

BEAST lab

Genomics of Disease in Wildlife Workshop
June 8, 2023

Activities:

1. We will perform a discrete trait analysis with a skyride model on equine+canine H3N8
2. We will perform a local clock analysis to compare whether there is a difference in evolutionary rate between these two parts of the phylogeny

Outline/background:

This activity/lab builds on the phylogenies we built in Nextstrain. In the 1960s, an H3N8 influenza virus was detected in horses in Florida. This lineage then transmitted between horses, resulting in a spillover into dogs. This lineage circulated in dogs for several years before dying out recently. In this tutorial, we will learn how to use BEAST to infer when H3N8 was transmitted from horses to dogs, how rapidly the virus population grew during transmission in horses and dogs, and measure whether the receptor binding protein (HA) is evolving at different rates in horses and dogs. We will evaluate run convergence and mixing using Tracer, plot posterior distributions, and generate a summary tree.

For both analyses in this lab, we will be using BEAST v.1.10. BEAST 1 and BEAST 2 are actually quite different pieces of software, and include different arrays of models. BEAST is developed by a large group of highly collaborative investigators, who specialize in different types of models. These investigators often stick with development in either BEAST 1 or BEAST 2, and as a result, the interaction interface and organization, is somewhat distinct between the two. For discrete trait analyses (phylogeography), I usually use BEAST 1. For anything involving structured coalescent models or birth-death models, I always use BEAST 2. Both BEASTs include a complementary package called Beauti that acts as a user interface to make BEAST input files. We will be using Beauti to generate our input files, which we will then run in BEAST. We will use Tracer to evaluate chain convergence, and TreeAnnotator to generate summary trees.

Input data:

The input data for both analyses is an alignment of publicly available influenza HA gene segments sampled globally. These data have been organized with metadata describing the date of sample collection, subtype, geographic region and country of sampling, and host group. These sequences were then aligned with MAFFT.

1. First, take a look at the input alignment. It is always a good idea to look at your alignment before starting a BEAST analysis. If there are issues with your alignment, you will likely have problems with BEAST as well. To take a look, open SeaView, and drag and drop your aligned fasta file, `h3nx_canine_equine_ha.fasta` into it.

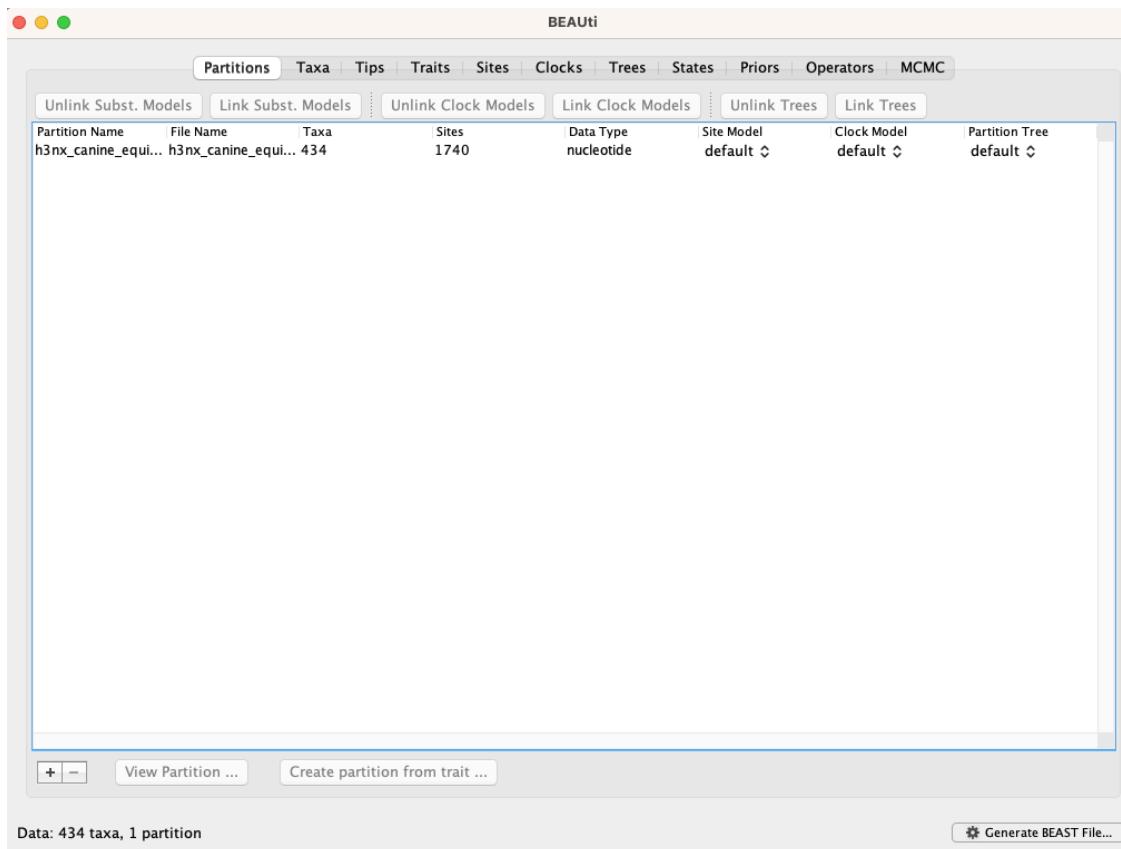
Analysis 1: Discrete traits + skyline model

In this analysis, we will estimate a time-resolved phylogeny using a skyride coalescent model. This model allows the population size to change over time, and both infers the appropriate number of intervals over which the population size has changed, and infers the population size across those intervals. The number of intervals is set to the number of taxa, such that there will be that many number of coalescent intervals for all of the taxa to coalesce. The model will use the time that lineages take to coalesce to estimate these population sizes, which we will summarize and can interpret as the degree of viral transmission intensity over time. In addition to estimating population size changes over time, we will also perform two reconstructions on the tree using a “discrete trait” approach. This approach is analogous to the reconstructions we performed in the Nextstrain lab to color internal nodes by host and geographic regions. We will perform the same analysis now in BEAST, and

compare results. BEAST will also output an estimate of the frequency of transmission events between groups, i.e., hosts or geographic regions. We will interpret these results as the rate of transmission between geographic regions, and between hosts. Finally, by visualizing these results on a phylogenetic tree, we can infer the timing of these events, i.e., the date at which H3N8 was transmitted from horses to dogs.

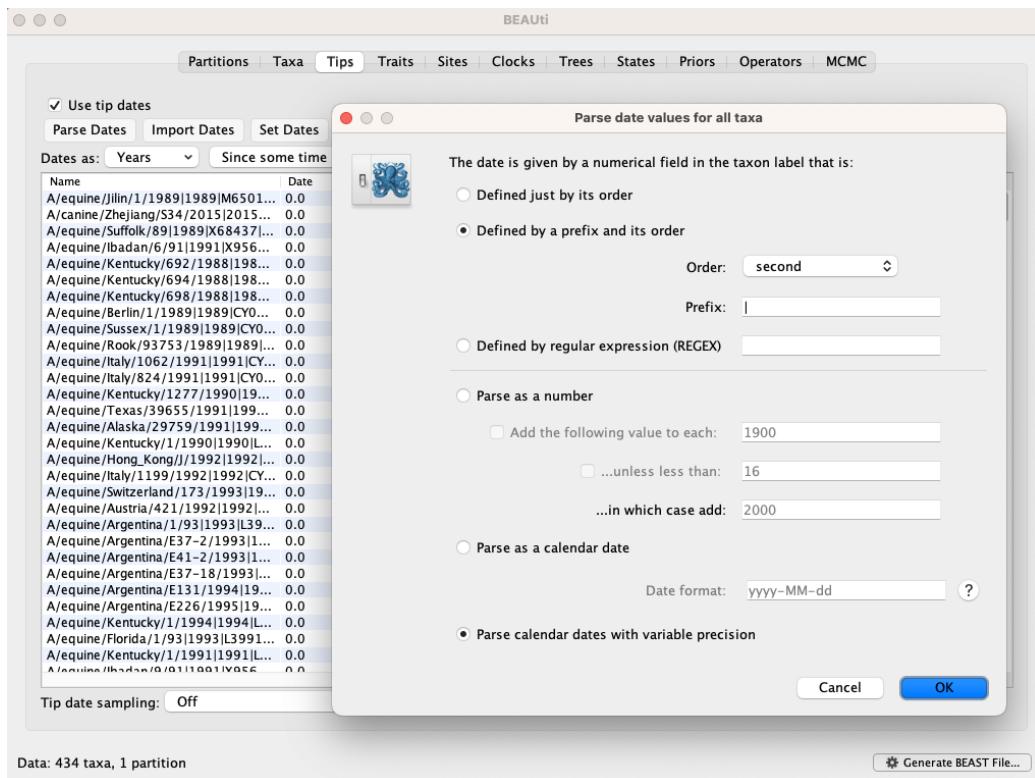
Part 1: Setting up the analysis

1. To get started, open Beauti by double clicking on it. To load your alignment into Beauti, drag your xml onto the main “Partitions” screen. It should load the alignment, and look like this:



You can see here that there are 434 sequences in this alignment (here, called “taxa”), and that the alignment is made of nucleotides.

2. Next, we will specify our tip dates. Click on the “Tips” tab and select “use tip dates.” BEAST allows you to either import tip dates separately, or to “guess” dates from information in your sequence headers. Our data has tip dates encoded in the sequence headers, so we can select that option. Select “Parse Dates”, and in the new pop-up window, select the options shown below. Parsing calendar dates with variable precision tells BEAST that we have some dates for which we have complete collection dates (including, year, month, and day), and others for which we only have a subset of information (usually just year, or year and month). Clicking this option tells BEAST to interpret the dates that are incomplete as less precise than complete dates.



Click on “Ok”, and the dates should now be auto-filled in the “Date” column. Notice that for samples with complete collection dates, the Uncertainty is set to 0.0, whereas samples with year-only collection dates have Uncertainty of 1.0, meaning 1 full year of uncertainty.

BEAUti

Partitions | Taxa | Tips | Traits | Sites | Clocks | Trees | States | Priors | Operators | MCMC

Use tip dates

Parse Dates | Import Dates | Set Dates | Clear Dates | Set Uncertainty

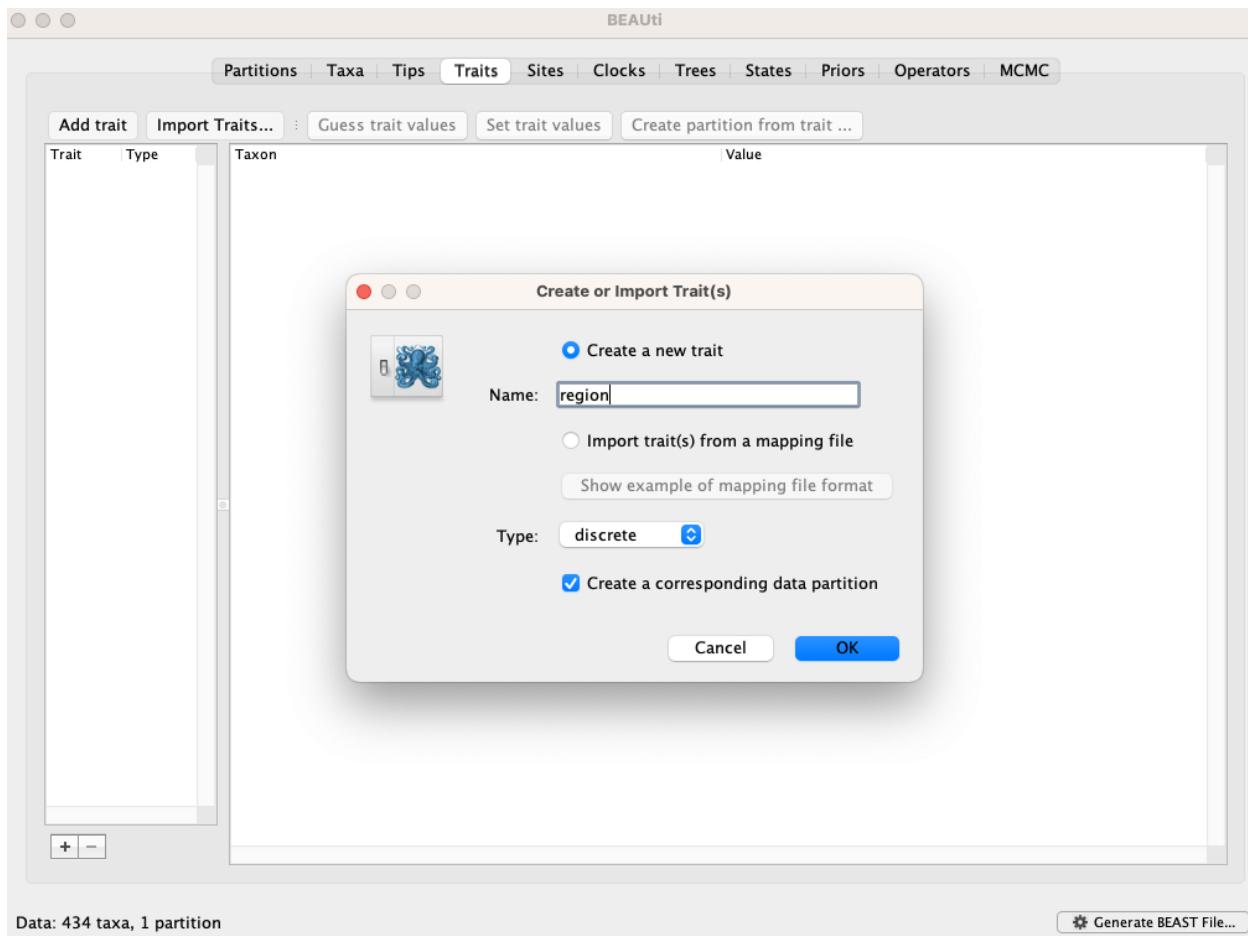
Dates as: Years | Since some time in the past | Specify origin date: | unable to parse date

Name	Date	Uncertainty	Height
A/equine/Jilin/1/1989 1989 M6501...	1989.0	1.0	33.183561643835674
A/canine/Zhejiang/S34/2015 2015...	2015.013698630137	0.0	7.169863013698659
A/equine/Suffolk/89 1989 X68437 ...	1989.0	1.0	33.183561643835674
A/equine/Ibadan/6/91 1991 X956...	1991.0	1.0	31.183561643835674
A/equine/Kentucky/692/1988 198...	1988.0	1.0	34.183561643835674
A/equine/Kentucky/694/1988 198...	1988.0	1.0	34.183561643835674
A/equine/Kentucky/698/1988 198...	1988.0	1.0	34.183561643835674
A/equine/Berlin/1/1989 1989 CY0...	1989.0	1.0	33.183561643835674
A/equine/Sussex/1/1989 1989 CY0...	1989.0	1.0	33.183561643835674
A/equine/Rook/93753/1989 1989 ...	1989.0	1.0	33.183561643835674
A/equine/Italy/1062/1991 1991 CY...	1991.0	1.0	31.183561643835674
A/equine/Italy/824/1991 1991 CY0...	1991.0	1.0	31.183561643835674
A/equine/Kentucky/1277/1990 19...	1990.0	1.0	32.183561643835674
A/equine/Texas/39655/1991 199...	1991.0	1.0	31.183561643835674
A/equine/Alaska/29759/1991 199...	1991.0	1.0	31.183561643835674
A/equine/Kentucky/1/1990 1990 L...	1990.0	1.0	32.183561643835674
A/equine/Hong_Kong/J/1992 1992 ...	1992.0	1.0	30.183561643835674
A/equine/Italy/1199/1992 1992 CY...	1992.0	1.0	30.183561643835674
A/equine/Switzerland/173/1993 19...	1993.0	1.0	29.183561643835674
A/equine/Austria/421/1992 1992 ...	1992.0	1.0	30.183561643835674
A/equine/Argentina/1/93 1993 L39...	1993.0	1.0	29.183561643835674
A/equine/Argentina/E37-2/1993 1...	1993.2410958904109	0.0	28.942465753424813
A/equine/Argentina/E41-2/1993 1...	1993.2438356164384	0.0	28.939726027397228
A/equine/Argentina/E37-18/1993 ...	1993.2410958904109	0.0	28.942465753424813
A/equine/Argentina/E131/1994 19...	1994.3917808219178	0.0	27.791780821917882
A/equine/Argentina/E226/1995 19...	1995.4164383561645	0.0	26.76712328767121
A/equine/Kentucky/1/1994 1994 L...	1994.0	1.0	28.183561643835674
A/equine/Florida/1/93 1993 L3991...	1993.0	1.0	29.183561643835674
A/equine/Kentucky/1/1991 1991 L...	1991.0	1.0	31.183561643835674
A/equine/Ibadan/6/91 1991 X956...	1991.0	1.0	31.183561643835674

Tip date sampling: Off | Apply to taxon set: All taxa

Data: 434 taxa, 1 partition | Generate BEAST File...

3. Next, we will set up our discrete trait analysis. We will add 2 traits: region, and host. Click on “Add Trait”, and name it “region”. Keep the type as “discrete” and keep the “Create a corresponding data partition” checked. It should look like this:



Now, to specify those values, click on “Guess Trait values” and set the region trait to second from last, with the “|” as delimiter. Click “OK”, and the regions should auto-fill.

BEAUti

Partitions Taxa Tips Traits Sites Clocks Trees States Priors Operators MCMC

Add trait Import Traits... Guess trait values Set trait values Create partition from trait ...

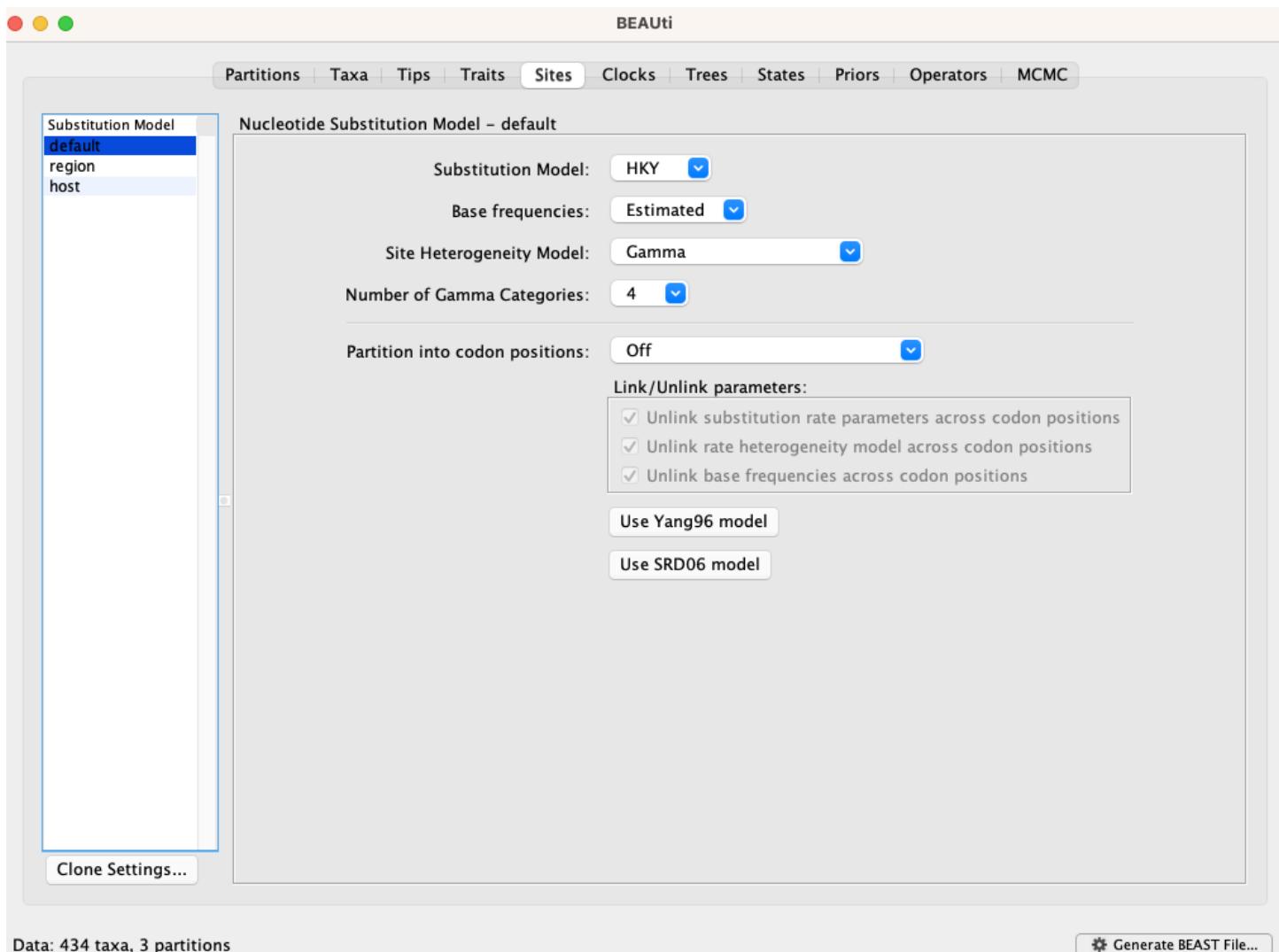
Trait	Type	Taxon	Value
region	discret...	A/equine/Jilin/1/1989 1989 M65018 H3N8 China China Equine	
		A/canine/Zhejiang/S34/2015 2015-01-06 MH018582 H3N8 China China Canine	
		A/equine/Suffolk/89 1989 X68437 H3N8 United Kingdom Europe Equine	
		A/equine/Ibadan/6/91 1991 X95637 H3N8 Nigeria Africa Equine	
		A/equine/Kentucky/692/1988 1988 CY030109 H3N8 USA North_America Equine	
		A/equine/Kentucky/694/1988 1988 CY030117 H3N8 USA North_America Equine	
		A/equine/Kentucky/698/1988 1988 CY030125 H3N8 USA North_America Equine	
		A/ [red dot] [grey dot] [grey dot]	Guess Trait Value for Taxa
			Extract values for trait 'region' from taxa labels
			The trait value is given by a part of string in the taxon label that is:
		<input checked="" type="radio"/> Defined by its order second from last	
		with delimiter	
		<input type="radio"/> Defined by regular expression (REGEX)	
		Cancel OK	
		A/equine/Argentina/E226/1995 1995-06-02 KX815362 H3N8 Argentina South_America Equine	
		A/equine/Kentucky/1/1994 1994 L39914 H3N8 USA North_America Equine	
		A/equine/Florida/1/93 1993 L39916 H3N8 USA North_America Equine	
		A/equine/Kentucky/1/1991 1991 L39918 H3N8 USA North_America Equine	
		A/equine/Ibadan/9/91 1991 X95638 H3N8 Nigeria Africa Equine	
		A/equine/Kentucky/1/1995 1995-01-01 MF182447 H3N8 USA North_America Equine	
		A/equine/Kentucky/2/1986 1986 M24727 H3N8 USA North_America Equine	
		A/equine/Kentucky/1/1987 1987 M24728 H3N8 USA North_America Equine	
		A/equine/Tennessee/5/1985 1985 M24726 H3N8 USA North_America Equine	

+ -

Data: 434 taxa, 2 partitions Generate BEAST File...

Repeat the same process to make a trait called “host”, which is specific as the last partition with the “|” delimiter.

- Now, we will set our nucleotide substitution model. For this analysis, select an HKY model, with estimated base frequencies, and a Gamma site heterogeneity model with 4 categories.



We also need to set a “substitution model” for our discrete trait analysis. This corresponds to the rate of transition between our categories (e.g., North America to South America or canine to equine). On the left side of the screen, click on “region” and select “symmetric substitution model” and “infer social network with BSSVS”. The symmetric substitution model means that the rate of transmission between regions is the same. For example, the inferred transmission rate from North America to South America is equal to South America to North America, so we will only infer 1 rate per pair of locations. Although this may not be realistic, for a first pass, this will greatly reduce the number of parameters we need to estimate. BSSVS is a method for reducing computational complexity by allowing transmission rates to be turned fully on and off. It essentially adds 0/1 indicator variables to each estimated rate, allowing the model to only infer the rates that matter. I don’t know why it is called “inferring a social network”, but BSSVS is useful and I generally always use it.

Repeat the same process for the host category, but set the transmission rates to be asymmetrical. For both traits, you’ll see an option for “set up a GLM”. A GLM, or generalized linear model is a cool way to extend these analyses to incorporate other information into a phylodynamic reconstruction. However, they are beyond the scope of this tutorial.

5. Setting clocks. People feel very specifically about clocks and have their own preferences. We will be investigating using alternative, host-specific clocks in the 2nd activity. For this, keep a strict clock for all categories.

6. In the Trees section, we specify what type of model we want for our phylogeny. We would like to investigate how the population size of H3N8 influenza changed during its introduction and spread in horses and dogs. To do so, we will be using a skyride model. The skyride model attempts to estimate the appropriate number of bins that are necessary to model the changes in population size over time.

BEAUTi

Partitions | Taxa | Tips | Traits | Sites | Clocks | **Trees** | States | Priors | Operators | MCMC

Link tree prior for all trees

Trees
default

Tree prior shared by all tree models

Tree Prior: Coalescent: GMRF Bayesian Skyride

Smoothing: Time-aware

For the Skyride, tree model/tree prior combination not implemented by BEAST. The Skyride is only available for a single model partition in this release. Please try the Skygrid or link all tree models.

Minin VN, Bloomquist EW, Suchard MA (2008) Mol Biol Evol 25, 1459–1471
[Skyride Coalescent].

Citation: Drummond AJ, Nicholls GK, Rodrigo AG, Solomon W (2002) Genetics 161, 1307–1320 [Serially Sampled Data].

Tree Model – default

Random starting tree

UPGMA starting tree

User-specified starting tree

Select user-specified tree: no tree loaded

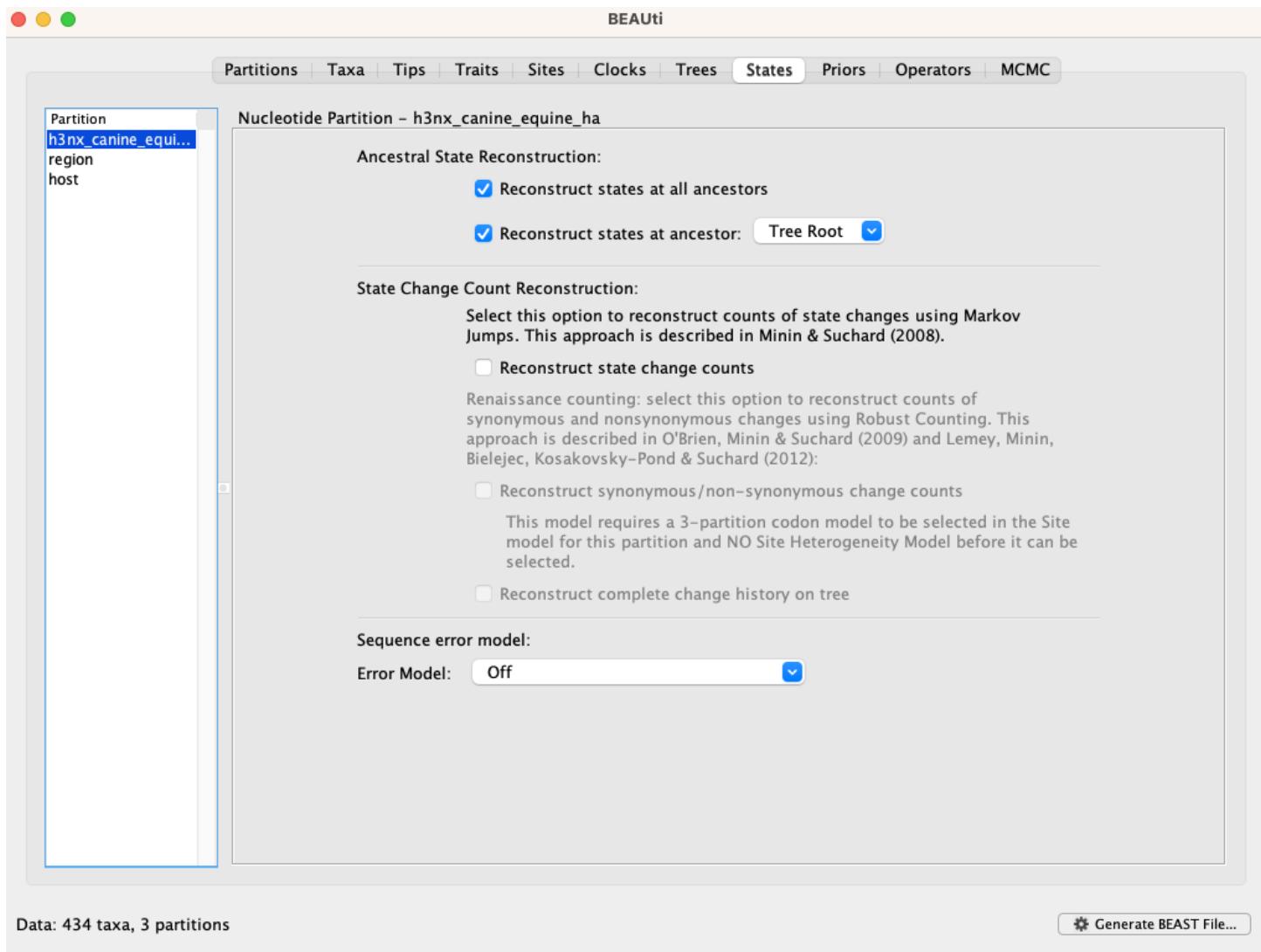
Export format for tree: Newick

Import user-specified starting trees from NEXUS format data files using the 'Import Data' menu option. Trees must be rooted and strictly bifurcating (binary).

Data: 434 taxa, 3 partitions

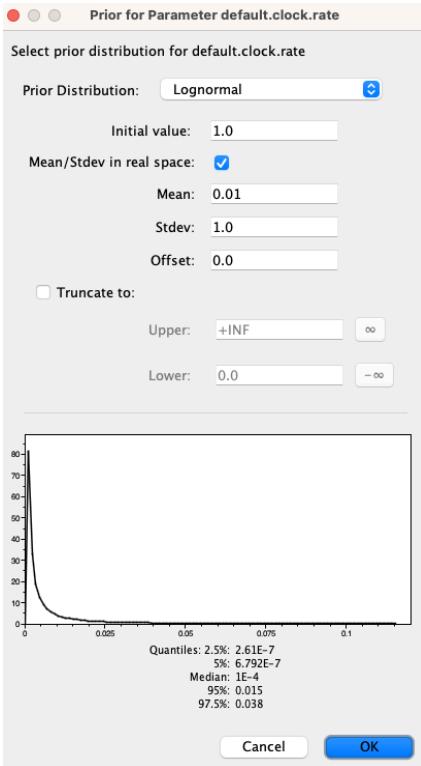
Generate BEAST File...

7. States. This specifies whether we would like to reconstruct ancestral states at internal nodes. This usually doesn't add significant computational time, and is sometimes fun and useful. For the nucleotide partition, this simply means inferring nucleotide mutations onto the internal nodes. For the region and host partitions, this means that we will infer the region or host onto internal nodes. Go ahead and select both boxes under "Ancestral State Reconstruction" for all 3 data partitions.



8. Picking priors is always at least a little bit tough. We want our priors to be reasonable, but we don't want them to constrain our analysis from exploring a variety of parameter values that could be probable. A good rule of thumb is that we want to pick broad, uninformative, but reasonable priors. Here are a few we can pick:

Clock rate: For the clock rate, we will use a lognormal distribution. Click on the “Mean/Stdev in real space” box, and check out the distribution as shown in the window below. We have a reasonable idea for what the clock rate should be for influenza, but we still don't want to influence the analysis too heavily. Leaving the defaults gives us a prior distribution with a 95% confidence interval of $\sim 10^{-7}$ to 10^{-2} . We should be almost positive that our real clock rate should fall within this range, so this is a reasonable range to set it.



The rest: the rest of the priors we can leave as is. We don't have incredibly strong prior knowledge about these other parameters, and the defaults are reasonable. Leave as is.

9. Operators. The operators in BEAST specify how parameters are chosen/searched. The correct operators and their weights are usually determined by the scientists who developed and tested the models that are included in BEAST, so usually the defaults are pretty good. I have only ever changed an operator once, and it was when I was manually editing my xml to add in some analyses. You can read more about the specific types of operators and how they work in the BEAST documentation. The weight represents how frequently those different parameters will be updated/sampled. Higher weighted operators operate more frequently, while lower weighted ones operate less frequently. Usually parameters that are more complex/difficult to infer are assigned higher weighted operators. You'll see on your screen that parameters relating to the tree topology inference are weighted the highest, followed by inferring the rates for region and host. This is because the tree topology is the hardest thing to infer with the most parameters, so it requires the most operational time within the MCMC chain.

BEAUti

Partitions Taxa Tips Traits Sites Clocks Trees States Priors Operators MCMC

Auto Optimize : Operator mix: classic operator mix

In use	Operates on	Type	Tuning	Weight	Description
<input type="checkbox"/>	Multiple	adaptiveMultivariate	1.0	30.0	Adaptive Multivariate Normal
<input checked="" type="checkbox"/>	kappa	scale	0.75	1.0	HKY transition-transversion parameter
<input checked="" type="checkbox"/>	frequencies	deltaExchange	0.01	1.0	frequencies
<input checked="" type="checkbox"/>	alpha	scale	0.75	1.0	gamma shape parameter
<input checked="" type="checkbox"/>	default.clock.rate	scale	0.75	3.0	substitution rate
<input checked="" type="checkbox"/>	default.Substitution rate and he...	upDown	0.75	3.0	Scales substitution rates inversely to node heights of the tree
<input checked="" type="checkbox"/>	region.clock.rate	scale	0.75	3.0	substitution rate
<input checked="" type="checkbox"/>	region.Substitution rate and he...	upDown	0.75	3.0	Scales substitution rates inversely to node heights of the tree
<input checked="" type="checkbox"/>	host.clock.rate	scale	0.75	3.0	substitution rate
<input checked="" type="checkbox"/>	host.Substitution rate and heig...	upDown	0.75	3.0	Scales substitution rates inversely to node heights of the tree
<input checked="" type="checkbox"/>	Tree	subtreeSlide	1.0	30.0	Performs the subtree-slide rearrangement of the tree
<input checked="" type="checkbox"/>	Tree	narrowExchange	n/a	30.0	Performs local rearrangements of the tree
<input checked="" type="checkbox"/>	Tree	wideExchange	n/a	3.0	Performs global rearrangements of the tree
<input checked="" type="checkbox"/>	Tree	wilsonBalding	n/a	3.0	Performs the Wilson-Balding rearrangement of the tree
<input checked="" type="checkbox"/>	treeModel.rootHeight	scale	0.75	3.0	root height of the tree
<input checked="" type="checkbox"/>	Internal node heights	uniform	n/a	30.0	Draws new internal node heights uniformly
<input type="checkbox"/>	Tree	subtreeLeap	1.0	434.0	Performs the subtree-leap rearrangement of the tree
<input type="checkbox"/>	Tree	subtreeJump	1.0	43.4	Performs the subtree-jump rearrangement of the tree
<input checked="" type="checkbox"/>	gmrfGibbsOperator	gmrfGibbsOperator	2.0	2.0	Gibbs sampler for GMRF Skyride
<input checked="" type="checkbox"/>	region.rates	scaleIndependently	0.75	15.0	region.rates
<input checked="" type="checkbox"/>	region.indicators	bitFlip	n/a	7.0	region.indicators
<input checked="" type="checkbox"/>	host.rates	scaleIndependently	0.75	15.0	host.rates
<input checked="" type="checkbox"/>	host.indicators	bitFlip	n/a	7.0	host.indicators
<input checked="" type="checkbox"/>	region.root.frequencies	deltaExchange	0.75	1.0	region.root.frequencies
<input checked="" type="checkbox"/>	host.root.frequencies	deltaExchange	0.75	1.0	host.root.frequencies

Data: 434 taxa, 3 partitions

 Generate BEAST File...

BEAUti

Partitions | Taxa | Tips | Traits | Sites | Clocks | Trees | States | Priors | Operators | MCMC

Use classic priors/operators

Parameter	Prior	Bound	Description
kappa	* LogNormal [1, 1.25], initial=2	[0, ∞]	HKY transition-transversion parameter
frequencies	* Dirichlet [1,1]	[0, ∞]	base frequencies
alpha	* Exponential [0.5], initial=0.5	[0, ∞]	gamma shape parameter
default.clock.rate	LogNormal [0.01, 1], initial=1	[0, ∞]	substitution rate
region.clock.rate	Approx. Reference Prior, initia...	[0, ∞]	substitution rate
host.clock.rate	Approx. Reference Prior, initia...	[0, ∞]	substitution rate
treeModel.rootHeight	* Using Tree Prior in [59.1835...	[59.18356..., ∞]	root height of the tree
skyride.precision	* Gamma [0.001, 1000], initial...	[0, ∞]	GMRF Bayesian skyride precision
region.nonZeroRates	* Poisson [9]	n/a	the number of non-zero rates for BSSVS
region.frequencies	* Uniform [0, 1], initial=0.25	[0, 1]	discrete state frequencies
region.rates	* Gamma [1, 1], initial=1	[0, ∞]	discrete trait instantaneous transition rates
host.nonZeroRates	* Poisson [1]	n/a	the number of non-zero rates for BSSVS
host.frequencies	* Uniform [0, 1], initial=0.25	[0, 1]	discrete state frequencies
host.rates	* Gamma [1, 1], initial=1	[0, ∞]	discrete trait instantaneous transition rates
region.root.frequencies	* Uniform [0, 1], initial=0.25	[0, 1]	discrete state root frequencies
host.root.frequencies	* Uniform [0, 1], initial=0.25	[0, 1]	discrete state root frequencies

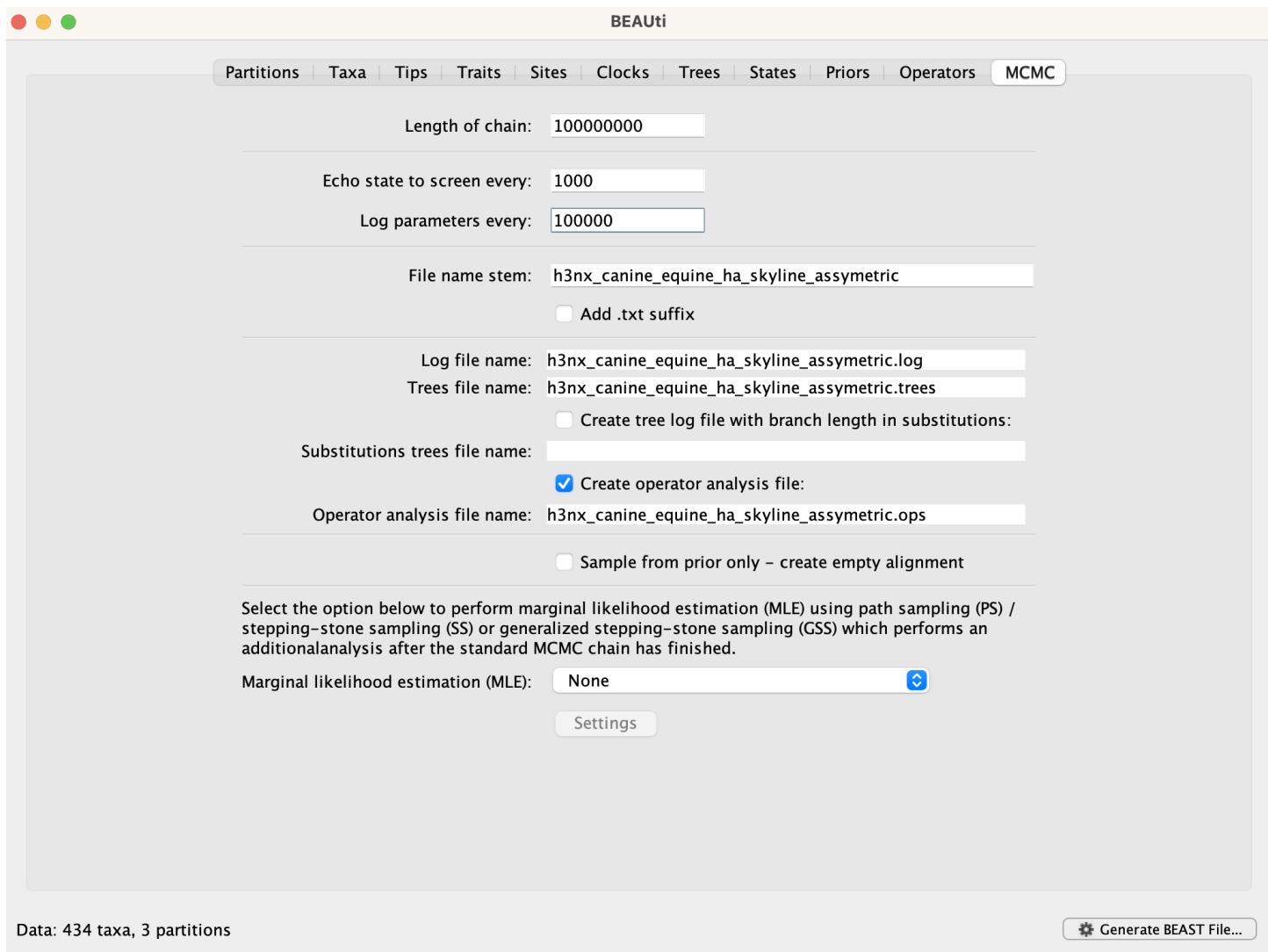
Specify prior probability distributions on each and every parameter of the current model.

Link parameters together | Link parameters into a hierarchical model | Unlink parameters

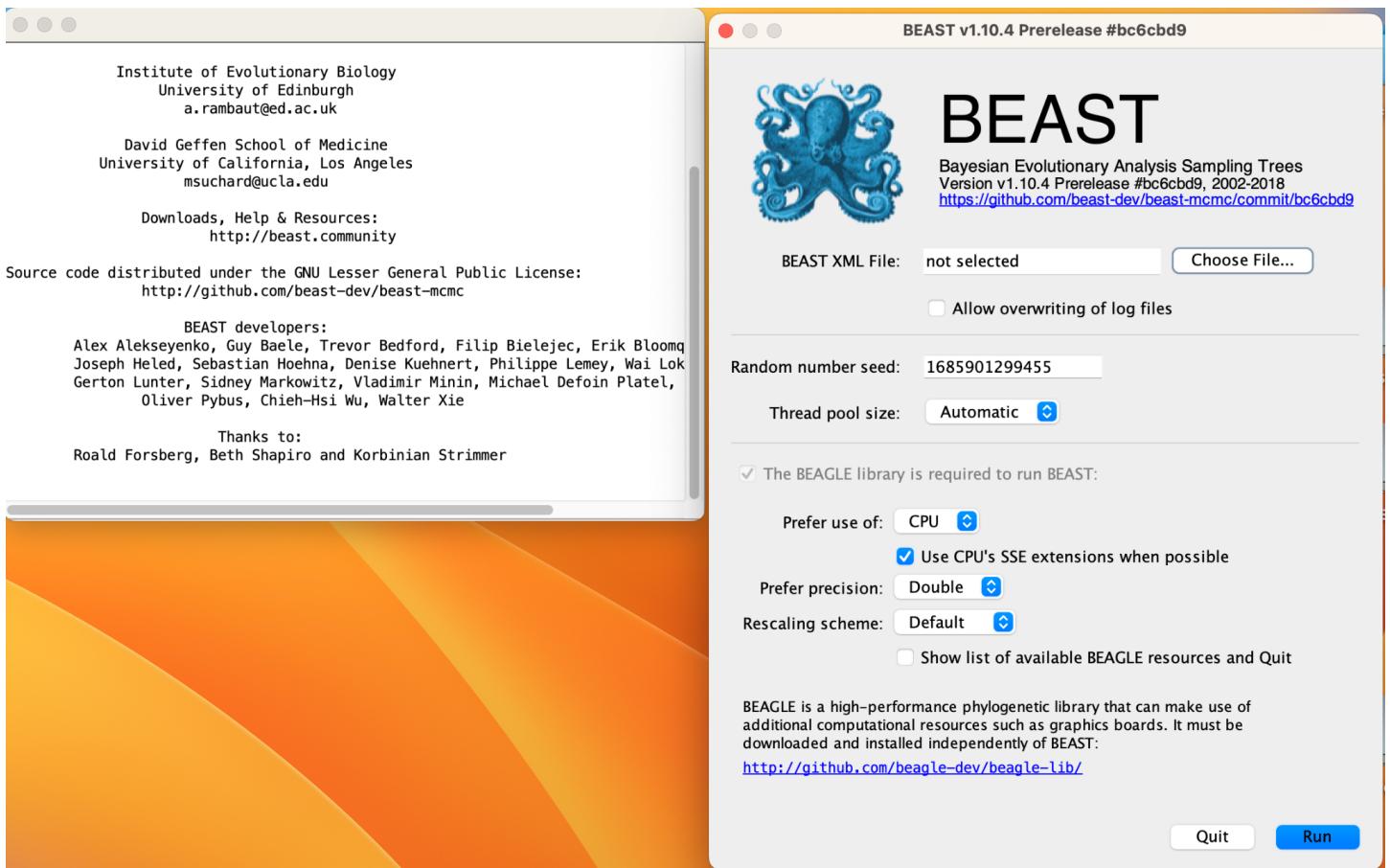
* Marked parameters currently have a default prior distribution. You should check that these are appropriate.

Data: 434 taxa, 3 partitions | Generate BEAST File...

10. MCMC chain. We're almost done! The last step is to specify how long you want to run the chain and how frequently you'd like to sample. A very common default is to run the chain for 50 million steps, sampling every 5,000. However, I've found that with this frequency of sampling, you can end up with enormous .trees output files and that these are very cumbersome to work with. BEAST analyses can be restarted in theory, but I always find it easier to just set a very long chain length, and then stop it prematurely if it converges faster than I anticipate. So for this analysis, we'll set the chain length to 100,000,000 generations, sampling every 100,000. As a rule of thumb, sampling ~0.1% of sampled steps will give you reasonable output files while sampling frequency enough to have a good sample of the posterior. Another good rule of thumb is that at the end, sampling ~1000 trees is a good target. Again, this keeps file sizes to a reasonable size while ensuring adequate sampling of the posterior. You can also specify output file names here. When you're done, click on "Generate BEAST file", and save it to a location on your laptop. The saved file will have a ".xml" extension, which is openable in a standard text editor. Feel free to open it up and take a look in Text Wrangler.



11. Start the analysis. Double click on BEAST v1.10.4 app to open it. This will open up a command line screen and a graphical user interface. To upload your xml file, click on “Choose File” and upload it. You’ll see a few options here, none of which we need to edit. The random seed number is simply a random set of numbers assigned to this particular job. You can also specify usage of CPUs and GPUs. In some older versions of BEAST, you had to select using BEAGLE, but now it is automatic. BEAGLE handles memory scaling in BEAST, and is necessary/very useful to use in basically all cases. Click on “Run” and it should begin!



Once the run starts, the screen should look like this:

```

BEAST v1.10.4 Prerelease #bc6cbd9, 2002-2018
Bayesian Evolutionary Analysis Sampling Trees
Designed and developed by
Alexei J. Drummond, Andrew Rambaut and Marc A. Suchard

Department of Computer Science
University of Auckland
alexei@cs.auckland.ac.nz

Institute of Evolutionary Biology
University of Edinburgh
a.rambaut@ed.ac.uk

David Geffen School of Medicine
University of California, Los Angeles
msuchard@ucla.edu

Downloads, Help & Resources:
http://beast.community

Source code distributed under the GNU Lesser General Public License:
http://github.com/beast-dev/beast-mcmc

BEAST developers:
Alex Alekseyenko, Guy Baele, Trevor Bedford, Filip Bielejec, Erik Bloomquist, Matthew Hall,
Joseph Heled, Sebastian Hoehna, Denise Kuehnert, Philippe Lemey, Wai Lok Sibon Li,
Gerton Lunter, Sidney Markowitz, Vladimir Minin, Michael Defoin Plateau,
Oliver Pybus, Chieh-Hsi Wu, Walter Xie

Thanks to:
Roald Forsberg, Beth Shapiro and Korbinian Strimmer

Using BEAGLE library v4.0.0 (PRE-RELEASE) for accelerated, parallel likelihood evaluation
2009-, BEAGLE Working Group - https://beagle-dev.github.io/
Citation: Ayres et al (2019) Systematic Biology 68: 1052-1061 | doi:10.1093/sysbio/syz020

Random number seed: 1685901529222

Loading additional development parsers from development_parsers.properties, which is additional set of parsers only available for development version ...
Parsing XML file: h3nx_canine_equine_ha_skyride_assymmetric-fixed-loggers.xml
File encoding: UTF8
Looking for plugins in /Users/lhmonclia/plugins

Read alignment: alignment
Sequences = 434
Sites = 1740
Datatype = nucleotide
Site patterns 'patterns' created from positions 1-1740 of alignment 'alignment'
pattern count = 1740
Read attribute patterns, 'region.pattern' for attribute, region
Read attribute patterns, 'host.pattern' for attribute, host

```

And eventually, it will start running and look like this. Notice on the far right hand column, you will see a printout of how quickly the analysis will run. This one on my laptop will take an expected 0.23 hours per million generations, or ~1 day to get to 100 million generations.

```

1.10. Virus Evolution. vey016. DOI:10.1093/ve/vey016
Using BEAGLE likelihood calculation library:
    Ayres et al (2012) BEAGLE: a common application programming interface and high-performance computing library for statistical phylogenetics. Syst Biol. 61, 170-173. DOI:10.1093/sysbio/syr100

TREE DENSITY MODELS
Skyride coalescent:
    Minin VN, Bloomquist EW, Suchard MA (2008) Smooth skyride through a rough skyline: Bayesian coalescent-based inference of population dynamics. Mol Biol Evol. 25, 1459-1471. DOI:10.1093/molbev/msn090

SUBSTITUTION MODELS
HKY nucleotide substitution model:
    Hasegawa M, Kishino H, Yano T (1985) Dating the human-ape splitting by a molecular clock of mitochondrial DNA. J. Mol. Evol.. 22, 160-174
Discrete gamma-distributed rate heterogeneity model:
    Yang Z (1994) Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: approximate methods. J. Mol. Evol.. 39, 306-314
Complex-diagonalizable, irreversible substitution model:
    Edwards CJ, Suchard MA, Lemey P, Welch JJ, Barnes I, Fulton TL, Barnett R, O'Connell TC, Coxon P, Monaghan N, Valdiosera CE, Lorenzen ED, Willerslev E, Baryshnikov GF, Rambaut A, Thomas MG, Bradley DG, Shapiro B (2011) Ancient hybridization and an Irish origin for the modern polar bear matriline. Current Biology. 21, 1251-1258

PRIOR MODELS
CTMC Scale Reference Prior model:
    Ferreira MAR, Suchard MA (2008) Bayesian analysis of elapsed times in continuous-time Markov chains. Canadian Journal of Statistics. 36, 355-368

# BEAST v1.10.4 Prerelease #bc6cbd9
# Generated Sun Jun 04 13:58:49 EDT 2023 [seed=1685901529222]
# /Users/lhmoncla/src/csu-infectious-disease-wildlife-course/beast-lab/h3nx-canine-equine/h3nx_canine_equine_ha_skyride_assymmetric-fixed-loggers.xml
# keywords: skyride
state Joint Prior Likelihood age(root) default.clock.rate region.clock.rate host.clock.rate region.no
nZeroRates host.nonZeroRates
0 -184277.9366 -24267.0602 -160010.8764 1444.87 1.00000 1.00000 1.00000 90 2 -
Underflow calculating likelihood. Attempting a rescaling...
1000 -78782.1905 -5323.1122 -73459.0784 1712.06 0.25470 0.23333 0.52394 79 2 -
2000 -67754.7160 -3969.1910 -63785.5250 1858.83 0.12244 0.11946 0.29555 55 2 -
3000 -64146.2391 -4048.6760 -60097.5632 1895.51 2.85893E-2 4.50713E-2 0.18733 43 2 -
4000 -61153.5705 -3633.1219 -57520.4486 1901.12 9.84405E-3 3.39147E-2 6.33413E-2 38 2 -
5000 -59840.8966 -3757.1825 -56083.7141 1904.52 8.78955E-3 3.36517E-2 1.34273E-2 33 2 -
6000 -58468.9208 -3640.7916 -54828.1292 1905.37 8.66187E-3 2.83237E-2 1.00199E-2 29 2 -
7000 -56882.5124 -3694.7916 -53187.7208 1905.88 7.85257E-3 2.26064E-2 1.16233E-2 28 2 -
8000 -56077.6372 -3691.0552 -52386.5821 1907.44 7.51445E-3 2.51472E-2 9.26071E-3 26 2 -
9000 -55185.2396 -3814.0480 -51371.1916 1909.97 4.54493E-3 2.66625E-2 1.09681E-2 27 2 -
10000 -54025.1232 -3993.5226 -50031.6006 1918.84 3.90258E-3 2.55705E-2 9.82184E-3 28 2 -
11000 -52783.7338 -3697.4834 -49086.2504 1923.54 3.95794E-3 2.02276E-2 6.29722E-3 28 2 0.23 hour
s/million states
12000 -52420.9951 -4032.6853 -48388.3098 1925.05 3.89878E-3 2.29289E-2 9.01973E-3 28 2 0.24 hour
s/million states
13000 -51663.5215 -3789.3068 -47874.2147 1925.91 3.84002E-3 1.99759E-2 7.93876E-3 30 2 0.23 hour
s/million states
14000 -51134.3529 -3775.7640 -47358.5890 1925.54 3.69647E-3 2.20645E-2 8.67158E-3 28 2 0.23 hour
s/million states
15000 -50590.1870 -3933.3031 -46656.8839 1926.39 3.52161E-3 2.128E-2 5.24574E-3 27 2 0.24 hour
s/million states

```

Note: if the above didn't work for you for some reason, you may have some sort of java version issue. To get around this, you can always open up a Terminal window, and drag the beast executable within the "BEAST v1.10.4/bin/beast" into the terminal window, followed by the path to your xml. That will also launch BEAST successfully.

12. Finally, although we won't do this here, it is always a good idea to run your BEAST runs in replicates, and then combine and compare the results. The reason this is necessary is that MCMC chains are random, so different chains initiating from the same xml file may start and traverse different parameter spaces. If multiple chains converge on very similar parameter distributions, this is a good indication that those distributions are frequently inferred as the most probable. If the different chains converge on very different results, that tells you that there are multiple combinations of parameters that are compatible with the alignment and model. Having an accurate sense of uncertainty is scientifically important, and a real benefit of Bayesian analyses. I usually launch 3 independent replicates of the same xml, simply by copying those xmls and renaming them something

like “it1-xmlname.xml”, “it2-xmlname.xml”, etc... Most computational clusters have batch systems that make this easy.

Part 2: Interpreting the analysis

1. We don't have a whole day to wait for this analysis to run, so I've run this beforehand on our cluster and uploaded the results for us to look at. To look at the results of our run, first open up “Tracer”. Tracer is a piece of software developed to visualizing BEAST MCMC chains.

To finish this out: have them check out the analysis when it is not done running yet and has not converged, and then have them check again when it is fully completed.

Do the skyride analysis and visualize.

Make an MCC tree, look at it in FigTree, answer some questions.

Analysis 2: Adding in a local clock

In this analysis, we will be exploring different clock rates, and adding in a fixed local clock to the analysis. When viruses jump to new host species, they often require adaptation to that new host. The HA gene encodes the influenza receptor binding protein, which is a common target for host-specific selection. It is a reasonable hypothesis that transmission from horses to dogs could impose selection on the HA gene, resulting in faster evolution and accumulation of amino acid changes. In this analysis, we will test that hypothesis by adding in a fixed local clock. This model allows the clock rate to vary among monophyletic groups in the tree. In our case, we will define two different host partitions in the data, canine and equine, and use a fixed local clock model to estimate the TMRCA for each host and the clock rate within each host species. We can then compare the estimates to determine whether there is evidence of increased adaptation in dogs following transmission from horses.

Part 1: Setting up the analysis

1. We will be using most of the same parameters for this analysis that we did in Analysis 1. So, starting with the Beauti input from Analysis 1, now go back to the 2nd tab, labeled “Taxa”. In order to assign different hosts different clock rates, we need to specify taxon groups, which we will do in this tab. In the bottom on this panel, click on the plus sign, which should add a new taxon group. Label the first one “equine_taxa” and select the equine sequences to include by selecting them in the left box and clicking on the right arrow to include them in the equine taxon set. Do the same for canine. For both, select “mono”. This enforces these groups to be monophyletic, which we know that they are. The results should look like below:

BEAUTI

Partitions Taxa Tips Traits Sites Clocks Trees States Priors Operators MCMC

Import Taxon Sets

Taxon Set	Mono?	Stem?	Tree	Age
canine_taxa	<input checked="" type="checkbox"/>	<input type="checkbox"/>	default	
equine_taxa	<input checked="" type="checkbox"/>	<input type="checkbox"/>	default	

Taxon set: equine_taxa

Excluded Taxa

A/canine/New_York/147926-5/2006|2006-12-18|CY...
A/canine/New_York/158402-1/2008|2008-12-31|CY...
A/canine/New_York/159903/2012|2012|MF173187|H...
A/canine/New_York/1623.1/2010|2010|MF173308|H...
A/canine/New_York/3699/2010|2010|MF173302|H3N...
A/canine/New_York/3821631/2011|2011-07-06|JX2...
A/canine/New_York/4986-2/2006|2006-01-16|CY06...
A/canine/New_York/5183-6/2006|2006-01-16|CY06...
A/canine/New_York/51854/2008|2008-04-24|CY067...
A/canine/New_York/6977983/2010|2010-09-05|JX2...
A/canine/PA/111788.2/2009|2009|KM359813|H3N8|U...
A/canine/PA/27637.3/2010|2010|KM359806|H3N8|U...
A/canine/PA/33225.4/2010|2010|KM359804|H3N8|U...
A/canine/PA/96978/2009|2009|KM359808|H3N8|U...
A/canine/Pennsylvania/10909/2007|2007-01-30|CY0...
A/canine/Pennsylvania/137154/2008|2008-10-28|CY...
A/canine/Pennsylvania/16699/2007|2007-02-15|CY0...
A/canine/Pennsylvania/94930-3/2007|2007-07-21|C...
A/canine/Pennsylvania/96978/2009|2009|MF173144|...
A/canine/South_Carolina/89215/2009|2009-12-05|JX...
A/canine/Sydney/6525/2007|2007-10-14|GU045762...
A/canine/Sydney/6692/2007|2007-10-15|GU045761...
A/canine/Texas/1/2004|2004|DQ124196|H3N8|Usa...
A/canine/VT/11039/2013|2013|KM359805|H3N8|Usa...
A/canine/VT/2782/2013|2013|KM359810|H3N8|Usa...
A/canine/Vermont/278213/2013|2013|MF173395|H3...
A/canine/Wyoming/86033/2007|2007-02-14|HQ917...
A/canine/Wyoming/86955/2007|2007-02-27|JX2353...
A/canine/Zhejiang/534/2015|2015-01-06|MH01858...
A/dog/Jacksonville/C3/2005|2005|EU402407|H3N8|U...
A/dog/Miami/F3/2005|2005|EU402408|H3N8|Usa|No...
434 taxa excluded

Select: taxon set...

Included Taxa

A/Equus_caballus/Ulaanbaatar/1201/2013|2013-07-...
A/Equus_caballus/Ulaanbaatar/1202/2013|2013-07-...
A/Equus_caballus/Ulaanbaatar/1203/2013|2013-07-...
A/Equus_caballus/Ulaanbaatar/2/2011|2011-07-05|...
A/Equus_caballus/Ulaanbaatar/3/2011|2011-07-05|...
A/Equus_caballus/Ulaanbaatar/4/2011|2011-07-05|...
A/Equus_caballus/Ulaanbaatar/5/2011|2011-07-05|...
A/Equus_caballus/Ulaanbaatar/6/2011|2011-07-05|...
A/Equus_caballus/USA/176576/2019|2019-09-...
A/Equus_ferus_caballus/USA/218776/2018|2018-12-...
A/Equus_ferus_caballus/USA/48519/2019|2019-03-2...
A/Equus_ferus_caballus/USA/51053/2019|2019-03-3...
A/Equus_ferus_caballus/USA/53954/2019|2019-04-0...
A/Equus_ferus_caballus/USA/72778/2019|2019-04-2...
A/Equus_ferus_caballus/USA/82889/2019|2019-05-0...
A/donkey/Nigeria/VRD19_ZM20_T10_19RS49-5/201...
A/donkey/Shandong/1/2017|2017-03-27|MG132047...
A/donkey/Xinjiang/5/2007|2007-12|EU794575|H3N8...
A/equine/Ahmedabad/1/2009|2009-04-26|GU3961...
A/equine/Ain/1/2014|2014|KY241318|H3N8|France|E...
A/equine/Alaska/29759/1991|1991|CY030157|H3N8...
A/equine/Algiers/1/1972|1972|CY032945|H3N8|Alge...
A/equine/Algiers/1972|1972|M24721|H3N8|Algeria|Af...
A/equine/Almaty/26/2007|2007-09-03|GU953266|H...
A/equine/Ankara/1/2013|2013-07-17|MF067527|H3...
A/equine/Argentina/1/93|1993|L39913|H3N8|Argenti...
A/equine/Argentina/E1134-7/2001|2001-07-31|KX8...
A/equine/Argentina/E131/1994|1994-05-24|KX8153...
A/equine/Argentina/E226/1995|1995-06-02|KX8153...
A/equine/Argentina/E2345-1/2012|2012-07-09|KX8...
A/equine/Argentina/E2345-5/2012|2012-07-09|KX8...
0 taxa included

Select: taxon set...

Data: 434 taxa, 3 partitions

2. Now navigate to the “clocks” panel. For the default clock rate, select “fixed local clock” instead of strict clock.
3. Now, navigate to the “priors” panel. You should now see a new set of priors for the clock rates. We should change the default for the “canine_taxa” and “equine_taxa” clock rates, which are specified as “default.canine_taxa.rate” and “default.equine_taxa.rate”. Because our a priori assumption of the clock rates is that they should not deviate too far beyond the overall influenza clock rate, we can use the same prior we used for the strict clock in Analysis 1, a Lognormal with a mean of 0.01, with mean in real space.

BEAUti

Partitions | Taxa | Tips | Traits | Sites | Clocks | Trees | States | Priors

Use classic priors/operators

Parameter	Prior	Bound	Description
tmrca(canine_taxa)	* Using Tree Prior	n/a	tmrca statistic for taxon set canine_taxa on tree defa
tmrca(equine_taxa)	* Using Tree Prior	n/a	tmrca statistic for taxon set equine_taxa on tree defa
kappa	* LogNormal [1, 1.25], initial=2	[0, ∞]	HKY transition-transversion parameter
frequencies	* Dirichlet [1,1]	[0, ∞]	base frequencies
alpha	* Exponential [0.5], initial=0.5	[0, ∞]	gamma shape parameter
default.clock.rate	LogNormal [R0.01, 1], initial=1	[0, ∞]	substitution rate
default.canine_taxa.rate	LogNormal [R0.01, 1], initial=1	[0, ∞]	substitution rate
default.equine_taxa.rate	LogNormal [R0.01, 1], initial=1	[0, ∞]	substitution rate
region.clock.rate	Approx. Reference Prior, initial=1	[0, ∞]	substitution rate
host.clock.rate	* Approx. Reference Prior, initial=1	[0, ∞]	substitution rate
treeModel.rootHeight	* Using Tree Prior in [59.1835... [59.18356...	[0, ∞]	root height of the tree
skyride.precision	* Gamma [0.001, 1000], initial=1	[0, ∞]	GMRF Bayesian skyride precision
region.nonZeroRates	* Poisson [9]	n/a	the number of non-zero rates for BSSVS
region.frequencies	* Uniform [0, 1], initial=0.25	[0, 1]	discrete state frequencies
region.rates	* Gamma [1, 1], initial=1	[0, ∞]	discrete trait instantaneous transition rates
host.nonZeroRates	* Poisson [1]	n/a	the number of non-zero rates for BSSVS
host.frequencies	* Uniform [0, 1], initial=0.25	[0, 1]	discrete state frequencies
host.rates	* Gamma [1, 1], initial=1	[0, ∞]	discrete trait instantaneous transition rates
region.root.frequencies	* Uniform [0, 1], initial=0.25	[0, 1]	discrete state root frequencies
host.root.frequencies	* Uniform [0, 1], initial=0.25	[0, 1]	discrete state root frequencies

Link parameters together | Link parameters into a hierarchical model | Unlink parameters

* Marked parameters currently have a default prior distribution. You should check that these are appropriate.

Data: 434 taxa, 3 partitions

Prior for Parameter default.canine_taxa.rate

Select prior distribution for default.canine_taxa.rate

Prior Distribution: Lognormal

Initial value: 1.0

Mean/Stdev in real space: ✓

Mean: 0.01

Stdev: 1.0

Offset: 0.0

Truncate to:

Upper: +INF

Lower: 0.0

Quantiles: 2.5%: 2.61E-7
5%: 6.792E-7
Median: 1E-4
95%: 0.015
97.5%: 0.038

Cancel OK

Generate BEAST File...

4. Save the xml as before, and launch!