Lab 05:
Maximum Likelihood Inference
and Models of DNA Sequence Evolution

Updated by Will Freyman

1 Before you begin

Please download:

1. jModelTest2:
   `https://drive.google.com/folderview?id=0ByrkKOPtF_n_OUs3d0dNcnJPYXM#list`
2. PAUP*:
   `http://people.sc.fsu.edu/~dswofford/paup_test/`
3. Primate mitochondrial DNA:
   `http://ib.berkeley.edu/courses/ib200/labs/04/primate-mtDNA.nex`

2 Introduction

Maximum likelihood (ML) is a statistical method for reconstructing trees. The likelihood is the probability of the data given a hypothesis $P(D|H)$. In phylogenetics the data is typically a sequence alignment, and the hypothesis includes both the tree topology and the model of character evolution. ML operates by trying to maximize the likelihood value; the tree with the highest likelihood value is considered the best tree. When using ML to build trees, we have to 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.

3 ML model selection using jModelTest

jModelTest [Darriba et al., 2012] is a tool to carry out statistical selection of best-fit models of nucleotide substitution. jModelTest uses the program PhyML [Guindon et al., 2010] to rapidly compute ML trees under a variety of different substitution models, and then provides different model selection strategies (such as AIC, BIC, likelihood-ratio tests) to pick the best fitting model. The "best" model and its parameter values can then be used in programs such as PAUP* or MrBayes to carry out more thorough tree searches.

3.1 Alternative approaches to jModelTest

Another aspect of finding the best fitting model is determining the actual partitioning scheme to use on molecular data, where each partition is assigned a different model of molecular evolution. The program PartitionFinder `http://www.robertlanfear.com/partitionfinder/` is a popular program for selecting both the best-fitting partitioning scheme and best-fitting model of molecular evolution for each partition.

Even if we have picked the "best" model for our data, this model may still not be a very good fit for the data, or there may be a number of models that fit equally as well. We can

deal with uncertainty in model selection by averaging over all possible models during inference instead of conditioning over the single "best" model. This *model averaging* approach is usually carried out in a Bayesian framework, read Huelsenbeck et al. [2004] if you are interested.

3.2 jModelTest instructions

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). 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. Click *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.

4 Models of Nucleotide Evolution

Most nucleotide and amino acid substitution models are in a class of mathematical models called continuous-time Markov chain (CTMC) models. A CTMC consists of a transition rate matrix Q that represents the instantaneous stochastic rates of change between any two states of the model. In phylogenetics the model states are nucleotide or amino acid states. The probability of transitioning between states can be found through the matrix exponential

$$P(t) = e^{Qt}$$

where t is the length (or time) of a branch in a phylogeny. The overall likelihood of the phylogeny can be calculated by plugging these probabilities into Felsenstein's pruning algorithm [Felsenstein, 1981].

4.1 The Transition Rate Matrix

Different models of nucleotide evolution are defined by different transition rate matrices. Here we mean *transition* as in transition from one character state to another, not as in transition/transversion mutations. Below, the initial state of the nucleotide is the first column, and the final state is the top row, however these are not usually shown in the matrix. Using the matrix exponential shown above, one can calculate the probability of one nucleotide changing into another on a branch with a given length. The most unconstrained 12 parameter matrix (which has a different rate for every possible transition) looks like this

$$Q = \begin{pmatrix} & A & C & G & T \\ A & -\alpha-\beta-\gamma & \alpha & \beta & \gamma \\ C & \delta & -\delta-\epsilon-\zeta & \epsilon & \zeta \\ G & \eta & \theta & -\eta-\theta-\iota & \iota \\ T & \kappa & \lambda & \mu & -\kappa-\lambda-\mu \end{pmatrix}$$

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, if the average rate of change of all the off diagonals was normalized to 1, you could eliminate a parameter for a total of 11 parameters.

This is the transition matrix for the Kimura two parameter model [Kimura, 1980]:

$$Q = \begin{pmatrix} -\alpha-2\beta & \beta & \alpha & \beta \\ \beta & -\alpha-2\beta & \beta & \alpha \\ \alpha & \beta & -\alpha-2\beta & \beta \\ \beta & \alpha & \beta & -\alpha-2\beta \end{pmatrix}$$

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

Many 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

$$\pi_i Q_{ij} = \pi_j Q_{ji}$$

where Q_{ij} is the instantaneous rate of change from nucleotide i to nucleotide j and π_i is the equilibrium frequency of nucleotide i. The equilibrium frequency is the frequency of that nucleotide if the substitution process is allowed to run forever, and can be considered another parameter. So the General Time Reversible (GTR) matrix looks like

$$Q = \begin{pmatrix} - & \pi_c r_{ac} & \pi_g r_{ag} & \pi_t r_{at} \\ \pi_a r_{ac} & - & \pi_g r_{cg} & \pi_t r_{ct} \\ \pi_a r_{ag} & \pi_c r_{cg} & - & \pi_t r_{gt} \\ \pi_a r_{at} & \pi_c r_{ct} & \pi_g r_{gt} & - \end{pmatrix}$$

with the diagonal filled in appropriately, and where the terms r_{ij} are called the *exchangeability rates*. 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. Some of these nested models are named:

- 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 [1980] - Kimura two-parameter model; two nucleotide substitutions types are allowed, those between transitions and transversions. Base frequencies are assumed equal.
- HKY85: Hasegawa et al. [1985] - Two nucleotide substitutions types are allowed, those between transitions and transversions. Base frequencies are allowed to vary.

Question 1:
DNA and amino acids are discrete characters. Phylogenetic CTMCs can be used to model all types of discrete character evolution. What would it mean to use the GTR model for a morphological character with 3 character states? Is this an ordered or unordered character state evolution model? What would the parameters be, and how many would there be?

4.2 Commonly used model extensions

All of the above models are often extended to include the proportion of invariable sites I and among-site rate variation Γ.

4.2.1 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 probability, λ_i, that that base pair would have the observed distribution if it could not change. This will be 1 if the base pair 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:

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

4.2.2 Among-site rate variation Γ

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 (a *discretized 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.

5 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.

AIC and BIC are both methods that assess model fit by penalizing complex models (models with more parameters). This is necessary to avoid overfitting statistical models. 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. It is rather unfortunately named – it is not really a Bayesian method as it is independent of the prior and actually only uses the likelihood value (like AIC). It is calculated almost exactly the same as AIC, except it penalizes extra parameters slightly more strongly.

1. Click the *Analysis* menu. Youll notice that you can now select "Do AIC Calculations", "Do BIC Calculations", or "Do DT Calculations"
2. 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*.
3. 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? jModelTest, like most other phylogenetic software uses log-likelihood (lnL) values. What is the log-likelihood value for the selected model?

Perform the BIC calculation (remember to select Write PAUP* block). Was the same model selected as with AICc? What is the log-likelihood value for this model?

What if these two criteria differ in their model selection?

6 Maximum Likelihood (ML) in PAUP*

First let's use the parameter values chosen by jModeltest. In the jModeltest 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 will looks something like:

```
BEGIN PAUP;
Lset base=(0.3585 0.3207 0.0844 ) nst=6  rmat=(2.0810 ... etc
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. Edit your Nexus file and paste the PAUP block from jModeltest directly into it. It can go after any `END;` statement. Execute the newly-edited sequence file in PAUP* again. Set the optimality criteria to likelihood, run a heuristic search, and then write your tree to a file with branch lengths.

```
paup> set criterion=likelihood
paup> hs
paup> savetrees file=aiccmltree.tre brlens
```

Question 3:
Make another tree using the BIC PAUP block. Open the two trees in FigTree and send me a screenshot.

7 CIPRES and RAxML

The CIPRES Science Gateway is a web server that hosts popular phylogenetic research tools.Analyses performed through CIPRES run on the NSF's XSEDE supercomputing infrastructure. Go to the CIPRES at `http://www.phylo.org/` If you have never used CIPRES, you'll need to register. Once you login, click on *Toolkit* to see all the programs available. This will be very useful for your projects!

RAxML (Randomized Axelerated Maximum Likelihood) is a program for ML 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). RAxML also uses a model called GTRCAT that approximates the full GTR+Γ model. 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. But RAxML is great for large datasets. RAxML requires a *phylip* formatted file. Open your nexus file in AliView and select *Save as Phylip (full names & padded).*

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

Back in CIPRES, under the tab *Home* create a new folder for today. When you create the folder, youll notice that two subfolders are created "Data" and "Tasks". Select your *Data* subfolder and upload the phylip file.

Click on the *Tasks* folder and the button *Create New Task*. Give the task a description, and then under the *Select Data* tab select your uploaded phylip file. Under the *Select Tool* tab select **RAxML-HPC2 on XSEDE**. Now click on *Set Parameters*. 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. Set the outgroup as `Lemur_catta`
4. Leave these defaults (explore these when you run your own analyses though)
5. Click *Advanced Parameters* Nucleic Acid Options
6. Configure Bootstrapping - make sure *Conduct Rapid Bootstrapping* is clicked.
7. Make sure *Conduct a rapid Bootstrap analysis and search for the best-scoring ML tree in one single program run. (-f a)* is clicked.
8. Increase the # of bootstrap iterations to 1000
9. Click *Save Parameters* then *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.

Please email me the following:

1. The answers to questions 1-5.
2. Screengrabs of your two PAUP* trees and your RAxML tree.

References

Diego Darriba, Guillermo L Taboada, Ramón Doallo, and David Posada. jmodeltest 2: more models, new heuristics and parallel computing. *Nature methods*, 9(8):772–772, 2012.

Joseph Felsenstein. Evolutionary trees from dna sequences: a maximum likelihood approach. *Journal of molecular evolution*, 17(6):368–376, 1981.

Stéphane Guindon, Jean-François Dufayard, Vincent Lefort, Maria Anisimova, Wim Hordijk, and Olivier Gascuel. New algorithms and methods to estimate maximum-likelihood phylogenies: assessing the performance of phyml 3.0. *Systematic biology*, 59(3):307–321, 2010.

Masami Hasegawa, Hirohisa Kishino, and Taka-aki Yano. Dating of the human-ape splitting by a molecular clock of mitochondrial dna. *Journal of molecular evolution*, 22(2):160–174, 1985.

John P Huelsenbeck, Bret Larget, and Michael E Alfaro. Bayesian phylogenetic model selection using reversible jump markov chain monte carlo. *Molecular Biology and Evolution*, 21(6):1123–1133, 2004.

Thomas H Jukes and Charles R Cantor. Evolution of protein molecules. *Mammalian protein metabolism*, 3:21–132, 1969.

Motoo Kimura. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *Journal of molecular evolution*, 16 (2):111–120, 1980.

Alexandros Stamatakis, Thomas Ludwig, and Harald Meier. Raxml-iii: a fast program for maximum likelihood-based inference of large phylogenetic trees. *Bioinformatics*, 21(4): 456–463, 2005.