Lab 07: Maximum Likelihood
Model Selection and RAxML Using CIPRES

Exercise 1: Maximum Likelihood (ML) Model Selection

Maximum likelihood (ML) is a statistical method for reconstructing trees. Basically, ML operates by calculating the following conditional equation: What is the likelihood of observing a data set given a phylogeny and a model of DNA sequence evolution? The tree with the highest likelihood score is considered the best tree. When using maximum likelihood to build trees, **we have to first select a model of DNA sequence evolution**. Let's do this first and then learn a bit more about ML and model selection while we wait for it to run.

jModelTest

1. Open jModelTest by clicking jModelTest.jar. The main window and menu should now be open.

2. Select **File → Load DNA Alignment** and then select your Nexus file (Use your own data or use the primate-mtDNA.nex file). Okay, *jModelTest* should have read the file and tells you how many sequences (basically how many OTUs) and how many sites (basically how many nucleotides) the file has.

3. Now select **Analysis → Compute likelihood scores**

4. A new window should now pop up with several options to choose from. Feel free to examine these different parameters on your own. For now set the default settings and make sure under "Base tree for likelihood calculations" that **ML Optimized** is selected. Select **Compute Likelihoods**.

5. Okay, let that run. How fast it takes will depend on your computer and your data, but it will be at least a few minutes. The program is computing likelihood scores for 88 different nucleotide substitution models. Let's learn a bit more about these different models while we twiddle our thumbs and wait.

Models of Nucleotide Change

The Transition Matrix

The transition matrix (not as in transition/transversion) is a matrix showing the instantaneous stochastic rate of change between any two nucleotides. It can be used to calculate the chance of one nucleotide changing into another on a branch with a given length.

The most unrestrained matrix would look like this:

	A	C	G	T
A	$-\alpha-\beta-\gamma$	α	β	γ
C	δ	$-\delta-\varepsilon-\zeta$	ε	ζ
G	η	θ	$-\eta-\theta-\iota$	ι
T	κ	λ	μ	$-\kappa-\lambda-\mu$

As you can see, the diagonals are all negative as each nucleotide will be changing away from itself at any instant, so that each row adds up to 0. Furthermore, the average rate of change of all the off diagonals is normalized to 1, so that you can eliminate another parameter for a total of 11 parameters.

On the other hand the Kimura two parameter model would look like this:

	A	C	G	T
A	$-\alpha-2\beta$	β	α	β
C	β	$-\alpha-2\beta$	β	α
G	α	β	$-\alpha-2\beta$	β
T	β	α	β	$-\alpha-2\beta$

Here there are two parameters, transition and transversion rate, which can be reduced to just one by normalizing the matrix.

Most programs (*PAUP** included) can only calculate matrices with reversible models. This means that change has an equal probability of happening in either direction on a branch. Thus trees can be evaluated as unrooted networks, making the computationally-intensive likelihood calculations much easier. For a model to be reversible it must be true that:

$$\pi_X R_{X>Y} = \pi_Y R_{Y>X}$$

where $R_{X>Y}$ is the instantaneous rate of change from nucleotide X to nucleotide Y, and π_X is the equilibrium frequency of nucleotide X. The equilibrium frequency is the frequency of that nucleotide if the substitution process is allowed to run forever, and can be considered another parameter. Thus any model in which $R_{X>Y}=\pi_Y r_{XY}$, will be reversible. So the General Time Reversible (GTR) matrix looks like:

	A	C	G	T
A	–	$\pi_C r_{AC}$	$\pi_G r_{AG}$	$\pi_T r_{AT}$
C	$\pi_A r_{AC}$	–	$\pi_G r_{CG}$	$\pi_T r_{CT}$
G	$\pi_A r_{AG}$	$\pi_C r_{CG}$	–	$\pi_T r_{GT}$
T	$\pi_A r_{AT}$	$\pi_C r_{CT}$	$\pi_G r_{GT}$	–

with the diagonal filled in appropriately. The sum of the equilibrium frequencies for all four bases must equal one, so that there are three equilibrium frequency parameters. Furthermore,

one of the rate parameters can be eliminated by normalizing the matrix, leaving eight parameters total. General Time Reversible (GTR) represents a family of nested models that encompass 64 models with different combinations of parameters. Nested models are special cases of more general models. We briefly learned about some of these nested models already:

- JC : Jukes and Cantor (1969) - All nucleotide substitutions are equal and all base frequencies are equal. This is the most restricted (=specific) model of substitution because it assumes all changes are equal.
- F81 : Felsenstein (1981) - All nucleotide substitutions are equal, base frequencies allowed to vary.
- K2P : Kimura two-parameter model, Kimura (1980) - Two nucleotide substitutions types are allowed, those between transitions and transversions. Base frequencies are assumed equal.
- HKY85: Hasegawa-Kishino-Yano (1985) - Two nucleotide substitutions types are allowed, those between transitions and transversions. Base frequencies are allowed to vary.

Question #1: **Which of these models is the least complex? Which of these models is the most parameter rich?**

Proportion of Invariable Sites (I)

This is a model that assumes some proportion of the sites, p_i, cannot change. Thus it makes two calculations for each base pair. First it calculates the chance, λ_i, that that base pair would have the observed distribution that it does if it could not change. This will be 1, if it is the same in all taxa, or 0, if there are any differences among the taxa. It then calculates the probability, λ_v, that it would have the observed distribution if it could change, using the transition matrix and the tree. Then it calculates the overall likelihood for that base as:

$$\lambda = p_i \lambda_i + (1\text{-}p_i) \lambda_v$$

Among-site rate variation (Γ, also abbreviated G by jModelTest)

Under the null hypothesis, all sites are assumed to have equal rates of substitution. One way of relaxing this assumption is to allow the rates at different sites to be drawn from a gamma distribution (with the mean value across all sites within a class, such as A-T, represented in the substitution matrix). The gamma distribution is used because the shape of the curve (α = shape parameter) changes dramatically depending on the parameter values of the distribution.

This calculation is done essentially the same way as it is for invariable sites. The likelihood is calculated for each value of the gamma distribution for each base pair and added together. In practice this is only done for a few values of the gamma distribution, as there are an infinite number of possible values for the gamma distribution and each likelihood calculation is computationally burdensome. This serves as a good approximation of a true gamma distribution.

jModelTest continued:

Once the program is finished computing the likelihood scores, we need a way to evaluate which one is best. Adding parameters to a model always increases the maximum likelihood of the data. However, if a model has too many parameters, then maximum likelihood becomes unreliable. Therefore to accept a new parameter into your model it must produce a significant increase in the

likelihood. How do you tell if a difference in likelihood is significant? We want the model that best explains our data without adding too many parameters.

6. Select **Analysis**. You'll notice that you can now select "Do AIC Calculations…", "Do BIC Calculations…", or "Do DT Calculations…"

7. Select **Do AIC Calculations**. Another window will pop up. Select **Use AICc correction, Calculate parameter importances, Do model averaging, and Write PAUP* block**. Select **Do AIC calculations**.

That was much quicker. What is AIC or AICc for that matter? What about BIC?

The Akaike Information Criterion (AIC) can be thought of as the amount of information that is lost when we use a specific model to approximate the real process of molecular evolution. Basically AIC compares several candidate models simultaneously and is used to compare both nested and non-nested models. AICc is used to correct for small sample size. AICc will approach AIC with larger sample sizes.

The Bayesian Information Criterion (BIC) can be alternately used. This criterion gives equal priors for all competing models and choosing the model with the smallest BIC is equivalent to selecting the model with the maximum posterior probability. [If you have never been introduced to Bayesian statistics, what I just said probably doesn't make any sense to you. Don't worry, this will become clearer after we introduce Bayesian Inference next week.]

8. Select **Results -> Show results table**. Click on AICc and then select the AICc column. The chosen model has the lowest AICc score and will be highlighted.

Question #2: **Which model was selected using AICc? What is the likelihood value for this? Use the BIC calculation (remember to select Write PAUP* block). Was the same model selected as with AICc? What is the likelihood value for this?**

What if these two criteria differ in their model selection? Currently AICc and BIC are generally accepted as the two best criteria. I would suggest running each in PAUP* to see if they affect your tree inference. Keep the jModelTest window open.

Exercise 2: Maximum Likelihood (ML) in PAUP*

First let's use the parameter values chosen by j*Modeltest*.

In the j*Modeltest* output file you will find a PAUP block that can be inserted directly into the Nexus file. Scroll up in the window to find this for the AICc. It starts

BEGIN PAUP;
and ends with
END;

This block changes the Likelihood Settings (Lset), by setting the base frequencies at equilibrium (Base), the number of substitution types (Nst), the rate matrix of instantaneous substitution rates

(Rmat), the among site rate variation (Rates), the shape of the gamma distribution (Shape), and the proportion of invariant sites (Pinvar).

Copy the PAUP block from the text file. I also include the line above this [! Likelihood settings …etc]. **Edit** your Nexus file and paste the PAUP block from j*Modeltest* directly into it. It can go after any END; statement.

Execute the newly-edited sequence file in PAUP* again. Change your working directory to a folder for the day if you'd like.

```
paup> execute filename
```

Set the optimality criteria to likelihood:

```
paup> set criterion=likelihood
```

Make sure you run a heuristic search.

```
paup> hs
```

You can use savetrees to write your tree to file and save information about the branch lengths:

```
paup> savetrees file=aiccmltree.tre brlens
```

A new tree file called "aiccmltree.tre" should appear in your specified folder.

Question #3: **Take a screen shot of your tree using the AICc PAUP block in FigTree and send it to me. Do the same for the BIC PAUP block.**

Exercise 3: Get Acquainted with CIPRES and RAxML

The CIPRES Science Gateway is a computational server that hosts popular phylogenetic research tools. There are four necessary steps to use these tools: Create an account, upload your data, submit your tasks, and analyze your output.

Go to the CIPRES Gateway website: http://www.phylo.org/index.php/portal/

Click on the icon: 

If you have never used CIPRES, you'll need to register. So go ahead and do that, then login.

Under the tab "**My Workbench**" create a new folder for today. When you create the folder, you'll notice that two subfolders are created → Data and Tasks.

Let's examine the "**Toolkit**" tab. Look at all the great resources for you to use!

Today we're going to use **RAxML-HPC2 on XSEDE**, but MrBayes and BEAST are available here and can be very useful for your projects. We'll discuss both of these programs next week in lab. RAxML (Randomized Axelerated Maximum Likelihood) is a program for Maximum Likelihood inference of large phylogenetic trees. This program employs heuristics to reduce

likelihood search time including building an initial tree under parsimony and incorporating a cooling schedule that allows "backward steps" during the hill-climbing process (see Stamatakis *et al*. 2005 for more details). Keep in mind that RAxML gets your likelihood tree quickly, but does not search through tree space as rigorously – there is always a trade-off.

You will first need to upload the file "primates.phy". Select your Data subfolder and Upload data. RAxML requires files in Phylip format like this file.

Question #4: **Describe the Phylip file format for me. What do the numbers at the top of the file indicate?**

Select **RAxML-HPC2 on XSEDE** and a new window pops up. Add a description for what you wan to call today's exercise. Select Input Data and select the primates.phy file you just uploaded. Select 38 Parametes Set.

There are many options here and I won't go into all of them.

1. You can select the maximum number of hours you want to run the analysis. This is a small data set so the default of 0.25 right now is fine.
2. Sequence Type is Nucleotide
3. Enter the number of taxa and enter the number of patterns (=nucleotides). For some reason I can only edit this after checking the box "I have a data set that may require more than 125 GB of memory" (which is not true in this case). Go ahead and click that so you can edit these fields and then un-click.
4. Set the outgroup as Lemur_catta
5. Leave these defaults (explore these when you run your own analyses though)
6. Advanced Parameters → Nucleic Acid Options
7. Use GTRGAMMA for the bootstrapping phase and GTRGAMMA for the final tree (takes longer)
8. Configure Bootstrapping → Select the box for "Conduct Rapid Bootstrapping"
9. If you want the best ML tree select Conduct a rapid Bootstrap analysis and search for the best-scoring ML tree in one single program run. (-f a)
10. Change the # of bootstrap iterations if you'd like – we can easily run 1000 on this data set.
11. Save Parameters → Save and Run Task

You will be sent an e-mail when your task is finished (very quickly for this example). Go back to CIPRES, select "view output". Download the **stdout.txt** file and examine this file to make sure your parameters were set correctly. You can check the likelihood value as well. Also download **RAxML_bipartitions.result**. This is your best ML tree with bootstrap values.

Question #5: **What was the likelihood value? How does this compare with your previous analyses from today? Load your tree into FigTree and make sure the bootstrap values are visible, take a screen shot, and send it to me.**