

First draft (03/2002) of

GASCUEL O., "Getting a Tree Fast: Neighbor Joining and Distance Based Methods", in Current Protocols in Bioinformatics, A. Baxevanis, D. Davison, C. Hogue, R. Page, L. Stein, G. Stormo (Eds), Wiley, 6.3.1-6.3.18, 2003.

For more see the final version.

Equipe Méthodes et Algorithmes pour la Bioinformatique

L.I.R.M.M., 161 rue Ada, 34392 - Montpellier Cedex 5 - FRANCE

Tel. (33) 467 41 85 47 - Fax (33) 467 41 85 00 - gascuel@lirmm.fr

<http://www.lirmm.fr/~w3ifa/MAAS/>

Abstract

Neighbor Joining (NJ) and other distance based approaches: BIONJ, WEIGHBOR and (to some extent) FITCH, are fast methods to build phylogenetic trees. This makes them particularly effective for large-scale studies or for bootstrap applications which require runs on multiple data sets (Unit 6.6). Like maximum likelihood methods (Unit 6.5), distance methods are based on a sequence evolution model that is used to estimate the distance matrix. Computer simulations indicate that the topological accuracy of BIONJ, WEIGHBOR and FITCH is significantly better than that of NJ, and that best distance based methods are equivalent to parsimony (Unit 6.3) in most cases, but become more accurate when the molecular clock is strongly violated or in the presence of long (e.g. outgroup) branches.

Introduction

Distance methods and especially Saitou and Nei's (1987) Neighbor Joining (NJ) are popular methods to reconstruct phylogenies from alignments of DNA or protein sequences (Unit 2.4). They are fast, allowing hundreds and even thousands of taxa to be dealt with by ordinary computers. Their speed greatly simplifies the use of the bootstrap procedure, which assess the confidence level of inferred clades (Unit 6.6). They provide a simple way to incorporate knowledge on the evolution of the sequences being studied, depending on how the distance matrix is estimated. Numerous simulation studies have demonstrated their topological accuracy, and they are not hampered by inconsistency (or "Felsenstein") zones as parsimony approaches. The popularity of NJ, among the numerous existing distance based methods, is explained by its speed and by the fact that it is only slightly outperformed by recent approaches, namely FITCH (Felsenstein 1997), BIONJ (Gascuel 1997a) and WEIGHBOR (Bruno, Succi and Halpern, 2000).

NJ and other current distance methods do not assume a molecular clock, as opposed to UPGMA (Sokal and Michener 1958) which is precluded for most phylogenetic studies. The basic assumption is that sequences have been evolving along a tree and independently among the lineages. This tree can differ of of the species tree in case of horizontal transfer or sequence duplication (Unit 6.1). Other assumptions are related to the sequence evolution model used to estimate distances. Models applicable to distance methods are homogeneous (i.e. constant over time) and assume that each site in the sequence evolves independently. However, some model parameters can differ from site to site. For example, mutation rates can vary across sites to represent structural/functional constraints on the residues, or the fast rate of the third codon position.

Distance methods are thus "model based", just like maximum likelihood methods. However, the way the computations are performed is much more simple and approximate.

So they are much faster than maximum likelihood methods, but do not achieve the same topological accuracy. The comparison with parsimony is more complicated, since parsimony is sometimes inconsistent, but accurate when no long (e.g. outgroup) branch tends to attract other branches and perturb the resulting tree. A good practical approach is then to avoid parsimony in these cases, and otherwise to run both parsimony and distance approaches and compare the results.

Application of any distance based method usually requires the following steps:

- (a) Choose a sequence evolution model and use it to estimate the distance matrix.
- (b) Run the tree building algorithm and eventually return to step (a), for example to check that the resulting tree is not too sensitive to the model parameter values. The influence of taxon sampling, notably the presence/absence of the outgroup taxa, has also to be checked.
- (c) Perform the bootstrap procedure to assess the significance level of the inferred clades.

I Basic Protocol: using NEIGHBOR from the PHYLIP package

A. Protocol Introduction

We describe the use of NEIGHBOR and other programs included in the PHYLIP 3.6 package, which is distributed by Joe Felsenstein (University of Washington) and is one of the most widely used softwares in phylogeny studies. Distance estimation is performed using DNADIST or PROTDIST. To achieve the bootstrap procedure, we first resample the sites using SEQBOOT, then apply DNADIST or PROTDIST, run NEIGHBOR, and extract the bootstrap tree using CONSENSE. Finally, the resulting tree can be drawn using a program such as TREEVIEW (Unit 6.2) or NJPLOT (Perrière and Gouy 1996).

B. Necessary Resources

i. Hardware

PHYLIP executables are available for pre-386 DOS, 386/486/Pentium DOS, Windows 3.1, Windows95/98/NT, 68k Macintosh, or PowerMac. It is also available in C source code to be compiled on Unix, Linux or VMS systems.

ii. Software

PHYLIP is free from <http://evolution.genetics.washington.edu/phylip.html>. The package contains C source codes, documentation files, and a number of different types of executables. Its web page contains information on PHYLIP and ways to transfer the executables, source code and documentation. The documentation is remarkably clear and complete, and provides a number of useful references.

iii. Data files

Sequence alignments, as obtained from multiple alignment programs (Unit 2.4), must be given to DNADIST or PROTDIST in the PHYLIP format:

- § The first line contains the number of taxa and sites; next come the taxon data with a new line per taxon.
- § Taxon names have ten-characters and must be blank-filled to be of that length.
- § The taxon names are followed by the sequences, which must either be “interleaved” or “sequential” (Figures 6.4.1 and 6.4.2). The sequences can have internal blanks in the sequence but there must be no extra blanks at the end of the terminated line. With DNA (or RNA) sequences, the three symbols N, X and ? indicate an unknown nucleotide while – indicates a deletion. With proteins, X, - and ? indicate an unknown amino acid, a deletion, and an unknown including deletion, respectively (see PHYLIP documentation *sequence.html* for more details).
- § In the case of multiple data sets, as provided by SEQBOOT, pseudo-alignments are given in the same format one after the other, without omitting the number of taxa and the number of sites at the beginning of each new set.

DNADIST or PRODIST compute a distance matrix from a sequence alignment. The file contains the number of taxa on its first line. Each taxon starts a new line with the taxon name, followed by the distance to the other taxa, and there is a new line after every nine distances. The default format is square (Figure 6.4.3) with zero distances on the diagonal.

Inferred trees are unrooted and written in Newick format. For example, the tree of Figure 6.4.4 is made of three subtrees, containing (Candida_tr, Candida_al and Saccharomy), (Taphrina_d and Protomyces) and (Athelia_bo, Spongipell and Filobasidi), respectively, as can be shown from its TREEVIEW representation (Figure

6.4.5). Each subtrees is made of two subtrees or taxa, and the numbers indicate the branch lengths or the bootstrap supports.

C. Stepwise procedure

i. Distance matrix estimation from DNA (or RNA) sequences using DNADIST

DNADIST first asks for the sequence file, and then for the matrix file. The default files are *infile* and *outputfile*, respectively, but we strongly recommend redefining these files to avoid deleting previously computed files. Then the menu of Figure 6.4.6 appears, which asks for important and sensitive choices. We only describe options requiring in depth explanations and where the default values have often to be changed. More details are given in the DNADIST documentation.

§ **D** defines the substitution model. All models assume that sites evolve independently.

The four available models are nested: Jukes-Cantor is a special case of Kimura, which is a special case of F84, which is a special case of LogDet. Jukes-Cantor assumes only one substitution rate, Kimura allows for a difference between transition and transversion rates, while F84 is similar to Kimura but allows for different frequencies of the four nucleotides, and LogDet does not impose any restriction on the 16 rates (except those induced by the Markovian nature of the process). So LogDet is the most flexible model, but is often overparametrized, unless the sequences are very long (say > 5000). F84 (the default option) is a good compromise, notably when the base frequencies are not equal. When they are almost equal, Kimura is a good choice, while Jukes-Cantor is oversimple in most cases.

Note that all sites (informative or not) must be given to DNADIST for these models to be used in the correct way.

§ **G** asks whether or not the substitution rates vary across sites. Biologically speaking, the answer is clearly yes. It has been demonstrated that the Gamma distribution, which is defined by a parameter usually denoted as α , is a good model to account for this variability. And α was estimated between 0.05 and 1.0 for numerous data sets (Yang 1996), which indicates that rates strongly vary across sites (variability increases as α decreases). However, the default option of DNADIST is to not correct for this variability (i.e. $\alpha = \infty$), which is a common practice. Jin and Nei (1990) recommend using $\alpha = 1.0$ or 2.0 . And we recently demonstrated (Guindon and Gascuel 2002), that uncorrected distances are often better suited, especially when the molecular clock is more or less satisfied. So a pragmatic approach is to use the default option, and to check whether or not using a reasonable value (e.g. 1.0 or 2.0) for α changes the result. A software program to estimate the most appropriate value of α is also available via our web page.

However, DNADIST does not use the standard α parameter, but rather the “coefficient of variation” (CV) that is equal to $1/\alpha^2$. We have CV=4.0, 1.0 and 0.25, when $\alpha=0.5$, 1.0 and 2.0, respectively. Moreover, the LogDet model cannot be combined with the gamma correction.

§ **T** asks for the transition/transversion ratio. The default value is 2.0, and there is no way to estimate this value within PHYLIP. Hopefully, the results are not very sensitive to the value of this parameter (unless extreme). And it is possible to estimate it using simple formulae from Kimura (1980).

§ **C** allows user-defined categories, for example to specify that third position bases have a different rate than first and second positions. This option allows the user to make up to 9 categories of sites, but, as for the LogDet model, using too many categories can make the model overparametrized. The user is asked for the relative

rates within each category. The assignment of rates to sites is then made by reading a file whose default name is "categories". An example and more details are given in the DNADIST documentation. There is no program from PHYLIP for estimating the different rates, but just as for the above ratio these parameters are not very sensitive (unless extreme).

§ **W** allows to select subsets of sites. Basically it has to remain “No” (the default value), unless the user wants to check the influence of various categories of sites.

§ **M** has to be used in the bootstrap procedure (see below). The user is then asked for the number of pseudo-alignments in the input file. Otherwise the default value (“No”) is required.

Once all options have been determined, the user types “Y” and the distance matrix is computed. With our working example of Figure 6.4.1 and all default values, DNADIST returns the matrix of Figure 6.4.3.

ii. Distance matrix estimation from proteins using PROTDIST

PROTDIST is analogous to DNADIST. We first have to provide the file names, and then to deal with the options. The main option is **P** that selects among four substitution models, which differ depending on the matrix of substitution rates:

§ The Dayhoff PAM 001 matrix. This matrix from Dayhoff et al. (1979) is an empirical one that scales probabilities of change from one amino acid to another, assuming that the total change between the two amino acid sequences is 1%. It allows the evolutionary distance to be computed in terms of expected fraction of amino acids changed.

§ The Jones-Taylor-Thornton model (1992). This is analogous to PAM, but the estimation of the probabilities of change was based on a much larger set of proteins. Thus it is to be preferred over the the original PAM.

- § Kimura's (1983) distance. This assumes only one substitution rate, and does not take into account which amino acids differ.
- § The Categories distance (due to Joe Felsenstein). The model is conceptually close to Kimura's (1980) two-parameter model for DNA sequences. The amino acids are grouped into a series of categories, and we distinguish between the transition (change within a category) and transversions (change from one category to another). When this option is selected, the user is asked for a number of other options (e.g. the amino acid categorization), but we suggest using default values that approximate the PAM model.

As already stated, the Jones-Taylor-Thornton model is to be preferred over PAM in any situation. But both induce heavy computations, and the same holds for the Categories model. So the Kimura model is a good option for large data sets or atypical (e.g. membrane) proteins. For the other options, see our comments on DNADIST.

iii. Tree construction using NEIGHBOR

NEIGHBOR is the PHYLIP implementation of Saitou and Nei's (1987) Neighbor Joining. It first asks for the matrix file, and then proposes to rename the "outfile". Once done, the user has to select among numerous options, which a priori have to be used with their default values. Finally, NEIGHBOR asks for the "outtree" file, which has to be renamed carefully because it contains the tree in Newick format. The resulting tree can be visualized in the outfile, but a better view is obtained by applying TREEVIEW to the outtree file.

Applying NEIGHBOR to the matrix of Figure 6.4.3, we obtain the tree (in Newick format):

```
(Candida_tr:0.01367,(Saccharomy:0.03307,((Protomyces:0.00957,Taphrina_d:0.01633):0.06809,(Filobasidi:0.05464,(Spongipell:0.01908,Athelia_bo:0.06002):0.04361):0.10745):0.03164):0.05098,Candida_a
```

1:0.00873); with a TREEVIEW representation equivalent to that of Figure 6.4.5 (which was obtained using BioNJ).

iii. Bootstrapping using SEQBOOT and CONSENSE

A tree such as that shown in Figure 6.4.5 does not indicate the reliability of the inferred clades. The bootstrap procedure (Unit 6.6) is a sound and accurate way to obtain this, and its use is greatly facilitated by the speed of distance methods. Within PHYLIP, the bootstrap procedure is achieved in four steps:

1. We first create pseudo-alignments from the sequences using SEQBOOT. To obtain more reliable results, the **R** option, which corresponds to the number of replicates, has to be changed from 100 (the default value) to 1000 (or more in large studies). SEQBOOT allows for site categories and weights (see DNADIST). The default value has to be conserved for the other options. However, we suggest switching the 2 option to avoid displaying the (extensive and useless) “progress of run” on the terminal.
2. We apply DNADIST to the pseudo-alignment file to obtain the pseudo-matrices. It has to be used as above, except that the number of data sets (replicates) must be given using the **M** option. Switching the 2 option is also relevant.
3. We apply NEIGHBOR to the pseudo-matrix file, indicating the number of matrices with the **M** option, and switching the 2 option.
4. Finally, we obtain the bootstrap tree by applying CONSENSE to the pseudo-tree file using all default options. As NEIGHBOR, CONSENSE returns “outfile” and “treefile”, or the corresponding files as renamed by the user. Outfile can be used to visualize the bootstrap tree, but a better view is obtained by applying TREEVIEW to outtree.

When applying these four steps (with 1000 replicates) to the original alignment of Figure 6.4.1, we obtain the bootstrap tree of Figure 6.4.6. The branch-lengths correspond to the bootstrap proportions, which are explicitly shown in the case of internal branches. Note that due to the random nature of the process, bootstrap proportions can differ slightly from one run to another.

II Alternative protocol: computing NJ trees using CLUSTAL

A. Protocol Introduction

This protocol describes the use of CLUSTAL (see Unit 2.4) to build neighbor joining (NJ) trees. Although CLUSTAL is not intended primarily as a tree building program, it is a useful tool for quickly getting a tree for a set of sequences. On the other hand, it does not provide the user with all possibilities of PHYLIP, notably concerning distance estimation. The program is available in two versions: CLUSTALX (Thompson, 1997), which has a graphical interface, and CLUSTALW, which has a text-based interface. CLUSTALW can be used interactively through a simple menu system or from the command line, which makes it a useful tool for batch processing alignments or generating phylogenies as part of a CGI script. This protocol will provide instructions for both the graphical interface of CLUSTALX, and the CLUSTALW command line.

B. Necessary Resources

i. Hardware

CLUSTAL can be run on Macintosh, Windows, and Unix systems. For full details see Unit 2.4.

ii. Software

CLUSTALX or CLUSTALW.

iii. Datafiles

See Unit 2.4 for the input formats.

CLUSTAL can output trees in a variety of formats. The default format is the Newick format used by many phylogenetic programs (see Unit 6.2 and above, 6.4.I).

CLUSTAL can also write trees in its own format, and can save the pairwise distances in PHYLIP format.

C. Procedure

i. Building a NJ tree

Before building a tree there are various options the user can set that control how the pairwise distances between sequences are computed, and the output format for the tree. In CLUSTALX these options are set using the commands on the Trees menu (Figure 6.4.8), in CLUSTALW they are set on the command line.

- § When choosing the “Exclude positions with gaps” command, any site with a gap in any sequences will be ignored when computing pairwise distances. A priori, this command should be chosen, because distance estimation from sequences with gaps does not have sound mathematical foundations. But removing all sites with a gap sometimes makes the phylogenetic signal so low that the resulting tree is no longer supported in the bootstrap procedure. So both approaches should be tested.

The command line equivalent is `/TOSSGAPS`.

- § If “Correct for multiple substitutions” is toggled on (indicated by a tick beside the menu command), then CLUSTALX will use either the Kimura 2-parameter model (Kimura, 1980) or Kimura's (1983) correction for nucleotides and proteins, respectively, to compute pairwise distances between sequences. This should be the default option. If this option is not chosen, then there is no correction for multiple substitutions.

The command line equivalent in CLUSTALW is `/KIMURA`

§ The default output tree format is the Newick (or “PHYLIP”) format. The command line is then: `/OUTPUTTREE=phylip`. But CLUSTAL can also write trees in its own format (`/OUTPUTTREE=nj`), and can save the pairwise distances in PHYLIP format (`/OUTPUTTREE=dist`).

Having decided on the options for the analysis and output (or having simply taken the defaults), the command “Draw N-J Tree” will construct the tree. In CLUSTALX the user is presented with a dialog box asking for confirmation of the output tree file name. Typically, the tree file is given the name of your sequence file plus the extension “.phb”. Click on “OK” to construct the tree. However, the “Draw N-J Tree” command is somewhat oddly named, as it doesn't “draw” the tree. To see the tree, you will need to use a tree drawing program such as TREEVIEW (Unit 6.2). The command line equivalent for an NJ tree using the default settings is:

```
clustalw/INFILE=your-aligned-sequence-file/TREE.
```

While to use the Kimura correction, and ignore all sites with gaps, the command line is:

```
clustalw/INFILE=your-aligned-sequence-file/TREE/KIMURA/TOSSGAPS.
```

ii. The bootstrap procedure

In addition to the options that affect tree construction above, there are additional options relevant to bootstrapping.

§ CLUSTAL stores the bootstrap values in the tree description inside square brackets, either as “branch labels” or as “node labels.” The alternative placements are controlled by the “Output Tree Format Options” command (Figure 6.4.8). As discussed in Unit 6.2, there is little consensus on how to store bootstrap values in tree descriptions. Widely used programs such as TREEVIEW (Unit 6.2) do not recognise bootstrap values stored as branch labels, and so in order to display these values in

TREEVIEW the bootstrap values must be placed on the nodes (command line equivalent: /BOOTLABELS=node).

Having decided on the options for the bootstrap analysis and output, the command “Bootstrap N-J Tree” will perform the bootstrapping. In CLUSTALX the user is presented with a dialog box asking for a “seed” for the random number generator used to create the bootstrap pseudoreplicates, the number of pseudoreplicates (“trials”) to generate (the default is 1000), and the name of the file to which the bootstrap tree will be written (typically the tree file is given the name of your sequence file plus the extension “.phb”) (Figure 6.4.8). Click on the “OK” button to perform the bootstrapping. The command line equivalent is: `clustalw/INFILE=your-aligned-sequence-file/BOOTSTRAP.`

III Alternative protocol: using BIONJ, WEIGHBOR or FITCH

A. Protocol Introduction

We provide the description of BIONJ and WEIGHBOR, which are PHYLIP compatible, and of FITCH that is available in PHYLIP. These three programs have a better topological accuracy than NEIGHBOR, and thus they are to be preferred over this one. However, the resulting trees are often close or identical to NEIGHBOR trees, at least with a low number of taxa. When this number increases, the various methods tend to return different trees, and their advantage over NEIGHBOR increases. BIONJ is about the same speed as NEIGHBOR, WEIGHBOR is about 400 times slower than NEIGHBOR, and FITCH still slower than WEIGHBOR (see below for more details). We do not describe the matrix distance computation and the bootstrap procedure, which are identical as with NEIGHBOR (6.4.I).

B. Necessary Resources

i. Hardware

BIONJ executables are available for Windows and Apple MacIntosh and in C source code. WEIGHBOR is available in C source code and has to be compiled on your own system. FITCH is available in PHYLIP and runs on numerous systems (see 6.4.I).

ii. Software

BIONJ is available free from <http://www.lirmm.fr/~w3ifa/MAAS/>. This web page contains documentation and articles, test sets, executables for Windows PC and PowerMac, and the C source code.

WEIGHBOR is available free from <http://www.t10.lanl.gov/billb/neighbor/>. This web page contains documentation, the seminal article, and the C source code. FITCH is included in PHYLIP (see 6.4.I)

iii. Datafiles

The file formats are identical to those described above (6.4.I).

C. Procedure

i. Using BIONJ

BIONJ asks for the distance matrix input file and the name of the tree output file. The distance matrix must be square and written in PHYLIP format. The file can contain one or several matrices, as obtained when using SEQBOOT plus DNADIST, but the user is not asked for the number of matrices. Then BIONJ returns as many trees as there are matrices. These trees are written in Newick format. In case of a single matrix, the resulting tree can be viewed using TREEVIEW, while with multiple matrices and trees, we have to use CONSENSE, just as with NEIGHBOR.

Applying BIONJ to the matrix of Figure 6.4.3, we obtain the tree of Figure 6.4.4 with TREEVIEW representation as shown in Figure 6.4.5. This tree differs from the NEIGHBOR tree (see 6.4.I) by the branch lengths but not by the topology, which is not surprising due to the low number of taxa.

ii Using WEIGHBOR

Just like BIONJ, WEIGHBOR asks for the input and output files, and the input file can contain one or several matrices. Then WEIGHBOR asks for the sequence length and the number of symbols, i.e. 4 for DNA or RNA sequences and 20 for proteins.

iii. Using FITCH

The menu of FITCH is analogous to that of NEIGHBOR. All options have a priori to conserve their default value, except **G** and **J** that can be used to search the tree space more intensively (at the expense of longer run times). **G** can be switched to “Yes” to search for global rearrangements that improve the least-squares fit of the tree. **J** takes advantage of the fact that FITCH does not systematically find the same tree, depending on the taxon ordering. When **J** is switched to **Yes**, FITCH asks for a seed to initiate the random ordering procedure, and then for the number of times the randomization procedure has to be used. The resulting tree is the best tree that is obtained from all random orderings. The higher their number the better the solution, but the longer the computing time. A value of 10 seems to be a reasonable compromise, but is too high for large data sets for which the **J** option has to be switched off.

IV Result interpretation

Phylogenetic trees reconstructed by distance methods do not fundamentally differ from trees reconstructed by any other approach (see Unit 6.1). The main specificity is related to branch lengths. NJ and BIONJ can provide negative branch length estimates, which have to be seen as null. Such negative values do not indicate any sort of “reverse evolution”. Null (or close to zero) branches indicate an irresolution of the tree, which may correspond to a multifurcation, but more likely reflects the weakness of the phylogenetic signal.

The strength of the inferred branches is measured by the bootstrap procedure. Short branches are generally poorly supported, but with distance based approaches it may happen that long branches also have a low support. So the bootstrap procedure must be used, which is done at low computation cost due to the speed of these approaches. The interpretation of bootstrap supports is a difficult question (see Unit 6.6), but any branch with a support lower than 50% should be considered as an irresolution (Berry and Gascuel 1996).

However, in some cases wrong inferences can have high bootstrap support. For example, when very long sequences are used (as is the case when several genes are combined within the same study), bootstrapping the data does not change the resulting tree, which may be partly erroneous. The stability of the tree then has to be tested by other approaches. Notably, the tree must be robust with respect to the presence/absence of the outgroup that possibly attracts some ingroup taxa, to model parameter variations, and to gene sampling when several genes are combined.

V Commentary

A. Background Information

i. The rationale of distance based approaches

Let S be the set of sequences being studied and T the true evolutionary tree of these sequences. Assume that the sequences have been correctly aligned, so that the sites correspond to homologous positions (see Unit 2.1 and 2.4 ?). Now consider the true number of substitutions that is attached to every branch of T , *i.e.* the number of substitutions that occurred in the past from the sequence situated at one branch extremity to the sequence at the other extremity. These substitution numbers are unknown but well defined. They induce the evolutionary distance between any pair of taxa, as the sum of the substitution numbers attached to the path separating both taxa in T . In other words, the evolutionary distance between any pair of taxa is equal to the number of substitutions from one sequence to the other. And, for mathematical reasons first discovered by Zaretskii (1965), there is an equivalence between the so defined distance and T . Knowing T and the substitution numbers per branch allows the computation of the pairwise distances between taxa. And, more importantly, the true tree T and the substitution numbers per branch can be reconstructed from the matrix D of pairwise evolutionary distances.

Obviously, and unfortunately, the true number of substitutions that separates any pair of taxa is unknown. Due to hidden (parallel or convergent) mutation events, the true number of substitutions is always greater than or equal to the number of observed differences between both sequences. When the number of differences is small, both quantities are close. But the gap increases when the evolutionary distance increases. So the distance based approach involves estimating the evolutionary distance from the observed

differences, assuming a stochastic model of sequence evolution. The simplest model, Jukes and Cantor's, supposes that all sites evolve independently and identically according to a Markovian process that is defined by a unique parameter representing the instantaneous probability of change from one nucleotide to another. This model establishes a mathematical relationship between the evolutionary distance (now defined as the ratio between the true number of substitutions and the sequence length) and the proportion of observed differences (Figure 6.4.7). More realistic models have been proposed, such as those described above (6.4.I), but the basic principle remains identical. We first compute an estimate \hat{D} of D , and then reconstruct an estimate \hat{T} of T using \hat{D} . And the accuracy of \hat{T} increases with the reliability of \hat{D} .

The estimated evolutionary distance matrix \hat{D} no longer exactly fits a tree, but is usually very close to a tree. For example, our working data set of Figure 6.4.1 has been extracted from TreeBASE (<http://www.treebase.org/treebase/index.html>) and corresponds to 67 Fungi sequences (accession #M520). Using DNADIST and NEIGHBOR with default options, we find a tree that explains more than 98% of the variance in the distance matrix. Then the resulting tree and the distance matrix are extremely close, so the mere principle of the distance approach appears to be well founded in this case (and in most cases).

Even when the estimated distance matrix is usually very close to a tree, tree reconstruction from such approximate matrix is much less obvious than in the ideal case where the matrix perfectly fits a tree. Various methods have been proposed, which differ by the criterion they optimize and by their tree building strategy. For all known criteria, the optimisation task is NP-hard (*i.e.* can require exponential computing time) so all practical methods are heuristic and do not guarantee that the best tree will be found. However, due to the closeness between the distance matrix and a tree, all (reasonable) methods usually find similar trees that are fairly accurate estimates of the true tree.

ii. Neighbor Joining algorithm

Neighbor Joining (NJ) is derived from ADDTREE (Sattath and Tverski 1977). It was proposed by Saitou and Nei (1987) and studied in depth by several authors (Studier and Keppler 1988; Rzhetsky and Nei 1993; Atteson 1997; Gascuel 1997b).

NJ is an agglomerative algorithm. At each step, it uses the distance matrix $\hat{D} = (\delta_{ij})$ where i and j are either taxa or clusters of taxa agglomerated during previous steps. Based on these distances, two taxa are selected to be merged. Denoting r as the number of “taxa” in \hat{D} , and Q_{ij} as the criterion value for the agglomeration of i and j , the pair agglomerated is the one minimizing

$$Q_{ij} = (r-2)\delta_{ij} - \Delta_i - \Delta_j \text{ where } \Delta_x = \sum_{y=1}^r \delta_{xy}. \quad (1)$$

Once the pair i, j to agglomerate is selected, NJ creates a new node u which represents the root of the new cluster. Then NJ estimates the branch lengths δ_{iu} and δ_{ju} and reduces the distance matrix by replacing the distances relative to taxa i and j by those between the new node u and any other node x using

$$\delta_{ux} = \frac{1}{2}(\delta_{ix} - \delta_{iu}) + \frac{1}{2}(\delta_{jx} - \delta_{ju}). \quad (2)$$

The process stops when $r = 2$, with the last branch length being equal to the last value in the distance matrix. The successive mergings achieved by NEIGHBOR are available in its outfile.

The Q criterion enables numerous interpretations, the most popular being that it corresponds to the least-squares length estimate of the tree under construction. Accordingly, NJ tends to produce a tree with minimal length. But more importantly, when applied to any tree distance D that perfectly fits a tree T , Q designates with certainty a pair of neighbors of T . This induces the statistical consistency of NJ, which is an essential property of phylogeny reconstruction methods: NJ recovers the true tree T with certainty, as soon as \hat{D} is sufficiently close to the true evolutionary distance matrix D .

iii. The BIONJ algorithm

The BIONJ algorithm (Gascuel 1997a) is a variant of NJ. It is based on the fact that NJ remains consistent when formula (2) is replaced by:

$$\delta_{ux} = \lambda_{ij}(\delta_{ix} - \delta_{iu}) + (1 - \lambda_{ij})(\delta_{jx} - \delta_{ju}), \quad (3)$$

where λ_{ij} is any number in $[0,1]$ that varies depending on the merged pair i, j but not on x . So once the pair i, j has been selected, BIONJ computes the value λ_{ij}^* that minimizes the sum of the variances of the δ_{ux} estimates. In this way, more reliable estimates will be available to select the pairs of taxa to be agglomerated during the next steps. Moreover, since the process is repeated at each step, these estimates will become better and better in comparison with NJ estimates as the algorithm proceeds.

To achieve this, BIONJ uses a simple first-order model of variances and covariances of evolutionary distance estimates obtained from sequences. This model indicates that the variance of any distance estimate δ_{xy} is approximately proportional to δ_{xy} , while the covariance of δ_{xy} and δ_{zt} is roughly proportional to the length of the intersection of paths (x, y) and (z, t) in the true tree T (Nei and Jin 1989; Bulmer 1991). This yields the formula:

$$\lambda_{ij}^* = \frac{1}{2} + \varphi,$$

where φ is a correction term that depends on δ_{iu} and δ_{ju} (at least when i and j are original taxa). When δ_{iu} and δ_{ju} are equal, then $\varphi = 0$, $\lambda_{ij}^* = 1/2$, and BIONJ is equivalent to NJ. When both differ, *i.e.* when the substitution rates vary among lineages, φ becomes not null and places more confidence on the shorter and hence more reliable distance. So BIONJ has a clear advantage over NJ when the molecular clock is markedly violated, while both methods are close in the opposite case.

IV WEIGHBOR

WEIGHBOR follows the same agglomerative scheme as NJ. It modifies the reduction step, in a way analogous to BIONJ, but also the selection step to take into account the high variance of long distance estimates. Instead of using criterion (2), WEIGHBOR combines two criteria. When i and j are neighbors in T and when \hat{D} perfectly fits T , then we have the two following properties:

Additivity: $\delta_{ik} - \delta_{jk}$ is independent of k ($\neq i, j$),

Positivity: $\delta_{ik} + \delta_{jl} - \delta_{ij} - \delta_{kl} \geq 0$ for any k, l ($\neq i, j$).

Since \hat{D} is imperfect, these properties are only approximately satisfied, and we have to find the pair i and j that fit them best. To achieve this, WEIGHBOR assumes that distance estimates are mutually independent and have Gaussian distribution with variance as induced by the Jukes and Cantor model. Within this model, the variance of the distance estimate is proportional to the distance around 0 (as in the BIONJ model), but increases exponentially when the distance becomes larger. This model allows to compute the likelihood that i and j are neighbors. Considering the above defined additivity, we have the following criterion (to be minimized):

$$Additivity(i, j) = \sum_{k \neq i, j} \frac{\left(\delta_{ik} - \delta_{jk} - \overline{(\delta_{ik} - \delta_{jk})} \right)^2}{Var(\delta_{ik}) + Var(\delta_{jk})},$$

where the bar denotes the average over k ($\neq i, j$). A similar criterion corresponds to the positivity property. *Additivity* is used to indicate the best pairs, which are finally selected using *Positivity*. This approach, which fully takes into account the high variance of long evolutionary distances, makes WEIGHBOR more resistant than NJ and BIONJ to the influence (attraction or distraction) of long branches.

V FITCH

FITCH is the implementation (Felsenstein 1997) of the basic principles described in the seminal paper of Fitch and Margoliash (1967). Its algorithmic strategy is not agglomerative but additive. FITCH constructs a tree by iteratively adding taxa to a growing tree. And at each step, it performs tree swapping to improve the goodness-of-fit, using nearest-neighbor interchange (i.e. exchange of subtrees separated by 3 branches). Finally, once a first tree has been constructed, it optionally (see above) performs a more extensive search in the tree space by considering global rearrangements: every subtree is removed from the tree and put back on in all possible ways so as to have a better chance of finding a better tree. The resulting tree may be sensitive to the initial taxon ordering, even when the swapping procedures tend to lower its influence. So the jumbling procedure (6.4.II) must be used, unless computational time constraints.

FITCH optimizes the weighted least-squares criterion. Let (δ_{ij}) be the matrix of distance estimates and (\hat{t}_{ij}) the distance matrix induced by the inferred tree \hat{T} and its branch lengths. The weighted least-squares fitting of \hat{T} is defined by:

$$WLS(\hat{T}) = \sum_{i \neq j} \frac{1}{VAR[\delta_{ij}]} (\hat{t}_{ij} - \delta_{ij})^2, \quad (4)$$

where $VAR[\delta_{ij}]$ is the variance of the δ_{ij} estimate. This criterion has to be minimized, and has value 0 when \hat{T} perfectly represents (δ_{ij}) . Various solutions are possible for the variance of δ_{ij} , which may be written as $VAR[\delta_{ij}] = \delta_{ij}^p$. When the power p is null, all variances are equal to 1.0 and the model is close to that of NJ. When $p = 1$, the variance of δ_{ij} is equal to δ_{ij} , and the model is equivalent to that of BIONJ without the covariance terms. But the best results are obtained with $p = 2$, which corresponds to the solution of Fitch and Margoliash (1967) and is quite close to the WEIGHBOR model. This is the default option of FITCH.

Criterion (4) not only concerns the topology of \hat{T} , but also its branch-lengths. Minimizing this criterion induces branch length estimates which have to be positive for the approach to be consistent. This is one other (to be conserved) default option of FITCH.

VI Method comparison

Numerous computer simulations have been performed to compare the topological accuracy of phylogeny reconstruction methods. The principle is: a) consider a “true tree”, b) evolve an initial random sequence along this tree to obtain “contemporary sequences”, c) reconstruct a tree from these sequences, d) finally, compare the inferred tree to the true tree. Drawing definitive conclusions from such a study is difficult because the results depend on the true tree, on the evolutionary conditions, and on numerous parameters. Moreover, most available studies have considered a low number of true trees and few taxa (usually 12 or less).

We recently tried to overcome these limits by randomly generating a large (5000) number of trees, with a realistic (40) number of taxa, under a broad variety of evolutionary conditions (maximum pairwise divergence uniformly drawn from $[0.1,1.1]$ and molecular clock varying from full satisfaction to strong violation). These data sets were used to compare the four methods discussed above, DNAPARS (a parsimony approach from the PHYLIP package, see Unit 6.4) and FASTDNAML (a maximum likelihood approach due to Olsen et al. (1994), see Unit 6.5). An article about this study is in preparation (joint work with Stephane Guindon). We summarize the main conclusions below.

Table 1 displays the average results. It appears that NJ is outperformed by BIONJ, which is outperformed by WEIGHBOR and FITCH that are equivalent. Moreover, DNAPARS is equivalent to WEIGHBOR and FITCH, while FASDNAML is clearly the best method. The ordering of distance methods is stable, we did not find any evolutionary condition where it is different. And the first position of FASTDNAML is also stable, it

outperforms the other methods in all conditions. But the position of DNAPARS is less stable, it performs well with the molecular clock and in the absence of long outgroup branches, but its performance is less good in the opposite conditions where it is not better than BIONJ.

	Topological accuracy	Run time
NJ	10.95%	0.005
BIONJ	10.58%	0.006
WEIGHBOR	9.96%	2.0
FITCH	10.08%	15.0
DNAPARS	9.97%	0.5
FASTDNAML	7.89%	230.0

Table 1: Simulation results with 5000 randomly generated 40-taxon trees. The topological accuracy is measured by the proportion of wrong branches in the inferred tree. The run times are given in seconds and correspond to the average time required to infer one of these 40-taxon trees, with a PC - Pentium 4 - 1.7 Ghz.

The contrast between methods in Table 1 can be seen as not very high, even when significant. The contrast between run times (also displayed in Table 1) is much more impressive. NJ, BIONJ and WEIGHBOR have computational time proportional to the third power of the number of taxa, but WEIGHBOR performs much more calculations and is about 400 times slower than NJ and BIONJ. Therefore WEIGHBOR is limited to few hundreds of taxa, while NJ and BIONJ can be used with thousands of taxa. FITCH has computational time proportional to the fourth power of the number of taxa, so it is limited to 100 taxa or less, especially when a bootstrap study is envisaged. DNAPARS is about 100

times slower than NJ and BIONJ, but a bit faster than WEIGHBOR, while FASTDNAML is so slow that its use is reserved to in depth studies with not many taxa.

The users are thus confronted to a compromise. They can obtain better trees using maximum-likelihood methods, but at the expense of long computing times when the number of taxa is high. The advantage of distance based approaches is their speed. NJ, or preferably BIONJ, combined with the bootstrap procedure, allows to rapidly obtain quite reliable phylogenetic trees together with the support of the inferred branches.

B. Critical Parameters/Troubleshooting

Distance based approaches are sensitive to the way evolutionary distances are estimated. When the sequences exhibit few differences, all sequence evolution models become equivalent, and the model choice is not crucial. For example, when two sequences have 0.1 sites that differ with 0.07 transitions and 0.03 transversions, the Jukes and Cantor estimate is equal to 0.1073, the Kimura two-parameter estimate to 0.1086 and the Jin and Nei estimate (with $\alpha=1.0$) to 0.1183. But the model choice becomes very sensitive when the maximum pairwise divergence among the sequences being studied becomes higher. Now considering two sequences with half of the sites being different with 0.35 transitions and 0.15 transversions, the Jukes and Cantor estimate is equal to 0.824, the Kimura two-parameter estimate to 1.037 and the Jin and Nei estimate ($\alpha=1.0$) to 2.940. So data sets with too high sequence divergence (say > 1.0) must be considered as suspicious and should be discarded. Note that the presence of such high divergence makes the alignment itself very difficult and subject to errors. With more reasonable maximum divergence, the stability of the results for model variations is a positive point. Moreover, the presence of distant outgroup taxa is a perturbation factor in all reconstruction steps (alignment, distance estimation and tree building) and should be avoided, at least in a first analysis.

C. Suggestions for Further Analysis

As can be seen from Table 1, maximum likelihood approaches (Unit 6.5) clearly outperform other methods. So with small data sets they should be a first choice when results obtained using distance methods are unsatisfactory; for example, when most branches have low bootstrap supports. With large data sets, a possibility is to first carry out an in depth study on small taxon subsets using maximum likelihood, notably concerning the sequence evolution model and its parameters, and then to use the findings in a distance approach.

Parsimony approaches do not outperform distance methods (see Table 1), but their principle is so different that finding the same tree using both is generally considered to be a strong support for that tree.

Distance methods are available in numerous phylogeny software packages. Notably, PAUP (release 4.0b10) provides very fast versions of NJ, FITCH and BIONJ, as well as a larger than DNADIST and PROTDIST variety of evolutionary distance estimates.

Finally, a new distance method (Desper and Gascuel, 2002), which combines the speed of NEIGHBOR and the topological accuracy of FITCH, is now available from author's URL (<http://www.lirmm.fr/~w3ifa/MAAS/>).

D. Internet Resources

An extensive list of phylogeny softwares, including numerous distance based methods, is available from Joe Felsenstein's web page:

<http://evolution.genetics.washington.edu/phylip/software.html>

E. References

- ATTESON, K. 1997. The Performance of the NJ Method of Phylogeny Reconstruction. Pages 133-148 *in* *Mathematical Hierarchies and Biology* (B. Mirkin, F.R. McMorris, F.S. Roberts and A. Rzhetsky, eds.). American Mathematical Society, Providence.
- BERRY V., and O. GASCUEL. 1996. Interpretation of bootstrap trees : threshold of clade selection and induced gain. *Mol. Biol. and Evol.* 13:999-1011.
- BRUNO, W.J., N.D. SOCCI, AND A.L. HALPERN. 2000. Weighted Neighbor Joining: A Likelihood-Based Approach to Distance-Based Phylogeny Reconstruction. *Mol Biol Evol* 17:189-197.
- BULMER, M. 1991. Use of the Method of Generalized Least Squares in reconstructing Phylogenies from Sequence Data. *Mol. Biol. Evol.* 8:868-883.
- DAYHOFF, M. O., R. M. SCHWARTZ, and B. C. ORCUTT. 1979. A model for evolutionary change in proteins. In *Atlas of Protein Sequence and Structure* (Dayhoff, M. O., ed.), 5:345-352.
- DESPER, R., and O. GASCUEL. 2002. Fast and Accurate Phylogeny Reconstruction Algorithms Based on the Minimum-Evolution Principle. *Journal of Computational Biology* 9(5):687-705.
- FELSENSTEIN, J. 1989. PHYLIP – Phylogeny Inference Package (Version 3.2). *Cladistics* 5:164–166.
- FELSENSTEIN, J. 1997. An alternating least-squares approach to inferring phylogenies from pairwise distances. *Syst. Biol.* 46:101-111.
- FITCH, W.M., and E. MARGOLIASH. 1967. Construction of phylogenetic trees. *Science* 155: 279-284.
- GASCUEL, O. 1997a. BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data. *Mol. Biol. Evol.* 14:685-695.
- GASCUEL, O. 1997b. Concerning the NJ algorithm and its unweighted version, UNJ. Pages 149-170 *in* *Mathematical Hierarchies and Biology* (B. Mirkin, F.R. McMorris, F.S. Roberts and A. Rzhetsky, eds.). American Mathematical Society, Providence.

- GUINDON, S., and O. GASCUEL. 2002. Efficient biased estimation of evolutionary distances when substitution rates vary across sites. *Mol. Biol. Evol.* 19:534-543.
- JIN, L., and M. NEI. 1990. Limitations of the evolutionary parsimony method of phylogenetic analysis. *Mol. Biol. Evol.* 7:82-102.
- JONES, D.T., W.R. TAYLOR, and J.M. THORNTON. 1992. The rapid generation of mutation data matrices from protein sequences. *Comput Appl Biosci.* 8:275-82.
- KIMURA, M. 1980. A simple method for estimating evolutionary rates of base substitutions through comparative studies of nucleotide sequences. *J. Mol. Evol.* 16:111-120.
- KIMURA, M. 1983. *The Neutral Theory of Molecular Evolution*. Cambridge University Press, Cambridge, England.
- NEI, M., and L. JIN. 1989. Variances of the Average Numbers of Nucleotide Substitutions within and between Populations. *Mol. Biol. Evol.* 6:290-300.
- OLSEN, G. J., H. MATSUDA, R. HAGSTROM, and R. OVERBEEK. 1994. FastDNAmL: a tool for construction of phylogenetic trees of DNA sequences using maximum likelihood. *Comput. Appl. Biosci.* 10: 41-8.
- PERRIÈRE, G. and M. GOUY. 1996. WWW-Query: An on-line retrieval system for biological sequence banks. *Biochimie* 78:364-369.
- RZHETSKY, A., AND M. NEI. 1993. Theoretical Foundation of the Minimum-Evolution Method of Phylogenetic Inference. *Mol. Biol. Evol.* 10:1073-1095.
- SAITOU N., AND M. NEI. 1987. The neighbor-joining method: a new method for reconstruction of phylogenetic trees. *Mol. Biol. Evol.* 4:406-425.
- SATTATH, S., and A. TVERSKY. 1977. Additive similarity trees. *Psychometrika* 42: 319-345.
- SOKAL, R. R., and MICHENER, C. D. 1958. A Statistical Method for Evaluating Systematic Relationships," *University of Kansas Science Bulletin*, 38, 1409-1438.
- STUDIER, J. A., and K. J. KEPPLER. 1988. A note on the neighbor-joining algorithm of Saitou and Nei. *Mol. Biol. Evol.* 5: 729-31.
- SWOFFORD, D.L., G.L. OLSEN, P.J. WADDELL, and D.M. HILLIS. 1996. Phylogenetic inference. Pages 407-514 *in* *Molecular Systematics* (D.M. Hillis, C. Moritz and B.K. Mable, eds.). Sinauer, Sunderland, MA.

- THOMPSON, J. D., GIBSON, T. J., PLEWNIAK, F., JEANMOUGIN, F., and HIGGINS, D. G. 1997. The ClustalX windows interface: flexible strategies for multiple sequence alignment aided by quality analysis tools. *Nucleic Acids Res.* 25: 4876-4882.
- YANG, Z. 1996. Among-site rate variation and its impact on phylogenetic analyses. *TREE* 11:367–372.
- ZARESTKII, K. 1965. Reconstructing a tree from the distances between its leaves. *Uspehi Matematicheskikh Nauk* 20:90-92 (in Russian).

Figure 6.3.1 Flowchart

8

Candida_al	0.0000	0.0939	0.0224	0.1737	0.1632	0.2507	0.2757	0.3050
Saccharomy	0.0939	0.0000	0.0966	0.1434	0.1582	0.2381	0.2064	0.2614
Candida_tr	0.0224	0.0966	0.0000	0.1791	0.1632	0.2591	0.2855	0.3160
Protomyces	0.1737	0.1434	0.1791	0.0000	0.0259	0.2235	0.2232	0.2820
Taphrina_d	0.1632	0.1582	0.1632	0.0259	0.0000	0.2585	0.2581	0.3318
Filobasidi	0.2507	0.2381	0.2591	0.2235	0.2585	0.0000	0.1386	0.1370
Spongipell	0.2757	0.2064	0.2855	0.2232	0.2581	0.1386	0.0000	0.0791
Athelia_bo	0.3050	0.2614	0.3160	0.2820	0.3318	0.1370	0.0791	0.0000

Figure 6.3.2: Distance matrix in square format.

```

C:\PHYLIP\exe\neighbor.exe
Please enter a new file name> test-matrix
neighbor.exe: the file "outfile" that you wanted to
use as output file already exists.
Do you want to Replace it, Append to it,
write to a new File, or Quit?
<please type R, A, F, or Q>
f
Please enter a new file name> test-outfile

Neighbor-Joining/UPGMA method version 3.6a2.1
Settings for this run:
N Neighbor-joining or UPGMA tree? Neighbor-joining
O Outgroup root? No, use as outgroup species 1
L Lower-triangular data matrix? No
R Upper-triangular data matrix? No
S Subreplicates? No
J Randomize input order of species? No. Use input order
M Analyze multiple data sets? No
0 Terminal type (IBM PC, ANSI, none)? <none>
1 Print out the data at start of run No
2 Print indications of progress of run Yes
3 Print out tree Yes
4 Write out trees onto tree file? Yes

Y to accept these or type the letter for one to change
y
neighbor.exe: the file "outtree" that you wanted to
use as output tree file already exists.
Do you want to Replace it, Append to it,
write to a new File, or Quit?
<please type R, A, F, or Q>
f
Please enter a new file name> test-outtree

```

Figure 6.3.3: The NEIGHBOR screen showing options for renaming files as well as options for settings and their defaults.

```
(( (Candida_tr:0.0137,Candida_al:0.0086):0.0526,Saccharomy:0.0316):0.0351,
 (Taphrina_d:0.0160,Protomyces:0.0098):0.0665,
 ((Athelia_bo:0.0600,Spongipell:0.0190):0.0480,Filobasidi:0.0612):0.0964);

(Candida_tr:0.01367,(Saccharomy:0.03307,((Protomyces:0.00957,
Taphrina_d:0.01633):0.06809,(Filobasidi:0.05464,(Spongipell:0.01908,
Athelia_bo:0.06002):0.04361):0.10745):0.03164):0.05098,Candida_al:0.00873);
```

Figure 6.3.4: Two trees in Newick format, which were obtained from the distance matrix in Fig. 6.3.2 by BIONJ and NEIGHBOR, respectively. Both trees have identical topologies, but slightly different branch lengths.

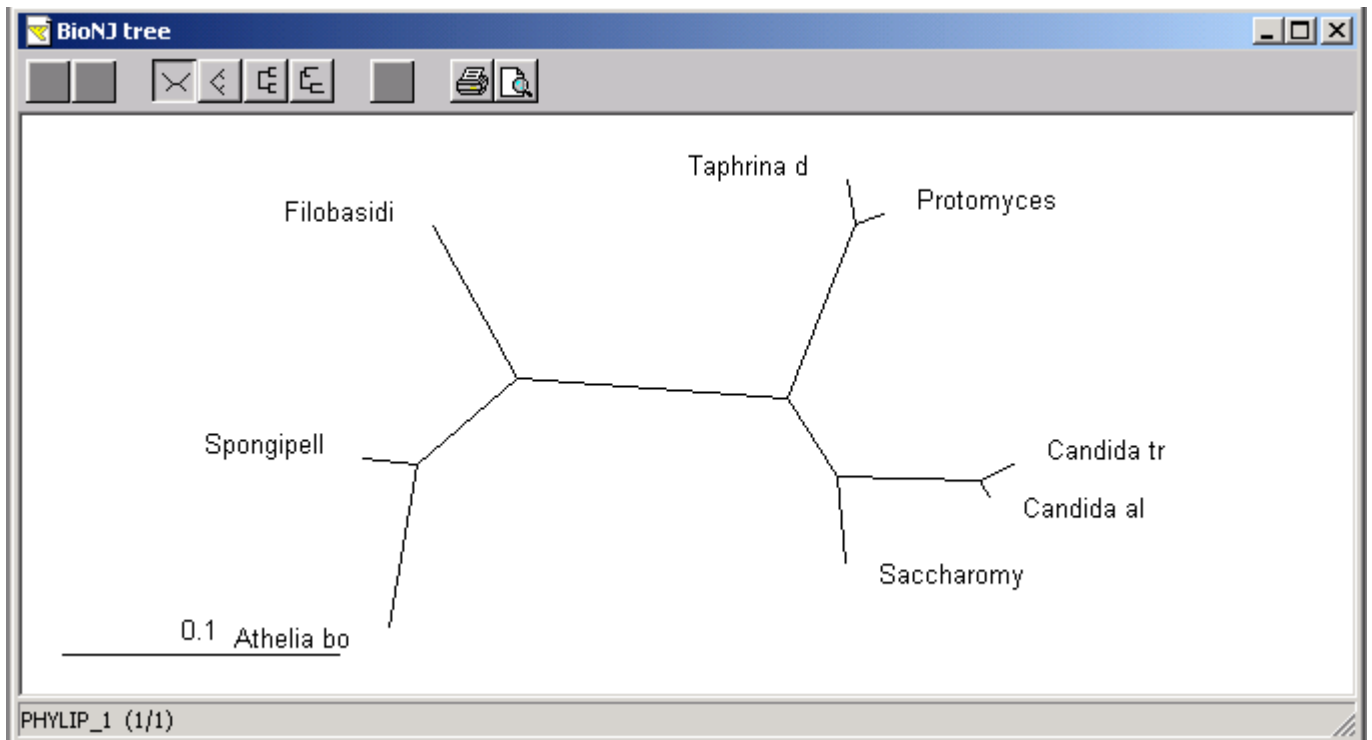


Figure 6.3.5: TREEVIEW representation of the BIONJ tree of Figure 6.3.4.

```

+Candida_tr
!
! +-Saccharomy
! !
4--5      +Protomyces
! ! +---3
! ! !    +Taphrina_d
! +-6
!      !      +---Filobasidi
!      +-----2
!              ! +Spongipell
!              +---1
!              +---Athelia_bo
!
+Candida_al

```

Figure 6.3.6: NEIGHBOR tree, as represented in the outfile.

```

8 95
Candida_al      AGTCTTAACCATAAACTATGCCGACTAGGGATCGGTTGTTGTTCTTTTATT-GACGCAAT
Saccharomy     AGTCTTAACCATAAACTATGCCGACTAGGGATCGGGTGGTGTFTTTTTTAAT-GACCCACT
Candida_tr     AGTCTTAACCATAAACTATGCCGACTAGGGATCGGTTGTTGTTCTTTTATT-GACGCAAT
Protomyces     AGTCTTAACCATAAACTATGCCGACTAGGGATCGGGCGATGTTCTTTTCTT-GACTCGCC
Taphrina_d     AGTCTTAACCATAAACTATGCCGACTAGGGATCGGGCGATGTTCTTTTCTT-GACTCGCC
Filobasidi     AGTCTTAACAGTAAACGATGCCGACTAGGGATCGGCCACGTCAATCTCT--GACTGGGT
Spongipell     AGTCTTAACAGTAAACTATGCCGACTAGGGATCGGGCGATCTCAAACCT-ATGTGTCGCT
Athelia_bo     AGTCTTAACAGTAAACTATGCCGACTAGGGATCGGACAACCTCAATTTTGATGTGTTGTT

CGGCACCTTACGAGAAATCA-AAGTCTTTGGGCCC
CGGCACCTTACGAGAAATCA-AAGTCTTTGGGTTC
CGGCACCTTACGAGAAATCA-AAGTCTTTGGGG?
CGGCACCTTATGAGAAAGGA-AAGTTTTTGGGTTC
CGGCACCTTATGAGAAAAA????????????????
CGGCACCTTACGAGAAATCA-AAGTCTTTGGGTTC
CGGCACCTTACGAGAAATCA-AAGTCTTTGGGTTC
CGGCACCTTACGAGAAATCA-AAGTCTTTGGGTTC

```

Figure 6.3.7: Alignment in interleaved PHYLIP format.

```

8 95
Candida_al AGTCTTAACCATAAACTATGCCGACTAGGGATCGGTTGTTGTTCTTTTATT-GACGCAAT
CGGCACCTTACGAGAAATCA-AAGTCTTTGGGCC
Saccharomy AGTCTTAACCATAAACTATGCCGACTAGGGATCGGGTGGTGTTTTTTTAAT-GACCCACT
CGGCACCTTACGAGAAATCA-AAGTCTTTGGGTTC
Candida_tr AGTCTTAACCATAAACTATGCCGACTAGGGATCGGTTGTTGTTCTTTTATT-GACGCAAT
CGGCACCTTACGAGAAATCA-AAGTCTTTGGGGG?
Protomyces AGTCTTAACCATAAACTATGCCGACTAGGGATCGGGCGATGTTCTTTTCTT-GACTCGCC
CGGCACCTTATGAGAAAGGA-AAGTTTTTGGGTTC
Taphrina_d AGTCTTAACCATAAACTATGCCGACTAGGGATCGGGCGATGTTCTTTTCTT-GACTCGCC
CGGCACCTTATGAGAAAAA??????????????
Filobasidi AGTCTTAACAGTAAACGATGCCGACTAGGGATCGGGCCACGTCAATCTCT--GACTGGGT
CGGCACCTTACGAGAAATCA-AAGTCTTTGGGTTC
Spongipell AGTCTTAACAGTAAACTATGCCGACTAGGGATCGGGCGATCTCAAACCTT-ATGTGTCGCT
CGGCACCTTACGAGAAATCA-AAGTCTTTGGGTTC
Athelia_bo AGTCTTAACAGTAAACTATGCCGACTAGGGATCGGACAACCTCAATTTTGATGTGTTGTT
CGGCACCTTACGAGAAATCA-AAGTCTTTGGGTTC

```

Figure 6.3.8: Alignment in sequential PHYLIP format.

```

C:\PHYLIP\exe\dnadist.exe
dnadist.exe: can't find input file "infile"
Please enter a new file name> test-DNA

dnadist.exe: the file "outfile" that you wanted to
use as output file already exists.
Do you want to Replace it, Append to it,
write to a new File, or Quit?
<please type R, A, F, or Q>
f
Please enter a new file name> test-matrix

Nucleic acid sequence Distance Matrix program, version 3.6a2.1

Settings for this run:
D Distance (F84, Kimura, Jukes-Cantor, LogDet)? F84
G Gamma distributed rates across sites? No
I Transition/transversion ratio? 2.0
C One category of substitution rates? Yes
W Use weights for sites? No
F Use empirical base frequencies? Yes
L Form of distance matrix? Square
M Analyze multiple data sets? No
I Input sequences interleaved? Yes
0 Terminal type (IBM PC, ANSI, none)? <none>
1 Print out the data at start of run No
2 Print indications of progress of run Yes

Y to accept these or type the letter for one to change

```

Figure 6.3.9: The DNADIST screen showing options for renaming files as well as options for settings and their defaults.

```
C:\PHYLIP\exe\protdist.exe
protdist.exe: can't find input file "infile"
Please enter a new file name> test-PROT

protdist.exe: the file "outfile" that you wanted to
use as output file already exists.
Do you want to Replace it, Append to it,
write to a new File, or Quit?
<please type R, A, F, or Q>
f
Please enter a new file name> test-matrix

Protein distance algorithm, version 3.6a2.1
Settings for this run:
P Use JTT, PAM, Kimura or categories model? Jones-Taylor-Thornton matrix
G Gamma distribution of rates among positions? No
C One category of substitution rates? Yes
W Use weights for positions? No
M Analyze multiple data sets? No
I Input sequences interleaved? Yes
Ø Terminal type (IBM PC, ANSI)? (none)
1 Print out the data at start of run No
2 Print indications of progress of run Yes

Are these settings correct? <type Y or the letter for one to change>
■
```

Figure 6.3.10: The PROTDIST screen showing options for renaming files as well as options for settings and their defaults.

```

C:\PHYLIP\exe\seqboot.exe
seqboot.exe: can't find input file "infile"
Please enter a new file name> test-BOOT

Bootstrapping algorithm, version 3.6a2.1

Settings for this run:
  D Sequence, Morph, Rest., Gene Freqs? Molecular sequences
  J Bootstrap, Jackknife, Permute, Rewrite? Bootstrap
  B Block size for block-bootstrapping? 1 (regular bootstrap)
  R How many replicates? 100
  W Read weights of characters? No
  C Read categories of sites? No
  F Write out data sets or just weights? Data sets
  I Input sequences interleaved? Yes
  0 Terminal type (IBM PC, ANSI, none)? (none)
  1 Print out the data at start of run No
  2 Print indications of progress of run Yes

  Y to accept these or type the letter for one to change
y
Random number seed (must be odd)?
19

seqboot.exe: the file "outfile" that you wanted to
use as output data file already exists.
Do you want to Replace it, Append to it,
write to a new File, or Quit?
(please type R, A, F, or Q)
f
Please enter a new file name> Pseudo-alignments

```

Figure 6.4.11: The SEQBOOT screen showing options for renaming files as well as options for settings and their defaults.

```
C:\PHYLIP\exe\consense.exe
consense.exe: can't find input tree file "intree"
Please enter a new file name> pseudo-trees

consense.exe: the file "outfile" that you wanted to
use as output file already exists.
Do you want to Replace it, Append to it,
write to a new File, or Quit?
<please type R, A, F, or Q>
f
Please enter a new file name> bootstrap-file

Consensus tree program, version 3.6a2.1

Settings for this run:
C      Consensus type (MRe, strict, MR, M1):  Majority rule (extended)
O      Outgroup root:                          No, use as outgroup species  1
R      Trees to be treated as Rooted:          No
T      Terminal type (IBM PC, ANSI, none):     <none>
1      Print out the sets of species:          Yes
2      Print indications of progress of run:   Yes
3      Print out tree:                         Yes
4      Write out trees onto tree file:         Yes

Are these settings correct? <type Y or the letter for one to change>
y

consense.exe: the file "outtree" that you wanted to
use as output tree file already exists.
Do you want to Replace it, Append to it,
write to a new File, or Quit?
<please type R, A, F, or Q>
f
Please enter a new file name> bootstrap-tree
```

Figure 6.4.12: The CONSENSE screen showing options for renaming files as well as options for settings and their defaults.

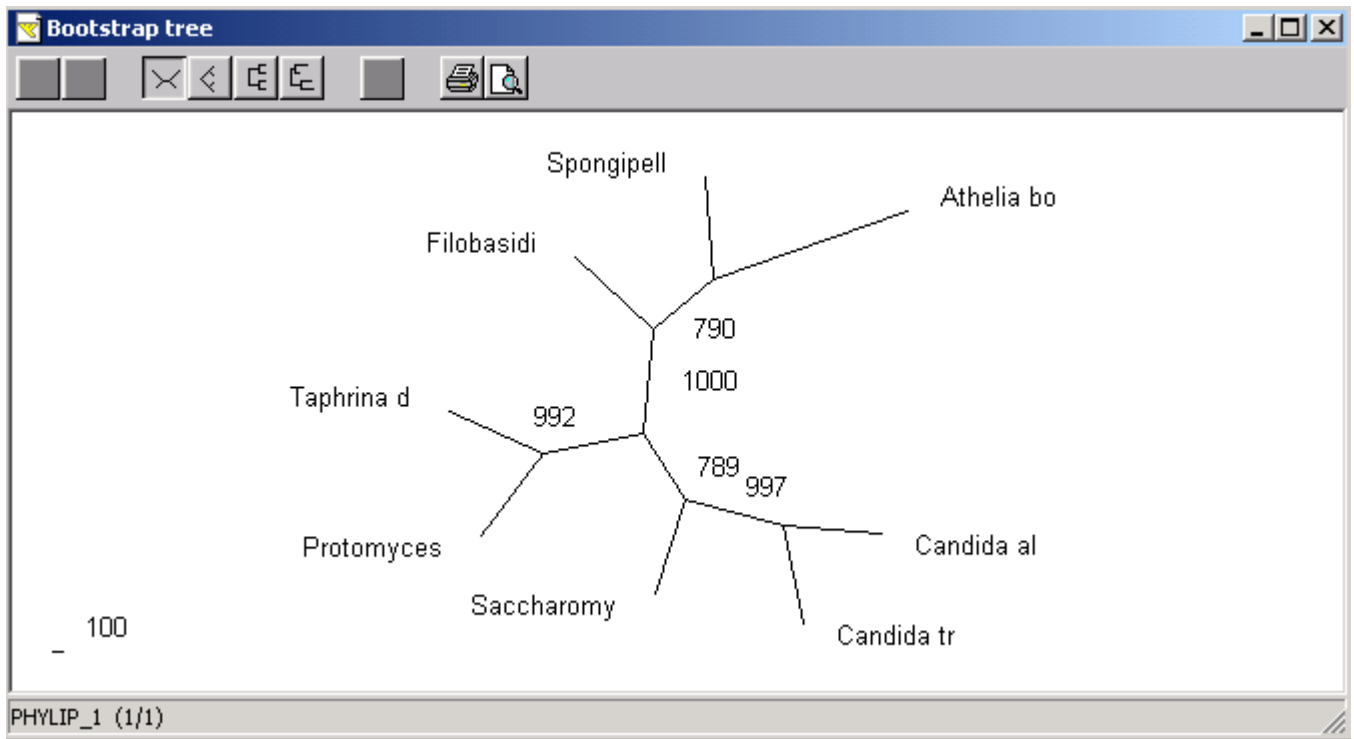


Figure 6.4.13: TREVIEW representation of the bootstrap tree that is obtained with NEIGHBOR.

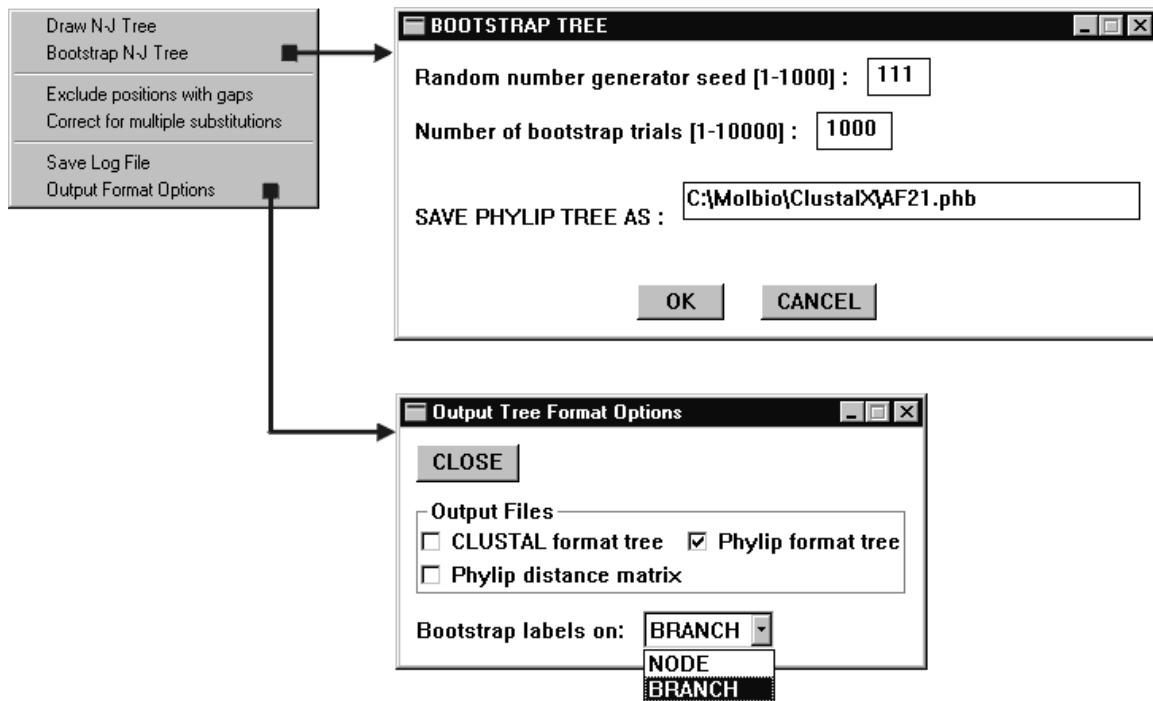


Figure 6.4.14: The Trees menu in the program ClustalX showing the menu commands and dialog boxes used to control how the program constructs neighbor joining trees.