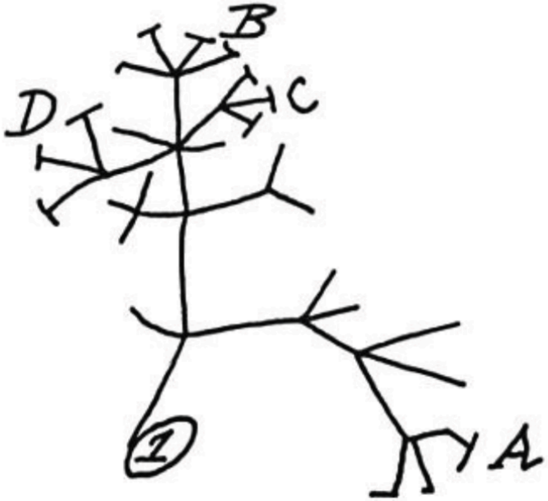
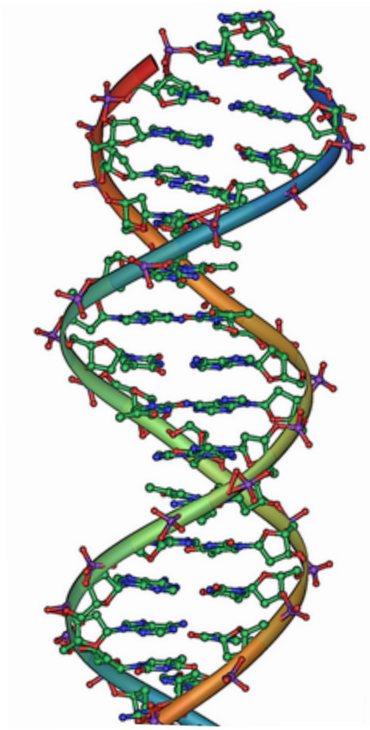
# Statistical Phylogenetic Analysis

Jeremy M. Brown
Dept. of Biological Sciences
Louisiana State University



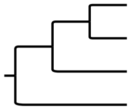


$$p(y^{rep}|y) = \int p(y^{rep}|\theta)p(\theta|y)d\theta$$

@ jembrown
#bodega13

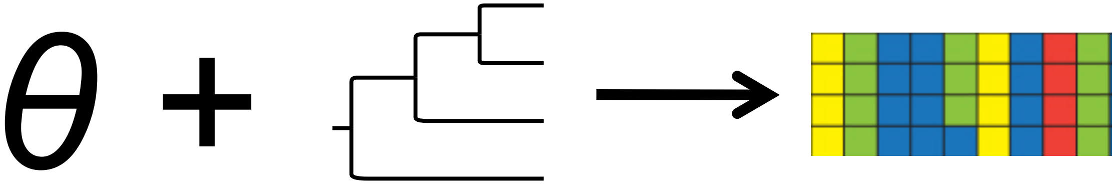
# The Players

$\theta$ = Sequence Evolution Model Parameters

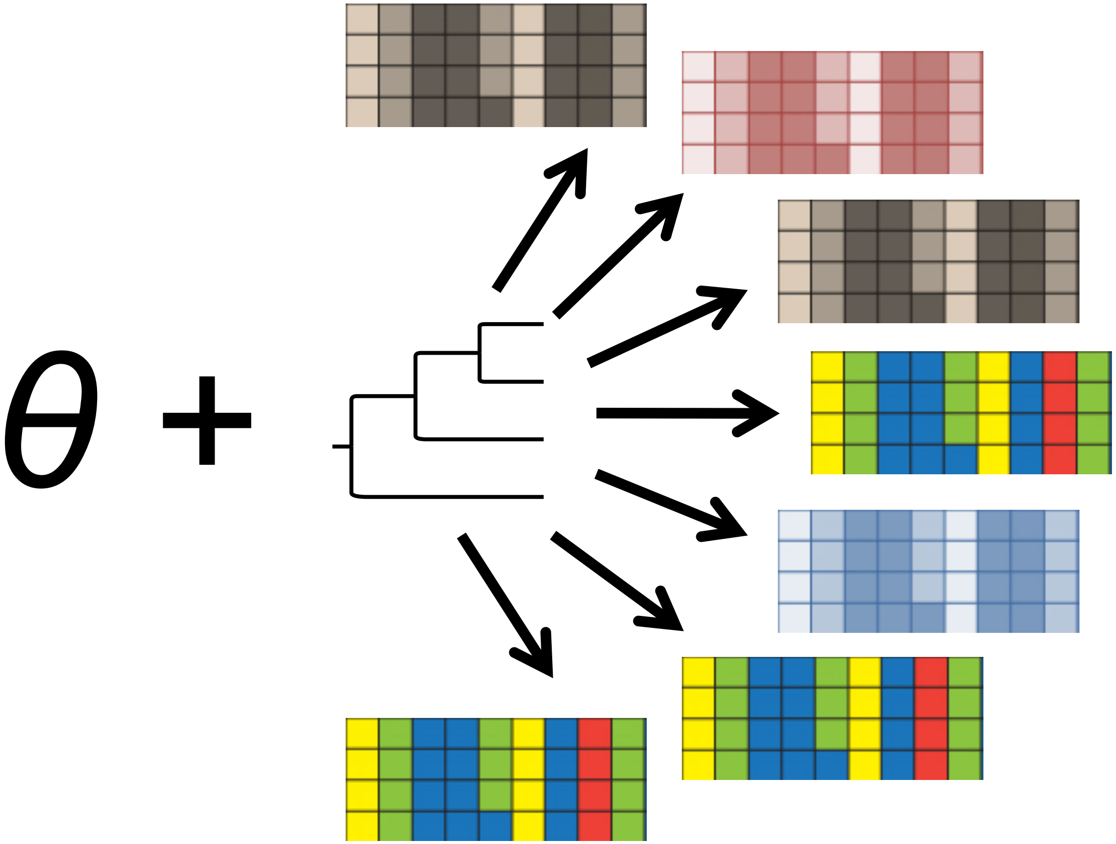
 = Tree Topology and Branch Lengths

= Sequence Alignment

# Simulation View of Models



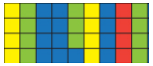
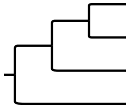
# Simulation View of Models



$\theta$ +

How frequently is some data observed when datasets are repeatedly generated with a particular tree and set of model parameters?

# The Likelihood Function

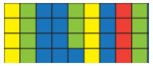
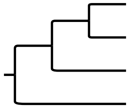
P(|, $\theta$)

Read as "the probability of the sequence data given a tree and a set of model parameter values".

The quantity by which the data provide information.

Compares how well different trees and models predict the observed data.

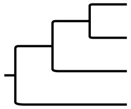
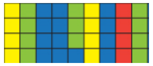
# The Likelihood Function

P(  |  , $\theta$)

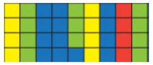
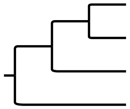
Read as "the probability of the sequence data given a tree and a set of model parameter values".

The quantity by which the data provide information.

Compares how well different trees and models predict the observed data or as a "measure of surprise".

NOT the same as P(  |  )

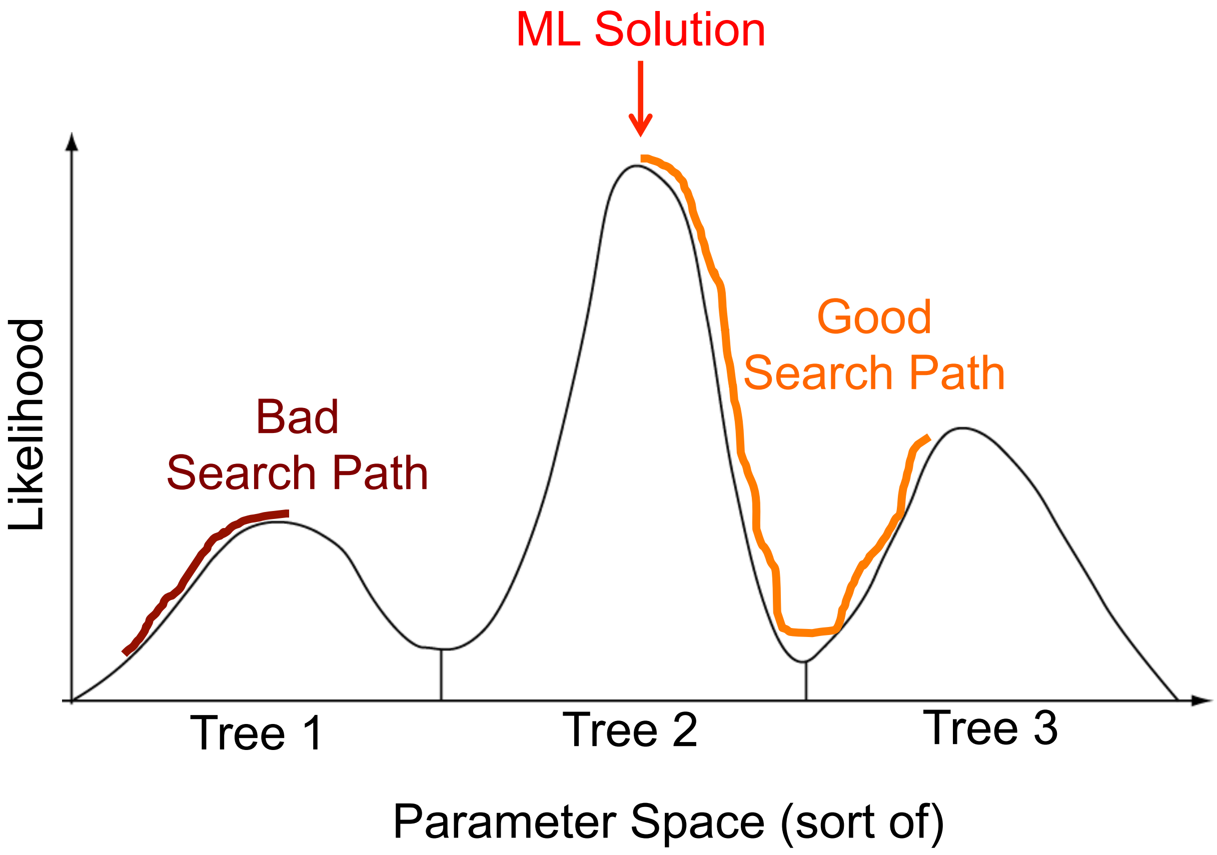
# Maximum Likelihood

P(  |  , $\theta$ )

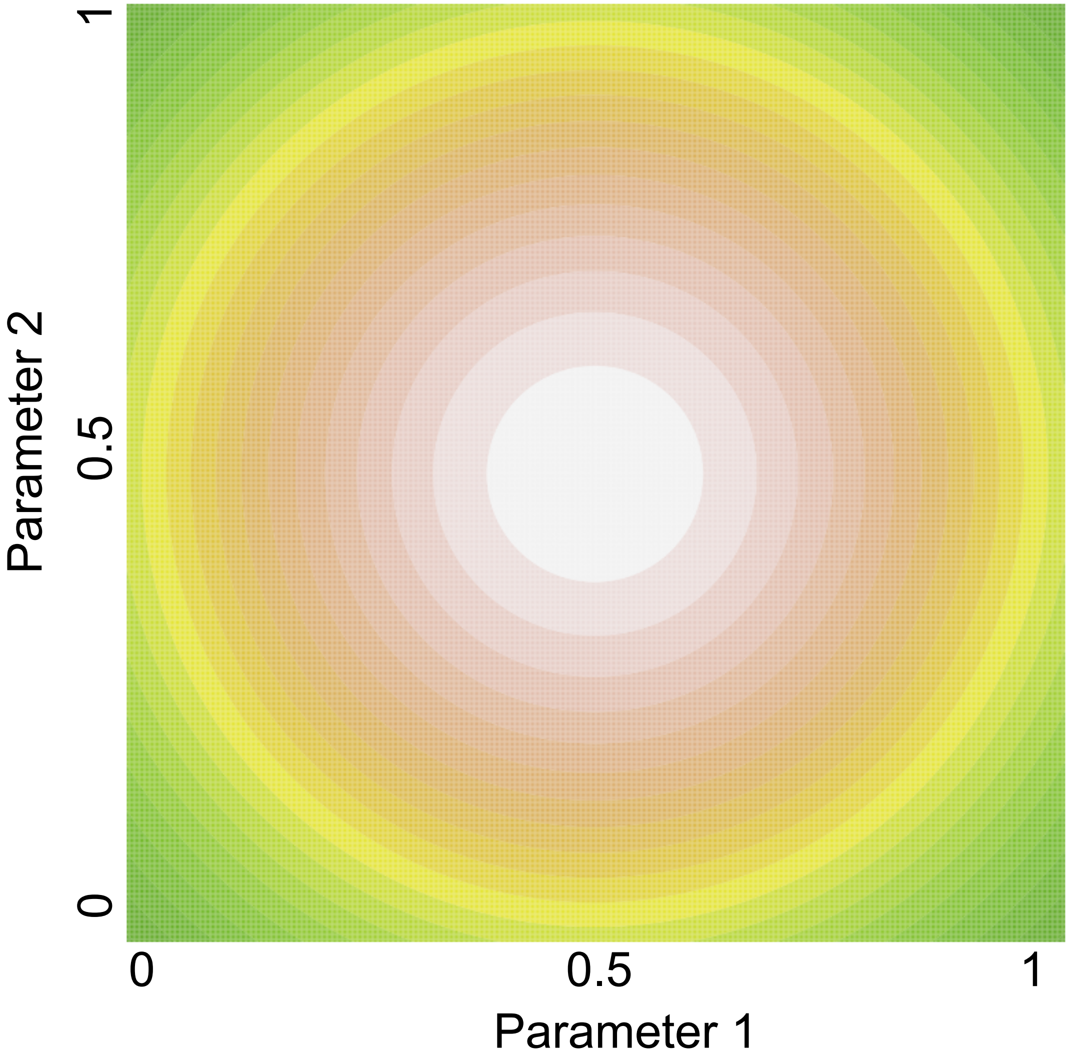
What tree and parameter values give the highest likelihood?

ML scores are just relative, so alone they doesn't tell us how confident we are in this solution, just that this is the preferred solution.

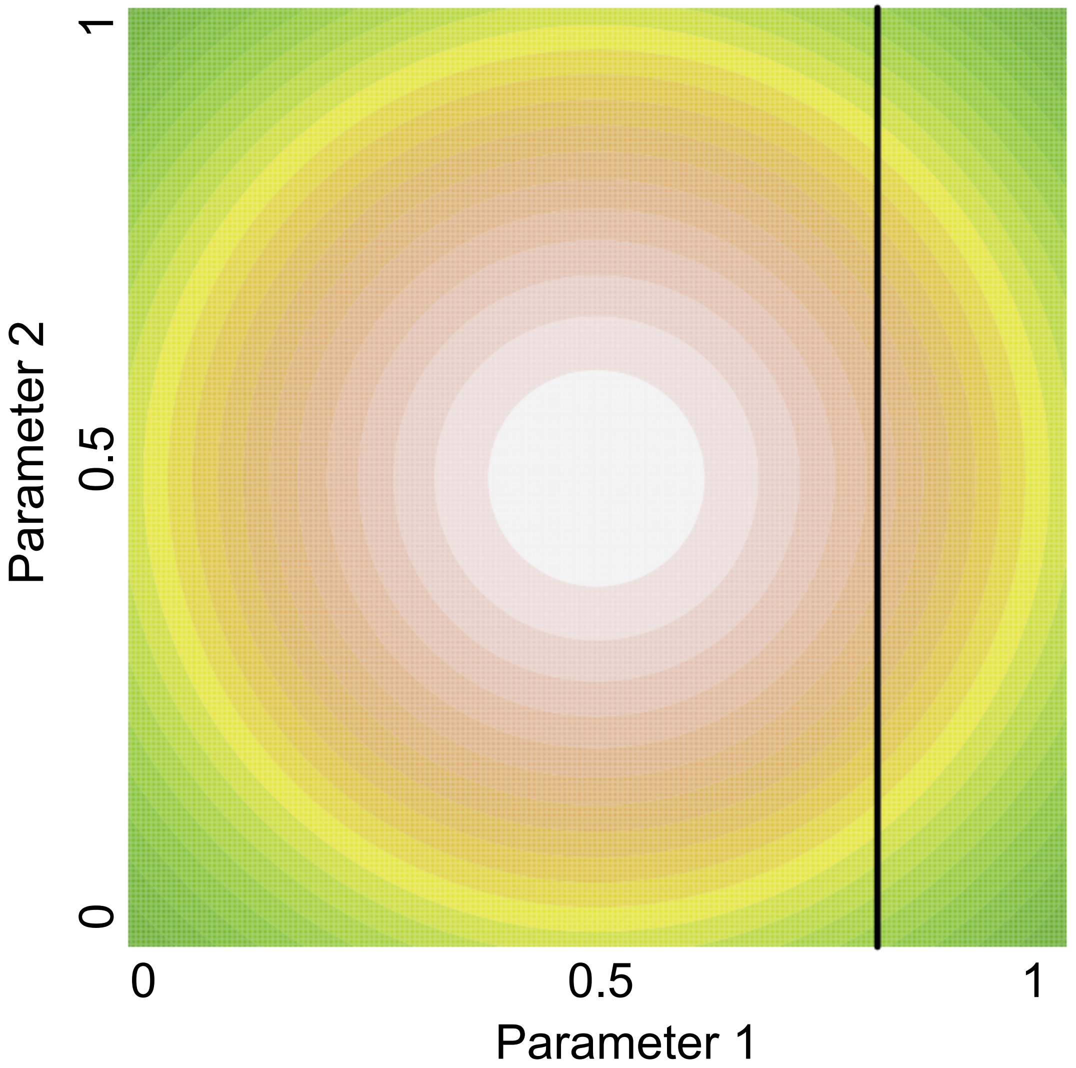
# Maximum Likelihood



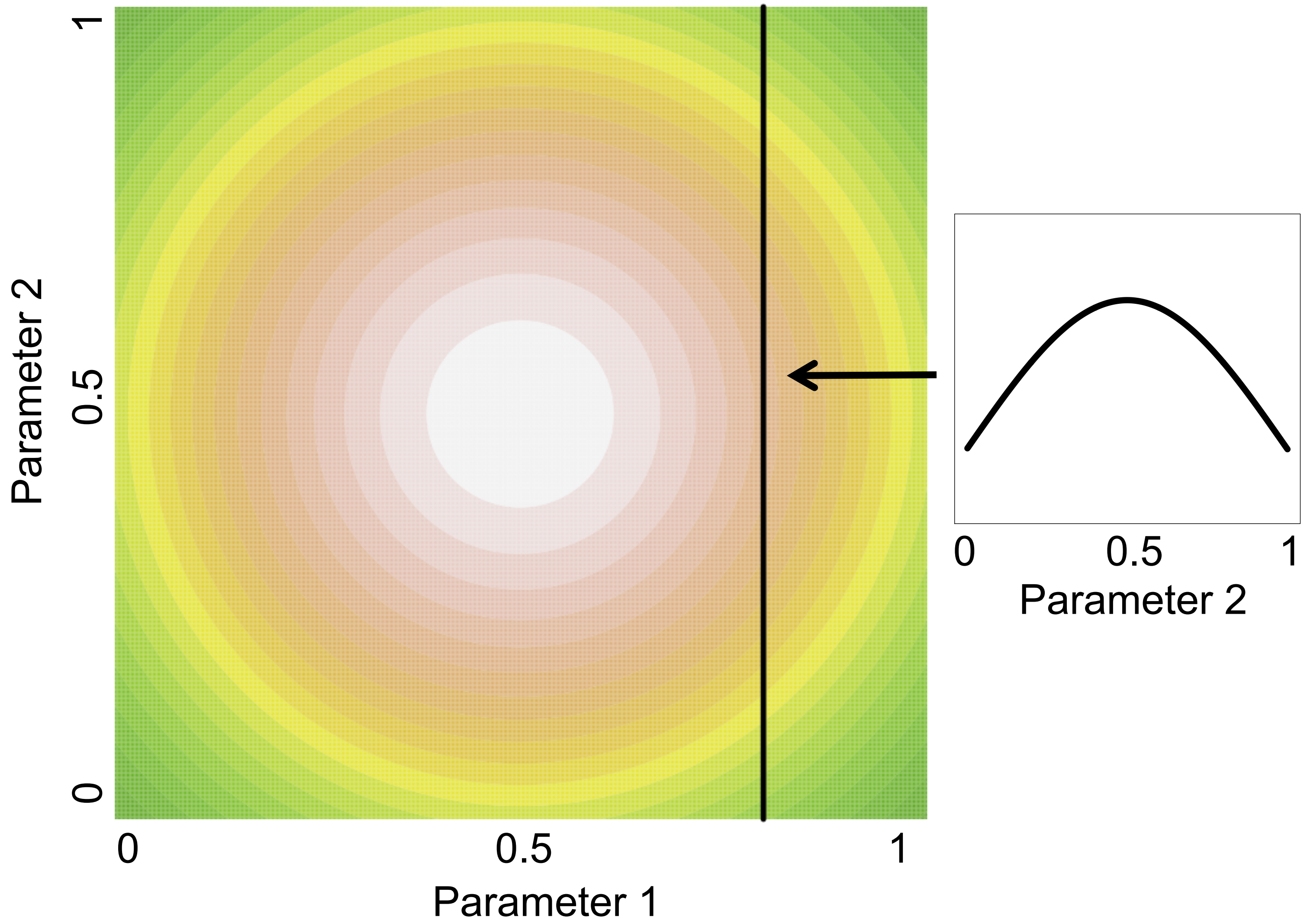
# More Parameters = Better ML Score



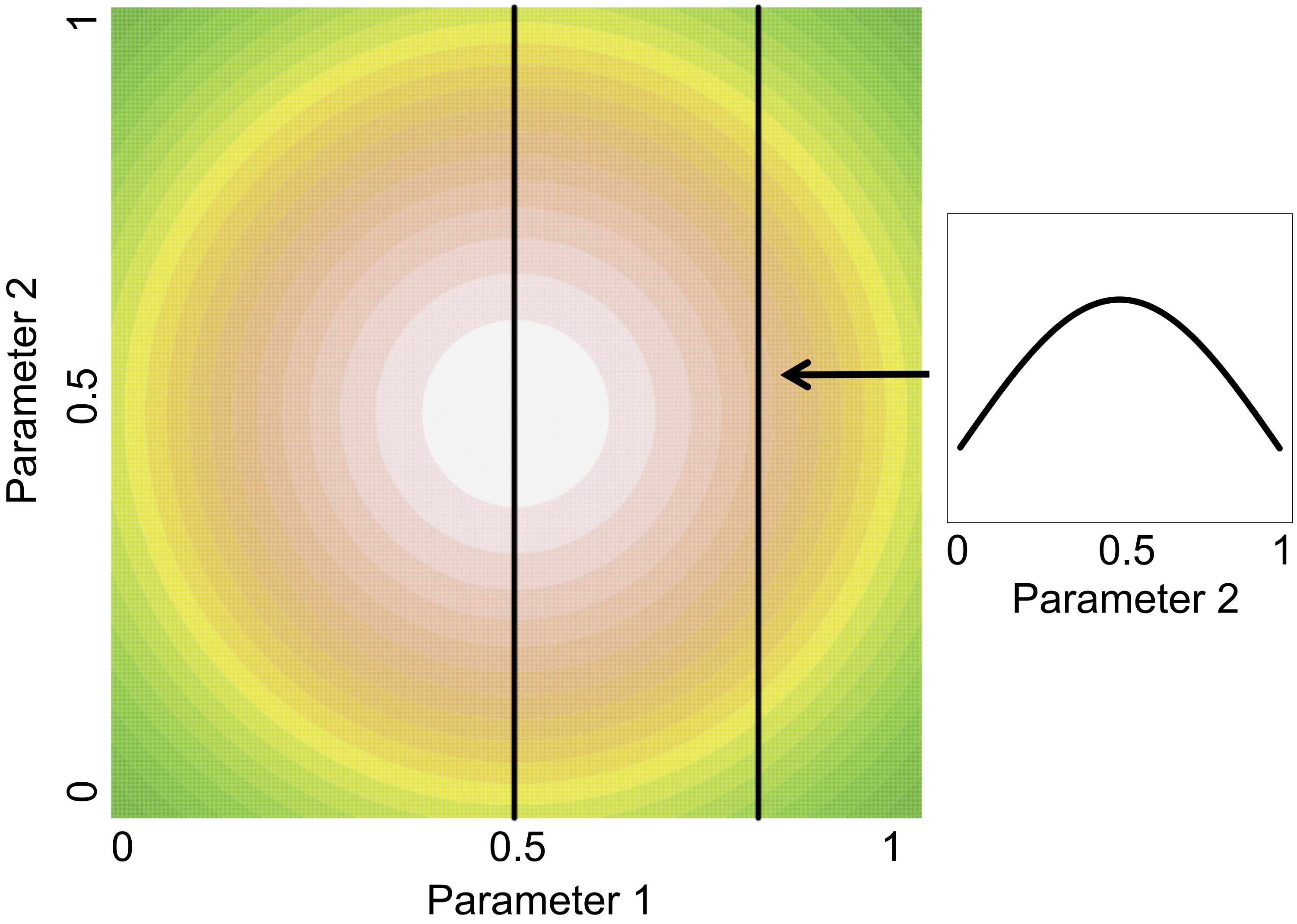
# More Parameters = Better ML Score



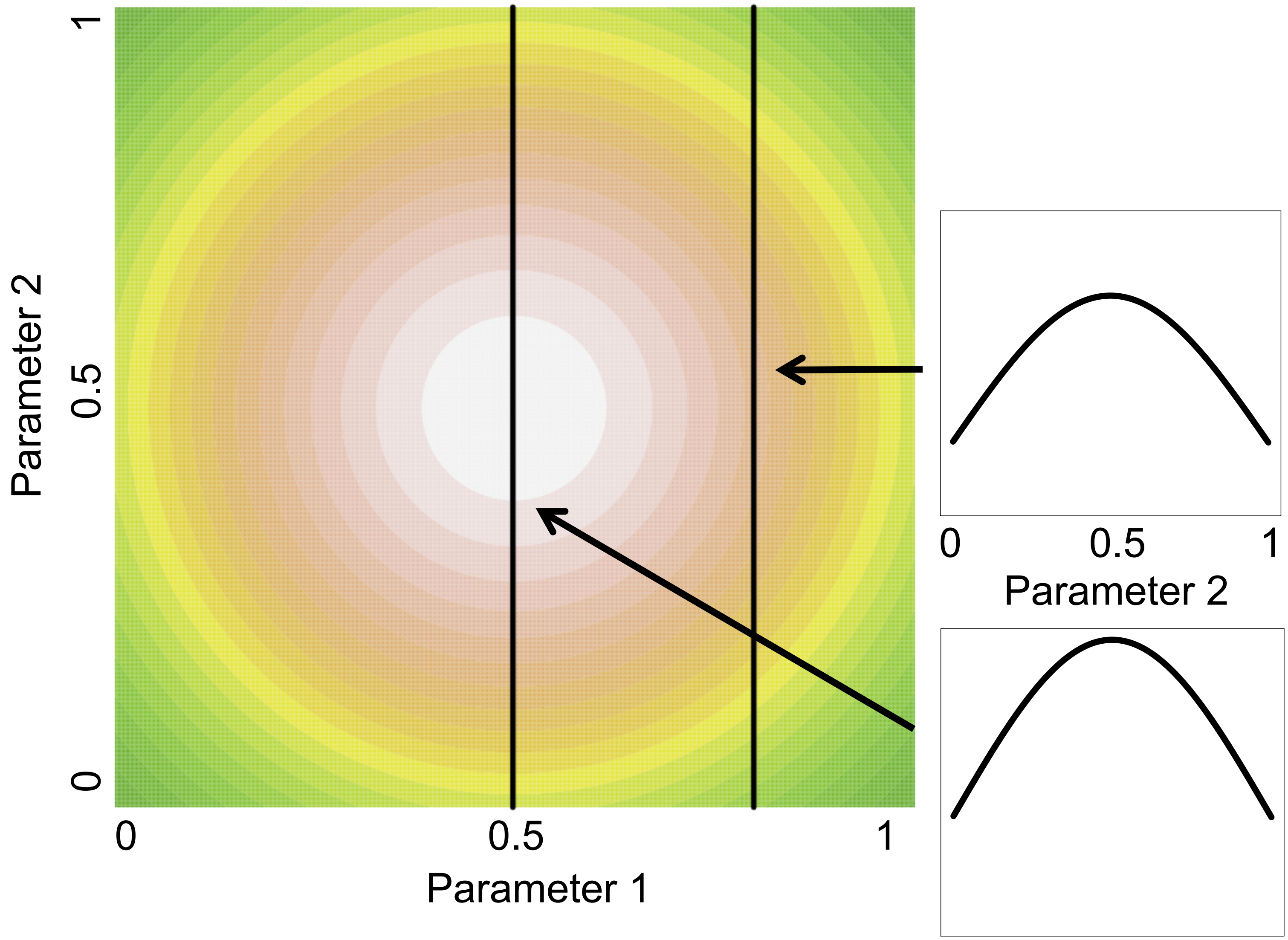
# More Parameters = Better ML Score



# More Parameters = Better ML Score

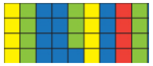
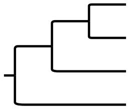


# More Parameters = Better ML Score



# Maximum Likelihood

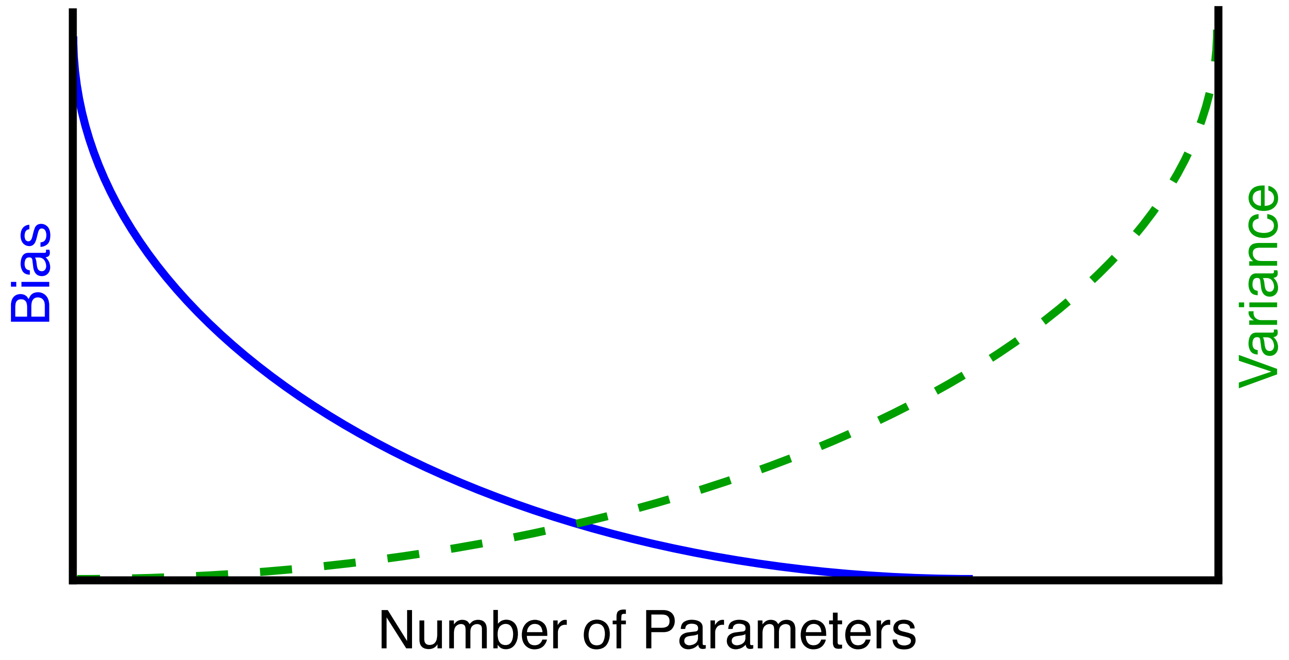
**Usually implicit**

P(|,$\theta$,$M$)

What tree and parameter values give the highest likelihood?

ML scores are just relative, so alone they doesn't tell us how confident we are in this solution, just that this is the preferred solution.

# ML-based Model Selection



# ML-based Model Selection Criteria

- AIC – Penalty from information theory

- BIC – Penalty designed to mimic a posterior

- DT – Penalty based on performance
  (e.g., branch-length estimation)

- LRT – Compares to expected increase in
  ML if simple model true

# ML-based Model Selection Criteria

Let’s try this with jModelTest!

# ML Inference

Usually done in Garli or RAxML for large datasets.

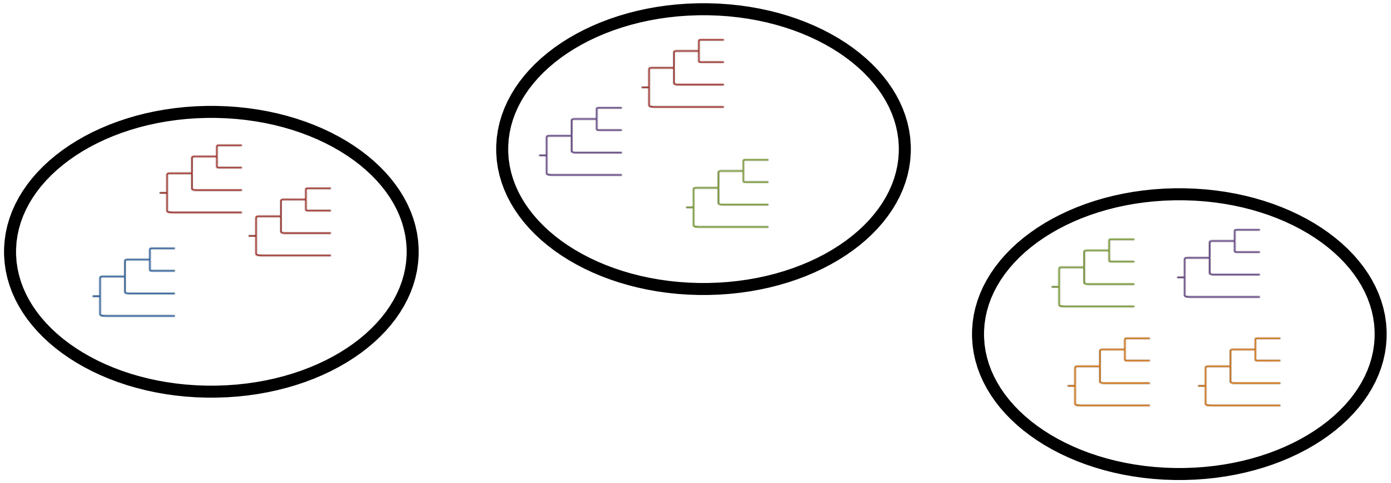
Garli uses a genetic algorithm to find the best trees.

# Genetic Algorithm

Uses evolutionary principles to solve complex problems:

- Selection
- Mutation
- Recombination
- Metapopulations

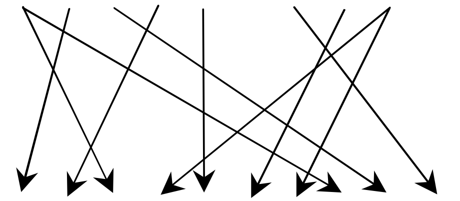
Fitness is equivalent to the quality of the solution (likelihood)



# Bootstrapping to Assess Confidence

Original alignment

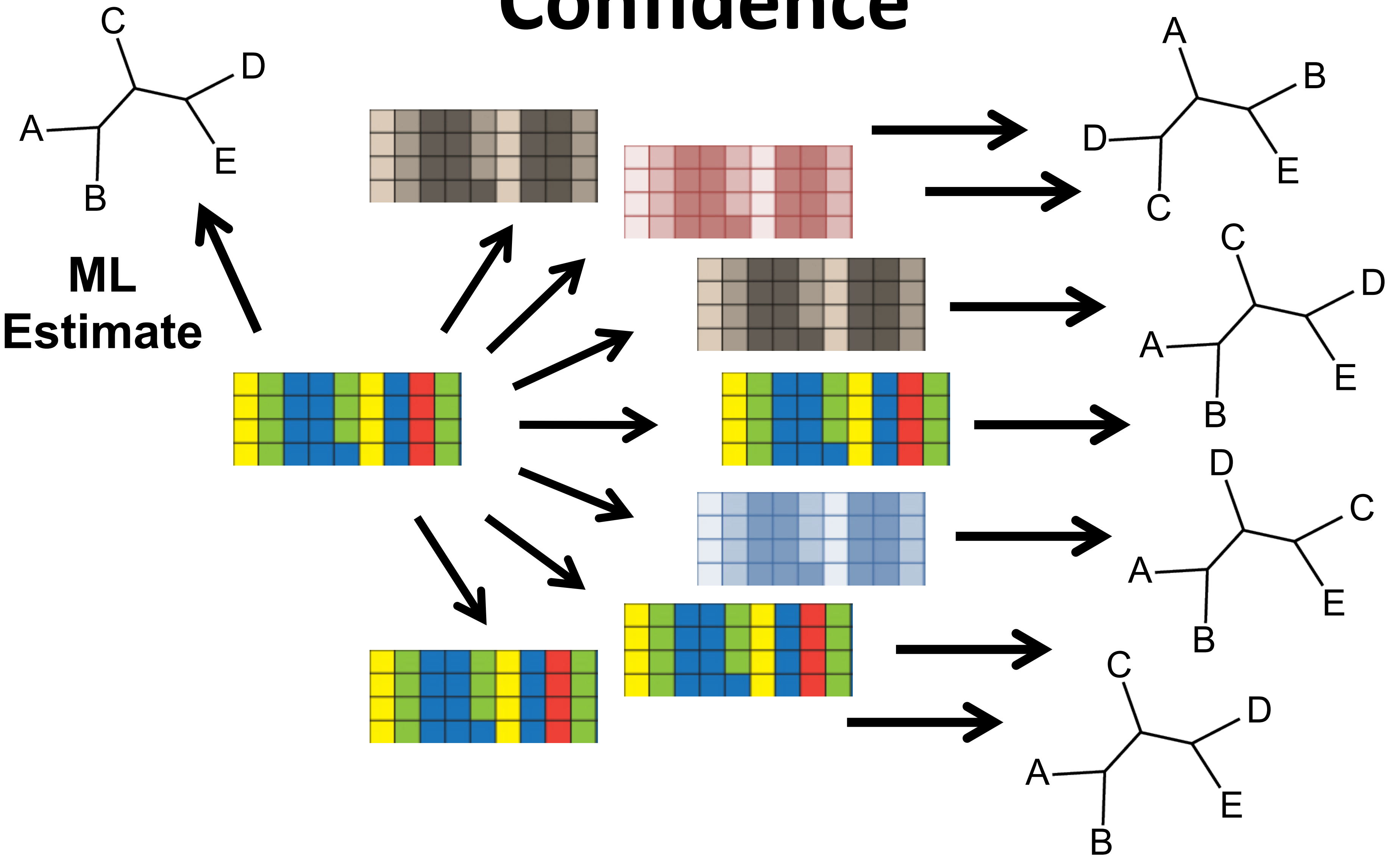
| Site | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| human | N | E | N | L | F | A | S | F | I | A |
| chimpanzee | N | E | N | L | F | A | S | F | A | A |
| bonobo | N | E | N | L | F | A | S | F | A | A |
| gorilla | N | E | N | L | F | A | S | F | I | A |
| orangutan | N | E | D | L | F | T | P | F | T | T |
| Sumatran | N | E | S | L | F | T | P | F | I | T |
| gibbon | N | E | N | L | F | T | S | F | A | T |



Bootstrap sample

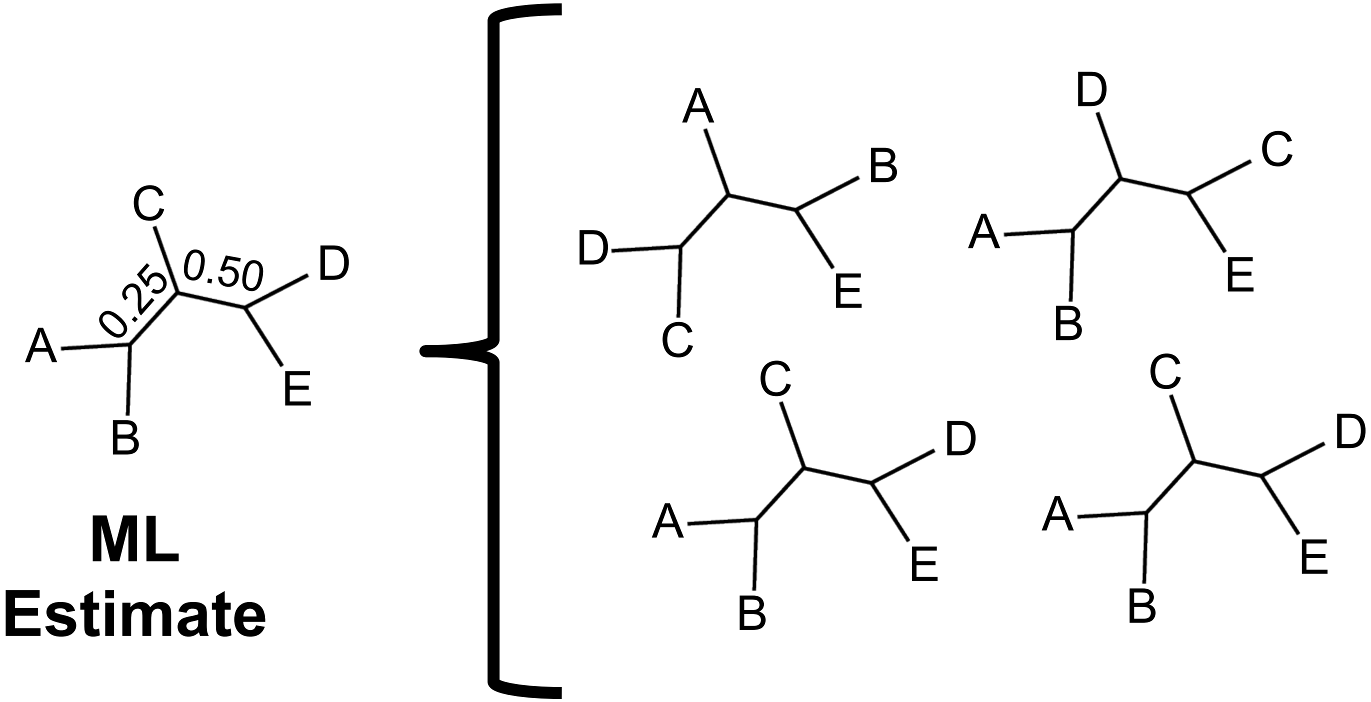
| Site | 2 | 4 | 1 | 9 | 5 | 8 | 9 | 1 | 3 | 7 |
|---|---|---|---|---|---|---|---|---|---|---|
| human | E | L | N | I | F | F | I | N | N | S |
| chimpanzee | E | L | N | A | F | F | A | N | N | S |
| bonobo | E | L | N | A | F | F | A | N | N | S |
| gorilla | E | L | N | I | F | F | I | N | N | S |
| orangutan | E | L | N | T | F | F | T | N | D | P |
| Sumatran | E | L | N | I | F | F | I | N | S | P |
| gibbon | E | L | N | A | F | F | A | N | N | S |

# Bootstrapping to Assess Confidence



ML Estimate

# Bootstrapping to Assess Confidence



**ML Estimate**

# Bootstrapping to Assess Confidence

Interpretations of the bootstrap:

- Repeatability

- 1 – False Positive Rate from a polytomy

- Probability branch is true

# Bootstrapping to Assess Confidence

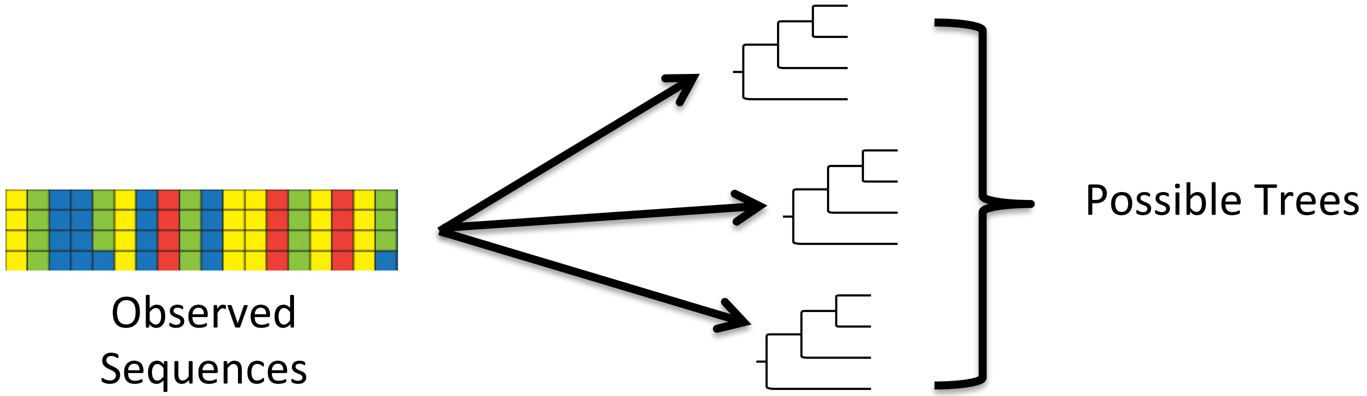
Interpretations of the bootstrap:

- Repeatability

- 1 – False Positive Rate from a polytomy

- ~~Probability branch is true~~

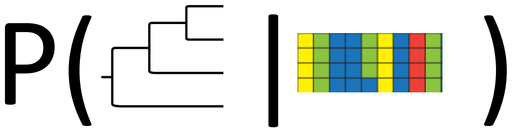
# Bayesian Inference



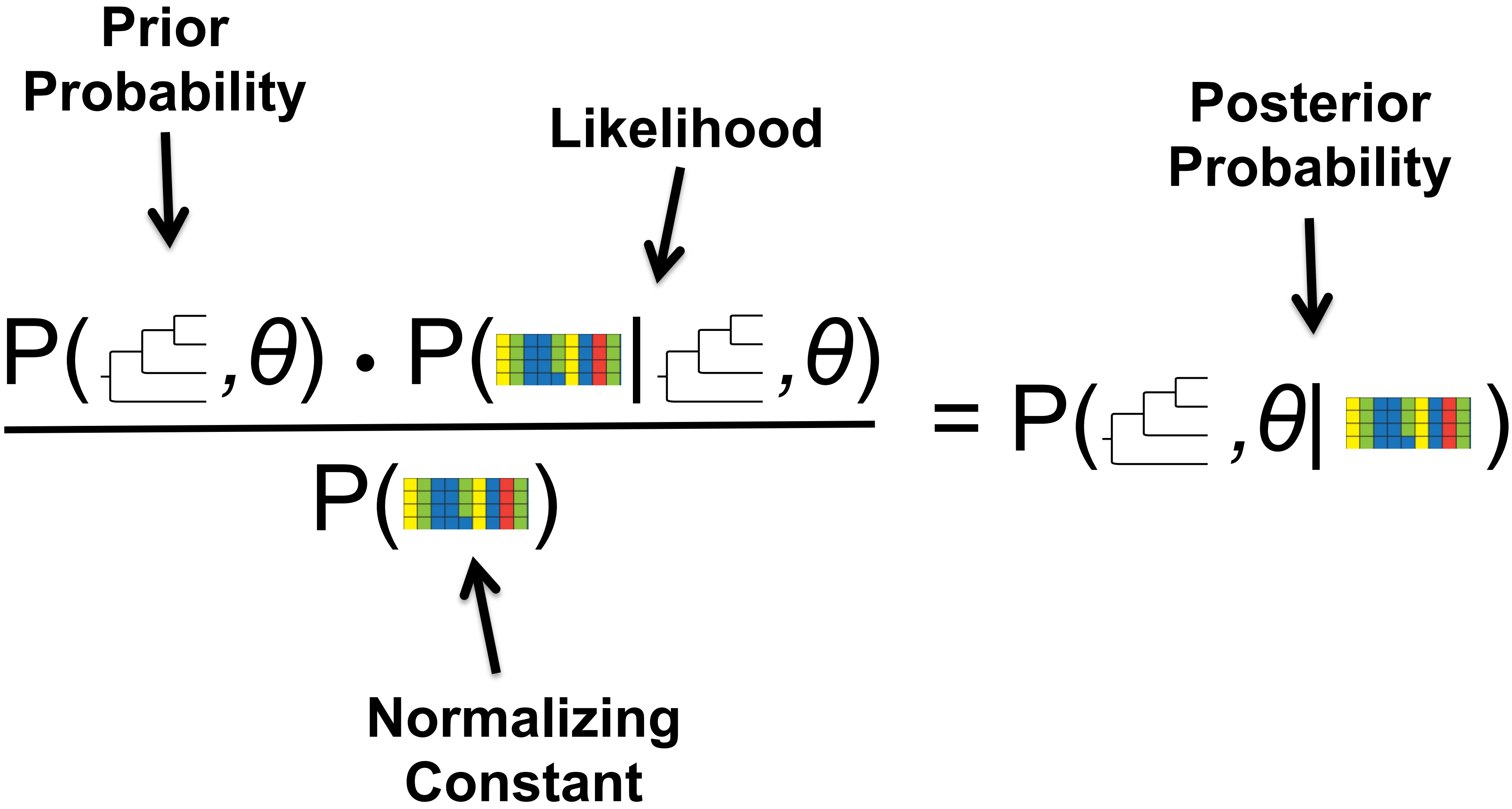
Observed Sequences

Possible Trees

Posterior Probability: Conditional on observed data (alignment), the probability that a particular tree (or part of a tree) is true.



# Bayes’ Theorem



Prior Probability

Likelihood

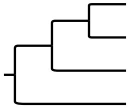
Posterior Probability

Normalizing Constant

# Marginalizing

**Joint Posterior Probability**

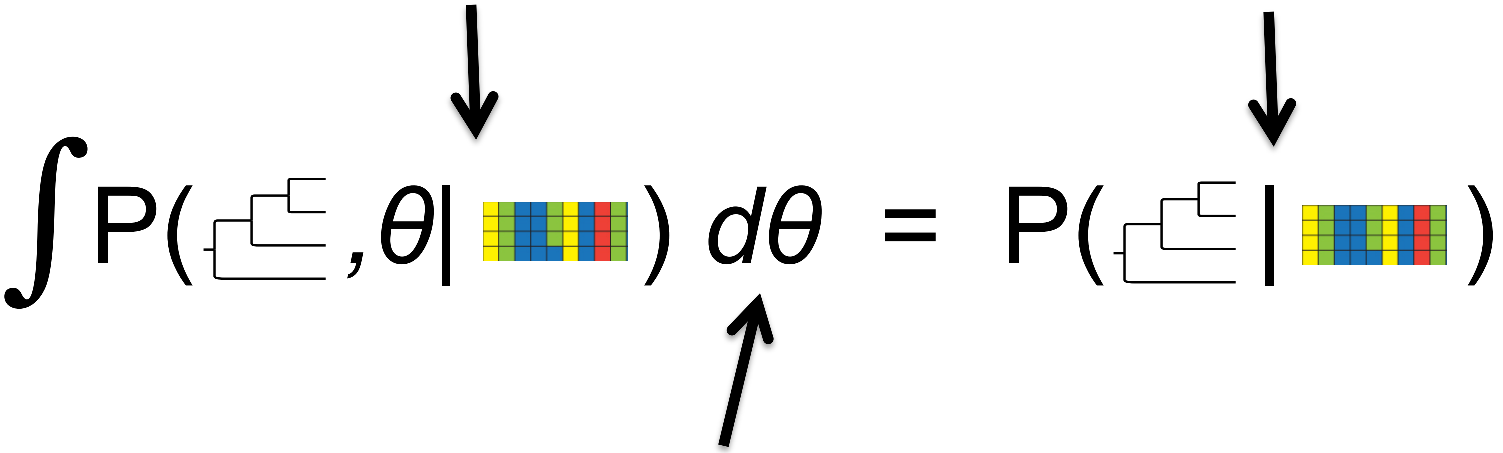
↓

P(  , $\theta$ | )

# Marginalizing

Joint
Posterior
Probability

Marginal
Posterior
Probability

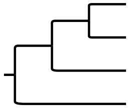
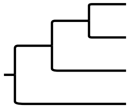
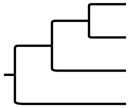


Integrating across all values of
model parameters

# Marginalizing

Trees

Model Parameter Vectors

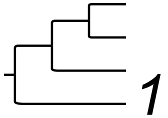
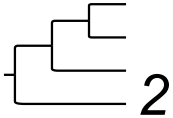
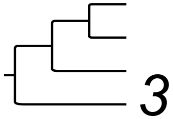
| |  1 |  2 |  3 | |
|---|---|---|---|---|
| $\theta_1$ | 0.10 | 0.07 | 0.12 | 0.29 |
| $\theta_2$ | 0.05 | 0.22 | 0.06 | 0.33 |
| $\theta_3$ | 0.05 | 0.19 | 0.14 | 0.38 |
| | 0.20 | 0.48 | 0.32 | |

Joint Probabilities

Marginal Probabilities

# Marginalizing Across Models

Trees

| Models |  1 |  2 |  3 | |
|---|---|---|---|---|
| $M_1$ | 0.10 | 0.07 | 0.12 | 0.29 |
| $M_2$ | 0.05 | 0.22 | 0.06 | 0.33 |
| $M_3$ | 0.05 | 0.19 | 0.14 | 0.38 |
| | 0.20 | 0.48 | 0.32 | |

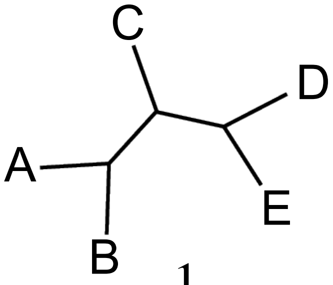
Joint Probabilities

Marginal Probabilities

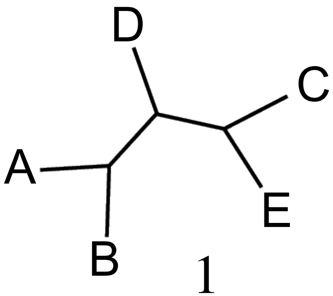
# What Priors to Use?

- The controversial part of Bayesian analysis
- Choice can vary by researcher
- Often chosen in an attempt to reflect prior ignorance
- Analysis can be run under several priors to assess sensitivity

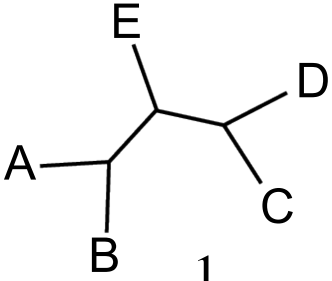
# Uniform Topology Prior



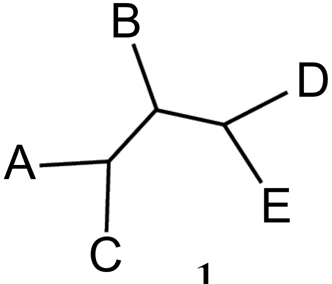
$\frac{1}{15}$



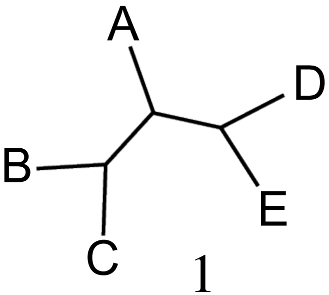
$\frac{1}{15}$



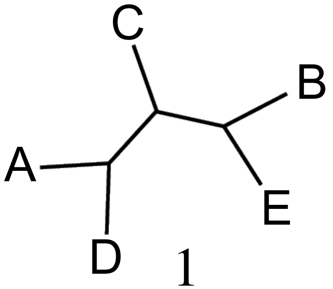
$\frac{1}{15}$



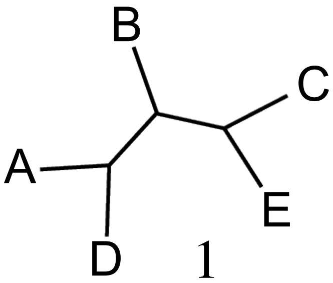
$\frac{1}{15}$



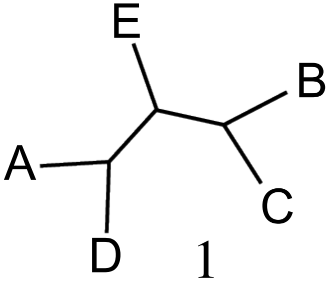
$\frac{1}{15}$



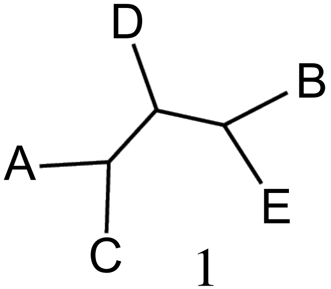
$\frac{1}{15}$



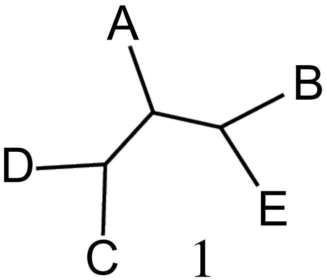
$\frac{1}{15}$



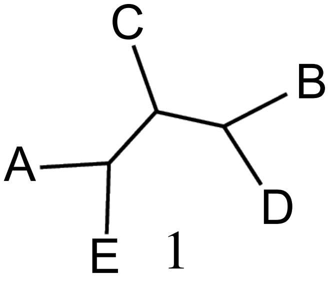
$\frac{1}{15}$



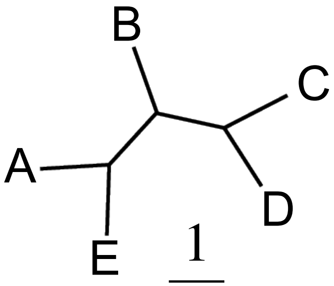
$\frac{1}{15}$



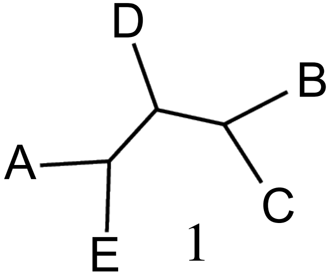
$\frac{1}{15}$



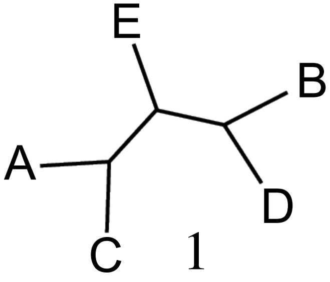
$\frac{1}{15}$



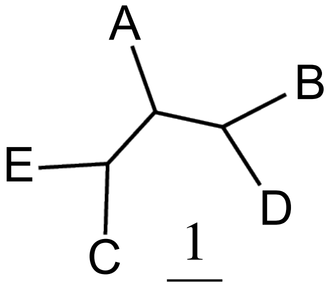
$\frac{1}{15}$



$\frac{1}{15}$

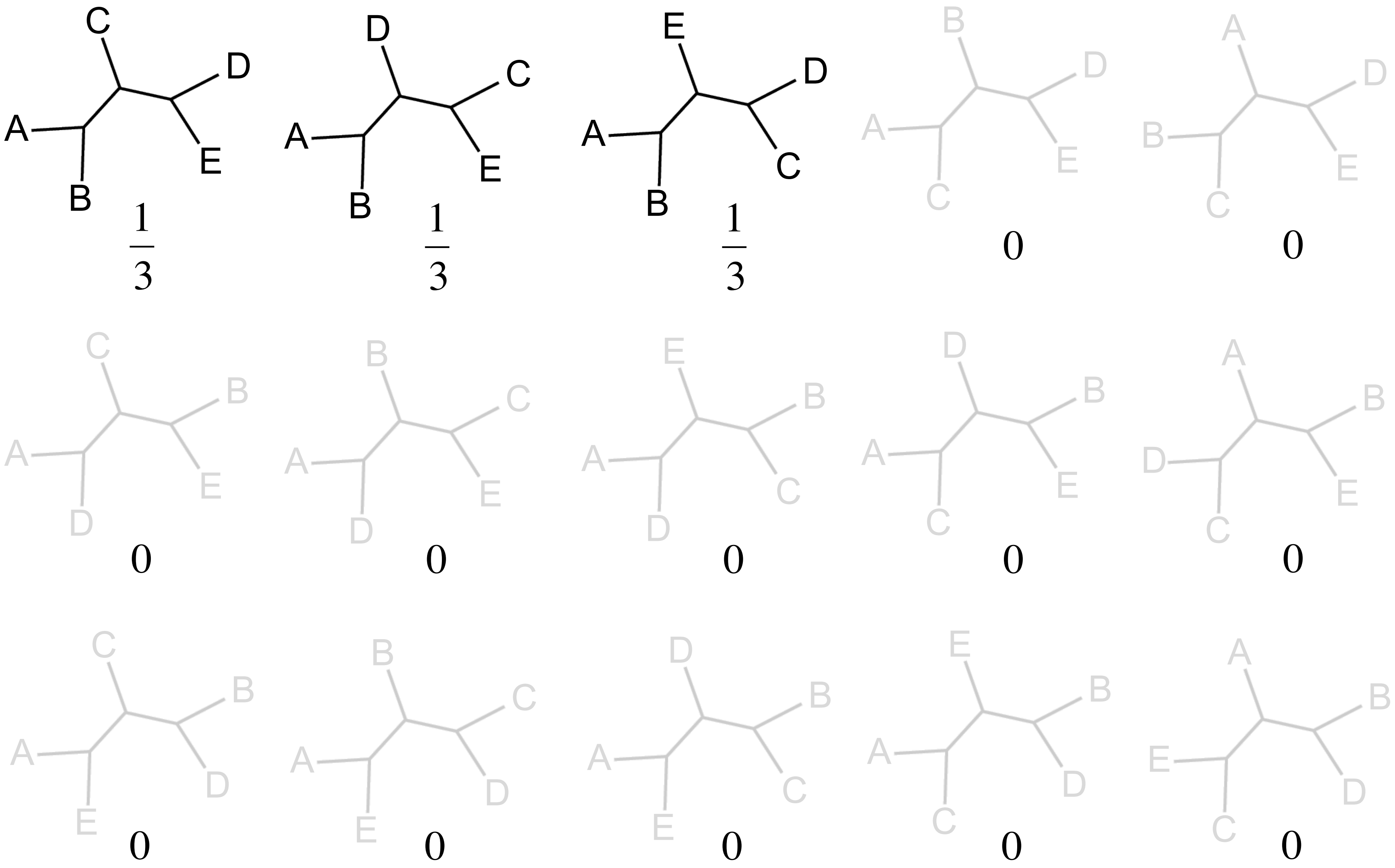


$\frac{1}{15}$

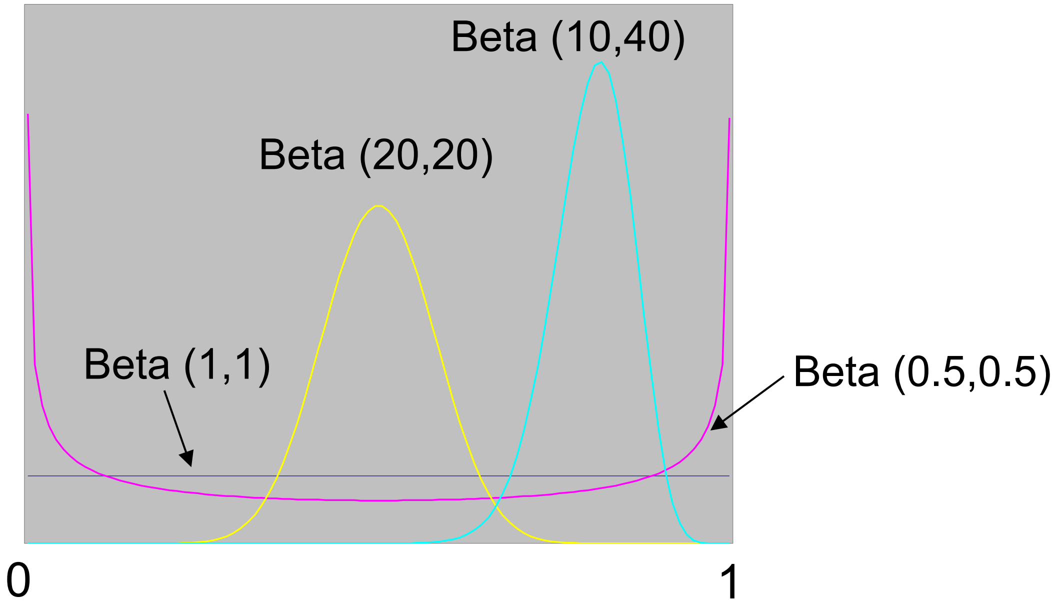


$\frac{1}{15}$

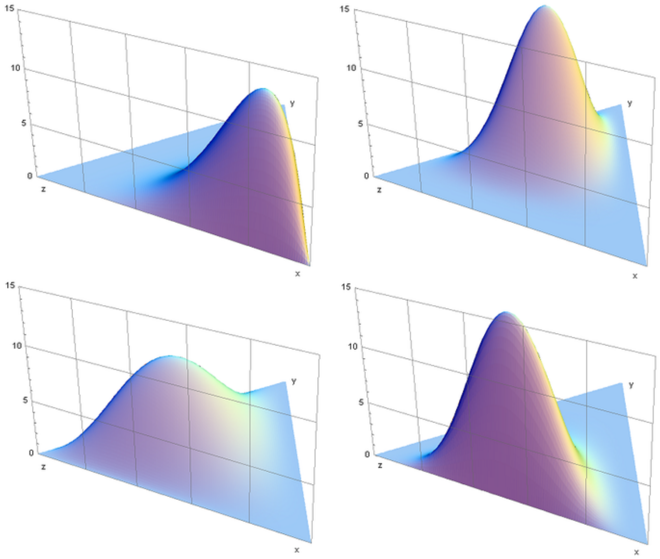
# (AB|CDE) Constraint Prior

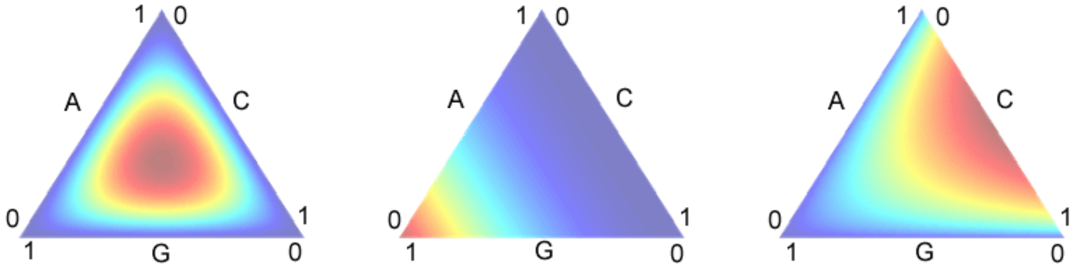


# Beta Distribution

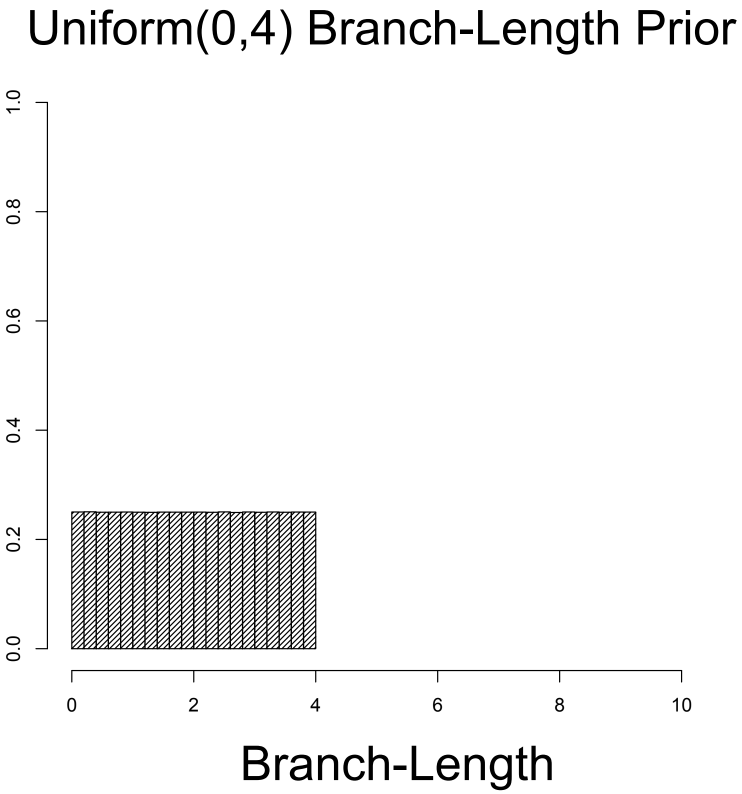


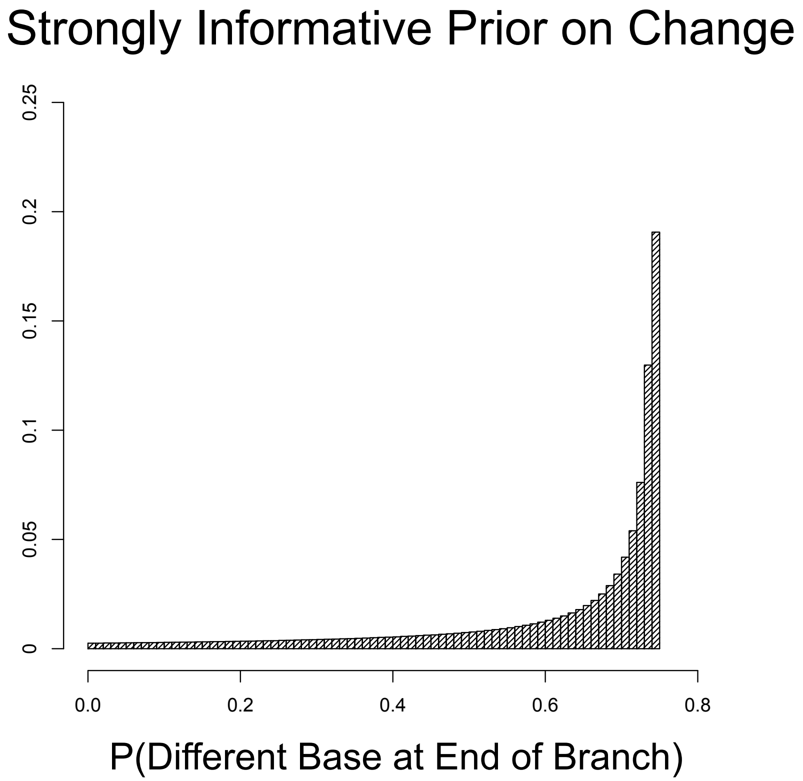
# Prior on Sets - Dirichlet Distribution





# Branch-Length Priors

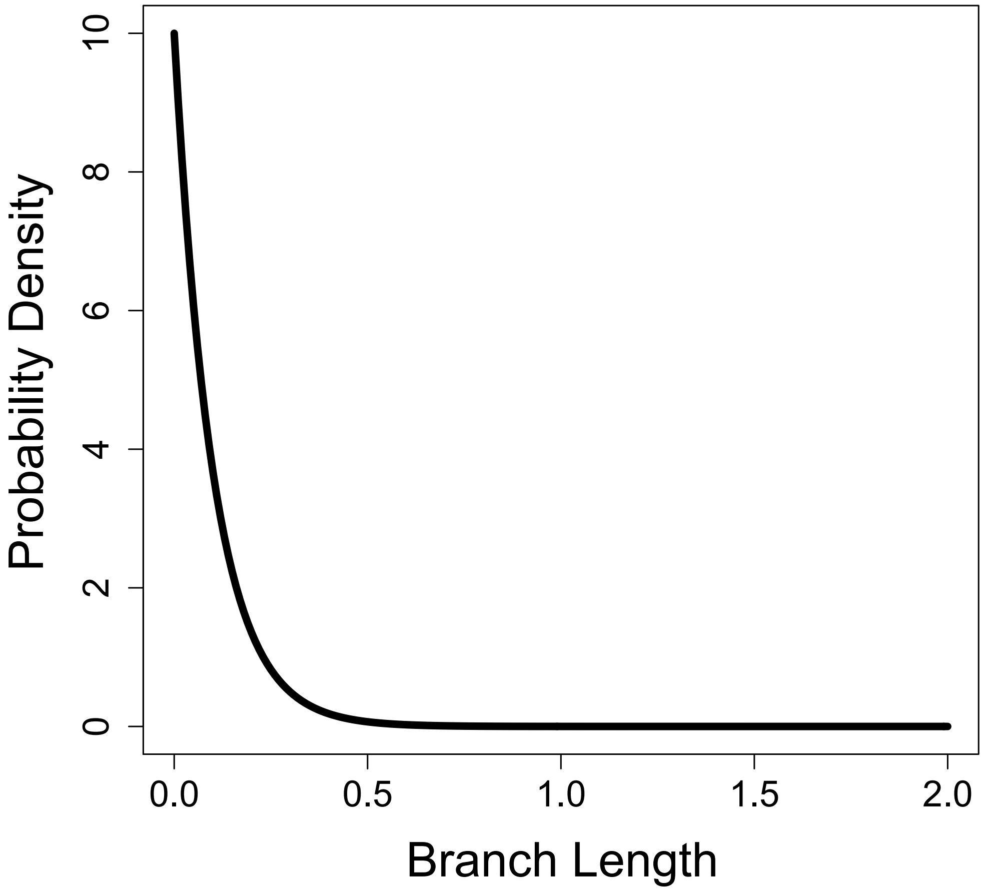




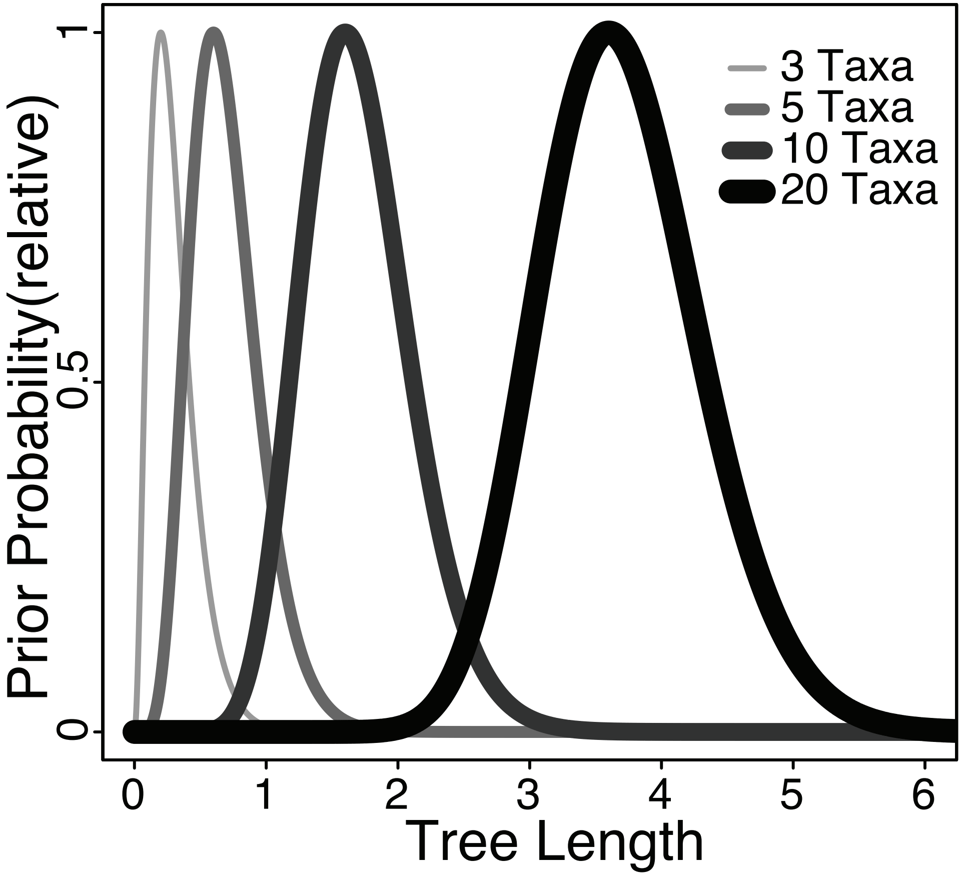
Idea from P.O. Lewis

# Branch-Length Priors

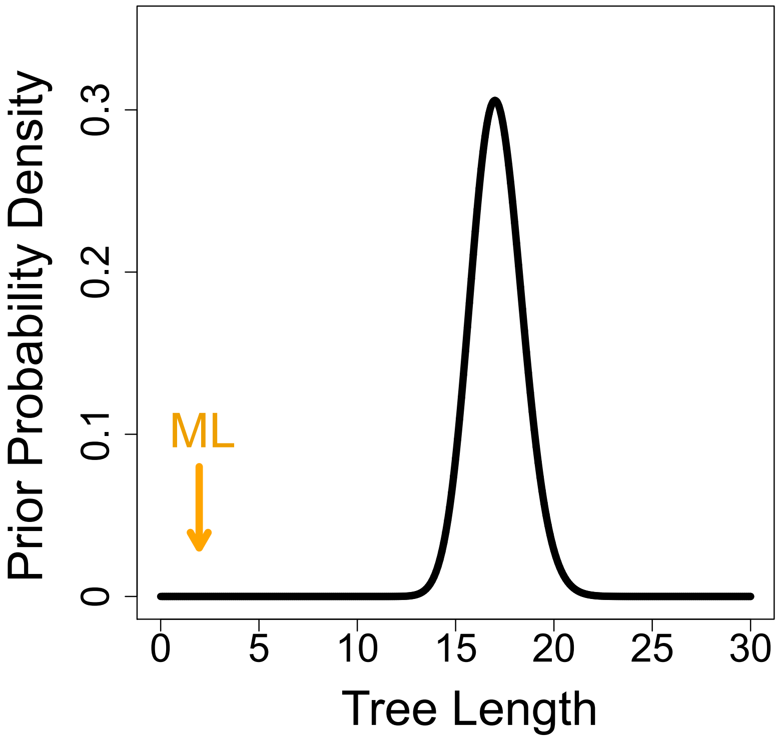
Default Exponential Branch-Length Prior (λ=10, mean=0.1)

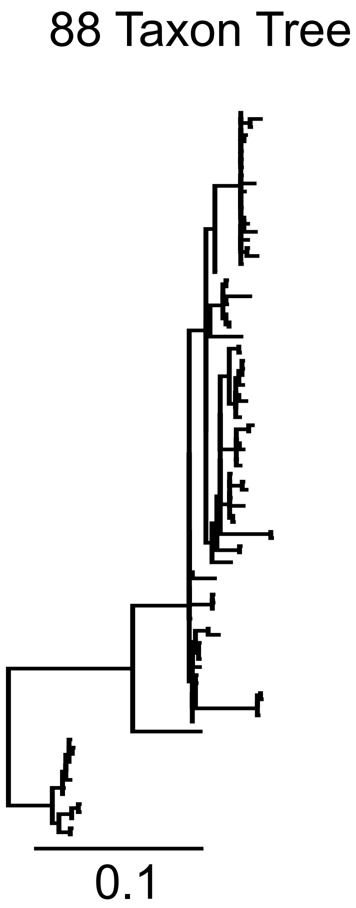


# Implied Tree-Length Prior

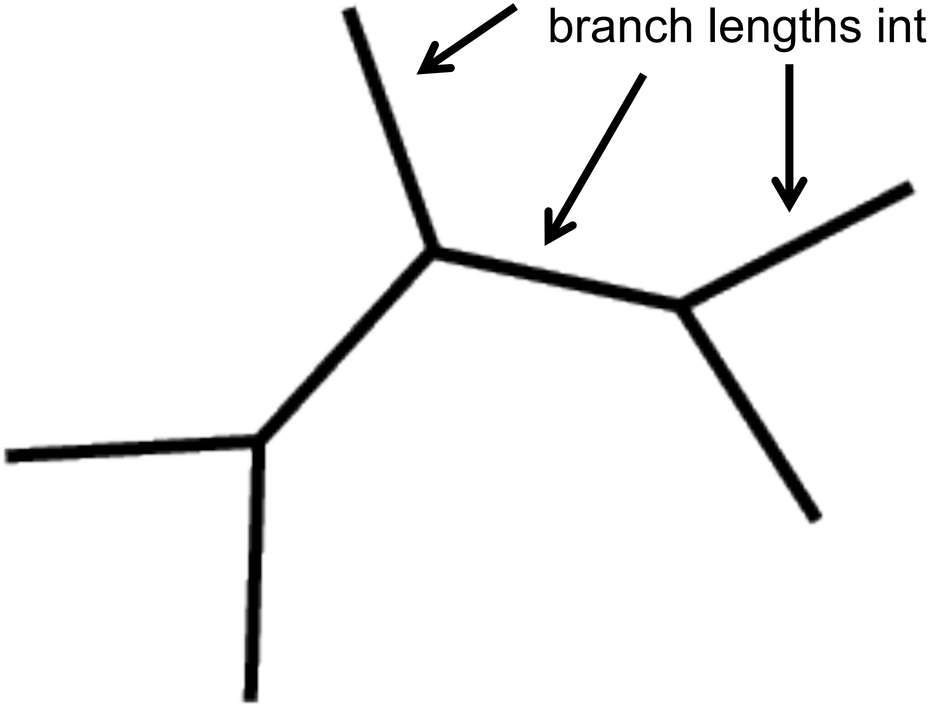


# Implied Tree-Length Prior





# Compound Branch-Length Prior



Dirichlet prior on partitioning of branch lengths into total tree length

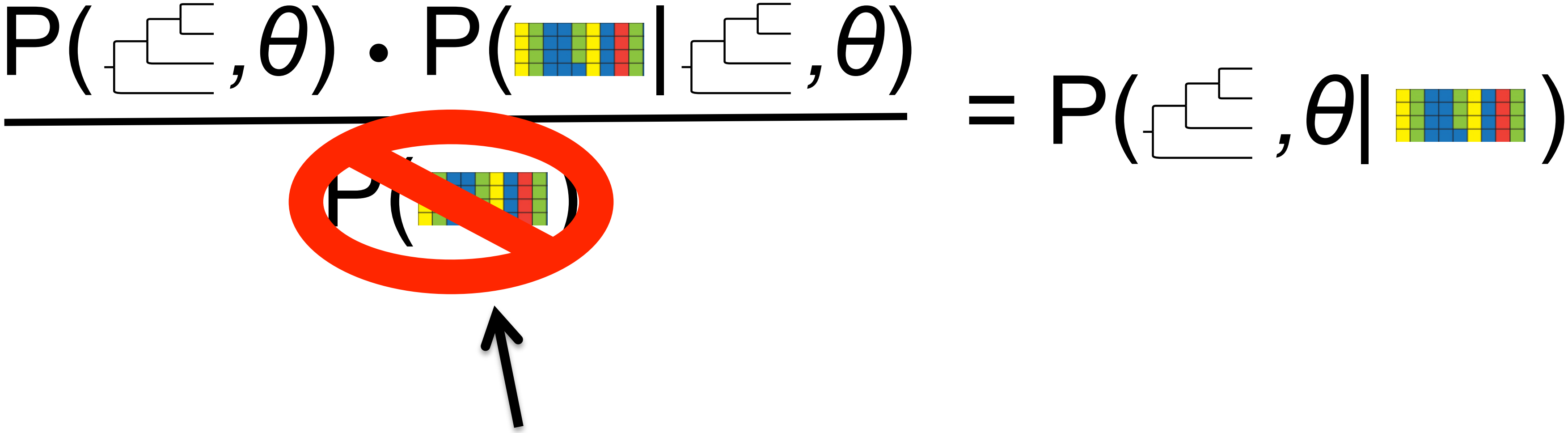
Total Tree Length Prior: (Inv) Gamma

Rannala et al. 2011. Mol. Biol. Evol.
Zhang et al. 2012. Syst. Biol

# Estimated Tree Lengths

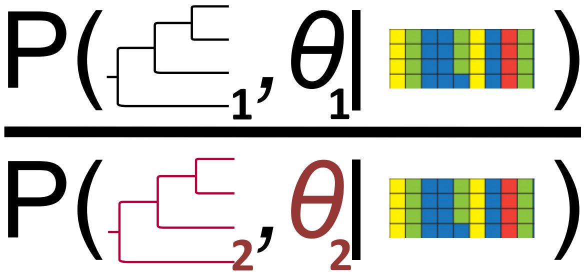
| Dataset | Clams | Frogs |
|---|---|---|
| **ML TL Estimate** | 1.96 | 0.55 |
| **Bayes TL Interval (MB Default)** | 10.7 - 17.7 | 1.77-3.29 |
| **Bayes TL Interval (Informed)** | 1.15 – 1.43 | 0.32 – 0.38 |
| **Bayes TL Interval (Compound Prior)** | 0.9 – 1.3 | 0.44 – 0.57 |

# Bayes’ Theorem

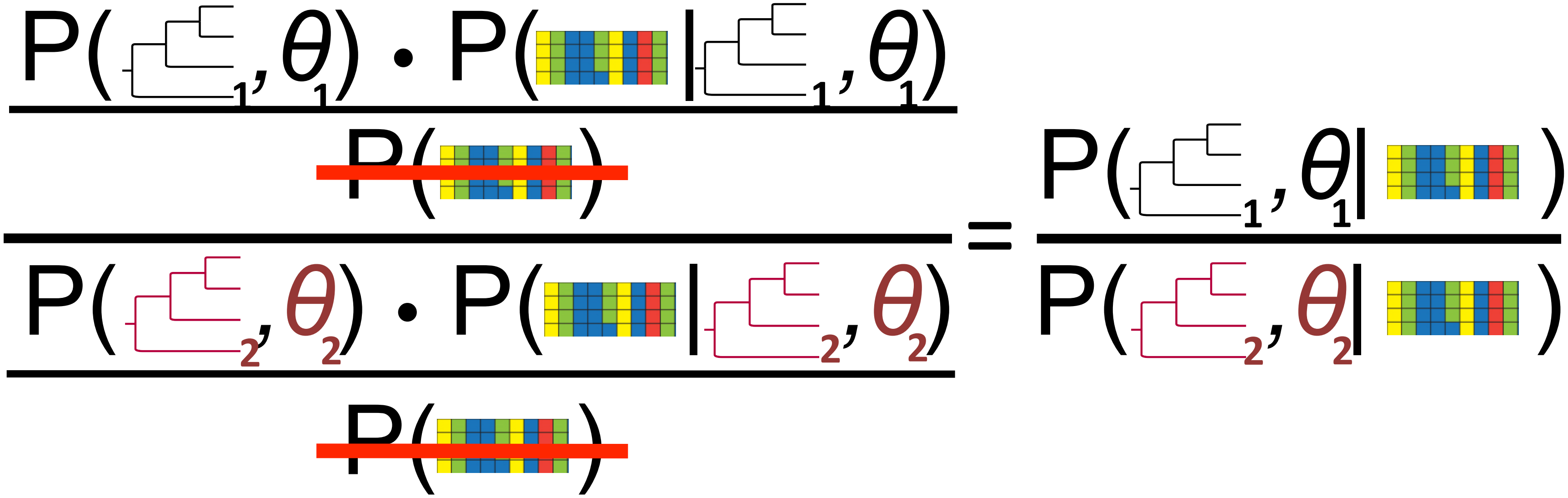


**No Analytical Solution**

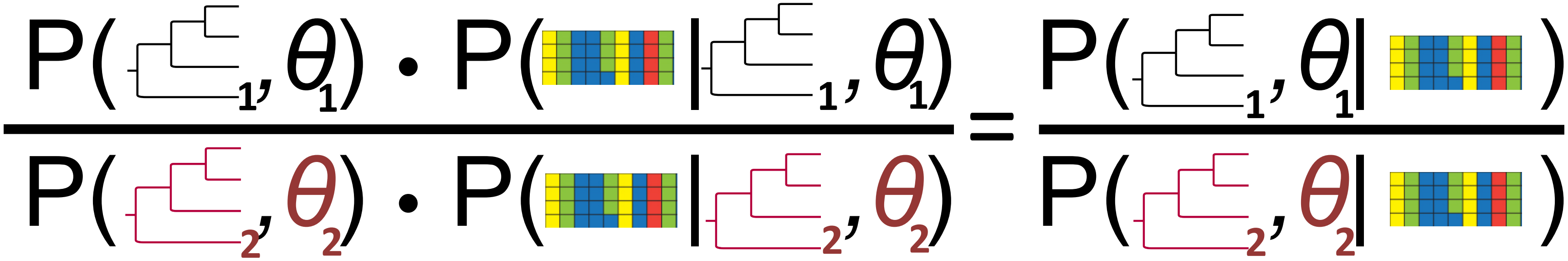
# Posterior Odds Ratio



# Posterior Odds Ratio

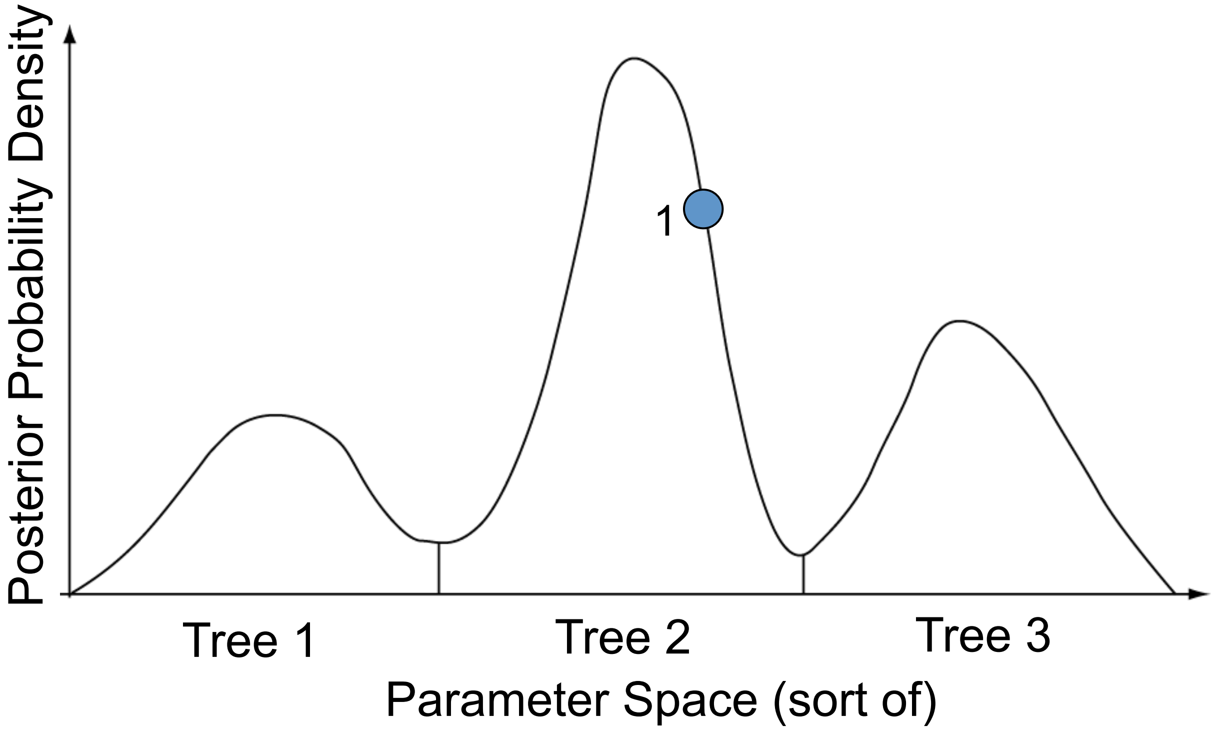


# Posterior Odds Ratio



# Markov chain Monte Carlo

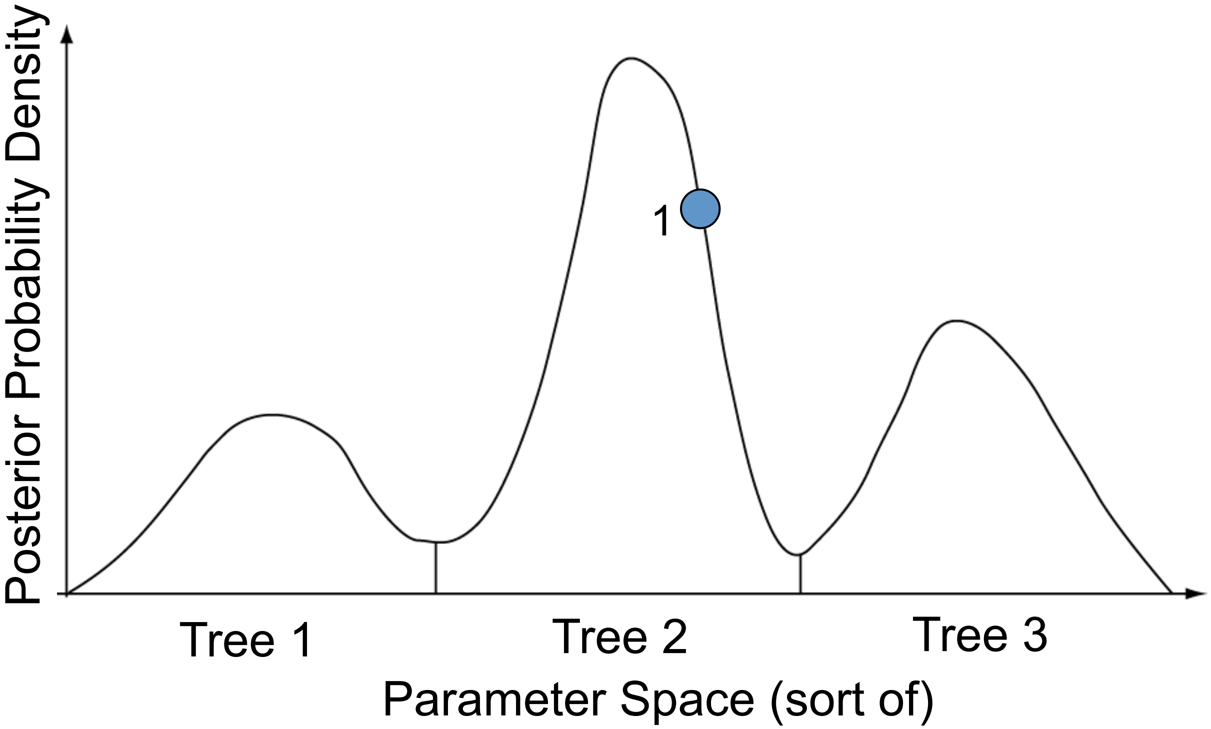
1. Start at an arbitrary point



*This slide "borrowed" from F. Ronquist*

# Markov chain Monte Carlo

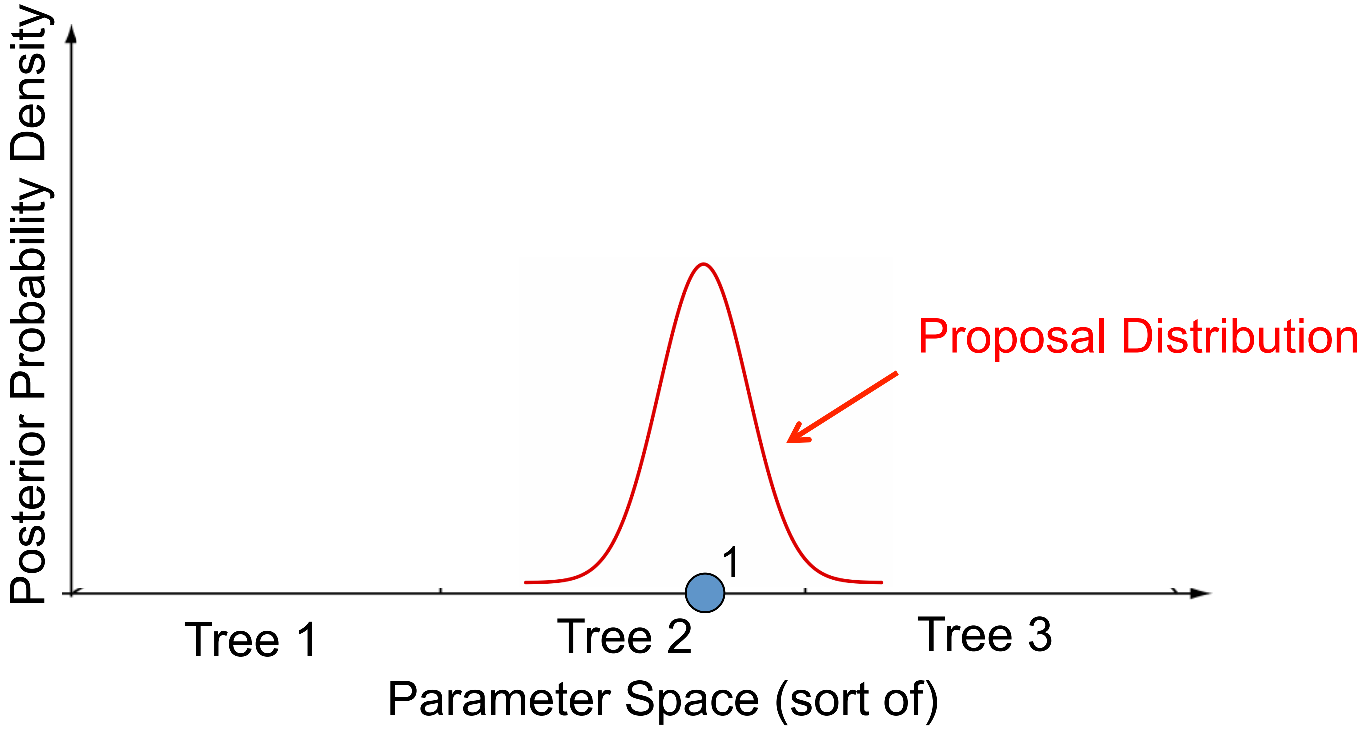
1. Start at an arbitrary point
2. Make a small random move



*This slide "borrowed" from F. Ronquist*

# Markov chain Monte Carlo

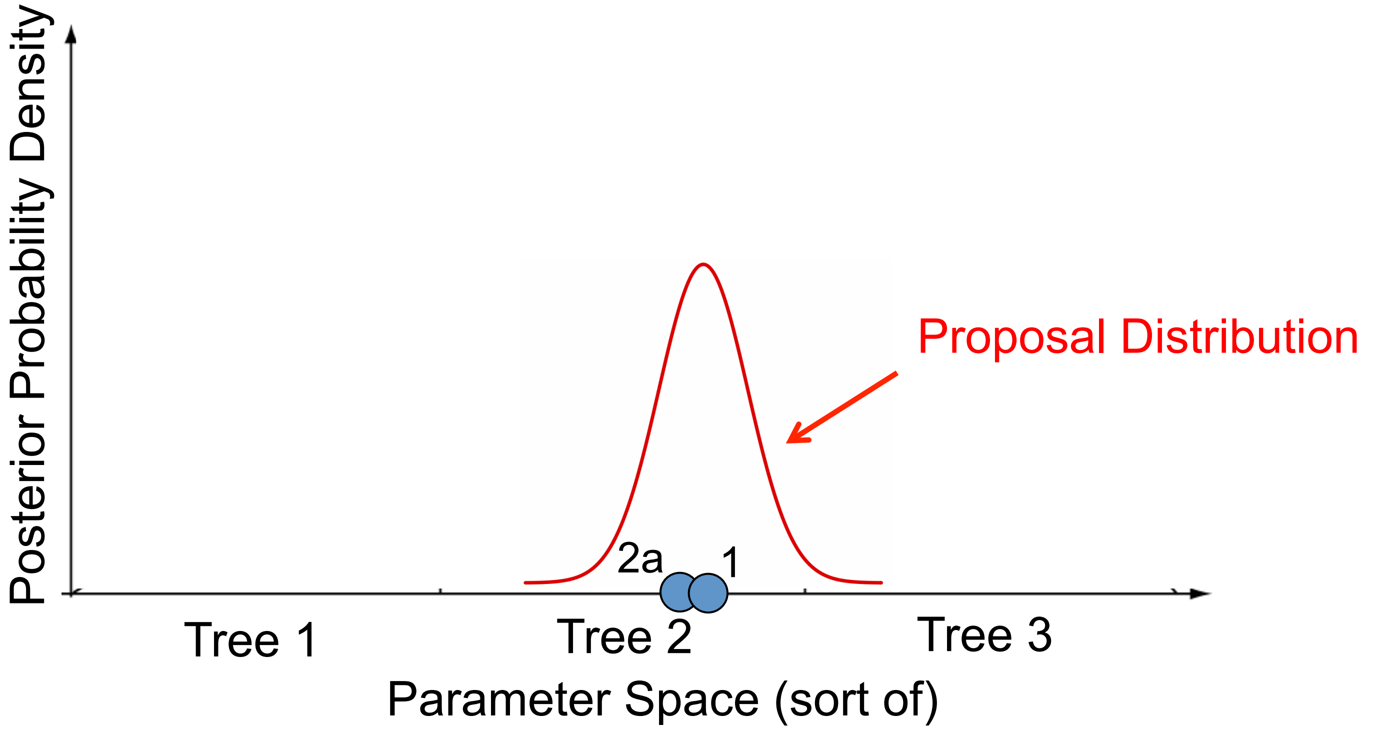
1. Start at an arbitrary point
2. Make a small random move



*This slide "borrowed" from F. Ronquist*

# Markov chain Monte Carlo

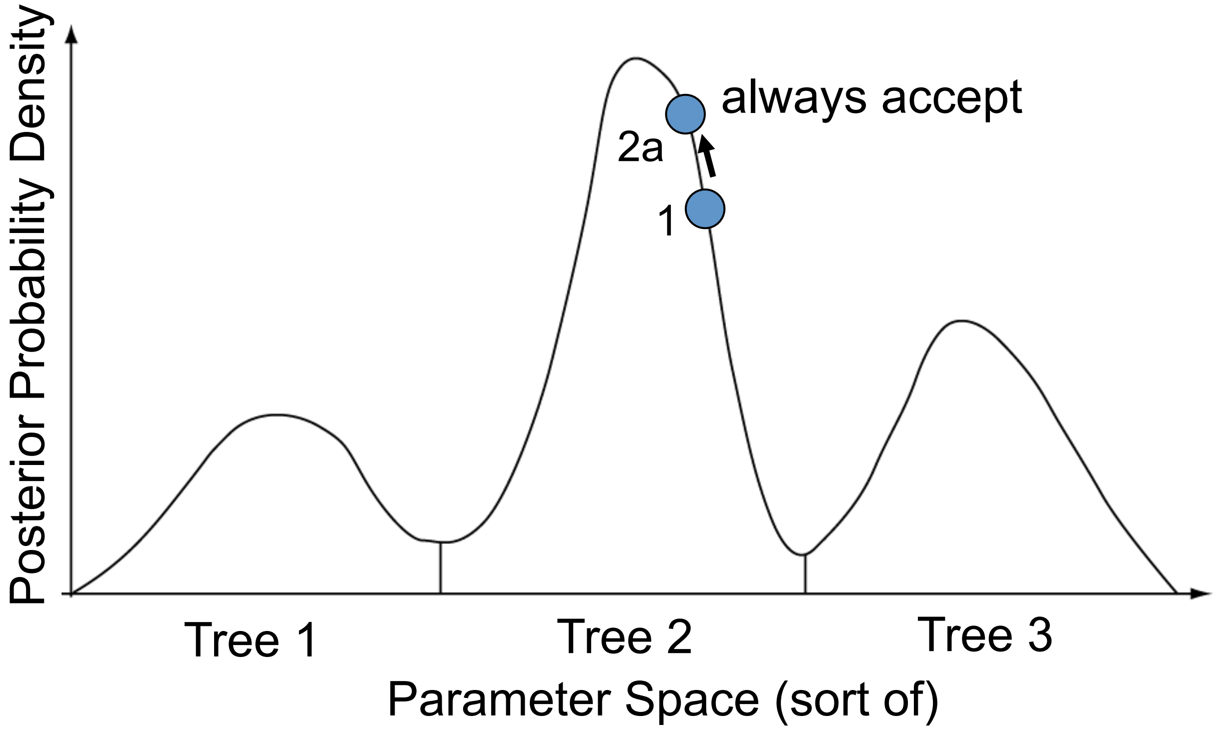
1. Start at an arbitrary point
2. Make a small random move



*This slide "borrowed" from F. Ronquist*

# Markov chain Monte Carlo

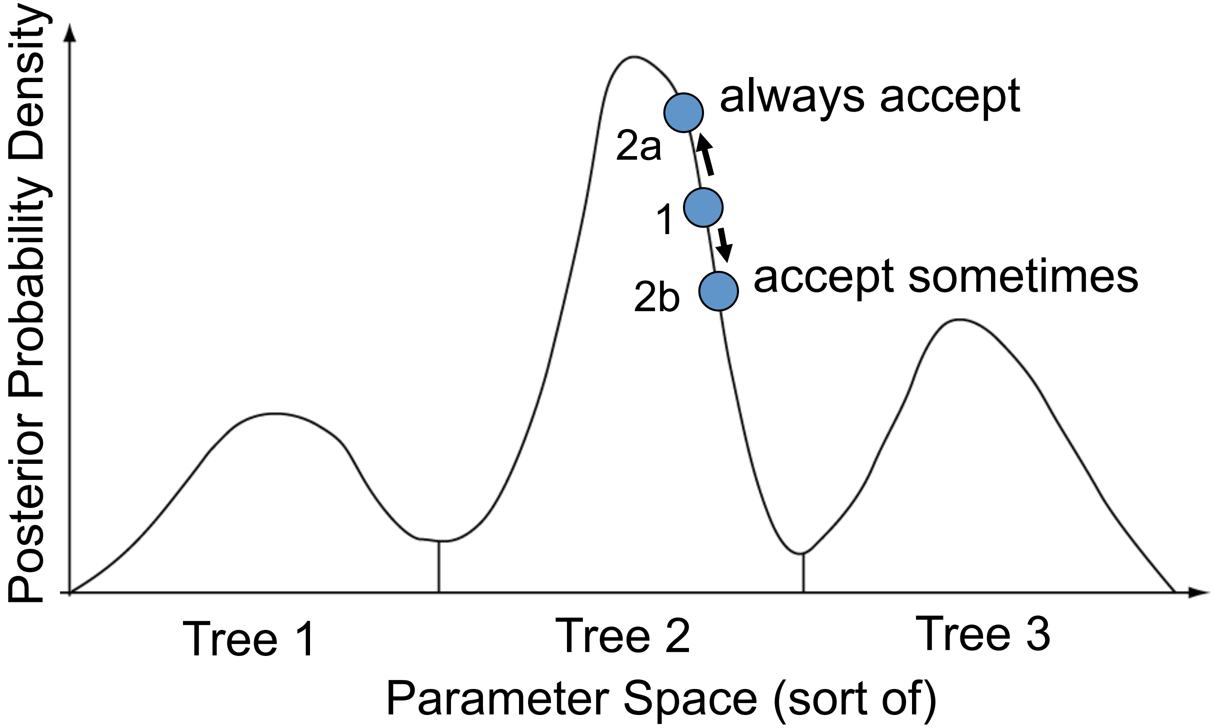
1. Start at an arbitrary point
2. Make a small random move
3. Calculate posterior density ratio (*r*) of new state to old state:
   a) r > 1 -> new state accepted
   b) r < 1 -> new state accepted with probability *r*. If new state not accepted, stay in the old state



*This slide "borrowed" from F. Ronquist*

# Markov chain Monte Carlo

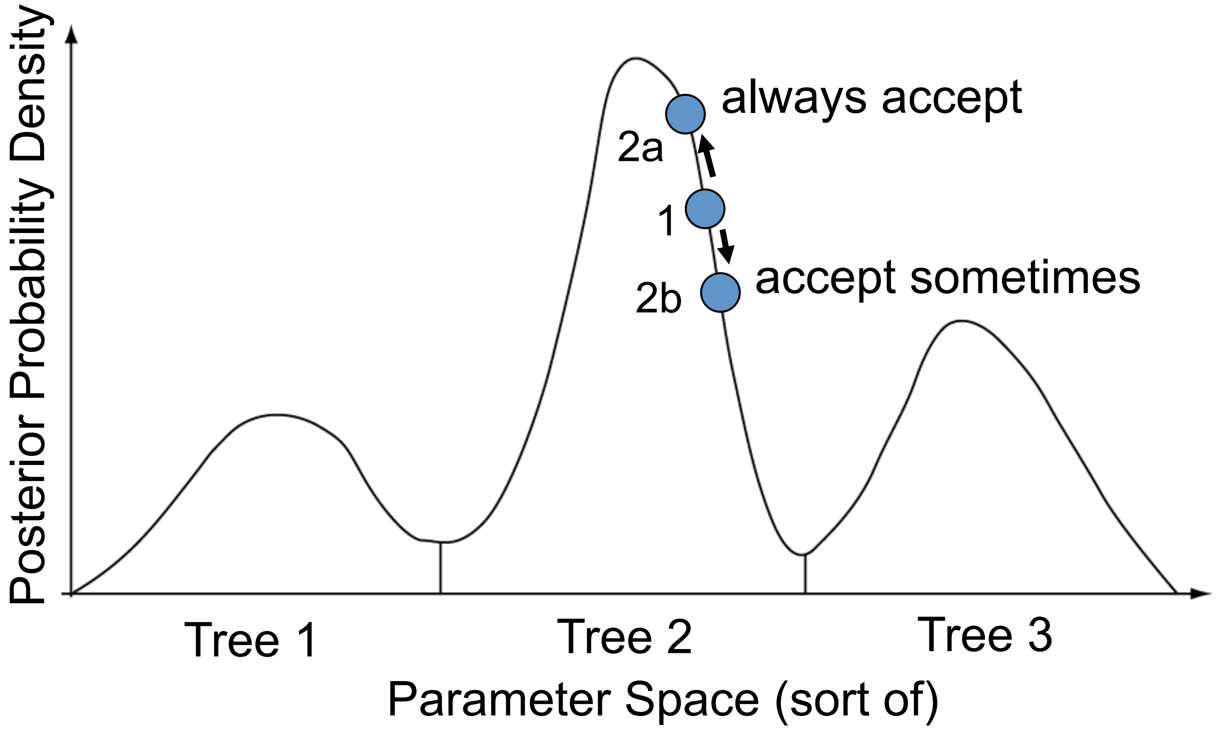
1. Start at an arbitrary point
2. Make a small random move
3. Calculate posterior density ratio (*r*) of new state to old state:
   a) r > 1 -> new state accepted
   b) r < 1 -> new state accepted with probability *r*. If new state not accepted, stay in the old state



*This slide "borrowed" from F. Ronquist*

# Markov chain Monte Carlo

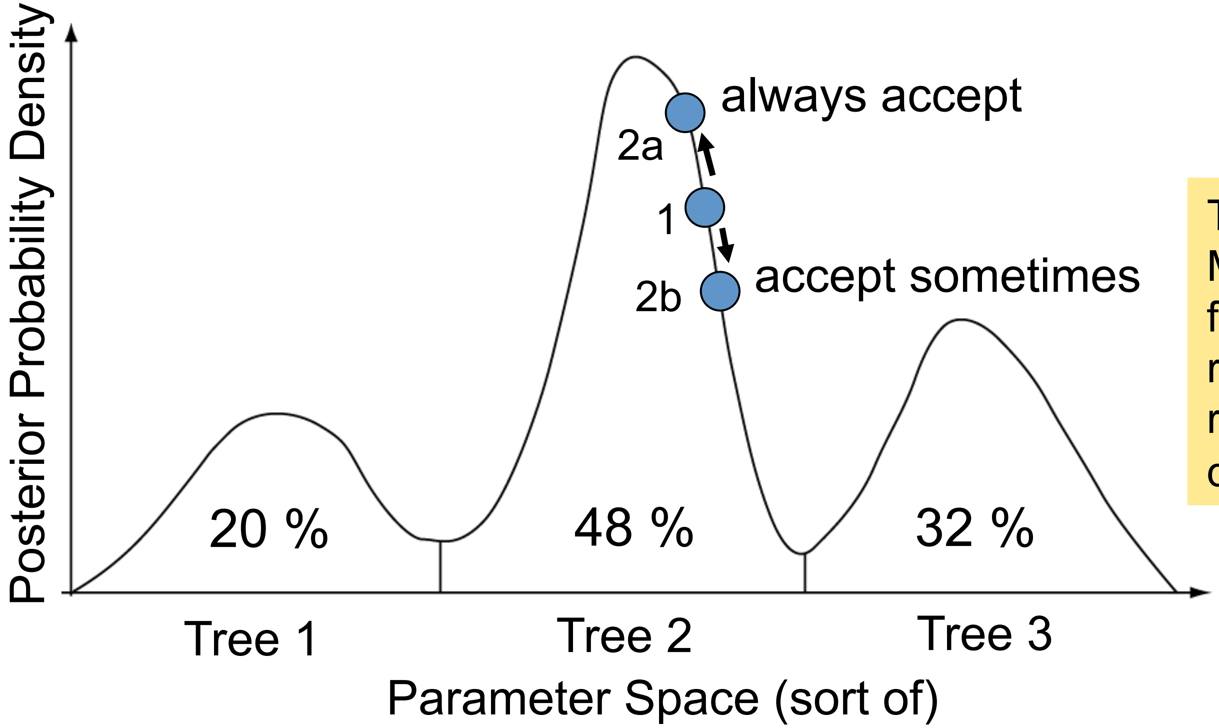
1. Start at an arbitrary point
2. Make a small random move
3. Calculate posterior density ratio (*r*) of new state to old state:
   a) r > 1 -> new state accepted
   b) r < 1 -> new state accepted with probability *r*. If new state not accepted, stay in the old state
4. Go to step 2 a BUNCH (x 10,000’s – x 10,000,000’s)



*This slide “borrowed” from F. Ronquist*

# Markov chain Monte Carlo

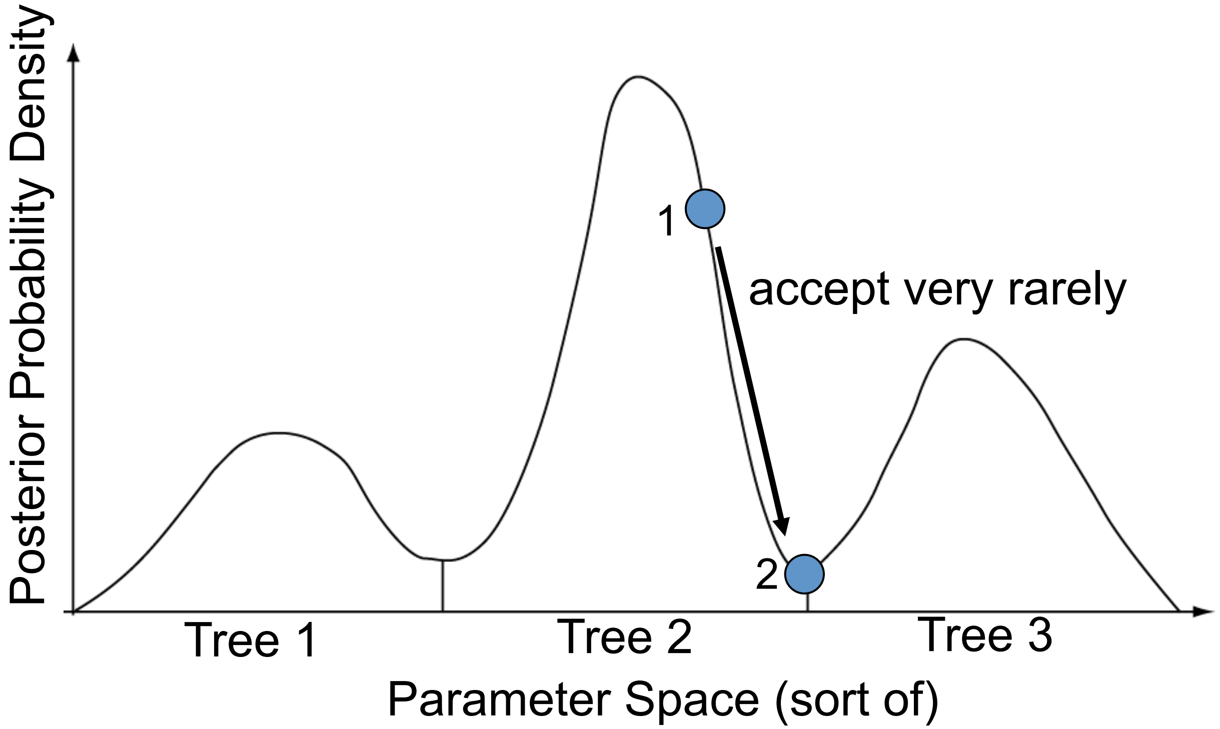
1. Start at an arbitrary point
2. Make a small random move
3. Calculate posterior density ratio (*r*) of new state to old state:
   a) r > 1 -> new state accepted
   b) r < 1 -> new state accepted with probability *r*. If new state not accepted, stay in the old state
4. Go to step 2 a BUNCH (x 10,000's – x 10,000,000's)



The proportion of time the MCMC procedure samples from a particular parameter region is an estimate of that region's posterior probability density

*This slide "borrowed" from F. Ronquist*

# Metropolis Coupling

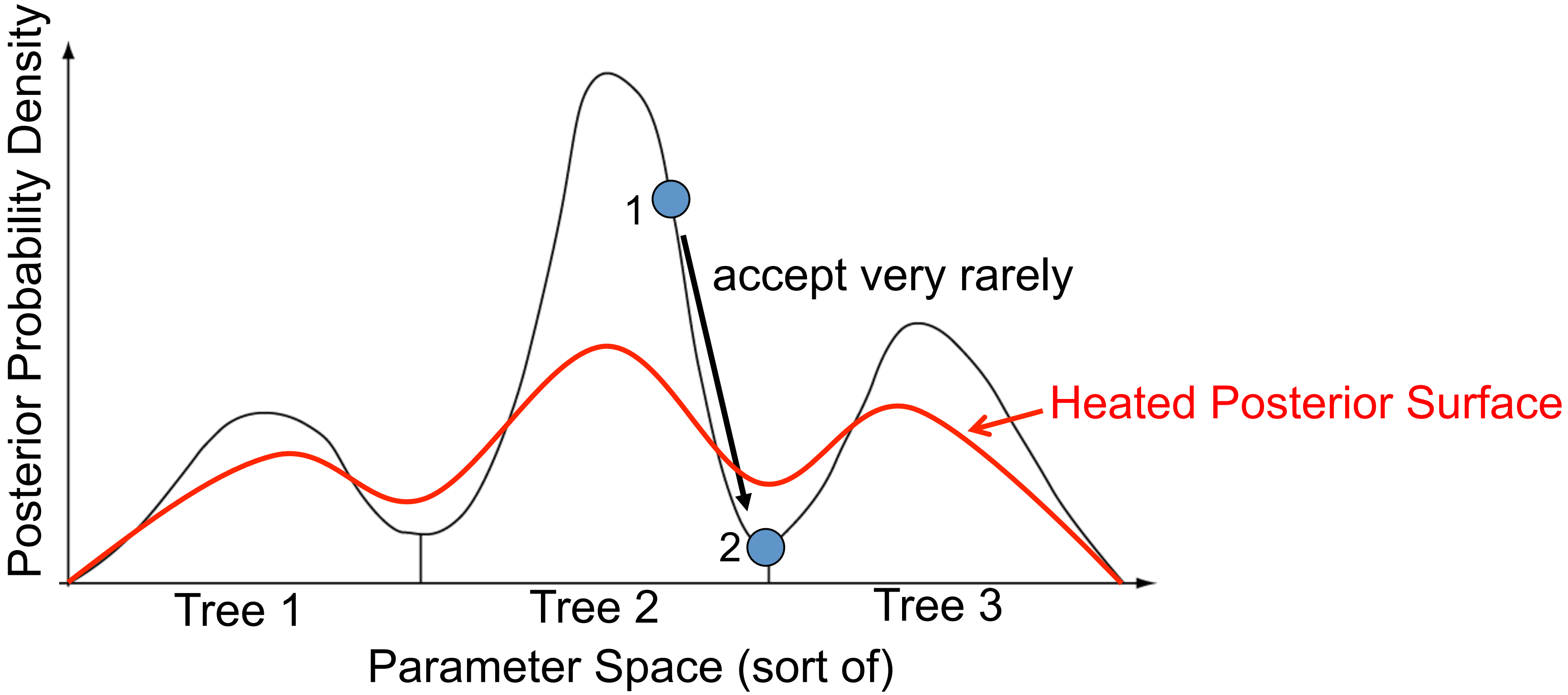


*This slide "borrowed" from F. Ronquist*

# Metropolis Coupling

- Same rules as regular MCMC, but now there are multiple chains with different ‘temperatures’.
- ‘Heated’ chains sample a ‘melted’ version of the posterior
- Only difference is that heated chains raise the ratio of posterior densities to (1-temp) when deciding whether to accept a move.

$r^{(1\text{-temp})}$ approaches 1 (flat surface) as temp. increases

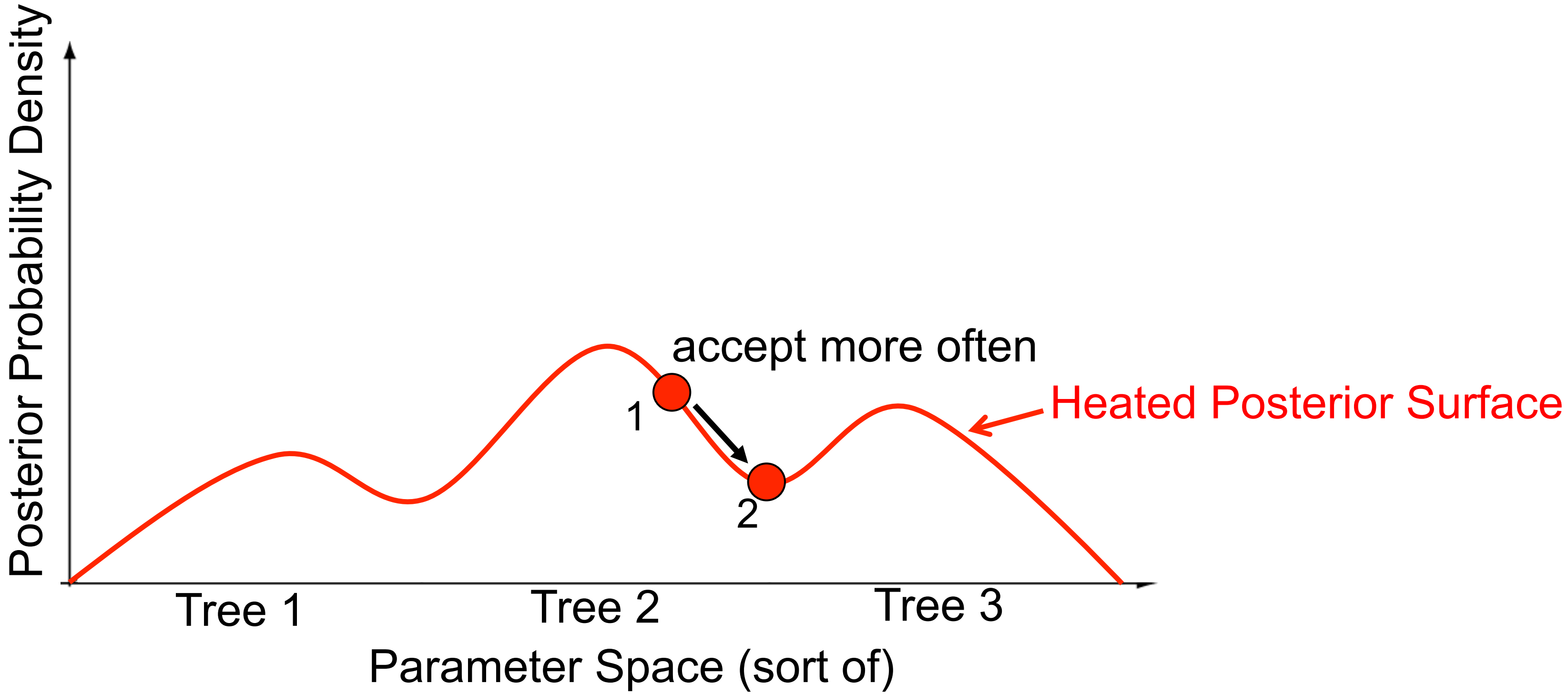


*This slide “borrowed” from F. Ronquist*

# Metropolis Coupling

- Same rules as regular MCMC, but now there are multiple chains with different ‘temperatures’.
- ‘Heated’ chains sample a ‘melted’ version of the posterior
- Only difference is that heated chains raise the ratio of posterior densities to (1-temp) when deciding whether to accept a move.

$r^{(1\text{-temp})}$ approaches 1 (flat surface) as temp. increases



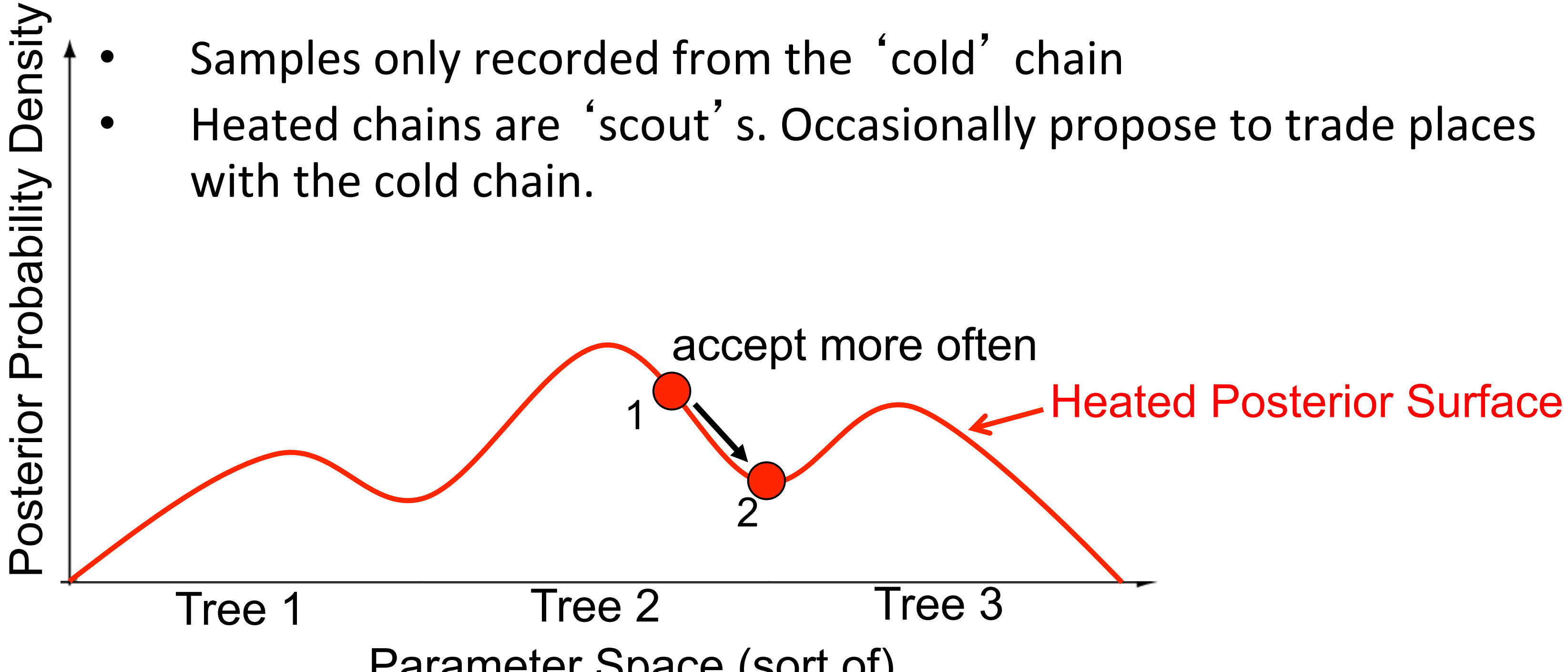
*This slide “borrowed” from F. Ronquist*

# Metropolis Coupling

- Same rules as regular MCMC, but now there are multiple chains with different ‘temperatures’.
- ‘Heated’ chains sample a ‘melted’ version of the posterior
- Only difference is that heated chains raise the ratio of posterior densities to (1-temp) when deciding whether to accept a move.

$r^{(1\text{-temp})}$ approaches 1 (flat surface) as temp. increases

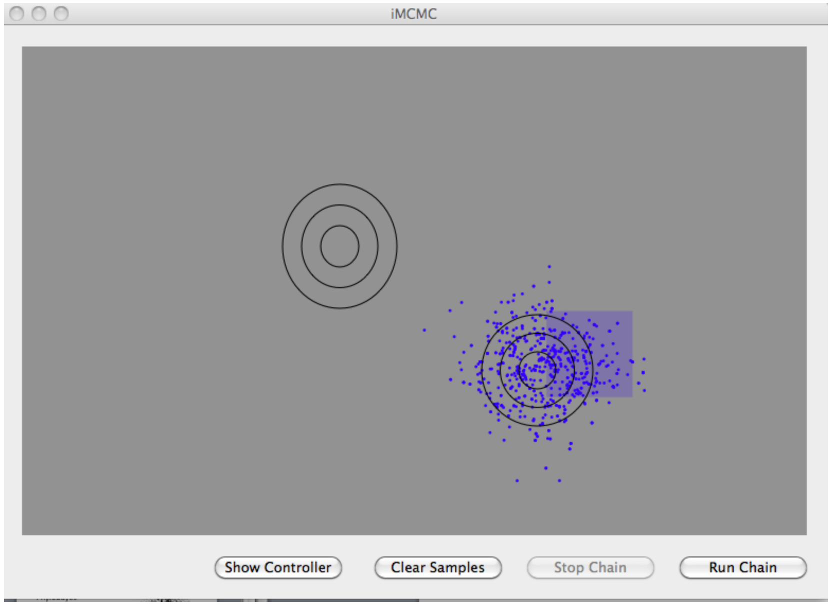
- Samples only recorded from the ‘cold’ chain
- Heated chains are ‘scout’s. Occasionally propose to trade places with the cold chain.

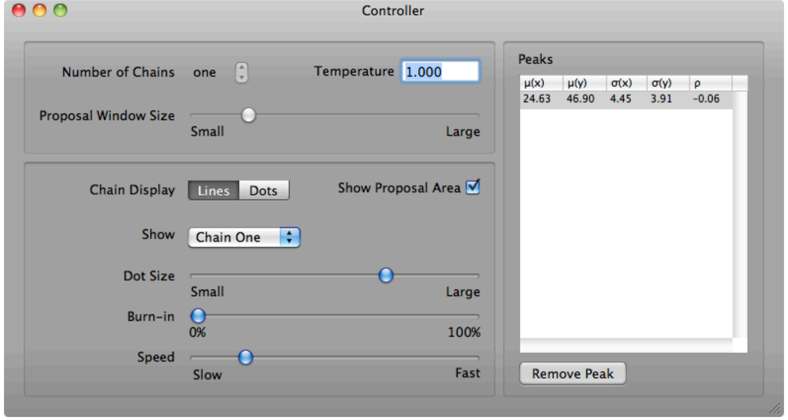


*This slide “borrowed” from F. Ronquist*

# Toy MCMC Demonstration

- MCRobot – PC (Lewis)
  - http://www.eeb.uconn.edu/people/plewis/software.php
- iMCMC – Mac (Huelsenbeck)
  - http://fisher.berkeley.edu/cteg/software.html





Number of Chains one

Temperature 1.000

Chain Display [✓ Lines] Dots

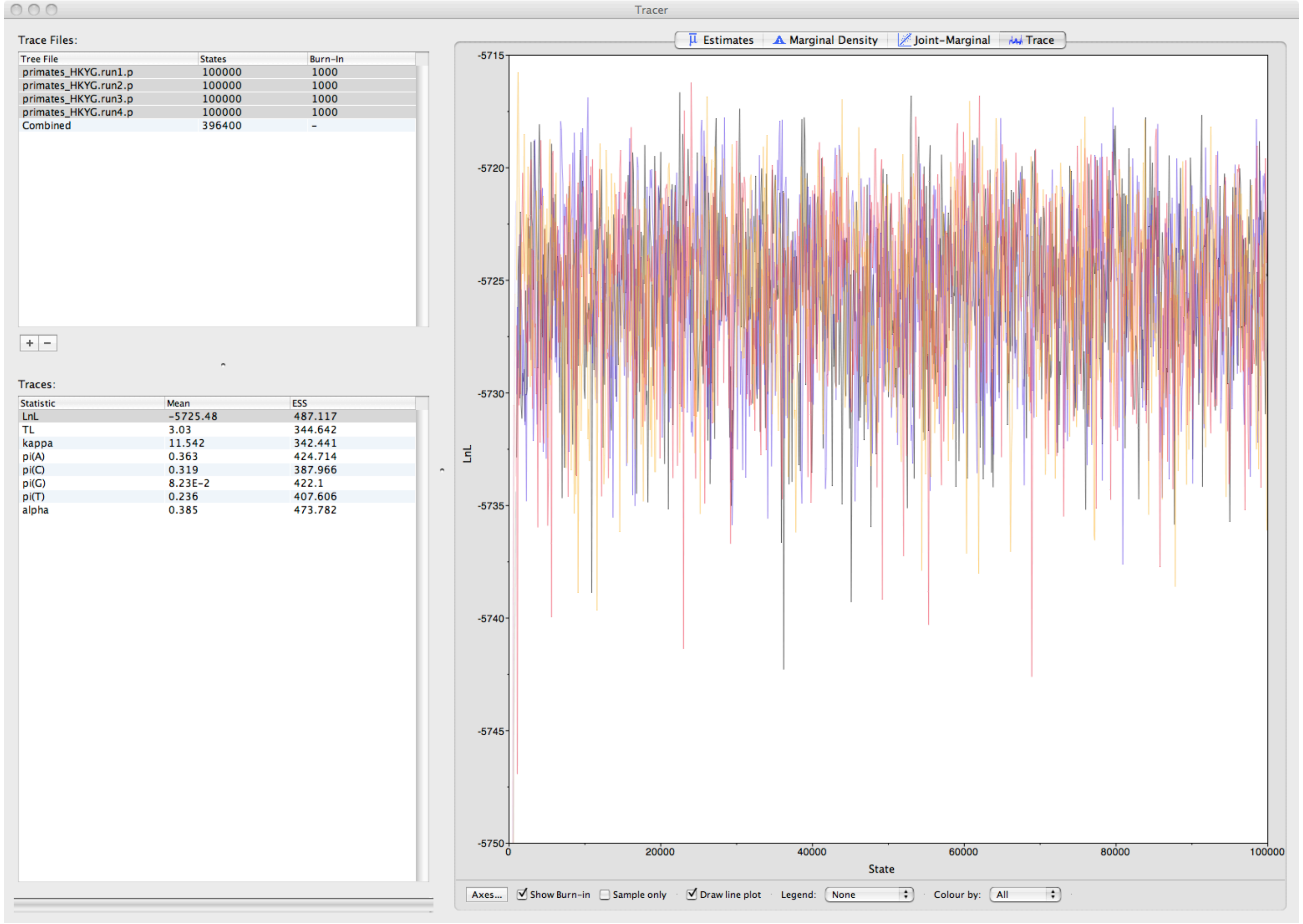
Show Proposal Area ☑

Show Chain One

Peaks

| μ(x) | μ(y) | σ(x) | σ(y) | ρ |
|---|---|---|---|---|
| 24.63 | 46.90 | 4.45 | 3.91 | -0.06 |

# Convergence of Scalars - Tracer



Trace Files:

| Tree File | States | Burn-In |
|---|---|---|
| primates_HKYG.run1.p | 100000 | 1000 |
| primates_HKYG.run2.p | 100000 | 1000 |
| primates_HKYG.run3.p | 100000 | 1000 |
| primates_HKYG.run4.p | 100000 | 1000 |
| Combined | 396400 | - |

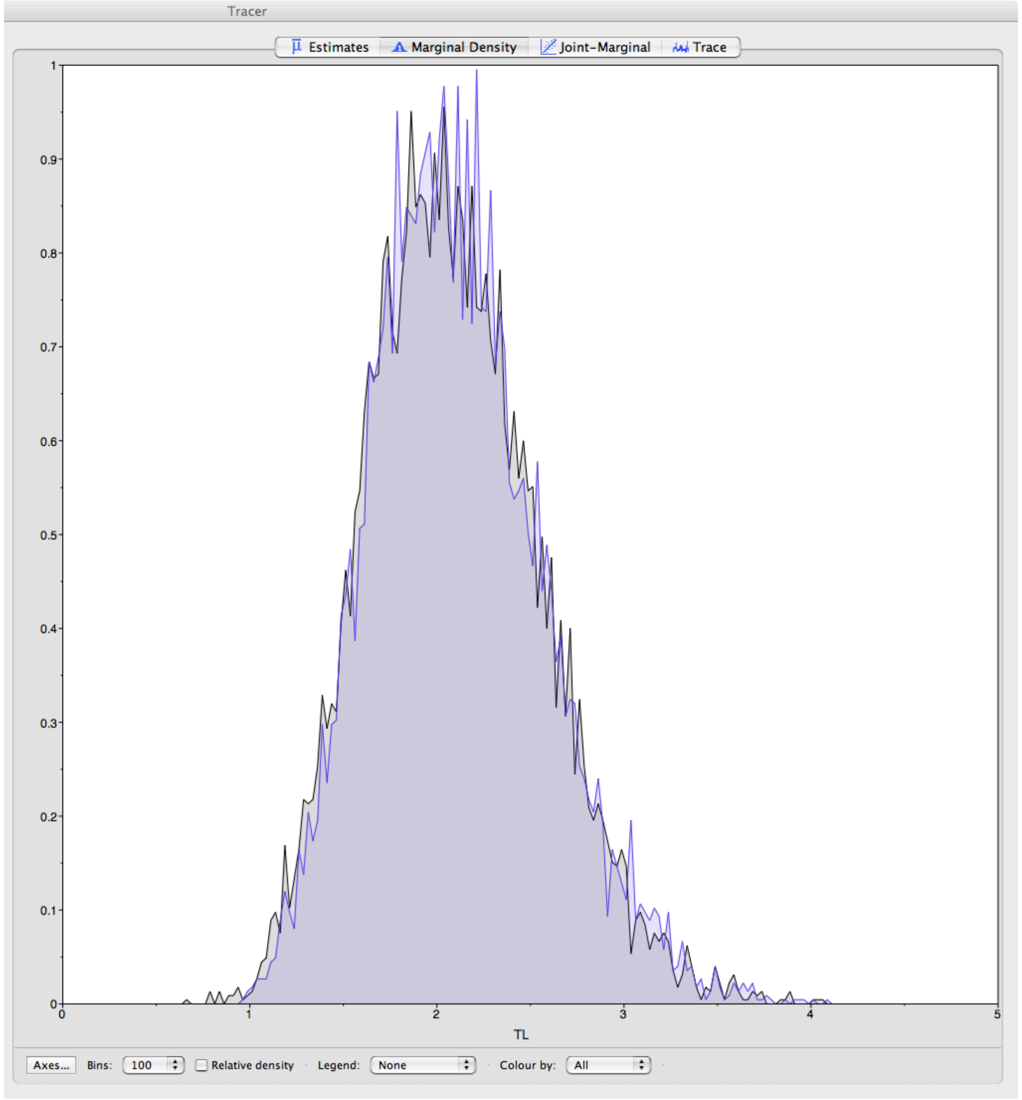
Traces:

| Statistic | Mean | ESS |
|---|---|---|
| LnL | -5725.48 | 487.117 |
| TL | 3.03 | 344.642 |
| kappa | 11.542 | 342.441 |
| pi(A) | 0.363 | 424.714 |
| pi(C) | 0.319 | 387.966 |
| pi(G) | 8.23E-2 | 422.1 |
| pi(T) | 0.236 | 407.606 |
| alpha | 0.385 | 473.782 |

☑ Show Burn-in □ Sample only ☑ Draw line plot Legend: None Colour by: All

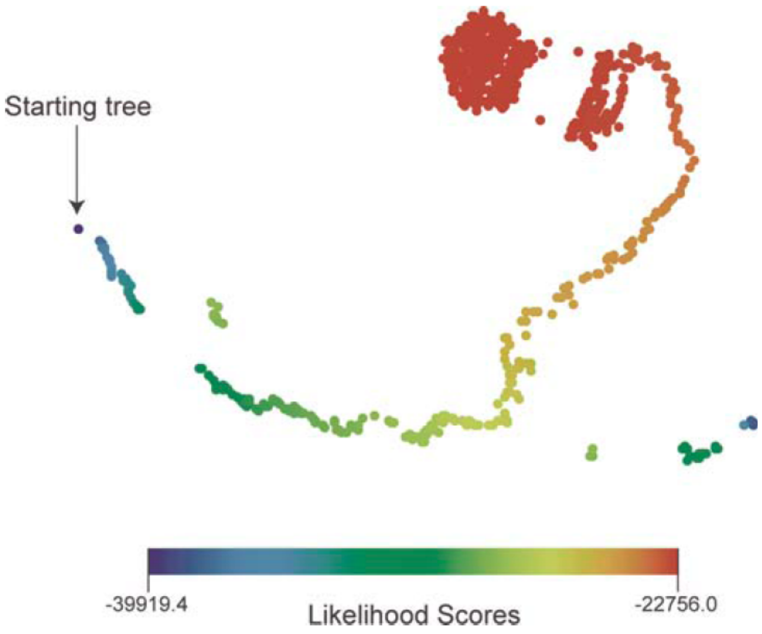
# Running on Empty

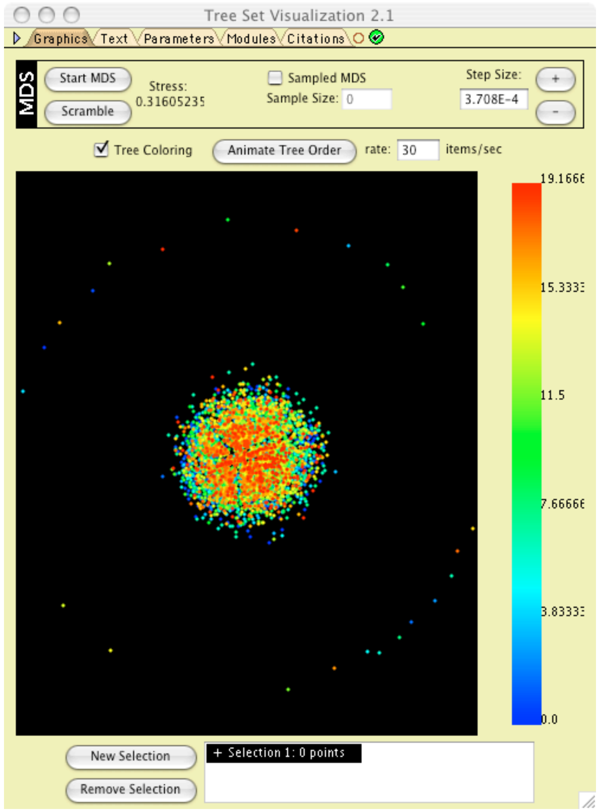
```
#NEXUS
begin data;
dimensions ntax=12 nchar=5;
format datatype=dna interleave=no gap=- missing=?;
matrix
Tarsius_syrichta    ?????
Lemur_catta         ?????
Homo_sapiens        ?????
Pan                 ?????
Gorilla             ?????
Pongo               ?????
Hylobates           ?????
Macaca_fuscata      ?????
M_mulatta           ?????
M_fascicularis      ?????
M_sylvanus          ?????
Saimiri_sciureus    ?????
;
end;
```



Bins: 100
Legend: None
Colour by: All

# Topological Convergence - TreeSetViz





Stress: 0.31605235

□ Sampled MDS

Sample Size: 0

Step Size: 3.708E-4

☑ Tree Coloring

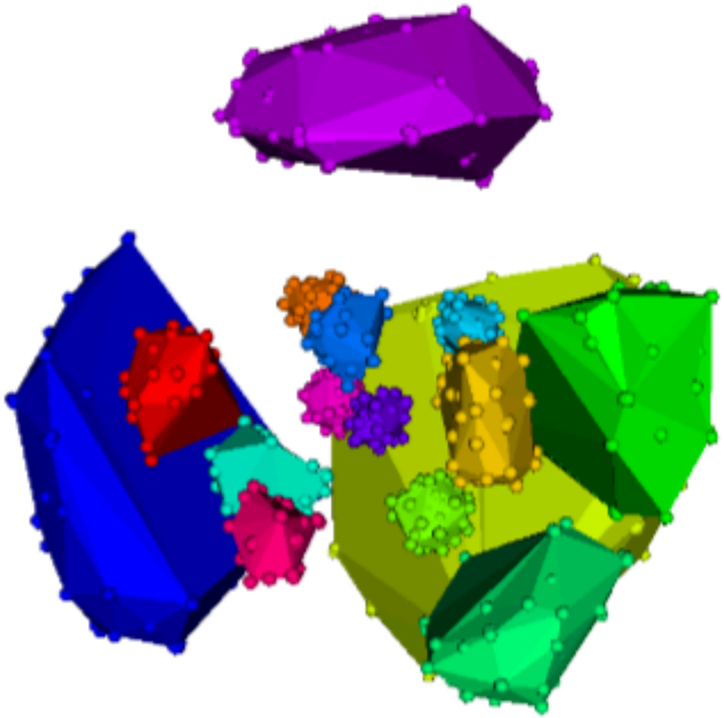
rate: 30 items/sec

TreeSetViz: http://comet.lehman.cuny.edu/treeviz/
Mesquite: http://www.mesquiteproject.org/mesquite/mesquite.html
Hillis et al., Analysis and Visualization of Tree Space, *Syst. Biol.*, 54(3): 471-482.

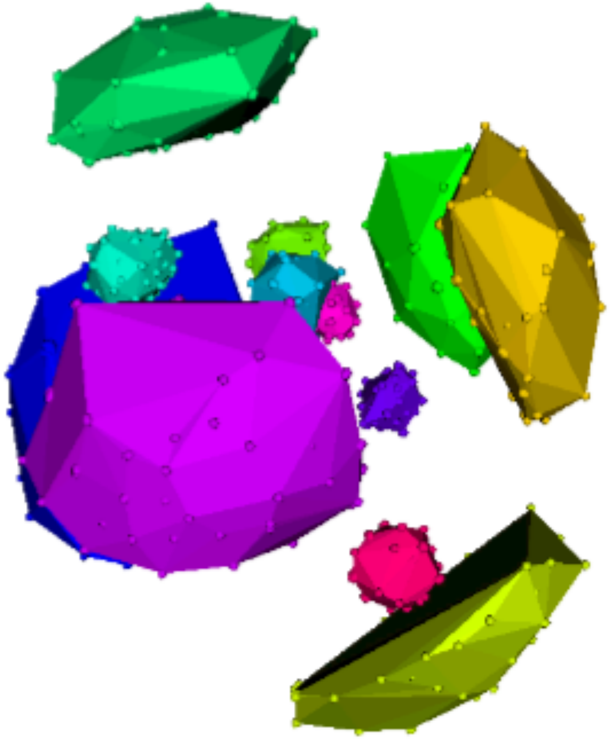
# Topological Convergence - TreeScaper

**Colors = Trees from Different Genes**

## Salamanders



## Mammals



http://bpd.sc.fsu.edu/index.php/diagnostic-software/104-treescaper

# Bayesian Phylogenetic Software

- MrBayes
- PhyloBayes
- BayesPhylogenies
- BEAST
- BEST
- BUCKy
- BAMBE
- Bali-Phy

# Bayesian Phylogenetic Software

- **MrBayes**
- PhyloBayes
- BayesPhylogenies
- BEAST
- BEST
- BUCKy
- BAMBE
- Bali-Phy

# MrBayes - Installation

Available at: http://mrbayes.sourceforge.net/

- – Executable versions for PC and Mac
- – Source code for compilation (command-line)
- – Serial, Parallel, and GPU-enabled (via Beagle)
- – Nice manual (in progress), tutorials, command reference
- – Current version: 3.2.1
    - Complete re-write (RevBayes) on its way

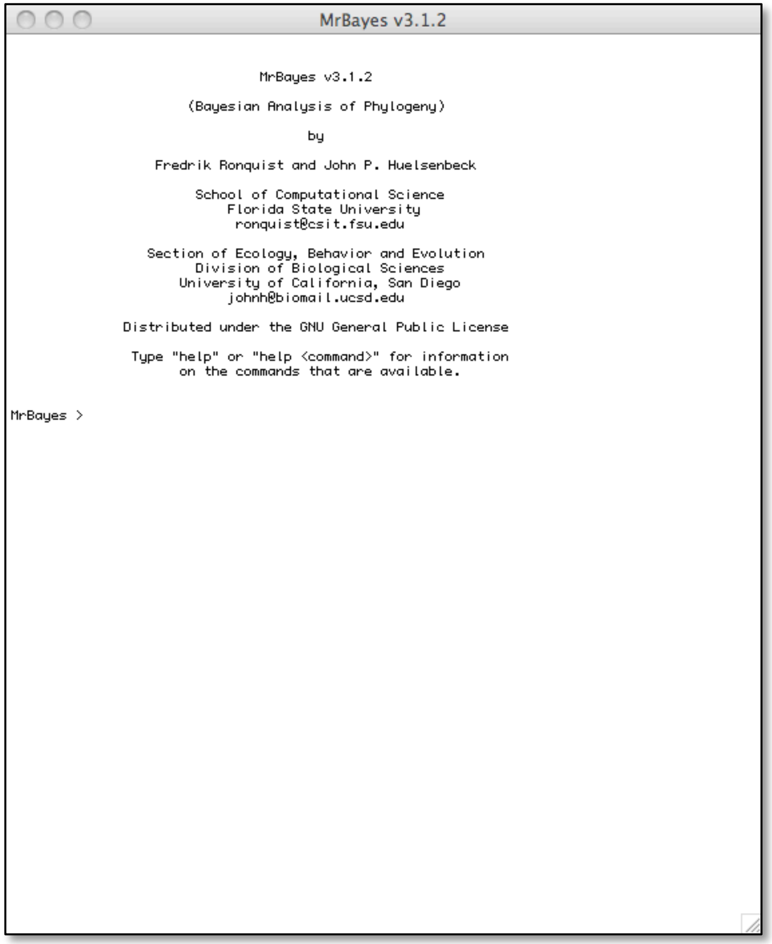
# MrBayes – Fire It Up

- Double click on icon or use command line in Unix

  > ./mb

   - or –

  > mb



```
                    MrBayes v3.1.2

          (Bayesian Analysis of Phylogeny)

                         by

        Fredrik Ronquist and John P. Huelsenbeck

            School of Computational Science
              Florida State University
               ronquist@csit.fsu.edu

        Section of Ecology, Behavior and Evolution
            Division of Biological Sciences
          University of California, San Diego
               johnh@biomail.ucsd.edu

      Distributed under the GNU General Public License

       Type "help" or "help <command>" for information
          on the commands that are available.


MrBayes >
```

# MrBayes – Help

*help*

- Use this command anytime you need more info
- No options will spit out a list of commands
- Help *command* gives more detail on that particular command.

**The most important command you will learn today!**

# MrBayes – First Command

*execute*

- Used to load in files
- Can apply to files with data, commands, or both
- Any files must be nexus formatted
- *Style: I often separate data and commands into separate files to facilitate multiple analyses*
- *Tip: Check line breaks of input files*

**Try the execute command with primates.nex…**

**MrBayes > execute primates.nex**

# MrBayes – Data

*include* and *exclude*

- Can decide which sites to use in an analysis after the data are loaded

**Try excluding sites 90-99….**

**Check their status using “charstat”…**

**Try re-including them…**

**Check status again…**

**Can exclude gaps and missing data as “missambig”…**

# MrBayes – Models

*lset*

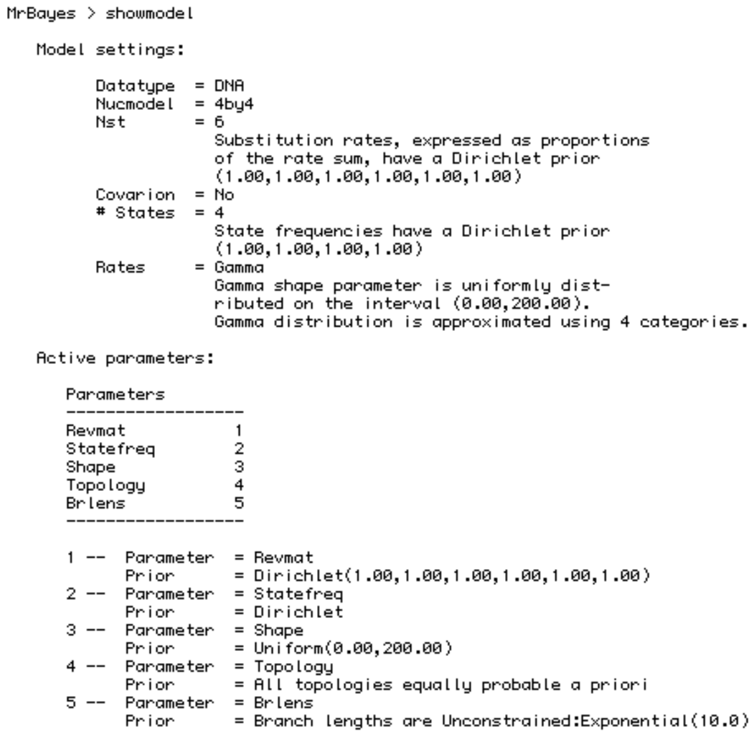
- Defines the form of the model and # of parameters
- nst (1,2,6) - # of substitution types
- rates (equal, G, I, G+I, or autocorrelated)
- other options available for doublet/codon models

**Try setting a GTR (nst=6) + G (rates=gamma) model…**

# MrBayes – Reviewing

*showmodel*

- Use this command to review the model and prior settings that you specified.



```
MrBayes > showmodel

   Model settings:

      Datatype  = DNA
      Nucmodel  = 4by4
      Nst       = 6
                  Substitution rates, expressed as proportions
                  of the rate sum, have a Dirichlet prior
                  (1.00,1.00,1.00,1.00,1.00,1.00)
      Covarion  = No
      # States  = 4
                  State frequencies have a Dirichlet prior
                  (1.00,1.00,1.00,1.00)
      Rates     = Gamma
                  Gamma shape parameter is uniformly dist-
                  ributed on the interval (0.00,200.00).
                  Gamma distribution is approximated using 4 categories.

   Active parameters:

      Parameters
      ------------------
      Revmat           1
      Statefreq        2
      Shape            3
      Topology         4
      Brlens           5
      ------------------

      1 --  Parameter  = Revmat
            Prior      = Dirichlet(1.00,1.00,1.00,1.00,1.00,1.00)
      2 --  Parameter  = Statefreq
            Prior      = Dirichlet
      3 --  Parameter  = Shape
            Prior      = Uniform(0.00,200.00)
      4 --  Parameter  = Topology
            Prior      = All topologies equally probable a priori
      5 --  Parameter  = Brlens
            Prior      = Branch lengths are Unconstrained:Exponential(10.0)
```

# MrBayes – Priors

*prset*

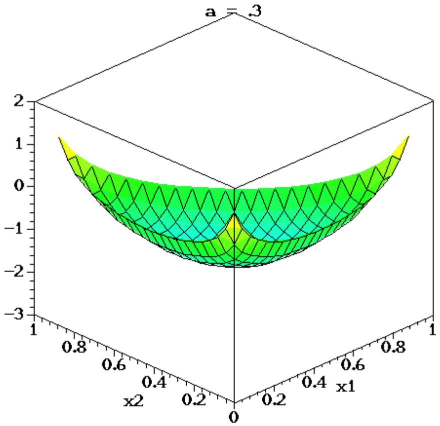
- Sets prior probabilities for all parameters (e.g., partition scaling, base frequencies, branch lengths, …)
- By default, models include variable base frequencies

**Try fixing base frequencies to be equal - statefreqpr=fixed(equal) …**

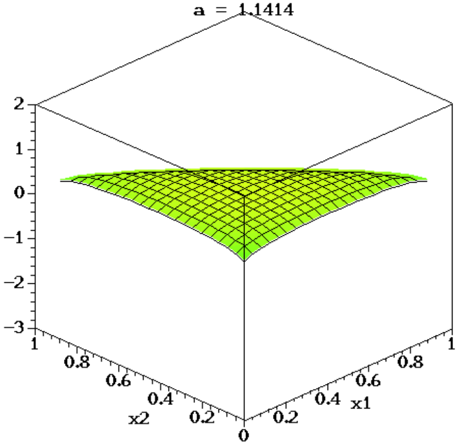
**Try fixing base frequencies at some arbitrary values…**

# Dirichlet Distribution

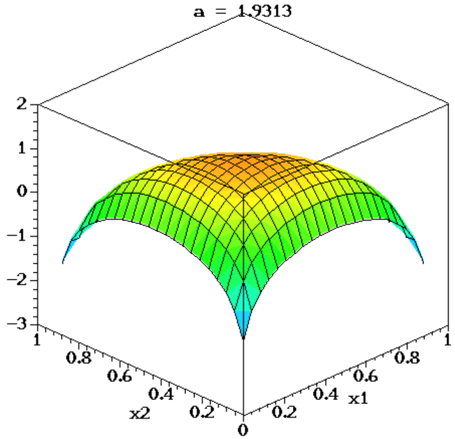
- Used to specify prior for a set of frequencies that have a fixed sum (e.g., relative rates, base freqs)



α = (0.3, 0.3, 0.3)



α = (1.14, 1.14, 1.14)



α = (1.93, 1.93, 1.93)

# MrBayes – Priors

*prset*

- Sets prior probabilities for all parameters

  (e.g., relative rates, base frequencies,
  relative rates, branch lengths, …)

- By default, models include variable base frequencies

**Try specifying a more informative prior (not fixed)…**

# MrBayes – Priors

*prset*

- Branch length priors can be important!
  - ➔ Define a prior for *every* branch (lots of params)
  - ➔ Can strongly affect posterior probabilities
  - ➔ Can strongly affect posterior branch lengths
- Usually an exponential prior for branch lengths. Value specified in prset is the *rate* of the exponential (1/mean).

**Try specifying a prior with a smaller mean (bigger rate) branch length…**

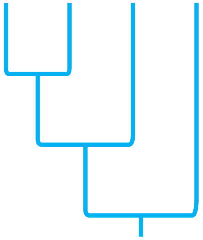
**Try specifying a prior with a larger mean (smaller rate) branch length…**

# MrBayes – MCMC

*mcmcp* and *mcmc*

- *mcmcp* sets parameters of analysis without starting the analysis
- *mcmc* sets parameters and **starts analysis**
- Some important options:
  - (i) ngen = # of “generations” before pausing
  - (ii) nruns = # of independent runs (important!)
  - (iii) nchains and temp -> metropolis-coupling (next)
  - (iv) samplefreq and printfreq -> frequency of printing to file and to screen

# MrBayes – Let ‘er rip



Try setting:

ngen = 10000

samplefreq = 200

printfreq = 200

nruns = 4

filename=primates1

**…type “mcmc” to start and watch the magic!**

# MrBayes – Output Files

Parameter (.p) files

- Contains likelihoods and parameter estimates for each generation
- Can plot "traces" of these values through time

Tree (.t) files

- Contains tree (with branch lengths) for each generation
- Use collection of trees to calculate consensus tree (or other topological summary statistics)

**Take a look in TextWrangler (or equivalent)…**

# MrBayes – Summarizing

*sump*

- Calculates summary statistics for scalar values

*sumt*

- Calculates summary information for trees (including a majority-rule consensus tree and the MPP tree)

Tracer

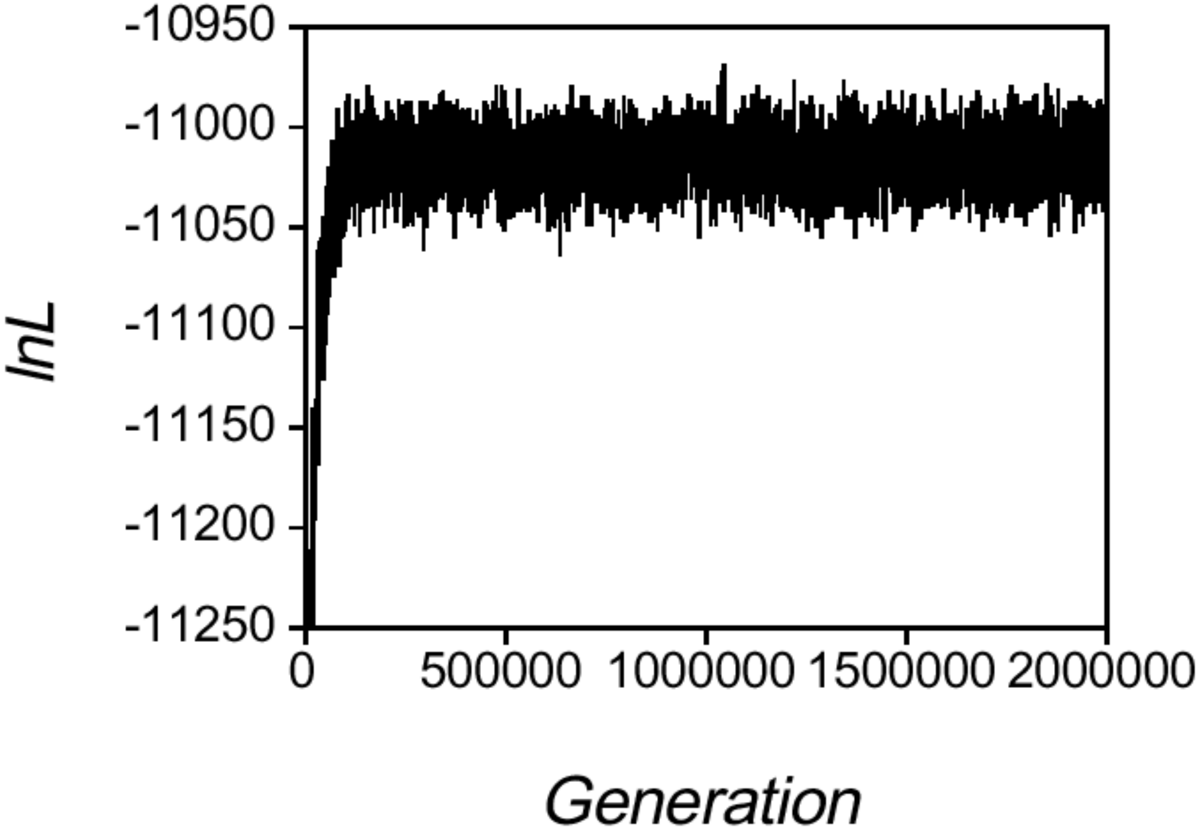
- http://tree.bio.ed.ac.uk/software/tracer/

AWTY

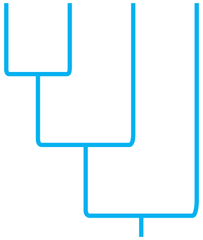
- http://ceb.csit.fsu.edu/awty/

# MrBayes – Summarizing

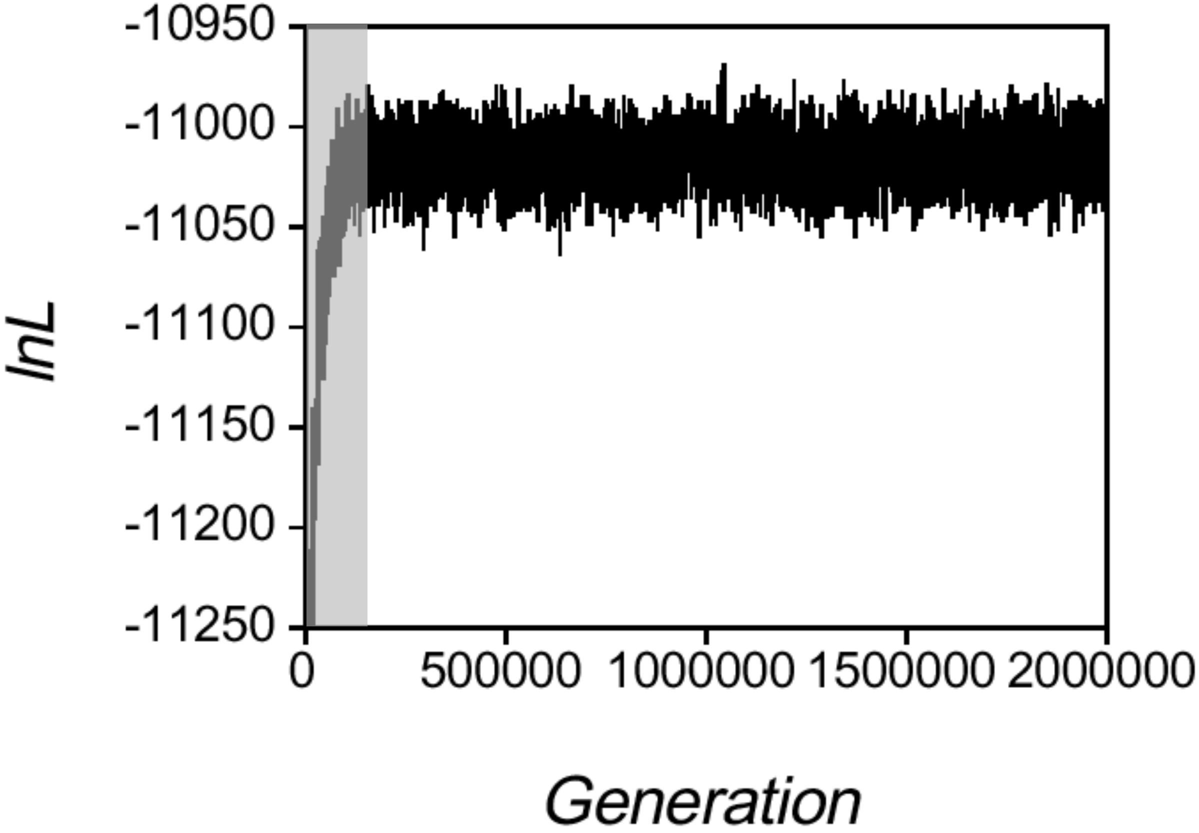
burnin



# MrBayes – Summarizing



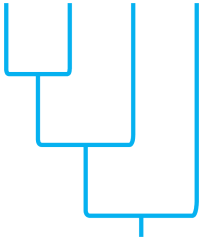
burnin



# MrBayes – Convergence

- Average Standard Deviation of Split Frequencies
  - Calculated from independent estimates across runs
- ESS (rough)
  - ~ equivalent number of independent samples
- PSRF (rough)
  - Compares between and within run variances
- Compare trees across runs (*post hoc*)
  - Simply compare topological estimates across runs

# MrBayes – MC$^3$



- Metropolis-Coupling – mcmcp (mcmc)
  - nchains – total number of chains
  - temp – degree of incremental heating

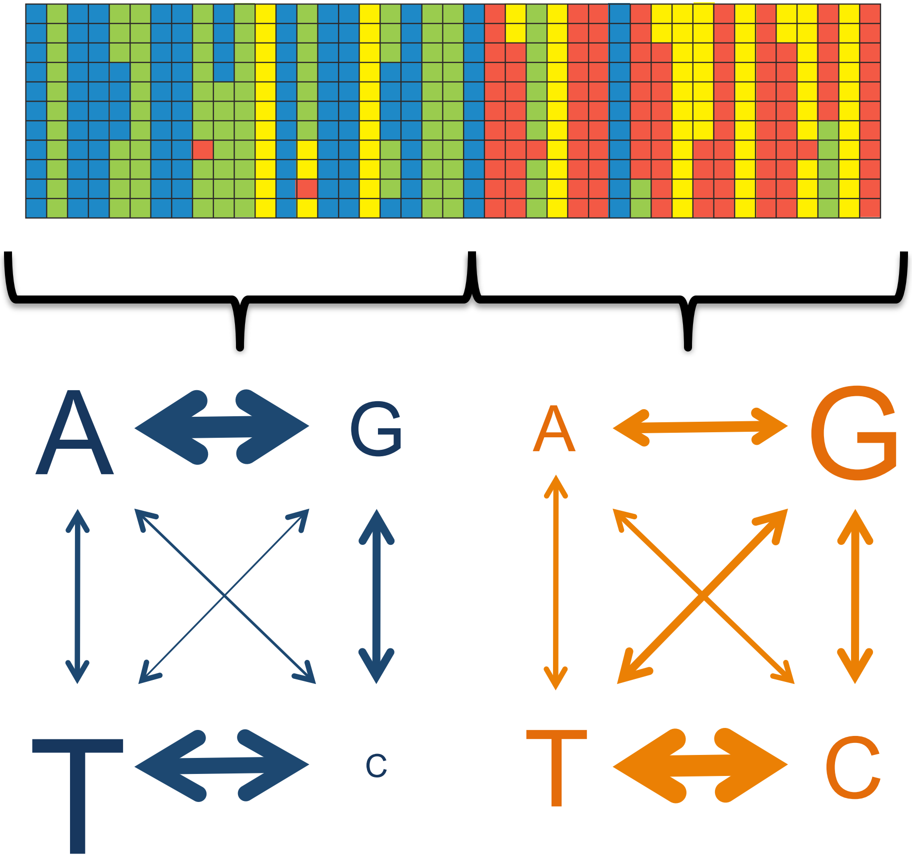
    (e.g., if temp = 0.2, heating is 0 (cold chain), 0.2, 0.4, 0.6, …)

**Try increasing the number of chains from 4 to 8…**

**Run analyses for 10K generations and compare run times…**

**How does screen output change?**

# MrBayes – Partitioning



# MrBayes – Partitioning

- Defining character sets
  - Charset *name* = *characters*
  - \3 denotes every 3rd position (for codon structure)
- Defining (and setting!) a partitioning scheme
  - partition cod.pos = 3: cod.pos.1, cod.pos.2, cod.pos.3;
  - set partition = cod.pos;
- Linking/Unlinking parameters across partitions
  - lset applyto=(all) nst=6 rates=gamma;
  - unlink revmat=(all) statefreq=(all) shape=(all);
- Variable Rates (scales branch lengths)
  - prset ratepr=variable;

# MrBayes – Partitioning

- Try:
  - Defining character sets by thirds of the data
  - Defining character sets by codon position
  - Unlinking ONLY base frequencies across partitions
  - Allowing variable rates of evolution across partitions
  - Assigning HKY (nst=2) to two partitions and GTR (nst=6) to one partition…

# MrBayes – Command File

```
#NEXUS

Begin MrBayes;
    execute primates.nex;
    lset nst=6 rates=gamma;
    prset brlenspr=Uconstrained:Exp(1);
    mcmc ngen=10000;
    sump;
    sumt;
End;
```

**Try creating a command file and executing it…**

# MrBayes – Command Line

- If compiled locally, you can run MrBayes from the command line (e.g., Terminal for Mac OS X)

```
> mb mb_cmds.nex
```

- Facilitates batch analyses, especially on a computing cluster

# MrBayes with Beagle

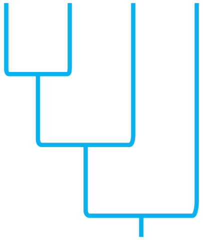
If you have an NVIDIA graphics card, you can take advantage of GPU-based parallelization via the Beagle library.  Especially helpful for codon and amino acid models.

How To:

- Check your graphics card (on Mac: "About This Mac")
- Download and install CUDA drivers from NVIDIA

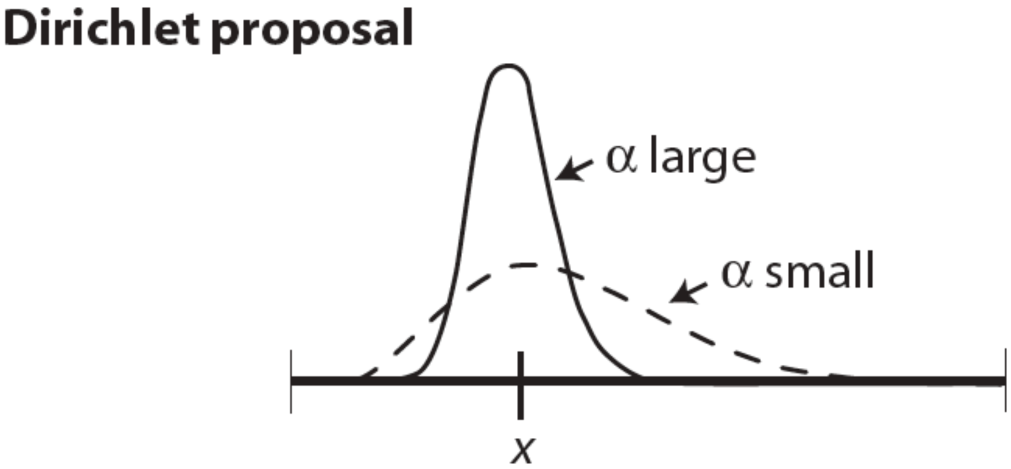
  (http://www.nvidia.com/Download/index.aspx?lang=en-us)
- Check that MrBayes sees it (*showbeagle*)
- Turn it on (set beagledevice=gpu usebeagle=yes)

# MrBayes – Proposals



- Proposal Distributions and Frequency
  - Vary by type of parameter
  - In theory, don’t affect estimated values, only the time it takes to get the estimates
  - In practice, can effectively kill the ability of the MCMC to find the “right” part of parameter space before your career ends
  - Set using the “props” command
    - *Currently this can only be done interactively*

# MrBayes – Proposals



New values are picked from a Dirichlet (or Beta) distribution centered on *x*.

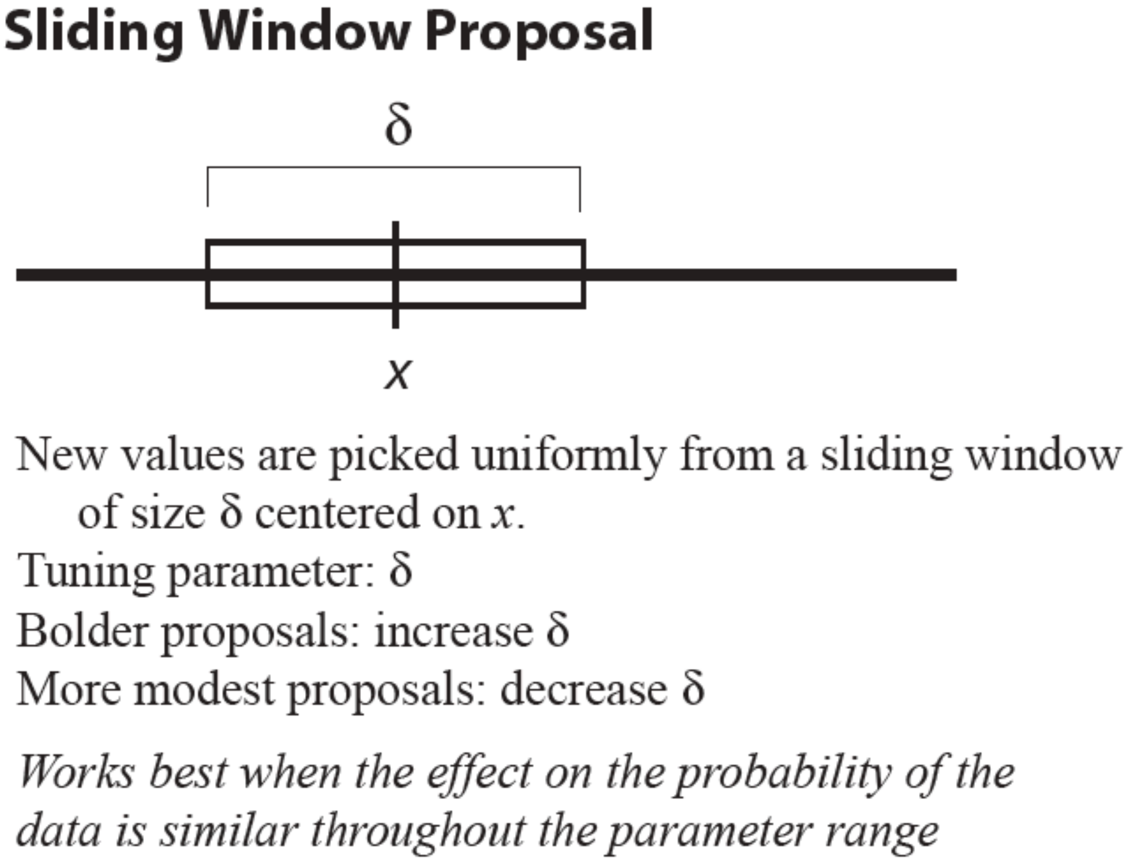
Tuning parameter: α

Bolder proposals: decrease α

More modest proposals: increase α

*Works well for proportions, such as revmat and statefreqs.*

# MrBayes – Proposals



## Sliding Window Proposal

New values are picked uniformly from a sliding window of size δ centered on *x*.

Tuning parameter: δ

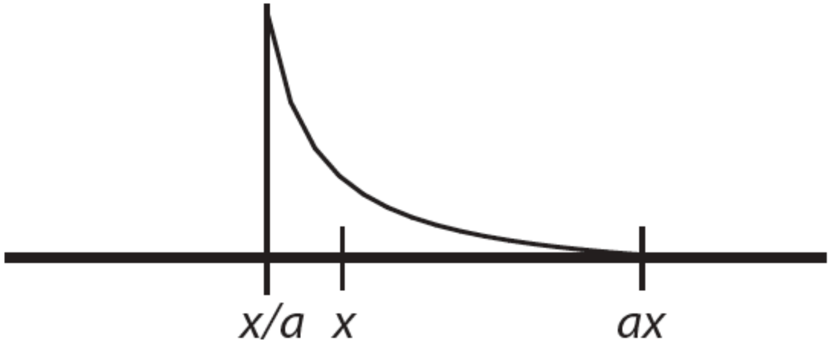
Bolder proposals: increase δ

More modest proposals: decrease δ

*Works best when the effect on the probability of the data is similar throughout the parameter range*

# MrBayes – Proposals

## Multiplier Proposal

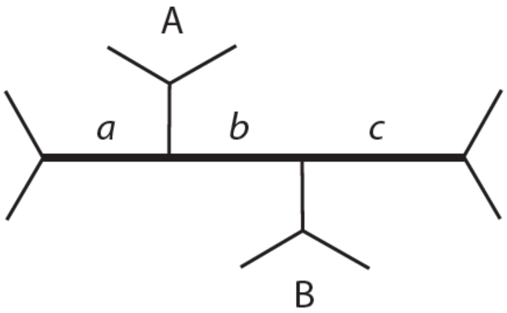


New values are picked from the equivalent of a sliding window on the log-transformed *x* axis.
Tuning parameter: λ = 2 ln *a*
Bolder proposals: increase λ
More modest proposals: decrease λ

*Works well when changes in small values of x have a larger effect on the probability of data than changes in large values of x. Example: branch lengths.*

# MrBayes – Proposals

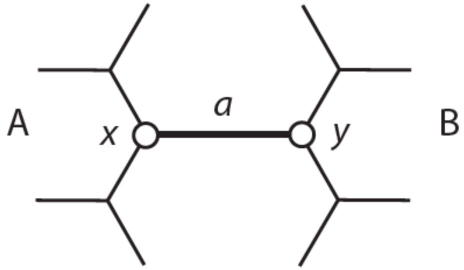
## LOCAL



Three internal branches - *a, b,* and *c* - are chosen at random.
Their total length is changed using a multiplier with tuning paremeter λ.
One of the subtrees A or B is picked at random.
It is randomly reinserted on $a + b + c$ according to a uniform distribution

Bolder proposals: increase λ
More modest proposals: decrease λ
Changing λ has little effect on the boldness of the proposal

## Extending TBR



An internal branch *a* is chosen at random
The length of *a* is changed using a multiplier with tuning paremeter λ
The node *x* is moved, with one of the adjacent branches, in subtree A, one node at a time, each time the probability of moving one more branch is *p* (the extension probability).
The node *y* is moved similarly in subtree B.

Bolder proposals: increase *p*
More modest proposals: decrease *p*
Changing λ has little effect on the boldness of the proposal.

# MrBayes – Proposals

Try this:

(1) Pick your favorite variable

(2) Turn off (almost) its proposals (set proposal rate close to 0)

(3) Start new (short) MCMC

(4) Either in Tracer or by looking directly at the parameter and likelihood values in the .p file, compare the rate of change for the parameter value and likelihood, especially early in the analysis.

# Bayesian Model Selection and Adequacy

# Don’t Choose!

Let the analysis sample different models for you (called reversible jump – RJ)

To sample sub-models of GTR:

MrBayes > lset nst=mixed

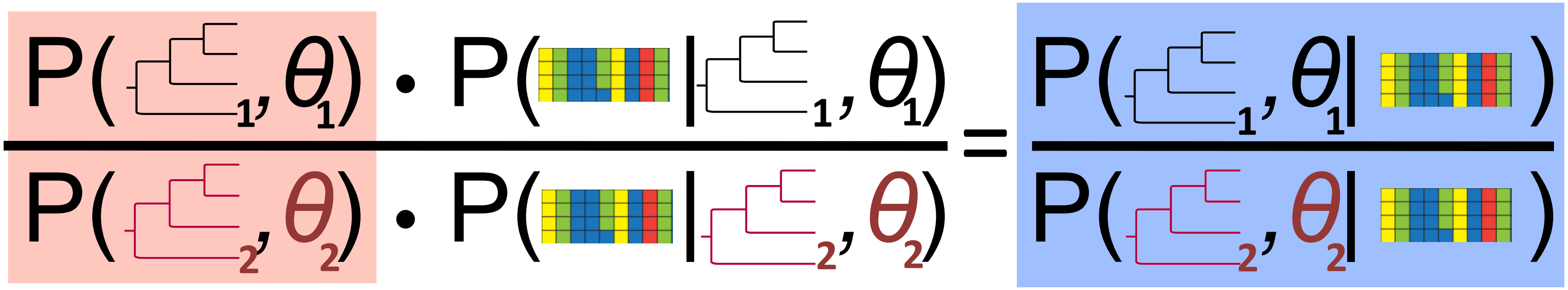
```
                          Posterior      Standard        Min.          Max.
Model                    Probability     Deviation     Probability   Probability
--------------------------------------------------------------------------------
gtrsubmodel[121123]         0.195          0.033          0.172         0.219
gtrsubmodel[121324]         0.089          0.005          0.086         0.093
gtrsubmodel[121323]         0.079          0.009          0.073         0.086
gtrsubmodel[121321]         0.060          0.037          0.033         0.086
gtrsubmodel[123324]         0.060          0.000          0.060         0.060
gtrsubmodel[121121]         0.053          0.009          0.046         0.060
--------------------------------------------------------------------------------
```

# Posterior Odds Ratio

Prior Odds

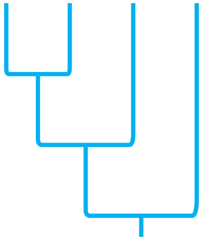
Bayes Factor

Posterior Odds



$$\frac{P(\ _1,\theta_1)\cdot P(\ |\ _1,\theta_1)}{P(\ _2,\theta_2)\cdot P(\ |\ _2,\theta_2)} = \frac{P(\ _1,\theta_1|\ )}{P(\ _2,\theta_2|\ )}$$

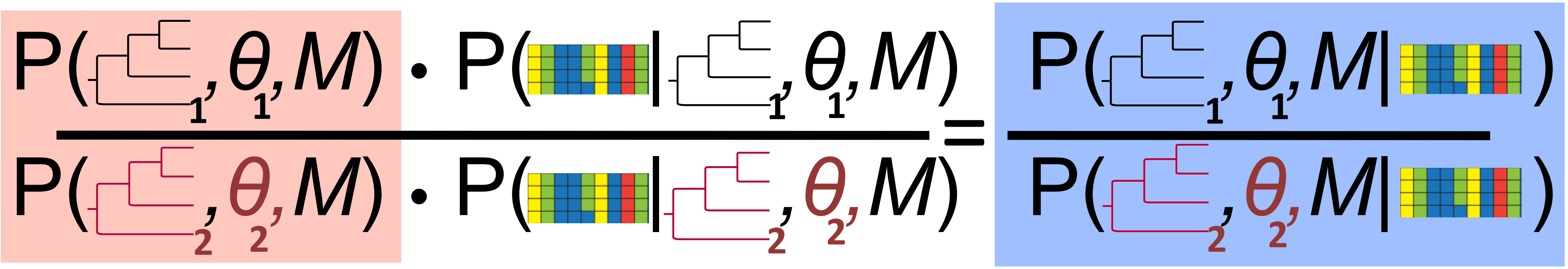
# Posterior Odds Ratio



Prior Odds

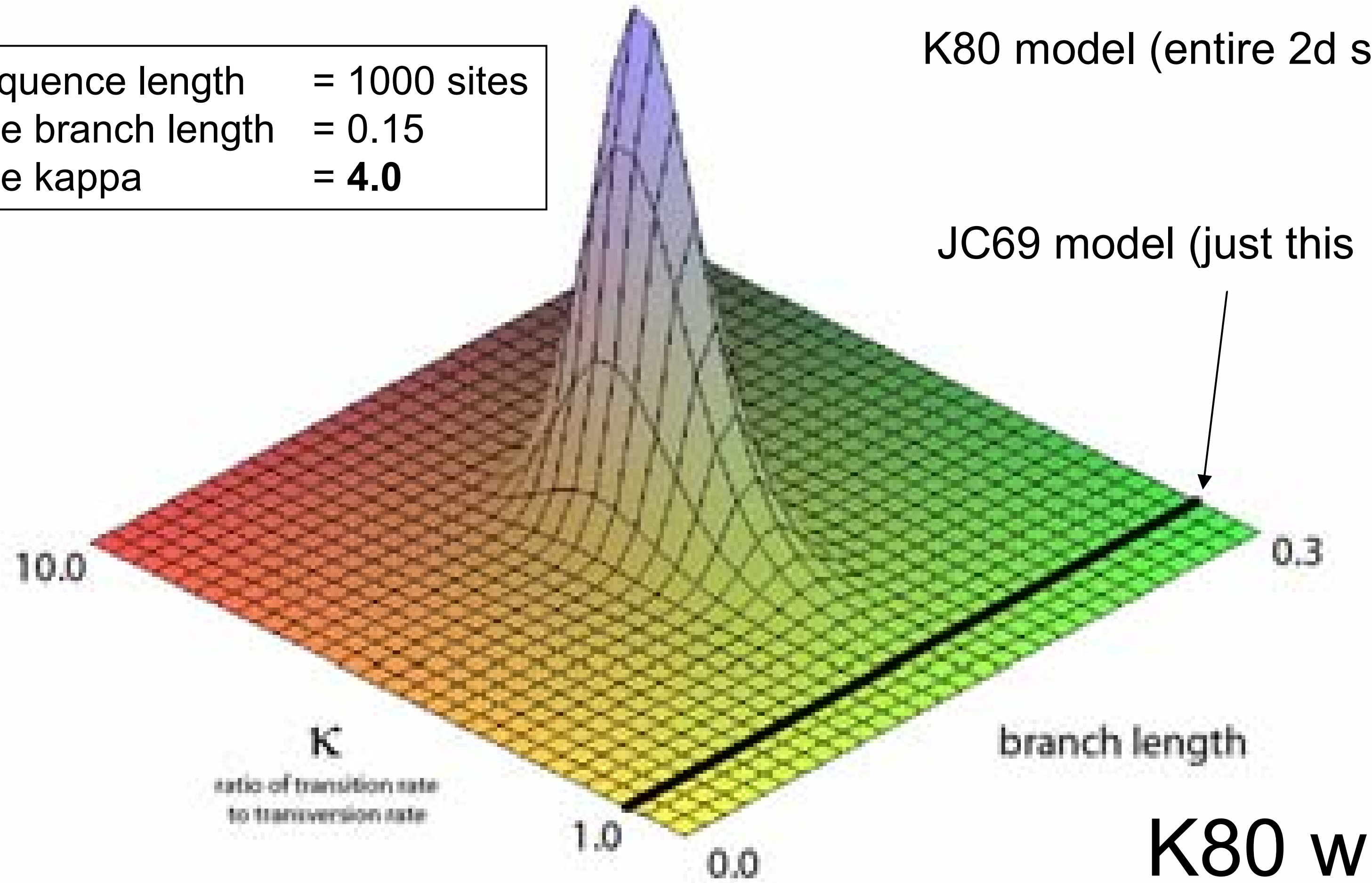
Bayes Factor

Posterior Odds



# Marginal Likelihood of a Model

sequence length = 1000 sites
true branch length = 0.15
true kappa = **4.0**



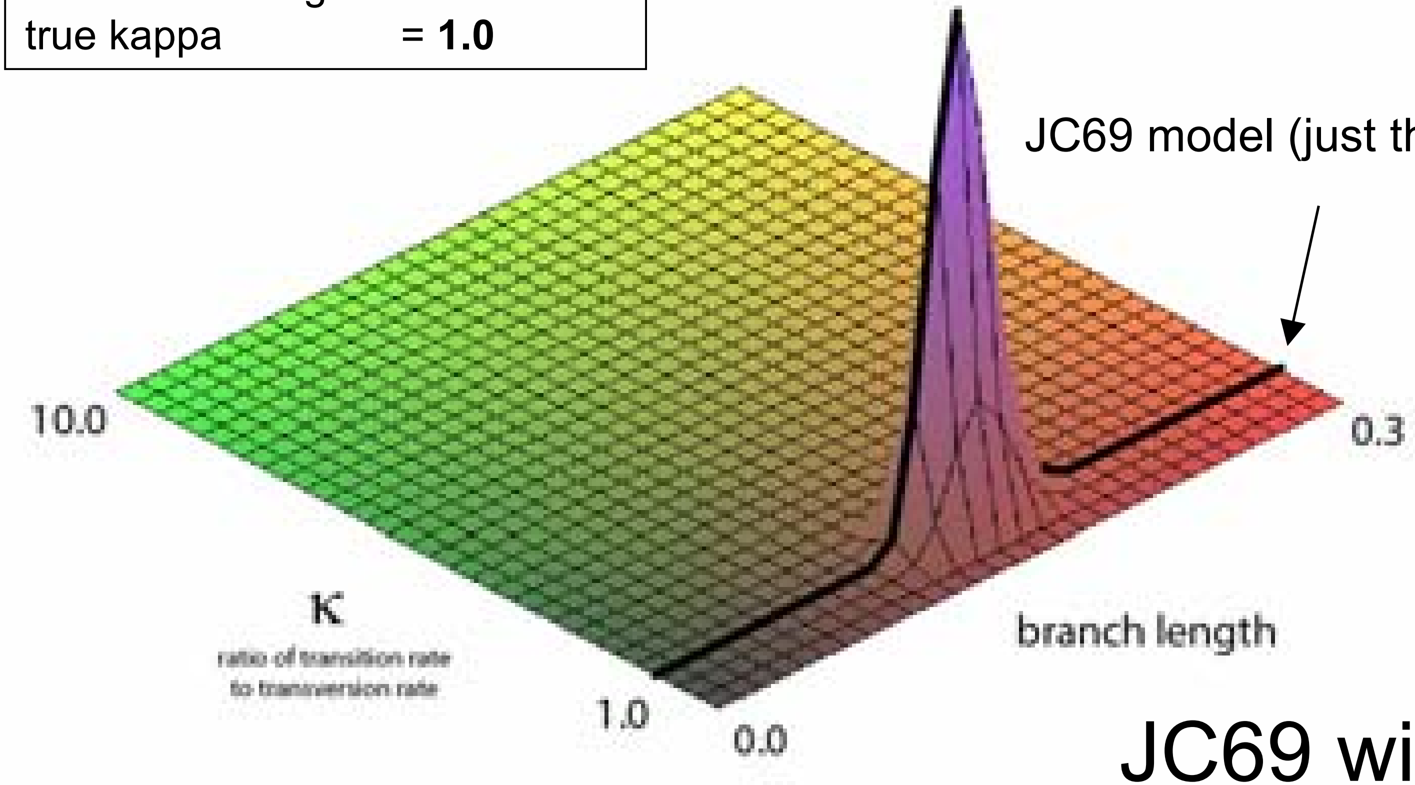
K80 model (entire 2d space)

JC69 model (just this 1d line)

K80 wins

# Marginal Likelihood of a Model

| | |
|---|---|
| sequence length | = 1000 sites |
| true branch length | = 0.15 |
| true kappa | = **1.0** |



K80 model (entire 2d space)

JC69 model (just this 1d line)

JC69 wins

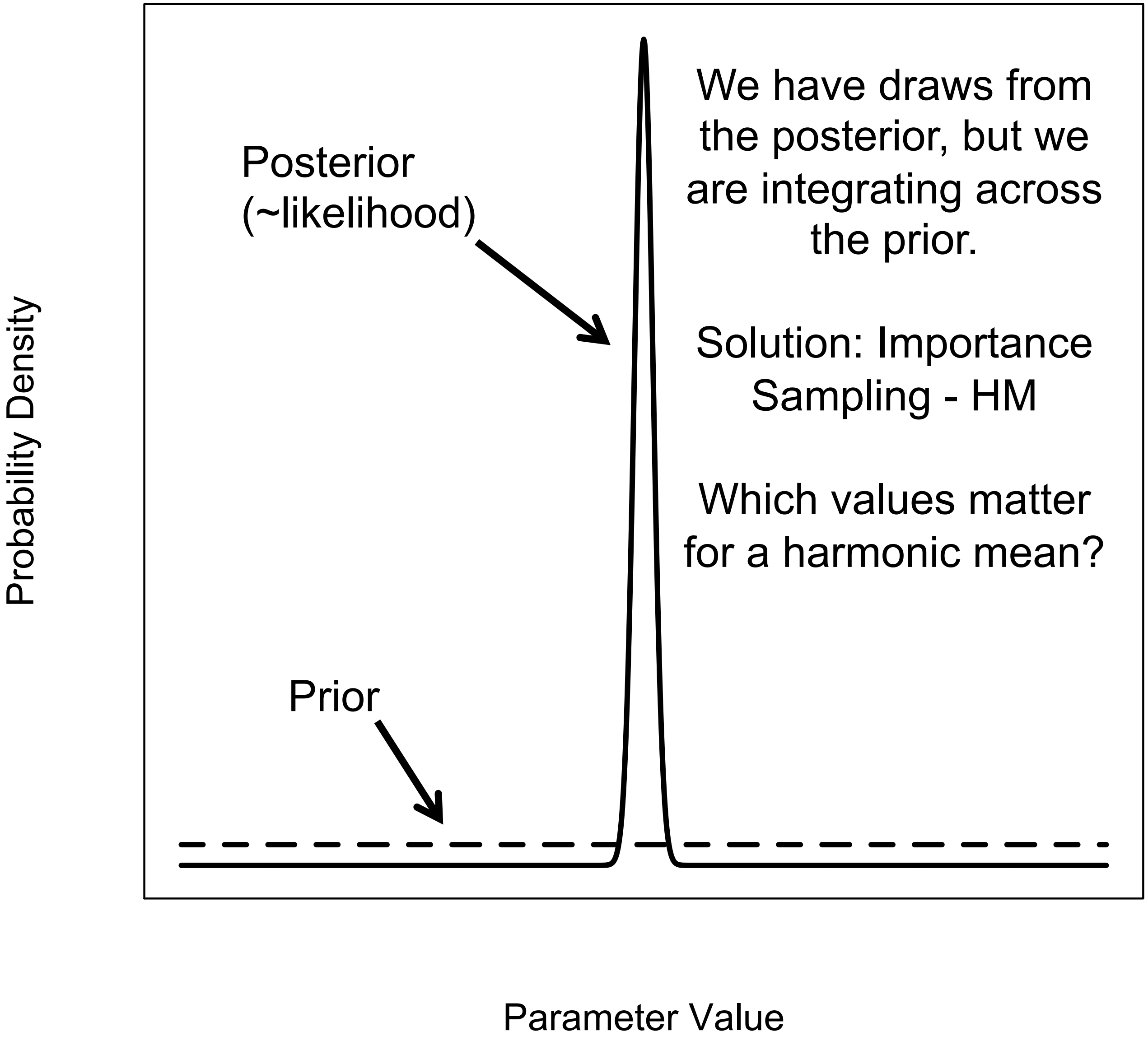
# Estimating Bayes Factors

- If using RJ (sampling the posterior distribution of models), you can calculate the Bayes factor as the ratio of posterior and prior odds

- Can also calculate the marginal likelihoods from separate analyses employing different models

  - Harmonic mean (easy, but biased)
  - Stepping stone (more accurate, but harder)
  - Thermodynamic integration (similar to SS – not implemented in MrBayes)
  - MORE SOON!

# MrBayes – Stepping Stone

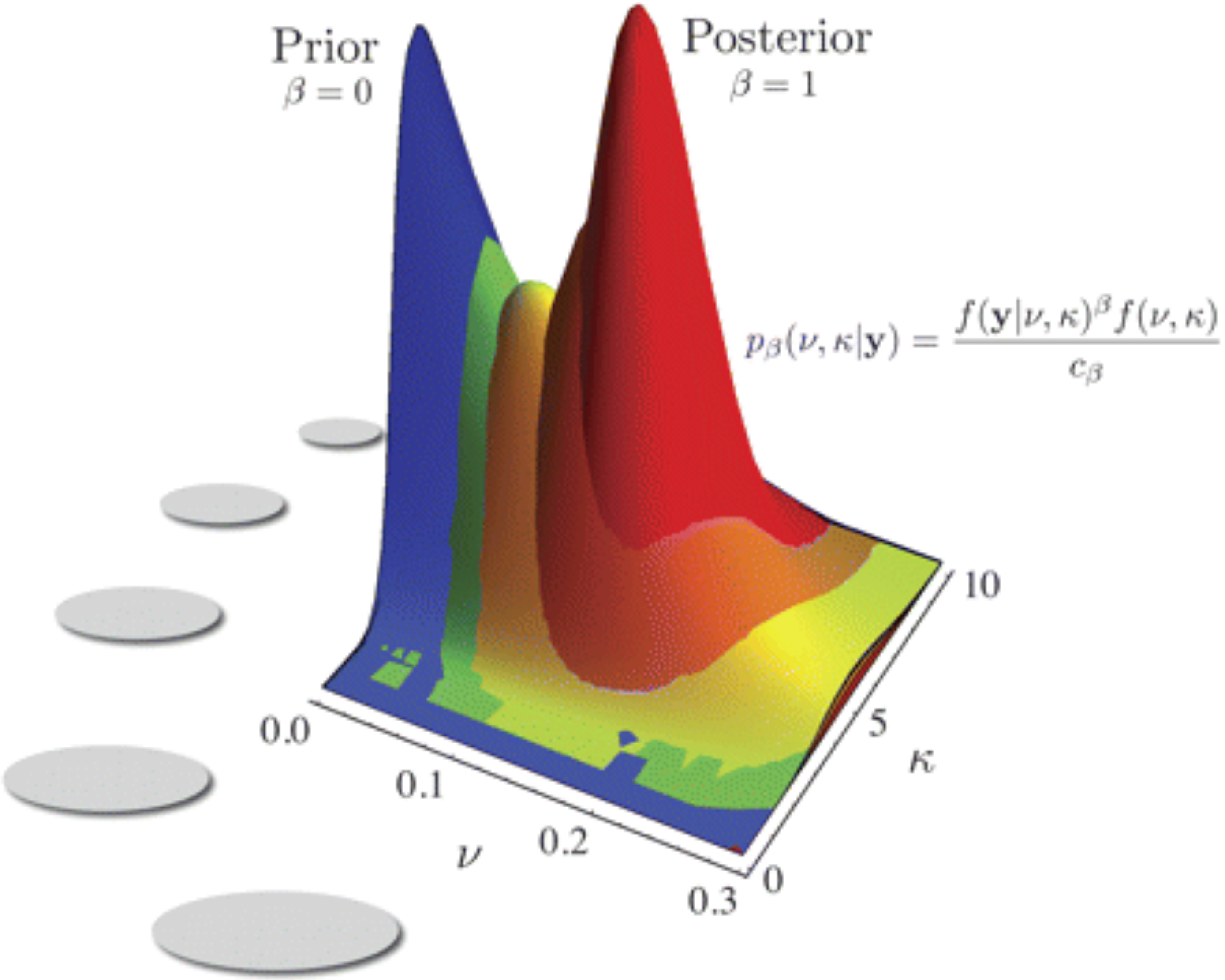


We have draws from the posterior, but we are integrating across the prior.

Solution: Importance Sampling - HM

Which values matter for a harmonic mean?

# MrBayes – Stepping Stone



# MrBayes – Stepping Stone

*ss* and *ssp*

*ssp* allows you to set the parameters of the stepping stone process without running. *ss* does the same, but starts the run.

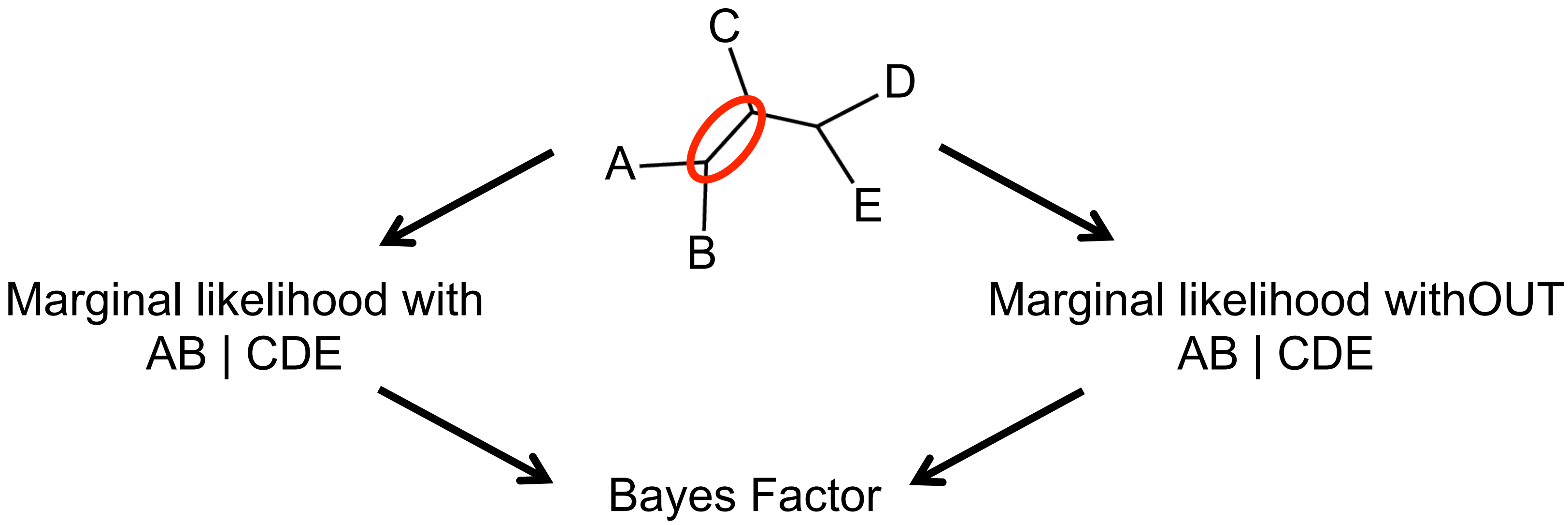
**Try the *ss* command with primates.nex…**

**MrBayes > ss**

# Topology BFs

Can also use stepping stone to calculate marginal likelihoods for sets of trees subject to particular constraints.

For instance, comparing the marginal likelihoods for a set of trees that all contain a branch to the set of trees that all do NOT contain that branch would give you a Bayes factor supporting that branch.



Marginal likelihood with AB | CDE

Marginal likelihood withOUT AB | CDE

Bayes Factor

# Model Choice v “Adequacy”

Which of the available models will perform best?

vs.

Is any given model sufficient to provide unbiased inferences?

# Model Choice v “Adequacy”

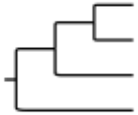
Or, better, **Plausibility**

Which of the available models will perform best?

vs.

Is any given model sufficient to provide unbiased inferences?

# Posterior Prediction

Could have come from $P($  $,\theta|$ $)$ ?

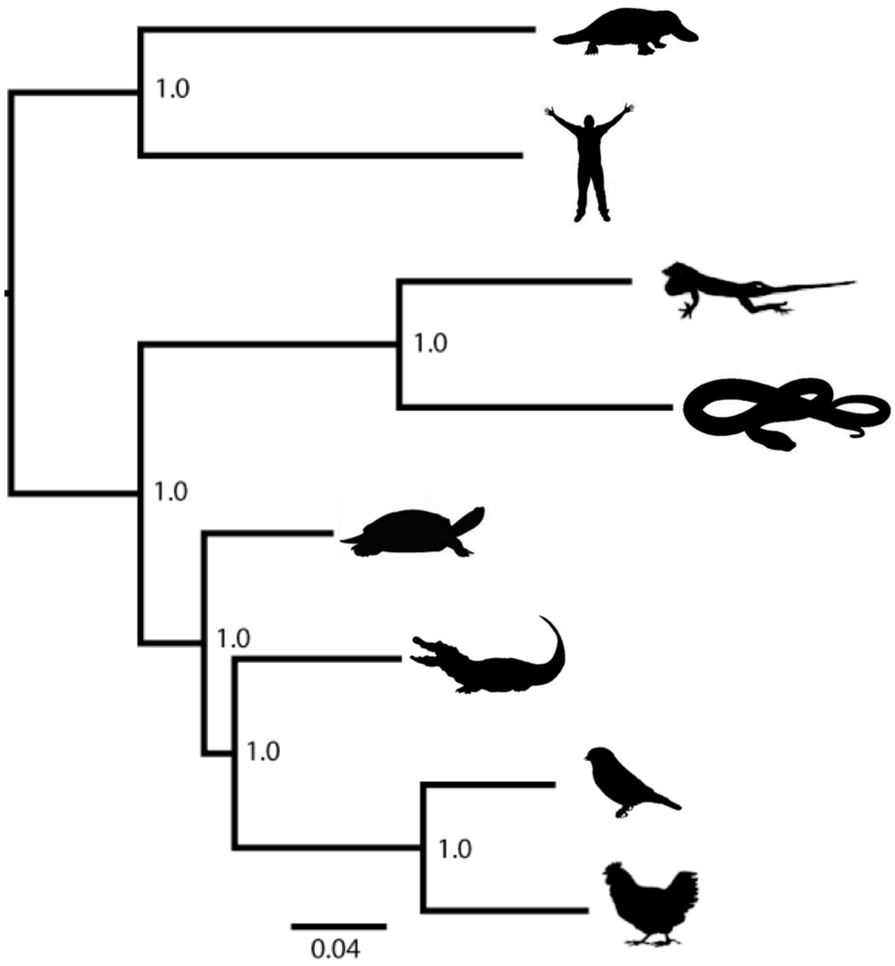
Could the model and priors plausibly have given rise to the data?

If not, phylogenies may be **biased**

# Big Data = Strong Support

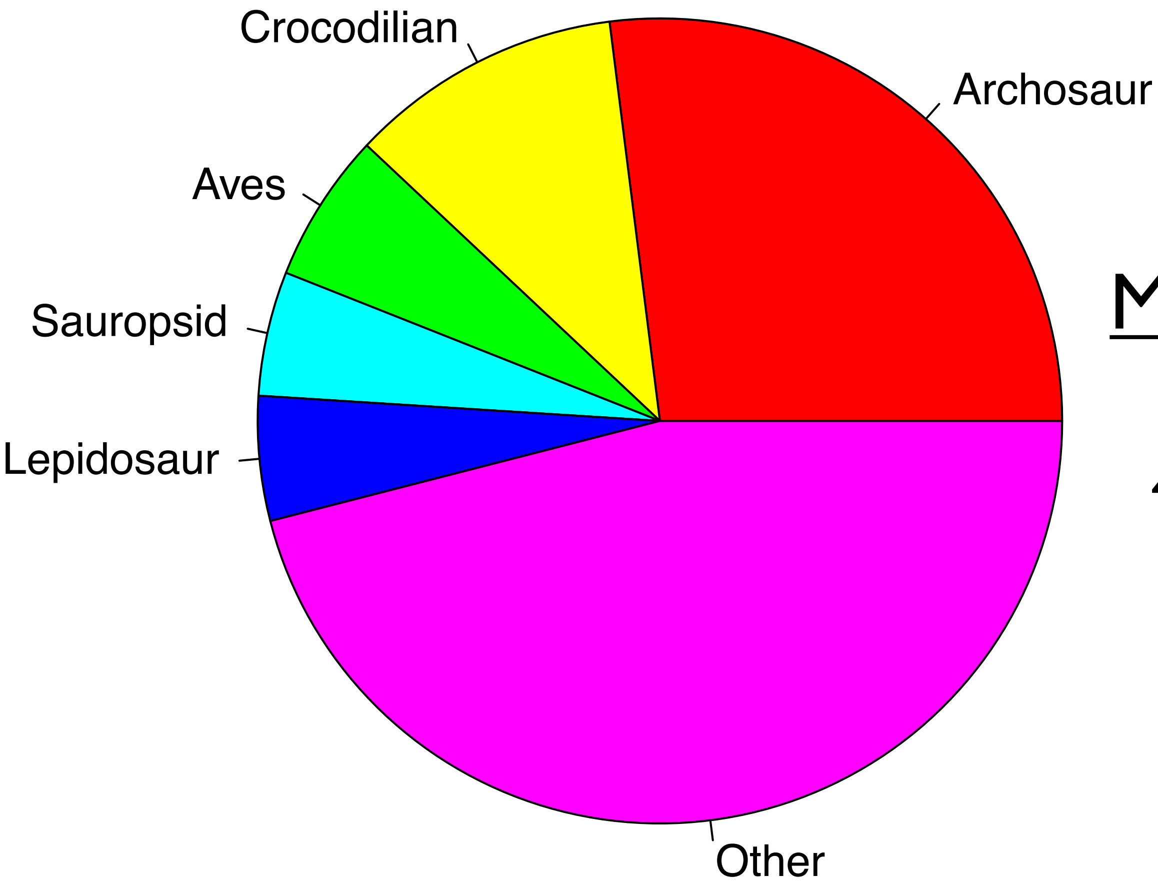
1,955 Genes

Bayesian Tree



From Bob Thomson (Painted Turtle Genome Project)

# Diff’t Genes - Diff’t Trees



<u>Many genes, few taxa</u>

1,144 Genes

4-8 Taxa Per Gene

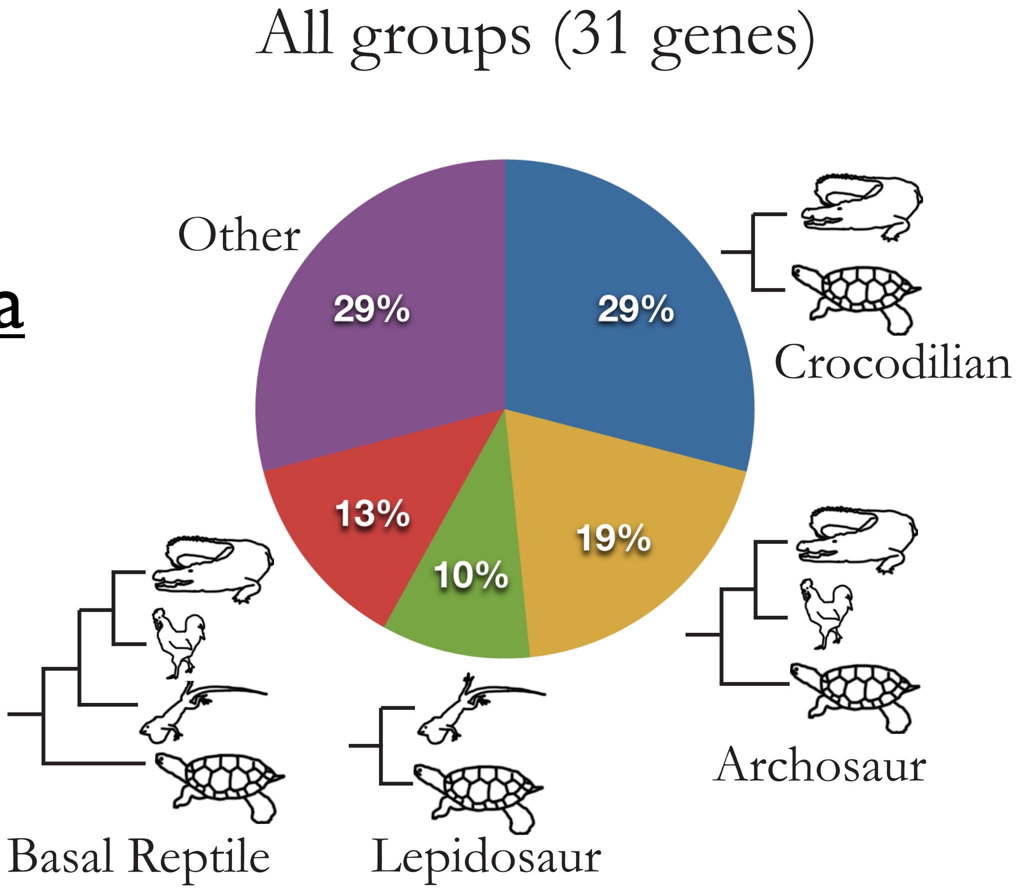
Alignments from Bob Thomson (Painted Turtle Genome Project)

# Diff’t Genes - Diff’t Trees

<u>Few genes, many taxa</u>

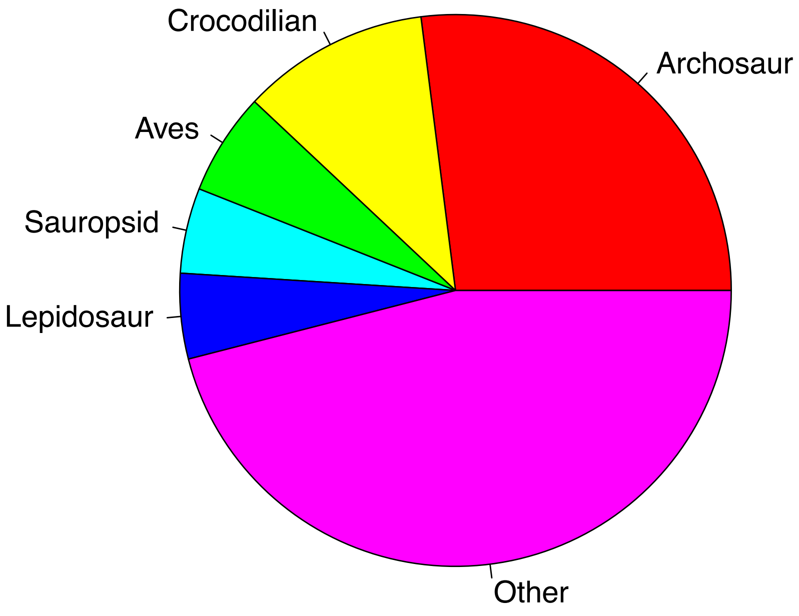
129 Taxa

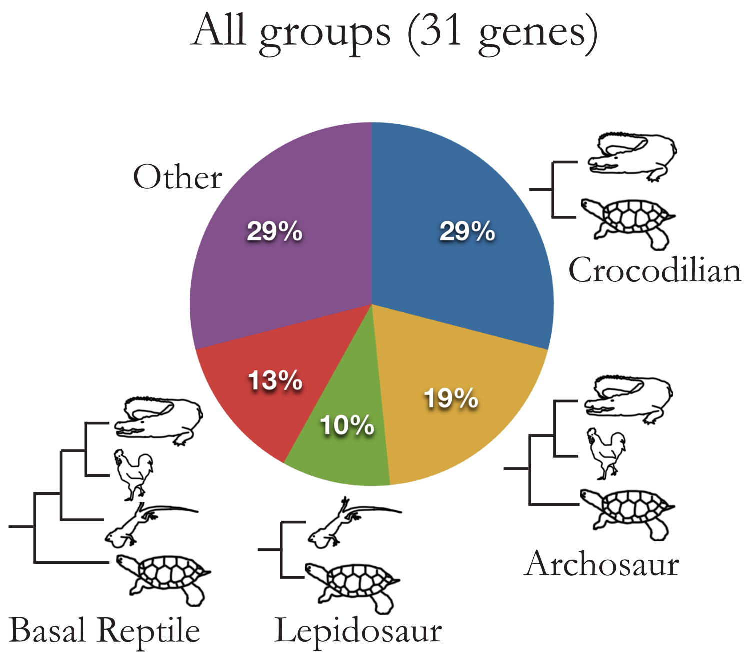
31 genes



Fong, Brown, Fujita, and Boussau. 2012. PLoS One. 7:e48990.

# Diff’t Genes - Diff’t Trees





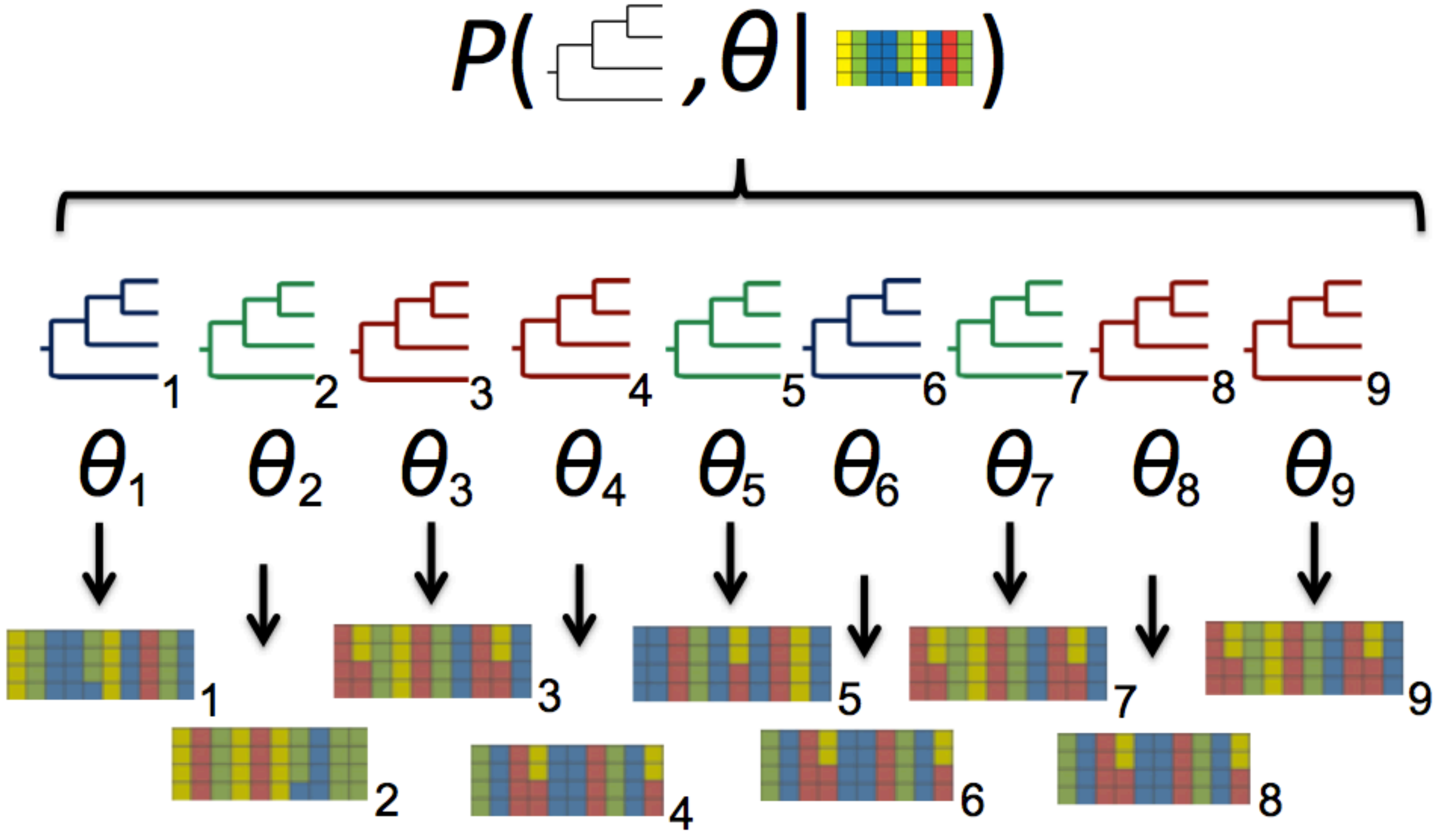
Biological or Methodological Variation?

# Posterior Prediction

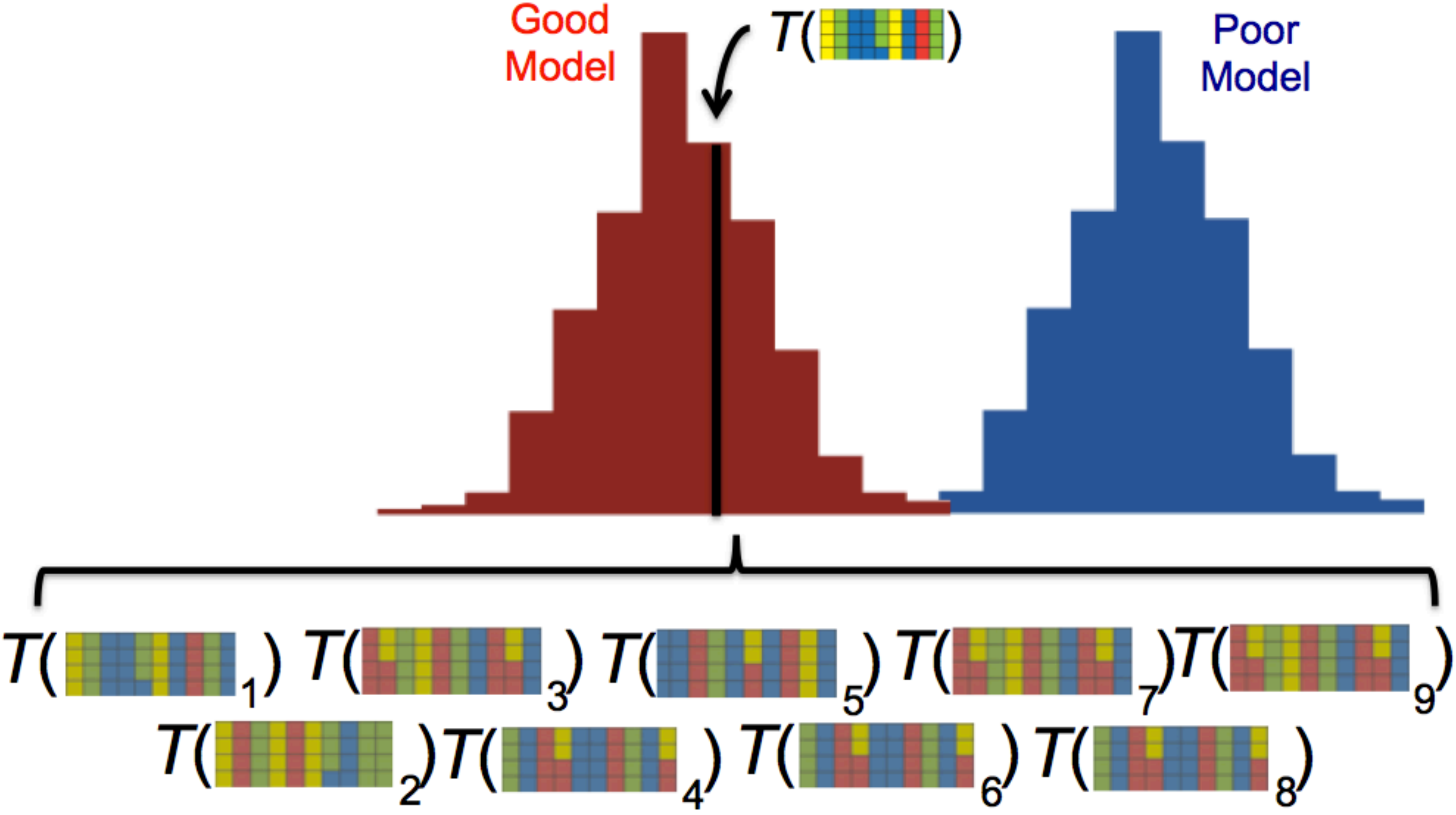
“We do not like to ask, ‘Is our model true or false?’, since probability models in most analyses will not be perfectly true…The more relevant question is, ‘Do the model’s deficiencies have a noticeable effect on the substantive inferences?’”

-A. Gelman, J.B. Carlin, H.S. Stern, and D.B. Rubin
Bayesian Data Analysis

# Posterior Prediction

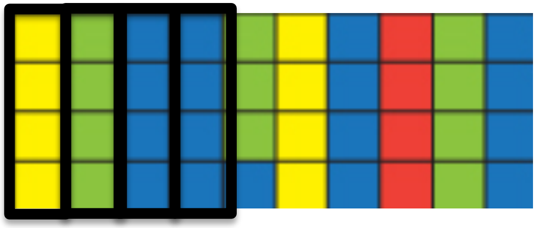


# Posterior Prediction



# Posterior Predictive Simulation

## Previously proposed statistic*: Multinomial Likelihood

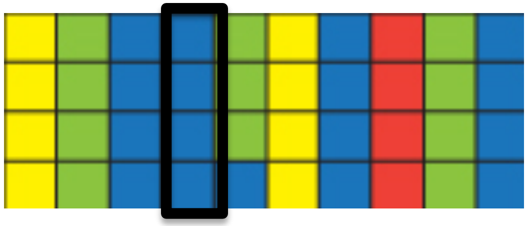


Based on the frequency of different site patterns

*Goldman, 1993; Bollback, 2002

# Posterior Predictive Simulation

## Previously proposed statistic*: Multinomial Likelihood



Based on the frequency of different site patterns

Tests if the assumed and generating models
produce data with similar site pattern frequencies

Intuitively appealing, but very sensitive to branch-length biases.
Can reject adequacy, **even when inferred phylogeny is correct**

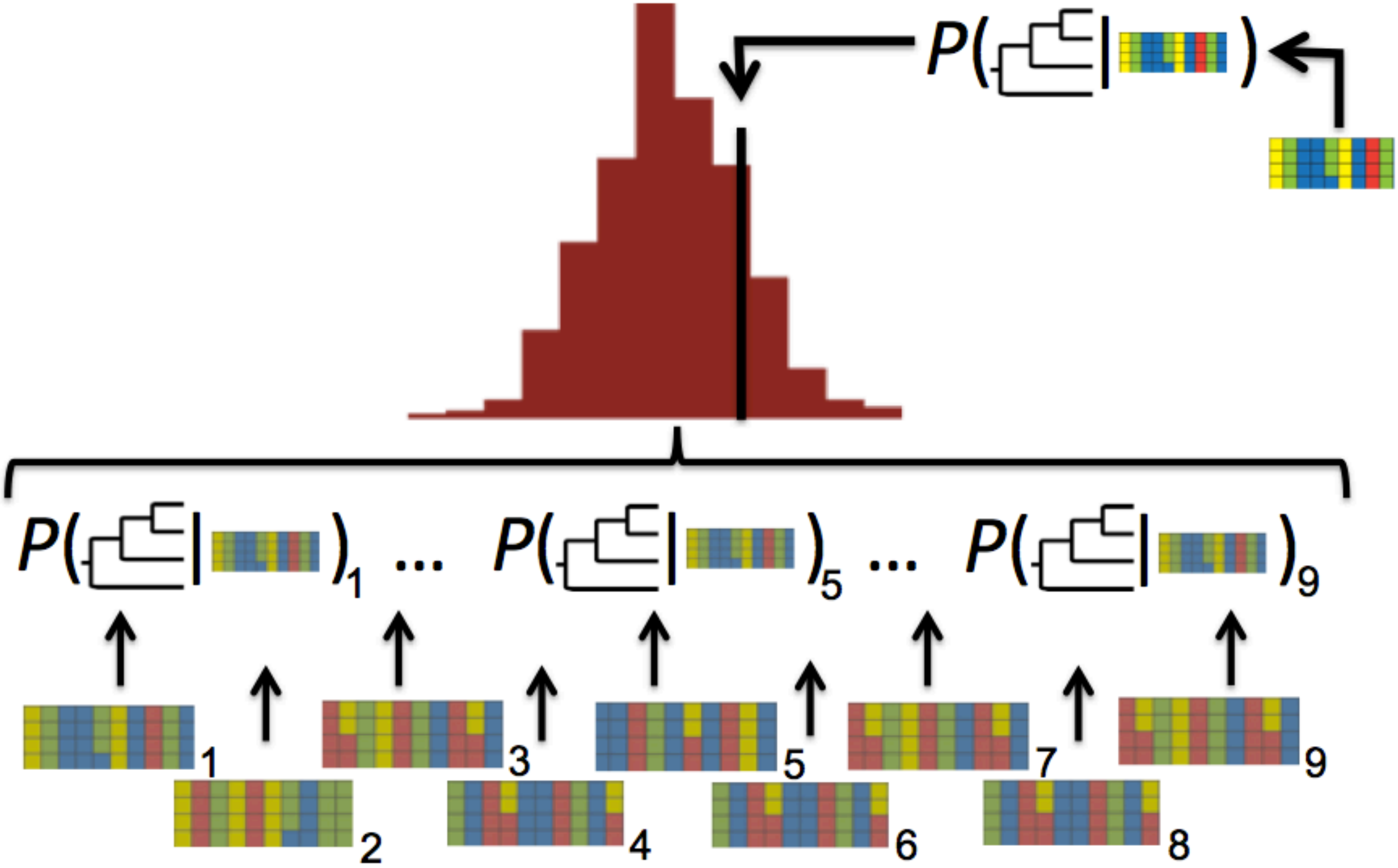
*Goldman, 1993; Bollback, 2002

# Posterior Predictive Simulation

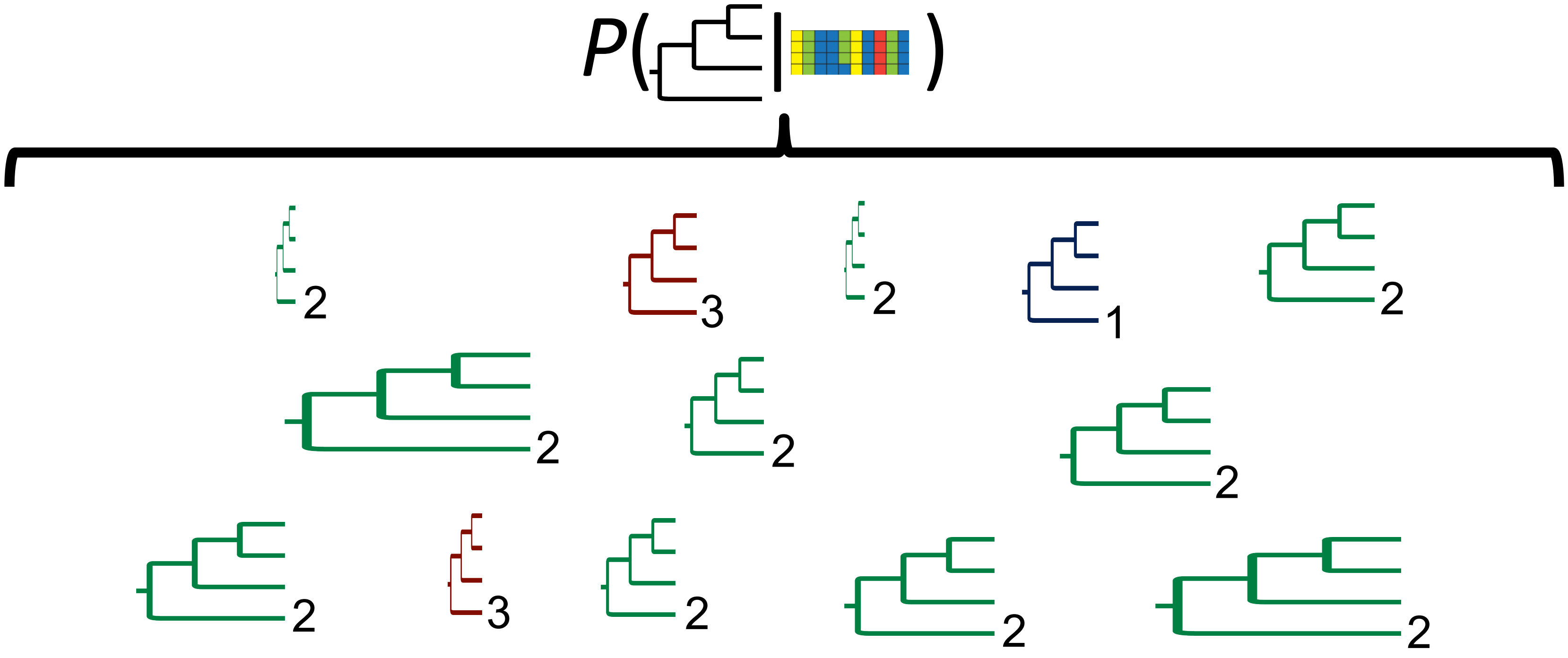
Previously proposed statistics:

- Multinomial Likelihood
- Number of Unique Site Patterns
- Frequency of Invariant Sites
- Heterogeneity of Base Frequencies
- Number of parsimony-inferred "parallel" sites

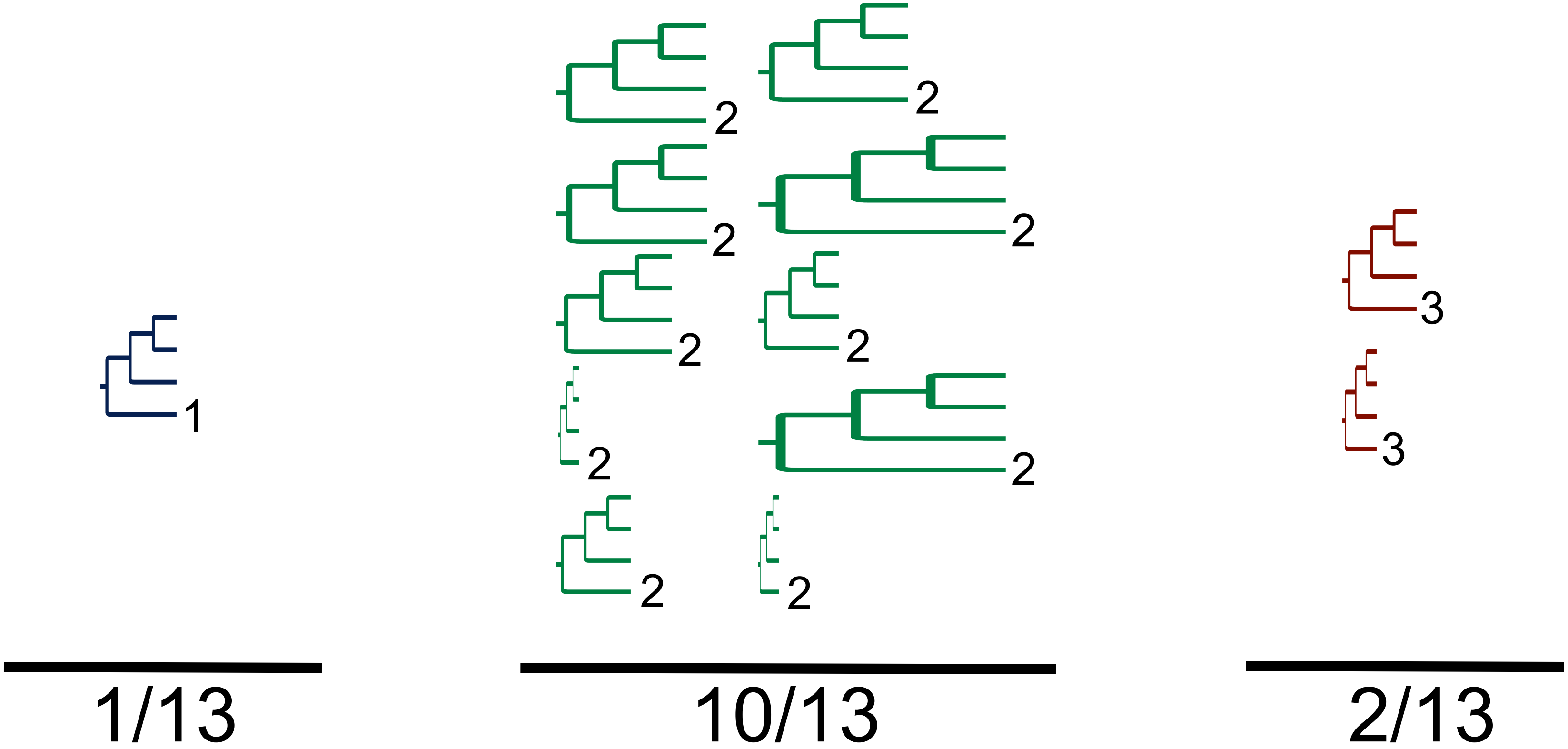
# Posterior Prediction



# Marginal Test Quantities

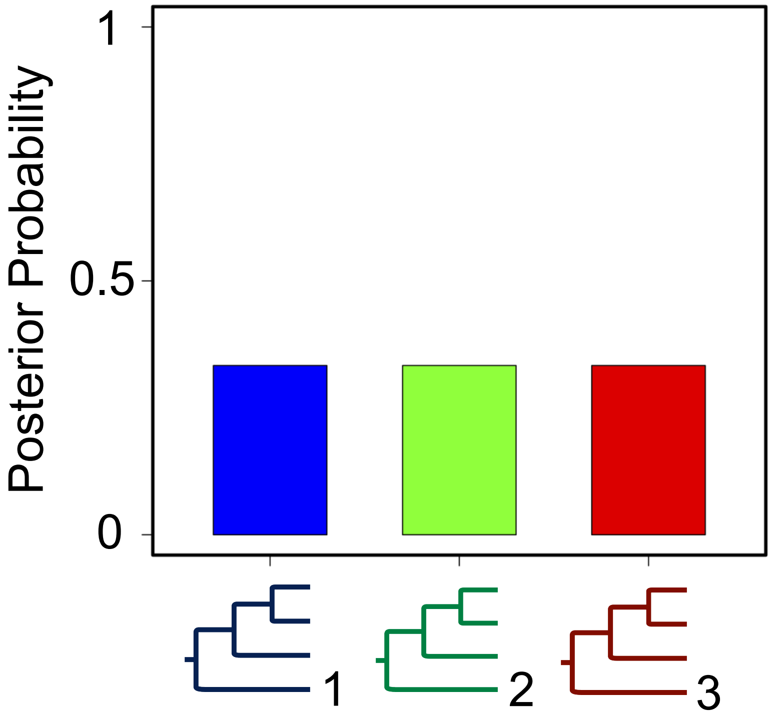


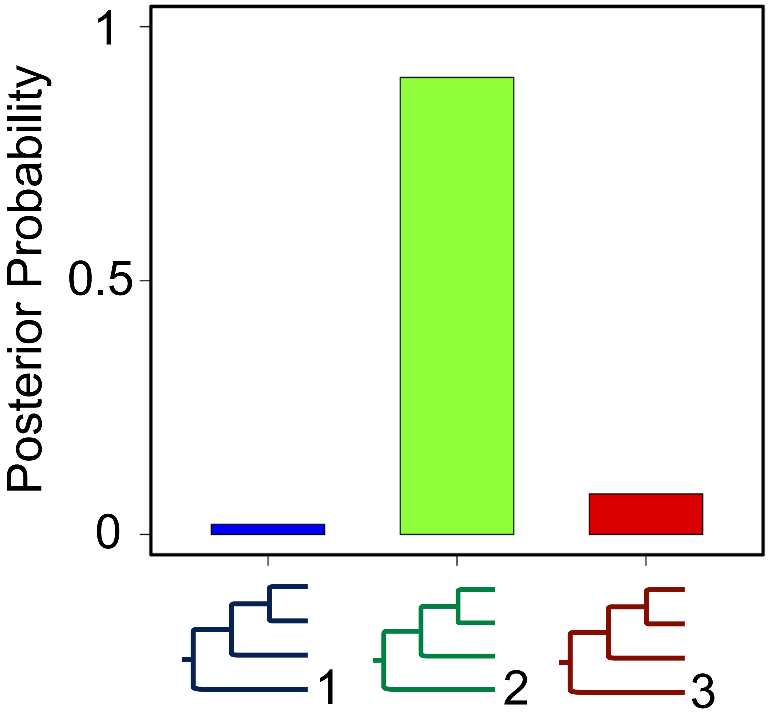
# Marginal Test Quantities - Topology



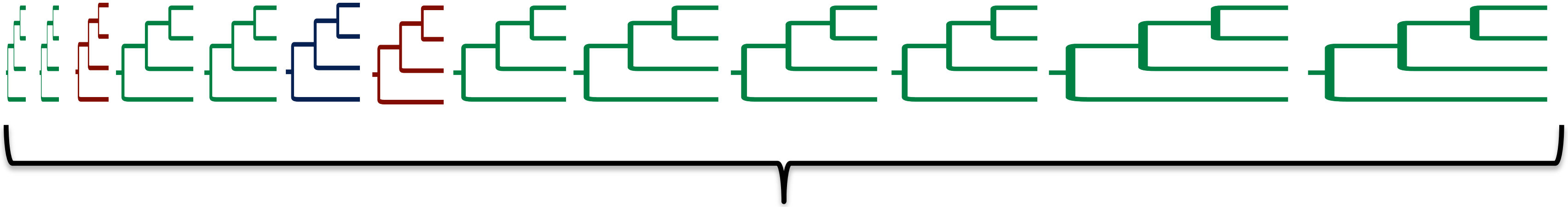
Integrating across variation in branch lengths (nuisance parameter)

# Entropy Quantifies Support Across Topologies


Higher Entropy


Lower Entropy

# Marginal Test Quantities – Tree Length



Mean Tree Length = 3.15

Integrating across topologies (nuisance parameter)

# Posterior Prediction

What kinds of inferences might we care about?

- Overall topological inference

- Branch-specific support (posteriors)

- Tree (or branch) length

- Support from individual sequence positions
  (Identify specific biased sites)

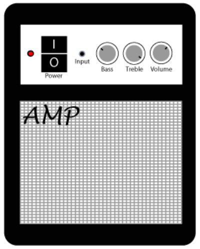
# Try It Out With AMP

1. Perform Bayesian data analysis
   (e.g., MrBayes)

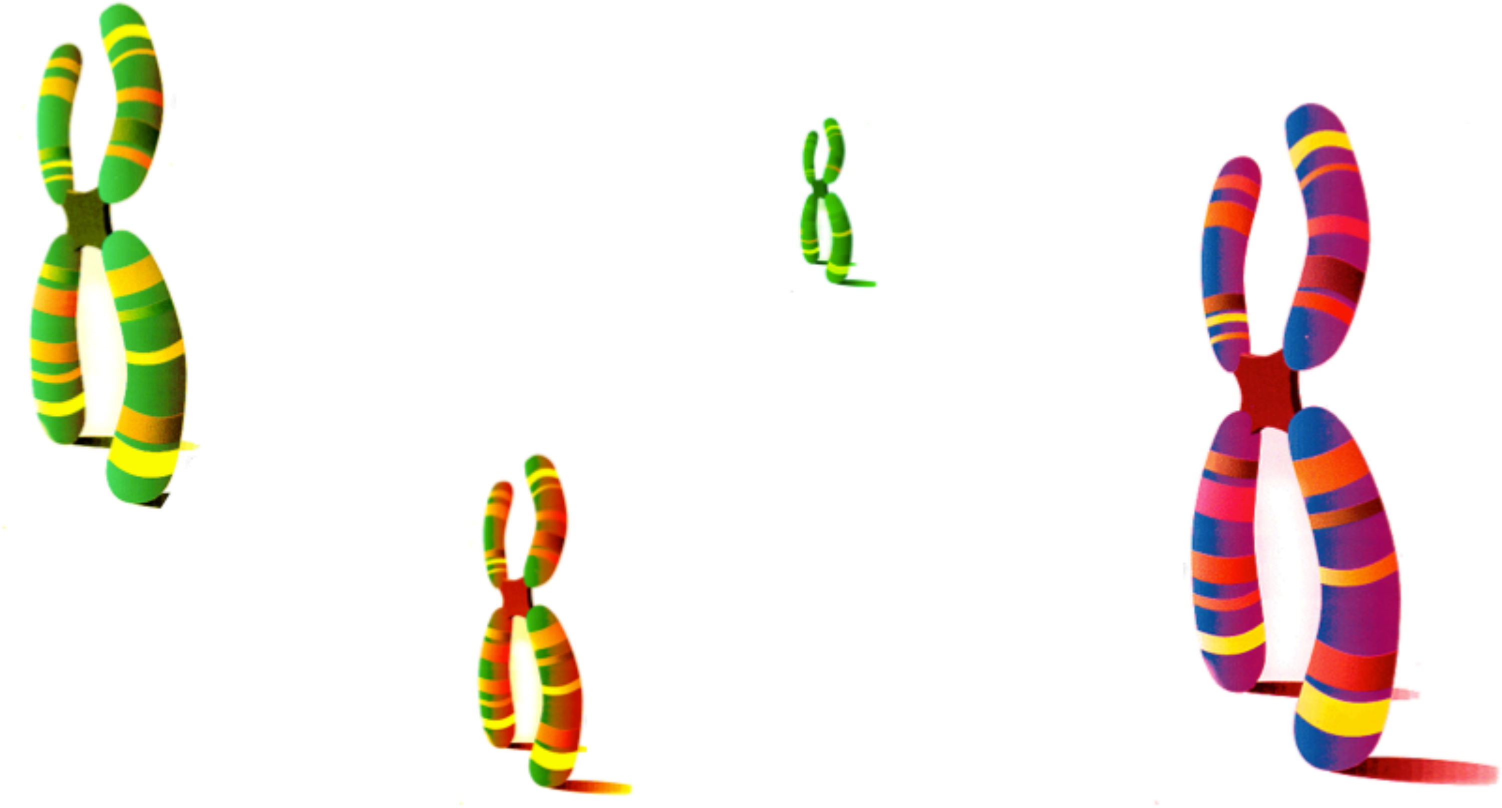
2. Simulate posterior predictive data
   (e.g., PuMA or MAPPS)

3. Analyze posterior predictive data sets
   (e.g., MrBayes)

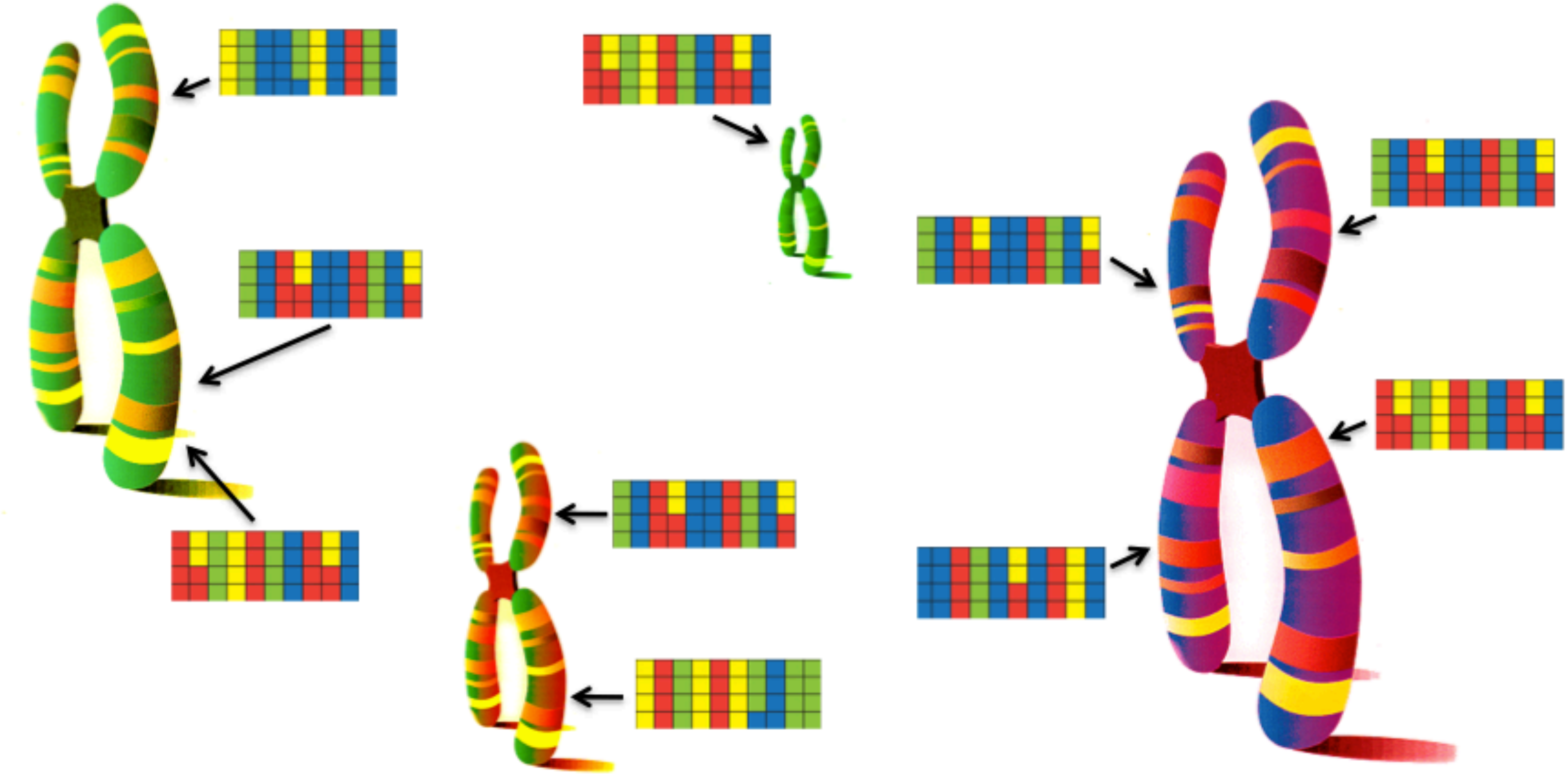
4. Calculate marginal posterior predictive *P*-values
   AMP (http://code.google.com/p/phylo-amp)



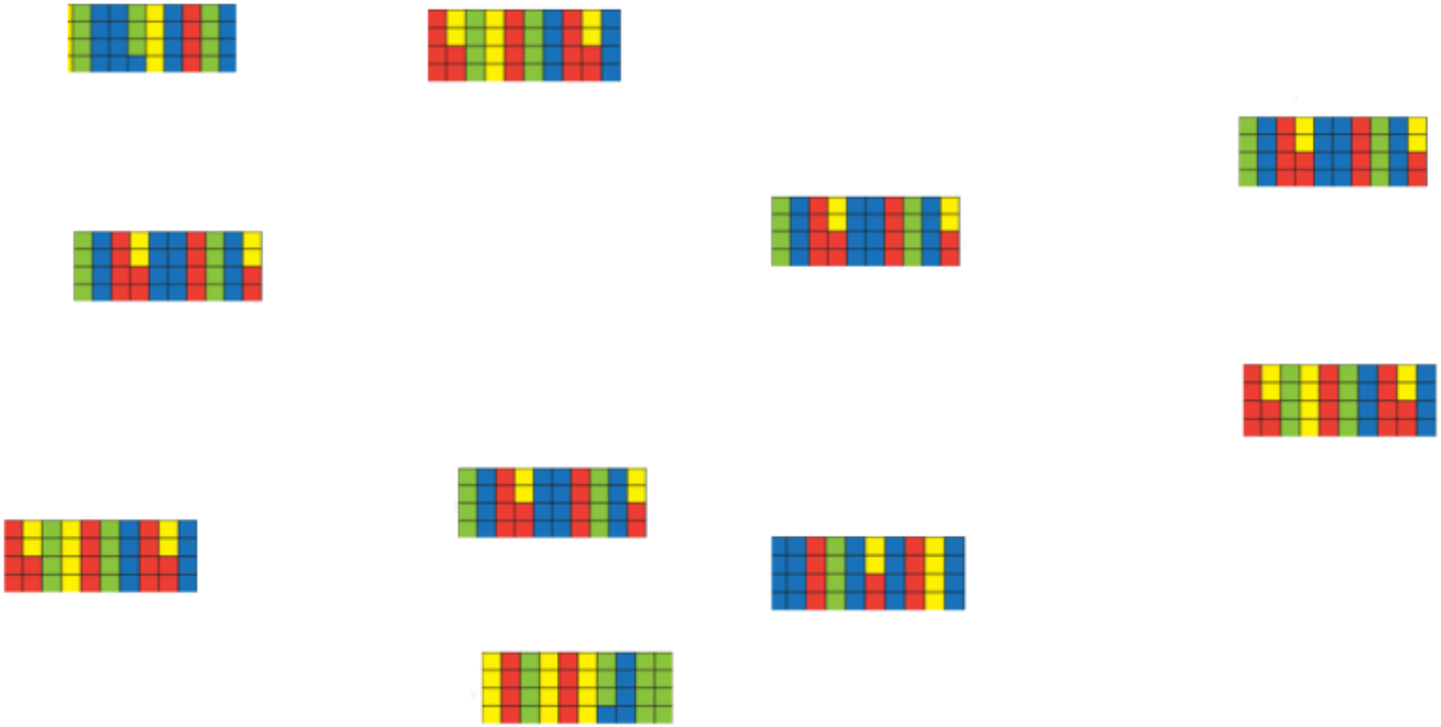
# Bottom-Up Phylogenomics



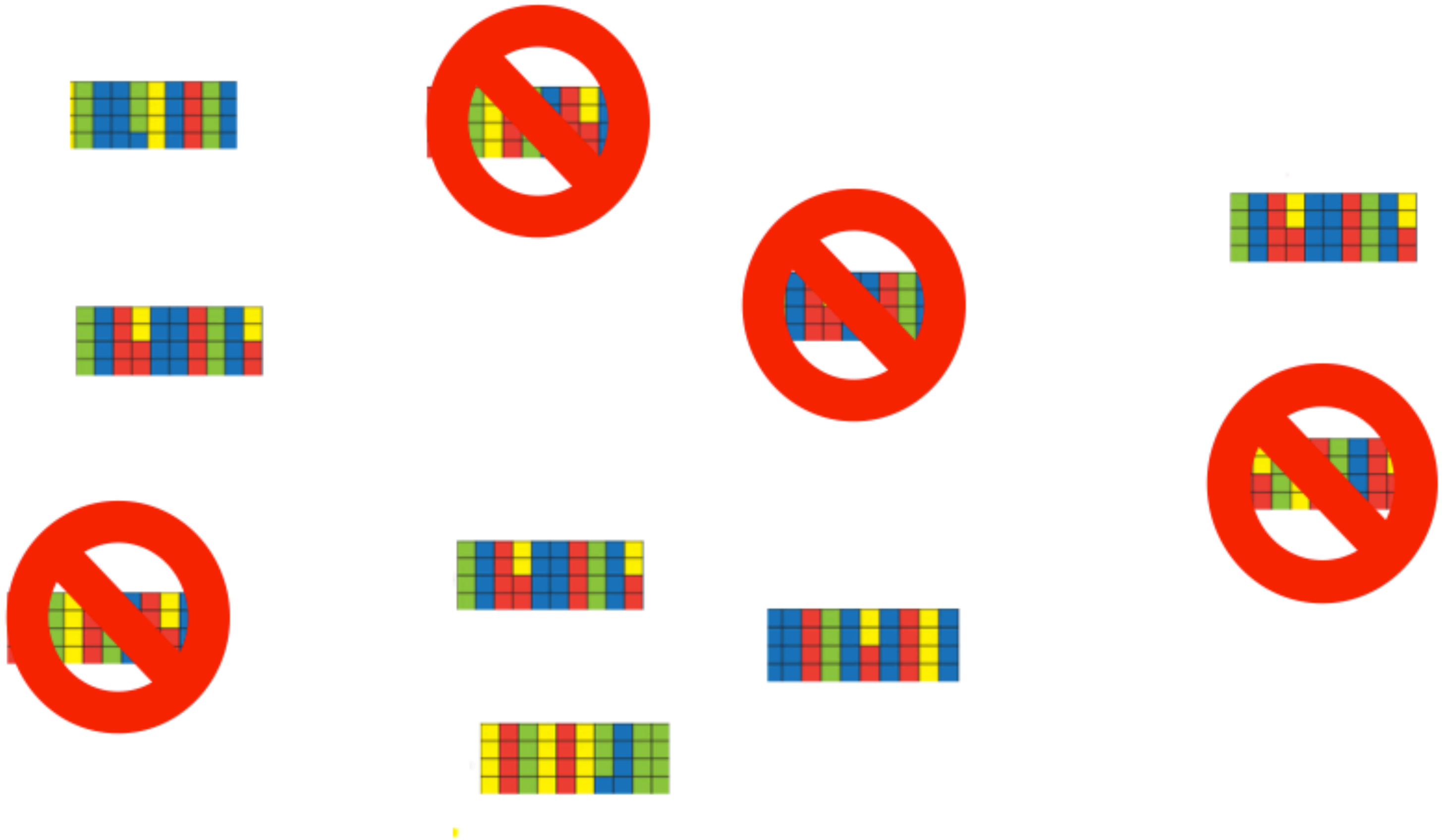
# Bottom-Up Phylogenomics



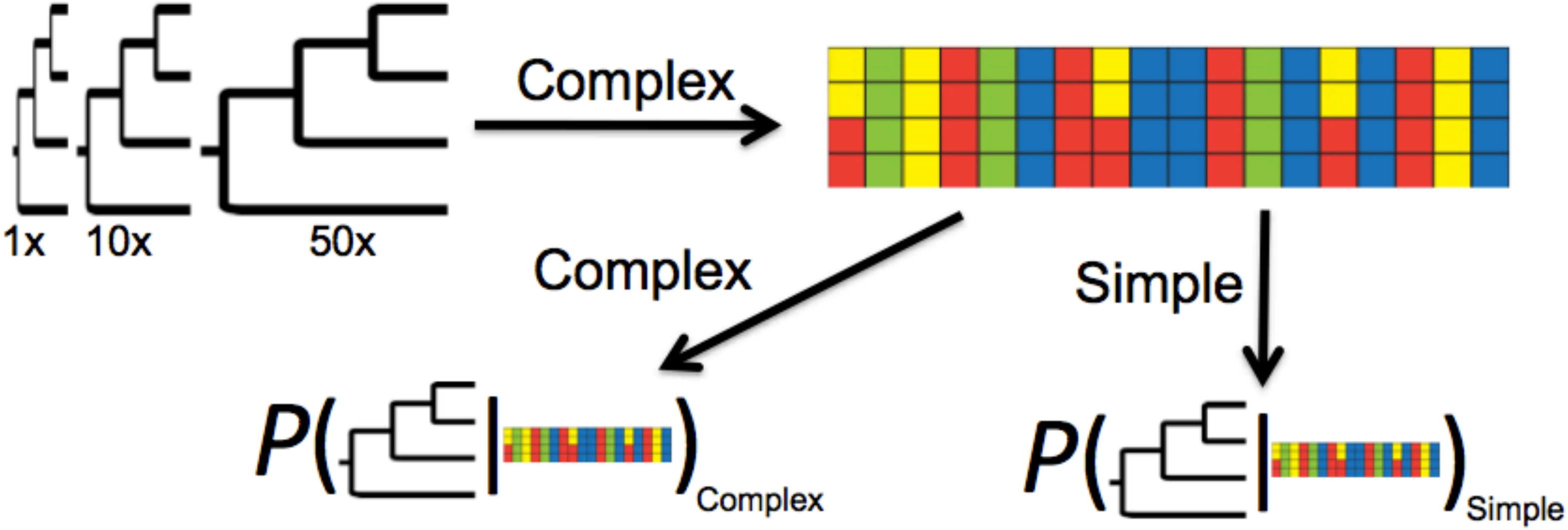
# Bottom-Up Phylogenomics



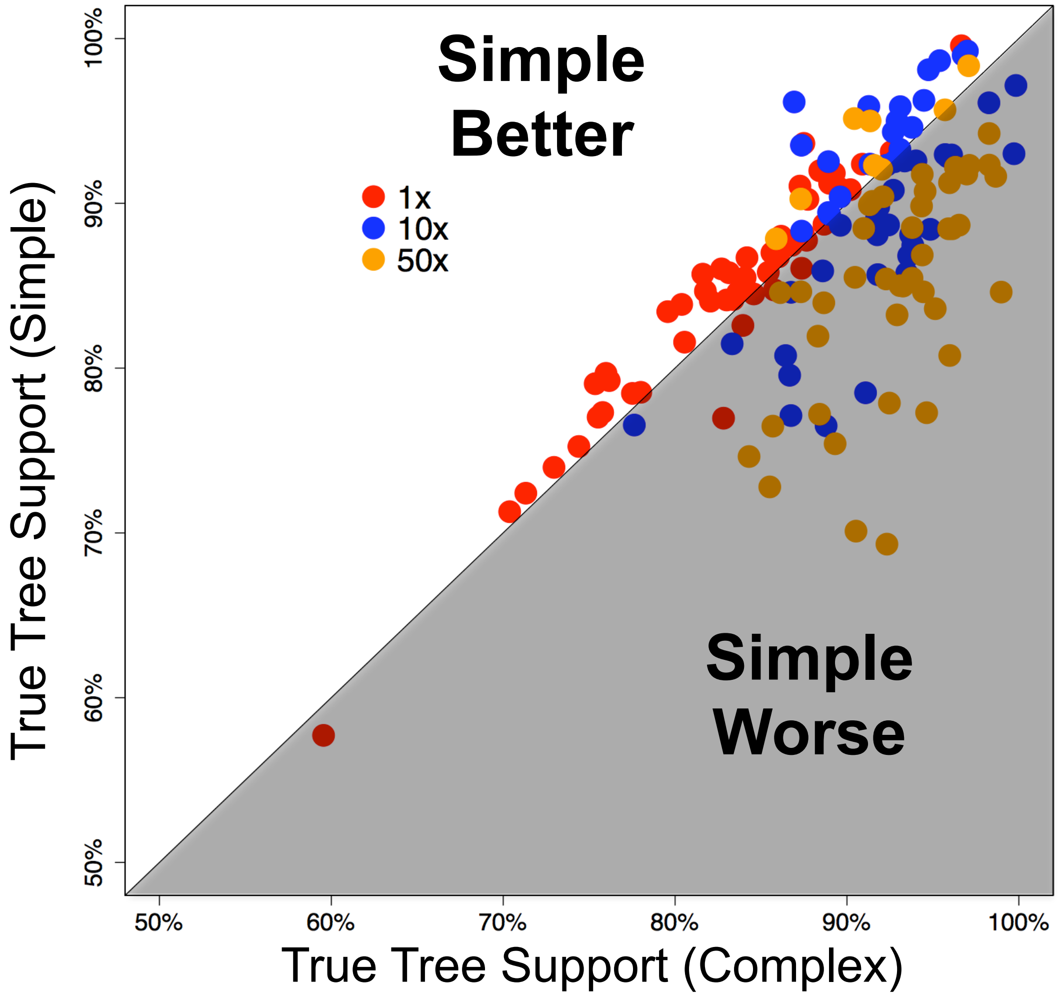
# Bottom-Up Phylogenomics



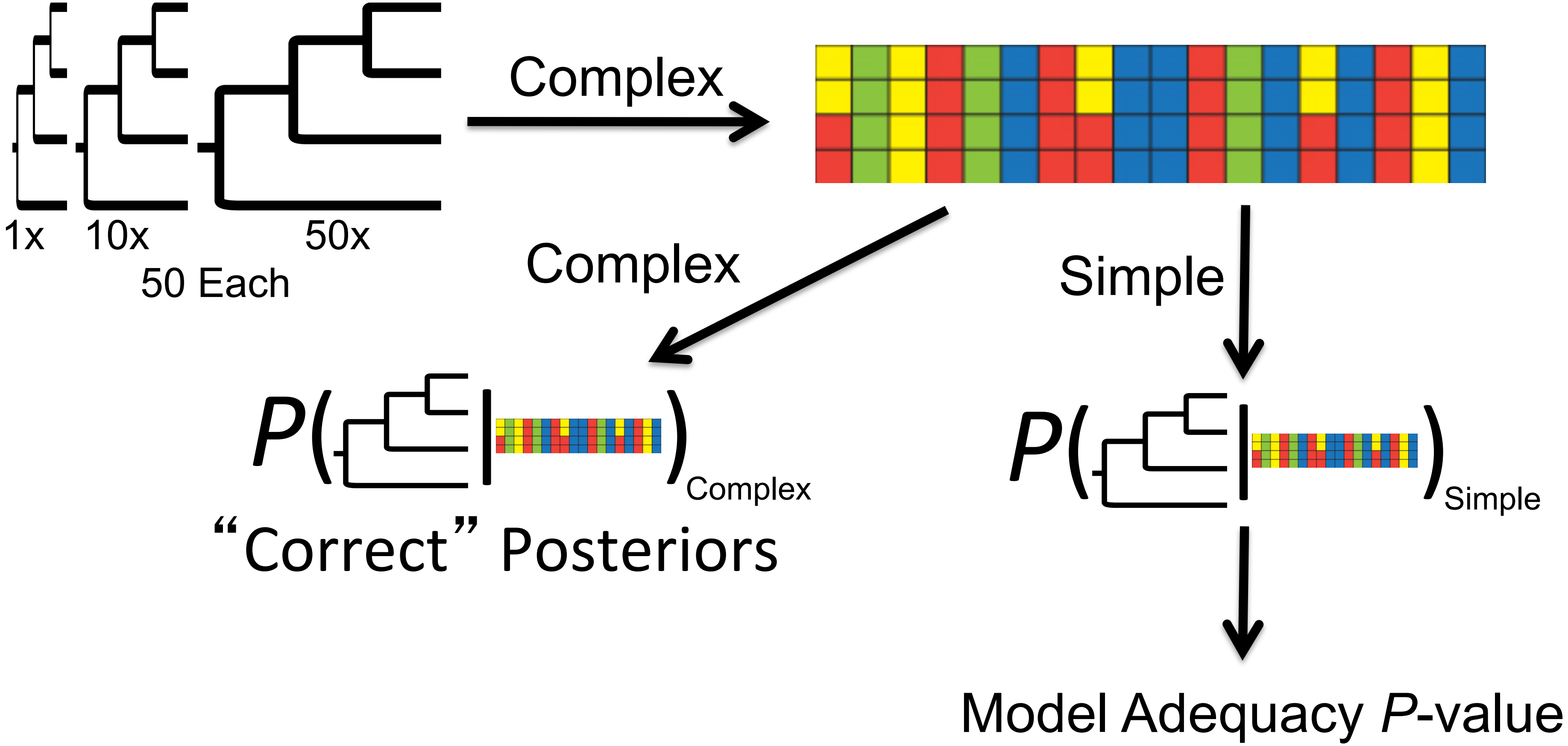
# Simulation Test



# Simple Model Biased with Longer Trees



# Testing Performance with Simulated Data



1x 10x 50x

50 Each

Complex

Complex

Simple

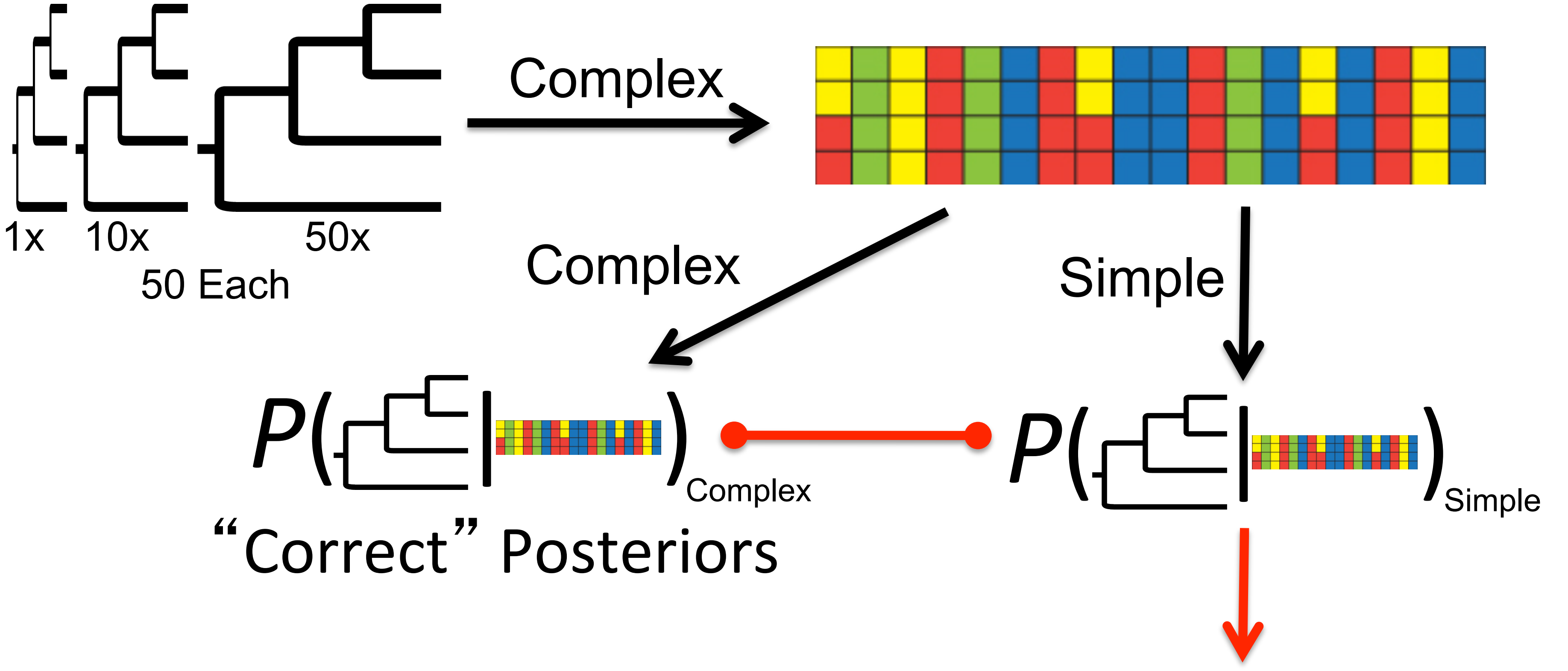
*P*( | )Complex

"Correct" Posteriors

*P*( | )Simple

Model Adequacy *P*-value

# Testing Performance with Simulated Data



1x 10x 50x

50 Each

Complex

Complex

Simple

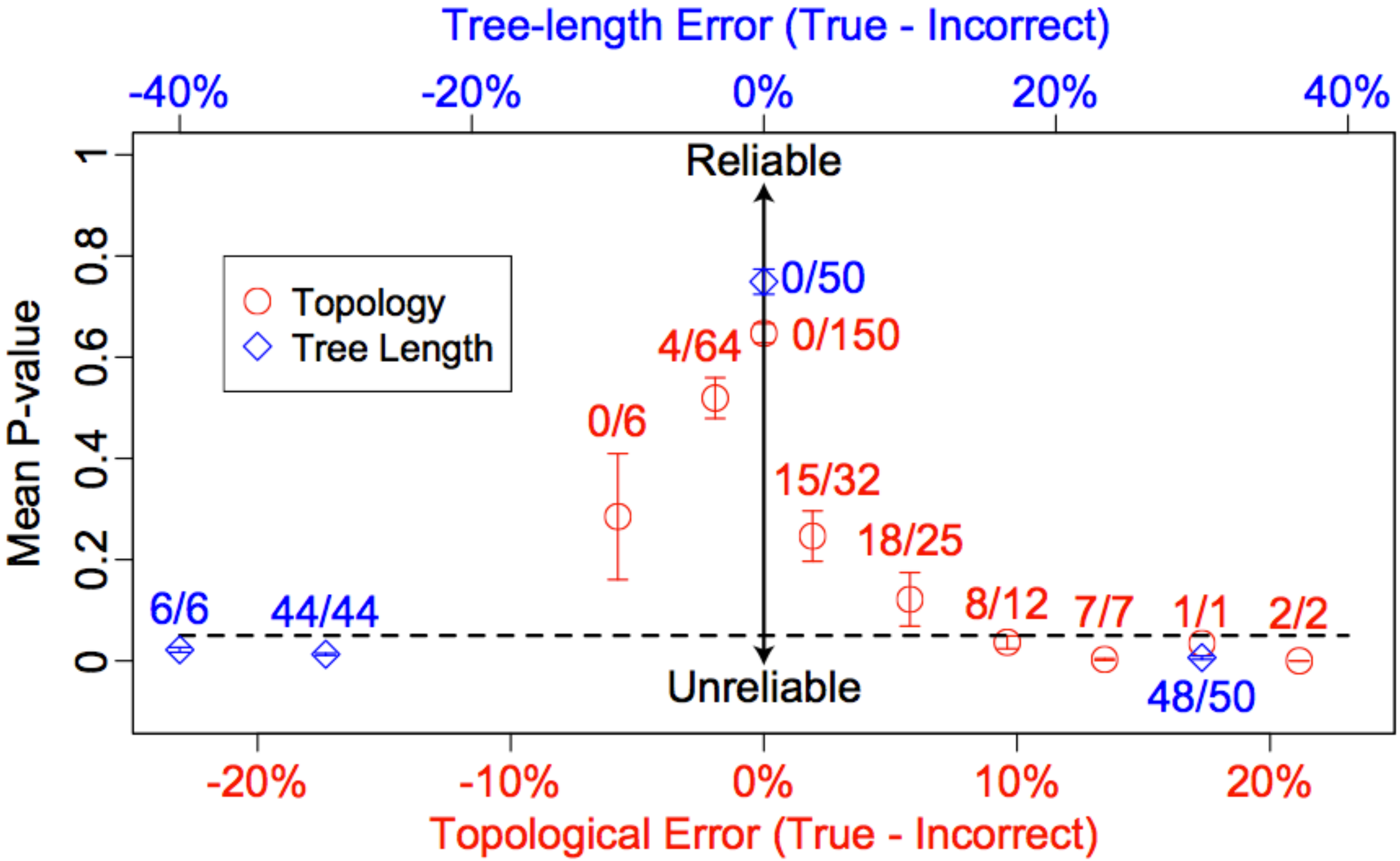
$P(\ \cdot\ |\ \cdot\ )_{\text{Complex}}$

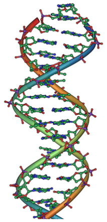
“Correct” Posteriors

$P(\ \cdot\ |\ \cdot\ )_{\text{Simple}}$

Model Adequacy *P*-value

# Simulation Test



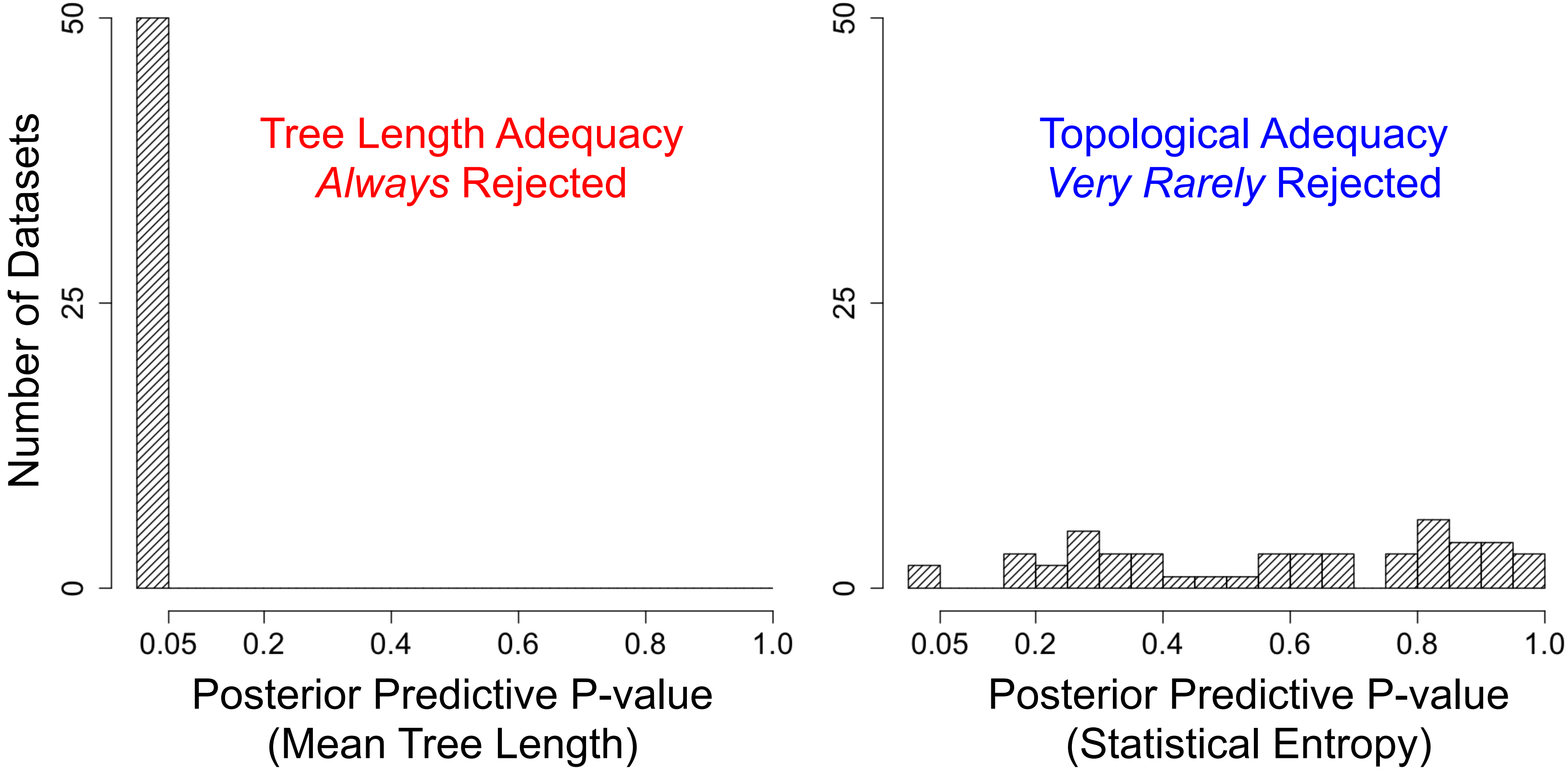


# Treelength Test Quantity
# Detects Biased Branch-Length Inference

- 50 replicate datasets
- Bipartition posteriors nearly identical to true branch-length prior
- Tree Lengths overestimated

# Treelength Test Quantity
# Detects Biased Branch-Length Inference

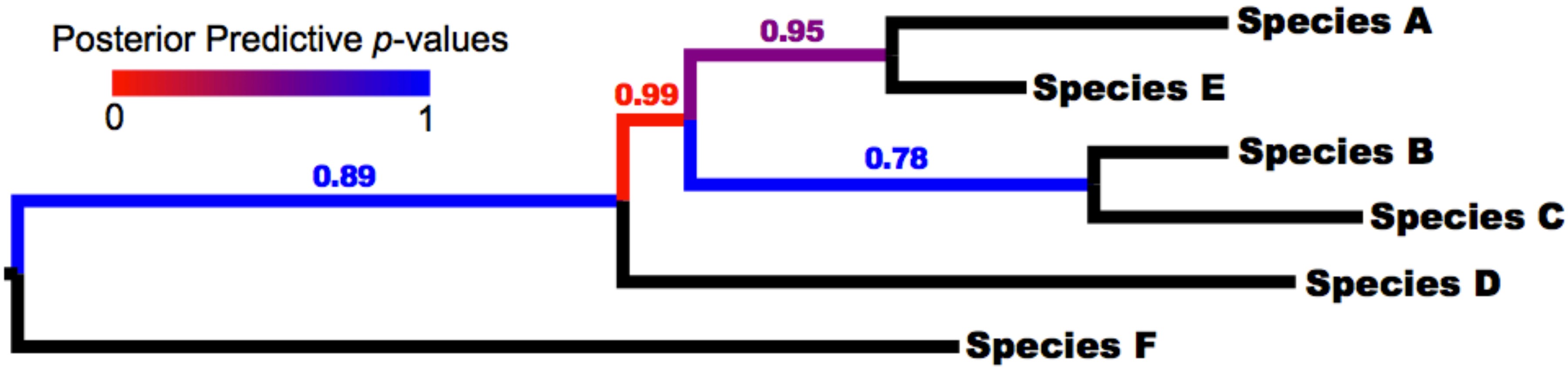
- 50 replicate datasets
- Bipartition posteriors nearly identical to true branch-length prior
- Tree Lengths overestimated

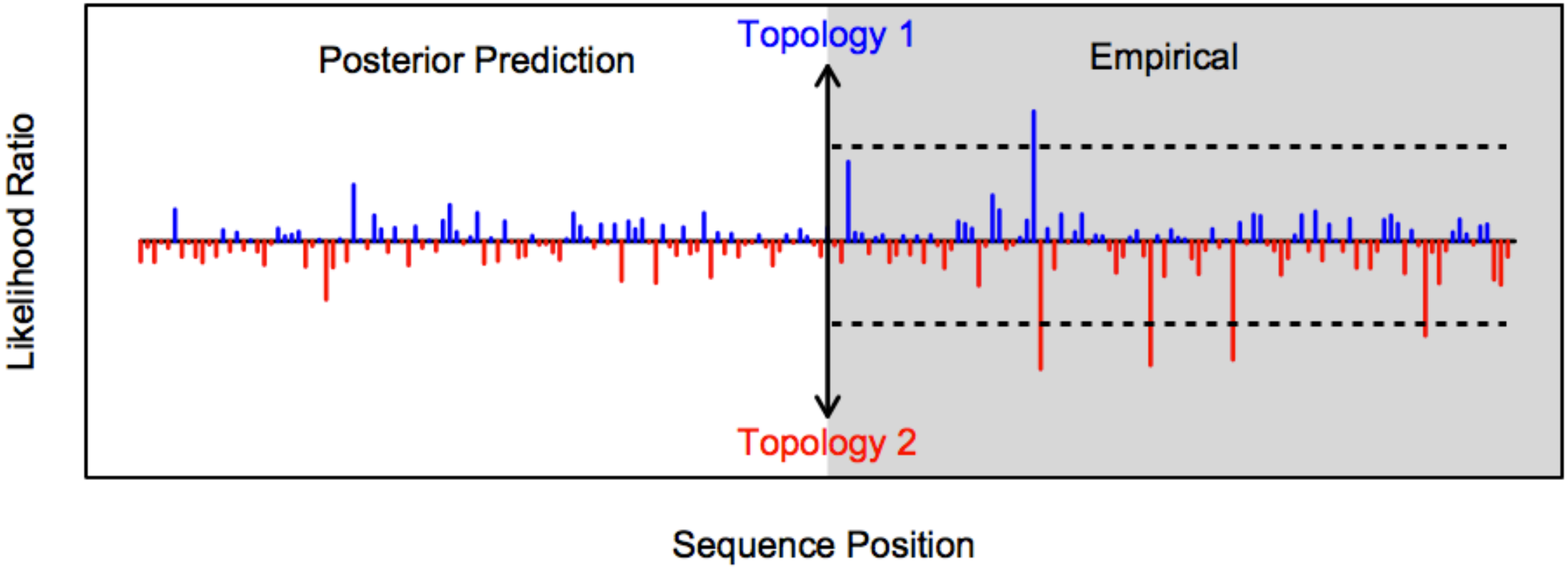


Tree Length Adequacy
*Always* Rejected

Topological Adequacy
*Very Rarely* Rejected

# Posterior Prediction





# The End