**Bioinformatics Lab 7**

**RAxML, MrBayes, and BEAST2**

**Week 8**

**Goal**

To estimate fast ML and Bayesian phylogenies from large datasets using concatenated alignments. Alternatively, to estimate a Bayesian phylogeny from a smaller dataset using a partitioned mitochondrial alignment.

**Links to relevant tutorials / data**

* The RAxML-NG ReadMe:

<https://github.com/amkozlov/raxml-ng/blob/master/README.md>

* BEAST2 tutorial: <https://taming-the-beast.org/tutorials/Introduction-to-BEAST2/>
* The MrBayes website: <https://nbisweden.github.io/MrBayes/download.html> and GitHub page: <https://github.com/NBISweden/MrBayes>

**Operating system**

Mac, Linux, or PC (although RAxML-NG may not work well on a PC). BEAST is a standalone program that should work well on any system, but is more complex to set up an analysis.

**Background**

This lab involves installing and running **RAxML-NG, MrBayes, or BEAST2**.

RAxML-NG is a phylogeny inference program that uses maximum-likelihood (ML) as an optimality criterion. It is one of the fastest ML/Bayesian (versus distance or parsimony) methods out there, making it the preferred choice for very large datasets. It can be used for tree estimation, bootstrapping to assess support for a tree topology, evaluating the likelihood of alternative models of substitution, etc.

MrBayes is a program for Bayesian inference, so uses posterior distributions of model parameters as its optimality criteria. MrBayes provides a flexible framework that can conduct all sorts of analyses, but we will only use it for tree estimation.

BEAST2 is also a program for Bayesian tree inference, especially for partitioned alignments. It provides lots of flexibility in assignment of priors, especially for divergence time data (e.g. using molecular clock assumptions), but as such it is useful only for smaller datasets of just a handful of loci.

I am providing three msa datasets to choose from. The primate.phy is 13,472 base pairs from 12 species of primates and should run quickly on MrBayes or RAxML. The primate-mtDNA.nex file is 898 base pairs of a three mitochondrial genes.

**Steps**

1. ***Preparing the input files***
2. Download the data from Canvas (under Phylogenetics Module). These are alignments in Nexus format (.nex) or Phylip format (.phy), which are similar to the fasta format that we’ve been working with.
3. Choose phylip format for RAxML-NG, Nexus format for MrBayes, and the mtDNA alignment for BEAST2. I suggest using a different folder for each phylogenetics software that you’re running.
4. [Optional: upload your alignment to the Cipres Science Gateway (<https://www.phylo.org/>) and run it on the jModelTest to estimate the best model of sequence evolution to use for your alignment. You will need to make an account, but it’s free. Then upload your msa to a new data folder, and run an analysis, selecting jModelTest as the tool to use.]
5. ***RAxML-NG***
6. If you haven’t already, download RAxML-NG from <https://github.com/amkozlov/raxml-ng>. Unzip and move the unzipped folder containing the binary to an appropriate location. You can then call the binary in terminal using ./raxml-ng from within the containing folder, by specifying the full path from wherever you are (e.g., “/Applications/raxml-ng\_v1.1.0\_macos\_x86\_64/raxml-ng”), or by adding the location to your PATH.
7. Perform a “single tree inference on DNA alignment” (estimate a tree) using RAxML-NG, following the instructions in the Readme.

*Note: If using the Epinecrophylla alignment, this will take at least several hours to run and uses quite a bit of RAM. I recommend testing it quickly in class and then starting it sometime when you aren’t going to need your computer (e.g., at night before you go to bed) so you can leave it running. If the default single tree inference analysis is impractical on your machine, either find a different computer/cluster to run it on, or reduce the number of tree searches performed. By default, RAxML-NG conducts searches on 20 starting trees (10 random, 10 parsimony). You can reduce this using the –tree flag as follows:*

*--tree pars{N},rand{M}*

*Where N and M are the number of searches to run on parsimony and random trees, respectively. Reduce these from the default 10 as needed (although keep in mind you may end up with a poor tree if you do too few searches).*

**Since I am on a Windows computer, RAxML GUI was used instead. 100 bootstraps and the GTR + Gamma model (as per the results of the Modeltest) were used, with all remaining parameters on their default settings.**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

1. View your tree! It can be viewed in the free GUI program FigTree (<http://tree.bio.ed.ac.uk/software/figtree/>), as well as in R using various R packages. Note that FigTree requires a jdk (java development kit): <https://www.oracle.com/java/technologies/downloads/>. *Note: Your tree will look nicer once you re-root it on the outgroup. Find the lemur sample (or the branch leading to the lemur and tarsier) and click “Reroot” in FigTree.*

**“Best scoring ML tree with support values” – Initial tree:**

**A diagram of a tree

Description automatically generated**

**Re-rooted tree on the Tarsius/Lemur clade:**

*A diagram of a number of species

Description automatically generated with medium confidence*

1. [Optional: If you completed step 5 and have time, you may want to complete a bootstrap analysis to evaluate support for your best tree (particularly if you are using your own dataset). To do this, try using the instructions in the Readme (steps 2-4). Alternatively, rather than re-estimate the tree, you can use the existing tree from step 6 using the –support flag.]

**RAxML GUI has the option to include the bootstrap analysis in the tree creation process, which can be seen in the screenshots above.**

1. ***MrBayes***
2. If you haven’t already, install MrBayes. A pre-compiled version is available for Windows on the Installation page of the MrBayes website. If you are on a Mac, the instructions at <https://github.com/NBISweden/MrBayes> worked for me. Here is the MrBayes manual: <https://github.com/NBISweden/MrBayes/blob/develop/doc/manual/Manual_MrBayes_v3.2.pdf>
3. Navigate to your working folder (the one containing the Nexus alignment output in step 4). Start MrBayes. The default install location of MrBayes is generally “/usr/local/bin/”, so try typing:

/usr/local/bin/mb

**Since I am using the Windows executable version of MrBayes, this doesn’t apply. I simply placed the “primate.nexus” file in the same folder as the MrBayes .exe.**

1. Load the data:

execute primate.nexus

A screenshot of a computer

Description automatically generated

1. Set the evolutionary model to GTR with gamma-distributed rate variation and a proportion of invariable sites:

lset nst=6 rates=invgamma

1. Start the MCMC:

mcmc ngen=20000 samplefreq=100 printfreq=100 diagnfreq=1000

A screenshot of a computer

Description automatically generated

1. After the 20,000 generations specified above, you will be prompted to decide if you want to continue the analysis. Typically, you would want to continue it if the value of “standard deviation of split frequencies” printed to the screen is above 0.01. We will probably be able to reach that benchmark in a reasonable amount of time with this dataset, but if not, it is okay to stop the analysis after 20,000 generations for the purposes of this assignment.

A screenshot of a computer

Description automatically generated

1. Once the analysis is stopped, summarize the posterior distribution of parameter values including substitution model parameters and ESS (effective sample size) values. You typically want average ESS values >100 for a publishable analysis.

sump

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Only one of these parameters has an average ESS of over 100, meaning that this analysis would not be publishable.**

1. [Optional: View traces of the posterior parameter distributions (these should end in “.p”) using the program Tracer (<http://tree.bio.ed.ac.uk/software/tracer/>). We will use this program more in the next week or two with the BEAST analysis, but could be fun to get a sneak preview with this first Bayesian analysis.]

**“Run1”:**

**A screenshot of a computer

Description automatically generated**

**“Run2”:**

A screenshot of a computer

Description automatically generated

1. Summarize the posterior distribution of trees (this is printed to a file ending in :

sumt

A screenshot of a computer

Description automatically generated

1. The file produced by the above step includes posterior probability for each node in the tree. As with the RAxML tree, it can be viewed in FigTree or R (see above). It may still be quite unresolved after a short MCMC!

**Initial tree:**

A diagram of a number of species

Description automatically generated with medium confidence

**Re-rooted tree (on the Lemur sample, as it was not possible to re-root it on the Tarsius/Lemur polytomy):**

A diagram of a family tree

Description automatically generated

1. ***BEAST2***
2. If you haven’t already, install BEAST2: <https://www.beast2.org/> This should be as easy as clicking on the appropriate file for your operating system and adding it to your Applications folder.
3. Follow the instructions in the Introduction to BEAST2 tutorial using the primate-mtDNA.nex alignment file: <https://taming-the-beast.org/tutorials/Introduction-to-BEAST2/>.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**Final tree after following the tutorial (Lemur/Tarsius clade was already the outgroup):**

**A diagram of a number of people

Description automatically generated with medium confidence**

**Products**

* + One ML or one Bayesian tree from concatenated alignments, or one Bayesian tree from partitioned alignments, each with 12 tips representing some primate species, including an outgroup. The Bayesian trees might be unresolved (too short an analysis), but the ML tree should be decent.
  + Export a .pdf / .png / .jpg file of your tree to your Bioinformatics folder and push your data to GitHub.

**All trees in their full size (both .pdf and .jpg formats) were pushed to my GitHub.**

* + Write a short Methods and Results text based on the program that you used (RAxML-NG, MrBayes, or BEAST2). Outline 1) what you did, 2) what you found, and 3) why you think it is interesting/relevant. This should be three paragraphs (one for each of the three topics):
  1. **Utilizing the executable version of MrBayes for Windows, I created a Bayesian phylogenetic tree containing 12 primate species. 20,000 Markov Chain Monte Carlo simulations were performed, using the GTR evolutionary model with a gamma-distributed rate of variation and proportion of invariable sites.**
  2. **Although the tree is not publishable (as only one of the average ESS values is above 100), it appears to be very resolved, with all nodes possessing a 1 (100%) probability value with the exception of the most basal node, which does not possess a probability value at all (curiously enough, it also lacks a probability value in the maximum likelihood tree produced by RAxML GUI 2.0 using 100 bootstraps, the GTR + Gamma model as per the Modeltest run, and the remaining settings on their default values). Additionally, the Tarsius and Lemur samples form a polytomy (as opposed to the clade they form in the RAxML tree), indicating that maximum likelihood is the optimal method for this specific dataset.**
  3. **Overall, phylogenetic trees such as the ones produced in this exercise can give us insight into the evolutionary relationships between organisms, when certain traits evolved, when species first arose etc. A phylogenetic tree of primates is also of particular interest to us humans as it can offer these insights regarding our own history and evolution.**