

## Exercises for phylogenetic trees

The exercises use the programs in the **Phylip** package (version 3.5) from Joe Felsenstein's web page. There are three ways of assessing the programs

- (1) running the DOS/Windows versions installed on your computer in the Phylip directory,
- (2) running the programs via the web interface at Institut Pasteur (via workshop links page),
- (3) running the programs via the web interface at KVL implemented by Peter Sestoft (via workshop links page),
- (4) logging into a computer at KVL via Secure Shell, and running the programs there in a Unix environment.

Method (2) provides the most friendly user interface (data are pasted into fields on the page or read from local files, and results are presented as web pages). Note that you must supply an e-mail address to which the server sometimes (probably when too busy to process the request of analysis immediately) will send a message that the analysis is finished. Method (3) provides easy access to standard versions of the programs and automatic collection of output, however the program settings are fixed.

Methods (1) and (4) have rather old-fashioned user interfaces, with input files specified at a prompt (unless a file named **infile** is present in the directory from which the program is called), options specified by letters in an option list, and the output directed to a file named **outfile**. As this file is overwritten in every new run of any of the programs, it is a wise policy to rename precious results into another file. For (4), use one of the **deltagerXX** directories as your working directory.

Data files for the exercises are available, (1) in the Phylip directory, (2,3) at the biolinks web page, and (4) in the directory into which you login via Secure Shell (copy them into your working directory).

Documentation of the programs (including description of methods and options) is available as text documents in the Phylip directory, at the Pasteur server and at web pages accessible via the biolinks page. The documentation states that random number seed (initialisers) should be of the form  $4n+1$  (e.g. 1001 or 2065), even if the programs only prompt the user for an odd number; it is advised to comply with the documentation's recommendation.

Problems for exercises 2–4 have been provided by Henrik Christensen, KVL.

### Exercise 1: parsimony and rooted vs. unrooted trees

In this exercise we will use the illustration data set for parsimony,

AAG  
AAA  
GGA  
AGA

to discuss the distinction between rooted and unrooted trees, and to interpret the output of the parsimony analysis. The file name of the dataset is `ex1.dat`. Skip the exercise if you are familiar with rooted and unrooted trees, and with the input and output format for programs in the `Phylip` package.

The file `exph1.dat` is a text (ASCII) file containing lines

```
4 3
seq1      AAG
seq2      AAA
seq3      GGA
seq4      AGA
```

The input format for sequences is evident from the listing: the first line contains the number of sequences and the number of sites (in free format), and the following lines contain a label of exactly 10 characters followed by the letters of the sequences. The sequences must be aligned. Allowed letters are the four bases `A,C,G,T` and a number of other letters corresponding to combinations hereof, of which we will only need `R` (`A` or `G`) and `N` (any of the four bases).

Use program `dnapars`, with its standard settings and additional print switches 1, 4 and 5 turned on. Study the output (file `outfile` with methods (1) and (4)). Convince yourself that the data have been read correctly into the program (guess the meaning of “.” in the display of the sequences).

Draw the corresponding *unrooted* tree by hand. Draw also the unrooted tree corresponding to the optimal tree presented in the lecture, and convince yourself that they are identical. Furthermore, draw by hand all possible (3) unrooted trees with 4 leaves.

Finally, study the distribution of the (minimal) number of substitutions on sites and on nodes. Try to grasp the meaning of the “yes”, “no” and “maybe” labels on the branches in the tree. Be aware that the meaning of “.” is different in this listing.

## Exercise 2: evolution of primates

In this exercise we will use a small real data set with 5 sequences of 846 sites for 5 primates (including humans), which has been studied among others by Brown et al. 1982, *J. Mol. Evol.* 18, 225-239. The sequences have accession numbers V00658, V00659, V00672, V00675 and D38112 in GenBank. For the humane sequence only positions 11094–11989 have been used. The dataset is in file `exph2.dat`, and the top of the file has the form

```
5 846
chimpanzeeAAGCTTCACC GCGCAATTA TCCTCATAAT CGCCCACGGA CTTACATCCT
gibbonxxxxAAGCTTTACA GGTGCAACCG TCCTCATAAT CGCCCACGGA CTAACCTCTT
gorillaexxAAGCTTCACC GCGCAGTTG TTCTTATAAT TGCCCACGGA CTTACATCAT
homosapienAAGCTTCACC GCGCAGTCA TTCTCATAAT CGCCCACGGA CTTACATCCT
orangutangAAGCTTCACC GCGCAACCA CCCTCATGAT TGCCCATGGA CTCACATCCT

CATTATTATT CTGCCTAGCA AACTCAAATT ATGAACGCAC CCACAGTCGC
CCCTGCTATT CTGCCTTGCA AACTCAAAC TCGAACGAAC TCACAGCCGC
CATTATTATT CTGCCTAGCA AACTCAAAC TCGAACGAAC CCACAGCCGC
CATTACTATT CTGCCTAGCA AACTCAAAC TCGAACGCAC TCACAGTCGC
CCCTACTGTT CTGCCTAGCA AACTCAAAC TCGAACGAAC CCACAGCCGC
```

The listing shows further details of the input data format. Columns of blanks (in all sequences) are allowed (for ease of display) and ignored by the programs. Sequences may be split over multiple lines as shown (so-called interleaved data format). One must be careful not to leave blank spaces at the end of lines or in separating lines.

The primary purpose of the exercise is to explore the programs implementing the parsimony, neighbour joining and maximum likelihood algorithms — and to explore the evolutionary tree for primates.

### **A): Parsimony method**

Proceed as in exercise 1 using the `dnapars` program. Draw again the resulting *unrooted* tree by hand. You may also try the `dnapenny` program which implements the branch and bound algorithm known to achieve the overall most parsimonious tree (very slow for large trees).

### **B): Neighbour joining method**

Neighbour joining is based on a distance or dissimilarity matrix which contains the distances between the sequences. Generate a distance matrix with the `dnadist` program, using standard settings. Study the output file, and proceed using it as input to the neighbour joining program `neighbor`. Study the results. Compare with the tree obtained by parsimony — is it the same tree? Try also the UPGMA method, and compare the tree structure and the branch length. Convince yourself that the molecular clock property holds.

Other interesting options to change with this procedure relate to the distance matrix; one may choose different methods and a different ratio between transitions and transversions. Experiment a bit with them, if you like; in particular, try the ML method.

### **C): Maximum likelihood estimation**

Use on either of the web servers the `fastDNAMl` program, otherwise the `dnaml` program. Run the program with standard settings, and compare the results (both tree structure and branch lengths) with those above. Experiment (moderately) with the options `G` and `J` to see if you can get another (better) fit by more extensive search among trees.

### **D): Bootstrapping**

Any of the analyses above can be supplemented by bootstrapping. In the Phylip package this is done in a three-step procedure: (i) generate e.g. 100 bootstrap resamples, (ii) analyse all the resamples using the method of interest, and (iii) summarise the tree information from the 100 analyses. For part (iii), use the `consense` program which computes consensus trees by the majority-rule method. Parts (i) and (ii) may be performed directly at the Pasteur web interface by setting suitable options, whereas in the package one must use the `seqboot` program and input multiple data sets (option `M`) to the program performing the analysis in (ii). As input to the `consense` program one must give the trees generated in step (ii) which can be found in the output file `treefile` (rename it as `infile` for use with the `consense` program).

Try a bootstrap analysis with 100 resamples for at least one of the methods (note that the ML method may be rather slow), and study the output. Locate the bootstrap proportions for the different forks in the majority consensus tree, and examine also the less frequent trees. Discuss the interpretations.

Finally, summarise the results of all the analyses — which may be compared with palaeontologic data, as in Benton, M. J., 1997, *Vertebrate Palaeontology*, Chapman & Hall. Summarise also your experience with the three types of algorithms (data, assumptions and stability of results).

### Exercise 3: primates – continued

The data set from Exercise 2 has in fact at position 555 the symbol **N** (~ any letter) in the orangutang sequence. The interpretation hereof is unclear; it may be an error in the sequence. Discuss how to handle this potential problem, and try out if it leads to changes in the resulting trees.

### Exercise 4: protein sequences for primates

The first part of the primate sequences (“NADH dehydrogenase subunit 4”) have been translated to proteins, giving us 149 proteins in each of the 5 sequences. The first 50 positions of the sequences are

```
chimpanzeeSFTGAIILIIAHGLTSSLLFCLANSNYERTHSRIIILSQGLQTLPLIAF
gibbonxxxxSFTGATVLIIAHGLTSSLLFCLANSNYERTHSRIIILSRGLQALLPLIAF
gorillaexxSFTGAVVLIIAHGLTSSLLFCLANSNYERTHSRIIILSQGLQTLPLIAL
homosapienSFTGAVILIIAHGLTSSLLFCLANSNYERTHSRIIILSQGLQTLPLIAF
orangutangSFTGATLMIHGLTSSLLFCLANSNYERTHSRIIILSQGLQTLPLIAL
```

In the Phylip package there are programs for parsimony analysis and for distance matrix calculation (e.g. based on the Dayhoff PAM matrix) similar to those for DNA sequences. There is no program for maximum likelihood analysis in the Phylip package; the program `protml` has been moved into the Molphy package (see mention at Felsenstein’s page about other phylogeny programs). Analyse the protein data with parsimony and neighbour joining algorithms, and compare the results with those previously obtained (note: compare also the distance matrices). For a discussion of analyses based on DNA vs. protein sequences, see the documentation file for the `protml` program (which is still in the Phylip package).