Integrative Biology 200A ***University of California, Berkeley***
"PRINCIPLES OF PHYLOGENETICS" ***Spring 2010***
Updated by Nick Matzke

Lab 6: Introduction to PAUP*

Today we will be learning about some of the basic features of **PAUP*** *(***P***hylogenetic* **A***nalysis* **U***sing* **P***arsimony [*and other methods]), a phylogenetics program developed by David Swofford. Supposedly, it is pronounced, "pop star." PAUP* can infer phylogenies using distance, parsimony and likelihood. Today, we will run these types of analyses using a sample mtDNA data set. We will learn how to use more features in PAUP* in later labs.*

PAUP is not quite the grand-daddy of command-line phylogenetics programs – probably that would be PHYLIP – but PAUP has been widely used for a long time, and many other command-line programs use similar commands to load data, process trees, etc. Even if you decide that TNT or MrBayes or some such is better for your particular phylogenetics problem, PAUP may well have utility functions that you will find useful.

Setup:

- PAUP* is not free, but we have a class license. Install the appropriate version on your laptop from one of the class CDs.
- Once you have it installed, check to make sure that it works by using the command line terminal to navigate to wherever you installed it, and type "paup".
- We may have to troubleshoot a bit to get everyone up and running.
- Download *primate-mtDNA.nex* from the class website, or use your own aligned DNA if you have it.

Assignment:
There are no questions in this lab, just work through it. However, the assignment to email me is:

1. Describe what the bootstrapping procedure is. E.g., not the computational details, but what basically is going on. Imagine you were doing it by hand -- you start with a data matrix, and then what do you to it?
2. Describe the basic difference between parsimony and likelihood methods of estimating a phylogeny.
3. Send me a short update on how your data-gathering is going, including a NEXUS file with some data from your groups in it – DNA, characters, whatever.

(Due next Tuesday)

EXERCISE I: Basic PAUP.

Before we get started, a word about preparing files for use in PAUP. PAUP takes a nexus (.nex) file as input. You can edit these files with a text editor (such as Notepad or Text Wrangler) or with MacClade or Mesquite. It is easy to make nexus files if you know what to put. Ask me if you want to know more about the nexus file format.

- Open PAUP*.
- To load data, type "excecute <filename>". You may have to include the full path for the file, depending on where you saved it.
    - On a Mac, you can drag-and-drop a file into the terminal, and this will paste the text of the full path.
    - On Windows, the Windows Explorer file viewing program will show you the whole path just like a web browser shows you a URL. (You may have to turn this address bar on, via the View menu I think.)
- PAUP will tell you a few things about the file: how many taxa and characters it has in it, and what nucletotide ambiguity codons are being used. For instance, right now "R" is treated as either an A or G for analytical purposes.
- Let's find out a little bit about our file. First, we'll get a brief summary of the character status. First, we're going to look at the character summary. Type:

  ```
  cstatus
  ```

  then press enter. (Menu equivalent: **DATA** menu, **Show character status (Brief summary.)**) The output tells you the current optimality criteria (parsimony, likelihood or distance), the number of characters, the character status, coding, number of parsimony informative characters, etc. These are some of the basic summary statistics of your data.
- Now, we'll look at the pairwise distance between the taxa. Type

  ```
  showdist
  ```

  This is useful because it tells you how many differences there are between any two taxa. Taxa that have 0 distance between them can't be differentiated by PAUP. Since they have many equally parsimonious arrangements, they can make your analysis run for a very long time without getting closer to an optimal tree. So it is good to know in advance how many you are dealing with.

Other commands

Some other commands you might want to try out are:

```
showmatrix
```

which lets you look at the data matrix

```
tstatus
```

which shows you the taxon status

```
Taxon-status summary:
  Original data matrix contains 12 taxa
  No taxa have been deleted
  No taxa have been assigned to the outgroup: outgroup
defaults to first taxon (Lemur catta)
```

```
taxset
```

which can be used to create sets of taxa. It is used as:

```
taxset NewNameForSet = ListOfTaxa
```

Try typing:

```
paup> taxset homs = Homo_sapiens Pan Pongo
paup> taxset other = 5-7 9 10
```

```
charset
```

`charset` does the same thing except for characters. PAUP* automatically includes a few character sets:

`Constant:` all invariant characters
`Gapped:` all characters with a gap for at least one taxon.

`Missambig:` all characters with a gap or ambiguous character for at least one taxon.

`Remainder:` all characters not previously referenced in the command.

`Uninf:` all characters that are constant as well as autapomorphic.

`Pos1:` all characters defined by current CodonPosSet as first positions.

`Pos2:` all characters defined by current CodonPosSet as second positions.

`Pos3:` all characters defined by current CodonPosSet as third positions.

`Noncoding:` all characters defined by current CodonPosSet as non-protein-coding sites.

Try typing:

```
charset firsthalf = 1-400
```

`include and exclude`

can be used to exclude characters from the anaylsis and then put them back in. Try:

```
exclude firsthalf
```

then type `cstatus` to see what's different.

`delete and undelete`

delete and undelete can be used to exclude taxa from the analysis and put them back in. Try:

```
paup> delete homs
Taxon-deletion status changed:
  3 taxa deleted
  Total number of taxa now deleted = 3
  Number of nondeleted taxa = 9
```

`?`

keep in mind that you can always type a ? after a command to see brief help about it:

```
delete ?
```

Before you move on to the rest of the exercises, `include all` and `undelete all` so that your results will reflect all twelve taxa and all 800ish characters…

EXERCISE II: Defining an outgroup.

It is easy to set an outgroup using PAUP. For most of the calculations, it is not important to set an outgroup before you do your analysis – PAUP calculates unrooted trees and does not consider character polarity to be set by the outgroup. To set "Lemur catta" as the outgroup, type:

```
outgroup Lemur_catta /only

Outgroup status changed:
  1 taxon transferred to outgroup
  Total number of taxa now in outgroup = 1
  Number of ingroup taxa = 11
```

This sets the Lemur as the only outgroup. If you don't type `/only` PAUP will add a new outgroup to an existing outgroup. Try it:

```
outgroup gorilla

Outgroup status changed:
  1 taxon transferred to outgroup
  Total number of taxa now in outgroup = 2
  Number of ingroup taxa = 10
```

That's silly. Let's move the gorilla back to the ingroup:

```
ingroup gorilla

Outgroup status changed:
  1 taxon transferred to ingroup
  Total number of taxa now in outgroup = 1
  Number of ingroup taxa = 11
```

And check what happened:

```
tstatus

Taxon-status summary:
  Original data matrix contains 12 taxa
  No taxa have been deleted
  Designated outgroup taxa:
    Lemur catta
```

EXERCISE III: Analysis

Now to do some actual analysis. In order to compare the different methods in PAUP, we'll try a distance analysis, a parsimony analysis, and a maximum liklihood analysis. We'll just use the default settings for today; in other labs we'll learn more about what the settings mean and how to change them.

First, change your working directory to the desktop:

```
paup> cd /Users/labuser/Desktop
```

Distance

Our first analysis will be a distance analysis. PAUP can run multiple types of distance analyses, but for today we'll use the default settings. First, we need to set the optimality criterion to distance:

```
paup> set criterion=distance
  Optimality criterion set to distance.
```

Then, we can start the heuristic search:

```
paup> hs
```

PAUP will now tell you about what it is doing, by reporting which options are set to what, and then after a few moments it will tell you what the results of the search were:

```
Heuristic search completed
Total number of rearrangements tried = 546
Score of best tree(s) found = 1.09002
Number of trees retained = 1
Time used = <1 sec (CPU time = 0.00 sec)
```

This means that PAUP looked at 546 rearrangements, and found one best tree. Now let's look at our tree:

```
paup> showtrees
```

PAUP will show you the one tree that it found, in an ascii display. Then you can use savetrees to write your tree to file:

```
paup> savetrees file=distree.tre
```

A new tree file called "distree.tre" should appear on your desktop.

Maximum Likelihood

Maximum likelihood (ML) is a statistical method for reconstructing trees. We'll discuss ML in lecture. 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 select a model of DNA sequence evolution. Today we'll use the Jukes-Cantor model, which assumes all substitution types are equal.

Set the optimality criteria to likelihood:

```
paup> set criterion=likelihood
  Optimality criterion set to likelihood.
```

Next, we have to specify the DNA substitution model. We will use the Jukes-Cantor model, which has all substitution rates equal and assumes equal base frequencies over time. The way to set this is:

```
lset rates=equal basefreq=equal;
```

(Menu equivalent: First, go to likelihood settings and verify that the substitution model is set to All rates equal ("1 st") Second, go to Base frequency and select Assume equal frequencies. After you have made these changes, select OK.)

ML analyses are notorious for their slow computational speed. Make sure you run a heuristic search!

Now let's look at our tree:

```
paup> showtrees
```

PAUP will show you the one tree that it found, in an ascii display. Then you can use savetrees to write your tree to file:

```
paup> savetrees file=mltree.tre
```

A new tree file called "mltree.tre" should appear on your desktop.

Parsimony

Parsimony is the optimality criterion that minimizes the number of changes on the tree. Like maximum likelihood (and unlike distance methods,) it is a phylogenetic method that distinguishes between symplesiomorphy and synapomorphy. Change the optimality criteria back to Parsimony.

```
paup> set criterion=parsimony
Optimality criterion set to parsimony.
```

Then, run a heuristic search (hs):

```
paup> hs
```

This time PAUP found two trees. In order to see both of them, type:

```
paup> showtrees 1-2
```

To see a consensus tree type:

```
contree;
```

Then save your trees:

```
paup> savetrees file=parstree.tre
```

EXERCISE IV: Estimating support by bootstrapping

You might be interested to know the support for your tree. One measure of support is called the ***Bootstrap.*** *Bootstrapping is a statistical method of resampling the data with replacement. We'll go over this more in class. Bootstrapping provides number on the nodes (0-100%) that correspond to the support. The highest support value is 100, while values below between 50 -70 are usually considered weak. It's important to know that values below 50 aren't shown. In fact, branches below 50 are collapsed and shown as a polytomy.*

```
paup> bootstrap nreps=200 treefile=boot.tre
search=heuristic/ addseq=random;
```

This will save your bootstrap output to a file called boot.tre, which records all the replicate trees produced during the bootstrapping process.

(Menu equivalent: Go to the analysis menu and select Bootstrap/jackknife. Set the number of replicates to 200 and the Type of search to Full heuristic, select Continue, then Search.)

PAUP will show you output with the support for each node shown on an ascii tree.

EXERCISE V: Looking at your tree files

Copy the *primate-mtDNA.nex* file and paste it on the desktop.

- Open Mesquite, then open the copy of primate-mtDNA.nex you placed on the desktop.
- To import the tree files, first go to **Taxa&Trees> Import File with Trees> Include Contents…** then choose one of your tree files. Repeat with your other two tree files, and your boot.tre file.
- Now you can look at all four files and compare them. To do this, open each file in a tree window by going to **Taxa&Trees>New Tree Window…>Stored Trees>** then choosing the treefile you want to see.
- You can repeat this with each file. The bootstrap file has many many trees that were created by resampling the data. The bootstrap tree that PAUP showed you was a consensus tree of this data.

Enjoy your new-found PAUP skills!