

Lecture 11 – Phylogenetic Uncertainty and Support

I. Nodal Support – Most often, the question of greatest interest has to do with how well supported are the various groups that are present in the optimal topology.

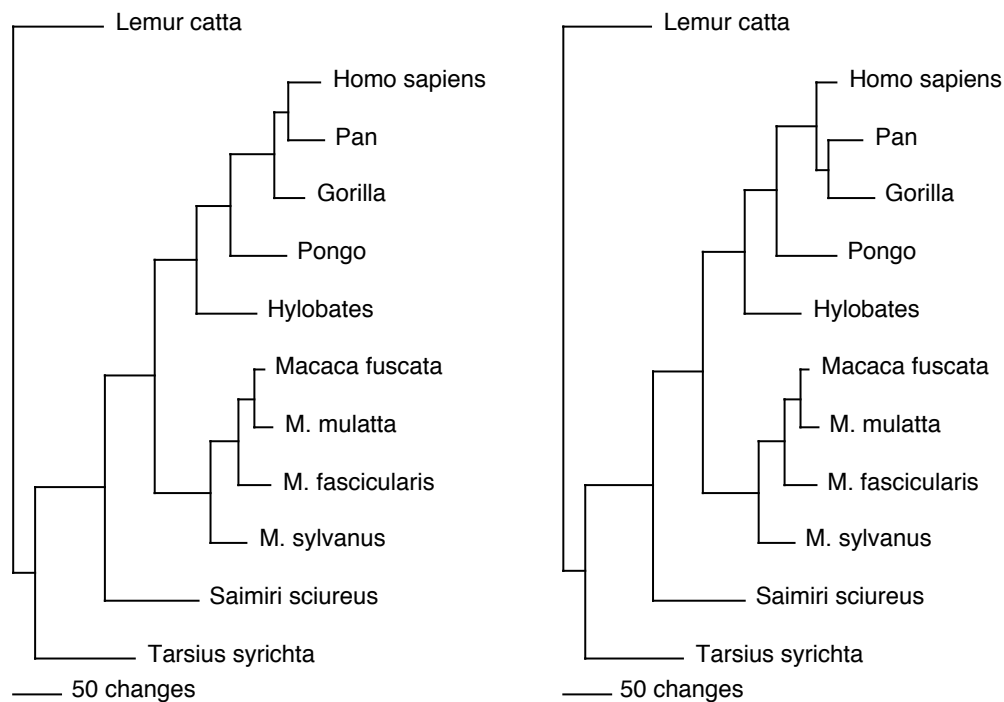
Again, methods that have been developed to assess this are either explicitly statistical or are non-statistical.

A. Decay Index

Bremer (1988. *Evolution*, 42:795) developed a parsimony approach to assess how nodal from a non-statistical perspective. The idea is that if an MP tree is, say 452 steps, and group (A,B,C) is found on that tree, we may wish to know how much longer is the best tree that doesn't contain group (A,B,C). This is the decay index for that group.

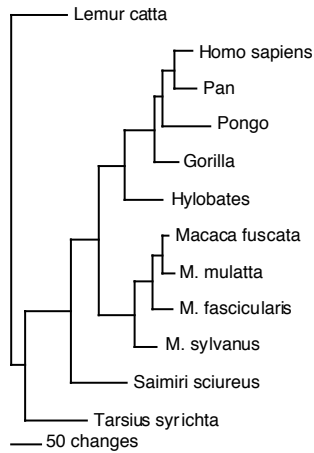
So for the primate mtDNA data set (Hayasaka et al., 1988. *Mol. Biol. Evol.*, 5:626) that accompanies PAUP as a sample data set there are two MP trees, each of length 996 steps.

On one tree, *Homo* & *Pan* are sister taxa whereas on the other *Pan* and *Gorilla* are sisters.

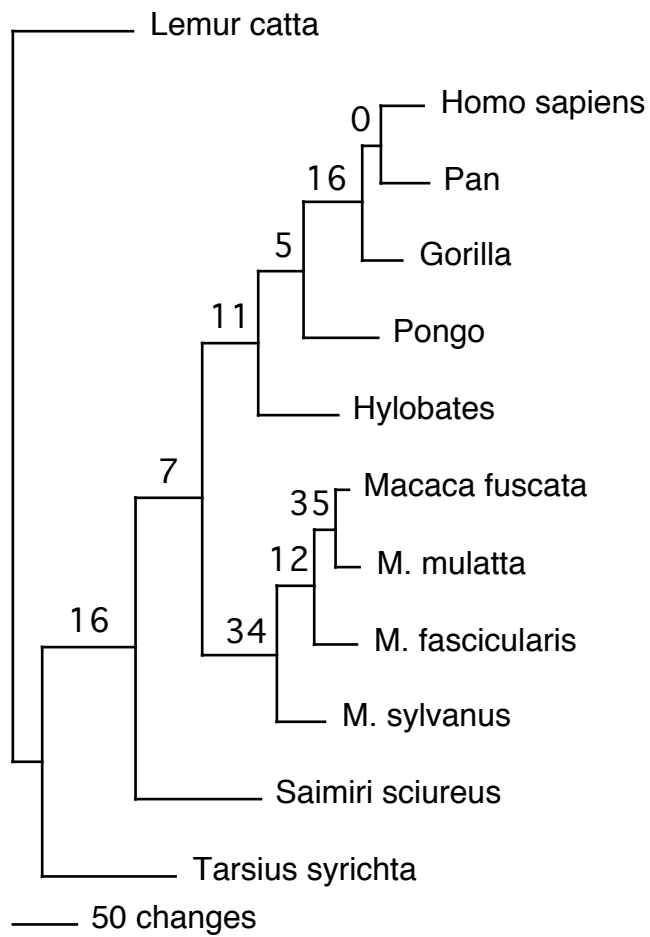


Therefore the each of these two groups has a decay index of 0.

The group (*Homo*, *Pan*, *Gorilla*) occurs on both. The shortest tree that doesn't contain this group (below) is 1012 steps. Therefore, the *Homo/Pan/Gorilla* node has a decay index of 16.



This can be done for each node, to give the following:



Each node has a decay index associated with it that indicates how much longer the shortest tree is that doesn't contain the particular node.

This certainly provides an assessment of how strongly our data supports each particular hypothesis of relationships; those that have higher decay indices are more strongly supported.

However, these are just numbers, and it's very difficult to decide how large a decay index is meaningful.

So we're left with decay indices lacking an appropriate statistical interpretation, but they're the measure of choice for examining nodal support for Hennigians.

B. Bootstrap Support.

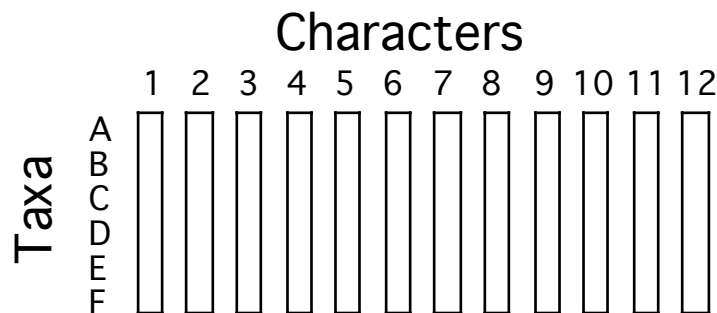
Felsenstein (1985. *Evolution*, 39:783) was the first to suggest using the bootstrap approach to assessing nodal support in phylogenies.

For those unfamiliar with it, there's an excellent introduction to it from a general perspective on pages 345 & 346 in the text.

The method was developed by Efron in 1979 to develop confidence intervals in cases where the true underlying distribution of a variable can't be assessed. The idea is that the distribution of the original sample (if it's large enough) will convey much about the underlying true distribution.

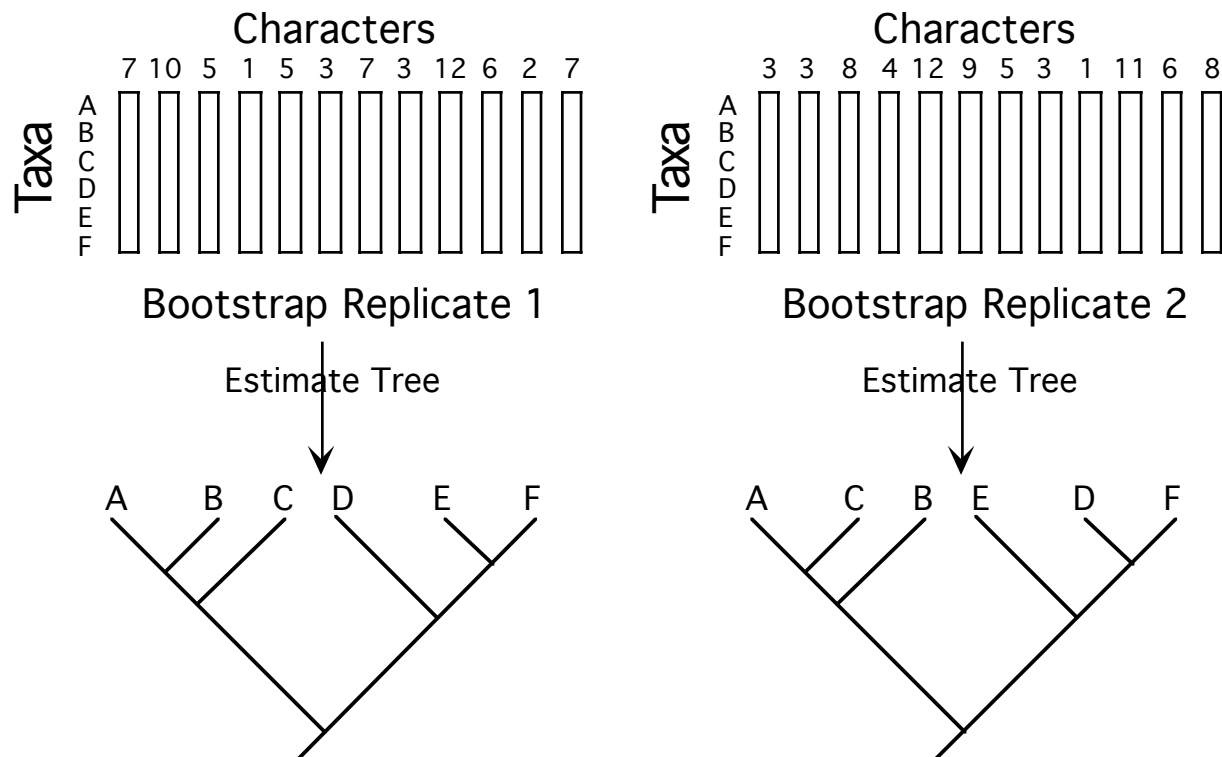
So we can treat our original sample as if it is the true distribution (it certainly estimates the true distribution), and take repeated samples of our original sample to mimic the variability we would see if we could resample from the true distribution.

In the case of phylogenies, we are interested in resampling characters from the original sample (i.e., the data) that we have collected.



So in phylogenetics, we sample characters to estimate the phylogeny. We then treat that sample of characters as estimating some underlying true distribution of characters.

The columns (characters) are re-sampled with replacement, and usually each pseudo-replicate is the same size as the original data set.



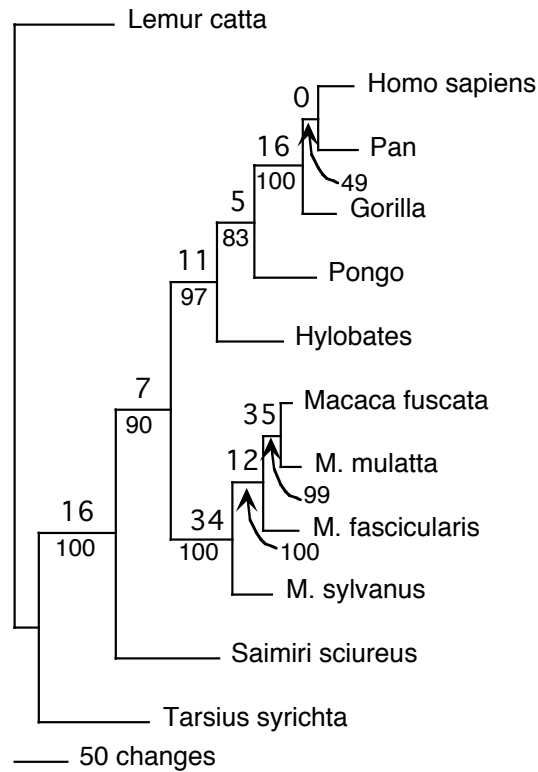
We then generate some large number (usually between 100 & 2000) of pseudo-replicate data sets by randomly selecting characters with replacement from the original sample.

We estimate the phylogeny for each of these pseudo-replicate data sets to generate a cloud of trees. These can be generated under any of the optimality criteria we've discussed, and the variation among these trees provides a measure of uncertainty in our original phylogenetic estimate.

A majority-rule consensus tree is used to compute the percentage of bootstrap replicates in which any particular node is found in the ML, MP or ME tree for that replicate.

This percentage is the bootstrap value for that node.

This is one of the two MP trees for the primate data, with MP bootstrap values indicated below each branch, and decay indices above the branch.



There are several things we need to point out:

- 1) The size of the original sample is a critical factor in the performance of the bootstrap. This makes intuitive sense. The original sample is taken as a proxy for the underlying parametric distribution; it must be large enough to reflect relevant features of the distribution.
- 2) For large data sets, bootstraps can take a very long time. Felsenstein (1993 Phylip Manual) has suggested that the uncertainty measured by bootstrap resampling is much larger than the error associated with extensive branch swapping in estimating optimum tree for each bootstrap replicate.

So one could do a FastBoot analysis, in which only a stepwise addition (or NJ) tree is built for each bootstrap replicate. This is at one extreme.

At the other extreme, one may conduct a full heuristic search for each bootstrap replicate (including stepwise addition with multiple random addition sequences and TBR branch swapping).

An intermediate strategy that has been shown to work well (DeBry & Olmstead, 2000. Syst. Biol., 49:171) is doing a greedy search on each bootstrap replicate that involves retaining only a single optimal tree in memory (i.e., MAXTREES = 1).

3) As long as we're using a consistent estimator of phylogeny, bootstrap values on nodes tend to be conservative as confidence intervals (Hillis & Bull, 1993. Syst. Biol., 42:182). If we're using an inconsistent estimator, of course bootstrap analysis may give us high confidence in incorrect nodes (i.e., if we're in the Felsenstein Zone), or they may give us too much confidence in a correct node (i.e., if the true tree is in the inverse Felsenstein zone).

Lots of work has been done on how to interpret the bootstrap values, and Joe does an excellent job summarizing that work on pages 335 – 345.

My take is that they can be taken as estimates of the statistical confidence we can place in a node (or anything else we may be estimating using the phylogeny), but that in some cases they're conservative and in other cases they're too liberal. As such, we need to treat them cautiously, and think about conditions that lead to each type of bias.

C. Bayesian Estimation of Nodal Support

Just in the last few years, Bayesian statistics have been applied to phylogenetics. I'll just give a brief introduction to Bayesian methods for those of you who are not familiar with the approach.

The idea of Bayesian analysis is intuitively very appealing. Given some data, a likelihood model, a quantification of prior our knowledge, we can calculate the probability of that some hypothesis is true.

The formalization of this is provided by Bayes' Theorem:

$$P(H_i | D) = \frac{P(H_i)P(D | H_i)}{\sum_{i=1}^s P(H_i)P(D | H_i)}$$

where $P(h_i | D)$ is the posterior probability of hypothesis i , given the data, D ;

$P(h_i)$ is the prior probability of hypothesis i (this is the quantification of prior knowledge);

$P(D | h_i)$ is the probability of the data, given hypothesis i (this is the regular likelihood function).

The denominator is the product of these summed across all s competing hypotheses.

So for phylogeny estimation, we can describe Bayes' Theorem as:

$$P(\tau_i | D) = \frac{P(\tau_i)P(D | \tau_i)}{\sum_{i=1}^s P(\tau_i)P(D | \tau_i)}$$

and this calculates the probability of tree i , given the data, the prior, and assuming the model.

So the $P(\tau_i)$, the prior probability of tree i , is usually set to $1/s$, where s is the number of possible trees. This represents an admission of ignorance, and is called a flat prior or a uniform prior (or an uninformative prior).

The summation in the denominator then is across all topologies.

The denominator is impossible to compute. In order to calculate it, we need to calculate the likelihood of all possible trees. This rendered fully Bayesian analysis impossible for the 25 years between the advent of computational phylogenetics and the paper by Ranalla and Yang (1996. *J. Mol. Evol.*, 43:304) and Mau's dissertation (1996) that led to the current popularity of Bayesian estimation.

This is due to the application of Markov Chain Monte Carlo to Bayesian estimation.

MCMC is an approach that allows one to derive a sample from an unknown distribution by growing a chain of states that are sampled from this distribution. This is accomplished by proposing a change to the current state and accepting that proposal following a set of rules.

So for phylogenies, we start the mcmc with some tree, let's say it's a random tree. This is the state of the chain in the first generation.

We then propose a change in the tree by proposing a random NNI (or some other type of tree rearrangement).

If the new tree has a higher posterior probability than the first, we accept the new tree.

This decision is made by calculating the ratio of the posterior probability of the new to previous state (tree):

$$R = \frac{P(\tau_{i+1})P(D | \tau_{i+1})}{\sum_{i=1}^s P(\tau_i)P(D | \tau_i)} \frac{P(\tau_i)P(D | \tau_i)}{\sum_{i=1}^s P(\tau_i)P(D | \tau_i)}$$

So if $R > 1$, we accept the new tree (it has a higher posterior probability than the previous tree).

If $R < 1$, we draw a random probability (between 0 & 1). If this is $< R$, we accept the change, if not, we return to the previous state (tree).

By examining the acceptance ratio, we can see a couple of simple things.

First, the impossible denominators for each state cancel. We never have to compute the impossible denominator.

Second, if the priors are the same across all topologies:

$$P(\tau_{i+1}) = P(\tau_i) = 1/s$$

The priors cancel and the likelihood function determines the shape of the posterior probability distribution (of trees).

So using these rules, and starting anywhere in tree space, if we run the chain long enough, eventually the frequencies of states in the chain (i.e., trees), will converge to their frequencies in the posterior probability distribution.

Another way of saying this is that once the chain reaches equilibrium, it samples trees proportionally to their posterior probability.

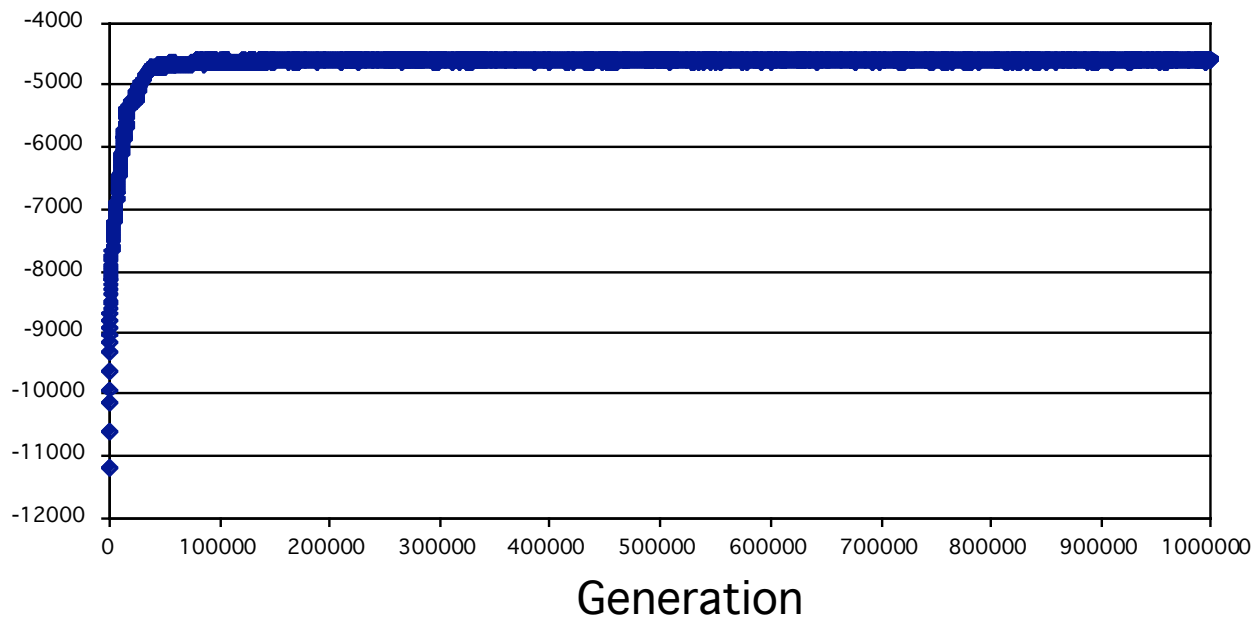
Now we need to make a few points.

First, all that matters in theory is that all the potential manifestations of the states (i.e., all trees) can be reached by a series of single steps of our proposal mechanism (i.e., NNI branch swaps). As long as this is the case, the MCMC will converge on the stationary probability distribution if it's run long enough.

Nevertheless, convergence properties will vary. Proposal mechanisms that change the state very little will take very (very, very....) long chains to provide an adequate sample because space will be explored too narrowly. Conversely, proposal mechanisms that change the state too dramatically will result in most proposals being rejected and the chain sticking on a state for a long time.

Far too little has been published on the effect of proposal mechanisms on Bayesian estimation of phylogenies.

Second, we have to discard early generations of the chain, because it takes a while for the chain to converge traverse space. This is called the burn-in, and a first-approximation of the burn-in usually entails a plot of the likelihood of the current tree across generations.



So once we've run the chain for ca. 100K generations, we seem to have converged, at least with respect to the likelihood.

So we can sample the tree every so often from this chain to generate a collection of trees (say 10,000 of them). Ideally, each topology will be present in the sample at a frequency that's proportional to its posterior probability.

We can compute the majority-rule consensus tree from this collection of trees to see how frequent a particular node is in the sample. This becomes **our posterior probability that the node is correct, conditional on the data, the priors, and likelihood model.**

There are lots of other issues to deal with regarding Bayesian estimation.

First, the phylogeny problem is a terribly complex one. Remember that the likelihood of a particular tree topology includes a vector of branch lengths, each of which is estimated with uncertainty and a vector of model parameters, each of which is estimated with uncertainty.

So we can take our model, start the mcmc with initial states for model parameters, and include proposals to change each of those as well as topology in our mcmc.

This means that we need to place priors on each of these parameters as well. Typically, researchers using MrBayes do something like this:

Parameters

```
-----  
Revmat          1  
Statefreq       2  
Shape           3  
Pinvar          4  
Topology        5  
Brlens          6  
-----  
  
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(0.05,50.00)  
4 -- Parameter = Pinvar  
   Prior       = Uniform(0.00,1.00)  
5 -- Parameter = Topology  
   Prior       = All topologies equally probable a priori  
6 -- Parameter = Brlens  
   Prior       = Branch lengths are unconstrained:  
                 Exponential(10.0)
```

Proposed changes to these parameters are usually as follows:

The chain will use the following moves:

```
With prob. Chain will change  
3.57 % param. 1 (revmat) with multiplier  
3.57 % param. 2 (state frequencies) with Dirichlet proposal  
3.57 % param. 3 (gamma shape) with multiplier  
3.57 % param. 4 (prop. invariants) with beta proposal  
53.57 % param. 5 (topology and branch lengths) with LOCAL  
10.71 % param. 5 (topology and branch lengths) with extending TBR  
10.71 % param. 6 (branch lengths) with multiplier  
10.71 % param. 6 (branch lengths) with nodeslider
```

So the advantage of this is that we're estimating nodal probabilities that actually account for uncertainty in all the other parameters.

In ML bootstrap analysis, we're using point estimates (the ML values) of each parameter (jointly estimated).

The disadvantage is that now we're generating an even more complex parameter space through which to traverse.

This will make it more difficult to converge, and increase the chance that we have false stationarity by being trapped on local region of high posterior probability.

This can be ameliorated by running several chains simultaneously (Metropolis-Coupled MCMC, or MC³), and every now and then proposing a state from a different chain and trying to switch.

This is called heating the chains, because we're trying a potentially large step every now and then.

If the independent chains all converge to the same plateau, we can have some hope that we're not getting stuck on a local optimum.

Also we need to assess convergence with respect to the particular parameter of interest.

In addition, we have the problem of truncating priors.

Say we have a parameter such as the gamma distribution shape parameter, which is unbounded.

If we want to use a flat prior for this parameter, we need to truncate it. In the example above, it's truncated at the lower end by 0.05 and at the upper end by 50. This certainly seems reasonable, but it creates a bias.

Joe gives an example on page 305 where different truncations of a flat prior on a branch length leads to Bayesian estimate that excludes the ML estimate.

So remember that the goal isn't to provide a point estimate of the phylogeny, but the posterior distribution of trees.

It's this distribution that we should be sampling from in testing hypotheses and making the cool inferences that Luke will be teaching you about.