

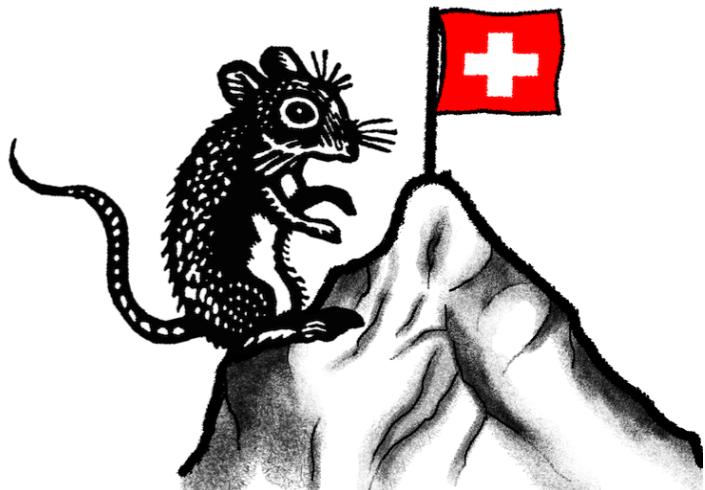


Taming the BEAST

Bayesian Evolutionary Analyses by Sampling Trees

Louis du Plessis

1. What goes into a BEAST model?
2. BEAST2 workflow
3. Introduction to BEAST2 tutorial
4. BEAST best practice



We all have one thing in common...



We all use **BEAST** to answer questions
about our data

Bayesian phylogenetic and phylodynamic inference

$$P(\text{model} \mid \text{data}) = \frac{P(\text{data} \mid \text{model})P(\text{model})}{P(\text{data})}$$

Likelihood

Posterior

Prior

Marginal Likelihood of the data

Bayesian inference

(Data and model parameters are both described by probabilities)

Prior → $P(\text{model})$

- Have some degree of belief in our hypothesis
- All model parameters have priors, whether you specify them or not

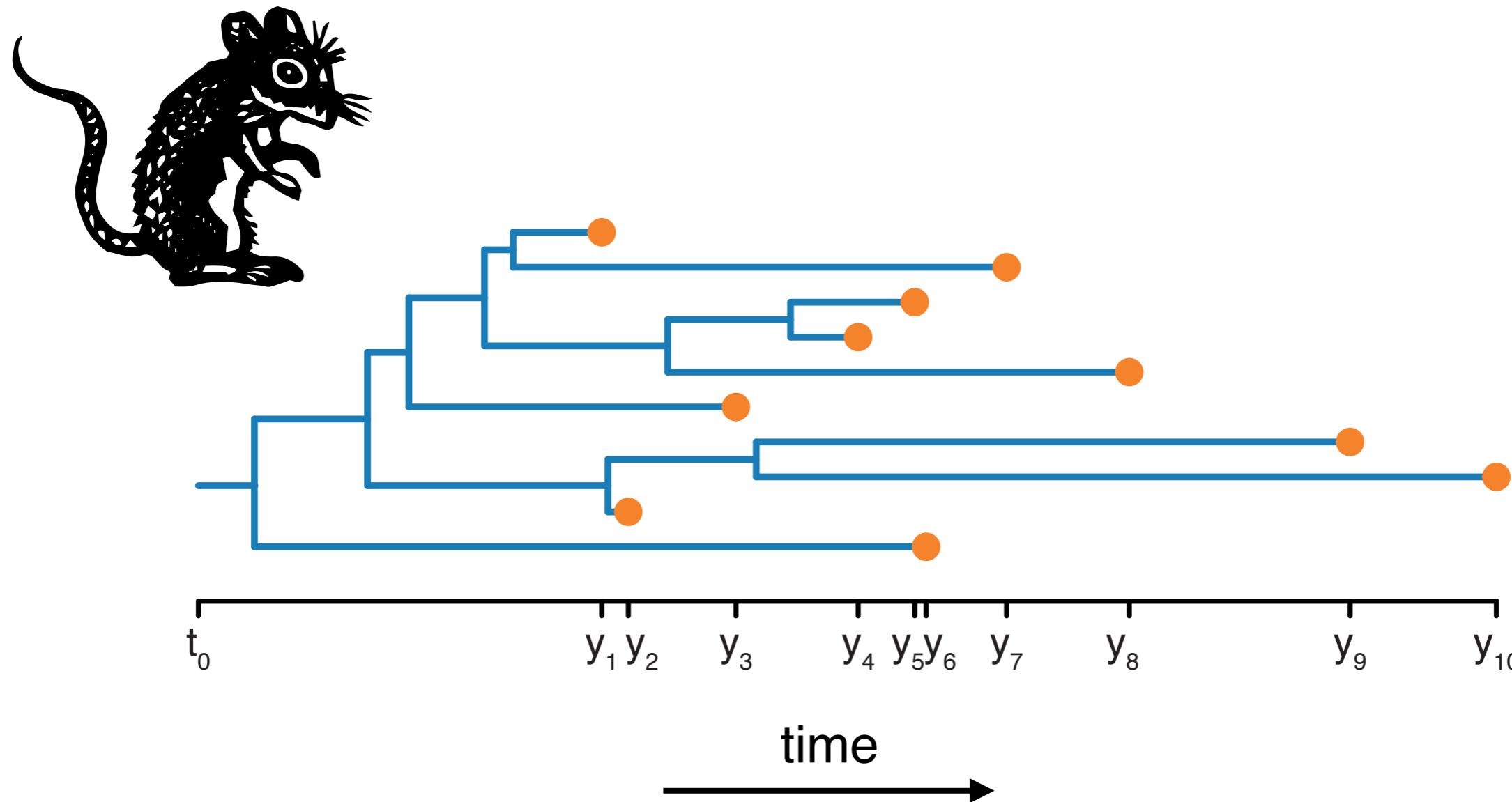
Likelihood → $P(\text{data} \mid \text{model})$

- Likelihood is proportional to the probability of observing the data given a hypothesis

Posterior → $P(\text{model} \mid \text{data})$

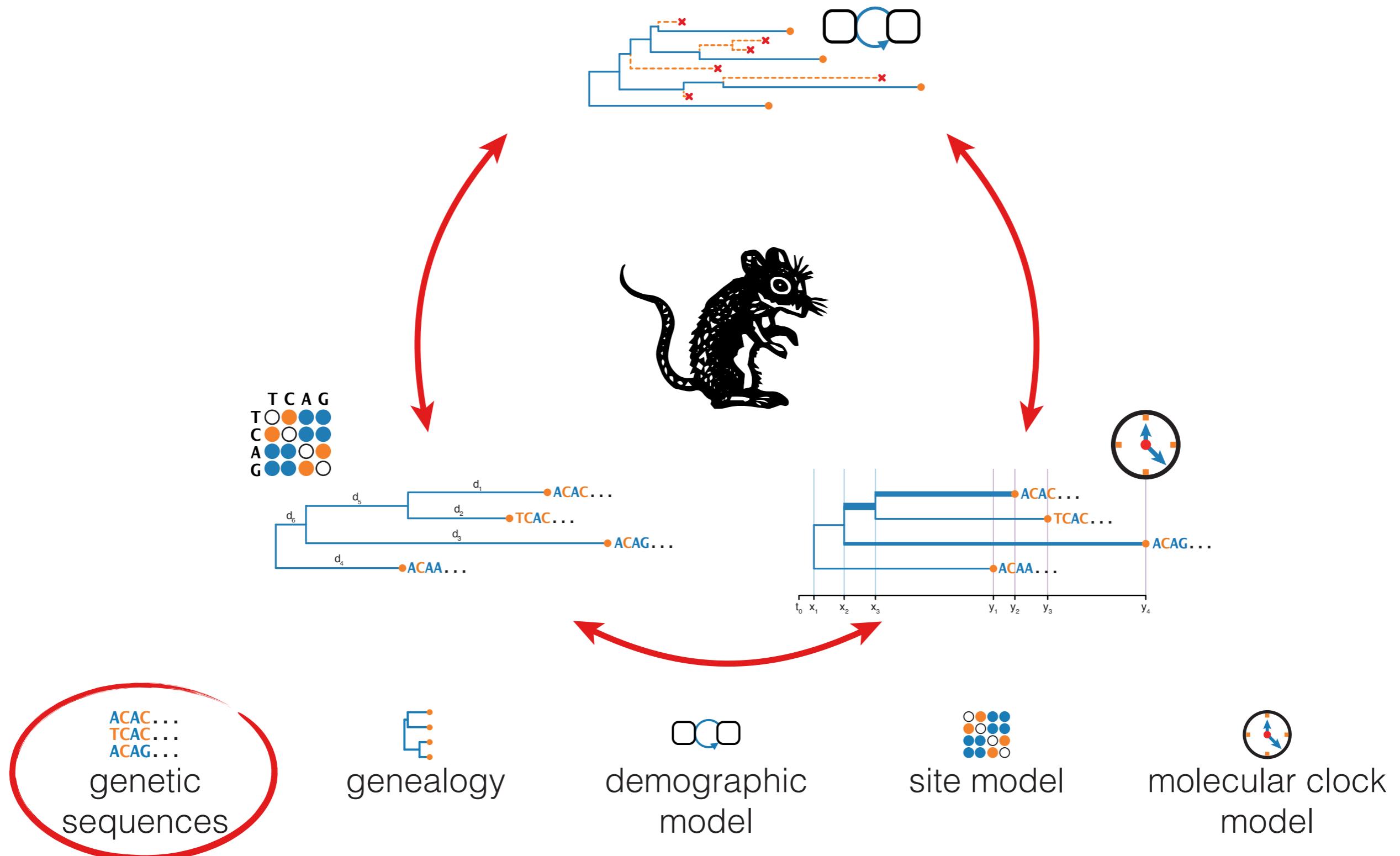
- Combines information from the data (**likelihood**) and previous knowledge (**prior**)

Rooted time-trees



Fundamental data structure in **BEAST**
is a **rooted time-tree**

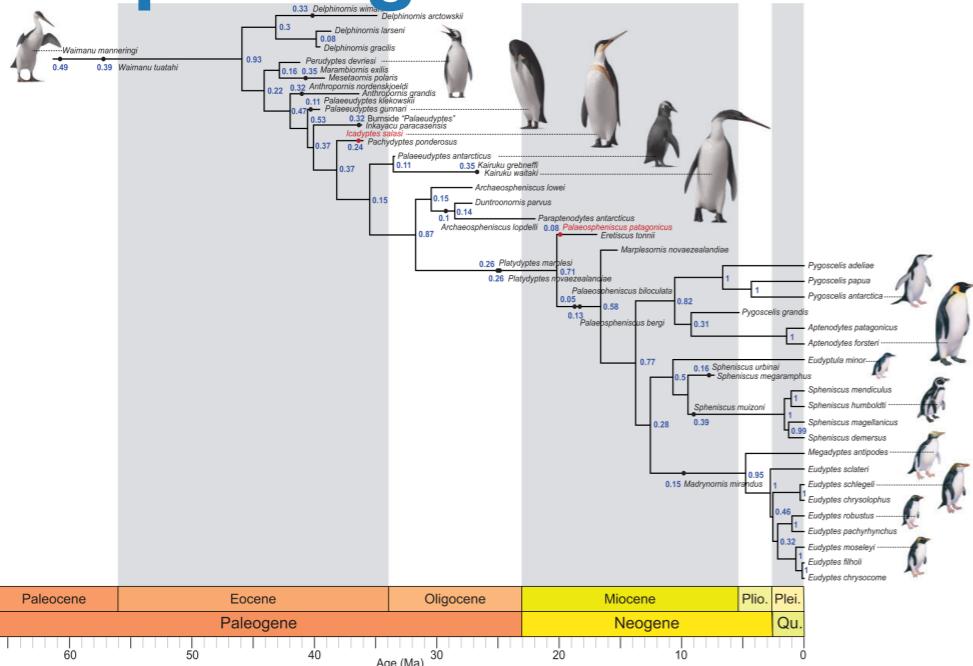
What goes into a **BEAST** model?



The data

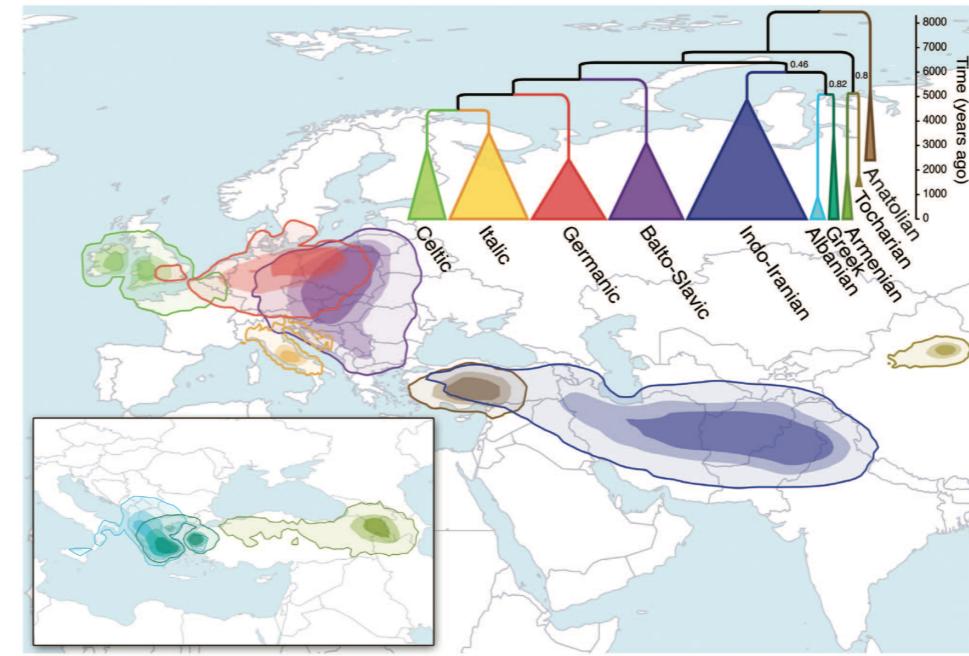
- Typically an alignment of DNA or RNA sequences
- Can also be amino acids or codons
- Often split data into multiple partitions
 - Multiple genes
 - 1st, 2nd and 3rd codon positions
- Does not **have** to be genetic sequences!

Morphological traits



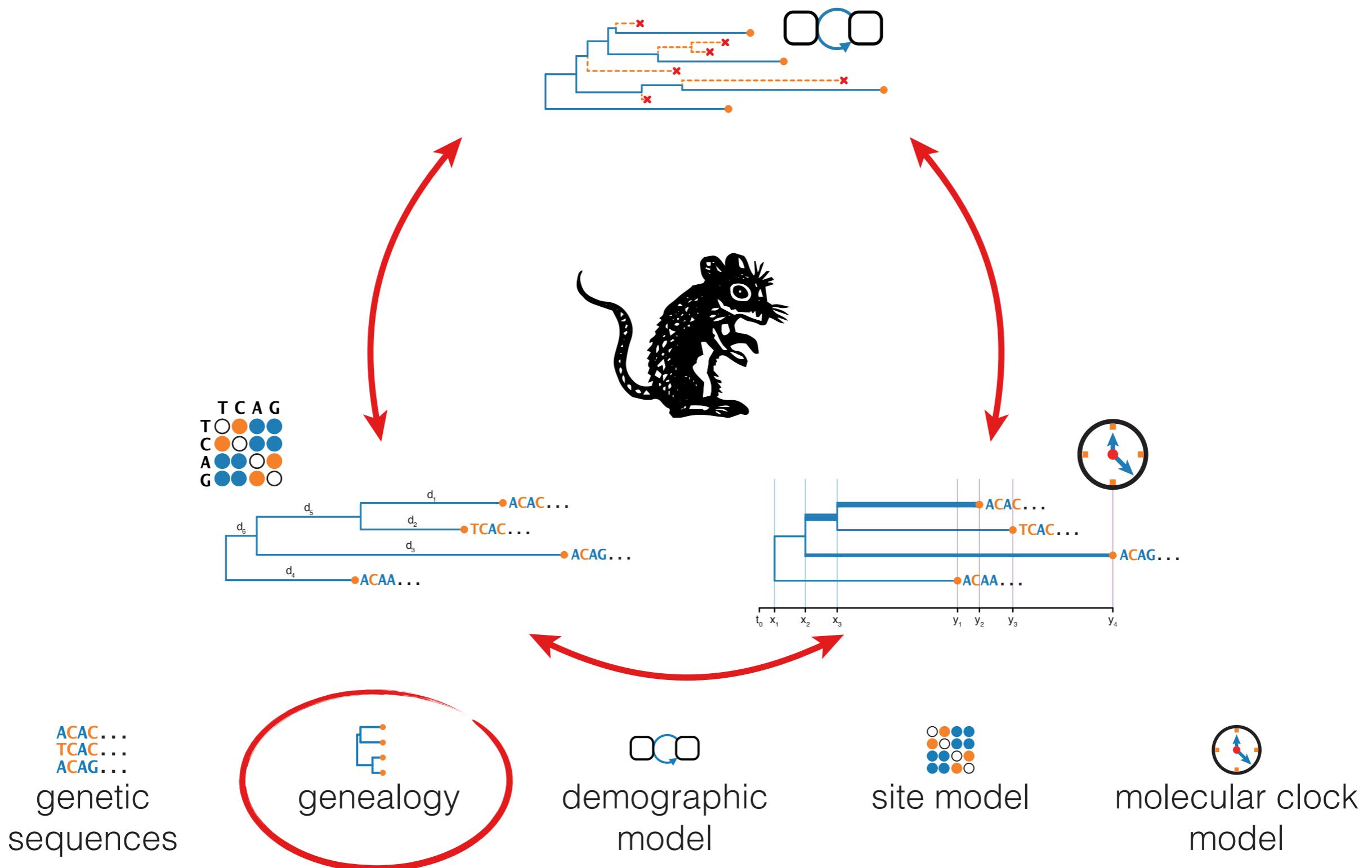
Gavryushkina et al. *Systematic Biology* 2017

Roots of words



Bouckaert et al. *Science* 2012

What goes into a **BEAST** model?



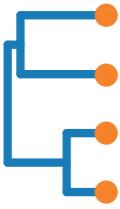
genetic
sequences

genealogy

demographic
model

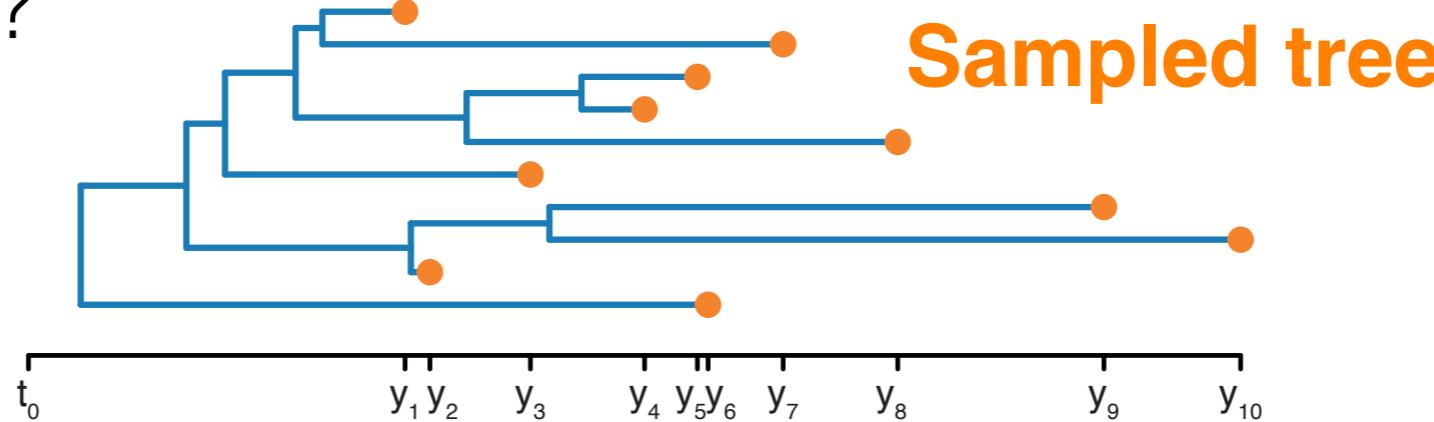
site model

molecular clock
model

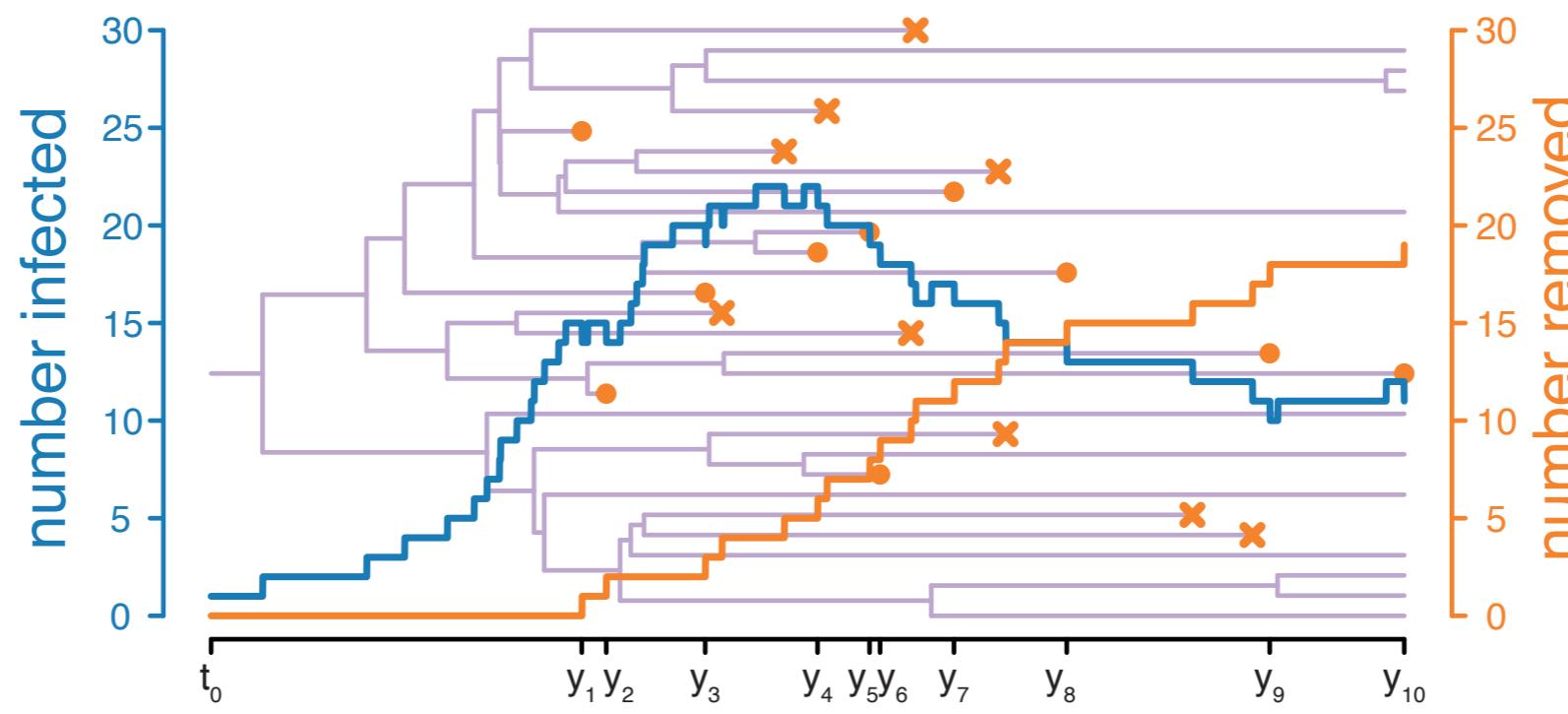


The genealogy (tree)

- What are the ancestral relationships between the sequences in our dataset?



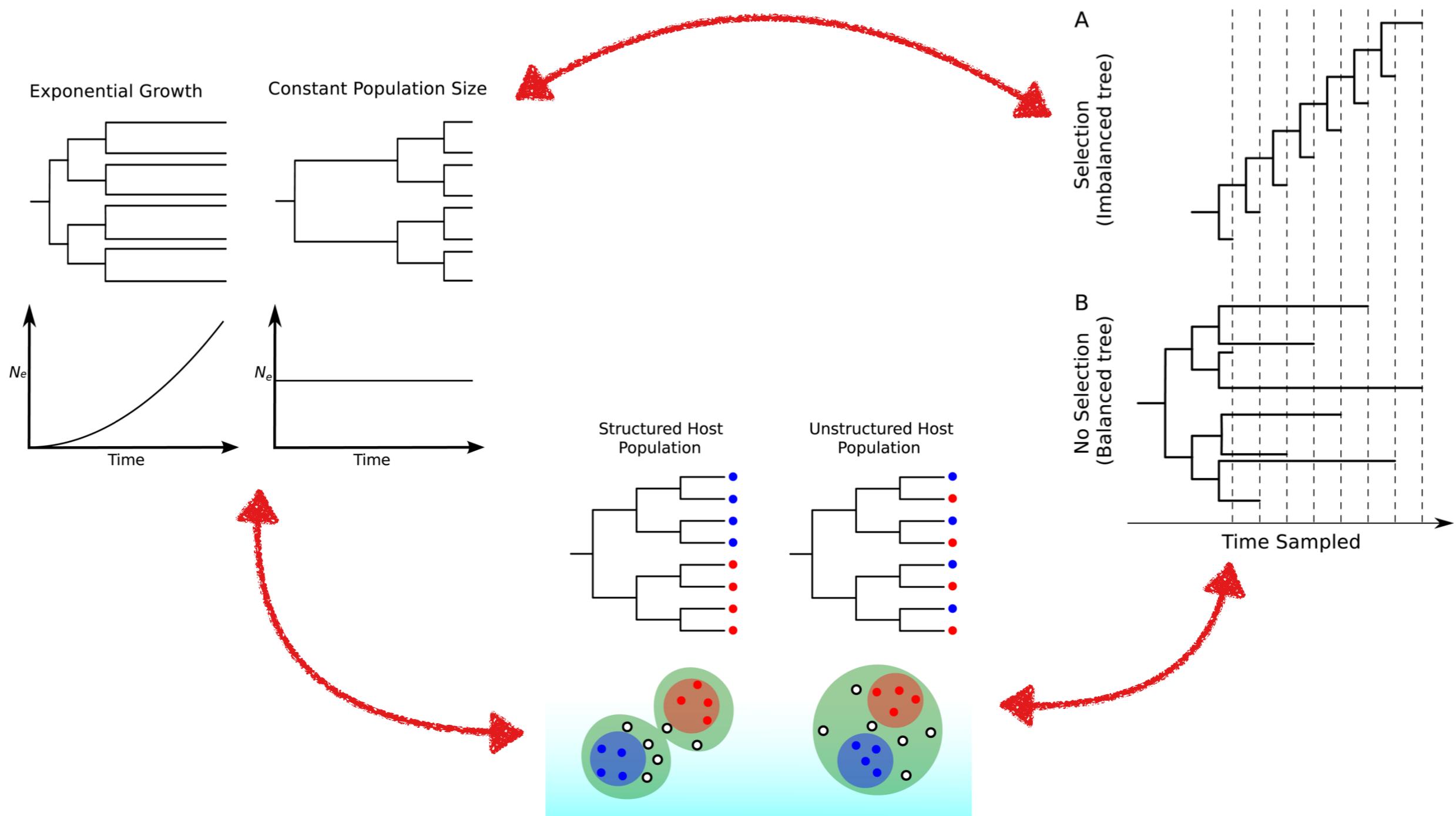
Full tree



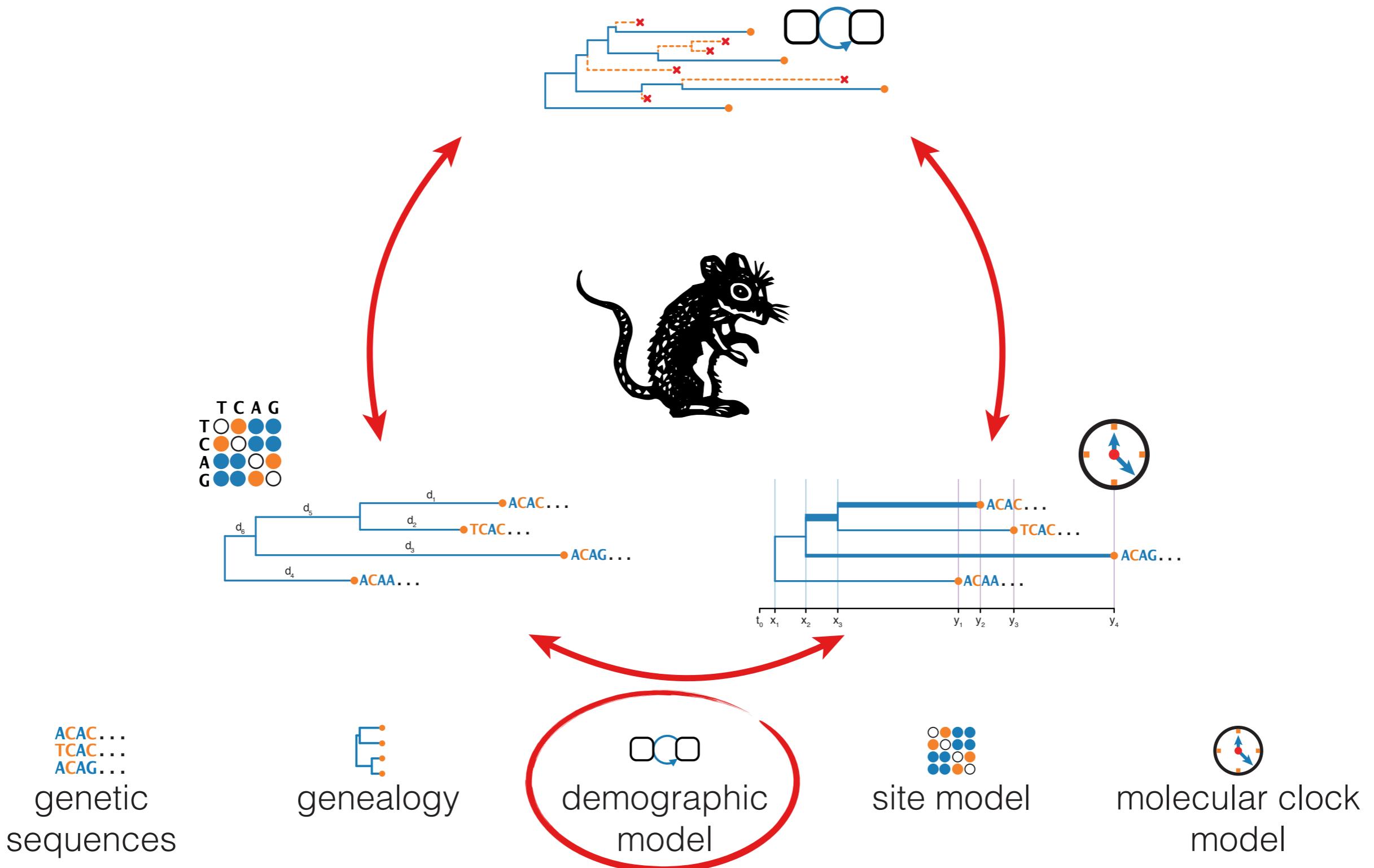
- Only the relationships between the **sampled** sequences!



Different population dynamics generate different trees

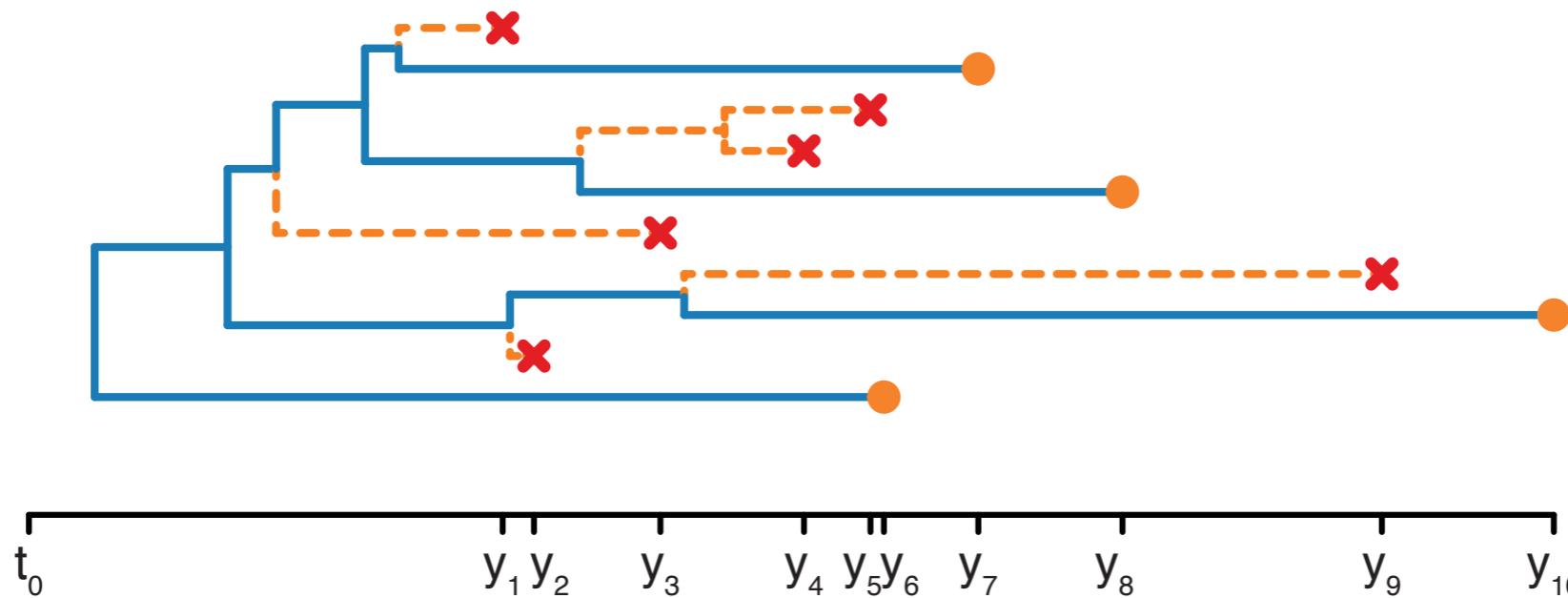


What goes into a **BEAST** model?





Demographic model

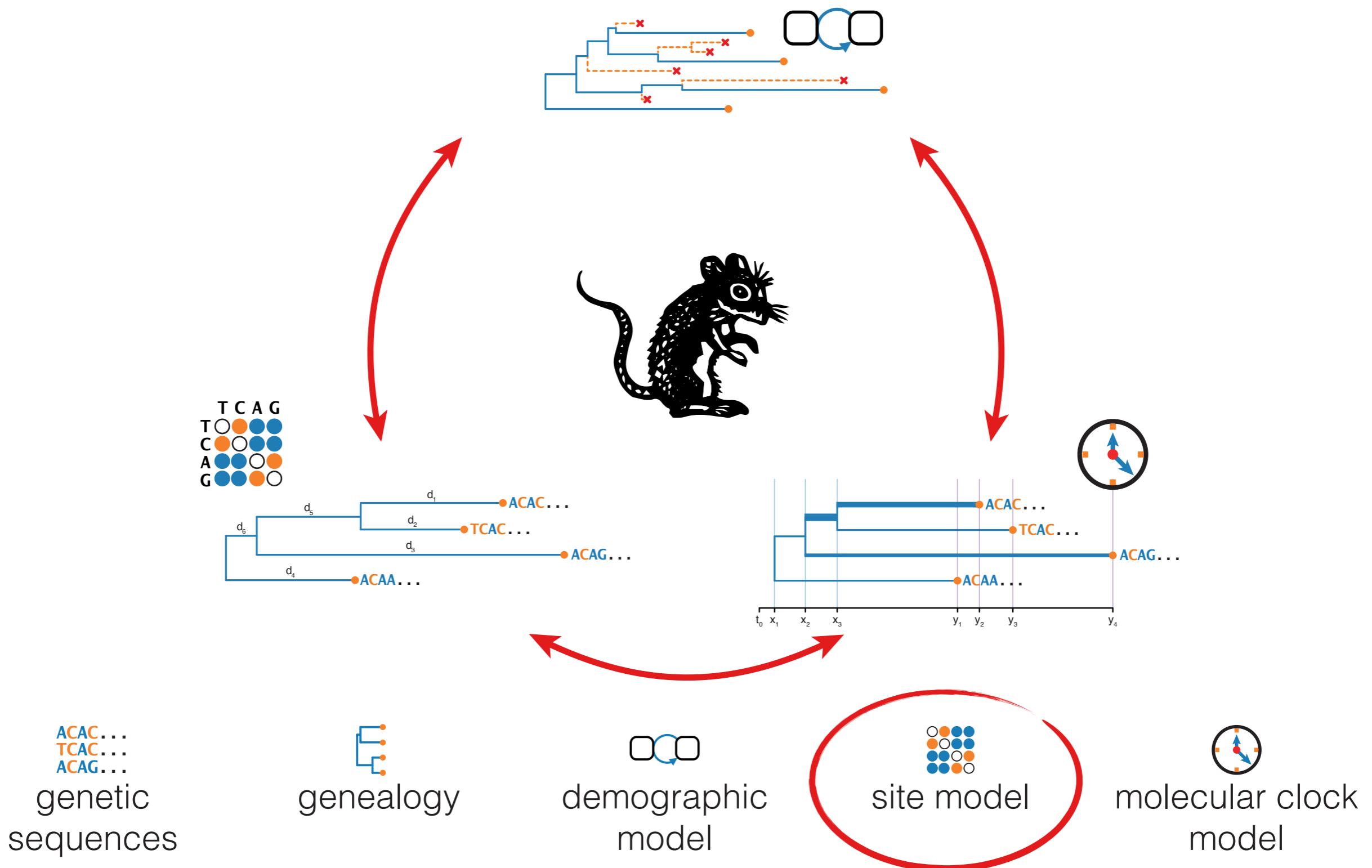


- Describes the population/speciation dynamics
- How does the population grow over time?
- How does the species diversity change over time?

$$P(E | \text{oco}) \xleftarrow{\text{Tree-prior}}$$

- How likely is the genealogy given a demographic model?
- Usually a birth-death or a coalescent model

What goes into a **BEAST** model?



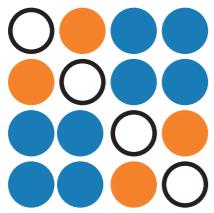
genetic
sequences

genealogy

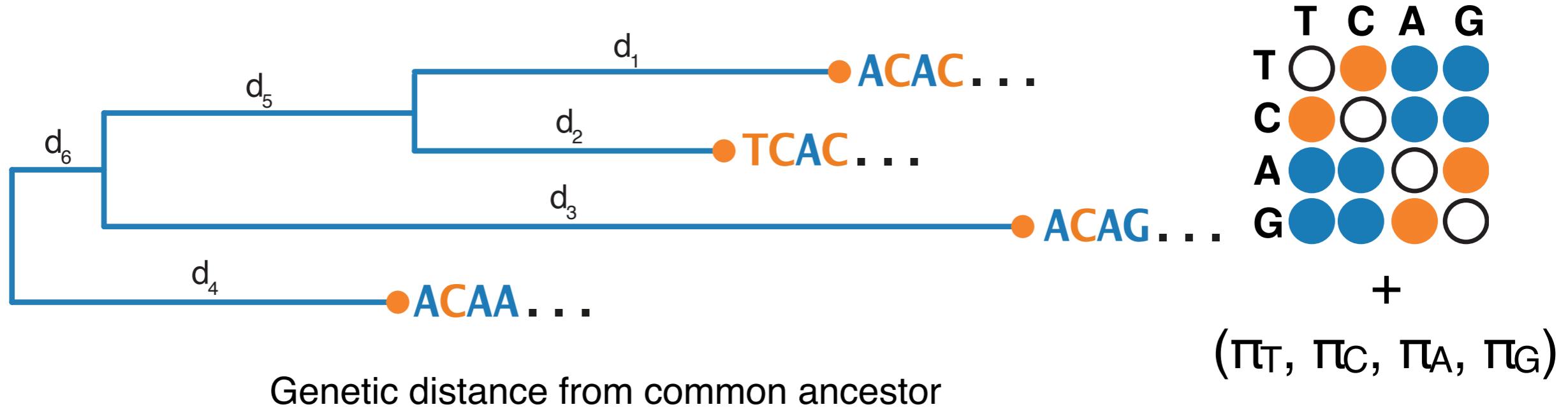
demographic
model

site model

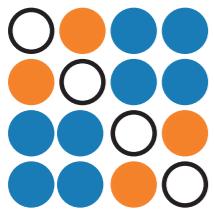
molecular clock
model



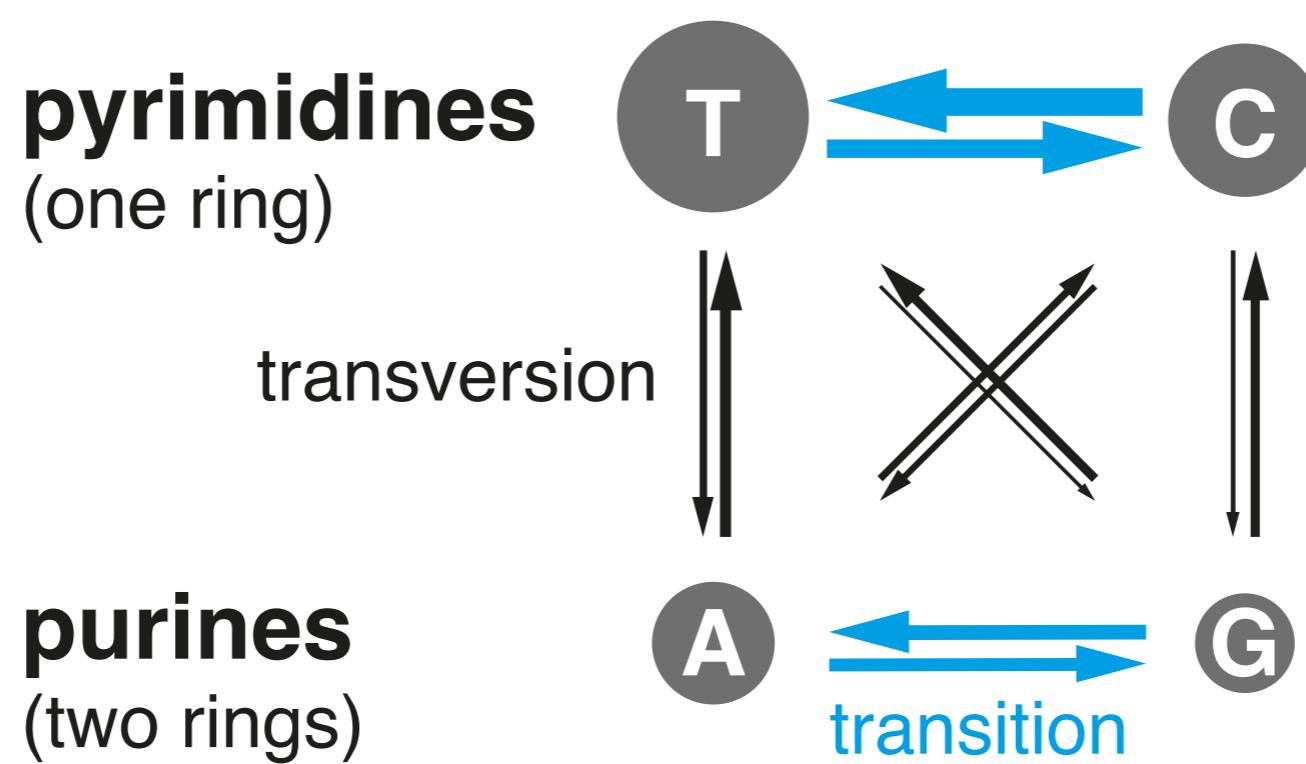
Site model



- Links the genome sequences to the genealogy
- We observe sequences at the tips, not their histories
- Multiple substitutions at the same site means not all substitutions are observed
- To infer the evolutionary history we need to take **all possible evolutionary trajectories** into account!

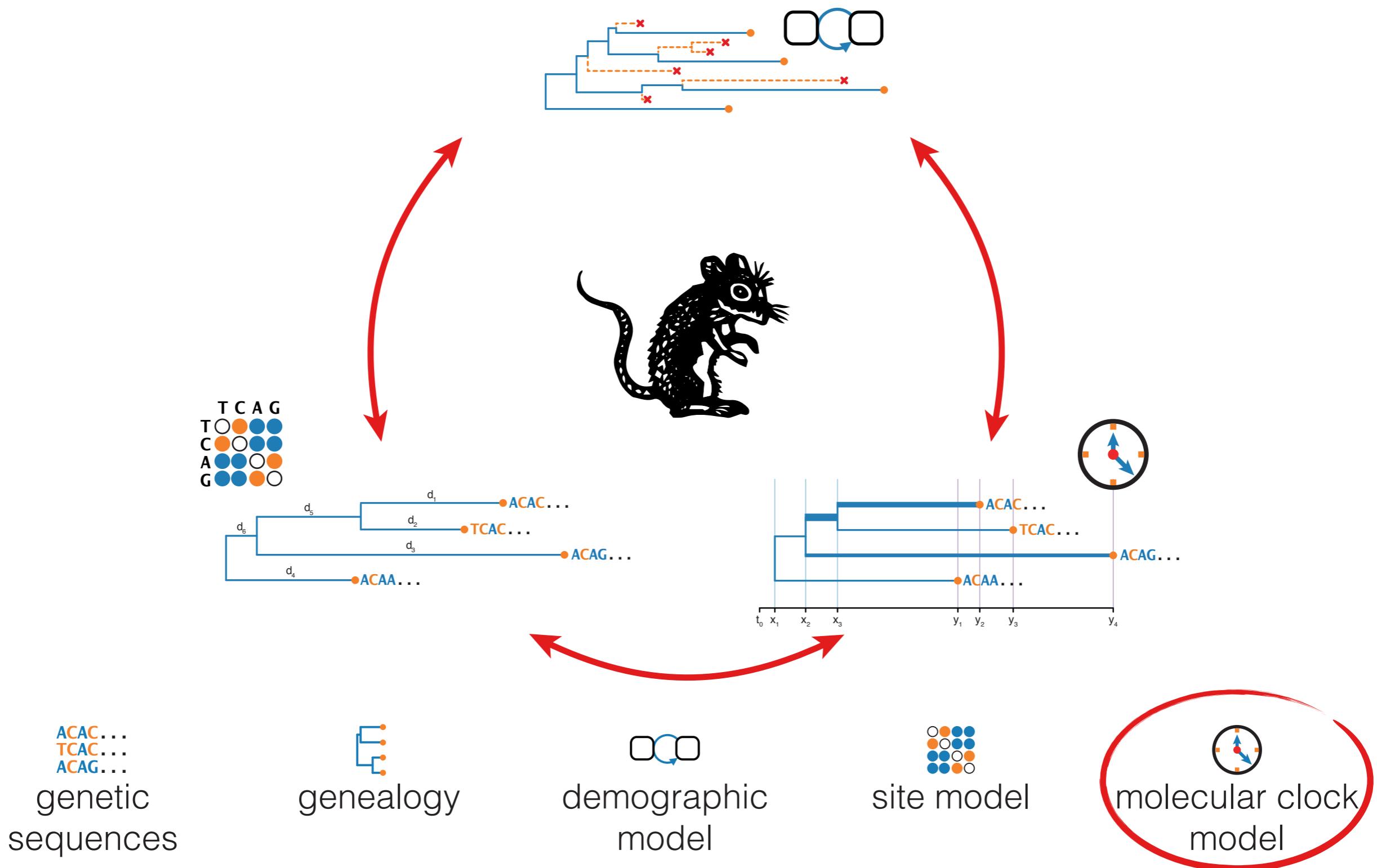


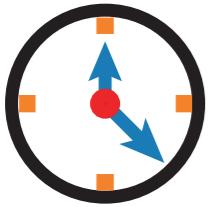
Substitutions as a Markov process



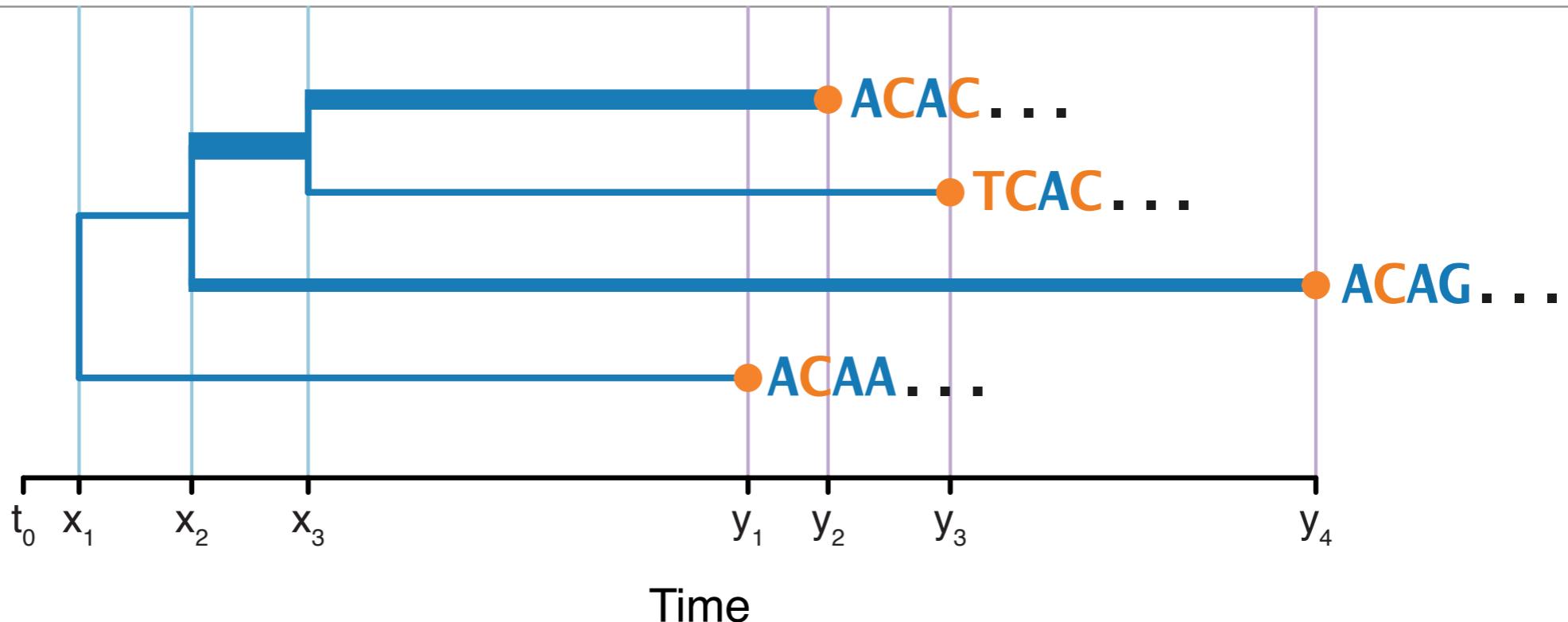
- Assume every site is evolving independently
- Assume nucleotide substitutions at each site is governed by a Markov process
- Account for rate heterogeneity between sites:
 - Proportion of invariant sites
 - Discrete Γ model
 - Multiple partitions with different rates

What goes into a **BEAST** model?





Molecular clock model



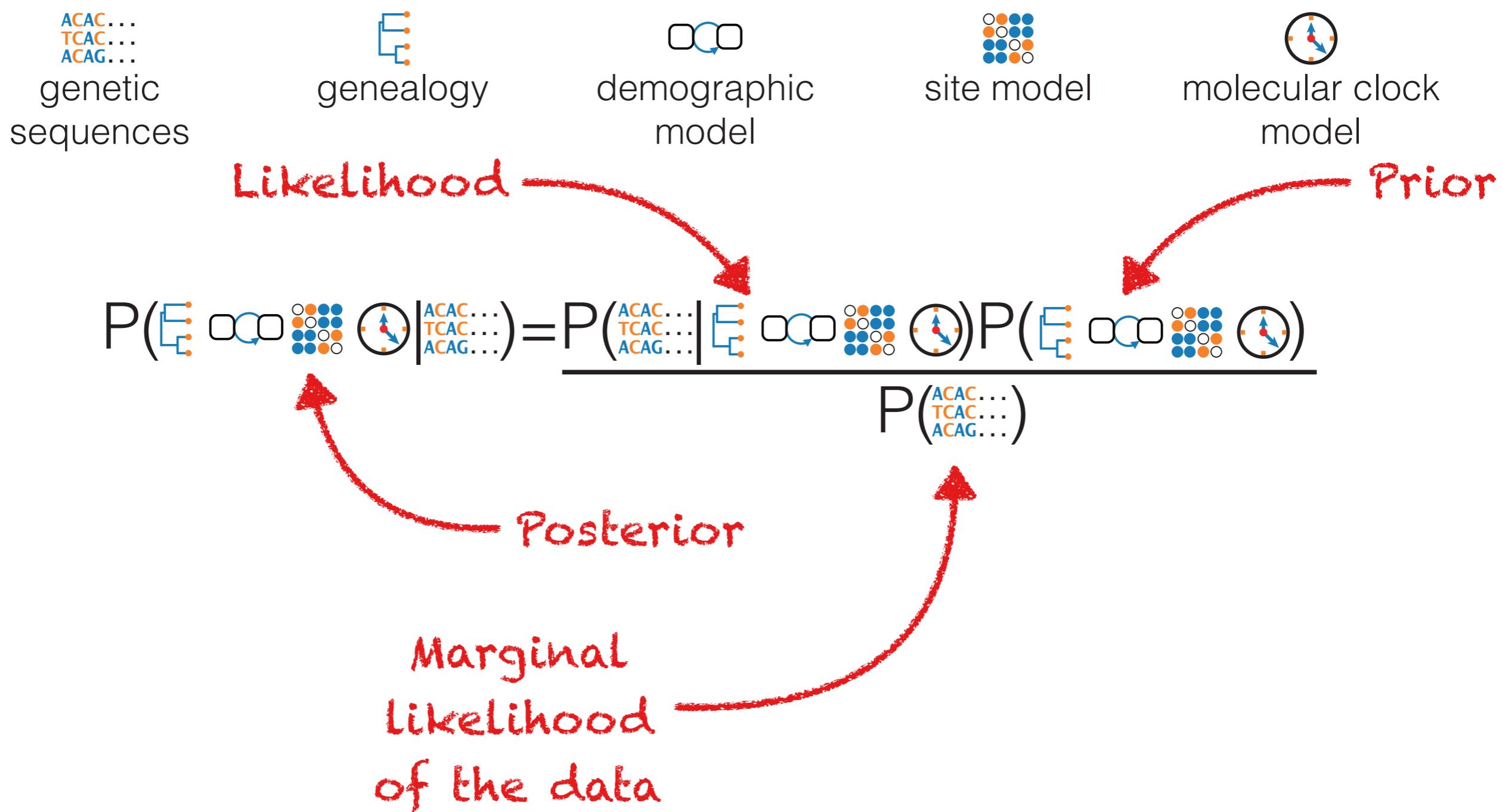
- Scales branch lengths to calendar time
- How long does it take for substitutions to appear?
- Different branches may have different clock rates
- Priors on internal nodes can help to calibrate the clock

Putting it all together



$$P(\text{model} \mid \text{data}) = \frac{P(\text{data} \mid \text{model})P(\text{model})}{P(\text{data})}$$

Putting it all together



Putting it all together

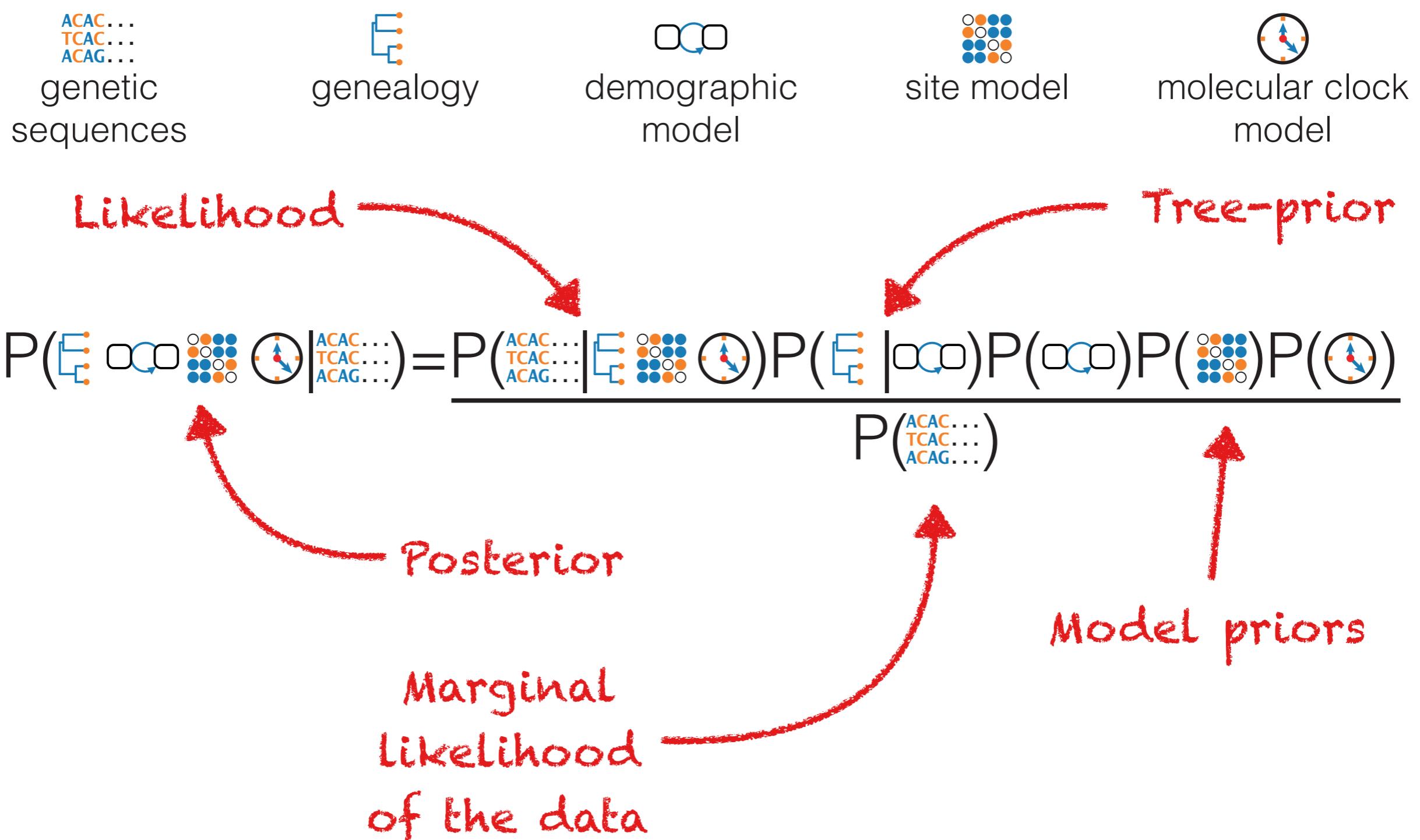


$$P(E \circlearrowleft \circlearrowright \bullet \bullet | ACAC \dots) = \frac{P(ACAC \dots | E \circlearrowleft \circlearrowright \bullet \bullet) P(E \circlearrowleft \circlearrowright \bullet \bullet)}{P(ACAC \dots)}$$

Assume independence

$$P(E \circlearrowleft \circlearrowright \bullet \bullet | \circlearrowleft \circlearrowright \bullet \bullet) = P(E | \circlearrowleft \circlearrowright) P(\circlearrowleft \circlearrowright) P(\bullet \bullet) P(\circlearrowleft \circlearrowright \bullet \bullet)$$

Posterior distribution in BEAST2



How can we find the posterior?

- We want to calculate the posterior distribution

$$P(\text{Sequence} \mid \text{Data}) = \text{Posterior Distribution}$$


- But we cannot easily calculate the marginal likelihood

$$P(\text{Data}) \rightarrow ?$$

→ use **MCMC!** (Markov-chain Monte Carlo)

- MCMC is a stochastic algorithm that performs a random walk on the posterior, preferentially sampling high-density areas

MCMC

(Markov-chain Monte Carlo)

- MCMC draws samples from the posterior
→ output is a list of values that can approximate the posterior
- Only need to compare which posterior density is higher
→ So we only need the ratio of posteriors
(marginal likelihoods cancel out!)

$$\frac{P(\text{model}_1 \mid \text{data})}{P(\text{model}_2 \mid \text{data})} = \frac{\frac{P(\text{data} \mid \text{model}_1)P(\text{model}_1)}{P(\cancel{\text{data}})}}{\frac{P(\text{data} \mid \text{model}_2)P(\text{model}_2)}{P(\cancel{\text{data}})}}$$

Marginal distributions

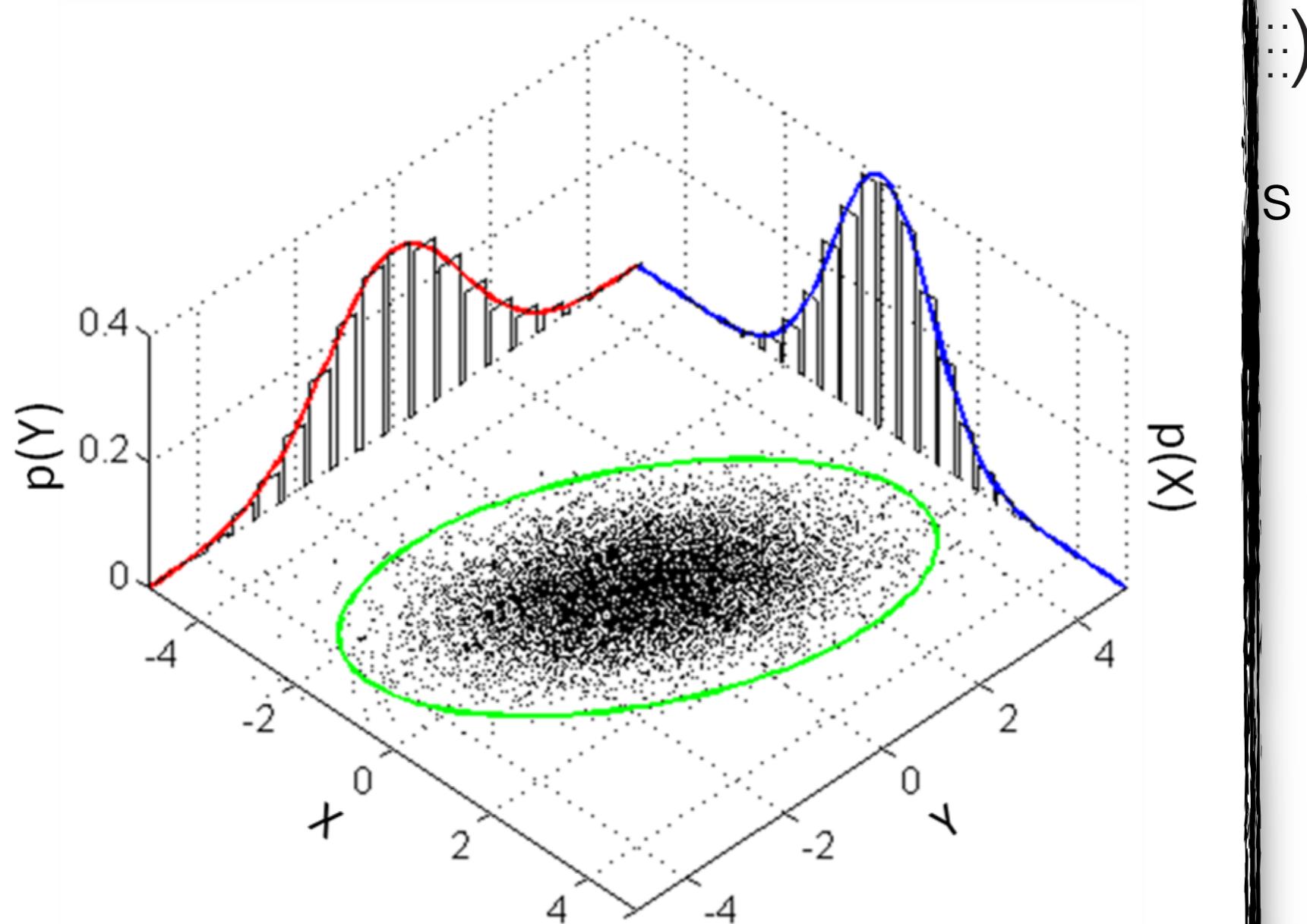
- We only have the joint posterior: $P(\text{E} \text{ } \text{ } \text{ } \text{ } \text{ } | \text{ACAC}, \text{TCAC}, \text{ACAG}, \dots)$
- But we want distributions for each of the parameters we are interested in → marginalize

$$P(\phi) = \int_{\theta} P(\phi|\theta)P(\theta)d\theta$$

Margin

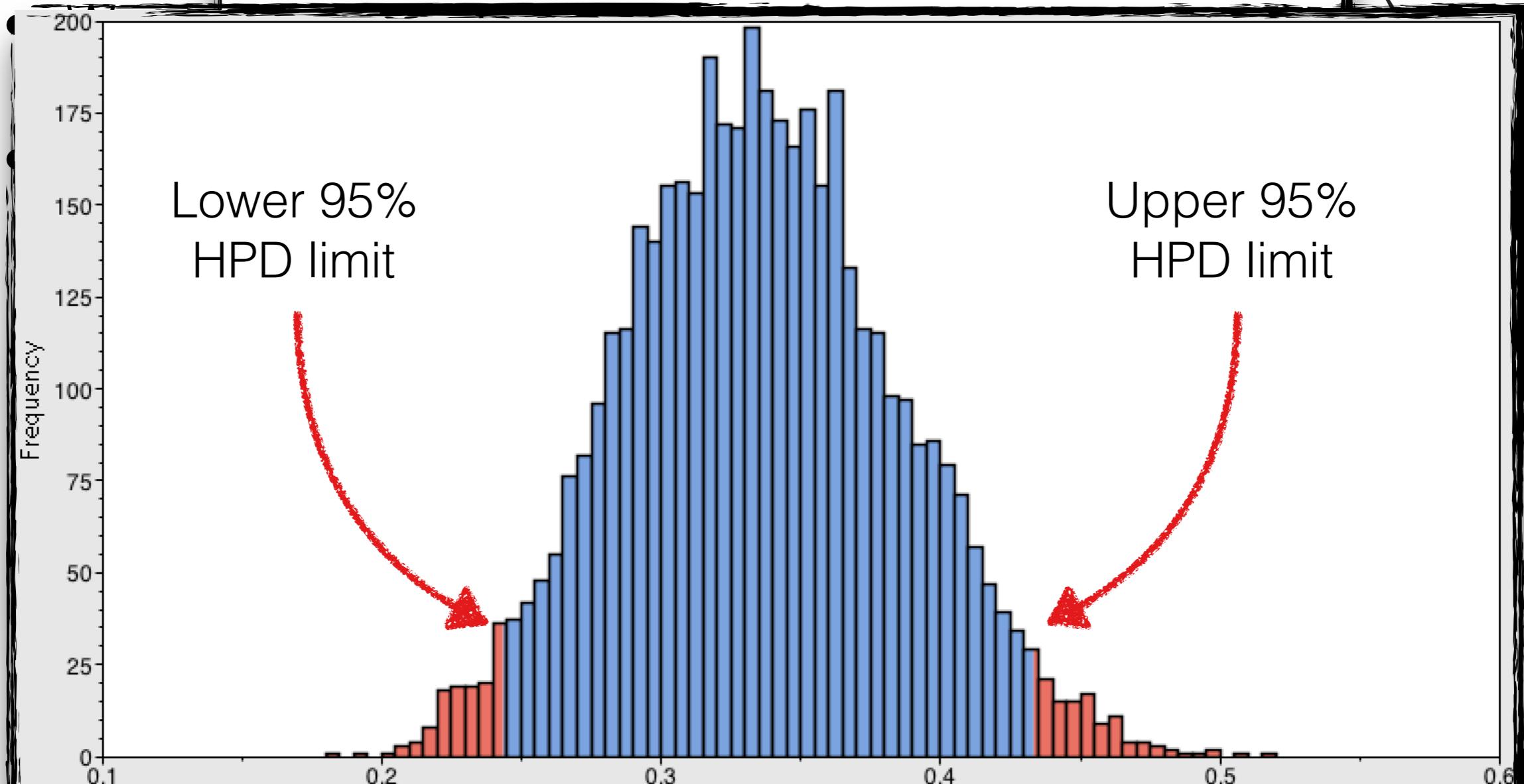
In practice

- We often
- But we
- we are



Margin

In practice



Operators

Target distribution



Proposal distribution

- Used to decide where to step to next
 - The choice only affects the **efficiency** of the algorithm
 - In BEAST1 and BEAST2 operators are used to propose the next step
 - A parameter (or multiple parameters) are selected and perturbed to propose a step

Operators

Target distribution

Pro

- Operators are a part of the MCMC **algorithm**, not the **model!**
 - Tuning operators can help to improve efficiency, but should not change the results

the next step

- A parameter (or multiple parameters) are selected and perturbed to propose a step

MCMC in practice

Before

- Decide on the length of the chain
(total number of steps to take)
- Decide on the sampling frequency
(how often to record samples so
that they are uncorrelated)

After

- Discard burn-in
(until stationary state is reached)
- Assess convergence and mixing

MCMC in practice

Before

- Decide on the length of the chain
(total number of steps to take)
- Decide on the sampling frequency
(how often to record samples so that they are uncorrelated)

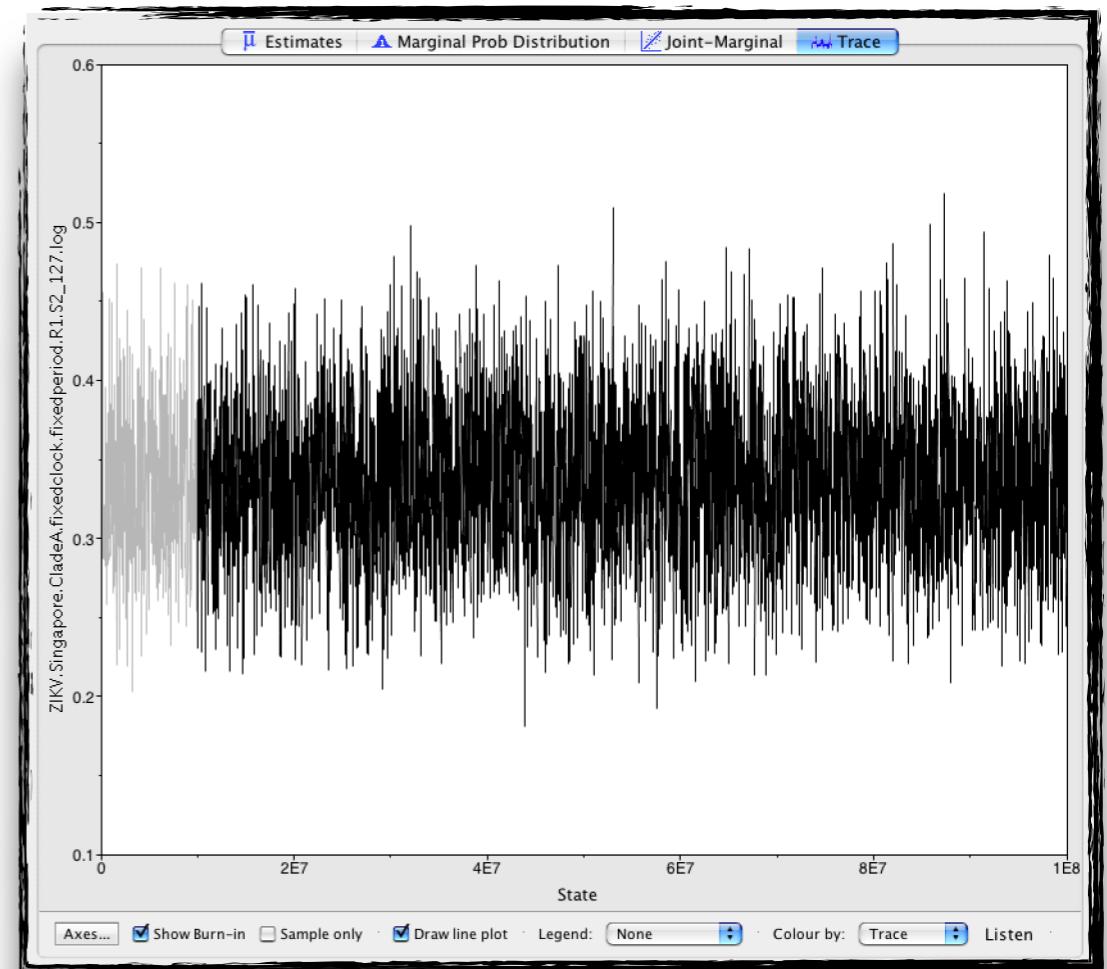
After

- Assess convergence and mixing
 - stationary state is reached)

More than 10,000 samples is a waste of space
(but need to sample at the right frequency)

What we hope will happen

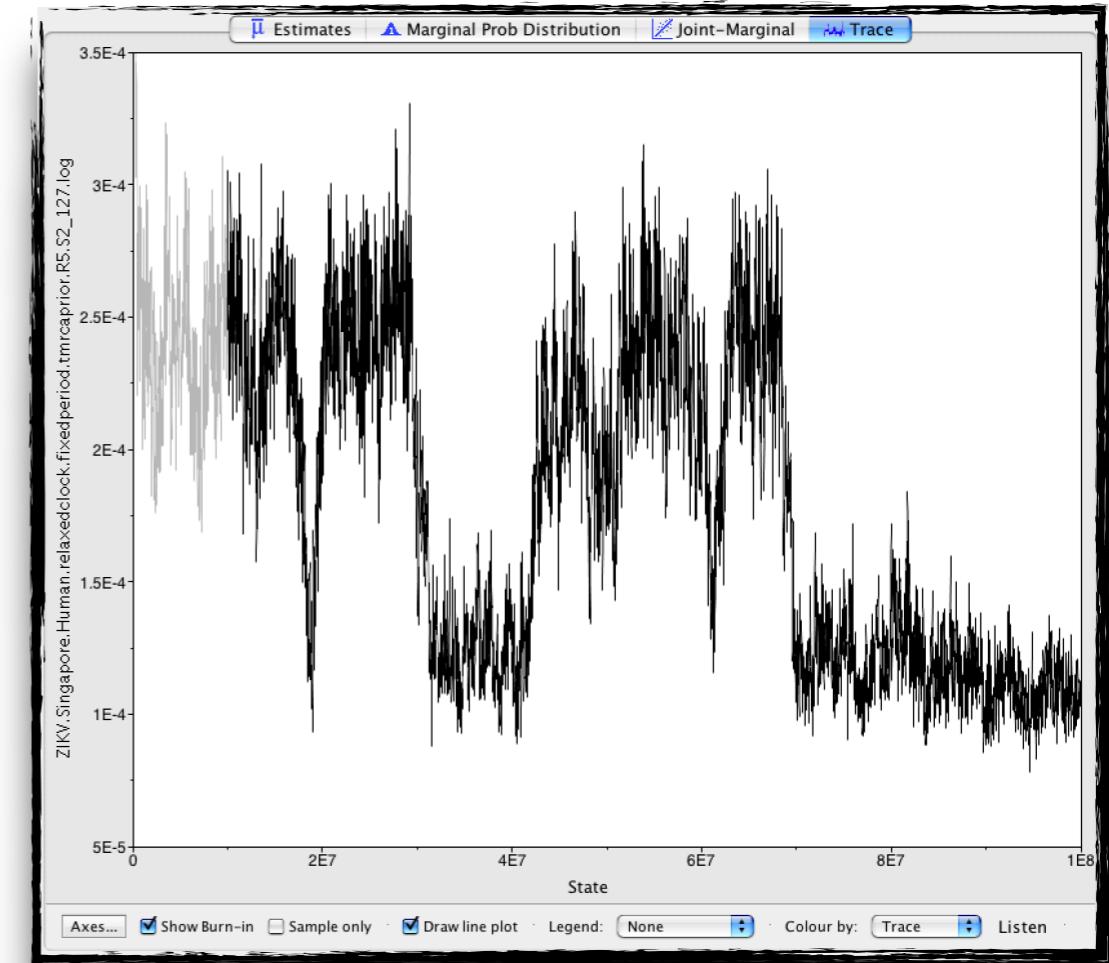
- The MCMC algorithm samples efficiently from high density areas of the posterior distribution
- We end up with a **good** approximation of the posterior distribution in **finite** time
- Appearance of white noise
- Everything is awesome!



Mixing well! 😊

Questions to ask...

- Is the chain **mixing** well?
- Are samples uniformly drawn from all over the stationary distribution?
- “Sticky chain”



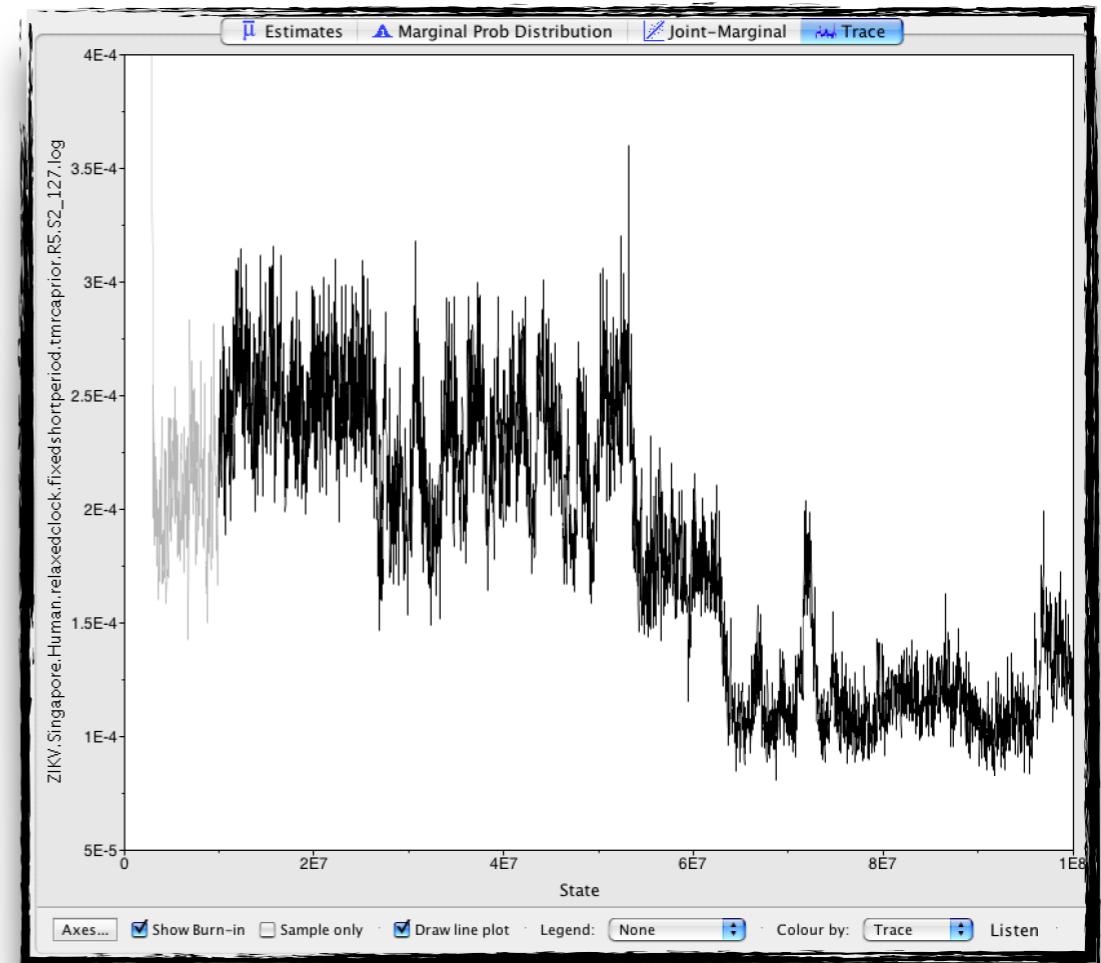
Solutions

- MCMC gets stuck in some states for long times
- Tune operators to make better proposals

Not mixing! 😞

Questions to ask...

- Has the chain **converged** to the stationary distribution?
- Did we pass the burn-in?

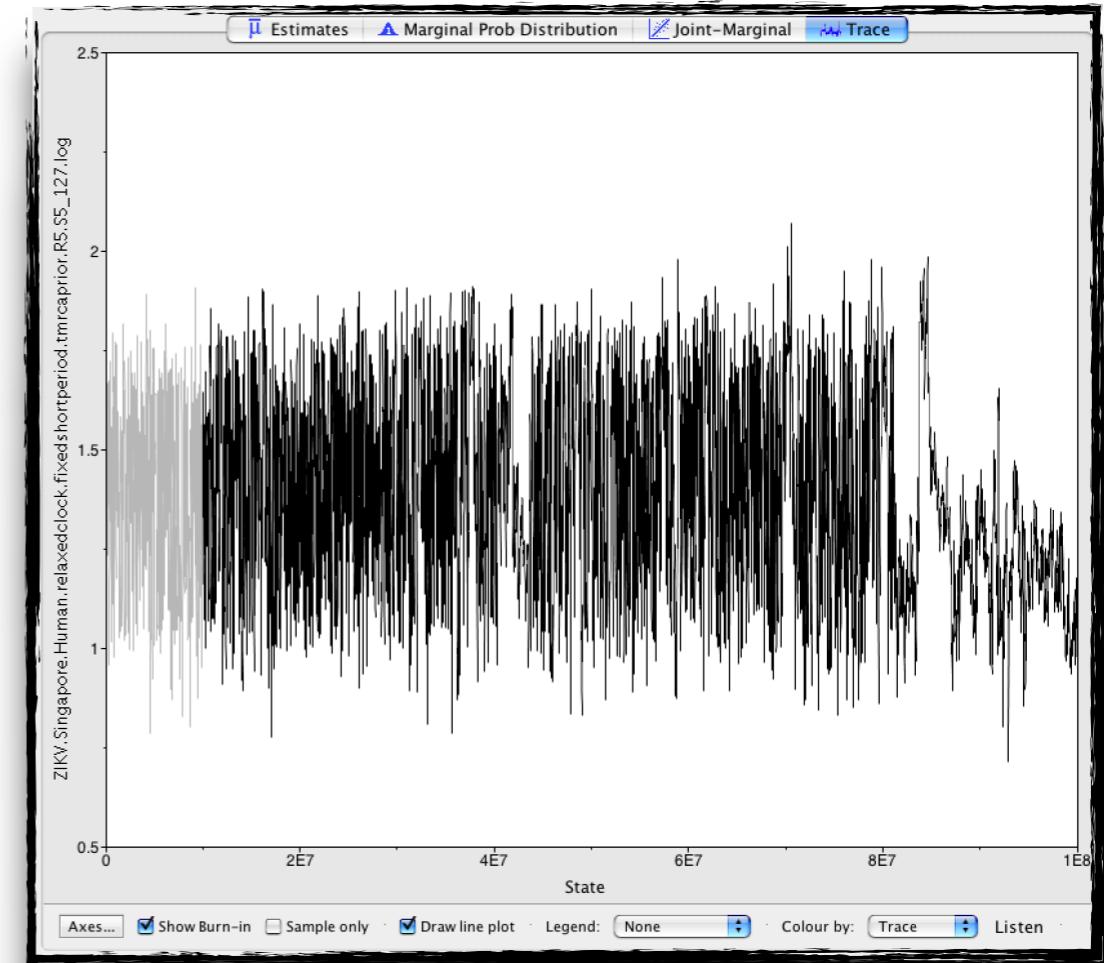


Not converged! 😓

Solution: Run for longer

Questions to ask...

- Are we there yet?
- How do we know if the chain is long enough?



Solution

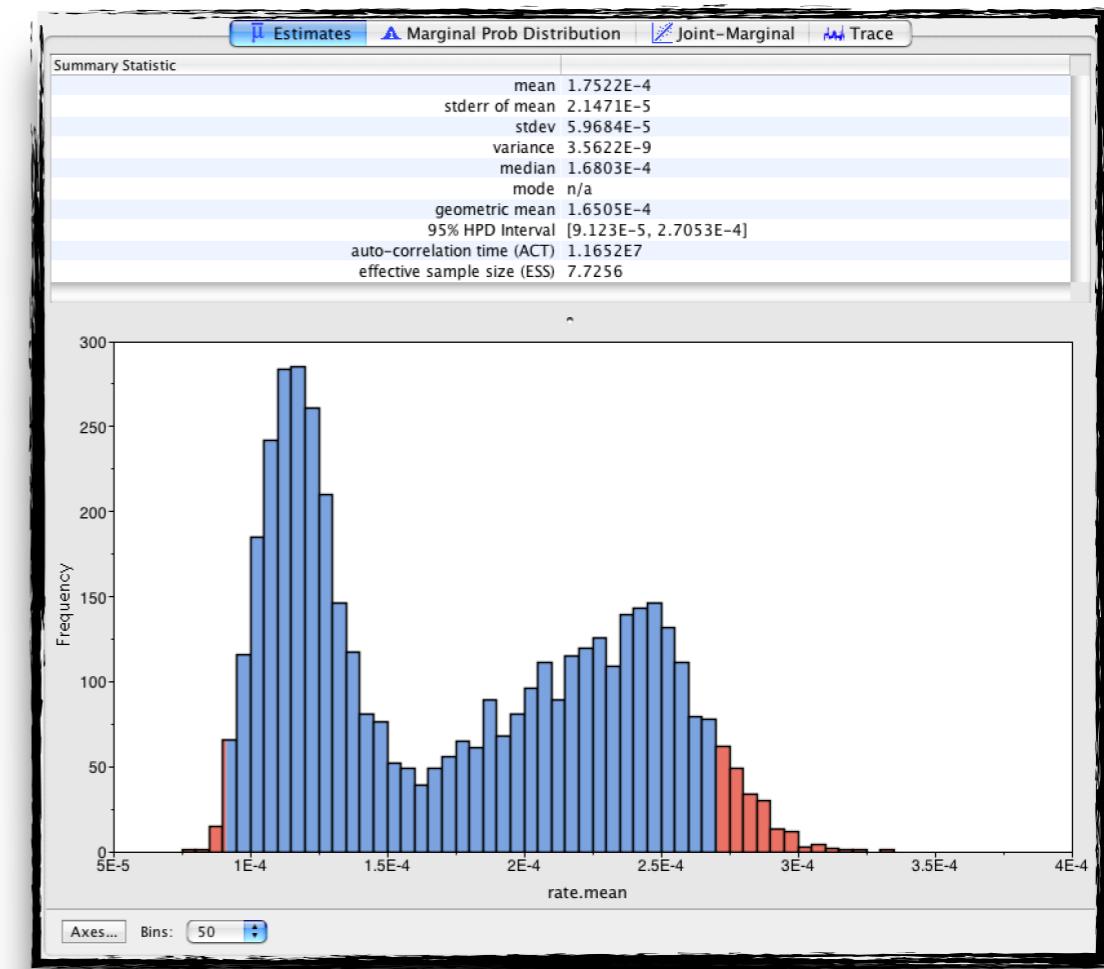
- Run multiple chains
- Combine chains
- Check that all chains give the same result

Still not converged! 😞

What if the answer is not what we wanted?

What is happening here?

- If the chain converged and mixed well then this is due to the data and model choice
- The model supports a bimodal posterior distribution
- May not be the answer we wanted but it may be the truth
- Should we change the model or parameterisation?
- How would we know if another model fits better?



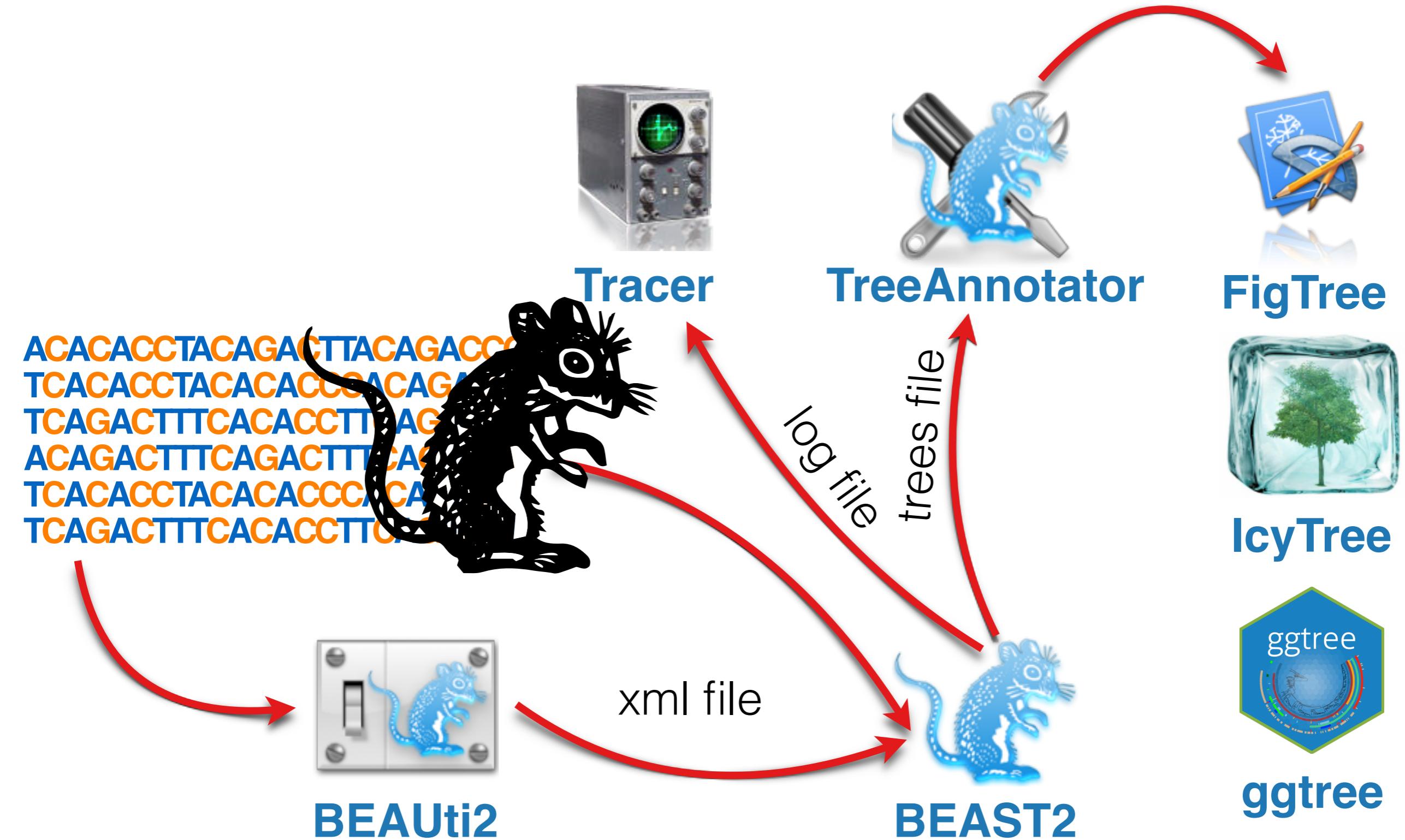
Is this a problem? 🤔

Solution: Be more open-minded

FANTASTIC BEASTS

AND WHERE
TO FIND THEM

BEAST2 workflow



BEAUti2

(<http://beast2.org>)



GUI for setting up BEAST2 input file in xml format

Input:

- Sequence alignment

Output:

- BEAST2 configuration file (xml file)

BEAUti 2: Standard /Users/louis/Documents/Taming_the_BEAST/Tutorials-Git/Introduction-to-BEAST2/xml/Primates.xml

Partitions Tip Dates Site Model Clock Model Priors MCMC

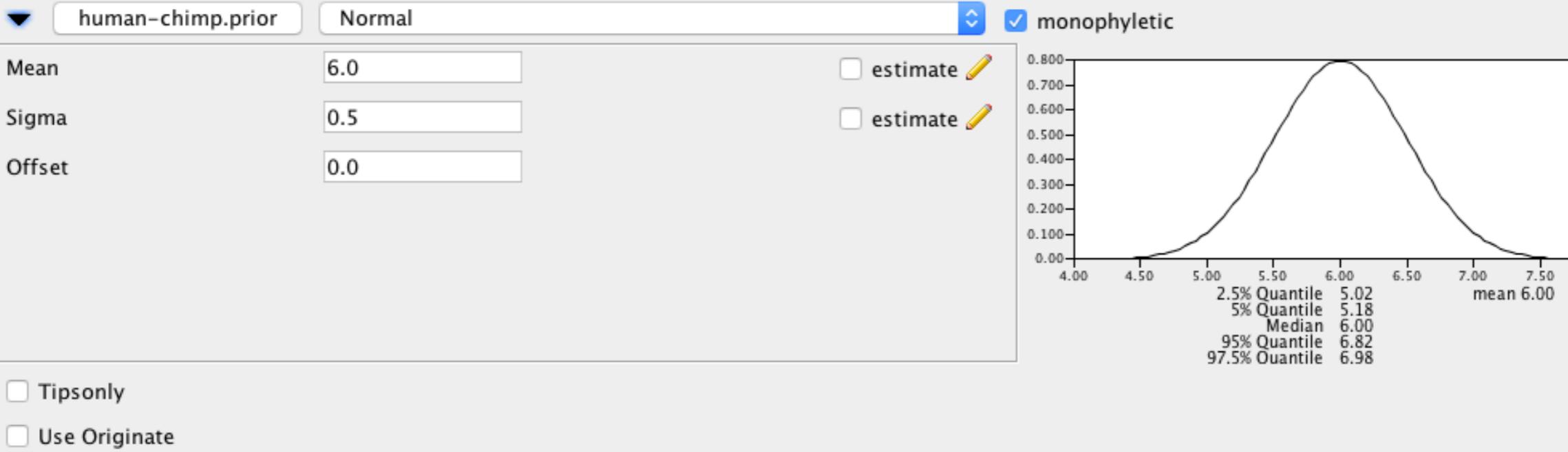
Link Site Models Unlink Site Models Link Clock Models Unlink Clock Models Link Trees Unlink Trees

Name	File	Taxa	Sites	Data Type	Site Model	Clock Model	Tree	...
noncoding	primate-mtDNA	12	205	nucleotide	noncoding	clock	tree	
1stpos	primate-mtDNA	12	231	nucleotide	1stpos	clock	tree	
2ndpos	primate-mtDNA	12	231	nucleotide	2ndpos	clock	tree	
3rdpos	primate-mtDNA	12	231	nucleotide	3rdpos	clock	tree	

+ - r Split

Partitions Tip Dates Site Model Clock Model Priors MCMC

▶ Tree.t:tree	Calibrated Yule Model		
▶ birthRateY.t:tree	Gamma	initial = [1.0] $[-\infty, \infty]$	Calibrated Yule speciation process birth rate for t:3rdpos
▶ clockRate.c:clock	Uniform	initial = [1.0] $[-\infty, \infty]$	substitution rate of partition c:3rdpos
▶ gammaShape.s:1stpos	Exponential	initial = [1.0] $[-\infty, \infty]$	Prior on gamma shape for partition s:1stpos
▶ gammaShape.s:2ndpos	Exponential	initial = [1.0] $[-\infty, \infty]$	Prior on gamma shape for partition s:2ndpos
▶ gammaShape.s:3rdpos	Exponential	initial = [1.0] $[-\infty, \infty]$	Prior on gamma shape for partition s:3rdpos
▶ gammaShape.s:noncoding	Exponential	initial = [1.0] $[-\infty, \infty]$	Prior on gamma shape for partition s:noncoding
▶ kappa.s:1stpos	Log Normal	initial = [2.0] $[0.0, \infty]$	HKY transition-transversion parameter of partition s:1stpos
▶ kappa.s:2ndpos	Log Normal	initial = [2.0] $[0.0, \infty]$	HKY transition-transversion parameter of partition s:2ndpos
▶ kappa.s:3rdpos	Log Normal	initial = [2.0] $[0.0, \infty]$	HKY transition-transversion parameter of partition s:3rdpos
▶ kappa.s:noncoding	Log Normal	initial = [2.0] $[0.0, \infty]$	HKY transition-transversion parameter of partition s:noncoding



Primates_long.xml UNREGISTERED

```
39
40 <run id="mcmc" spec="MCMC" chainLength="2500000">
41   <state id="state" storeEvery="5000">
42     <tree id="Tree.t:tree" name="stateNode">
43       <taxonset id="TaxonSet.noncoding" spec="TaxonSet">
44         <alignment id="noncoding" spec="FilteredAlignment" filter="1,458-659,897-898">
45           <data idref="primate-mtDNA"/>
46         </alignment>
47       </taxonset>
48     </tree>
49     <parameter id="mutationRate.s:noncoding" name="stateNode">1.0</parameter>
50     <parameter id="gammaShape.s:noncoding" name="stateNode">1.0</parameter>
51     <parameter id="kappa.s:noncoding" lower="0.0" name="stateNode">2.0</parameter>
52     <parameter id="kappa.s:1stpos" lower="0.0" name="stateNode">2.0</parameter>
53     <parameter id="gammaShape.s:1stpos" name="stateNode">1.0</parameter>
54     <parameter id="mutationRate.s:1stpos" name="stateNode">1.0</parameter>
55     <parameter id="kappa.s:2ndpos" lower="0.0" name="stateNode">2.0</parameter>
56     <parameter id="gammaShape.s:2ndpos" name="stateNode">1.0</parameter>
57     <parameter id="mutationRate.s:2ndpos" name="stateNode">1.0</parameter>
58     <parameter id="kappa.s:3rdpos" lower="0.0" name="stateNode">2.0</parameter>
59     <parameter id="gammaShape.s:3rdpos" name="stateNode">1.0</parameter>
60     <parameter id="mutationRate.s:3rdpos" name="stateNode">1.0</parameter>
61     <parameter id="birthRateY.t:tree" name="stateNode">1.0</parameter>
62     <parameter id="clockRate.c:clock" name="stateNode">1.0</parameter>
63   </state>
64
65   <init id="RandomTree.t:tree" spec="beast.evolution.tree.RandomTree" estimate="false" initial="@Tree.t:tree" taxa="@noncoding">
66     <populationModel id="ConstantPopulation0.t:tree" spec="ConstantPopulation">
67       <parameter id="randomPopSize.t:tree" name="popSize">1.0</parameter>
68     </populationModel>
69   </init>
70
71   <distribution id="posterior" spec="util.CompoundDistribution">
72     <distribution id="prior" spec="util.CompoundDistribution">
73       <distribution id="CalibratedYuleModel.t:tree" spec="beast.evolution.speciation.CalibratedYuleModel" birthRate="@birthRateY.t:tree" tree="@Tree.t:tree"/>
74       <prior id="CalibratedYuleBirthRatePrior.t:tree" name="distribution" x="@birthRateY.t:tree">
75         <Gamma id="Gamma.0" name="distr">
76           <parameter id="RealParameter.0" estimate="false" name="alpha">0.001</parameter>
77           <parameter id="RealParameter.01" estimate="false" name="beta">1000.0</parameter>
78         </Gamma>
79       </prior>
80       <prior id="ClockPrior.c:clock" name="distribution" x="@clockRate.c:clock">
81         <Uniform id="Uniform.0" name="distr" upper="Infinity"/>
82       </prior>
83     </distribution>
84   </distribution>
85 </run>
```

Line 1, Column 3

0 misspelled words

Spaces: 4

XML

BEAST2

(<http://beast2.org>)



- Bayesian **e**volutionary **a**nalysis by **s**ampling **t**rees
- Performs MCMC analyses of sequences under selected sequence evolution and tree (epidemiological/speciation) model
- Similar to BEAST 1.8.4/1.10 but completely separate
- BEAST2 and BEAST1 have a common origin, have much of the same functionality but each has their own specific niche
- BEAST2 has a modular design that makes it easy to extend

Input:

- xml file

Outputs:

- log file
- trees file
- state file

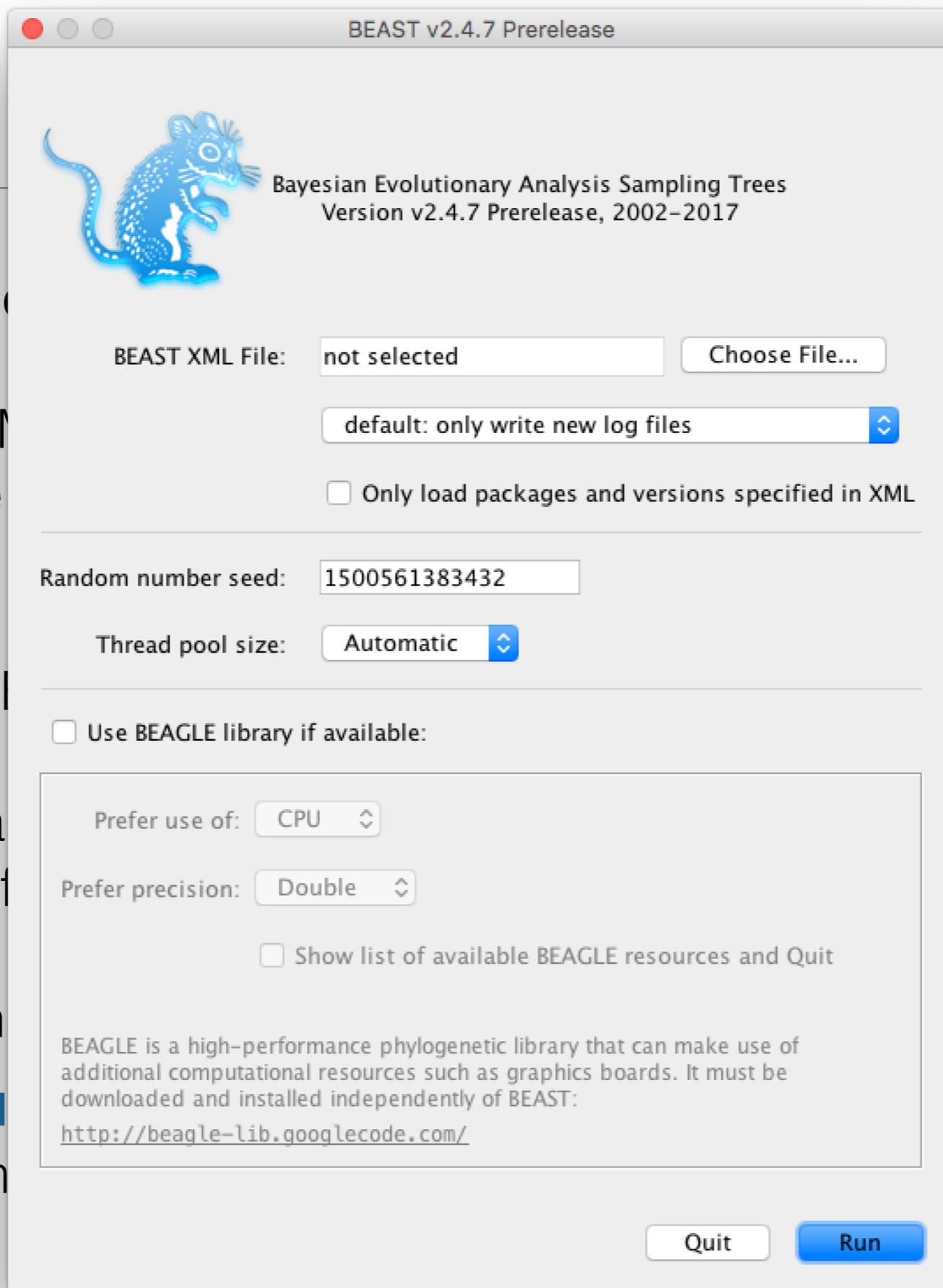
BEAST2

(<http://beast2.org>)

- Bayesian computation
- Performs MCMC sampling under a sequence evolution model
- Similar to JModelTest
- BEAST2 also performs the same function
- BEAST2 has a graphical user interface

Input

- XML files



- state file



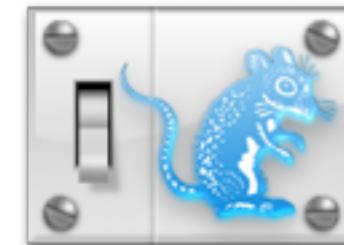
selected
speciation)

ate

e much of
cific niche

to extend

BEAST2 packages



- Independent researchers can develop their own BEAST2 packages
- Packages can be frequently updated without waiting for the next BEAST2 release
- Packages add new models or completely new functionality
- Phylogeography, bacterial ARG inference, morphological models, model selection and averaging, stochastic simulations etc.
- Install new packages through BEAUti2



BEAST2 packages

- Index of available packages for BEAST v2.4.*
 - Packages for the next release
 - Packages for quality control
 - Phylogenetic models for simulation
 - Install new packages through BEAUti2
- BEAST 2 Package Manager

List of available packages for BEAST v2.4.*

Name	Installed	Latest	Dependencies	Link	Detail
BEAST	2.4.7	2.4.7			BEAST core
bacter	1.2.1	1.2.1			Bacterial ARG inference.
BASTA		2.3.1			Bayesian structured coalescent approximation
bdmm	0.2.0	0.2.0	MultiTypeTree		pre-release of multitype birth-death model (aka birth-death skyline - handles serially sampled tips, piec...
BDSKY	1.3.3	1.3.3			birth death skyline - handles serially sampled tips, piec...
BEAST_CLASSIC	1.3.0	1.3.0	BEASTLabs		BEAST classes ported from BEAST 1 in wrappers
BEASTLabs	1.7.0	1.7.1			BEAST utilities, such as Script, multi monophyletic c...
BEASTShell		1.3.0			BEAST Shell - BeanShell scripting for BEAST
BEASTvntr		0.1.1			Variable Number of Tandem Repeat data, such as micro...
bModelTest	1.0.4	1.0.4	BEASTLabs		Bayesian model test for nucleotide subst models, gamm...
CA		1.2.1			CladeAge aPackage for fossil calibrations
DENIM		0.3.0			Divergence Estimation Notwithstanding ILS and Migration
EpiInf		5.0.1	SA		Inference of epidemic trajectories
GEO_SPHERE		1.1.2	BEASTLabs		Whole world phylogeography
Mascot		0.0.2			Marginal approximation of the structured coalescent
MASTER		5.1.1			Stochastic population dynamics simulation
MGSM		0.2.1			Multi-gamma and relaxed gamma site models
MM		1.0.5			Enables models of morphological character evolution
MODEL_SELECTION		1.3.4	BEASTLabs		Select models through path sampling/stepping stone an...

Latest [Install/Upgrade](#) [Uninstall](#) [Package repositories](#) [Close](#) ?

Tracer

(<http://beast.community>)



- Analyse log files from BEAST2 runs
- Check mixing, ESS, ACT, parameter correlations
- Overview of posterior parameter estimates
- Tracer is a diagnostic tool **only** — perform final analyses in a statistical package like R!
- Comparisons of several analyses

Input:

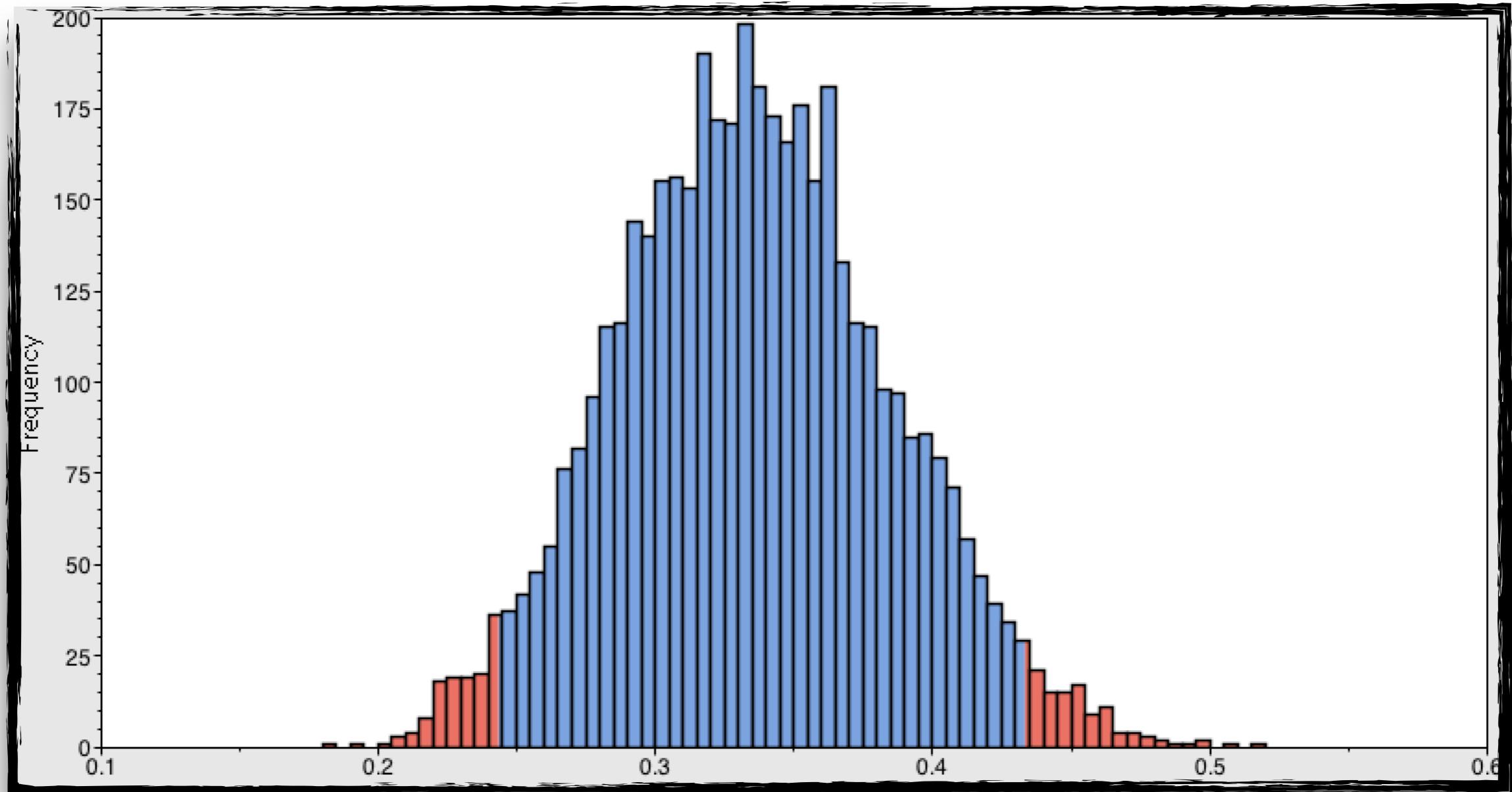
- log file

Output:

- Gain insight

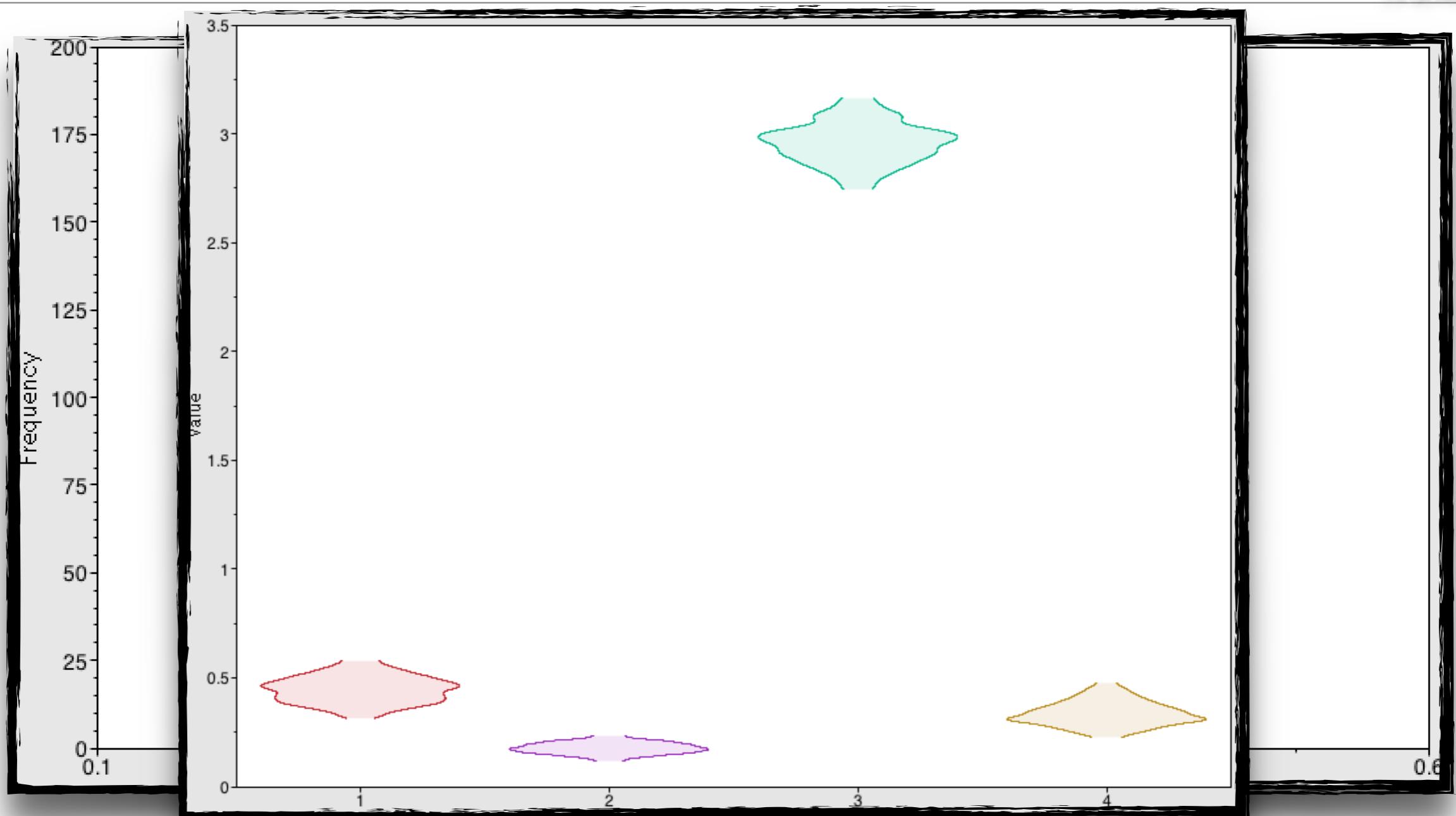
Tracer

(<http://beast.community>)



Tracer

(<http://beast.community>)

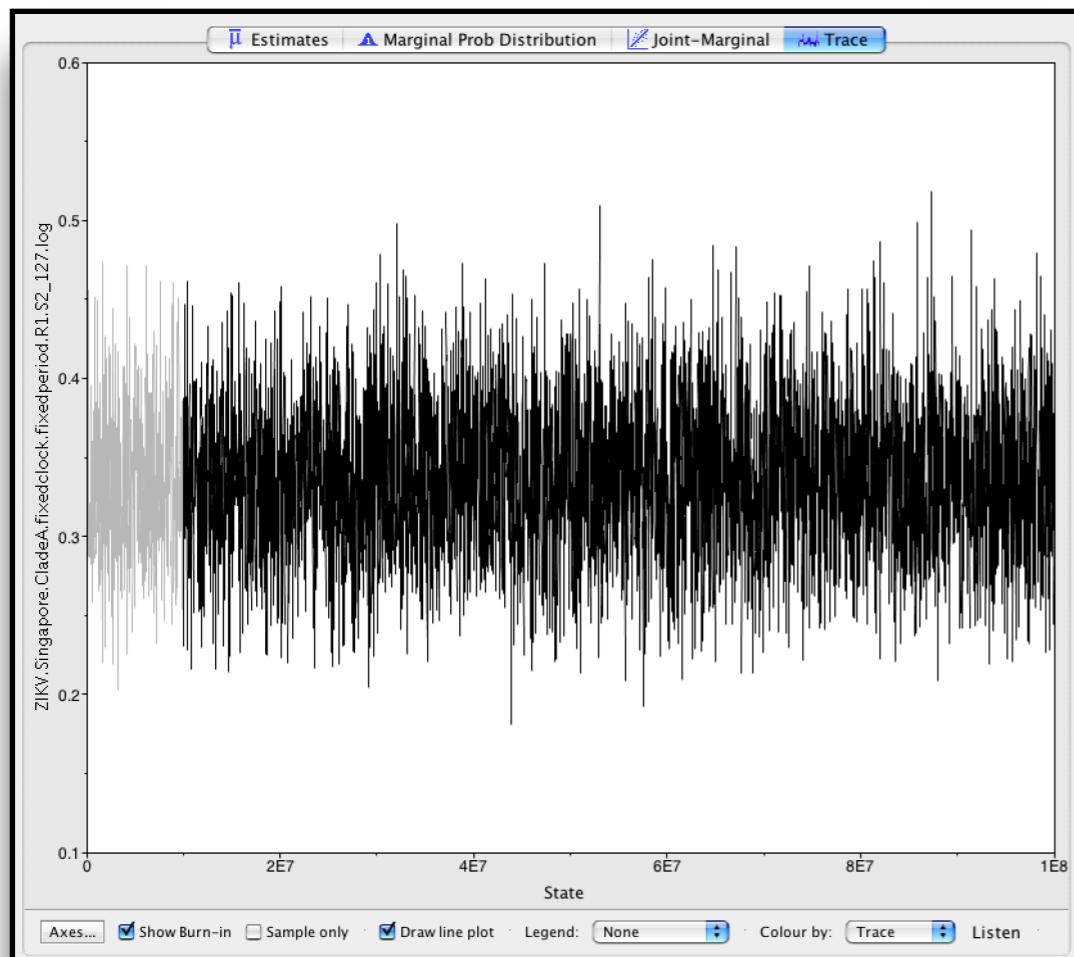


Tracer

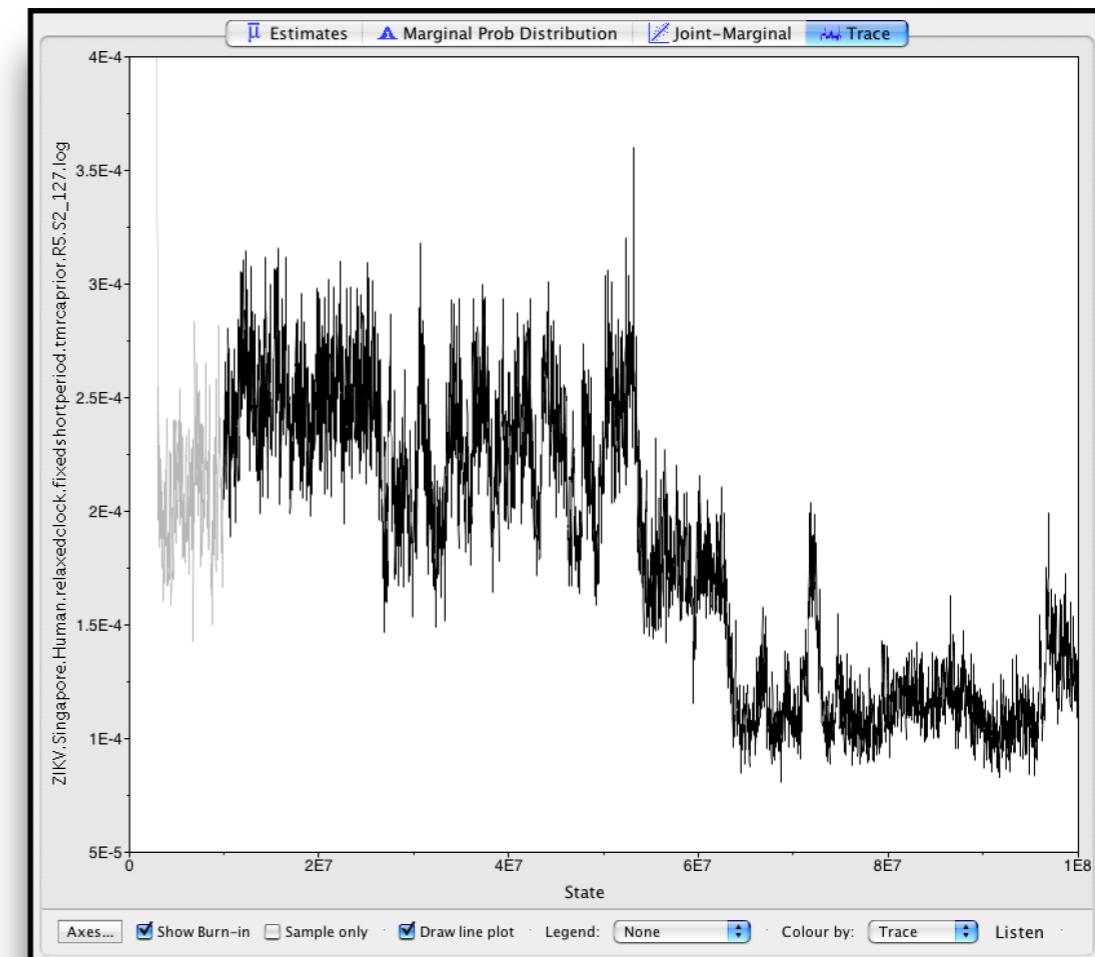
(<http://beast.community>)



Look at the chains first!



Mixing well! 😊



Not mixing! 😢



TreeAnnotator

(Included with BEAST2)

- Analyse trees file from BEAST2 runs
- Produces MCC tree with node annotations (posterior probability)
- Note that the MCC tree is just a summary and may never actually appear in the trees file!

Input:

- trees file
(many trees)

Output:

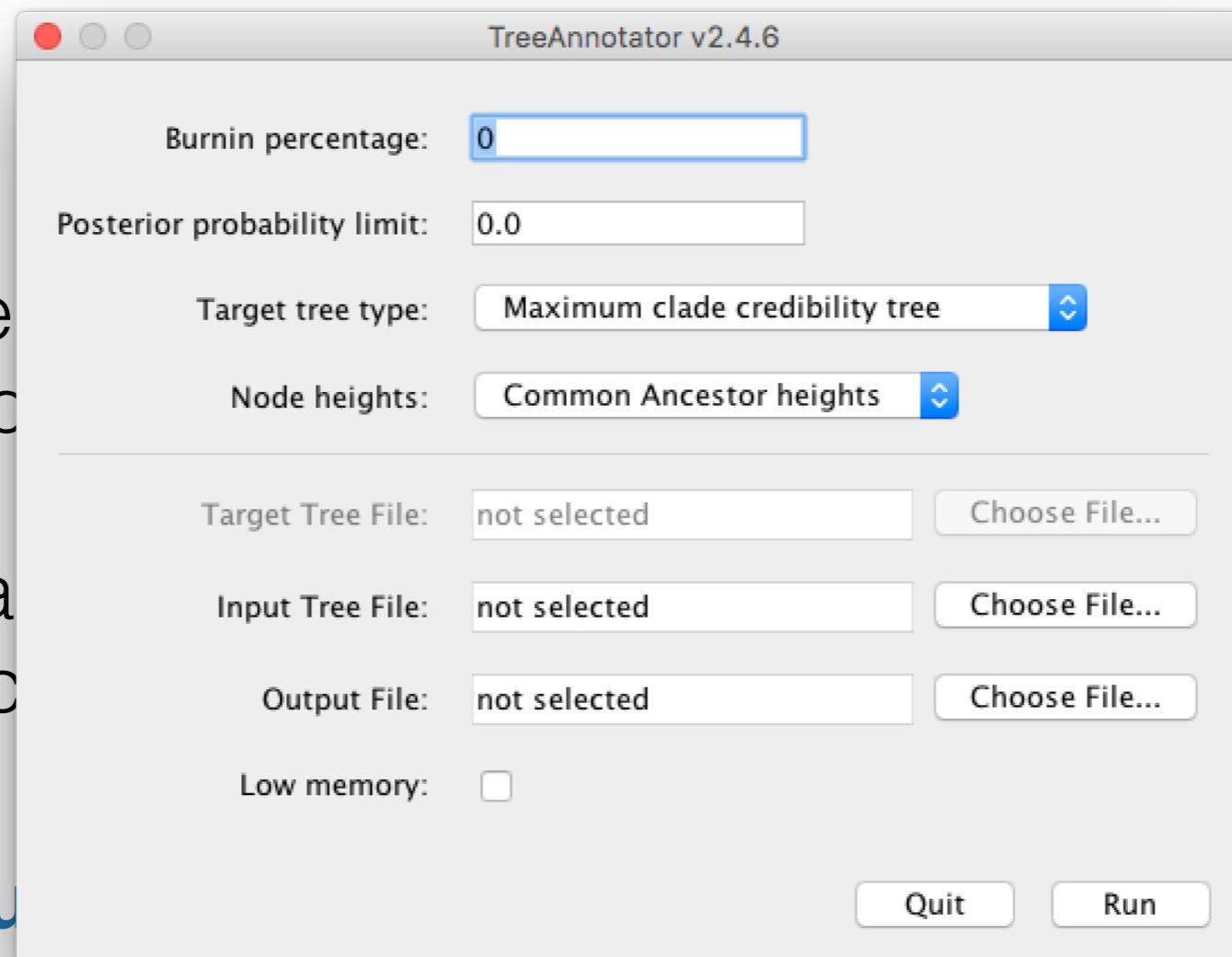
- MCC tree
(one tree)



TreeAnnotator

(Included with BEAST2)

- Analyse
- Produce
(posterior)
- Note that
never ac



Input

- trees file
(many trees)
- MCC tree
(one tree)

FigTree

(<http://tree.bio.ed.ac.uk/software/figtree/>)



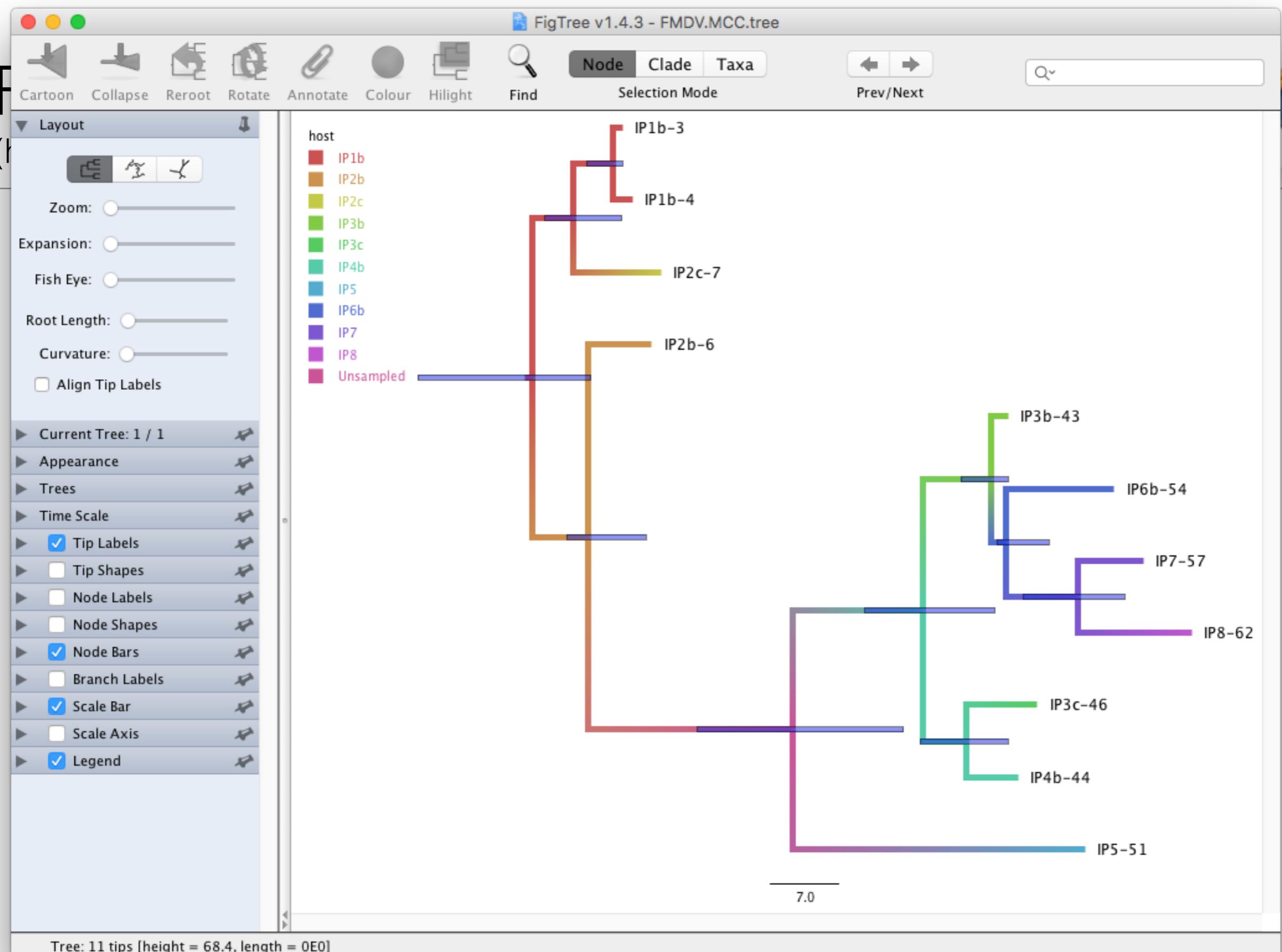
- Visualise trees from BEAST2 runs
- Annotate branches and nodes with probabilities and labels

Input:

- trees file

Output:

- Insight
- Figures



IcyTree

(<https://icytree.org>)



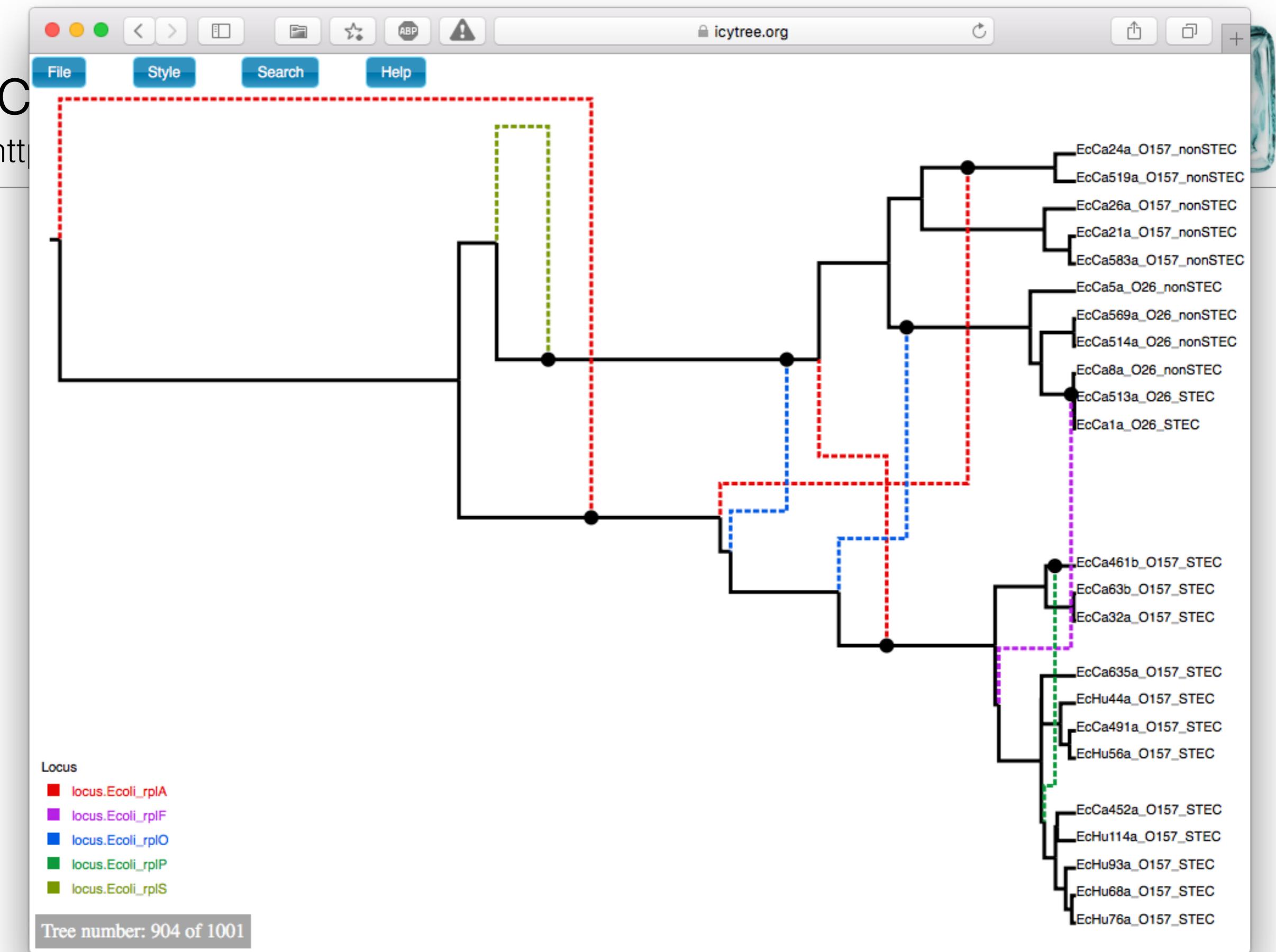
- Visualise trees from BEAST2 runs
- Annotate branches and nodes with probabilities and labels
- Better suited for structured models and ancestral recombination graphs (ARGs)
- Faster than FigTree for analysing many trees

Input:

- trees file

Output:

- Insight
- Figures

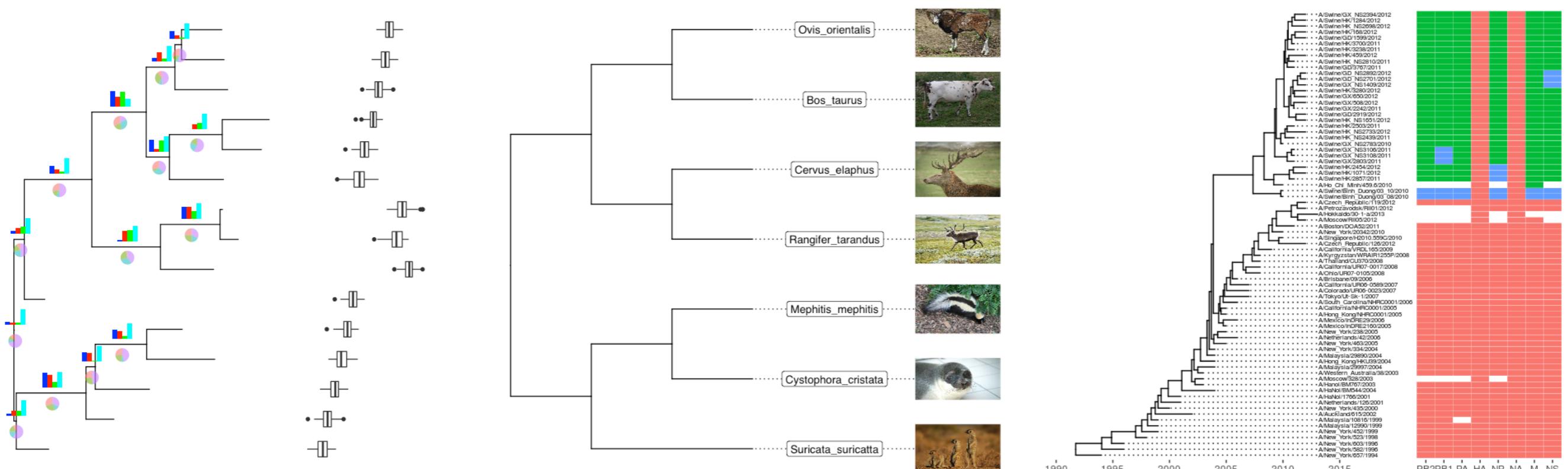


ggtree

(<https://guangchuangyu.github.io/software/ggtree/>)



- R-package to visualise trees using ggplot grammar
- Works with BEAST2 tree files (and many other packages)
- Can be easily annotate trees with other analyses in R



Tutorial

Introduction to BEAST2

This is a simple introductory tutorial to help you get started with using BEAST2 and its accomplices.
by Jūlija Pečerska , Veronika Bošková and Louis du Plessis

Tutorial

- Introduction-to-BEAST2.pdf
- Github repository
- License
- Statistics

Data

- primate-mtDNA.nex

XML

- Primates_long.xml
- Primates.xml

Output

- primate-mtDNA_long.trees
- primate-mtDNA.log
- Primates.MCC.tree
- primate-mtDNA_long.log
- primate-mtDNA.trees

Background

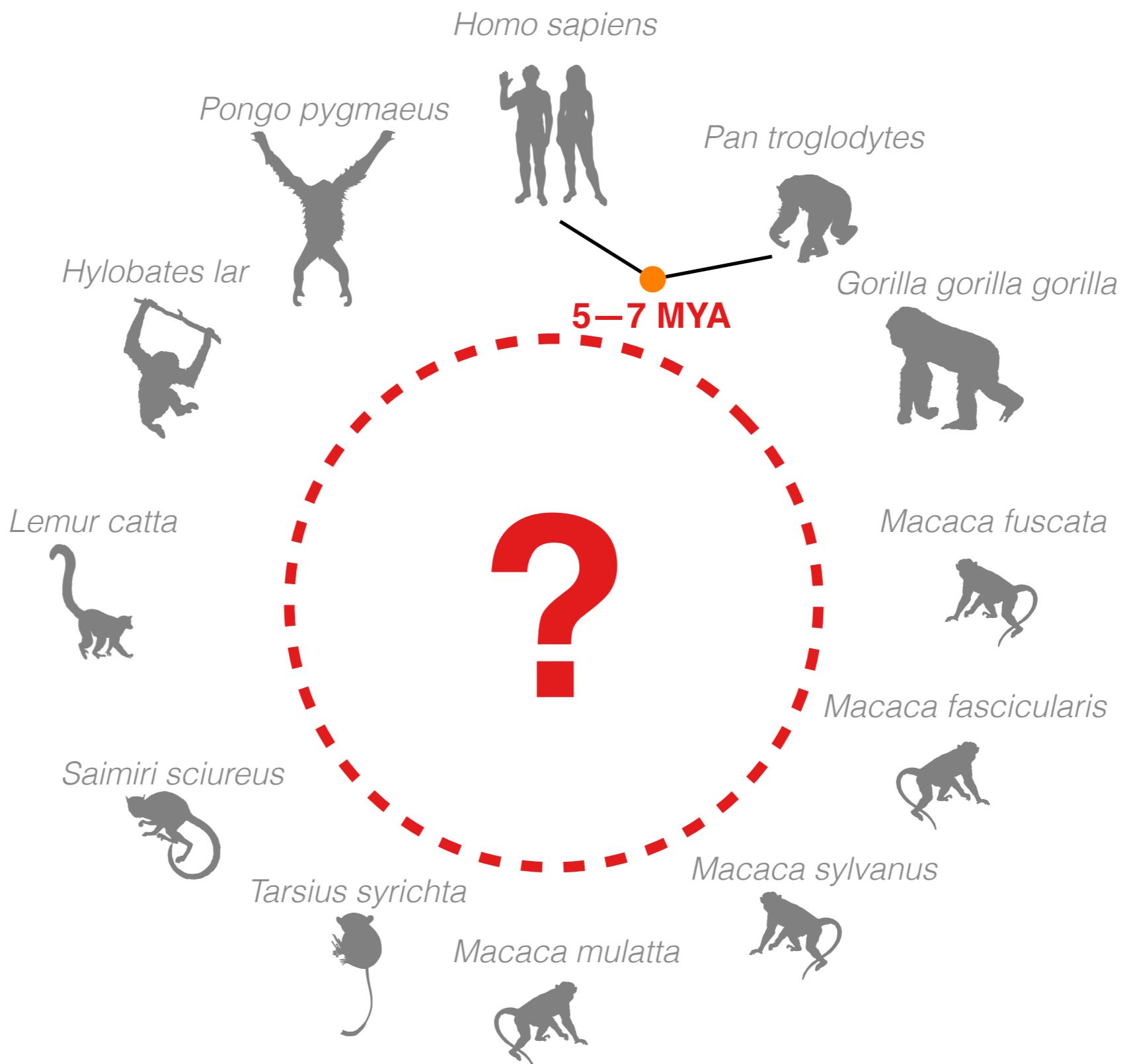
Before diving into performing complex analyses with the BEAST2 one needs to understand the basic workflow and concepts. While BEAST2 tries to be as user-friendly as possible, the amount of possibilities can be overwhelming.

In this simple tutorial you will get acquainted with the basic workflow of BEAST2 and the software tools most commonly used to interpret the results of analyses. Bear in mind that this tutorial is designed only to help you get started using BEAST2. This tutorial does not discuss all the choices and concepts in detail, as they are discussed in further tutorials. Interspersed throughout the tutorial are topics for discussion. These discussion topics are optional, however if you work through them you will have a much better understanding of the concepts discussed in this tutorial. Feel free to skip the discussion topics and come back to them later, while running the analysis file, or after finishing the whole tutorial.

Programs used in this Exercise

BEAST2 - Bayesian Evolutionary Analysis Sampling Trees 2

BEAST2 (<http://www.beast2.org>) is a free software package for Bayesian evolutionary analysis of molecular



non-coding

ACAC	...
TCAC	...
ACAG	...

1st position

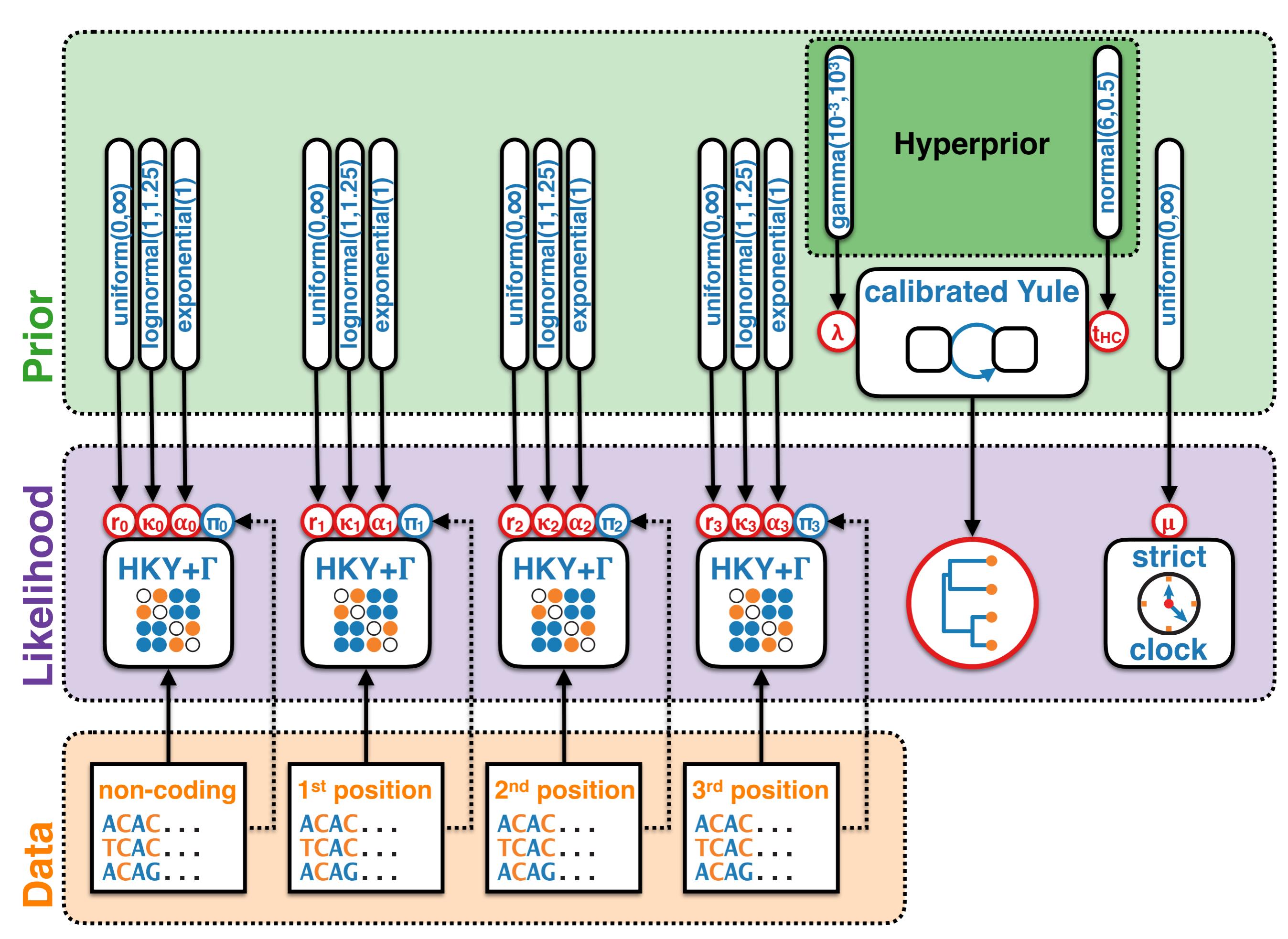
ACAC	...
TCAC	...
ACAG	...

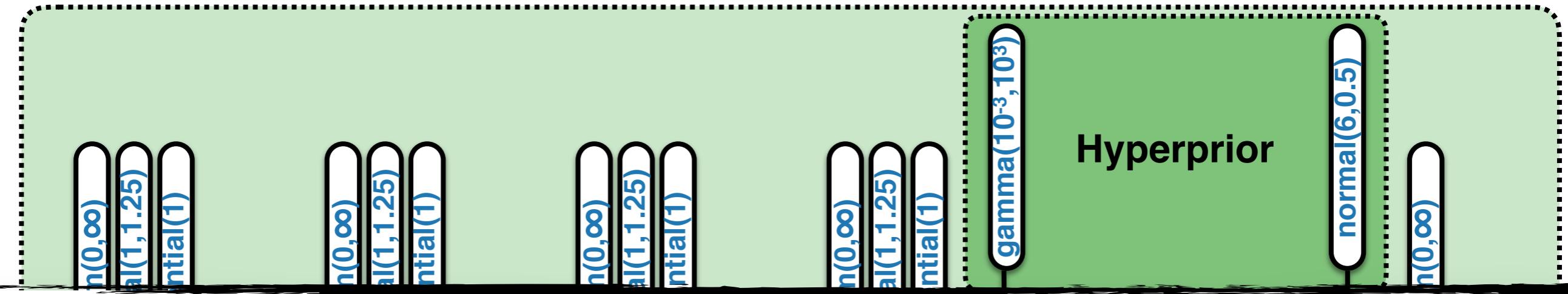
2nd position

ACAC	...
TCAC	...
ACAG	...

3rd position

ACAC	...
TCAC	...
ACAG	...





Can you write down the posterior?

$$P(\text{Data} | \text{Parameters}) = \frac{P(\text{Parameters} | \text{Data})}{P(\text{Parameters})}$$



Data

non-coding

ACAC
TCAC
ACAG

1st position

ACAC
TCAC
ACAG

2nd position

ACAC
TCAC
ACAG

3rd position

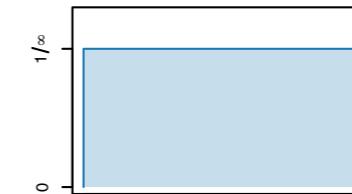
ACAC
TCAC
ACAG

Parameters

Priors



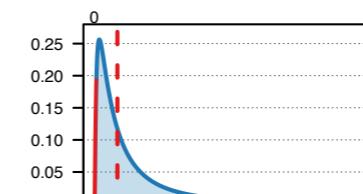
Relative mutation rate



uniform(0,∞)



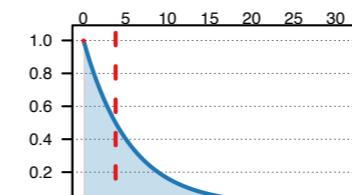
Transition-transversion ratio



lognormal(1,1.25)



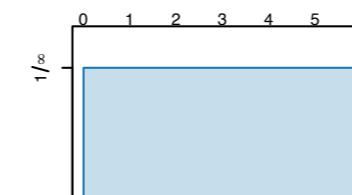
Shape parameter for discrete Γ model



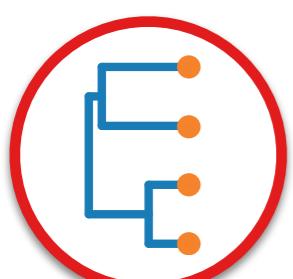
exponential(1)



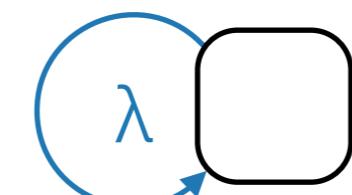
Clock rate



uniform(0,∞)



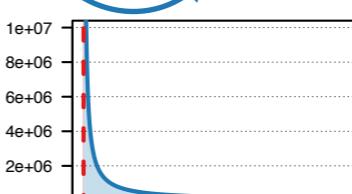
Tree



calibrated Yule



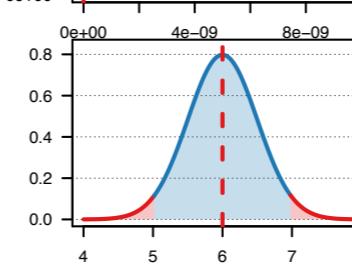
Speciation (birth) rate



gamma(10^-3,10^3)

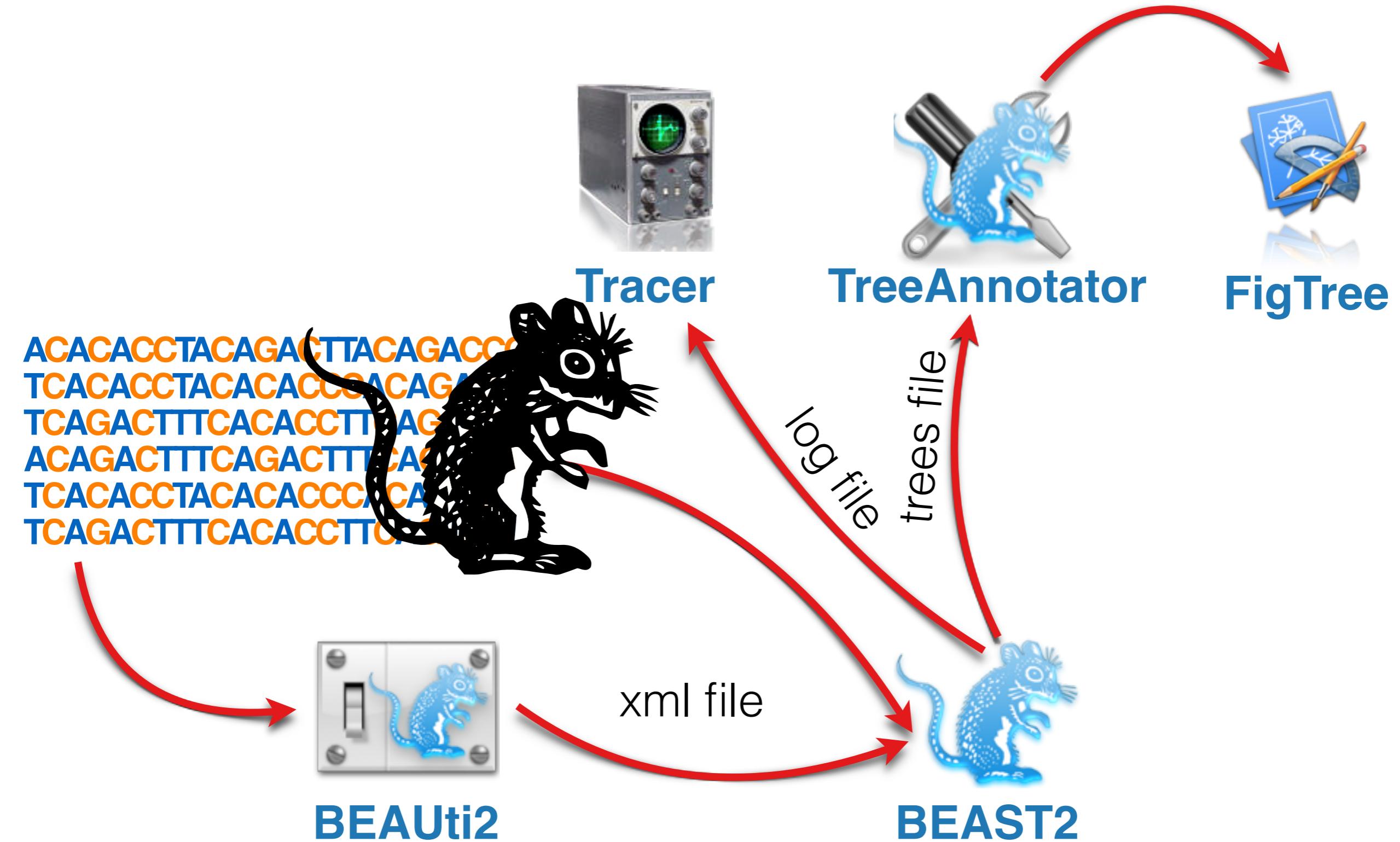


Calibration time
(Human-Chimpanzee)



normal(6,0.5)

BEAST2 workflow



Tools of the trade

BEAST2

Software implementing MCMC for model parameter and tree inference

BEAUTi2

Part of BEAST2 package for setting up the input file (.xml)

Tracer

Analysis of BEAST1 and BEAST2 output files (.log)

TreeAnnotator

Analysis of BEAST2 output files (.trees)

FigTree, IcyTree, ggtree

Visualisation of trees (.trees)

BEAST best practice

(This is just a guideline and each analysis is unique)

Before you begin

- 1) Know your data
- 2) Plan your analysis carefully

Before you run the analysis

- 3) Ask someone else to look at your XML file
- 4) Sample from the prior (run without data)

Actually running the analysis

- 5) Run analysis with multiple chains

After the analysis

- 6) Combine chains
- 7) Assess convergence and mixing
- 8) Ask someone else to look at your log files