUti** window.

Then select **nucleotide** in the dropdown menu and click **OK**.

You can view the alignment by double-clicking on the name of the alignment in BEAUti. Since we only have one partition there is nothing more we can do in the **Partitions** panel and proceed to specifying the tip dates.

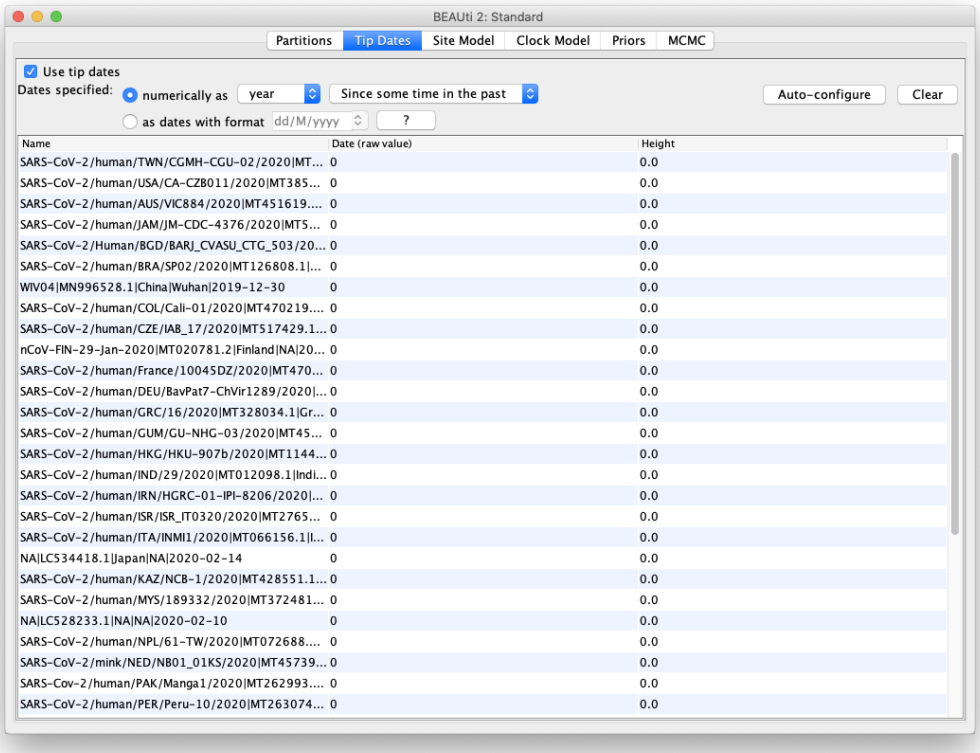

Figure 4: Tip dates panel.

3.1.3 Setting up tip dates

The next step is to set the sampling dates for all of our sequences, which is crucial to obtain accurate age estimates on the phylogeny. This is done in the **Tip Dates** tab.

> Select the **Tip Dates** tab.
>
> Select the **Use tip dates** option.

The panel for setting sampling dates will look as shown in Figure 4. By default, all sampling dates are set to $t = 0$, i.e. the present. We *could* input the dates manually, however this is time-consuming, so instead we are going to use the **Auto-configure** option, which allows us to set the dates automatically from the sequence labels.

The sequence labels (headers in the FASTA file) contain sampling times specified as dates in the format year-month-day. The first step is to specify this date format.

> At the top of the panel, select the **as dates with format** option, then select **yyyy-M-dd** as the format in the dropdown menu.

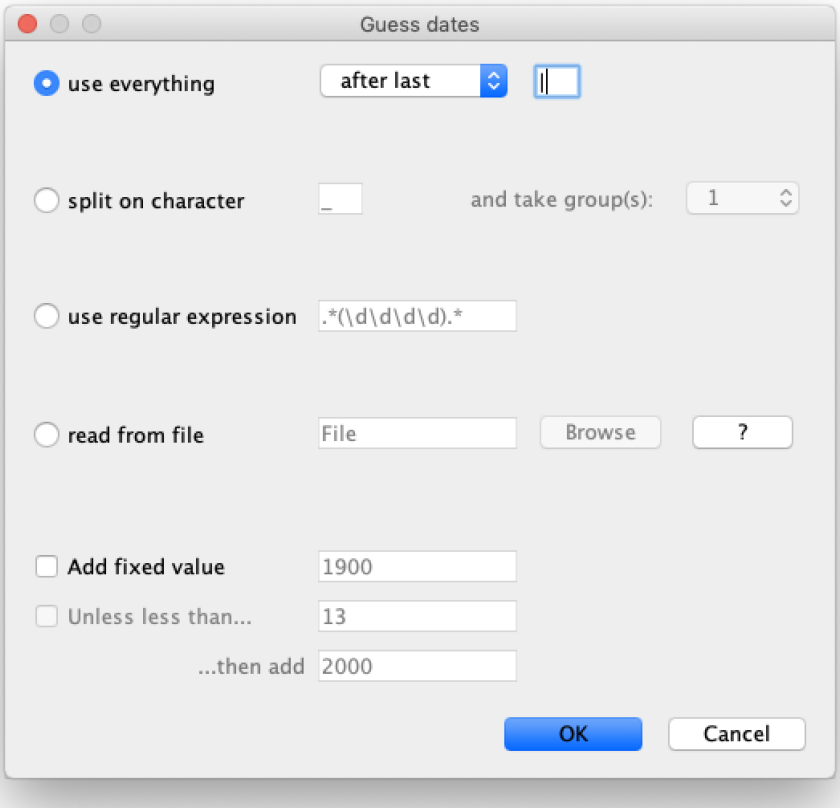

Figure 5: Auto-configure options.

Now we need to tell BEAUti where the date is stored in the sequence name. There are several different options.

Click on the **Auto-configure** button.

In the panel that opens, set the configuration to **use everything**, **after last** and |. The correct configuration is shown in Figure 5.

Click **OK**.

An error message will appear, telling us that not all dates could be configured manually. The problem sequences are shown in red. As shown in Figure 6, we can see that there are two problem sequences, where the day is missing from the sampling date. We need to edit those sequences manually to set the day, and we choose to set both sequences to have been sampled on the 15th.

Double click on the date of the first problem sequence.

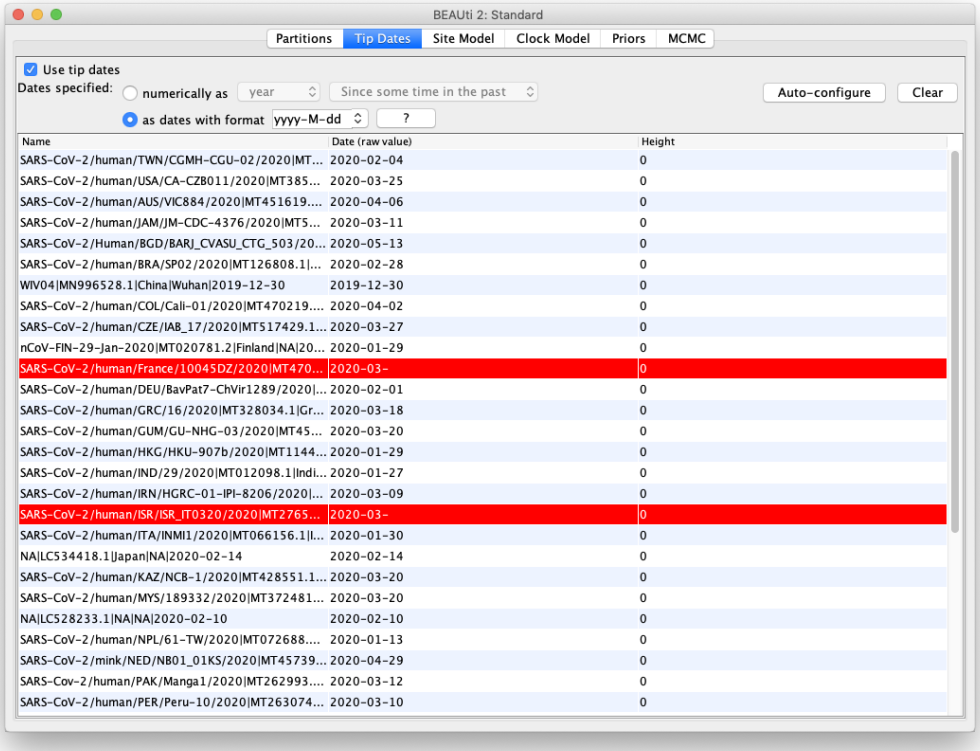

Figure 6: Problem sequences.

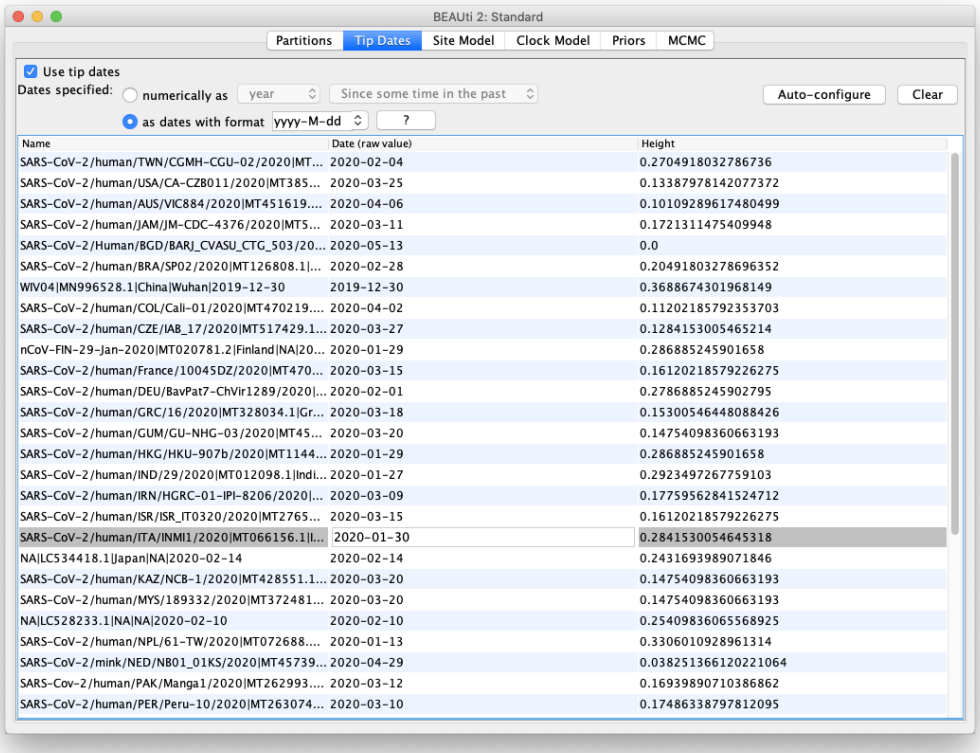

Figure 7: Tip dates panel with final setup.

Set the date to **2020-03-15** and press **Enter**. The error message will repeat.

Repear for the second problem sequence.

Now that dates have been set correctly for all sequences, BEAUti will convert them to heights which will be used in the tree. The final result should look like Figure 7.

3.1.4 Specifying the Site Model

Next, we need to set up the substitution model in the **Site Model** tab.

Navigate to the **Site Model** panel, where we can choose the model of nucleotide evolution that we want to assume to underly our dataset.

Our dataset is made of nucleotide sequences. There are four models of nucleotide evolution available in BEAUti2: **JC69, HKY, TN93** and **GTR**. The **JC69** model is the simplest evolutionary model. All the substitutions are assumed to happen at the same rate and all the bases are assumed to have identical frequencies, i.e. each base **A, C, G** and **T** is assumed to have an equilibrium frequency of 0.25. In the **HKY** model, the rate of transitions **A** $\leftrightarrow$ **G** and **C** $\leftrightarrow$ **T** is allowed to be different from the rate of transversions

A $\leftrightarrow$ **C, G** $\leftrightarrow$ **T**. Furthermore, the frequency of each base can be either "Estimated", "Empirical" or "All Equal". When we set the frequencies to "Estimated", the frequency of each base will be co-estimated as a parameter during the BEAST run. If we use "Empirical", base frequencies will be set to the frequencies of each base found in the alignment. Finally, if set to "All Equal", the base frequencies will be set to 0.25. The **TN93** model is slightly more complicated than **HKY**, by allowing for different rates of **A** $\leftrightarrow$ **G** and **C** $\leftrightarrow$ **T** transitions. Finally, the **GTR** model is the most general reversible model and allows for different substitution rates between each pair of nucleotides as well as different base frequencies, resulting in a total of 9 free parameters.

Topic for discussion: Which substitution model may be the most appropriate for our dataset and why?

Since we do not have any extra information on how the data evolved, the decision is not clear cut. The best would be to have some independent information on what model fits the Covid data the best. Alternatively, one could perform model comparison, or apply reversible jump MCMC (see for example the **bModelTest** and **substBMA** packages) to choose the best model. We will assume that we have done some independent data analyses and found the HKY model to fit our data the best. In general, this model captures the major biases that can arise in the analysis of nucleotide data.

Now we have to decide whether we want to assume all of the sites to have been subject to the same substitution rate or if we want to allow for the possibility that some sites are evolving faster than others. For this, we choose the number of gamma rate categories. This model scales the substitution rate by a factor, which is defined by a Gamma distribution. If we choose to split the Gamma distribution into 4 categories, we will have 4 possible scalings that will be applied to the substitution rate. The probability of a substitution at each site will be calculated under each scaled substitution rate (and corresponding transition probability matrix) and averaged over the 4 outcomes.

Topic for discussion: Do you think a model that assumes one rate for all the sites is preferable over a model which allows different substitution rates across sites (i.e. allows for several gamma rate categories)? Why or why not?

Once again, a proper model comparison, i.e. comparing a model without gamma rate heterogeneity to a model with some number of gamma rate categories, should ideally be done. We do not have any independent information on whether Gamma rate categories are needed or not. Thus, we take our best guess in order not to bias our analyses.

Let us therefore choose the HKY model with 4 gamma rate categories for the substitution rate.

Change the **Gamma Category Count** to 4, make sure that the estimate box next to the **Shape** parameter of the Gamma distribution is ticked and set **Subst Model** to **HKY**. Make sure that both **Kappa** (the transition/transversion rate ratio) and **Frequencies** are estimated. (Figure 8)

Notice that we estimate the shape parameter of the Gamma distribution as well. This is generally recommended, unless one is sure that the Gamma distribution with the shape parameter equal to 1 captures

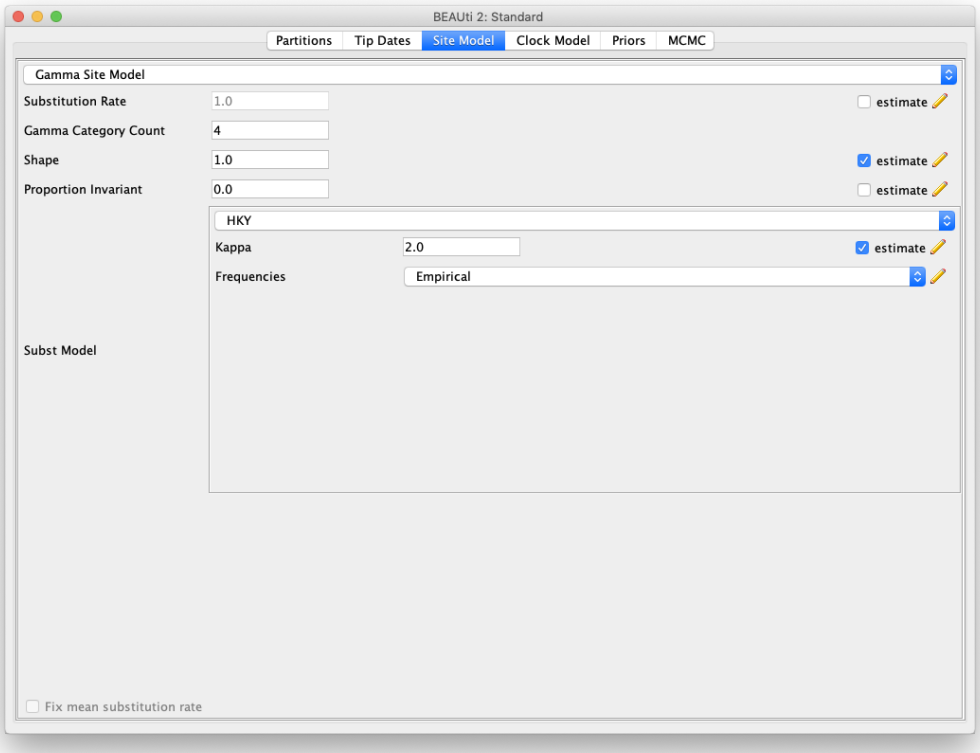

Figure 8: Specifying the substitution model.

exactly the rate variation in the given dataset. Notice also, that we leave the substitution rate fixed to 1.0 and do not estimate it. In fact, the overall substitution rate is the product of the clock rate and the substitution rate (one of the two acting as a scalar rather than a quantity measured in number of substitutions per site per time unit), and thus fixing one to 1.0 and estimating the other one allows for estimation of the overall rate of substitution. We will therefore use the clock rate to estimate the number of substitutions per site per year.

3.1.5 Specifying the Clock Model

Navigate to the **Clock Model** panel.

By default, four different clock models are available in BEAST2, allowing us to specify different models of lineage-specific substitution rate variation. The default model in BEAUti is the *Strict Clock*, which assumes a single fixed substitution rate across the whole tree. The other three models relax the assumption of a constant substitution rate. The *Relaxed Clock Log Normal* allows for the substitution rates associated with each branch to be independently drawn from a single, discretized log normal distribution (Drummond et al. 2006). Under the *Relaxed Clock Exponential* model, the rates associated with each branch are drawn from an exponential distribution (Drummond et al. 2006). Both of these models are uncorrelated relaxed clock models. The log normal distribution has the advantage that one can estimate its variance, which reflects the extent to which the molecular clock needs to be relaxed. In both models, BEAUti sets the **Number Of Discrete Rates** to -1 by default. This means that the number of bins that the distribution is divided into is equal to the number of branches. The last available model is the *Random Local Clock* which averages over all possible local clock models (Drummond and Suchard 2010).

Topic for discussion: Which clock model may be the most appropriate for our dataset and why? (SARS-CoV-2 gene sequences sampled over 3 months).

Since we are observing the sequence data from a single epidemic of SARS-CoV-2 virus in humans, we do not have any reason to assume different substitution rates for different lineages. Thus, the most straightforward option is to choose the default **Strict Clock** model (Figure 9). Note however, that a rigorous model comparison would be the best way to proceed with the choice of the clock model.

3.1.6 Specifying Priors

Navigate to the **Priors** panel.

We need to specify prior distributions for the:

- Tree
- Molecular clock model parameters
- Site model parameters

It is important to remember that a prior distribution is specified by the choice of distribution *and* the bounds we place on it.

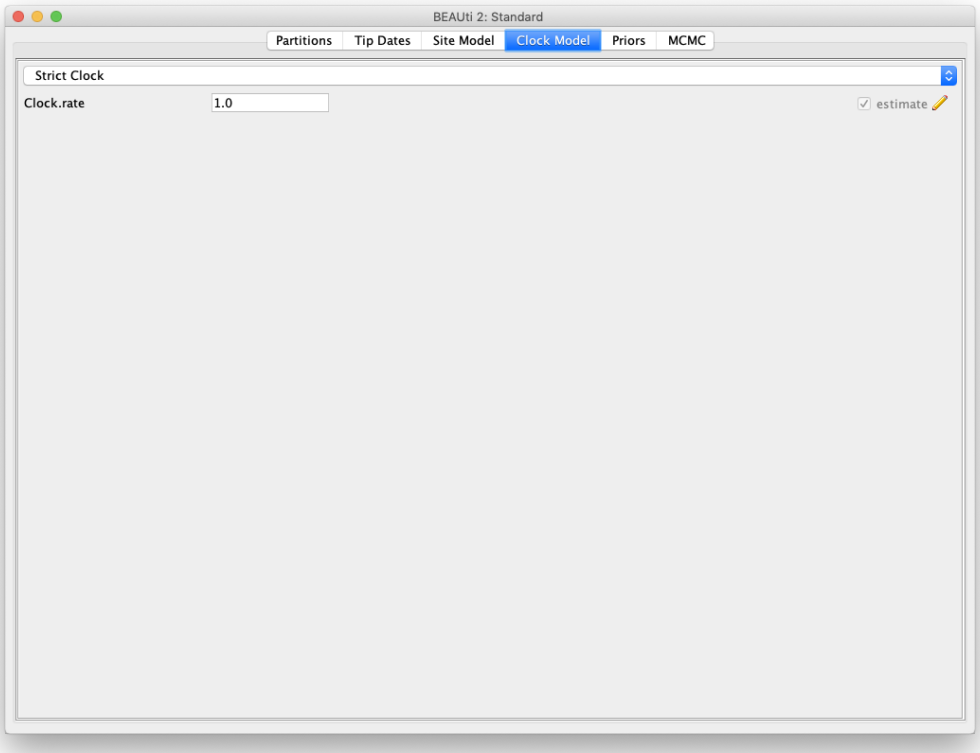

Figure 9: Specifying the clock model.

Tree prior

The choice of tree prior is quite tricky here, as the dynamics of the SARS-CoV-2 virus are likely to change over time due to public health measures and behaviour changes. On the other hand, our dataset is fairly small and contains sequences from very different parts of the world, so the amount of signal present may be too low to distinguish between phases of the epidemic. In the end, we choose a simple birth-death model of population dynamics with only one time interval for the reproductive number, R_e.

The birth-death model adds four additional hyperparameters, for which we in turn need to specify hyperpriors:

- The effective reproductive number, R_e
- The becoming uninfectious rate, δ
- The sampling proportion
- The origin time of the epidemic

In some cases we may fix some of the parameters of the birth-death model to external estimates, in which case we would not have to specify priors for them.

R_e is an important variable for the study of infectious diseases, since it defines the average number of secondary infections caused by an infected individual at a given time during the epidemic. In other words, it tells us how quickly the disease is spreading in a population. As long as R_e is above 1 the epidemic is likely to continue spreading, therefore prevention efforts aim to push R_e below 1. Note that as more people become infected and the susceptible population decreases, R_e will naturally decrease over the course of an epidemic, however treatment, vaccinations, quarantine and changes in behaviour can all contribute to decreasing R_e faster. In a birth-death process, R_e is defined as the ratio of the birth (or transmission) rate and the total death (or becoming non-infectious) rate. R_e for any infection is rarely above 10, so we set this as the upper value for R_e in our analysis.

For the **Tree** model, select the option **Birth Death Skyline Serial**.

Then, click on the arrow to the left of **reproductiveNumber** to open all the options for R_e settings (Figure 10). Leave all the settings on the default, since the default Log Normal prior is not too strong and is centered around 1. This is exactly what we want.

Then, click on the button where it says **initial = [2.0] [0.0, Infinity]**. A pop-up window will show up (Figure 11).

In the pop-up window change the **Upper**, the upper limit of the prior distribution, from Infinity to 10 and the **Dimension** of the R_e from 10 to 1 and click **OK**.

Notice that the pop-up window allows one to specify not only the **Dimension** but also the **Minordimension**. If the parameter is specified as a vector of n entries, we only use the **Dimension** with input n. If the parameter is specified as an $n \times m$ matrix, we then use the **Minordimension** to specify the number of columns (m) the parameter is split into. In the birth-death skyline model, we use the parameter vector only, and thus the **Minordimension** always stays specified as 1. (In fact, **Minordimension** is only used very rarely in any BEAST2 model).

After we have specified the prior for R_e, the next prior that needs our attention is the **becomeUnin-**

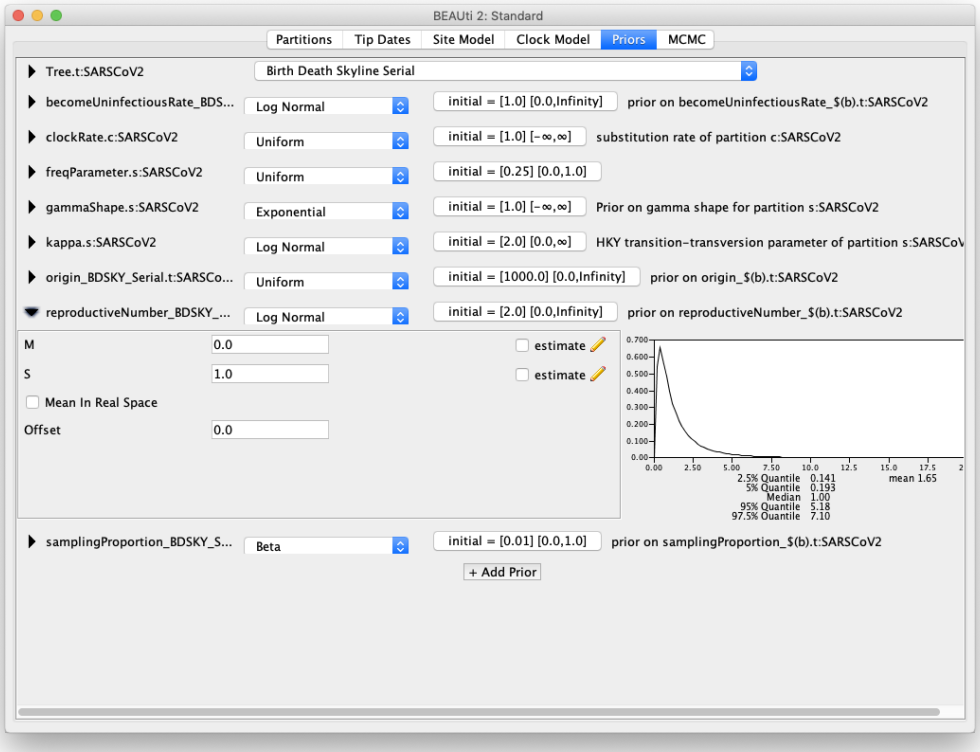

Figure 10: Specifying the tree prior.

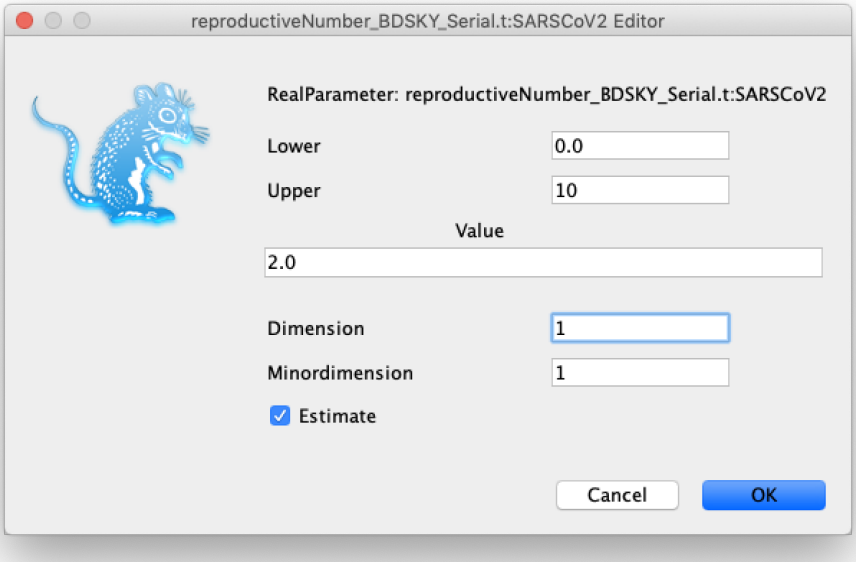

Figure 11: Specifying the 'R_e' prior.

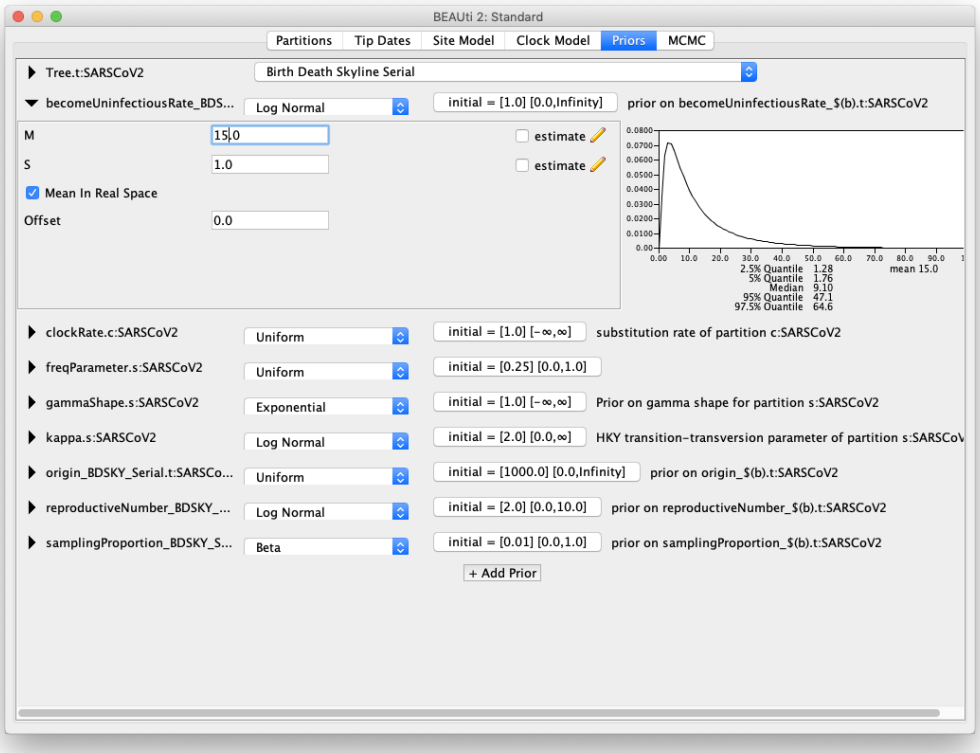

Figure 12: Specifying the becoming uninfectious rate prior.

fectiousRate. This specifies how quickly a person infected with COVID-19 recovers. From our personal experience, we would say that it takes around 3 to 4 weeks from infection to recovery (including around one week of incubation period). Since the rate of becoming uninfectious is the reciprocal of the period of infectiousness this translates to a becoming uninfectious rate of $365/30 \approx 12.2$ to $365/21 \approx 17.4$ per year (recall that we specified dates in our tree in years, and not days). Let us set the prior for **becomeUninfectiousRate** rate accordingly.

> Click on the arrow next to **becomeUninfectiousRate** and change the value for **M** (mean) of the default log normal distribution to 15 and tick the box **Mean In Real Space** which allows us to specify the mean of the distribution in real space (Figure 12).

Looking at the 2.5% and 97.5% quantiles for the distribution we see that 95% of the weight of our becoming uninfectious rate prior falls between 1.28 and 64.6, i.e. our prior on the period of infectiousness is between ≈ 5.65 and 285 days. Thus, our prior is quite diffuse. If we wanted to use a more specific prior we could decrease the standard deviation of the distribution (the **S** parameter).

For the next parameter, the sampling proportion, we know that we certainly did not sample every single infected individual. Therefore, setting a prior close to 1 would not be reasonable. Considering the size of our dataset, we expect only a proportion of less than 0.1% of all COVID cases to have been sampled. Here, we specify something on the order of 10^{-3}. The default prior for the sampling proportion is a Beta

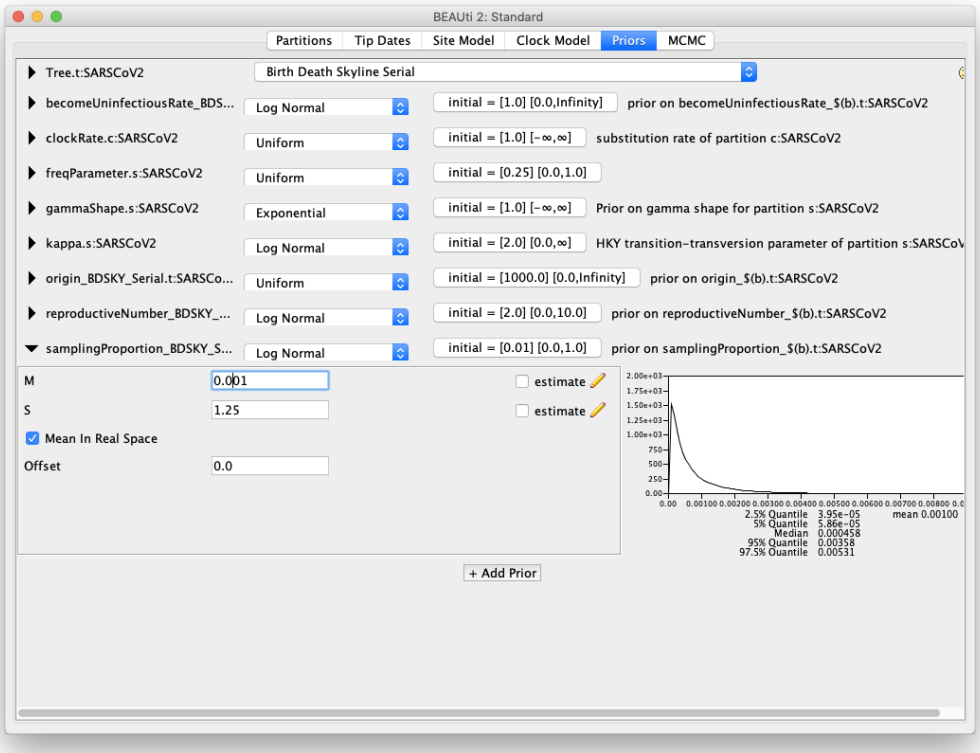

Figure 13: Specifying the sampling proportion prior.

distribution, which is only defined between 0 and 1, making it a natural choice for proportions. However, this is not the only prior that can be used, and here we specify a log-normal distribution, while ensuring that an appropriate upper limit is set, to prevent a sampling proportion higher than 1, which is invalid.

Click on the arrow next to the **samplingProportion** and change the distribution from **Beta** to **Log Normal**.

Next, change the value for the **M** (mean) to 0.001 and tick the box **Mean In Real Space** (Figure 13).

Also, make sure that the **Lower** is set to 0.0 and the **Upper** is set to 1.0.

Lastly, for the origin of the epidemic, we ask ourselves whether there is any reasonable expectation we might have in terms of when the infection started, i.e. what is the date when the ancestor of all of the sequences first appeared.

Topic for discussion: Do you have any feeling for what the origin should/could be set to?

The data span a period of 4 months and the epidemic is expected to have started in late 2019. The best guess for the origin parameter prior we could make is therefore on the order of at least 4, but probably no

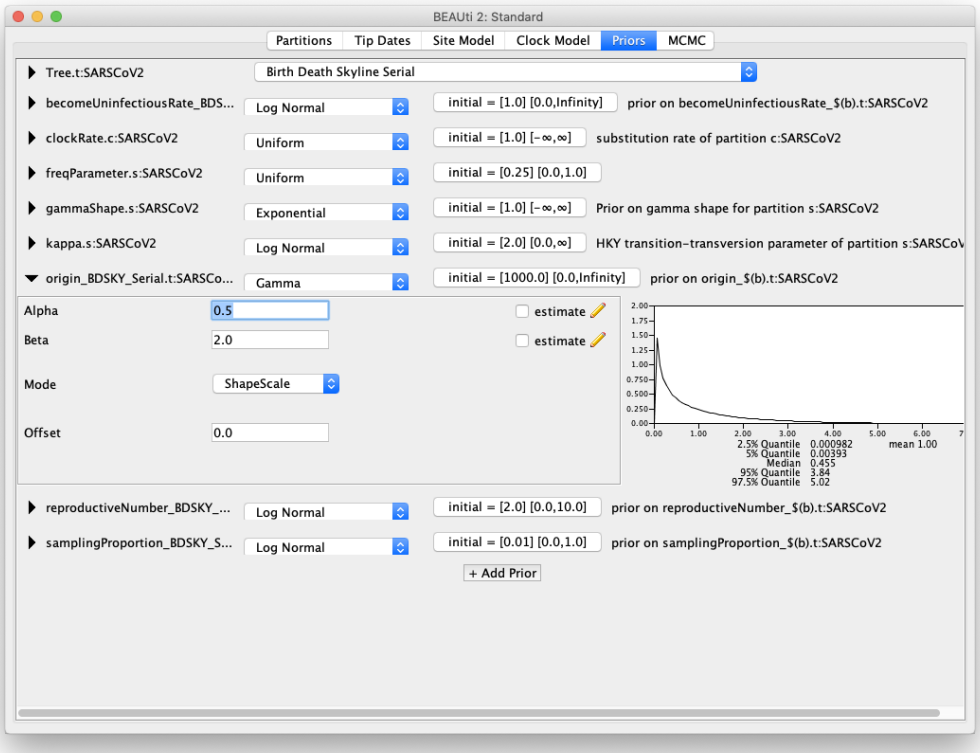

Figure 14: Specifying the origin prior.

more than 6 months. We set the prior according to this expectation. (Remember that branch lengths are measured in years).

> Click on the arrow next to the **origin** and change the prior distribution from **Uniform** to **Gamma** with **Alpha** parameter set to 0.5 and **Beta** parameter set to 2.0 (Figure 14).

Molecular clock model

We are using a strict clock model, which has only one parameter, the clock rate. This is the substitution rate, measured in substitutions per site per year (s/s/y).

> **Topic for discussion:** What substitution rate is appropriate for viruses? More specifically, what substitution rate is expected for SARS-CoV-2 genes, in your opinion?

By default, the clock rate in BEAST2 has a uniform prior between 0 and infinity. This is not only extremely unspecific, but also an improper prior (it does not integrate to 1). In general, a log-normal distribution works well for rates, since it does not allow negative values. Furthermore, it places most weight close to 0, while also allowing for larger values, making it an appropriate prior for the clock rate, which we expect to be quite low in general, but may be higher in exceptional cases. You could set your best guess as a prior

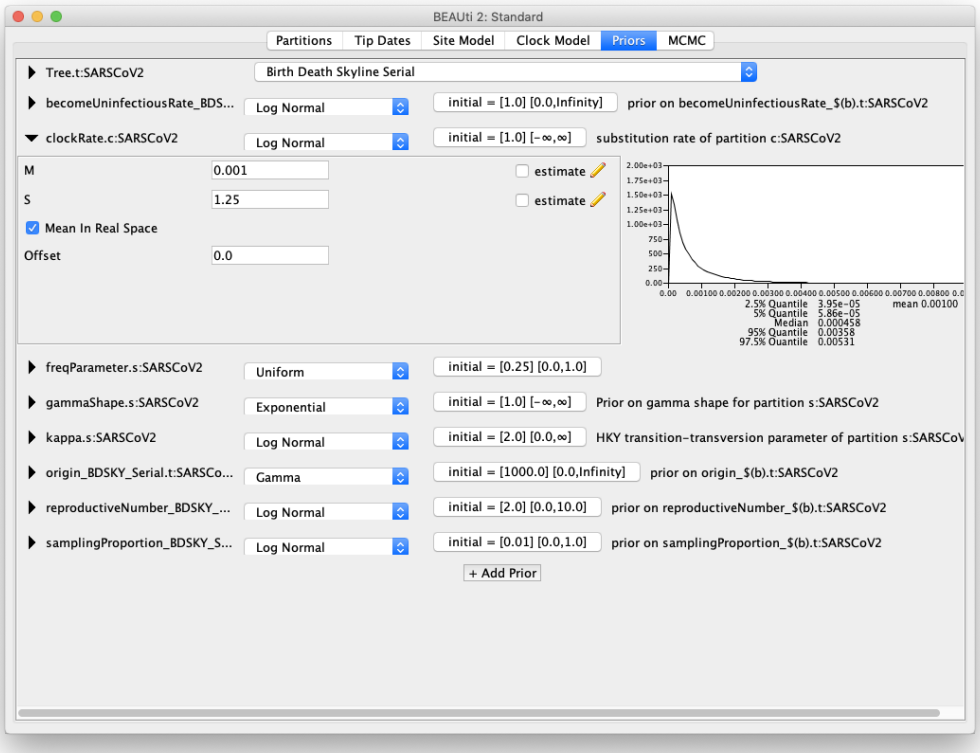

Figure 15: Specifying the clock rate prior.

by, for example, choosing a log-normal distribution centered around your best guess for the substitution rate.

Now consider the following information: SARS-CoV-2 virus is an RNA virus (Kawaoka 2006) and RNA viruses in general, have a mutation rate of $\approx 10^{-3}$ substitutions per site per year (Jenkins et al. 2002).

> **Topic for discussion:** Did you change your best guess, for the substitution rate appropriate for RNA viruses? What would it be? How would you specify the prior?

Our best guess would be to set the prior distribution peaked around 10^{-3} substitutions per site per year.

> Change the prior for the clock rate from a **Uniform** to **Log Normal** distribution. Click on the arrow next to the **clockRate** and change the value for **M** (mean) of the default log normal distribution to 0.001 and tick the box **Mean In Real Space** (Figure 15).

Site model

We used an HKY model, with Gamma-distributed rate heterogeneity with 4 categories and estimated equilibrium frequencies. Thus, we need to set priors for three parameters:

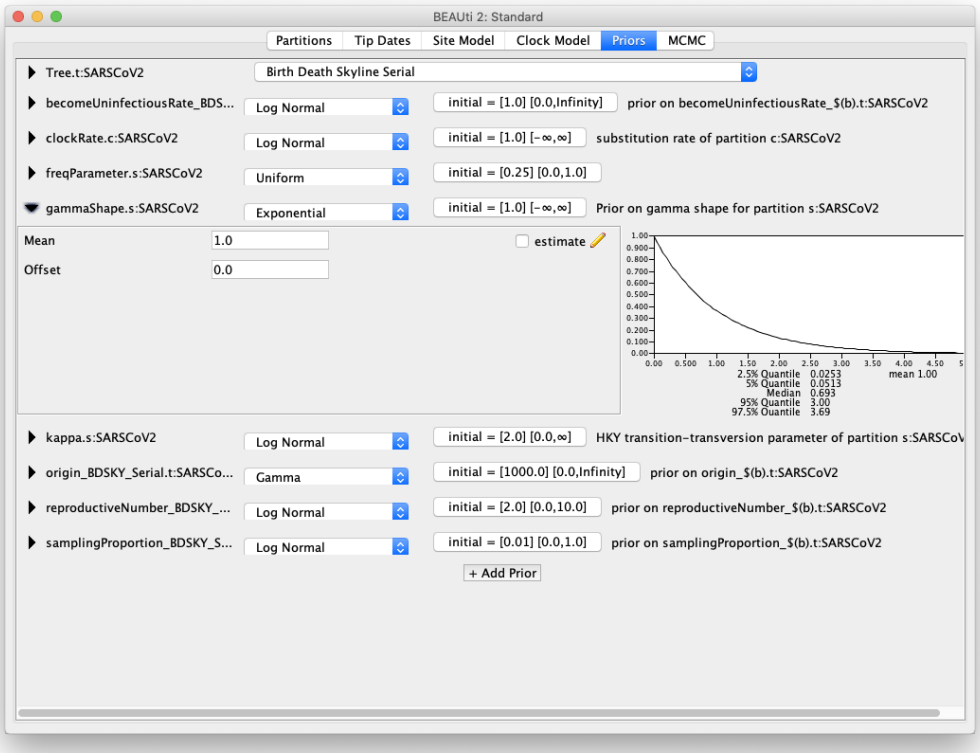

Figure 16: Specifying the gamma shape prior.

- The Gamma shape parameter, α
- The transition/transversion rate ratio, κ
- The equilibrium nucleotide frequencies (actually 4 parameters)

(The default priors for site models perform well in most scenarios and in practice rarely have to be changed. However, it is important not to forgot about them!)

The Gamma shape parameter governs the shape of the Gamma distribution of the rates across different sites. The default setting of the Gamma shape parameter of **alpha=beta=1.0** reflects our belief that on average, the rate scaler is equal to 1, i.e. on average all the sites mutate with the same substitution rate. The distribution on the gamma shape parameter allows us to deviate from this assumption. The default exponential distribution with **M** (mean) of 1.0 and 95%HPD of [0.0253,3.69] covers a wide range of possible shape parameters. This looks fine for our analysis, and thus, we leave the Gamma shape settings at its defaults (Figure 16).

We do not have any prior information on transition-transversion rate ratio besides the fact that it is a value usually larger than 1 (transitions are more frequent than transversions). We therefore set a weakly informative prior for this parameter. The default log normal prior perfectly fits to these requirements and usually does not need to be changed (Figure 17).

By default, BEAST2 sets a uniform prior between 0 and 1 for each of the equilibrium nucleotide frequencies (i.e. a uniform prior defined on the entire range of the parameter). It is rarely necessary to specify a strong

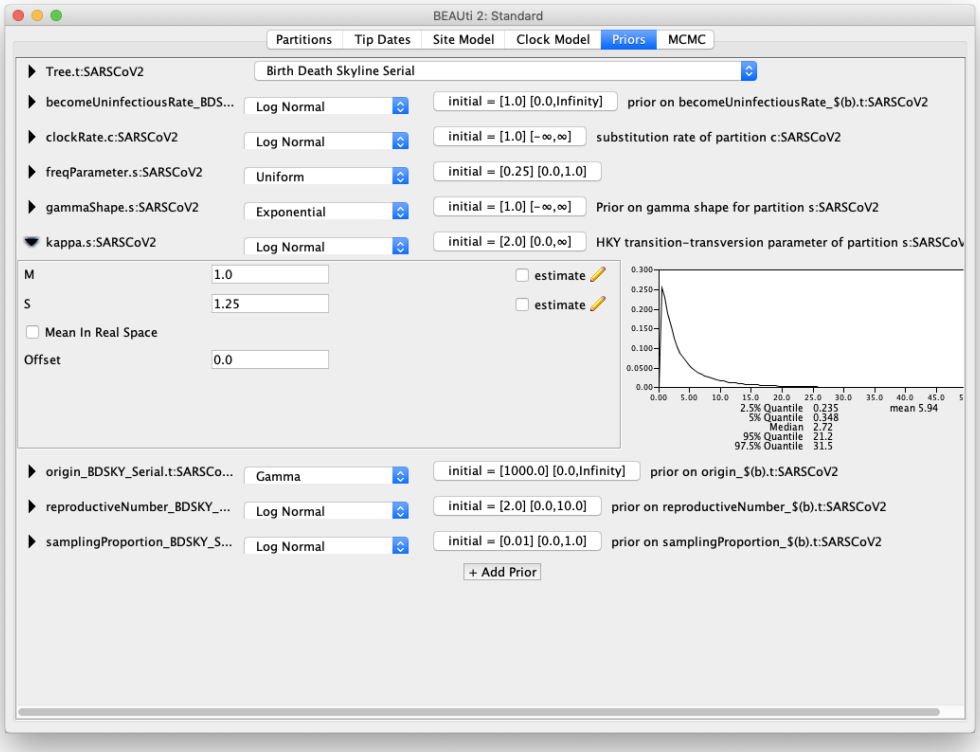

Figure 17: Specifying the kappa (transition/transversion ratio) prior.

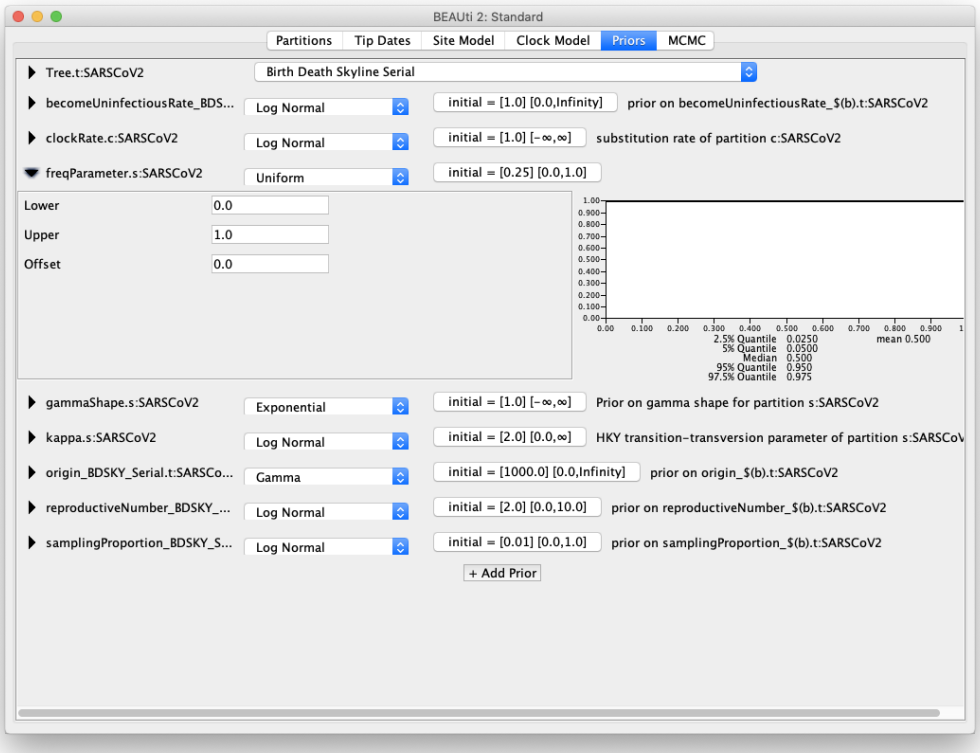

Figure 18: Specifying the equilibrium nucleotide frequencies prior.

prior for equilibrium frequencies. Equilibrium frequencies are usually easy to infer from the data and estimates do not have a large effect on other parameters. Thus, we can leave the prior as is (Figure 18).

Tip ages priors

When we imported our sequence alignment into BEAUti and set the sequence ages, we noticed that two sequences did not have the day of sampling specified, only the month and the year. We decided to set both sequences to the 15th of the month, but in reality we have no information to decide which day is most likely. Since our dataset only covers 4 months of sampling, fixing the wrong sampling ages for some sequences could bias our inference. A more rigorous strategy is to account for this uncertainty in the sampling date in the inference, which can be done by adding prior distributions on the age of these sequences.

To add an extra prior to the model, press the + **Add Prior** button below list of priors and select **MRCA Prior** from the drop-down menu.

You will see a dialogue box that allows you to select a subset of taxa from the phylogenetic tree. Once you have created a taxon set you will be able to add age information for its most recent common ancestor (MRCA) later on.

Set the **Taxon set label** to `MT470123.1`. Locate `SARS-CoV-2/human/France/10045DZ/2020|MT470123.1|France|NA`↩ `|2020-03-` in the left hand side list and click the » button to add it to this taxon set.

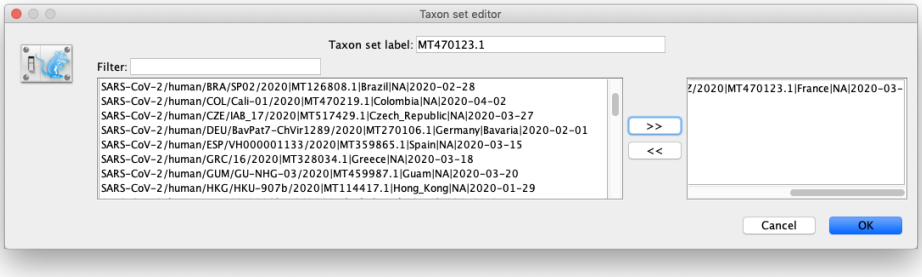

Figure 19: Taxon set containing the first sequence with no sampling day.

The taxon set should now look like Figure 19.

Click the **OK** button to add the newly defined taxon set to the prior list.

We now need to specify a prior distribution on this sequence age. We know that it was sampled between the 1st and 30th of March 2020, which corresponds to an age (in years) between 2020.162 and 2020.244. Since we have no additional information, we will set the age to a Uniform distribution between those two bounds.

Select **Uniform** from the drop-down menu to the right of the newly added **MT470123.1.prior**.

Expand the distribution options using the arrow button on the left.

Set the **Lower** of the distribution to **2020.162**.

Set the **Upper** of the distribution to **2020.244**.

Check the **Tipsonly** checkbox to indicate this prior is on the tip age.

The final setup of our new prior should look as shown in Figure 20.

The only thing left is to repeat the process for the second sequence with uncertain age, `SARS-CoV-2/human/ISR/ISR_IT0320/2020|MT276598.1|Israel|NA|2020-03-`. This sequence was also sampled in March 2020, so we can use the same prior distribution for it.

The final prior setup should look as shown in Figure 21.

3.1.7 MCMC

Navigate to the **MCMC** panel.

We want to shorten the chain length, in order for it to run in a reasonable time and we want to decrease the tree sampling frequency.

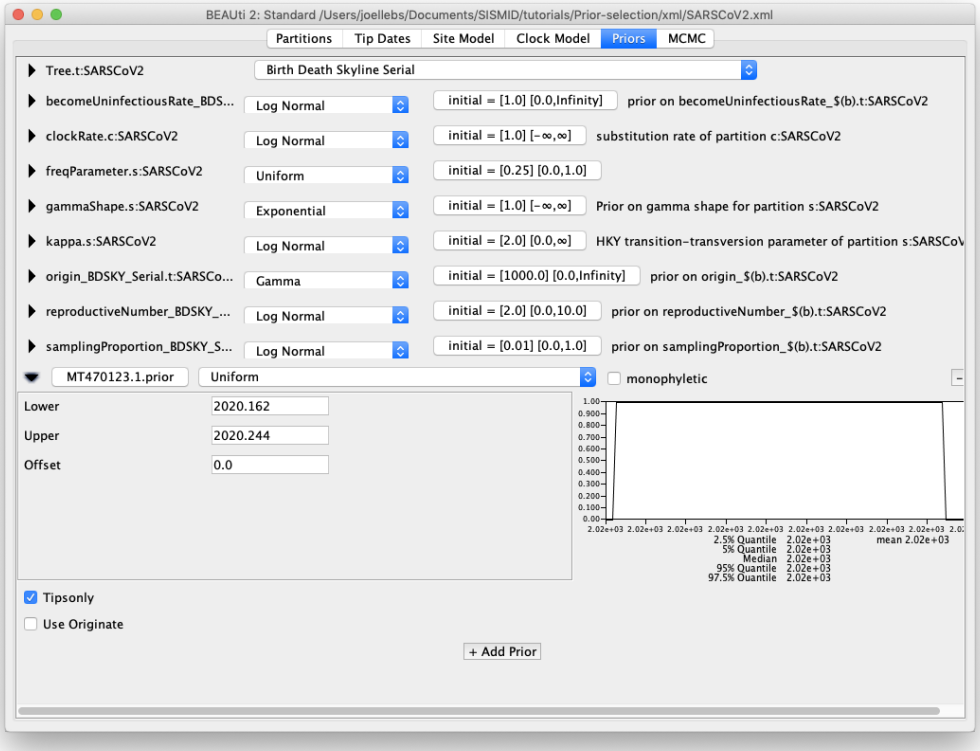

Figure 20: Tip prior setup.

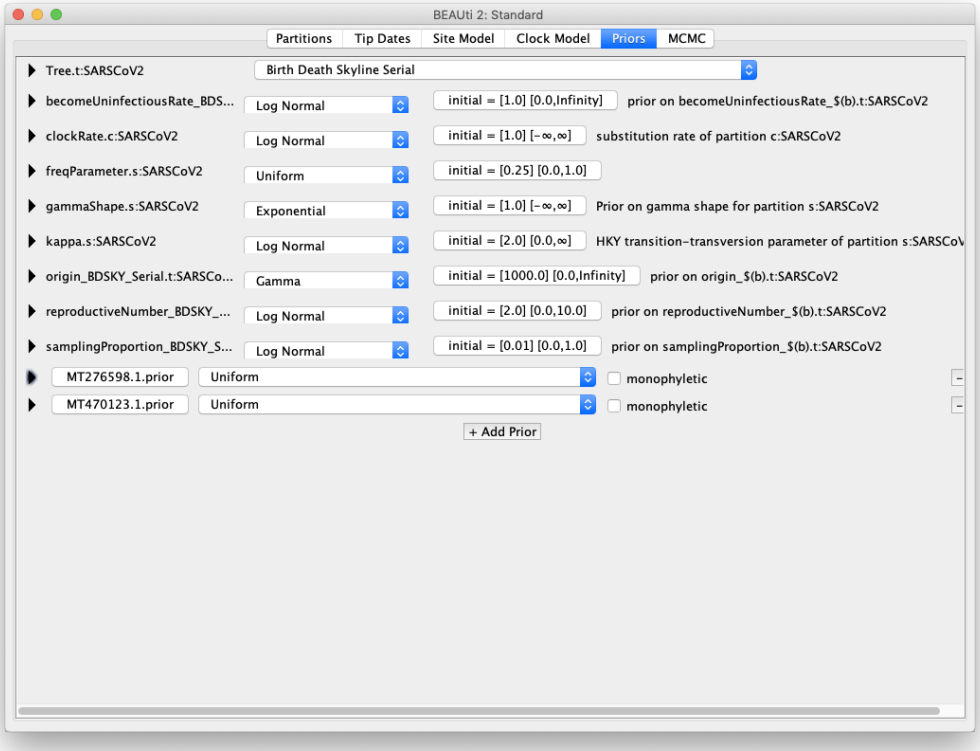

Figure 21: Final prior setup.

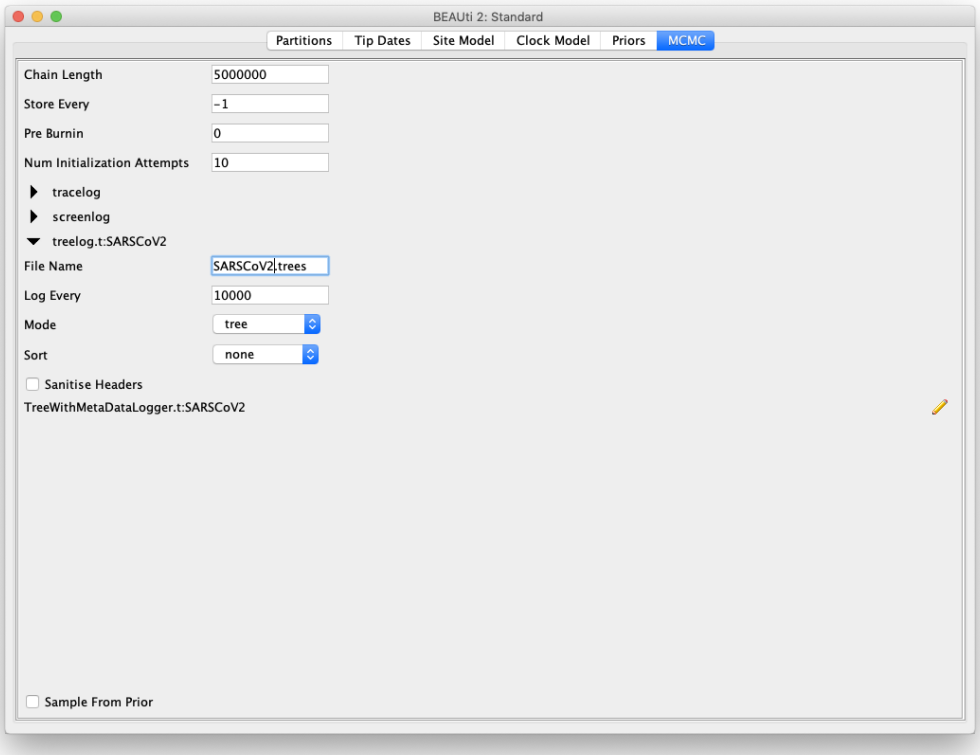

Figure 22: Specifying the MCMC properties.

Change the **Chain Length** from 10'000'000 to 5'000'000.

Click on the arrow next to the **treelog** and set the **Log Every** to 10'000 (Figure 22). Set the **File Name** to `SARSCoV2.trees`.

Now, all the specifications are done. We want to save and run the XML.

Save the XML file as `SARSCov2.xml`.

3.1.8 Running the analysis

Start **BEAST2** and choose the file `SARSCov2.xml`.

If you have **BEAGLE** installed tick the box to **Use BEAGLE library if available**, which will make the run faster.

Hit **Run** to start the analysis.

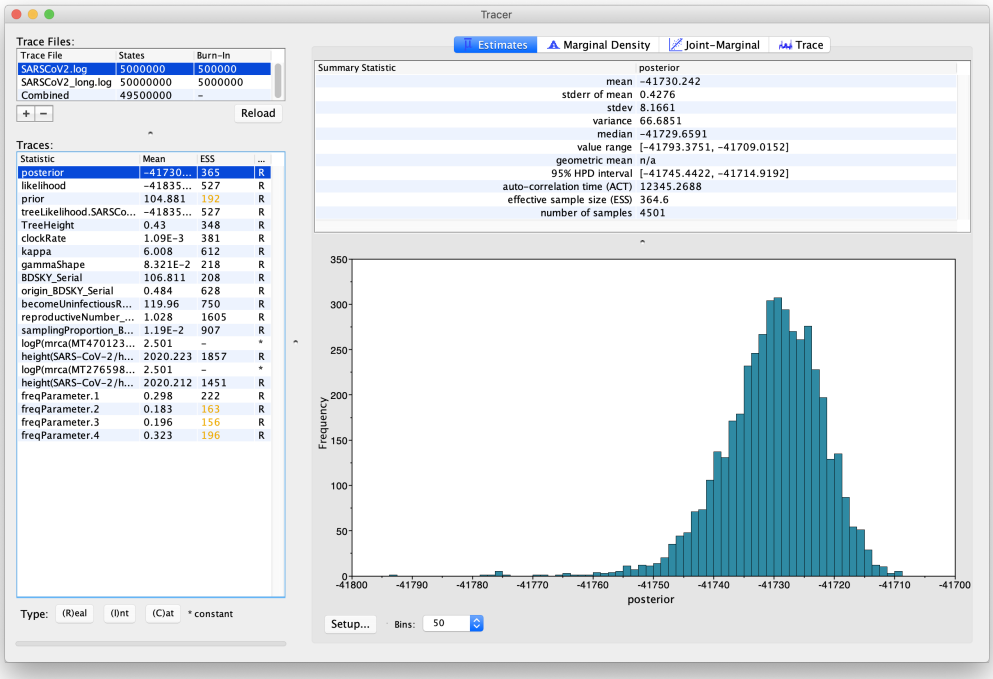

Figure 23: Loading the log file into Tracer.

3.1.9 Analysing the results

Load the logfile into **Tracer** to check mixing and the parameter estimates.

The first thing you may notice is that some of the parameters have low ESS (effective sample size below 200) marked in yellow (Figure 23). This is because our chain did not run long enough. However, the estimates we obtained with a chain of length 5'000'000 are very similar to those obtained with a longer chain.

Click on **clockRate** and then click on **Trace** to examine the trace of the parameter (Figure 24).

Note that the chain appears to have passed the burn-in phase and seems to be sampling from across the posterior without getting stuck in any local optima. You should always examine the parameter traces to check convergence; a high ESS value is not proof that a run has converged to the true posterior.

In the following we show the results we obtained with identical settings and a chain of 50'000'000 iterations.

Examine the posterior estimates for the **becomeUninfectiousRate**, **samplingProportion** and **clockRate** in Tracer. Do the estimates look realistic? Are they different from the priors we set and if so, how?

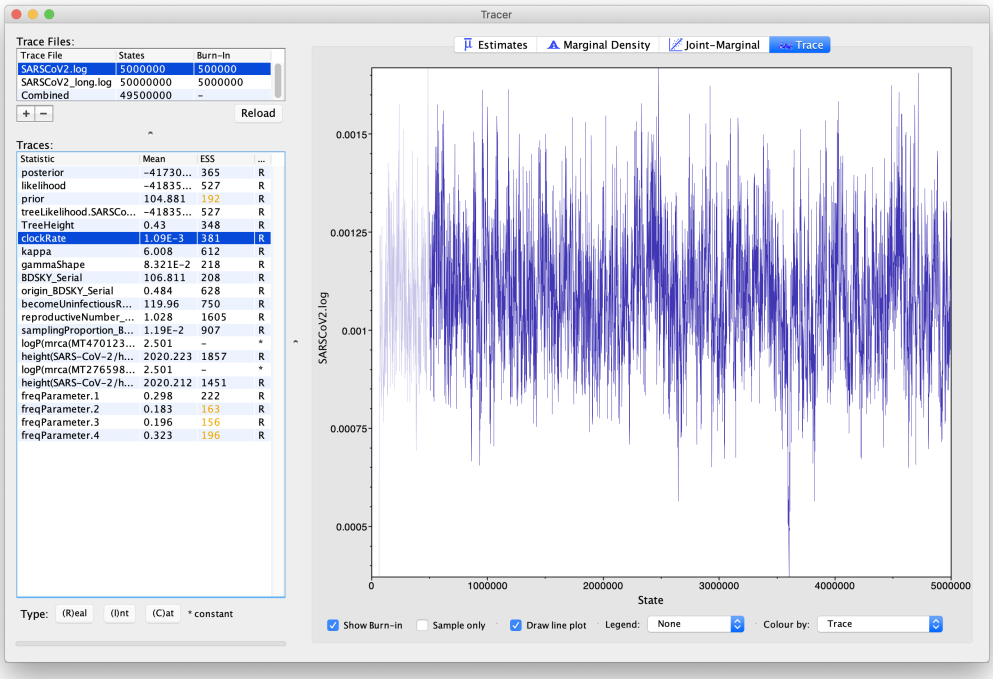

Figure 24: The trace of the clock rate parameter.

The estimated posterior distribution for the **becomeUninfectiousRate** has a median of 103.54 and a 95% HPD between 19.89 and 258.22 (Figure 25), which corresponds to an infectious period of between ≈ 1.41 and 18.35 days. This is a lot more specific than the prior we set, which allowed for a much longer infectious period. In this case there was enough information in the sequencing data to estimate a more specific becoming uninfectious rate. If we had relied more on our prior knowledge we could have set a tighter prior on the **becomeUninfectiousRate** parameter, which may have helped the run to converge faster, by preventing it from sampling unrealistic parameter values. However, if you are unsure about a parameter it is always better to set more diffuse priors.

We see that the sampling proportion (Figure 26) is estimated to be 6.82×10^{-3}. This is quite close the mean we set for the prior on the sampling proportion (0.001). Therefore we can question if this estimate is driven by the data or informed purely by our prior. One good way to check is to rerun the analysis without the sequences, by checking the **Sample From Prior** option in the **MCMC** tab in BEAUti.

Comparing the posterior distributions obtained with and without the sequences for the sampling proportion (Figure 27)shows that the distributions are indeed quite close. This indicates that there is little signal in our dataset to inform the sampling proportion estimate, and that it is mostly driven by the prior rather than the data. Since priors play an important role in Bayesian inference, comparing the results of your inference with an inference sampling from the prior is generally good practice.

Finally, we can look at the estimated age of the two sequences for which we decided to infer the sampling date (Figure 28). We can see that for both sequences, the estimated sampling date is towards the end of the month rather than the beginning, so our initial fixed date of March 15th would have been too early, especially for sequence `MT470123.1`.

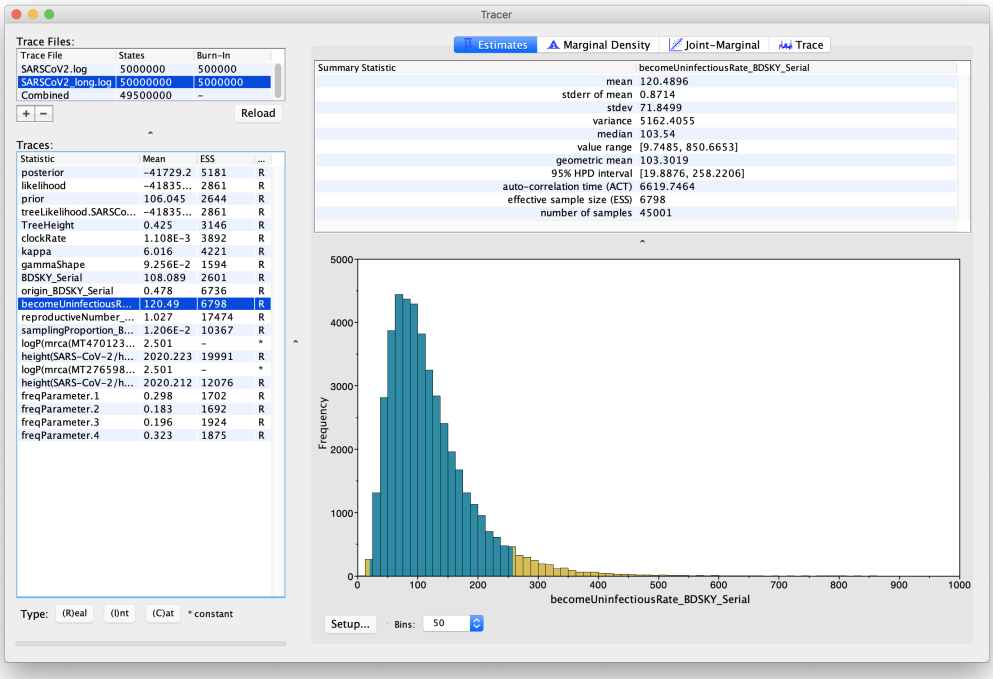

Figure 25: Estimated posterior distribution for the becoming uninfectious rate.

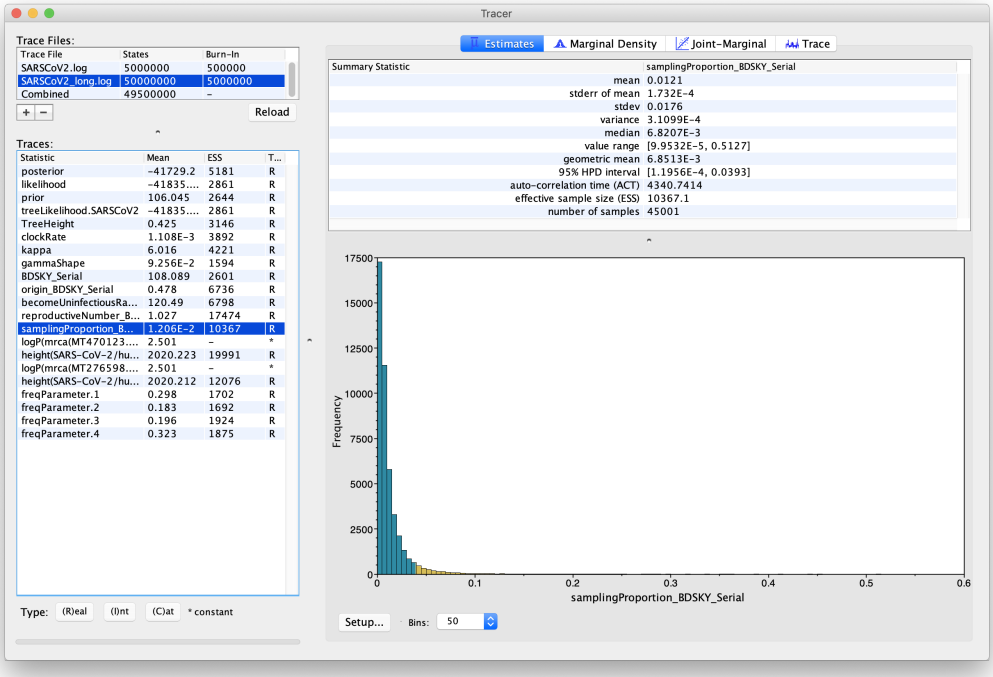

Figure 26: Estimated posterior distribution for the sampling proportion.

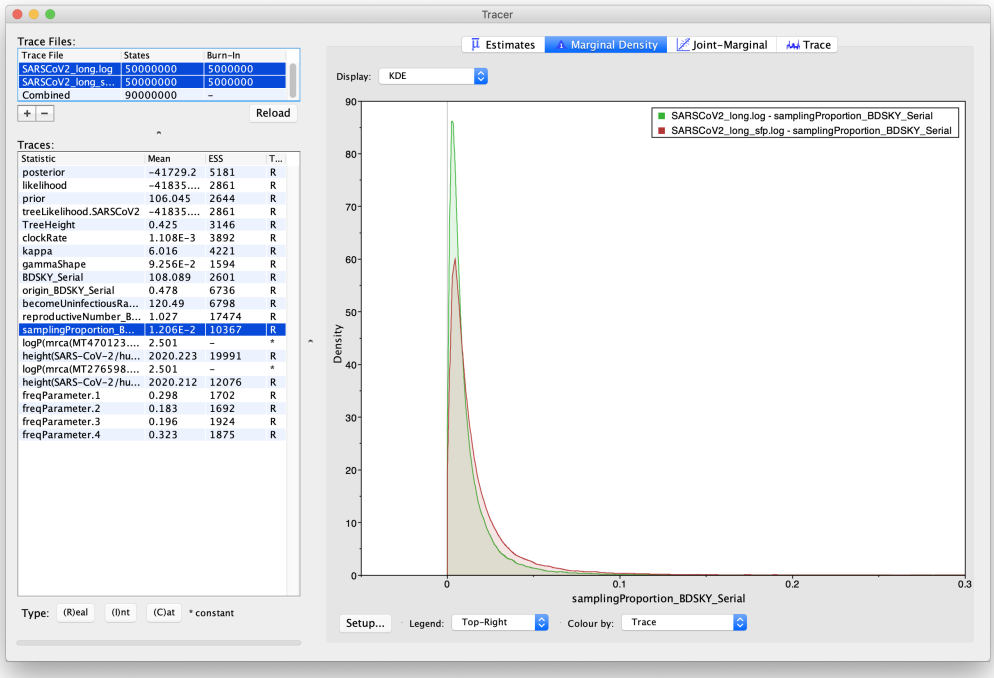

Figure 27: Estimated posterior distribution for the sampling proportion, with sequences (green) and without (red).

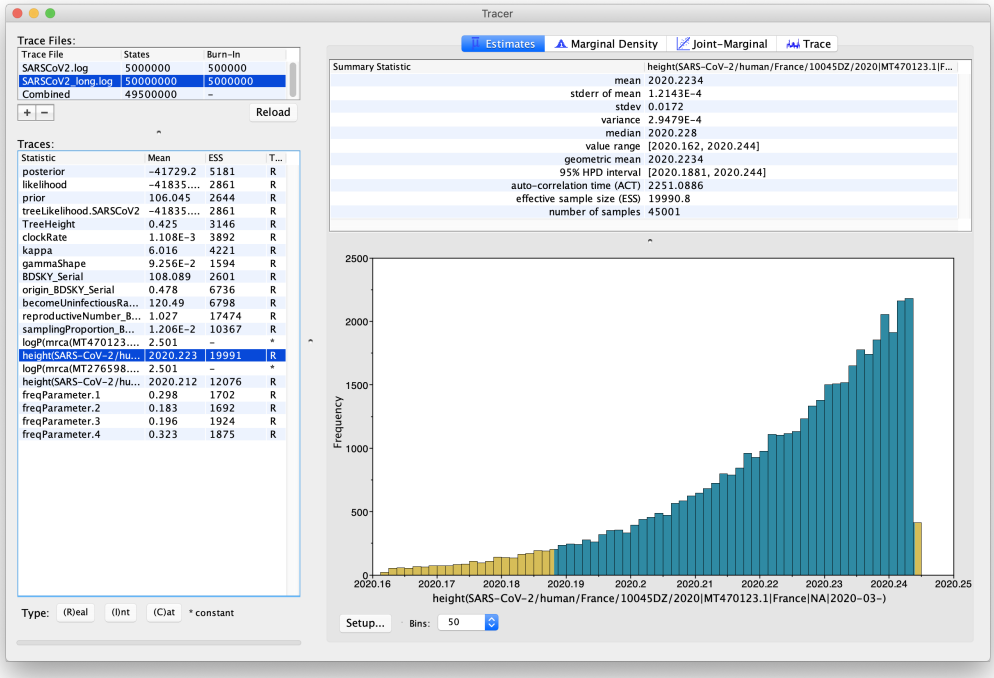

Figure 28: Estimated posterior distribution for the height of sequence MT470123.1.

4 Useful Links

- Bayesian Evolutionary Analysis with BEAST 2 (Drummond and Bouckaert 2014)
- BEAST 2 website and documentation: http://www.beast2.org/
- BEAST 1 website and documentation: http://beast.bio.ed.ac.uk
- Join the BEAST user discussion: http://groups.google.com/group/beast-users


This tutorial was written by Veronika Bošková, Venelin Mitov and Louis du Plessis for Taming the BEAST and is licensed under a Creative Commons Attribution 4.0 International License.

Version dated: July 20, 2020

Relevant References

Bouckaert, R, J Heled, D Kühnert, T Vaughan, CH Wu, D Xie, MA Suchard, A Rambaut, and AJ Drummond. 2014. Beast 2: a software platform for Bayesian evolutionary analysis. *PLoS computational biology* 10: e1003537.

Bouckaert, R et al. 2019. Beast 2.5: an advanced software platform for bayesian evolutionary analysis. *PLOS Computational Biology* 15:

Drummond, AJ, SY Ho, MJ Phillips, and A Rambaut. 2006. Relaxed phylogenetics and dating with confidence. *PLoS Biology* 4: e88.

Drummond, AJ and MA Suchard. 2010. Bayesian random local clocks, or one rate to rule them all. *BMC Biology* 8: 114.

Drummond, AJ and RR Bouckaert. 2014. *Bayesian evolutionary analysis with BEAST 2*. Cambridge University Press,

Jenkins, GM, A Rambaut, OG Pybus, and EC Holmes. 2002. Rates of molecular evolution in rna viruses: a quantitative phylogenetic analysis. *Journal of molecular evolution* 54: 156–165.

Kawaoka, Y. 2006. *Influenza virology: current topics*. Horizon Scientific Press,

Stadler, T, D Kühnert, S Bonhoeffer, and AJ Drummond. 2013. Birth–death skyline plot reveals temporal changes of epidemic spread in HIV and hepatitis C virus (HCV). *Proceedings of the National Academy of Sciences* 110: 228–233.