PAUP *

Phylogenetic Analysis Using Parsimony
* and other methods

Version 4.0 beta version

David L. Swofford

Beta Documentation

This documentation is currently under construction, therefore, we cannot guarantee the accuracy of the information described herein. A much more user-friendly manual is in progress. Revisions will be made available to registered users over the World Wide Web at **http://paup.csit.fsu.edu**. A list of frequently asked questions is also available at this website. Many of the commands and options described in this document are hyper-linked to facilitate and promote on-line use until a final draft is complete. Comments regarding this document should be addressed to paup-support@csit.fsu.edu.

PAUP* 4.0 beta version disclaimer and user agreement

The version of PAUP* 4.0 currently available is a beta version. We are not aware of bugs that would cause the program to obtain incorrect results, but they could certainly exist. Other bugs could cause the program to terminate abnormally or misbehave in other ways. Your use of this software implies your acceptance of its unfinished status. We would greatly appreciate receiving reports of any problems you encounter. Send them by email to: paup-support@csit.fsu.edu. Please include all details necessary to reproduce the problem, including: (1) type of machine and operating system version; (2) specific version number of PAUP* (e.g., PAUP*4.0b8); (3) a specific sequence of operations that will cause the problem to occur, beginning with start-up of the program; and (4) a sample data file and any other associated files that will enable the problem to be reproduced.

The following license agreement and disclaimer apply both to the beta copies and to the final release version:

PAUP* is licensed to individual users for the sole purpose of facilitating the scientific research of the licensee. This software may be used in more than one location or by more than one person provided that there is **no possibility** that it will be used by two or more people simultaneously. Generally, this qualification means either of the following: (1) the program may be used by a **single researcher** on any number of machines in his or her possession, or (2) the program may be installed on a **single computer** and used by one or more individuals who have access to that computer.

Suggested Citation

PAUP* 4.0 is in many ways comparable to a monographic article. In its approximately 180,000 lines of code, PAUP* implements numerous original concepts and ideas and contains many new algorithms. For these reasons, citation of the program in a book format is recommended:

Swofford, D. L. 2002. PAUP*. Phylogenetic Analysis Using Parsimony (*and Other Methods). Version 4. Sinauer Associates, Sunderland, Massachusetts.

TABLE OF CONTENTS

Commands used in the DISTANCES Block 27

Commands used in the PAUP Block 30

COMMAND REFERENCE

This document describes the PAUP* command format. Unless stated otherwise, these commands may be included in the PAUP block of a NEXUS file, or typed from the command line. The EDIT command may only be issued from the command line. All users will need to be familiar with the commands available in the TAXA, CHARACTERS, ASSUMPTIONS, SET, TREES, CODON, and DISTANCE blocks, which are normally placed in PAUP input files. In versions of PAUP* that provide a menu-based interface, only knowledge of these "block" commands is required, as other instructions to PAUP* can be accomplished through the menu system. Some users, however, may prefer a command-driven interface. Consequently, except for certain machine-specific features, PAUP* may be controlled entirely by a command-line system. Commands may also be placed into input files for convenience (i.e., to avoid repeated typing of complicated commands) and to provide a simple "batch" facility. If you have no experience using PAUP*, then we recommend reading the Quick Start Tutorial included on your distribution disk and also available on the World Wide Web at:

http://paup.csit.fsu.edu/downl.html

Command Format

The command descriptions below use some conventions that will be more fully described in subsequent editions of this reference document. Remember that upper-case items are to be entered as shown. Italicized items—e.g., user-item—represent variable items to be substituted by the user. Items inside of square brackets—e.g., [OPTIONAL_ITEM]—are optional. Items inside of curly braces and separated by vertical bars—e.g., { X | Y | Z }—are mutually exclusive options; only one of the choices indicated may be used. The default item (if any) is underlined, as in { ABC | DEF }. Options in commands are generally specified as "KEYWORD = VALUE" for options that have two or more potential values. For options that represent simple "on-off" ("Boolean") switches, KEYWORD = YES and KEYWORD = NO can be used to select or deselect options, respectively. Also, options may be selected by simply specifying KEYWORD.

Each command begins with a command-name and ends with a semicolon. Otherwise, the commands are completely free-format. A command may span any number of lines and whitespace (tabs and blanks) may be inserted at will. Input of PAUP* commands is case-insensitive, so you may enter command names, option keywords, etc., in any combination of upper- and lower-case characters (the only exception pertains to the MATRIX command). In addition, PAUP* allows abbreviation of command names and keywords to the shortest unambiguous truncation. Note that other NEXUS-conforming programs may not accept these abbreviations. MacClade 3.0, in particular, does not allow abbreviations, so if you want

your data file to be MacClade-compatible, then commands visible to MacClade (i.e., those not contained within a PAUP block) should not be abbreviated.

IDENTIFIERS

"Identifiers" are simply names given to taxa, characters, and other PAUP input elements such as character-sets, taxon-sets, and exclusion-sets. They may include any combination of upper- and lower-case alphabetic characters, digits, and punctuation. If the identifier contains any of the following characters:

( ) [ ] { } / \ , ; : = * ' " ` + - < >

or a blank, the entire identifier must be enclosed in single quotes. Underscores (_) are translated to blanks, unless the identifier is enclosed in single quotes. For example, the identifiers `Homo_sapiens` and `'Homo sapiens'` are equivalent, but distinct from `'Homo_sapiens'`. To include a single-quote in the identifier, you must use two consecutive single quotes. Any trailing blanks are stripped before the identifier is stored; leading blanks are preserved.

Examples of valid identifiers:

```
subterraneus
Mus_musculus
H._sapiens
'H. sapiens #429'
'Fred''s new sp.'
'rusticus (1)'
'"shoal bass"'
AMNION
_23
x21.02
myType
```

PAUP* imposes limits on the lengths of identifiers as defined below. If you use identifiers that exceed these limits, PAUP* simply truncates them to the maximum acceptable length. If you use names that are longer than the maximum lengths (e.g., for other NEXUS programs that allow longer identifiers), you should make sure that identifiers in the same class will be unique after truncation by PAUP*.

MacClade will not accept all-digit character or taxon names. PAUP* allows them, with a warning that taxon/character names have precedence over numbers (e.g., if the tenth character is named "5", the command EXSET *no_5=5 causes character number five to be excluded by MacClade and character number ten to be excluded by PAUP*). It is not advisable to use all-digit names in PAUP*, especially if you intend to also use the data file in MacClade. (See ALLDIGLAB on page 118 for a description of digit label handling.)

Taxon identifiers

For input, taxa may be referenced either by name or by number. The numbers are simply the row number of the taxon in the input data matrix. Note that if you delete taxa, the original taxon numbers are still used to refer to taxa.

The maximum length of a taxon name is 127 characters. However, in most output these names will be truncated to 16 characters due to overall line-length constraints. Thus, to avoid confusion you should make the first 16 characters of your taxon names unique. The full names will be used in graphical tree output. If you specify SET TAXLABELS=FULL, then full taxon names will be used for tree diagrams (but not in other contexts) in normal output as well (although they will still be truncated somewhat if the total output width cannot accommodate the long names).

Taxon names have priority over taxon numbers in input commands. This only becomes an issue if you use all-digit taxon names. For example, if five taxa in a data file are, for whatever reason, given the names

```
One
34
three
3
673
```

then the command `delete 3;` would cause deletion of the fourth taxon. Because of the potential confusion, you are strongly urged not to use all-digit taxon names.

> The name "ALL" is reserved for a taxon list containing all of the taxa in the data matrix (see below). You may not name a taxon "ALL".

Character identifiers

Ordinarily, PAUP* refers to characters by consecutive integers starting with 1. These integers are used both for input commands and in the output. You may also assign alphanumeric character names if you wish. The maximum length of a character name is 127, but PAUP* truncates character names to 10 characters in its output. Thus, you should choose names that are unique up to at least the first 10 characters. If you assign alphanumeric character names, these will also be used to identify characters in PAUP* output.

The rules for including punctuation, underscores, and blank characters are the same for character names as for taxon names (see page 6).

As for taxa, character names have priority over character numbers. For example, if you used all-digit character names (e.g., sequence positions or restriction-map locations) such as

```
2
4
11
23
89
102
```

and issued the command `exclude 4-5;` then all but the first and last characters would be excluded (i.e., *name* "4" through *number* 5). Because of the potential confusion, it is suggested that you always include at least one non-digit in alphanumeric character names.

> The name "ALL" is reserved for a character list containing all of the characters in the data matrix (see below). You may not name a character "ALL".

Other names

Identifiers are also used for TYPESET, WTSET, EXSET, CHARSET, TAXSET, and ANCSTATES definitions. These names follow the same rules as taxon and character names, and have a maximum length of 10 characters.

COMMON COMMAND ELEMENTS

The following elements are used in more than one command and are defined here to minimize redundancy.

Taxon lists

A **taxon list** is a sequence of one or more taxon identifiers (names and/or numbers. If two taxon identifiers are separated by a hyphen, this indicates that the range of taxa between the first taxon and the second taxon (inclusive) are to be included in the list. (The second taxon must have a higher number than the first taxon).

A taxon list composed of the reserved name "ALL" specifies all of the taxa in the data matrix.

Examples of valid taxon lists (assuming that corresponding taxon identifiers have been defined in the TAXA block) are as follows:

```
heteroclitus
M23-Q45 S1 T5;
1 3 5-8
all
```

Character lists

A **character list** is a sequence of one or more character identifiers (names and/or numbers). If two character identifiers are separated by a hyphen, this indicates that the range of characters between the first character and the second character (inclusive) are to be included in the list. (The second character must have a higher number than the first character). If the second

specification in the range is followed by a backslash (\), then an integer value immediately following the backslash represents an **_increment_**. For example, the list 3-24\3 consists of the characters 3, 6, 9, …, 21, 24. If a range-plus-increment is used, only those characters contained within the range are included in the list (e.g., the list 2-5\2 includes only characters 2 and 4).

The special identifier consisting only of an unquoted period refers to the last character in the data matrix (=NCHAR).

A character list composed of the reserved name "ALL" specifies all of the characters in the data matrix.

Examples of valid character lists (assuming that corresponding character identifiers have been defined in the SETS block) are as follows:

```
8
amnion appendages gizzard teeth;
1 two 5 7
11-.
1 3-7 16 31 28
LLSCALES-CPDSCALES
3-.\3
all
```

Character states

A **_character state_** is a single digit, alphabetic character, or other symbol that represents a valid character state as defined by the SYMBOLS list.

Tree lists

A **_tree list_** is simply a list of tree numbers referring to one or more trees currently stored in memory. If two tree numbers are separated by a hyphen, this indicates that the range of trees between the first tree number and the second tree number (inclusive) are to be included in the list. To include all trees currently in memory, specify "ALL" as the tree list.

Input/Output files

A number of PAUP* commands contain options to input or output data. Unless PAUP* is told otherwise, the default path-name for an input/output file is the path-name of the most recently accessed file. To override this default you must specify an explicit path-name. Path-names must follow operating system conventions. For example:

OS	Syntax
Macintosh	'Harddrivename:Tree Folder:treefile.trees'
Dos	C:\Tree_Fol\treefile.tre

Window95/NT   'C:\Tree Folder\treefile.trees'

UNIX   /home/users/Bob/tree_folder/treefile.trees

Unlike the PAUP* convention for identifiers, underscores (_) are not translated into blanks. File-names containing blanks must be enclosed within single-quotes for the Macintosh and Window95/NT releases and are not allowed under either the DOS or Unix releases. See your operating system documentation for other file naming conventions.

Commands used in the TAXA Block

The TAXA block specifies information about taxa. In previous versions of PAUP, both the taxa information and the character matrix were contained in the DATA block. However, in some instances the character matrix is not needed, making the option to input only taxa labels useful. While it is still permitted, we strongly recommend against use of a DATA block or definition of taxa in a CHARACTERS or DISTANCES block because this will prevent some programs from extracting taxon information from the file.

```
BEGIN TAXA;
    DIMENSIONS NTAX=number-of-taxa;
    TAXLABELS taxon-name taxon-name ...;
END;
```

DIMENSIONS must appear before **TAXLABELS**, and only one of each command is allowed per block.

DIMENSIONS

The **DIMENSIONS** command specifies the size of the data matrix. The syntax is:

DIMENSIONS NTAX = *number-of-taxa*

where *number-of-taxa* and are integer values.

TAXLABELS

The **TAXLABELS** command defines taxa, specifies their names, and determines their order. The syntax is:

TAXLABELS *taxon-name1 taxon-name2 … taxon-name* NTAX ;

See "Taxon identifiers" on page 7 for rules on the format of taxon names.

Taxa may also be defined in the CHARACTERS and DISTANCES blocks, if the NEWTAXA token is included in the DIMENSIONS command; see the descriptions of those blocks for details.

Commands used in the CHARACTERS Block

A CHARACTERS block contains the data matrix and other associated information. Taxa are usually not defined in a CHARACTERS block; if they are not defined here, this block must be preceded by a block which defines taxon labels and ordering (e.g., TAXA). The syntax is:

```
BEGIN CHARACTERS;
    DIMENSIONS
        [ NEWTAXA NTAX=number-of-taxa ]
        NCHAR = number-of-characters;
    [ FORMAT
        [ DATATYPE = { STANDARD | DNA | RNA | NUCLEOTIDE |
                PROTEIN } ]
        [ RESPECTCASE ]
        [ MISSING = symbol ]
        [ GAP = symbol ]
        [ SYMBOLS = "symbol symbol..." ]
        [ EQUATE = "symbol=entry [ symbol=entry... ]" ]
        [ MATCHCHAR = symbol ]
        [ LABELS = { NO | LEFT | RIGHT } ]
        [ TRANSPOSE ]
        [ INTERLEAVE ]
        [ [NO]TOKENS ]; ]
    [ ELIMINATE character-set; ]
    [ TAXLABELS taxon-name taxon-name...; ]
    [ CHARSTATELABELS
        character-number [ character-name ]
        [ / state-name [ state-name... ] ], ...; ]
    [ CHARLABELS character-name [character-name...]; ]
    [ STATELABELS character-number [ state-name [ state-name... ] ],...; ]
    MATRIX data-matrix;
END;
```

DIMENSIONS, **FORMAT**, and **ELIMINATE** must all precede **CHARLABELS**, **STATELABELS**, and **MATRIX**. **DIMENSIONS** must precede **ELIMINATE**. Only one of each command is allowed per block.

DIMENSIONS

The **DIMENSIONS** command specifies the size of the data matrix. Ordinarily, the CHARACTERS block is proceeded by a TAXA block which contains taxa information. However, if new taxa are to be defined in the CHARACTERS block, this must be indicated by the NEWTAXA subcommand. NEWTAXA, if present, must appear before the NTAX subcommand. The number following NCHAR is the number of characters in the data matrix.

FORMAT

The **FORMAT** command is used to specify information pertaining to the format of the data file. The DATATYPE subcommand must appear first in the command. The RESPECTCASE subcommand must appear before the MISSING, GAP, SYMBOLS, and MATCHCHAR subcommands.

Any or all of the following option specifications may be given:

DATATYPE = { <u>STANDARD</u> | DNA | RNA | NUCLEOTIDE | PROTEIN }

This subcommand specifies the class of data. If present, it must be the first subcommand in the FORMAT command. If DATATYPE = STANDARD, the SYMBOLS list is taken from the SYMBOLS = "*symbols-list*" item, above (default = "01"). If DATATYPE is set to one of the molecular sequence types, a predefined SYMBOLS list is used ("ACGT" for DNA, "ACTU" for RNA, and the standard one-letter amino acid codes for PROTEIN). In addition, standard ambiguity codes are implemented by predefined EQUATE macros.

RESPECTCASE

By default, PAUP* does not distinguish between upper- and lower-case character-state symbols in the data matrix. If you want upper- and lower-case representations of the same alphabetic character to refer to different character states, specify RESPECTCASE. This subcommand must appear before the SYMBOLS subcommand and is not applicable to DATATYPE=DNA, RNA, NUCLEOTIDE, and PROTEIN.

MISSING = *symbol*

The *symbol* specifies a character used to represent missing data. Any alphabetic, numeric, or other character that may be used as a character-state symbol may be used as the *missing-symbol*. If MISSING is not specified, it defaults to '?'. Whitespace is illegal as a missing data symbol, as are the following:

() [] {} ∕ \ , ; := * ' "` <>^

GAP = *symbol*

The *symbol* a character used to represent alignment gaps, corresponding to insertions and/or deletions. For example, gap=- would assign the hyphen as the gap character. The GAP setting is ignored unless DATATYPE is DNA, RNA, or PROTEIN.

Alignment gaps may be treated either as missing data or as an additional character-state (fifth base or 21st amino acid) using the GAPMODE option. Whitespace is illegal as a gap symbol, as are the following:

() [] {} ∕ \ , ; := * ' "` <>^

SYMBOLS = "*symbol symbol...*"

The *symbol* list defines a set of permissible symbols that may be used

to designate character states. The default SYMBOLS list is "01" for the STANDARD data type (see below), which means that the only (non-missing) character-state symbols permitted are '0' and '1'. If you want to use any other symbols to designate character states, you must explicitly define an alternate SYMBOLS list.

The format of a symbols-list is a sequence of single-character "symbols"; the entire list is then enclosed within double-quotes. Whitespace is not needed between elements: SYMBOLS="012" is equivalent to SYMBOLS="0 1 2".

EQUATE = "*symbol = entry* "

The EQUATE option provides a simple macro facility for translating character-state specifications in the data matrix to alternate character-state specifications. The symbol component must be a single-character. *Entry* is either a valid character-state or a character-state set .

Any number of EQUATE macros may be specified following the equal sign, but only one pair of double-quotes is used. For example:

```
format equate="U=T R={AG} .=- X=?";
```

EQUATE macros may not be defined recursively. That is, you cannot equate A to B and B to C, expecting A to be expanded to C. If you equate the same symbol to more than one expansion, the last definition applies. Case is significant in equate symbols. That is, MISSING=? EQUATE="E=(012) e=?" means that E will be interpreted as 0, 1, and 2, e as missing data.

MATCHCHAR = *symbol*

If a *symbol* is specified, any occurrence of that symbol in the data matrix is translated to the state (or state-set) occurring in the first row of the matrix. For example:

```
format datatype=dna gap=- matchchar=.;
matrix
        One  ATGCT ATCCG TCATG ACCTA
        Two  ..C.. .G... .--A. ..GG.
        ThreeC.... .G... .GGA. T....
        Four C..AA ---.. A...A .GTC.
;
```

Whitespace is illegal as a matching character symbol, as are the following:

() [] {} , ; = * ' "` / \<>^

LABELS = { NO | LEFT | RIGHT }

If LABELS = LEFT, taxon names (or character names, if the data matrix is transposed) begin each row of the data matrix (i.e., precede the character-state data). If LABELS = RIGHT, then these names end each row of the data matrix. The default is LABELS = LEFT. The LEFT and RIGHT options are PAUP* extensions. We recommend the use of

LABELS or NOLABELS for compatibility with other NEXUS programs. LABELS with no other qualifier is equivalent to LABELS = LEFT. Example input format:

```
begin data;
    dimensions ntax=4 nchar=5;
    format label = right;
    matrix
        00111 taxon1
        0111? taxon2
        11001 taxon3
        10000 taxon4
        ;
end;
```

TRANSPOSE

If TRANSPOSE is specified, rows of the data matrix correspond to characters and columns correspond to taxa. Otherwise, rows correspond to taxa and columns to characters. The following is an example of a transposed data matrix:

```
begin data;
    dimensions ntax=4 nchar=5;
    format transpose;
    taxlabels taxon1 taxon2 taxon3 taxon4;
    matrix
        char1    0011
        char2    0110
        char3    1100
        char4    1100
        char5    1?10
        ;
end;
```

INTERLEAVE

Specification of allows the data to be entered in "blocks" of characters. If the data is not transposed, then each "block" contains information for some of the characters, for all taxa. Taxa in each section must occur in the same order. This format is especially useful for molecular sequence data, where the number of characters can be large. A small interleaved matrix follows:

```
matrix
        taxon1     A C C T C G G C
        taxon2     A C C T C G G C
        taxon3     A C G T C G C T
        taxon4     A C G T C G C T
```

```
                          taxon1     T  T  A  A  C  G  A
                          taxon2     T  T  A  A  C  C  A
                          taxon3     C  T  C  A  C  C  A
                          taxon4     T  T  C  A  C  C  A
                          ;
```

The interleaved sections need not all be of the same length. In an interleaved matrix, new line characters are significant; they indicate that the next character information encountered applies to a different taxon (for non-transposed matrices).

[NO]TOKENS

The TOKENS subcommand is currently not supported by PAUP* 4.0 beta; however, this subcommand is described here because it will be supported in the final release 4.0. This subcommand specifies whether or not data matrix entries are single symbols, or can be tokens. If TOKENS is specified, then the data values must be full NEXUS tokens, separated by whitespace or punctuation as appropriate, as in the following example:

```
begin characters;
        dimensions nchar=3;
        format tokens;
        matrix
            taxon_1 absent red big
            taxon_2 absent blue small
            taxon_3 present blue small;
end;
```

[NO]TOKENS is the default and is not allowed for DATATYPES DNA, RNA, and NUCLEOTIDE. If TOKENS is invoked, the standard 3-letter amino acid abbreviations can be used with DATATYPE = PROTEIN, and defined state names for DATATYPE = STANDARD.

ELIMINATE

This command allows one to specify a list of characters that are to be excluded from consideration. PAUP* will completely ignore eliminated characters. (This is similar to the ZAP subcommand in version 3.1 of PAUP.) For example;

```
eliminate 4-100;
```

tells the program to skip over characters 4 through 100 when the matrix is read. Character-set names are not allowed in the character list. This command does not affect character numbers.

TAXLABELS

This command allows one to specify the names of the taxa. It serves to define taxa, and is only allowed in a CHARACTERS block if the NEWTAXA token is included in the **DIMENSIONS** statement.

CHARSTATELABELS

CHARSTATELABELS allows one to specify both the names of the characters and names of the states. This command was developed as an alternative to the older commands CHARLABELS and STATELABELS. Here is an example:

```
charstatelabels
        1 eye_color/red blue green,
        3 head_shape/round square,
        5 pronotum_size/small medium large
;
```

A forward slash (/) separates the character name and the state names, with a comma separating the information for different characters. If no state names are to be specified, the slash may be omitted; if no character names are to be specified, the slash must be included, but no token needs to be included between the character number and the slash. If state x is the last state to be named, then subsequent states need not be named but states 1 through x must be. If no name is to be applied to a state, enter a single underscore for its name. Character names must not correspond to another character name or number; thus, 1 is not a valid name for the second character listed. PAUP* does not use character-state label information; however, it will extract the name from a **CHARSTATELABELS** command.

CHARLABELS

This command allows one to specify names of characters:

```
charlabels
        flange microsculpture body_length
        hind_angles #_spines spine_size
        _ _ head_size pubescent_intervals
        head_color clypeal_margin;
```

Character labels are listed consecutively. If character x is the last character to be named, then subsequent characters need not be named, but characters 1 through x need to be. If no name is to be applied to a character, a single underscore can be used for its name. Character names must not correspond to another character name or number; thus, 1 is not a valid name for the second character listed.

The command should be used only for non-transposed matrices (in transposed matrices, the character labels are defined in the MATRIX command).

STATELABELS

Use this command to assign names to character states. PAUP* does not use this information in any way, however, the information is used by MacClade and possibly other NEXUS programs.

```
statelabels
        1 absent present,
        2 isodiametric transverse,
        3 '4.5-6.2mm' '6.3-7.0mm' '7.7-11.0mm',
        4 rounded subangulate angulate,
        10 0 '1-4' '6-9' '7-9' '8-9' 7 8 9,
        11 black rufous metallic flavous,
        12 straight concave,
    ;
```

State labels need not be specified for all characters. A comma must separate state labels for each character. State labels are listed consecutively within a character. If state x is the last state to be named, then subsequent states need not be named, but states 1 through x must be. If no name is to be applied to a state, enter a single underscore for its name.

MATRIX

In its standard format, the **MATRIX** command contains a sequence of taxon names and state information for that taxon. The matrix itself may appear in a variety of forms. For example:

```
begin data;
   dimensions nchar=5;
   matrix
       taxon1    00111
       taxon2    0111?
       taxon3    11001
       taxon4    10000
       ;
end;
```

```
        begin data;
           dimensions nchar=5;
           matrix
               taxon1
                   001
                   11
               taxon2
                   011
                   1?
               taxon3
                   110
                   01
               taxon4
                   100
                   00
        end;

        begin data;
           dimensions nchar=5;
           matrix taxon1 00111 taxon2 0111? taxon3 11001
                   taxon4 10000;
        end
```

Taxa need not be in same order as in the TAXA block, and the matrix need not contain all taxa. For interleaved matrices, all sections must have the same taxa in the same order.

Commands used in the DATA Block

The DATA block contains the data matrix and other associated information. This block is equivalent to a CHARACTERS block in which the NEWTAXA subcommand is included in the **DIMENSIONS** command. In versions 3.0-3.07 of MacClade and 3.0-3.1.1 of PAUP, both the data matrix and taxa were saved in a DATA block. However, in some instances the character matrix is not needed, making the option to input only taxa labels useful. All of the options described in the TAXA and CHARACTERS blocks are still permitted under the DATA block; however, we strongly encourage the use of the TAXA and CHARACTERS blocks for the sake of compatibility with other programs using the NEXUS-format.

The following CHARACTER and TAXA blacks are equivalent to the single DATA block:

```
begin taxa;
       dimensions ntax=3;
       taxlabels taxon_1 taxon_2 taxon_3;
end;
```

```
begin characters;
        dimensions nchar=12;
        format datatype=DNA;
        matrix
            taxon_1  CCATGGTACGT
            taxon_2  TCCATGCTACCC
            taxon_3  TCCATGGAACCC;
end;

begin data;
        dimensions ntax = 3 nchar=12;
        format datatype=DNA;
        matrix
            taxon_1  CCATGGTACGT
            taxon_2  TCCATGCTACCC
            taxon_3  TCCATGGAACCC;
end;
```

Commands used in the ASSUMPTIONS Block

The ASSUMPTIONS block houses assumptions about the data, or gives general directions as to how to treat it (e.g., which characters are to be excluded from consideration). Its syntax is:

BEGIN ASSUMPTIONS;
    [OPTIONS
        [DEFTYPE = *type-name*]
        [POLYTCOUNT = { MINSTEPS | <u>MAXSTEPS</u> }]
        [GAPMODE = { <u>MISSING</u> | NEWSTATE | INDEL }] ;]
    [USERTYPE *type-name*
        [({ <u>STEPMATRIX</u> | CSTREE })] = *usertype-description*;]
    [TYPESET [*] *typeset-name* [({ <u>STANDARD</u> | VECTOR })] =
        typeset-definition;]
    [WTSET [*] *wtset-name* [({ <u>STANDARD</u> | VECTOR } { <u>TOKENS</u> |
        NOTOKENS })] = *wtset-definition*;]
    [EXSET [*] *exset-name* [({ <u>STANDARD</u> | VECTOR })] = *character-set*;]
    [ANCSTATES [*] *ancstates-name* [({ <u>STANDARD</u> | VECTOR }
        { TOKENS | NOTOKENS })] =*ancstates-definition*;]
  END;

In earlier versions of PAUP, the CHARSET and TAXSET commands also appeared in the ASSUMPTIONS block; now they appear in the SETS block. The GAPMODE subcommand of the OPTIONS command of this block was originally housed in an **OPTIONS** command in the DATA block. As this subcommand dictates how data are to be treated, rather than provide details about the data themselves, it was moved into the ASSUMPTIONS block.

OPTIONS
_____

This command houses a number of disparate subcommands. They are all of the form subcommand = option.

DEFTYPE = *type-name*

This specifies the default character type for parsimony analyses. Whenever a character's type is not explicitly stated, its type is taken to be the default type. The default DEFTYPE is UNORD , but may be changed to one of the following: ORD, UNORD, DOLLO, DOLLO.UP, DOLLO.DN, IRREV, IRREV.UP, or IRREV.DN.

POLYTCOUNT = { MINSTEPS | MAXSTEPS }

This option specifies whether polytomies are treated as "hard" or "soft" (see Maddison (1989)) when counting the number of steps required by a character and when reconstructing ancestral states. PAUP* currently supports only POLYTCOUNT = MAXSTEPS ("hard" polytomies), and will ignore the POLYTCOUNT specification.

GAPMODE = { MISSING | NEWSTATE | INDEL }

Specifies how gaps are to be treated. GAPMODE = MISSING designates gaps to be treated in the same way as missing data; GAPMODE = NEWSTATE specifies that gaps are to be treated as an additional state (for DNA/RNA data, as a fifth base).

USERTYPE
_____

***** Give the predefined character types here. The USERTYPE command defines a character transformation type, as used in parsimony analysis to designate the cost of changes between states. USERTYPES must be defined before they are referred to in any TYPESET. Character-type-description must follow the rules for character-state tree or stepmatrix description.

The STEPMATRIX format is as follows:

```
usertype mymatrix (stepmatrix)=n
        s   s   s   s
        .   k   k   k
        k   .   k   k
        k   k   .   k
        k   k   k   .;
```

where n is the number of rows and columns in the step matrix, the s's are state symbols, and the k's are the cost for going between states. n can take any value _ 2. Diagonal elements may be listed as periods. If a change is to be prohibited, then one enters an "i" for infinity. Typically, the state symbols will be in sequence, but they need not be. The following matrices assign values identically:

```
usertype mymatrix (stepmatrix)=4
        0  1  2  3
        .  1  5  1
        1  .  5  1
        5  5  .  5
        1  1  5  .;

usertype mymatrix2 (stepmatrix)=4
        2  0  3  1
        .  5  5  5
        5  .  1  1
        5  1  .  1
        5  1  1  .;
```

The CSTREE format is very similar to the TREE format in a TREES block. That is, character state trees are described in the parenthesis notation following the rules given for Trees of taxa. Instead of taxon labels, character state symbols are used. Thus

```
usertype cstree-name (cstree) = [(list of
subtrees)][state-symbol]];
```

where each subtree has the same format as the overall tree and the subtrees are separated by commas.

TYPESET

A **TYPESET** specifies a set of character-type assignments for each character as used in parsimony analysis. Any characters not listed in the character-set have the default character type. The type names to be used are either the predefined ones or those defined in a **USERTYPE** command. Each value in a definition in VECTOR format must be separated by whitespace. The following are equivalent type sets:

```
typeset mytypes = Ord: 1 4 6, Unord: 2 3 5;

typeset mytypes (vector) = Ord Unord Unord Ord
         Unord Ord;
```

WTSET

A **WTSET** specifies a set of character-weight assignments for each character. Any characters not listed in the character-set have weight 1. The weights may be either integers or real numbers. Each value in a definition in VECTOR format must be separated by whitespace, unless the NOTOKENS option is invoked, in which case no whitespace is needed, and all weights must be integers in the range 0-9. The following are equivalent weight sets:

```
wtset mywts = 3: 1 4 6, 1: 2 3 5;
```

```
wtset mywts (vector) = 3 1 1 3 1 3;
```

EXSET

EXSET specifies a set of characters that may be jointly excluded. This is a standard object definition command. Any characters not listed in the character-set are included. The VECTOR format consists of 0's and 1's - a 1 indicates that the character is to be excluded; whitespace is not necessary between 0's and 1's. The following commands are equivalent, and serve to exclude characters 5, 6, 7, 8, and 12.

```
exset * toexclude = 5-8 12;

exset * toexclude (vector) = 000011110001;
```

If you precede the *exset-name* with an asterisk, any previously excluded characters are re-included and the characters specified by *character-list* are excluded.

ANCSTATES

The **ANCSTATES** command allows one to specify ancestral states.   Any valid state symbol can be used in the description for discrete data, and any valid value for continuous data. TOKENS is not allowed for DATATYPES DNA and RNA. If TOKENS is invoked, the standard 3-letter amino acid abbreviations can be used with DATATYPE = PROTEIN, and defined state names for DATATYPE = STANDARD. The following commands are equivalent:

```
ancstates ancestor = 0:1-3 5-7 12, 1:4 8-10, 2:11;

ancstates ancestor (vector) = 000100011120;
```

Any number of *ancstates-name:ancstates-definition* pairs, separated by commas, may be specified. You may also repeat the **ANCSTATES** command to define multiple ancestors, although only one ancestor can be in effect at any given time. If you precede the ancestor-name with an asterisk, that ancestor becomes the "current" ancestor. You can also assign the current ancestor with the **ASSUME** command.

Commands used in the SETS Block

This block stores sets of objects (characters, states, taxa, etc.). The general structure of the SETS block is as follows:

BEGIN SETS;
    [CHARSET *charset-name* [({STANDARD | VECTOR})]=*character-list* ;]

[TAXSET *taxset-name* [({ STANDARD | VECTOR })] = *taxon-list*;]
[CHARPARTITION *partition-name*
    [([{ [NO]TOKENS }]
    [{ STANDARD | VECTOR }])]
        = *subset-name* : *character-set*,...;]
[TAXPARTITION *partition-name*
    [([{ [NO]TOKENS }]
    [{ STANDARD | VECTOR }])]
        = *subset-name* : *taxon-set*,...;]
    END;

CHARSET

The **CHARSET** command specifies and names a set of characters; this name can then be used in subsequent CHARSET definitions or wherever a *character-list* is required. The VECTOR format consists of 0's and 1's: a 1 indicates that the character is to be included in the CHARSET; whitespace is not necessary between 0's and 1's. The name of a CHARSET cannot be equivalent to a character name or character number.

PAUP* includes the following predfined character-sets for all DATATYPES:

```
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.
```

There are four additional predefined character-sets for characters of DATATYPE = DNA, RNA, NUCLEOTIDE if a CODONS block containing a CODONPOSSET is supplied.

```
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.
```

TAXSET

This command defines a set of taxa. A **TAXSET** name can be used in subsequent **TAXSET** definitions or wherever a taxon-set is required. The name of a **TAXSET** cannot be equivalent to a taxon name or taxon number. The taxa to be included are described in a taxon-set. For example, the following command

```
taxset beetles = Omma-.;
```

defines the **TAXSET** "beetles" to include all taxa from taxon Omma to the last defined taxon.

The VECTOR format consists of 0's and 1's: a 1 indicates that the taxon is to be included in the **TAXSET**; whitespace is not necessary between 0's and 1's.

CHARPARTITION, TAXPARTITION

The partition divides the objects into several (mutually exclusive) subsets. They follow the same format, and will be described together.

There are several formatting options. The VECTOR format consists of a list of partition names. By default, the name of each subset is a NEXUS word (this is the TOKENS option). The NOTOKENS option is only available in the VECTOR format; this allows one to use single symbols for the subset names. Each value in a definition in VECTOR format must be separated by whitespace if the names are tokens, but not if they are NOTOKENS.

Example 1

The following two examples are equivalent:

```
taxpartition populations=1:1-3, 2:4-6, 3:7 8;

taxpartition populations (vector notokens) = 11122233;
```

Example 2

The following two examples are equivalent:

```
taxpartition mypartition=Chiricahua: 1-3,
    Huachuca: 4-6, Galiuro: 7 8;

taxpartition mypartition (vector) = Chiricahua
Chiricahua Chiricahua Huachuca Huachuca Huachuca
Galiuro Galiuro;
```

Example 3

The following example of the CHARPARTITION command creates a character partition named gfunc which defines coding and noncoding regions of the sequences.

```
charpartition gfunc = 1:2-457 660-896, 2:1 458-659
897-898;
```

Example 4

The following example of the CHARPARTITION command is equivalent to the Example 3 except that two predefined character sets are used to define the characters included in each partition. See the "CharSet" command described on page 23 for a complete list of predefined character sets.

```
charset coding = 2-457 660-896;
charset noncoding = 1 458-659 897-898;
charpartition gfunc = 1:coding, 2:noncoding;
```

Commands used in the TREES Block

The TREES block is used to input user-defined trees to PAUP*. A single **TREE** command is used for each tree; any number of **TREE** commands may be included in the block.

The syntax for the TREES block follows:

> BEGIN TREES [*block-name*] ;
>   [TRANSLATE *token taxon-name* [, *token taxon-name*] … ;]
>   [TREE [*] *name* = *tree-specification*;]
> END;

TRANSLATE

Use the **TRANSLATE** command to define mappings of arbitrary tokens appearing in the **TREE** command to valid taxon names. Ordinarily, the tokens are the integers 1 through NTAX. The tree description requires references to the taxa defined in a TAXA, CHARACTERS , DATA, or DISTANCES block.

TREE

Use the **TREE** command to define trees. If no rooting-specification is supplied, the default in PAUP* is to interpret the tree description as specifying a rooted tree. You can override this behavior by using a "special comment." The comment '[&U]' indicates that the tree should be

considered unrooted and the comment '[&R]' indicates that the tree should be considered rooted. For example:

```
tree mytree = [&R] ((1,2),(3,4));
```

is a rooted tree, whereas

```
tree mytree = [&U] ((1,2),(3,4));
```

is an unrooted tree. You will usually define *unrooted* trees. These trees are then rooted for output purposes using the outgroup or Lundberg rooting procedures. However, if you wish to define an ancestor for the full tree (see **ANCSTATES** command), if you are using directed characters, or if you want to evaluate trees under a maximum likelihood "clock" constraint you may wish to define rooted trees instead. In any case, you can convert rooted trees to unrooted trees and vice versa using the **ROOTTREES/DEROOTTREES** commands.

Note that in order to write the tree descriptions, unrooted trees may be rooted at any convenient point (including a terminal taxon or internal node); the position of the root is simply ignored when the tree is stored.

Another "special comment" stored in the TREES block and used by PAUP* is the tree weight comment. Tree weights are the reciprocal of the number of trees found in either a **BOOTSTRAP** or **JACKKNIFE** replicate. This option allows the combination of bootstrap results from runs performed at different times or on different machines and the recovery of results obtained prior to a system crash. For example the TREEFILE created by the **BOOTSTRAP** command might look like this:

```
[Trees found in bootstrap replicate #1]
tree B_1.1 = [&W 1] (1,((((2,4),5),3),6));

[Trees found in bootstrap replicate #2]
tree B_2.1 = [&W 1/2] (1,((((2,4),5),6),3));
tree B_2.2 = [&W 1/2] (1,(((2,4),(5,6)),3));
```

An example tree with branch lengths is

```
tree PAUP_1 = [&U] (1:22,(2:12,3:9):9,4:72);
```

PAUP* currently ignores the tree names; trees are referred to subsequently only by number, in the order in which they are presented in the TREES block. (MacClade uses the tree names, however.) If the name is preceded by an asterisk (*), the tree becomes the "default" tree. (E.g., a **DESCRIBETREES** command with no tree list will result in the description of the default tree.)

If a taxon is omitted from the tree specification, it is assumed to descend from the root of the tree described by the remaining taxa.

Commands used in the CODONS Block

The CODONS block contains information about the genetic code, the regions of DNA and RNA sequences that are protein coding, and the location of triplets coding for amino acids in nucleotide sequences. Numerous other commands used by other NEXUS-compliant programs may be included in the CODONS block. Currently only the CODONPOSSET command is implemented in PAUP*; all other CODONS block commands will be ignored.

```
BEGIN CODONS;
    [ CODONPOSSET [*] name [({ STANDARD | VECTOR})] =
         N: character-set,
         1: character-set,
         2: character-set,
         3: character-set;]
END;
```

CODONPOSSET

The **CODONPOSSET** command stores information about protein coding regions and the codon positions of nucleotide bases in protein-coding regions.

Those characters designated as 1, 2, or 3 are coding bases which are specified as being of positions 1, 2, and 3, respectively. Those characters designated as N are considered non-protein-coding. Those characters designated as ? are of unknown nature. Any unspecified bases are considered of unknown nature (equivalent to ?). If no **CODONPOSSET** statement is present, all bases are presumed of unknown nature.

For example, the following command

```
codonposset * coding=
        N:1-10,
        1:11-.\3,
        2:12-.\3,
        3:13-.\3;
```

designates bases 1-10 as noncoding, and positions of the remaining bases in the order 123123123....

Commands used in the DISTANCES Block

This block contains distance matrices. Taxa are usually not defined in a DISTANCES block; rather, this block should be preceded by a block that

defines taxon labels and ordering (e.g., TAXA). The syntax of the block is as follows:

```
BEGIN DISTANCES;
    [ DIMENSIONS [ NEWTAXA ] NTAX = number-of-taxa
        NCHAR = number-of-characters;]
    [ FORMAT
        [ TRIANGLE = { LOWER | UPPER | BOTH } ]
        [ [NO] DIAGONAL ]
        [ [NO] LABELS ]
        [ MISSING = SYMBOL ]
        [ INTERLEAVE ] ; ]
    [ TAXLABELS taxon-name taxon-name...;]
    MATRIX distance-matrix ;
END;
```

Commands must appear in the order listed. Only one of each command is allowed per block.

DIMENSIONS

The NTAX subcommand of this command is needed to process the matrix when some defined taxa are omitted from the distance matrix. The NCHAR subcommand is optional, and can be used to indicate the number of characters for those analyses that need to know how many characters (if any) where used to calculate the distances. NCHAR is not required for successful reading of the matrix.

As for the CHARACTERS block, taxa can be defined in a DISTANCES block if NEWTAXA precedes the NTAXA subcommand in the DIMENSIONS command. It is advised that new taxa not be defined in a DISTANCES block, for the reasons discussed in the description of the DATA block. NEWTAXA, if present, must be appear before the NTAX subcommand.

FORMAT

This command specifies the formatting of the MATRIX. The [NO]LABELS and MISSING subcommands are as described in the CHARACTERS block.

TRIANGLE = { <u>LOWER</u> | UPPER | BOTH }

This subcommand specifies whether only the lower left half of the matrix is present, or only the upper right, or both halves. Below is one example of an upper triangular matrix and one of a matrix with both halves included.

```
begin distances;
        format triangle = upper;
        matrix
           taxon_1 0.0 1.0 2.0 4.0  7.0
           taxon_2     0.0 3.0 5.0  8.0
           taxon_3         0.0 6.0  9.0
           taxon_4             0.0 10.0
           taxon_5                  0.0;
end;

begin distances;
        format triangle = both;
        matrix
           taxon_1     0  1.0  2.0  4.0  7.0
           taxon_2   1.0    0  3.0  5.0  8.0
           taxon_3   2.0  3.0    0  6.0  9.0
           taxon_4   4.0  5.0  6.0    0 10.0
           taxon_5   7.0  8.0  9.0 10.0    0;
end;
```

DIAGONAL

If DIAGONAL is turned off, the diagonal elements are not included

```
format nodiagonal;
        matrix
           taxon_1
           taxon_2   1.0
           taxon_3   2.0  3.0
           taxon_4   4.0  5.0  6.0
           taxon_5   7.0  8.0  9.0 10.0;
```

If TRIANGLE is not BOTH, then there will be one row that contains only the name of a taxon. This row is required. If TRIANGLE = BOTH, then the diagonal must be included.

INTERLEAVE

The INTERLEAVE subcommand in the DISTANCE block is currently not supported by PAUP* 4.0 beta; however, this subcommand is described here because it will be supported in the final release 4.0. The INTERLEAVE format is similar to that described in the CHARACTERS block, although the interleaved distance matrices take a slightly different form:

```
taxon_1    0
taxon_2    1   0
taxon_3    2   3   0
taxon_4    4   5   6
taxon_5    7   8   9
taxon_6   11  12  13

taxon_4    0
taxon_5   10   0
taxon_6   14  15   0;
```

As in the CHARACTERS block, new line characters in interleaved matrices are significant, in that they indicate a switch to a new taxon.

TAXLABELS

TAXLABELS allows one to specify the names and ordering of the taxa. This command serves to define taxa, and is allowed only if the NEWTAXA token is included in the **DIMENSIONS** statement.

MATRIX

The **MATRIX** command contains the distance data.

Commands used in the PAUP Block

Commands described in this section are specific to PAUP*. They may be included in the PAUP block of a NEXUS file or typed from the command line.

Unless otherwise specified, options specified in these commands are "persistent;" i.e., they retain their values between successful invocations of the command. Persistence of options simplifies typing of commands because one a command requiring a large number of options has been entered once, subsequent invocations of the command need not respecify all of the options. The drawback is that you can become temporarily confused if you forget that a previously specified option remains in effect until you explicitly override it.

OPTIONS AND SUBCOMMANDS AFFECTING MULTIPLE COMMANDS

Several options apply to a number of commands. Specification of these options on one command affects all commands that use the same options. To minimize redundancy, these options are described in the following sections rather than in the description for each individual command to which the option applies.

Tree-searching options

These options pertain to the commands that request searching for trees.

KEEP = *real-value*

If *real-value* is zero, only the best trees found will be saved. If *real-value* ≥ zero, all trees of score ≤ *real-value* will be saved. By default, *real-value* = 0, so that only the best trees found will be saved.

ENFORCE = YES | <u>NO</u>

ENFORCE requests that topological constraints be enforced; i.e., trees that are not compatible with the constraint tree are not evaluated. If the CONSTRAINTS option (see below) is not used to specify a constraint tree, the "current" constraint tree is used.

DSTATUS = *integer-value* | NONE

Information on the progress of the search (number of trees examined, number of trees saved, etc.) may output to the display buffer using the DSTATUS setting. Specify an *integer-value* to control how frequently (in seconds) information is output to the display or specify DSATUS = NO to suppress this information. The default setting for the Macintosh versions of PAUP* is DSTATUS = NONE and DSTATUS = 60 for all other versions.

CONSTRAINTS = *constraint-tree-name*

The specified constraint tree, which must have been defined in a previous CONSTRAINTS command, becomes the current constraint tree. You must also specify ENFORCE if you want to search under constraints.

CONVERSE = YES | <u>NO</u>

If CONVERSE is specified in conjunction with ENFORCE, only trees that are not compatible with the constraint tree are evaluated. CONVERSE = NO reverses the effect of a previous CONVERSE specification.

STATUS = <u>YES</u> | NO

Information on the progress of the search (number of trees examined, number of trees saved, etc.) is output to a status window. STATUS = NO suppresses this window.

Tree-rooting options

ROOT = <u>OUTGROUP</u> | LUNDBERG | MIDPOINT

The ROOT option is used to specify how unrooted trees are to be rooted prior to output. You can choose OUTGROUP rooting, using whichever outgroup you have selected; MIDPOINT rooting, which roots the tree at its midpoint; or LUNDBERG rooting, which requires that a previous ANCSTATES command has been issued. By default, OUTGROUP rooting is in effect.

OUTROOT = <u>POLYTOMY</u> | PARAPHYL | MONOPHYL

> If outgroup-rooting is currently selected, there are three options for output. The outgroup can make up a polytomy next to the ingroup (POLYTOMY, the default); or it can be made to be paraphyletic relative to the ingroup (PARAPHYL); or the monophyletic sister group to the ingroup (MONOPHYL).

Tree output options

TCOMPRESS = YES | <u>NO</u>

> Specify TCOMPRESS to output tree diagrams in a "vertically compressed" format. The resulting diagram is not as aesthetically appealing, but it allows more of a large tree to be seen on one screen (or in one window), and it takes less paper to print.

Options for character-matrix listings

SHOWEXCLUDED = YES | <u>NO</u>

> Unless SHOWEXCLUDED is specified, "excluded" characters are not shown in character-matrix listings.

CMLABELS = <u>YES</u> | NO

> By default, character names are used to label the columns of character-matrix listings. If you want to use numbers even when character names are available, specify CMLABELS = NO.

CMCSTATUS = YES | <u>NO</u>

> If CMCSTATUS is specified, characters that are constant, "zapped," uninformative, or excluded are identified by asterisks at the top of each column of a character-matrix listing.

CMCOLWID = *column-width*

> The value specified for *column-width* determines the number of columns used for each character in the data matrix. The default is CMCOLWID=2, so that one blank column appears between each column of character state data. For sequence data, you may want to use CMCOLWID=1 in order to fit more characters onto each line of output.

CMSHOWEQ = <u>YES</u> | NO

> Unless CMSHOWEQ has been specified, if the possible state assignments to an interior node correspond to a multistate taxon code specified in an EQUATE macro, the corresponding EQUATE character is shown rather than the equivalent set of character states.

Other options

OPT = <u>ACCTRAN</u> | DELTRAN | MINF

> The OPT= subcommand determines how the characters are

optimized on the tree(s) in memory. ACCTRAN (the default) uses "accelerated transformation", DELTRAN uses "delayed transformation", while MINF optimizes so as to minimize the "f-value" (Farris, 1972; Swofford and Maddison, 1987). See the section on character-state optimization for detailed discussion of these options.

ANCSTATES = *ancestral-states-name*

Change the ancestor currently in effect to the ancestral-states-name defined in an earlier **ANCSTATES** command (or to STANDARD). This option affects searching and character-state reconstruction algorithms.

?

The ? command is a synonym for HELP. ? with no arguments requests a list of the available commands. See the "Help" command described on page 69 for further information.

Syntax

[*PAUP command*] ?;

!

Use the ! command to execute a UNIX command from within PAUP*. You can type a command that contains a semicolon by enclosing the entire command within single-quotes. You can open a temporary UNIX shell by typing, for example, !csh or !sh. When you are ready to resume your PAUP* session, type exit or Ctrl-D to exit the shell and return to PAUP*. Note, some shells respond differently to this command.

Syntax

! [*unix-command*];

AGREE

Use the AGREE command to calculate agreement subtrees.

Syntax

AGREE [*tree-list*] [*/options*];

Menu equivalent

Trees > Agreement subtrees ...

Available options

- ALL = <u>NO</u> | SETS | TREES | BOTH
- SHOWTREE = <u>YES</u> | NO
- TREEFILE = *tree-file-name*
- *REPLACE = YES | <u>NO</u>
- *APPEND = YES | <u>NO</u>
- TCOMPRESS = YES | <u>NO</u>

*Option is nonpersistent

Description of options

ALL = <u>NO</u> | SETS | TREES | BOTH

Use ALL to calculate agreement subtrees for the trees specified in the AGREE [*tree-list*]. If ALL = SETS, a list of the agreement sets and the taxa included within each set is given. If ALL = TREES, all of the agreement subtrees are printed, overriding SHOWTREE = NO if this option was specified.

SHOWTREE = <u>YES</u> | NO

Shows the agreement subtree(s) given in the [*tree-list*].

TREEFILE = *tree-file-name*

If TREEFILE is specified, a description of all agreement subtrees is output to a file containing a NEXUS-format TREES block. This option is not persistent; you must specify it on every AGREE command for which you want a tree file to be saved. See "Input/Output files" on page 9 for instructions to specify the full path name of a file.

The following options apply only if FILE is specified

REPLACE = YES | <u>NO</u>
APPEND = YES | <u>NO</u>

If the specified file already exists, you will be prompted for confirmation that the existing file should be replaced. Explicit specification of REPLACE suppresses this prompt; the existing file will be quietly overwritten by the new data.
Alternatively, you may specify APPEND, in which case a new TREES block will be concatenated to the end of an existing file.

TCOMPRESS = YES | <u>NO</u>

See "Tree output options" on page 32.

ALLTREES

Use the ALLTREES command to perform an exhaustive search of all possible tree topologies.

Syntax

ALLTREES [*options*];

Menu Equivalent

Analysis > Exhaustive Search ...

Available options

- KEEP = *real-value* | NO
- ENFORCE = YES | NO
- CONSTRAINTS = *constraint-name*
- CONVERSE = YES | NO
- FD = NO | BARCHART | HISTOGRAM
- NCLASSES = *integer-value*
- FDFILE = *frequency-distribution-file-name*
- SCOREFILE = *file-name-for-scores*
- *REPLACE = YES | NO
- STATUS = YES | NO
- DSTATUS = *integer-value* | NONE

*Option is nonpersistent

Description of options

FD = NO | BARCHART | HISTOGRAM

Unless FD = NO is specified, a frequency distribution of tree scores is output. (Normally, obtaining the frequency distribution is the only reason for doing an exhaustive rather than branch-and-bound search, so you will probably never specify this option.) FD = BARCHART is the default so that the number of trees at each score is output. If FD = HISTOGRAM, tree scores are pooled into classes, the size of which is specified by the NCLASSES option.

NCLASSES = *integer-value*

Integer-value specifies the number of class intervals for the frequency distribution of tree scores. By default, NCLASSES=20, so that adjacent tree scores are pooled into 20 classes of tree scores. This option is only relevant when FD = HISTOGRAM.

FDFILE = *frequency-distribution-file-name*

> If FDFILE is specified, the data for the frequency distribution of tree scores are saved to a text file for input to other programs. The FDFILE option is not persistent. *Frequency-distribution-file-name* specifies the name of the file to receive the frequency distribution. See "Input/ Output files" on page 9 for instructions to specify the full path name of a file.

SCOREFILE = *file-name-for-scores*

> Use SCOREFILE to save a text file of tree scores found during the exhaustive search.

> REPLACE = YES | <u>NO</u>

> > If FDFILE or SCOREFILE options are used and the specified file(s) already exists you will be prompted for confirmation that the existing file should be replaced. REPLACE suppresses this prompt; the existing file(s) will be quietly overwritten by the new data.

See "Tree-searching options" on page 31 under "Options Affecting Multiple Commands" for a description of the following options:

> KEEP = *KEEP-SCORE*
> ENFORCE = YES | <u>NO</u>
> CONSTRAINTS = *CONSTRAINT-TREE-NAME*
> CONVERSE = YES | <u>NO</u>
> STATUS = <u>YES</u> | NO
> DSTATUS = *integer-value* | NONE

ANCSTATES

The ANCSTATES command, used to define ancestral states, is ordinarily issued from within the ASSUMPTIONS block. You may also issue it from the command line or from within a PAUP block. See "Commands used in the ASSUMPTIONS Block" on page 19 for the description of this command.

ASSUME

Use the ASSUME command to invoke a type set, weight set, or exclusion set, or to select an ancestor . The options under this command are described in the "Commands used in the ASSUMPTIONS Block" on page 19.

Syntax

> ASSUME [*options*];

Menu equivalent

> Data > Choose Assumptions Sets ...

Available options

- TYPESET = *typeset-name*
- WTSET = *wtset-name*
- EXSET = *exset-name*
- ANCSTATES = *ancstates-name*

BANDB

Use the BANDB command to search for trees using the branch-and-bound algorithm.

Syntax

BANDB [*options*];

Menu equivalent

Analysis > Branch and Bound Search ...

Available options

- UPBOUND = *real-value*
- KEEP = *real-value* | <u>NO</u>
- ENFORCE = YES | <u>NO</u>
- CONSTRAINTS = *constraint-name*
- CONVERSE = YES | <u>NO</u>
- FDONLY = YES | <u>NO</u>
- FDTYPE = <u>BARCHART</u> | HISTOGRAM
- NCLASSES = *integer-value*
- FDFILE = *frequency-distribution-file-name*
- *REPLACE = YES | <u>NO</u>
- ADDSEQ = <u>FURTHEST</u> | ASIS | SIMPLE | MAXMINI | KMAXMINI
- MULTREES = <u>YES</u> | NO
- STATUS = <u>YES</u> | NO
- DSTATUS = *integer-value* | NONE

*Option is nonpersistent

Option descriptions

UPBOUND = *upper-bound*

Use this option to specify an upper bound on the score of the best tree(s). If you do not specify an upper bound on the score of the best tree (or if you specify UPBOUND=0), PAUP* computes a starting

upper bound via the stepwise addition algorithm.

FDONLY = YES | <u>NO</u>

If FDONLY = YES, then a frequency distribution of tree scores is output for all trees less than or equal to the score specified by the KEEP option. This option is not permitted unless it is used in conjunction with the KEEP score option.

FDTYPE = <u>BARCHART</u> | HISTOGRAM

The default FDTYPE specifies that the number of trees at each tree score is output as a BARCHART. If FDTYPE = HISTOGRAM, tree scores are pooled into classes, the size of which is specified by the NCLASSES option.

NCLASSES = *integer-value*

Integer-value specifies the number of class intervals for the frequency distribution of tree scores. By default, NCLASSES = 20, so that adjacent tree scores are pooled into 20 classes of tree scores. This option is only available when FDTYPE = HISTOGRAM.

FDFILE = *frequency-distribution-file-name*

If FDFILE is specified, the data for the frequency distribution of tree scores are saved to a text file for input to other programs. The FDFILE option is not persistent. *Frequency-distribution-file-name* specifies the name of the file to receive the frequency distribution. See "Input/Output files" on page 9 for instructions to specify the full path name of a file.

REPLACE = YES | <u>NO</u>

If FDFILE is used and the specified file already exists you will be prompted for confirmation that the existing file should be replaced. REPLACE suppresses this prompt; the existing file will be quietly overwritten by the new data.

ADDSEQ = <u>FURTHEST</u> | ASIS | SIMPLE | MAXMINI | KMAXMINI

ADDSEQ specifies the way in which taxa are selected for next addition to the tree at the current node of the search tree. FURTHEST is usually the fastest, although it is not permitted unless all characters are of type ORD or UNORD (it will automatically be overridden by SIMPLE in this case).

MULTREES = <u>YES</u> | NO

Ordinarily, PAUP* saves all minimal trees it finds during the branch-and-bound search. You can use MULTREES = NO to save only one of the best trees found. If you only want to know the score of the best tree(s), use this option. The single tree found is guaranteed to be of minimum score and the search often runs much faster.

See "Tree-searching options" on page 31 under "Options Affecting Multiple Commands" for a description of the following options:

KEEP = *keep-score*
ENFORCE = YES | <u>NO</u>
CONSTRAINTS = *constraint-tree-name*
CONVERSE = YES | <u>NO</u>
STATUS = <u>YES</u> | NO
DSTATUS = *integer-value* | NONE

BASEFREQS

Use the BASEFREQS command to show base frequencies for each taxon. In addition, this command gives the expected base frequencies for each taxon and the Chi-square test of homogeneity of base frequencies across taxa.

Syntax

BASEFREQS ;

Menu equivalent

Data > Base Frequencies

BOOTSTRAP

Use the BOOTSTRAP command to perform a bootstrap analysis using either a branch-and-bound or a heuristic search.

Syntax

BOOTSTRAP   [*options*] [/ *heuristic-search-options* | *branch-and-bound-search-options*] ;

Menu equivalent

Analysis > Bootstrap/Jackknife ...

Available options

- BSEED = *integer-value*
- NREPS = *integer-value*
- SEARCH = <u>HEURISTIC</u> | BANDB | FASTSTEP | NJ | UPGMA
- CONLEVEL = *integer-value*
- KEEPALL = YES | <u>NO</u>
- WTS = IGNORE | <u>SIMPLE</u> | REPEATCNT | PROPORTIONAL
- NCHAR = <u>CURRENT</u> | *number-of-characters*
- GRPFREQ = <u>YES</u> | NO
- TREEFILE = *bootstrap-tree-file-name*
- FORMAT= <u>NEXUS</u> | ALTNEXUS | FREQPARS | PHYLIP | HENNIG

- BRLENS = YES | NO
- *REPLACE = YES | <u>NO</u>
- CUTOFFPCT = *integer-value*

*Option is nonpersistent

Description of options

BSEED = *integer-value*

By default, PAUP* references the system clock to obtain a value used to seed a random number generator. An explicit seed may be specified to override the system clock default. (See "INITSEEDS = 0 | 1" on page 116 for more details.)

NREPS = *integer-value*

Use NREPS to specify the number of bootstrap replications (resamplings) to be performed. The default is 100.

SEARCH = <u>HEURISTIC</u> | BANDB | FASTSTEP | NJ | UPGMA

A tree search is performed for each bootstrap resampling of the characters using the options specified under SEARCH. If SEARCH = FASTSTEP, tree searches in each replication are performed using one random-sequence-addition replication and no branch swapping. SEARCH = NJ | UPGMA may be used only if the distance criterion has been selected (see the "Set" command described on page 114).

CONLEVEL = *integer-value*

Use CONLEVEL to specify the minimum proportion of the bootstrap replicates (expressed as a percentage) in which a group must appear if it is to be included in the bootstrap consensus tree. For example, to obtain a bootstrap consensus tree that shows only those groups which occurred on more than 80% of the trees, you would specify CONLEVEL=80. CONLEVEL must be at least 50, which is the default.

WTS = IGNORE | <u>SIMPLE</u> | REPEATCNT | PROPORTIONAL

If WTS = IGNORE, any previously applied character weights are ignored. If WTS = SIMPLE, then characters are sampled with equal probability but character weights are applied. If WTS = REPEATCNT, then weights are treated as repeat counts. WTS = REPEATCNT is only allowed if all weights are integers. If WTS=PROPORTIONAL, then characters are sampled in proportion to their weights. WTS=PROPORTIONAL is available even when non-unit character weights are in effect.

NCHAR = <u>CURRENT</u> | *number-of -characters*

Use NCHAR to specify the number of characters to sample in each replicate. You will almost always want to use NCHAR = CURRENT, however, the option to sample more or fewer than the number of currently included characters may be useful in certain specialized contexts. Using character weights is not allowed when NCHAR is set

to a value other than the number of included characters.

GRPFREQ = <u>YES</u> | NO

Unless GRPFREQ = NO, a table of bootstrap partition frequencies is output to the display buffer.

KEEPALL = YES | <u>NO</u>

If you request KEEPALL, groups occurring at frequencies less than CONLEVEL will also be retained in the bootstrap consensus as long as they are compatible with all groups that are already included in the consensus. Effectively, this forces CONLEVEL=50, because any group occurring in 50% or more of the replicates will automatically be compatible with all more frequently occurring groups.

TREEFILE = *bootstrap-tree-file-name*

If TREEFILE is specified, a description of all trees found in each bootstrap replicate is given in the NEXUS tree format. Trees contain a weight comment for the purposed of combining bootstrap results from runs performed at different times or on different machines and for recovering results obtained prior to a system crash. The weight is the reciprocal of the number of trees found in a replicate. The *bootstrap-tree-file-name* must follow the conventions described under *"Input/Output files" on page 9*

REPLACE = YES | <u>NO</u>

If TREEFILE is used and the specified file already exists you will be prompted for confirmation that the existing file should be replaced. REPLACE suppresses this prompt and the existing file will be quietly overwritten by the new data.

FORMAT = <u>NEXUS</u> | ALTNEX | FREQPARS | PHYLIP | HENNIG

Specifies the type of treefile to be produced. NEXUS requests a file containing the standard NEXUS TREES block using a translation table (which greatly reduces the amount of disk space required to store the trees). ALTNEX also specifies a NEXUS TREES block, but no translation table is used (the full taxon names are included in each tree description). FREQPARS requests a treefile for the FREQPARS program described by Swofford and Berlocher (1987). PHYLIP requests a treefile for input to version 3.x of Felsenstein's (1991) PHYLIP package. HENNIG requests a treefile for version 1.5 of Farris's (1988) Hennig86 program.

BRLENS = YES | <u>NO</u>

If BRLENS = YES is specified, the descriptions of the trees saved in the bootstrap tree file will include branch lengths if the program corresponding to the FORMAT setting supports them. BRLENS = NO reverses the effect of a previous BRLENS specification. Note, the BRLENS option pertains only to those trees saved in the bootstrap tree file. Invoking this option will have no effect on the bootstrap consensus tree.

CUTOFFPCT = *integer-value*

Use CUTOFFPCT to specify the minimum bootstrap partition frequency to be displayed in the partition frequency table. Partitions occurring at a frequency below this value are not shown.

Example

In the following example the BOOTSTRAP command is used to execute a heuristic search with random addition sequence on 200 bootstrap replicate data sets. Optimal trees for each replicate are saved to a file named boot.tre.

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

CD

Use the CD command to change the current working directory for the Windows, Unix and DOS versions.

Syntax

CD *<new-directory>*;

Entering the command with no arguments simply reports the current directory.

CHARPARTITION

Use the CHARPARTITION to define a partition of the characters.

Syntax

CHARPARTITION *partition-name=partition-definition*;

The CHARPARTITION command is ordinarily issued from within the SETS block described on page 22. However, you may also issue it from the command line or from within a PAUP block.

CHARSET

Use the CHARSET command to define a "character set." Character sets are simply groups of characters that can be referred to by a single name in other commands.

Syntax

CHARSET *character-set-name=character-list*;

The *character-set-name* must not be identical to any of the original character names.

The CHARSET command is ordinarily issued from within the SETS block described on page 22. However, you may also issue it from the command line or from within a PAUP block.

CLEARTREES

Use the CLEARTREES command to empty the tree buffer.

Syntax

CLEARTREES [NOWARN = YES | NO];

Menu equivalent

Trees > Clear Trees

Ordinarily you will be prompted for conformation. If you specify NOWARN=YES, the tree buffer is silently emptied.

CONDENSE

Use the CONDENSE command to collapse zero-length branches into polytomies for all trees and then keep only those trees that are unique after the collapsing is accomplished.

Syntax

CONDENSE [*options*];

Menu equivalent

Tree > Condense Trees ...

Available options

- COLLAPSE = NO | MAXBRLEN | MINBRLEN | AMBEQUAL
- DELDUPES = YES | NO

Description of Options

COLLAPSE = NO | MAXBRLEN | MINBRLEN | AMBEQUAL

Unless COLLAPSE = NO is specified, branches are collapsed according to the specified criterion. MAXBRLEN is the method used in PAUP version 3 (and Hennig86). MINBRLN collapses a branch if it is possible for it to have zero length. AMBEQUAL collapses a branch if the MPRset of the two incident nodes are identical. These latter two

methods were introduced in Goloboff's NONA program. We do not recommend their use, but they are available for those who wish to use them.

DELDUPES = <u>YES</u> | NO

Unless DELDUPES is specified, duplicate trees will be eliminated.

CONSTRAINTS

Use the CONSTRAINTS command to define a constraint tree.

Syntax

> CONSTRAINTS *constraint-name* [(BACKBONE) | (MONOPHYLY) |
> (CONVEXITY)]=*tree-specification* | *taxpartition-name*;

The *tree-specification* must follow the tree format described on page 25. The *taxpartition-name* must follow the format described under TAXPARTITION on page 24.

Example 1

The following example assumes that there are 10 taxa, and taxa 2, 3, 5, 7 and 9 are a monophyletic assemblage.

```
constraints lagomorph (monophyly) =
(1,4,6,8,10,(2,3,5,7,9));
```

Example 2

Because monophyly is the default keyword indicating the constraint type and taxa 1, 4, 6, 10 connect directly to the root node, the above example could also be written:

```
constraints lagomorph = ((2,3,5,7,9));
```

CONTREE

Use the CONTREE command to request computation of strict, semistrict (combinable component), Adams, and/or majority-rule consensus trees .

Syntax

> CONTREE [*tree-list*][/*options*];

The tree list specifies which trees to include in the consensus; the default is "ALL."

Menu equivalent

Trees > Compute Consensus

Available options

- PERCENT = *integer-value*
- LE50 = YES | <u>NO</u>
- STRICT = <u>YES</u> | NO
- SEMISTRICT = YES | <u>NO</u>
- MAJRULE = YES | <u>NO</u>
- ADAMS = YES | <u>NO</u>
- USETREEWTS = YES | <u>NO</u>
- INDICES = YES | <u>NO</u>
- GRPFREQ = <u>YES</u> | NO
- SHOWTREE = <u>YES</u> | NO
- TREEFILE = *tree-file-name*
- *REPLACE = YES | <u>NO</u>
- *APPEND = YES | <u>NO</u>
- TCOMPRESS = YES | <u>NO</u>
- ROOT = <u>OUTGROUP</u> | LUNDBERG | MIDPOINT
- OUTROOT = <u>POLYTOMY</u> | PARAPHYL | MONOPHYL

*Option is nonpersistent

Description of options

STRICT = <u>YES</u> | NO

By default, a strict consensus tree is computed. Use STRICT = NO to suppress this computation.

SEMISTRICT = YES | <u>NO</u>

Specify SEMISTRICT = YES to request computation of a semistrict (combinable component) consensus tree. SEMISTRICT = NO reverses the effect of a previous SEMISTRICT specification.

MAJRULE = YES | <u>NO</u>

Specify MAJRULE = YES to request computation of a semistrict (combinable component) consensus tree. MAJRULE = NO reverses the effect of a previous MAJRULE specification.

The following options apply only if MAJRULE is in effect:

PERCENT = *integer-value*

The *integer-value* specifies the percentage of the trees on which a group must appear in order to be retained in the majority-

rule consensus. A group must appear on more than this percentage of the trees before it is retained. The default is 50.

LE50 = YES | N<small>O</small>

If LE50 = YES is specified, groups occurring on less than 50% of the trees are retained in the consensus if they are compatible with the groups already on the tree. LE50 = N<small>O</small> reverses the effect of a previous LE50 specification.

G<small>RP</small>F<small>REQ</small> = Y<small>ES</small> | N<small>O</small>

By default, a table is output that shows all partitions (or groups) occurring on at least one tree and the frequency of each such group. G<small>RP</small>F<small>REQ</small> = N<small>O</small> can be used to suppress this output.

A<small>DAMS</small> = Y<small>ES</small> | N<small>O</small>

Specify A<small>DAMS</small> = Y<small>ES</small> to request computation of an Adams consensus tree. A<small>DAMS</small> = N<small>O</small> reverses the effect of a previous A<small>DAMS</small> specification.

U<small>SE</small>T<small>REE</small>W<small>TS</small> = Y<small>ES</small> | N<small>O</small>

Trees output to a treefile in bootstrap and jackknife contain a weight comment (see page 25). These weights are the reciprocal of the number of trees found in a replicate. The "GetTrees" command described on page 67 will optionally store these weights, and the majority-rule consensus calculator will use them when U<small>SE</small>T<small>REE</small>W<small>TS</small> = Y<small>ES</small>. This allows calculation of majority-rule consensus trees that correspond exactly to the consensus tree output by the original bootstrap or jackknife command. This option allows the combination of bootstrap results from runs performed at different times or on different machines and the recovery of results obtained prior to a system crash. Also, it permits graphics-quality output of bootstrap trees after redefining an outgroup or otherwise changing rooting options by rereading the trees (with weights) and computing/printing a majority-rule consensus tree.

Example

The following example recalls bootstrap trees saved during three seperate bootstrap executions. See page 67 for information regarding the G<small>ET</small>T<small>REES</small> command. By including the U<small>SE</small>T<small>REE</small>W<small>TS</small> option, the C<small>ON</small>T<small>REE</small> command will display the bootstrap majority rule tree for the combined trees contained in the three seperate bootstap tree files.

```
gettrees file=tfile1.out StoreTreeWts=yes mode=3;
gettrees file=tfile2.out StoreTreeWts=yes mode=7;
gettrees file=tfile3.out StoreTreeWts=yes mode=7;
contree all/strict=no majrule=yes usetreewts=yes;
```

I<small>NDICES</small> = Y<small>ES</small> | N<small>O</small>

Specify I<small>NDICES</small> to request calculation of a variety of consensus

indices. The following indices are computed by PAUP* and are described in Rohlf (1982) and Swofford (1991):

Component information (consensus fork)
Nelson-Platnick term information
Nelson-Platnick total information
Mickevich's consensus information
Colless weighted consensus fork (proportion max. information)
Schuh-Farris levels sum
Rohlf's CI(1)
Rohlf's -ln CI(2)

SHOWTREE = YES | NO

Unless SHOWTREE = NO, the consensus tree (or trees if more than one consensus method is specified) is printed to the display buffer.

TREEFILE = *file-specification*

If TREEFILE is specified, a description of all consensus trees computed is output to a file containing a NEXUS-format TREES block. This option is not persistent; you must specify it on every CONTREE command for which you want a tree file to be saved. See "Input/Output files" on page 9 for instructions to specify the full path name of a file.

APPEND = { YES | NO }
REPLACE = { YES | NO }

If the specified file already exists, you will be prompted for confirmation that the existing file should be replaced. Explicit specification of REPLACE suppresses this prompt and the existing file will be quietly overwritten by the new data. Alternatively, you may specify APPEND, in which case a new TREES block will be concatenated to the end of an existing file.

See "Options And Subcommands Affecting Multiple Commands" on page 30 for a description of the following options:

ROOT = OUTGROUP | LUNDBERG | MIDPOINT
OUTROOT = POLYTOMY | PARAPHYL | MONOPHYL
TCOMPRESS = YES | NO

CSTATUS

Use the CSTATUS command to request a listing of character-status information for all characters.

Syntax

CSTATUS [*options*];

Menu equivalent

Data > Show Character Status ...

Available options

- FULL = YES | <u>NO</u>
- EXCLUDED = <u>SHOW</u> | HIDE

Description of options

FULL = YES | <u>NO</u>

For each character, the following information is output if FULL = YES:

1. The number and name (if any) of the character.

2. The character's current type.

3. If the character is constant (invariant), excluded, or uninformative.

4. The character's current weight. The weight value is put into parentheses if the character is excluded.

5. A list of the states observed for the character.

EXCLUDED = <u>SHOW</u> | HIDE

If EXCLUDE = HIDE then only the status of the currently included characters will be displayed.

CTYPE

Use the CTYPE command to assign character types to characters.

Syntax

CTYPE *type-name:character-list*[, *type-name:character-list*]...;

The type-name must be one of the standard character types (ORD, UNORD, DOLLO, DOLLO.UP, DOLLO.DN, IRREV, IRREV.UP, or IRREV.DN) or the name of a user-defined character type (see, "UserType" on page 20). Each character-list consists of one or more character number, character name (see, "CharLabels" on page 16), or character set name (see, "CharSet" on page 23). The characters specified by character-list are assigned the immediately preceding character-type. Any number of character-type:character-list pairs, separated by commas, may be specified.

Menu equivalent

Data > Set Character Types ...

Example

The following example of the Ctype command assigns the user-defined character type "tcost" to only characters at the 3rd codon position. See page 42 for details regarding the CHARSET command.

```
charset 3rdpos:3-.\3;
ctype tcost:3rdpos;
```

DEFAULTS

Use the DEFAULTS command to specify default option settings for another command. The options set using this command do not persist after you quit the application.

Syntax

DEFAULTS *command-name default-option-setting. . . ;*

Options for the following commands may be set:

ALLTREES
BANDB
BOOTSTRAP
CONTREE
DESCRIBE
HSEARCH
LAKE
LENFIT
RANDTREES
SAVETREES

Example

```
defaults hsearch swap=nni addseq=closest;
hsearch;
```

is equivalent to:

```
hsearch swap=nni addseq=closest;
```

DELETE

Use the DELETE command to delete a taxon from subsequent analyses.

Syntax

DELETE *taxon-list* [/[ONLY][*options*]];

Unless ONLY is specified, taxa specified in the taxon-list are simply added to the set of currently deleted taxa. If taxa have already been deleted and

you want only those taxa specified in taxon-list to remain deleted, specify ONLY; any currently deleted taxa not explicitly specified in the list will be restored.

Menu Equivalent

Data > Delete-Restore Taxa ...

Available options

- *PRUNE = YES | <u>NO</u>
- *CLEARTREES = YES | <u>NO</u>
- CONDENSE = YES | <u>NO</u>

*Option is nonpersistent

Description of options

The following three options pertain only if there are trees in memory that will become invalidated by the deletion of taxa. If you do not specify PRUNE or CLEARTREES, the program will prompt for your desired action. PRUNE and CLEARTREES are mainly useful for batch file processing, where you do not want the program to stop and wait for a response from the user before continuing.

PRUNE = YES | <u>NO</u>

If PRUNE = YES is specified, newly deleted taxa will be removed ("pruned") from the trees currently in memory, which otherwise remain unmodified.

CLEARTREES = YES | <u>NO</u>

If CLEARTREES is specified, any trees currently in memory are simply deleted.

CONDENSE = YES | <u>NO</u>

If CONDENSE is specified, any duplicate trees that result from the removal of taxa (PRUNE option) are deleted.

DEROOTTREES

Use the DEROOTTREES command to convert all trees in memory from a rooted to an unrooted representation. The circumstances under which you would need to use this command are rather unlikely.

Syntax

DEROOTTREES ;

Menu equivalent

Trees > Deroot Trees

DESCRIBETREES

Use the DESCRIBETREES command to output tree diagrams and associated information. Information regarding tree length is given according to the currently selected optimality criterion.

Syntax

DESCRIBETREES [*tree-list*] [/*options*];

The tree-list specifies the numbers of the trees you wish to describe. If this is the first DESCRIBETREES command and you do not specify a tree list, only the first tree is described.

Menu equivalent

Trees > Describe Trees ...

Available options

- PLOT = NONE | <u>CLADOGRAM</u> | PHYLOGRAM | BOTH
- ROOT = <u>OUTGROUP</u> | LUNDBERG | MIDPOINT
- OUTROOT = <u>POLYTOMY</u> | PARAPHYL | MONOPHYL
- BRLENS = YES | <u>NO</u>
- LABELNODE = <u>YES</u> | NO
- XOUT = <u>NONE</u> | TERMINAL | INTERNAL | BOTH
- MPRSETS = YES | <u>NO</u>
- APOLIST = YES | <u>NO</u>
- CHGLIST = YES | <u>NO</u>
- PATRISTIC = YES | <u>NO</u>
- HOMOPLASY = YES | <u>NO</u>
- FVALUE = YES | <u>NO</u>
- DIAG = YES | <u>NO</u>
- TCOMPRESS = YES | <u>NO</u>
- CMLABELS = YES | <u>NO</u>
- CMSHOWEQ = <u>YES</u> | NO
- CMCOLWID = *integer-value*
- CMCSTATUS = YES | <u>NO</u>

Description of options

PLOT = NONE | <u>CLADOGRAM</u> | PHYLOGRAM | BOTH

If PLOT = CLADOGRAM, branch lengths on the tree have no meaning, and taxa are aligned at the right edge of the diagram. If PLOT = PHYLOGRAM, branch lengths are drawn proportionally to the number of changes assigned to each branch. PLOT = BOTH requests output of the tree diagram in both CLADOGRAM and PHYLOGRAM formats.

BRLENS = YES | <u>NO</u>

Requests output of a table of assigned, minimum-possible, and maximum-possible branch lengths.

CHGLIST = YES | <u>NO</u>

Requests output of a list of changes in each character.

APOLIST = YES | <u>NO</u>

If set to YES a list of the apomorphic characters is displayed. The list includes the characters that have changed along each branch, the consistency index for each character, and the original and transformed character. The double arrow "==>" under the change column represents unambiguous changes and the single arrow "-->" represents ambiguous changes.

DIAG = YES | <u>NO</u>

Requests output of character diagnostics, including the minimum-possible, assigned, and maximum-possible length of each character, and goodness-of-fit measures based on these quantities.

PATRISTIC = YES | <u>NO</u>

Requests output of the patristic distance matrix.

HOMOPLASY = YES | <u>NO</u>

Requests output of the pairwise homoplasy matrix.

LABELNODE = <u>YES</u> | NO

Ordinarily, internal nodes on the tree diagram are labeled with a node number that is referenced by other output information (change lists, apomorphy lists, etc.). LABELNODE = NO can be used to suppress the labeling of internal nodes.

XOUT = <u>NONE</u> | TERMINAL | INTERNAL | BOTH

Requests output of a table of character-state assignments for each tree. XOUT = INTERNAL requests output of the character-states assigned to internal nodes for each tree. XOUT = TERMINAL requests a listing of the original data matrix. (This option is not particularly useful, as you would ordinarily use the SHOWMATRIX command to list the data matrix.) XOUT = BOTH requests a listing of the original

data matrix plus the states assigned to internal nodes.

MPRSETS = YES | <u>NO</u>

Requests a listing of the possible character-state assignments (MPR-sets) for each tree (parsimony criterion only).

FVALUE = YES | <u>NO</u>

Requests output of the F-value and F-ratio.

See "Options And Subcommands Affecting Multiple Commands" on page 30 for details.

ROOT = <u>OUTGROUP</u> | LUNDBERG | MIDPOINT
OUTROOT = <u>POLYTOMY</u> | PARAPHYL | MONOPHYL
TCOMPRESS = YES | <u>NO</u>
CMLABELS = <u>YES</u> | NO
CMSHOWEQ = YES | <u>NO</u>
CMCSTATUS = YES | <u>NO</u>
CMCOLWID = *column-width*

DOS

Use the DOS command to execute a DOS command and return immediately to PAUP* or to enter a DOS shell.

Syntax

DOS [*dos-command*];

If you do not specify a command, PAUP* will open a DOS shell. Type `exit` at the DOS prompt when you want to return to PAUP*. If you specify a command, control is returned to PAUP* immediately after the command has finished. You can type a command that contains a semicolon by enclosing the entire command within single-quotes.

DSCORES

Use the DSCORES command to calculate scores of trees in memory according to distance criterion. The *tree-list* indicates which trees are to be evaluated.

Syntax

DSCORES [*tree-list*][/*options*];

Menu equivalent

Trees > Tree Scores > Distance ...

Available options

- *SORTTREES = <u>NO</u> | YES
- SCOREFILE = *file-name-for-scores*
- *REPLACE = <u>NO</u> | YES
- *APPEND = <u>NO</u> | YES
- *DISPLAYOUT   <u>NO</u> | YES
- DISTANCE = USER | TOTAL | MEAN | ABS | <u>P</u> | JC | F81 | TAJNEI | K2P | F84 | HKY85 | K3P | TAMNEI | GTR | CUSTOM | ML | LOGDET | UPHOLT | NEILI
- RATES = <u>EQUAL</u> | GAMMA
- SHAPE = *real-value*
- PINVAR = *real-value*
- REMOVEFREQ = <u>PROPORTIONAL</u> | EQUAL
- ESTFREQ = <u>ALL</u> | CONSTANT
- ALLSITESMEAN = <u>YES</u> | NO
- SUBST = <u>ALL</u> | TV | TI | TRATIO
- CLASS = (*CAC CAG CAT CCG CCT CGT*)
- BASEFREQ = EQUAL | <u>EMPIRICAL</u>
- MISSDIST = <u>INFER</u> | IGNORE
- OBJECTIVE = <u>ME</u> | LSFIT
- POWER = *integer-value*
- NEGBRLEN = PROHIBIT | ALLOW | <u>SETZERO</u> | SETABSVAL
- DCOLLAPSE = YES | <u>NO</u>
- WTS = <u>REPEATCNT</u> | IGNORE

Description of options

SORTTREES = <u>NO</u> | YES

> When multiple trees are evaluated under the DSCORES command SORTTREES = YES will sort tree scores so that they are displayed in ascending order.

SCOREFILE = *file-name-for-scores*

> If SCOREFILE is specified, distance scores under the current objective function for each tree evaluated is saved to a text file. The *file-name-for-scores* must follow the conventions described under "Input/Output files" on page 9.

> > REPLACE = YES | <u>NO</u>
> > APPEND = YES | <u>NO</u>

> > > If the *file-name-for-scores* already exists, you will be prompted for confirmation before the existing file is replaced. Use REPLACE = YES to suppress this prompt; the existing file will

be quietly overwritten by the new data. Alternatively, you may specify APPEND, in which case a new TREES block will be concatenated to the end of an existing file.

DISPLAYOUT   NO | YES

Set DISPLAYOUT = YES if you wish to suppress output to the display buffer.

The remaining options for this command are the same as those used under the DSET command. See page 55 for a description of these options.

DSET

Use the DSET command to set options for distance analysis.

Syntax

DSET [*options*];

Menu equivalent

Analysis > Distance Settings ...

Available options

- DISTANCE = USER | TOTAL | MEAN | ABS | <u>P</u> | JC | F81 | TAJNEI | K2P | F84 | HKY85 | K3P | TAMNEI | GTR | CUSTOM | ML | LOGDET | UPHOLT | NEILI
- RATES = <u>EQUAL</u> | GAMMA
- SHAPE = *real-value*
- PINVAR = *real-value*
- REMOVEFREQ = <u>PROPORTIONAL</u> | EQUAL
- ESTFREQ = <u>ALL</u> | CONSTANT
- ALLSITESMEAN = <u>YES</u> | NO
- SUBST = <u>ALL</u> | TV | TI | TRATIO
- CLASS = (*cAC cAG cAT cCG cCT cGT*)
- BASEFREQ = EQUAL | <u>EMPIRICAL</u>
- MISSDIST = <u>INFER</u> | IGNORE
- OBJECTIVE = <u>ME</u> | LSFIT
- POWER = *integer-value*
- NEGBRLEN = PROHIBIT | ALLOW | <u>SETZERO</u> | SETABSVAL
- DCOLLAPSE = YES | <u>NO</u>
- WTS = <u>REPEATCNT</u> | IGNORE

Description of options

DISTANCE = USER | TOTAL | MEAN | ABS | P | JC | F81 | TAJNEI |
K2P | F84 | HKY85 | K3P | TAMNEI | GTR |
CUSTOM | ML | LOGDET | UPHOLT | NEILI

Use the DISTANCE option to specify the distance measure used to calculate the scores of the trees in memory. The following distances may be used:

Distance	Definition
USER	user-defined distance (see "Commands used in the DISTANCES Block" on page 27)
TOTAL	total number of pairwise character differences
MEAN	mean number of pairwise character differences (adjusted for missing data)
ABS	absolute distances
P	uncorrected distance often referred to as the *p*-distance or dissimilarity (*D*) distance.
JC	Jukes and Cantor (1969); equal base frequencies, one substitution types
F81	Felsenstein (1981); unequal base frequencies, one substitution type
TAJNEI	Tajima and Nei (1984); unequal base frequencies, one substitution type
K2P	Kimura (1980); equal base frequencies, unequal Ti:Tv
F84	Felsenstein (1984); unequal base frequencies, unequal Ti:Tv
HKY85	Hasegawa, Kishino, and Yano (1985); unequal base frequencies, unequal Ti:Tv
K3P	Kimura (1981); equal base frequency; 3 substitution types
TAMNEI	Tamura and Nei (1993); unequal base frequency; 3 substitution types
GTR	general time-reversible model (Lanave et al. 1984, and Rodriquez et al. 1990) ; unequal base frequencies, 6 substitution types.
CUSTOM	user defines the base frequencies and number of substitution types
ML	maximum likelihood estimator of distance under current maximum likelihood settings

Distance	Definition
LOGDET	log-determinant (Steel, 1994; Lockhart et al., 1994) or paralinear (Lake, 1994) distances
UPHOLT	restriction-site distance of Upholt (1977)
NEILI	restriction-site distance of Nei and Li (1979)

Example 1

The following example of the DSET command demonstrates one of the ways to specify a maximum likelihood model when the DISTANCE option is set to ML. On line two the LSCORES (page 83) command is used to estimate all six substitutions rate classes and the gamma shape parameter on the neighbor joining (see page 94) generated from line one. The LSET (page 87) command is then used to fix the parameter values to those estimated by the LSCORES command.

```
nj;
lscore 1/nst=6 rmatrix=estimate rates=gamma
        shape=estimate;
lset rmatrix=previous shape=previous;
dset distance=ml;
```

The following set of options apply only to the calculation of pairwise distances:

RATES = <u>EQUAL</u> | GAMMA

Unless RATES = GAMMA, rates across all characters are assumed to be equal. If RATES = GAMMA, rates across sites are assumed to follow a continuous gamma distribution. The shape of the gamma distribution is equal to the value specified for the SHAPE option. This option is not available for ABS, P, LOGDET, or ML distances. To specify gamma distributed rates for ML distances use the "LSet" command described on page 87.

SHAPE = *real-value*

Use SHAPE to specify the shape of the gamma distribution used to accommodate among-site rate variation. Again, this option is not available for ABS, P, LOGDET, or ML distances. To specify the gamma-shape parameter for ML distances use the "LSet" command described on page 87.

PINVAR = *real-value*

Use PINVAR to specify the proportion of invariable sites (i.e., sites that are unable to accept substitutions); the remaining sites are assumed to vary at the same rate.

The following options are available only if a *real-value* has been specified under PINVAR:

REMOVEFREQ = <u>PROPORTIONAL</u> | EQUAL

> If REMOVEFREQ = EQUAL, the same number of identical sites is removed from each nucleotide in a given pairwise comparison. If REMOVEFREQ = PROPORTIONAL, then the number of identical sites removed is proportional to the frequency of each base.

ESTFREQ = <u>ALL</u> | CONSTANT

> If REMOVEFREQ = PROPORTIONAL, then the frequency of each base may be estimated from the entire data set (EST-FREQ = ALL) or only the invariable sites (ESTFREQ = CON-STANT).

ALLSITESMEAN = <u>YES</u> | NO

> Adjust the distance so that it represents the mean number of substitutions over all sites, rather than over only the variable sites.

SUBST = <u>ALL</u> | TV | TI | TRATIO

> By default PAUP* counts or estimates substitutions of all types (SUBST = ALL). For many DNA distances, the distance represents the mean number of substitutions per site that have accumulated since a pair of taxa last shared a common ancestor. You can restrict PAUP* to count or estimate changes representing transition, changes from a purine (A or G) to pyrimidine (C or T) (SUBST = TI) or transversion, changes from a purine to a purine and pyrimidine to a pyrimidine (SUBST = TV). Finally, PAUP* will output the transition/traversion ratio for the distance chosen when SUBST = TRATIO.

CLASS = (*cAC cAG cAT cCG cCT cGT*)

> If DISTANCE = CUSTOM, then substitution types may be assigned to any of six classes. For example, substitution types may be pooled into common classes to reflect changes between structurally similar molecules (e.g. transitions and transversions). A user is free to decide among the many possible ways that substitution types may be divided or combined.

Example 2

The following example pools transitions into a common class and defines two seperate classes for transversions.

```
dset distance=custom class=(a b a a c a);
```

BASEFREQ = EQUAL | <u>EMPIRICAL</u>

> If DISTANCE = CUSTOM, base frequencies used to estimated pairwise distances may be based on either the empirical base frequency or equal base frequencies (i.e., frequency of each base equals 1/4).

MISSDIST = <u>INFER</u> | IGNORE

> Unless MISSDIST = IGNORE, PAUP* infers "missing-data" sites (including gaps) and ambiguity codes by distributing them proportionately to unambiguous changes. You may also choose to ignore sites for the affected pairwise comparison (IGNORE).

The following set of options apply to the calculation of branch lengths and tree scores:

OBJECTIVE = <u>ME</u> | LSFIT

> Choose between minimum evolution (ME) and least-squares (LSFIT) objective functions. See Swofford et al. (1996) for a discussion of these objective functions.

> > POWER = *integer-value*

> > Sums of squared deviations in least-squares analyses (OBJECTIVE = LSFIT) may be unweighted (POWER = 0), weighted by the inverse (POWER = 1) of the distance, or weighted by the squared inverse (POWER = 2). POWER = 2 corresponds with the method of Fitch and Margolish (1967).

Example 3

The following use of the DSET command corresponds to the method of Fitch and Margolish (1967).

```
dset distance=jc objective=lsfit power=2;
```

NEGBRLEN = PROHIBIT | ALLOW | <u>SETZERO</u> | SETABSVAL

> Use NEGBRLEN to specify negative branch length handling. If NEGBRLEN = PROHIBIT, then PAUP* constrains branch lengths to be nonnegative. By default, PAUP* allows but sets to zero negative branch lengths when computing the value of the objective function (SETZERO). You may also allow negative branch lengths (ALLOW) or an absolute value (SETABSVAL) when computing the value of the objective function.

DCOLLAPSE = YES | <u>NO</u>

> Use DCOLLAPSE to collapse branches of effectively zero length when searching. If a branch has a length less than or equal to 1×10^{-8}, it is eliminated from the tree.

WTS = <u>REPEATCNT</u> | IGNORE

> If WTS=IGNORE, character weights are ignored. The default (WTS = REPEATCNT) treats integer weights as repeat counts.

EDIT

Use the EDIT command to edit a file using PAUP*'s editor.

Syntax

> EDIT [*file-specification*];

This command is only valid in implementations of PAUP* that provide built-in editing capabilities.

EXCLUDE

Use the EXCLUDE command to exclude one or more characters from tree-score calculations.

Syntax

> EXCLUDE *character-list* [/ONLY];

Unless /ONLY is specified, characters specified in the character-list are simply added to the set of currently excluded characters. If some characters have already been excluded but you want only those characters specified in *character-list* to remain excluded, specify /ONLY; any currently excluded characters not explicitly specified in the list will be re-included. PAUP* includes several predefined characters that may be used in conjunction with or without a character-list. See "CharSet" command described on page 23 for more details.

Menu equivalent

> Data > Include-Exclude Characters ...

EXECUTE

Use the EXECUTE command to request processing of an input file. The input file should be a valid NEXUS file.

Syntax

> EXECUTE *file-specification*;

See "Input/Output files" on page 9 for details regarding the format of *file-specification*

The input file may contain any or all of the following: TAXA blocks, CHARACTERS blocks, DATA blocks, ASSUMPTIONS blocks, TREES blocks, DISTANCES blocks, CODONS blocks, and valid PAUP commands. Although not required, you should place PAUP commands inside of a PAUP block so that other programs (e.g., MacClade) can use the same file. PAUP commands are processed exactly as if they had been entered from the command line, with only a few exceptions (e.g., the EDIT command cannot be issued from a file). Commands are processed until the end of the file is reached, or a QUIT command is encountered.

Menu equivalent

File > Execute "*your-data-file*"

EXPORT

Use the EXPORT command to save data in a non-NEXUS format.

Syntax

EXPORT [*options*];

Menu equivalent

File > Export Data ...

Available options

- FILE = *export-file-name*
- FORMAT = <u>PHYLIP</u> | HENNIG | TABTEXT | TEXT | NEXUS |
- TREES = YES | <u>NO</u>
- LINEBREAKS = <u>NATIVE</u> | MACINTOSH | DOSWINDOWS | UNIX
- CHARSPERLINE = *integer-value* | <u>ALL</u>
- INTERLEAVED = YES | <u>NO</u>
- MSTAXAHENNIG = <u>BINCODE</u> | MISSING
- UNORDPHYLIP = <u>EXCLUDE</u> | UNMODIFIED
- ORDPHYLIP = EXCLUDE | <u>BINCODE</u> | UNMODIFIED
- *REPLACE = YES | <u>NO</u>

*Option is nonpersistent

Description of options

FORMAT = <u>PHYLIP</u> | HENNIG | TABTEXT | TEXT | NEXUS

The FORMAT option specifies the format in which data are saved to the export file. PHYLIP requests data to be output to version 3.4 of Felsenstein's (1991) PHYLIP package. HENNIG requests the output format for version 1.5 of Farris's (1988) Hennig86 program. Data can also be saved as either tab delimited text (TABTEXT) or simple text (Text). NEXUS requests that the data be stored in the standard NEXUS TAXA and CHARACTER blocks.

TREES = YES | <u>NO</u>

If either the PHYLIP or HENNIG format is choosen TREES=YES will save the trees currently in memory and the data set to the export file.

LINEBREAKS = <u>NATIVE</u> | MACINTOSH | DOSWINDOWS | UNIX

>   Specifies the line termination used in the output file.

CHARSPERLINE = *integer-value* | <u>ALL</u>

>   Specifies the number of characters written per line.

INTERLEAVED = YES | <u>NO</u>

>   If INTERLEAVED = YES, data are stored in "blocks" , which contain
>   information for some of the characters and all of the taxa.

MSTAXAHENNIG = <u>BINCODE</u> | MISSING

UNORDPHYLIP = <u>EXCLUDE</u> | UNMODIFIED

ORDPHYLIP = EXCLUDE | <u>BINCODE</u> | UNMODIFIED

FILE = *export-file-name*

>   Specify the *export-file-name* containing non-NEXUS-formatted data.
>   (See "Input/Output files" on page 9 for details regarding the file-
>   name format).

>   >   *REPLACE = YES | <u>NO</u>

>   >   If REPLACE is not specified explicitly and the file already
>   >   exists, you will receive a warning and will have the
>   >   opportunity to cancel the command before the contents of the
>   >   existing file are erased.

EXSET

The EXSET command, used to define "exclusion sets," is ordinarily issued
from within the ASSUMPTIONS block. However, you may also issue it from
the command line or from within a PAUP block. See "Commands used in
the ASSUMPTIONS Block" on page 19 for the description of this command.

FILTER

Use the FILTER command to filter trees according to score, constraints, or
other criteria.

Syntax

>   FILTER [*options*] | OFF | PURGE ;

If OFF is specified, trees that do not satisfy the specified filtering criteria are retained. If PURGE is specified, trees that are currently hidden by a tree filter will be deleted from memory.

Menu equivalent

Trees > Filter

Available options

- REVERSE = YES | <u>NO</u>
- *BEST = YES | <u>NO</u>
- MAXSCORE = *real-value*
- MINSCORE = *real-value*
- NUMLE = *integer-value*
- NUMGE = *integer-value*
- CONSTRAINTS = *constraint-name*
- SD = *integer-value*
- FROM = *integer-value*
- *LESSRESOLV = YES | <u>NO</u>
- PERMDEL = YES | <u>NO</u>
- ALWAYS = YES | <u>NO</u>
- BINARY = YES | <u>NO</u>

*Option is nonpersistent

Description of options

Any combination of the following filtering criteria can be specified.

REVERSE = YES | <u>NO</u>

If REVERSE = YES, the condition of the specified filter is reversed.

MAXSCORE = *real-value*

Use MAXSCORE to specify the maximum score of trees retained by the filter. Trees with higher (worse) scores are not kept.

Example

In line one of the following example the FILTER command retains trees with scores less than or equal to 500. The second line retains trees with scores greater than or equal to 500.

```
filter maxscore=500;
filter maxscore=500 reverse=yes;
```

MINSCORE = *real-value*

> Use MINSCORE to specify the minimum score of trees retained by the filter. Trees with lower (better) scores are not kept.

BEST = YES | <u>NO</u>

> If BEST=YES only those trees that are optimal according to the current criterion will be retained.

NUMLE = *integer-value*

> Keep trees only if their position in memory (tree number) is less than or equal to the specified *integer-value*.

NUMGE = *integer-value*

> Keep trees only if their position in memory (tree number) is greater than or equal to the specified *integer-value*.

CONSTRAINTS = *constraint-name*

> Only those trees that are compatible with the specified constraint tree are retained.

SD = *integer-value* FROM = *integer-value*

> Only those trees within *integer-value* symmetric difference distance units of the specified tree number are retained.

LESSRESOLV = YES | <u>NO</u>

> Keep a tree only if more highly resolved compatible trees do not exist.

PERMDEL = YES | <u>NO</u>

> Permanently delete those trees not retained by the filter.

ALWAYS = YES | <u>NO</u>

> Apply filter even when no trees satisfy the filtering criteria.

BINARY = YES | <u>NO</u>

> Keep a tree only if none of its internal nodes are greater than degree three.

FACTORY

Use the FACTORY command to reset all options to factory default settings.

Syntax

FACTORY ;

Menu equivalent

Options > Restore Option Settings ...

FSTATUS

Use the FSTATUS command to show current status of data, log, and tree file.

Syntax

FSTATUS;

GAMMAPLOT

Use the GAMMAPLOT command to show a gamma- distribution plot.

Syntax

GAMMAPLOT [*options*];

Menu equivalent

Options > Gamma Distribution ...

Available options

- SHAPE = *real-value* | ESTIMATE | PREVIOUS
- NCAT = *integer-value*
- REPRATE = <u>MEAN</u> | MEDIAN

Description of options

SHAPE = *real-value* | *Previous*

If SHAPE = PREVIOUS, the plot of the gamma distribution is drawn using the most recently estimated shape parameter (see the "LSet" command described on page 87) . To obtain a plot of the gamma distribution with a user-defined shape parameter specify SHAPE = *real-value.* PAUP* uses the default value of 0.5 for this option.

NCAT = *integer-value*

Use NCAT to specify the number of categories into which the continuous gamma distribution is divided in order to obtain a discrete approximation. The default is 4.

REPRATE = <u>MEAN</u> | MEDIAN}

Use REPRATE to specify whether the mean or the median is used represent the rates within each category of the gamma distribution.

GENERATE TREES

Use the GENERATE TREES command to generate a set of random trees or all-possible trees.

Syntax

GENERATE TREES {ALL | RANDOM} [*options*];

If ALL is specified, all possible trees given the total number of terminally labelled taxa are drawn. Depending on the version of PAUP* that you are using, the ALL modifier may reach the upper limit on the number of trees that can be stored. For example, if only unrooted strictly bifurcating trees are drawn, then a maximum of 8 taxa for 16-bit systems, 12 taxa for 32-bit systems, and 19 taxa for 64-bit system can be included in the analysis. When Random is specified, PAUP* will draw trees according to a Equiprobable or Markovian model (see below).

Menu equivalent

Trees > Generate Trees ...

Available options

- MODEL = <u>EQUIPROBABLE</u> | MARKOVIAN
- NTREES = *integer-value*
- SEED = *integer-value*
- ROOTED = YES | <u>NO</u>
- NONBINARY = YES | <u>NO</u>
- CONSTRAINTS = *constraint-name*
- *CONVERSE = <u>NO</u> | YES

*Option is nonpersistent

Descriptions of options

MODEL = <u>EQUIPROBABLE</u> | MARKOVIAN

If RANDOM is specified, the EQUIPROBABLE model dictates that every possible tree has the same probability. The MARKOVIAN model assumes that trees are generated by a random (pure-birth) branching process.

NTREES = *integer-value*

If RANDOM is specified, NTREES gives the number of trees to be generated. The default is 100.

SEED = *integer-value*

If RANDOM is specified, PAUP* references the system clock to obtain a value used to seed a random number generator. An explicit seed

may be specified to override the system clock default. (See "INITSEEDS = 0 | 1" on page 116 for more details.)

ROOTED = { YES | <u>NO</u> }

If ROOTED = YES, rooted trees will be generated, otherwise, the generated trees will be unrooted.

NONBINARY = { YES | <u>NO</u> }

When ALL is specified, NONBINARY = YES requests that trees are drawn from the set of all possible binary and nonbinary trees. By default (NONBINARY = NO), only binary trees are generated.

If ALL is specified the following options apply (see "Options And Subcommands Affecting Multiple Commands" on page 30 for descriptions of these options).

CONSTRAINTS = *constraint-nam*e
CONVERSE = <u>NO</u> | YES

GETTREES

Use the GETTREES command to load trees into memory from a file containing a NEXUS-format TREES block.

Syntax

GETTREES [*options*];

Menu equivalent

Trees > Get Trees from File ...

Available options

- FILE = *tree-file-name*
- FROM = *integer-value*
- TO = *integer-value*
- MODE = 1 | 2 | <u>3</u> | 4 | 5 | 7
- ALLBLOCKS = YES | <u>NO</u>
- *ROOTED = YES | <u>NO</u>
- *UNROOTED = YES | NO
- DUPTREES = <u>KEEP</u> | ELIMINATE
- STOREBRLENS = YES | <u>NO</u>
- STORETREEWTS = YES | <u>NO</u>
- WARNTREE = <u>YES</u> | NO

*Option is nonpersistent

Description of options

FILE = *tree-file-name*

Specify the *tree-file-name* containing a NEXUS-format TREES block. (See "Input/Output files" on page 9 for details regarding the file-name format)

FROM = *integer-value*

If nonzero, *integer-value* specifies the number of the first tree in the file to get.

TO = *integer-value*

If nonzero, *integer-value* specifies the number of the last tree in the file to get.

ALLBLOCKS = YES | <u>NO</u>

If ALLBLOCKS = YES, then all of the TREES blocks contained within the specified file will be processed.

MODE = 1 | 2 | <u>3</u> | 4 | 5 | 7

By default (MODE=3), any preexisting trees in memory are replaced by the trees read from the file. The MODE setting allows you to alter this behavior. Let M = the set of trees originally in memory and T = the set of trees from the tree file. The following mode values are then available:

1 =   replace M by T – M (i.e., keep trees from the file that are not originally in memory)

2 =   replace M by T ∩ M (keep trees from the file that are *also* originally in memory)

3 =   replace M by T (i.e., replace all trees in memory by all trees from the file)

4 =   replace M by M – T (i.e., keep trees in memory that are *not* also in the file)

5 =   replace M by M ⊕ T (i.e., keep trees that are either currently in memory or in the file, but not both places)

7 =   replace M by M ∪ T (i.e., append trees from file to trees originally in memory)

ROOTED = YES | <u>NO</u>

If directed character types are in effect for one or more characters and the trees being input are unrooted, you will ordinarily be asked if you want to convert the trees to rooted trees (otherwise, you will not be able to use the new trees unless you change the character types or root them later). If you specify ROOTED explicitly, this conversion will occur automatically without a prompt.

UNROOTED = YES | NO

> If the trees being input are rooted but all character types are currently undirected, you will ordinarily be asked if you want to convert the trees to unrooted trees. If you specify UNROOTED explicitly, this conversion will occur automatically without a prompt.

DUPTREES = KEEP | ELIMINATE

> Duplicate trees from the TREEFILE are eliminated if DUPTREES = ELIMINATE.

STOREBRLENS = YES | NO

> If branch lengths are present in the TREES block, then STOREBRLENS = YES stores branch lengths.

STORETREEWTS = YES | NO

> If tree weights are present in the TREES block, then STORETREEWTS = YES stores tree weights. Tree weights are typically the reciprocal of the number of trees found in either a BOOTSTRAP or JACKKNIFE replicate. This option allows the combination of bootstrap results from runs performed at different times or on different machines and the recovery of results obtained prior to a system crash. (See "USETREEWTS = YES | NO" on page 46 for more details)

WARNTREE = { YES | NO }

> If WARNTREE = NO, trees currently in memory will be quietly overwritten by the trees in the tree-file. Otherwise, you will be prompted for confirmation.

Example

The following example of the GETTREES command reads the trees and branch length information (if present) from the file "treefile.tre" without overwriting the trees currently in memory.

```
gettrees file=treefile.tre mode=7 storebrlens=yes;
```

HELP

Use the HELP command to obtain help on using PAUP*'s command-line interface.

Syntax

HELP [{COMMANDS | *command-name*}]

If invoked with no arguments, HELP produces a list of the available commands. If COMMANDS (or CMDS) is specified, a one-line description of each command is output. If a command-name is specified, HELP provides information of that command.

Example 1

The following example requests a lists of available commands.

```
help;
```

Example 2

The following example requests a list of available commands, with a one-line description of each.

```
help commands;
```

Example 3

The following example requests help on the BANDB command.

```
help bandb;
```

Note: The help file distributed with this beta test version is out-of-date at this time.

HOMPART

Use the HOMPART command to perform test for homogeneity of partitioned data sets. This test was described as the incongruence-length difference test by Farris et al. (1995).

Syntax

HOMPART[*options*][/*heuristic-search-options*|
          branch-and-bound-search-options];

Menu equivalent

Analysis > Partition Homogeneity Test ...

Available options

- PARTITION = *character-partition-name*
- NREPS = *integer-value*
- SEED = *integer-value*
- SEARCH = <u>HEURISTIC</u> | BANDB

Description of options

PARTITION = *character-partition-name*

Use PARTITION to specify the *character-partition-name* to be used in the analysis. To learn how to setup Character-partitions see

CHARPARTITIONS page 24.

NREPS = *integer-value*

Use NREPS to specify the number of homogeneity replicates to be performed.

SEED = *integer-value*

By default, PAUP* references the system clock to obtain a value used to seed a random number generator. An explicit seed may be specified to override the system clock default. (See "INITSEEDS = 0 | 1" on page 116 for more details.)

SEARCH = HEURISTIC | BANDB

A search is performed for each data partition replicate using the option specified for SEARCH. Options for either of the search options may be specified after a slash. (See the "BandB" command described on page 37 and the "HSearch" command described on page 71 for a description of the available options.)

Example

The following example of the HOMPART command uses the character partition definition named "genes" (see page 42), specifies 1000 randomizations using the random number seed 123, and uses a branch and bound search to obtain the sum of tree lengths for each partition.

```
charpartition genes=cytb:1-200, 12S:201-.;
hompart partition=genes nreps=1000 seed=123
search=bandb;
```

HSEARCH

Use the HSEARCH command to search for optimal trees using heuristic algorithms.

Syntax

HSEARCH [*options*] ;

Menu equivalent

Analysis > Heuristic Search ...

Available options

- *START = STEPWISE | NJ | CURRENT | *tree-number* [-*tree-number*]
- ADDSEQ = SIMPLE | CLOSEST | AsIs | RANDOM | FURTHEST
- NREPS = *integer-value*
- RSEED = *integer-value*

- SAVEREPS = YES | <u>NO</u>
- RSTATUS = YES | NO
- REFTAX = *integer-value*
- HOLD = *integer-value* | NO
- SWAP = NONE | NNI | SPR | <u>TBR</u>
- MULTREES = <u>YES</u> | NO
- KEEP = *real-value* | <u>NO</u>
- NBEST = *integer-value* | <u>NO</u>
- ALLSWAP = YES | <u>NO</u>
- RECONLIMIT = *integer-value* | <u>INFINITY</u>
- ENFORCE = YES | <u>NO</u>
- CONSTRAINTS = *constraint-name*
- CONVERSE = YES | <u>NO</u>
- USENONMIN = YES | <u>NO</u>
- STEEPEST = YES | <u>NO</u>
- ABORTREP = YES | <u>NO</u>
- NCHUCK = *integer-value*
- CHUCKSCORE = *real-value* | <u>NO</u>
- *RETAIN = YES | <u>NO</u>
- STATUS = <u>YES</u> | NO
- DSTATUS = *integer-value* | NONE
- RANDOMIZE = <u>ADDSEQ</u> | TREES
- TIMELIMIT = *interger-value* | <u>NONE</u>
- REARRLIMIT = *integer-value* | <u>NONE</u>
- LIMITPERREP = <u>NO</u> | YES

*Option is nonpersistent

Description of options

START = <u>STEPWISE</u> | <u>NJ</u> | CURRENT | *tree-number* [-*tree-number*]

When parsimony or likelihood is selected as the optimality criterion (see the "Set" command described on page 114) stepwise addition becomes the default procedure to obtain starting trees for branch swapping. Neighbor joining becomes the default procedure when distance is selected as the optimality criterion. Neighbor joining (START = NJ) may be used to obtain starting trees only if the optimality criterion is set to maximum likelihood or distance. Alternatively, you can request that all of the trees currently in memory (START = CURRENT) or a subset of the current trees (START = *tree-number* [-*tree-number*]) be used as a starting points for branch swapping.

ADDSEQ = <u>SIMPLE</u> | CLOSEST | <u>ASIS</u> | RANDOM | FURTHEST

Use ADDSEQ to specify the addition sequence to be used in the stepwise addition procedure. SIMPLE addition sequence is the default procedure used when parsimony is selected as the optimality criterion; ASIS is the default under the likelihood criterion.

Options for ADDSEQ = SIMPLE:

REFTAX = *reference-taxon-number*

By default, the first taxon in the data file is used as the reference taxon. Use REFTAX to specify an alternate reference taxon. This option is relevant only for unrooted-tree searches. For rooted-tree searches, the hypothetical ancestor (see "Ancstates" on page 36 and "Assume" on page 36) is used as the reference taxon.

Options for ADDSEQ = RANDOM:

NREPS = *integer-value*

Use NREPS to specify the number of random-addition-sequence replications to be performed. The default is 10.

RSEED = *integer-value*

By default, PAUP* references the system clock to initialize the seed used to generate pseudorandom numbers used to obtain random-addition-sequences (i.e., RSEED = 0). An explicit seed may be specified to override the system clock default. (See "INITSEEDS = 0 | 1" on page 116 for more details.)

SAVEREPS = YES | <u>NO</u>

If SAVEREPS = YES, then optimal trees from each repetition are saved even if they are not optimal over all repetitions.

RANDOMIZE = <u>ADDSEQ</u> | TREES

If RANDOMIZE = TREES, initial trees are random trees rather than trees found by random-addition-sequence.

RSTATUS = YES | <u>NO</u>

If you specify RSTATUS = YES, a status report showing the results of each random-addition-sequence replication is output. The status report can be useful in evaluating the effectiveness of the heuristic search while the search is proceeding.

Example 1

The following example shows the syntax used to execute a heuristic search with 20 random-addition-sequence replicates. A random number seed of 123 is used to generate random-addition-sequences.

```
hsearch addseq=random nreps=20 rseed=123;
```

HOLD = *integer-value* | NO

> Specifies the number of trees to be held at each cycle of the stepwise-addition procedure. By default, HOLD = 1, so that a single tree is held at each step. While setting *n* > 1 sometimes improves the score of the tree found by stepwise addition it will aslo increase the time required to complete the addition of sequences.

SWAP = <u>TBR</u> | SPR | NNI | NONE

> Specifies the algorithm used by branch-swapping: TBR = tree bisection-reconnection, SPR = subtree pruning-regrafting, NNI = nearest-neighbor interchange, NONE = No branch swapping performed. NNI rearrangements are a subset of those done by SPR, and SPR rearrangements are a subset of those done by TBR. Ordinarily, you will use TBR, but SPR or NNI can be used to reduce search times.

MULTREES = <u>YES</u> | NO

> Ordinarily, PAUP* saves all minimal trees it finds during branch swapping. You can override the default with MULTREES=NO to save only one of the best trees found. Use of MULTREES = NO is not recommended, as it can drastically reduce the ability of branch swapping to find the best tree. (This option is synonymous with MULPARS option in earlier versions of PAUP.)

NBEST = *integer-value* | <u>NO</u>

> Use NBEST to save the N best trees discovered during the search according to the optimality criterion in effect. The NBest option cannot be used when random-addition-sequence is in effect (i.e., ADDSEQ = RANDOM).

> ALLSWAP = YES | <u>NO</u>

> > Unless ALLSWAP = YES is specified, the trees saved by setting NBest > 1 will include only those trees encountered during a search for optimal trees; no special effort is made to find additional trees that might be still better. If ALLSWAP = YES is requested, then all trees saved are themselves input to the branch swapping procedure, making it much more likely that the optimal NBEST trees will be found. Setting ALLSWAP = YES can dramatically increase the search time, however.

RECONLIMIT = *integer-value* | <u>INFINITY</u>

> The reconnection limit option limits the scope of SPR and TBR rearrangements. If an *integer-value* is specified, the value corresponds to the sum of the number of nodes between each reconnected branch and the nodes incident to the branch that was cut for pruning (SPR) or bisection (TBR). A rearrangement limit of 1 corresponds to nearest-neighbor interchanges. The default limit is set to infinity.

USENONMIN = YES | <u>NO</u>

It is possible that the trees in memory at the time branch-swapping begins are not all equal in score. By default, only the best available trees are input to the branch-swapping procedure. If you want to swap on nonminimal trees as well, specify USENONMIN=YES.

STEEPEST = YES | <u>NO</u>

Specify STEEPEST to request use of the steepest-descent modification in the branch-swapping procedure. STEEPEST = NO reverses the effect of a previous STEEPEST specification.

NCHUCK = *integer-value*
CHUCKSCORE = *real-value* | <u>NO</u>

If this pair of options is used, no more than the number of trees specified by the NCHUCK option of score greater than or equal to the score specified by the CHUCKSCORE option will be retained in a search (or in a random-addition-sequence replicate). These options allow a user to effectively set a maxtrees limit (see page 119) when swapping on less "promising" trees and remove the limit for more "promising" trees.

Example 2

In the following example of the HSEARCH command the NCHUCK and CHUCKSCORE options are used to save a maximum of 100 trees with scores of 600 or above; no limit is imposed for trees less than 600.

```
hsearch nchuck=100 chuckscore=600;
```

ABORTREP = YES | <u>NO</u>

If ABORTREP is requested and NCHUCK and CHUCKLEN values have been specified, the current random-addition-sequence replicate will be aborted if the "chucking" limits are hit.

RETAIN = *number-of-trees-to-retain*

The first *number-of-trees-to-retain* will be retained in memory throughout the search. (Ordinarily, all trees initially in memory will be replaced by trees found during the search.) The RETAIN option is not persistent; you must reset it for every HSEARCH command if you want to continue retaining the same set of initial trees.

TIMELIMIT = *integer-value* | <u>NONE</u>

If an *integer-value* is specified, the search will be limited to a user-specified length of time, where the *integer-value* is the number of seconds to search.

REARRLIMIT = *integer-value* | <u>NONE</u>

If an *integer-value* is specified, the search will be limited to a user-specified number of tree rearrangements (branch swaps), where the *integer-value* is the maximum number of rearrangements to be

performed.

LIMITPERREP = <u>NO</u> | YES

By default, the TIMELIMIT and REARRLIMIT options apply to the entire search in random-addition-sequence searches. However, if you request LIMITPERREP=YES, then the limits are imposed separately for each addition-sequence replicate. For replicated searches such as the bootstrap (page 39), the jackknife (page 77), and the partition-homogeneity test (page 70) the limits are always applied separately for each search.

Example 3

The following example of the HSEARCH command imposes a rearrangement limit of 5000 for each of 20 random-addition-sequence replicates.

```
hsearch addseq=random nreps=20 rearrlimit=5000
limitperrep=yes;
```

See "Options And Subcommands Affecting Multiple Commands" on page 30 for a description of the following options:

KEEP = *keep-score*
ENFORCE = YES | <u>NO</u>
CONSTRAINTS = *constraint-name*
CONVERSE = YES | <u>NO</u>
STATUS = <u>YES</u> | NO
DSTATUS = *integer-value* | NONE

INCLUDE

Use the INCLUDE command to re-include characters that were previously excluded.

Syntax

INCLUDE *character-list* [/ONLY];

Unless /ONLY is specified, characters specified in the character list are simply removed from the set of currently excluded characters. If you want only those characters specified in the list to be included, specify /ONLY; characters not explicitly specified in the list will then be excluded. PAUP* includes several predefined characters that may be used in conjunction with or without a character-list. See "CharSet" command described on page 23 for more details.

Menu equivalent

Data > Include-Exclude Characters ...

INGROUP

Use the INGROUP command to return one or more taxa to the ingroup.

Syntax

INGROUP *taxon-list* [/ONLY];

Unless /ONLY is specified, taxa specified in the taxon-list are simply removed from the current outgroup. If you want only those taxa specified in taxon-list to be included in the ingroup, specify /ONLY; any taxon that is not explicitly specified in the list will be transferred to the outgroup.

Menu equivalent

Data > Define Outgroup ...

JACKKNIFE

Use the JACKKNIFE command to perform jackknife analysis.

Syntax

JACKKNIFE   [*options*][/*heuristic-search-options*|*branch-and-bound-search-options*];

Menu equivalent

Analysis > Boostrap/Jackknife ...

Available options

- PCTDELETE = *real-value*
- JSEED = *integer-value*
- NREPS = *integer-value*
- SEARCH = HEURISTIC | BANDB | FASTSTEP | NJ | UPGMA
- RESAMPLE = NORMAL | JAC
- CONLEVEL = *integer-value*
- KEEPALL = YES | NO
- WTS = IGNORE | SIMPLE | REPEATCNT | PROPORTIONAL
- GRPFREQ = YES | NO
- TREEFILE = *tree-file-name*
- FORMAT = NEXUS | ALTNEXUS | FREQPARS | PHYLIP | HENNIG
- *REPLACE = YES | NO
- CUTOFFPCT = *integer-value*

*Option is nonpersistent

Description of options

PCTDELETE = *real-value*

Use PCTDELETE to specify the percentage of the data set to be deleted for each jackknife replicate.

JSEED = *integer-value*

By default, PAUP* references the system clock to initialize the seed used to generate pseudorandom numbers for character resampling. An explicit seed may be specified to override the system clock default. (See "INITSEEDS = 0 | 1" on page 116 for more details.)

NREPS =*integer-value*

Use NREPS to specify the number of jackknife replications to be performed.

SEARCH = HEURISTIC | BANDB | FASTSTEP | NJ | UPGMA

A tree search is performed for each jackknife sample using the method specified under SEARCH. If SEARCH = FASTSTEP, tree searches in each replication are performed using one random-sequence-addition replication and no branch swapping. SEARCH = NJ | UPGMA may be used only for the distance criterion.

RESAMPLE = NORMAL | JAC

If RESAMPLE = JAC, then the JAC program in Hennig86 (Farris) is emulated.

CONLEVEL = *integer-value*

Use CONLEVEL to specify the minimum proportion of the bootstrap replicates (expressed as a percentage) in which a group must appear if it is to be included in the bootstrap consensus tree. For example, to obtain a bootstrap consensus tree that shows only those groups which occurred on more than 80% of the trees, you would specify CONLEVEL=80. CONLEVEL must be at least 50, which is the default.

KEEPALL = YES | NO

If you request KEEPALL, groups occurring at frequencies less than CONLEVEL will also be retained in the jackknife consensus as long as they are compatible with all groups that are already included in the consensus. Effectively, this forces KEEPALL=50, because any group occurring in 50% or more of the replicates will automatically be compatible with all more frequently occurring groups.

WTS = IGNORE | SIMPLE | REPEATCNT | PROPORTIONAL

If WTS = IGNORE, any previously applied character weights are ignored. If WTS = SIMPLE, then characters are sampled with equal probability but character weights are applied. If WTS = REPEATCNT, then weights are treated as repeat counts. WTS = REPEATCNT is only

allowed if all weights are integers. If WTS=PROPORTIONAL, then characters are sampled in proportion to their weights. WTS=PROPORTIONAL is available even when non-unit character weights are in effect.

GRPFREQ = Y̲E̲S̲ | NO

Unless GRPFREQ = NO, a table of jackknife partition frequencies is shown.

TREEFILE = *tree-file-name*

If TREEFILE is specified, a description of all trees found in each jackknife replication is given in the NEXUS tree format. Trees contain a weight comment for the purpose of combining jackknife results from runs performed at different times or on different machines and for recovering results obtained prior to a system crash. The weight is the reciprocal of the number of trees found in a replicate. The *tree-file-name* must follow the conventions described under *"Input/Output files" on page 9*

REPLACE = YES | N̲O̲

If TREEFILE is used and the specified file already exists you will be prompted for confirmation that the existing file should be replaced. REPLACE suppresses this prompt and the existing file will be quietly overwritten by the new data.

FORMAT = N̲E̲X̲U̲S̲ | ALTNEX | FREQPARS | PHYLIP | HENNIG

Specifies the type of treefile to be produced. NEXUS requests a file containing the standard NEXUS TREES block using a translation table (which greatly reduces the amount of disk space required to store the trees). ALTNEX also specifies a NEXUS TREES block, but no translation table is used (the full taxon names are included in each tree description). FREQPARS requests a treefile for the FREQPARS program described by Swofford and Berlocher (1987). PHYLIP requests a treefile for input to version 3.4 of Felsenstein's (1991) PHYLIP package. HENNIG requests a treefile for version 1.5 of Farris's (1988) Hennig86 program.

CUTOFFPCT = *integer-value*

Use CUTOFFPCT to specify the minimum jackknife partition frequency to be displayed in the partition frequency table. All partitions below this integer-*value* are pooled into a single class.

LAKE

Use the LAKE command to perform an analysis using Lake's (1987b) method of linear invariants. This command is available only when DATATYPE = DNA | RNA | NUCLEOTIDE.

Syntax

LAKE [*options*];

Menu equivalent

Analysis > Lake's Invariants ...

Available options

- MODE = <u>CHOOSE4</u> | ALLQUART | FOURGRPS
- SPECTDIST = YES | <u>NO</u>
- BRLENS = YES | <u>NO</u>
- SUMTABONLY = YES | <u>NO</u>
- EXACTN = *integer-value*
- TAXA = *taxon-list* FOR MODE=CHOOSE4
- GRPA = *taxon-list* 1 FOR MODE=FOURGRPS
- GRPB = *taxon-list* 2 FOR MODE=FOURGRPS
- GRPC = *taxon-list* 3 FOR MODE=FOURGRPS
- GRPD = *taxon-list* 4 FOR MODE=FOURGRPS

Description of options

MODE = <u>CHOOSE4</u> | ALLQUART | FOURGRPS

If there are more than 4 (non-deleted) taxa in the data set, this option specifies whether you want to choose 4 taxa for analysis (MODE=CHOOSE4), analyze all possible quartets (MODE = ALLQUART), or divide the taxa into four groups and analyze all quartets containing one member from each group (MODE=FOURGRPS).

SPECTDIST= YES | NO

SPECTDIST requests the output of the spectral distribution (the number of positions falling into each of the 36 possible patterns considered informative by Lake's method).

BRLENS = YES | NO

BRLENS requests output of branch-lengths calculated by Lake's "operator metric" (Lake, 1987a).

SUMTABONLY = YES | NO

SUMTABONLY limits the output to a summary table of results rather than outputting results for each quartet.

EXACTN = *integer-value*

An exact binomial test (rather than the chi-square approximation) will be used to test the significance of deviations of invariant scores

from 0 if the number of informative positions is less than n.

TAXA = *taxon-list*

If MODE = CHOOSE4, you must specify exactly four taxa; the *taxon-list* must contain exactly four taxa. If MODE = ALLQUART, you may limit the number of included taxa by selecting at least four taxa.

GRPA = *taxon-list-for-first-group*
GRPB = *taxon-list-for-second-group*
GRPC = *taxon-list-for-third-group*
GRPD = *taxon-list-for-fourth-group*

If MODE = FOURGRPS, you must assign at least one taxon to each of the four groups; GRPA, GRPB, GRPC, and GRPD.

LEAVE

Use the LEAVE command to terminate processing of an input file. Ordinarily, PAUP* continues processing until the end of an input file is reached. If, for whatever reasons, you do not want some of the commands in the input file to be processed, insert a LEAVE command into the file at the point where you want execution to stop. PAUP* then continues as if the end of the file were reached at that point. LEAVE has no effect if issued interactively.

Syntax

LEAVE;

LOADCONSTR

Use the LOADCONSTR command to load constraints from treefile.

Syntax

LOADCONSTR [*options*];

Menu equivalent

Analysis > Load Constraints ...

Available options

- FILE = *tree-file-name*
- *APPEND = YES | <u>NO</u>
- *ASBACKBONE = YES | <u>NO</u>

*Option is nonpersistent

Description of options

FILE = *tree-file-name*

> Use FILE to specify the *tree-file-name* containing constraint trees.

APPEND = YES | <u>NO</u>

> If APPEND = YES, then the constraints contained in the input file are added to any constraints already in memory.

ASBACKBONE = YES | <u>NO</u>

> Use ASBACKBONE to load the trees as backbone constraints. Backbone constraint trees should contain only a subset of the taxa currently in memory.

LOG

By default, output generated by PAUP* goes only to the "display buffer," a region of memory set aside exclusively for this purpose. PAUP*'s main display window is used to view this information. The LOG command may be used to request direction of PAUP output to a file (e.g., for subsequent printing).

Syntax

LOG [*options*];

Menu equivalent

File > Log Output to Disk ...

Available options

- FILE = *log-file-name*
- *REPLACE = YES | <u>NO</u>
- *APPEND = YES | <u>NO</u>
- *START = YES | <u>NO</u>
- *STOP = YES | <u>NO</u>
- FLUSHLOG = YES | <u>NO</u>

*Option is nonpersistent

Description of options

FILE = *log-file-name*

> The name and location of the the log file may be specified using the FILE command. If the *file-specification* contains any of the characters equal-sign (=), semicolon (;), colon (:), or blank, it must be enclosed within single-quotes. See "Input/Output files" on page 9 for

information on how to specify platform specific path names.

REPLACE = YES | <u>NO</u>

If REPLACE is not specified explicitly and the file already exists, you will receive a warning and will have the opportunity to cancel the command before the contents of the existing file are erased.

APPEND = YES | <u>NO</u>

If APPEND is specified, subsequent output is appended to the previous contents (if any) of the file. Otherwise, subsequent output will overwrite the original contents of the file. The APPEND/REPLACE setting is retained between invocations of the LOG command, unless the file is changed by a FILE= directive.

START = YES | <u>NO</u>

If START is specified logging is initiated to the named file. If FILE is not specified explicitly, a default name is assigned.

FLUSHLOG = YES | <u>NO</u>

Specification of FLUSHLOG causes the file's buffer to be flushed after every line of output. Ordinarily, this degrades system performance and is not recommended. However, there may be situations in which immediate flushing is useful.

Example

The following example uses the LOG command to save basic information about the trees (see the "DescribeTrees" command described on page 51) in memory to the file named "myoutput.log"; after which logging is stopped. In the final line logging is resumed and the subsequent output is appended to the original log file.

```
log file=myoutput.log;
describetrees;
log stop;
log start append;
```

LSCORES

Use the LSCORES command to calculate the likelihoods of trees in memory.

Syntax

LSCORES [*tree-list*] [/*options*];

Menu equivalent

Trees > Tree Scores > Likelihood ...

Available options

- SITELIKES = YES | <u>NO</u>
- CATEGLIKES = YES | <u>NO</u>
- *SORTTREES = YES | <u>NO</u>
- STOREASWTS = YES | <u>NO</u>
- SCOREFILE = *file-name-for-scores*
- *REPLACE = YES | <u>NO</u>
- *APPEND = YES | <u>NO</u>
- *DISPLAYOUT   <u>NO</u> | YES
- *PARSAPPROX = YES | <u>NO</u>
- KHTEST = <u>NONE</u> | NORMAL | RELL | FULLOPT
- SHTEST = <u>NONE</u> | RELL | FULLOPT
- BOOTREPS = *integer-value*
- TAILKH = 1 | 2
- NST = 1 | <u>2</u> | 6
- TRATIO = *real-value* | ESTIMATE | PREVIOUS
- RMATRIX = (*rAC rAG rAT rCG rCT*) | ESTIMATE | PREVIOUS
- RCLASS = (*cAC cAG cAT cCG cCT cGT*)
- VARIANT = <u>HKY</u> | F84
- BASEFREQ = <u>EMPIRICAL</u> | EQUAL | ESTIMATE | PREVIOUS | (*frqA frqC frqG*)
- RATES = <u>EQUAL</u> | GAMMA | SITESPEC
- SHAPE = *real-value* | ESTIMATE | PREVIOUS
- NCAT = *integer-value*
- REPRATE = <u>MEAN</u> | MEDIAN
- WTS = <u>REPEATCNT</u> | IGNORE
- INITBRLEN = <u>ROGERS</u> | LS | *real-value*
- MAXPASS = *integer-value*
- SMAXPASS = *integer-value*
- DELTA = *real-value*
- SDELTA = *real-value*
- USEAPPROX = <u>YES</u> | NO
- APPROXLIM = *real-value*
- ADJUSTAPPLIM = <u>YES</u> | NO
- LOGITER = YES | <u>NO</u>
- PINVAR = *real-value* | ESTIMATE | PREVIOUS
- RECON = <u>MARGINAL</u> | JOINT
- ALLPROBS YES | <u>NO</u>

- SITERATES = PARTITION[:*charpartition-name*] | RATESET[:*rateset-name*] | PREVIOUS
- CLOCK = YES | <u>NO</u>
- USERBRLENS = YES | <u>NO</u>
- MINMEMREQ = YES | <u>NO</u>
- STARTVALS = <u>PARSAPPROX</u> | ARBITRARY
- PARAMCLOCK = <u>STANDARD</u> | RAMBAUT | BRLENS | SPLITTIMES | MDRAMBAUT | THORNE
- LCOLLAPSE = <u>YES</u> | NO
- MLDISTFORLS = YES | <u>NO</u>
- OPTBRLENS = <u>NEWTON1D</u> | BRENT1D | PRAXIS

*Option is nonpersistent

Description of options

SITELIKES = YES | <u>NO</u>

Use SITELIKES to output single-site likelihood scores.

CATEGLIKES = YES | <u>NO</u>

If gamma and/or invariable-sites rates are specified, the relative contribution to the likelihood for a site from each rate category is available when single-site likelihoods are requested (i.e., SITELIKES=YES)

SORTTREES = YES | <u>NO</u>

When multiple trees are evaluated under the LSCORES command SORTTREES = YES will sort tree scores so that they are displayed in ascending order. Note, SORTTREES = YES may noticably increase the time required to display tree scores.

STOREASWTS = YES | <u>NO</u>

Use STOREASWTS to store the site-likelihood scores as character-weights. The reason for doing this are esoteric; there would ordinarily be no reason to use this option.

SCOREFILE = *file-name-for-scores*

If SCOREFILE is specified, likelihood scores and parameter estimates for each tree evaluated are saved to a text file. The *file-name-for-scores* must follow the conventions described under "Input/Output files" on page 9.

REPLACE = YES | <u>NO</u>
APPEND = YES | <u>NO</u>

If the *file-name-for-scores* already exists, you will be prompted for confirmation before the existing file is replaced. Use REPLACE = YES to suppress this prompt; the existing file will be quietly overwritten by the new data. Alternatively, you

may specify APPEND, in which case a new TREES block will be concatenated to the end of an existing file.

DISPLAYOUT   NO | YES

Set DISPLAYOUT = YES if you wish to suppress output to the display buffer.

PARSAPPROX = YES | NO

Use PARSAPPROX to specify parsimony-based approximations of likelihood model parameters (i.e., the gamma shape parameter and the parameters used in the instantaneous rate matrix).

KHTEST = NONE | NORMAL | RELL | FULLOPT

Use KHTEST to perform the Kishino-Hasegawa test (Kishino and Hasegawa, 1989) and to specify the distribution of the test statistic. If NONE (the default) is specified, the test is not performed. If NORMAL is specified a normal distribution is used for the test distribution. RELL generates a test distribution by the resampling estimated log-likelihood method (Kishino et al. 1990). FULLOPT is similar to RELL except that log-likelihood scores are estimated using a fully optimized model.

TAILKH = 1 | 2

Use TAILKH to select whether *P*-values are calculated from one or two tails of the test distribution.

SHTEST = NONE | RELL | FULLOPT

Use SHTEST to perform the Shimodaira-Hasegawa test (Shimodaira and Hasegawa,1999; see also Goldman et al., 2000) and to specify the distribution of the test statistic. If NONE (the default) is specified, the test is not performed. RELL generates a test distribution by the resampling estimated log-likelihood method (Kishino et al. 1990). FULLOPT is similar to RELL except that log-likelihood scores are estimated using a fully optimized model.

BOOTREPS = *integer-value*

Use BOOTREPS to specify the number of bootstrap replicates used to generate a test distribution when RELL or FULLOPT is selected for the KHTEST or SHTEST options. The default number of bootstrap replicates is 1000.

Example

The following example of the LSCORES command returns the emperical base frequencies, the estimated transition/transversion ratio, and the negative log likelihood of all trees in memory.

```
lscore all/nst=2 basefreq=empirical tratio=estimate;
```

See the "LSet" on page 87 for a description of the remaining options.

LSET

Use the LSET command to set options for maximum likelihood analysis.

Syntax

LSET [*options*];

Menu equivalent

Analysis > Likelihood Settings ...

Available options

- NST = 1 | <u>2</u> | 6
- TRATIO = *real-value* | ESTIMATE | PREVIOUS
- RMATRIX = (*rAC rAG rAT rCG rCT*) | ESTIMATE | PREVIOUS
- RCLASS = (*cAC cAG cAT cCG cCT cGT*)
- VARIANT = <u>HKY</u> | F84
- BASEFREQ = <u>EMPIRICAL</u> | EQUAL | ESTIMATE | PREVIOUS | (*frqA frqC frqG*)
- RATES = <u>EQUAL</u> | GAMMA | SITESPEC
- SHAPE = *real-value* | ESTIMATE | PREVIOUS
- NCAT = *integer-value*
- REPRATE = <u>MEAN</u> | MEDIAN
- WTS = <u>REPEATCNT</u> | IGNORE
- INITBRLEN = <u>ROGERS</u> | LS | *real-value*
- MAXPASS = *integer-value*
- DELTA = *real-value*
- USEAPPROX = <u>YES</u> | NO
- APPROXLIM = *real-value*
- ADJUSTAPPLIM = <u>YES</u> | NO
- LOGITER = YES | <u>NO</u>
- PINVAR = *real-value* | ESTIMATE | PREVIOUS
- RECON = <u>MARGINAL</u> | JOINT
- ALLPROBS YES | <u>NO</u>
- SITERATES = PARTITION[:*charpartition-name*] | RATESET[:*rateset-name*] | PREVIOUS
- CLOCK = YES | <u>NO</u>
- ZEROLENTEST = <u>NO</u> | FULL | CRUDE
- USERBRLENS = YES | <u>NO</u>
- MINMEMREQ = YES | <u>NO</u>

- STARTVALS = <u>PARSAPPROX</u> | ARBITRARY
- PARAMCLOCK = <u>STANDARD</u> | RAMBAUT | BRLENS | SPLITTIMES |
          MDRAMBAUT | THORNE
- LCOLLAPSE = <u>YES</u> | NO
- MLDISTFORLS = YES | <u>NO</u>
- OPTBRLENS = <u>NEWTON1D</u> | BRENT1D | PRAXIS

*Option is nonpersistent

Description of options

The following options are used to specify the substitution model
parameters:

NST = 1 | <u>2</u> | 6

> Use NST to specify the number of substitution types. For example,
> if NST = 1, then all substitution types are pooled into a single rate.

TRATIO = *real-value* | ESTIMATE | PREVIOUS

> Use TRATIO to specify a transition/transversion ratio. If TRATIO =
> PREVIOUS, then the Transversion:Transition ratio in memory will be
> applied. Use TRATIO = ESTIMATE to obtain the maximum likelihood
> estimate for the data set and tree topology currently in memory. The
> default value for this option is 2. TRATIO is ignored unless NST = 2.

RMATRIX = (*rAC rAG rAT rCG rCT*) | ESTIMATE | PREVIOUS

> If NST = 6, then the values for a six-parameter instantaneous rate
> matrix may be specified, estimated using maximum likelihood
> (ESTIMATE), or set to the values currently in memory (PREVIOUS).

RCLASS = (*cAC cAG cAT cCG cCT cGT*)

> If RMATRIX = ESTIMATE, then a submodel of the six-parameter rate
> matrix may be specified. For example, substitution types may be
> pooled into common classes to reflect changes between structurally
> similar molecules (e.g. transitions and transversions). A user is free
> to decide among the many possible ways that substitution types may
> be divided or combined. One restriction of the GTR model is the
> Tamura-Nei model.

Example 1

The following example pools transitions into a common rate class and
defines two seperate rate classes for transversions. Rate classes will be
estimated using maximum likelihood.

```
lset nst=6 rmatix=estimate rclass=(a b a a c a);
```

Use the following options to specify the base frequency parameters:

VARIANT = <u>HKY</u> | F84

> If TRATIO is used, then you may choose between two variants of the 2-parameter model for unequal base frequencies; that is, the model described by Felsenstein (1984) or Hasegawa-Kishino-Yano (1985).

BASEFREQ = <u>EMPIRICAL</u> | EQUAL | ESTIMATE | PREVIOUS | (*frqA frqC frqG*)}

> By default PAUP* uses the empirical base frequencies (EMPIRICAL). Base frequencies may also be estimated by maximum likelihood for a data set and tree topology (ESTIMATE), assumed equal (EQUAL), explicitly specified (*frqA frqC frqG*), or set the values currently in memory (PREVIOUS).

The following options are used to specify among-site rate variation parameters:

RATES = <u>EQUAL</u> | GAMMA | SITESPEC

> Use RATES to specify a model of among-site rate variation. By default, PAUP* assumes that all sites change at the same rate; however, many molecular studies suggest that equal rates is not a valid assumption. You may choose between two models to compensate for among site rate-variation.

SHAPE = *real-value* | ESTIMATE | PREVIOUS

> Use the SHAPE option to specify the shape of the gamma distribution used to accommodate among-site rate variation. If SHAPE = ESTIMATE, then the maximum likelihood estimate of this parameter is obtained for a given data set and tree topology. If SHAPE = PREVIOUS, then the value of SHAPE already in memory is used. The default value for this option is 0.5.

NCAT = *integer-value*

> Use NCAT to specify the number of categories to divide the discrete approximation of the gamma distribution.

REPRATE = <u>MEAN</u> | MEDIAN

> Use REPRATE to specify whether the mean or the median is used to represent the rates within each category of the discrete approximate gamma distribution.

PINVAR = *real-value* | ESTIMATE | PREVIOUS

> Use PINVAR to specify the proportion of sites unable to accept substitutions; the remaining sites are assumed to vary at the same rate. The default value for this option is 0.

SITERATES = PARTITION[:*charpartition-name*] | RATESET[:*rateset-name*] | PREVIOUS

> Use SITERATES to obtain a single rate estimate for a set of characters. The default for this option is none.

Example 2

The following example uses a descrete gamma distribution with four rate categories to estimate across-site rate heterogeneity. The shape parameter of the gamma distribution is estimated using maximum likelihood.

lset rates=gamma ncat=4 shape=estimate;

Example 3

The following example estimates a single rate for each subset of the data as defined in the CHARPARTITION command. See "CharPartition" on page 42 for more detail regarding this command.

```
charpartition genes=g1:1-300, g2:301-600, g3:601-700;
lset rates=sitespec siterates=partition:genes;
```

Example 4

The following example defines a specific rate for each subset of the data as defined by the RATESET command. See "RateSet" on page 106 for more dtails regarding this command.

```
rateset genes=1:1-300, 2:301-600, 3:601-700;
lset rates=sitespec siterates=rateset:genes;
```

Use the following options to specify the starting parameter values for the likelihood estimation:

STARTVALS = <u>PARSAPPROX</u> | ARBITRARY

By default, PAUP* uses parsimony-based approximations for the starting values of some parameters estimated by maximum likelihood. If STARTVALS = ARBITRARY, then arbitrary values are used. The parsimony-based starting point is usually much closer to the optimal value and convergence to the optimum will therefore occur more quickly. In some cases, however, the parsimony estimate can be a poor choice and convergence will occur more quickly from an arbitrary starting point.

INITBRLEN = <u>ROGERS</u> | LS | *real-value*

Use INITBRLEN to specify the method for initializing starting branch lengths. By default, PAUP* uses the Rogers-Swofford method (ROGERS); however, you may also specify a least-squared method (LS), or an arbitrary value (*real-value*).

MLDISTFORLS = YES | <u>NO</u>

Development of this option is in progress.

USEAPPROX = <u>YES</u> | NO

Use USEAPPROX to reject trees if approximate likelihood exceeds the exact likelihood by more than a specified amount. By default PAUP*

rejects all trees with a likelihood score greater than the five percent of the exact value. In so doing, PAUP* evaluates fewer topologies dramatically reducing overall search times.

APPROXLIM = *real-value*

Use APPROXLIM to specify the percent deviation from the exact likelihood score before a tree is no longer considered for branch swapping.

ADJUSTAPPLIM = <u>YES</u> | NO

Use ADJUSTAPPLIM to dynamically readjust the limit specified under APPROXLIM.

Use the following options to specify performance tuning settings:

MAXPASS = *integer-value*

Use MAXPASS to set the maximum number of branch-length smoothing passes. Setting MAXPASS to a smaller value can speed search times, but the objective function may not be fully optimized.

DELTA = *real-value*

Use DELTA to set the value for stopping smoothing passes. In other words, when a score changes less then the *real-value* specified, PAUP* assumes that the branch lengths have converged to their optimal values for a give tree.

OPTBRLENS = <u>NEWTON1D</u> | BRENT1D | PRAXIS

This option is currently under development.

LOGITER = YES | <u>NO</u>

If LOGITER = YES, then the progress of the branch-length iterations will be output to the display buffer. This output can be extensive and it is ordinarily not useful.

MINMEMREQ = YES | <u>NO</u>

If MINMEMREQ = YES, then the memory requirements will be minimized. The speed of most operations will be penalized if this option is specified.

Miscellaneous options:

RECON = <u>MARGINAL</u> | JOINT

Use RECON to specify the method used to reconstruct ancestral state characters.

ALLPROBS = YES | <u>NO</u>

Use ALLPROBS to show marginal probabilities of all base assignments when ancestral states are reconstructed.

CLOCK = YES | <u>NO</u>

> If CLOCK = YES, then the molecular clock model is enforced.

> > PARAMCLOCK = <u>STANDARD</u> | RAMBAUT | BRLENS | SPLITTIMES | MDRAMBAUT | THORNE

> > If PARAMCLOCK=STANDARD (the default), the multidimensional version of Rambaut's parameterization is used when one or more subsitution-model parameters are being estimated. If no parameters other than branch lengths are being estimated, the one-dimensional version of Rambaut's parameterization is used. To force a specific method to be used regardless of the number of parameters being estimated, use PARAMCLOCK=<anything-other-than-standard>.

ZEROLENTEST = <u>NO</u> | FULL | CRUDE

> Unless ZEROLENTEST=NO, a likelihood-ratio test for whether a branch-length is zero is performed when branch lengths are output under the DESCRIBETREES command (page 51). The best test, ZEROLENTEST=FULL, optimizes all branch lengths under the constraint that one of the branches is zero, for each branch tested. The Full option can be slow for large trees, so an alternative test is available (ZEROLENTEST=CRUDE) in which the likelihood of the tree is determined after setting each branch, in turn, to zero length, with no readjustment to the other branches. This is the same test implemented by Joe Felsenstein in DNAML (who also used the word "crude" to describe this method in his documentation).

USERBRLENS = YES | <u>NO</u>

> If USERBRLENS = YES, then user-input branch lengths are used when evaluating user-trees.

LCOLLAPSE = <u>YES</u> | NO

> Use LCOLLAPSE to collapse branches of effectively zero length when searching. Branches have length less than or equal to 1×10^{-8} are collapsed.

WTS = <u>REPEATCNT</u> | IGNORE

> If WTS=IGNORE, character weights are ignored. The default (WTS = REPEATCNT) treats integer weights as repeat counts.

MATRIXREP

Use dthe MATRIXREP command to save trees in "matrix representation." A NEXUS file is created that contains a dummy character for each clade on the tree; analysis of the resulting data set under parsimony will regenerate the original tree.

Syntax

MATRIXREP [*options*];

Menu equivalent

Trees > Matrix Representation ...

Available options

- FILE = *tree-file-name*
- BRLENS = <u>NO</u> | YES
- FROM = *starting-tree-number*
- TO = *ending-tree-number*
- *REPLACE = <u>NO</u> | YES
- *APPEND = <u>NO</u> | YES

*Option is nonpersistent

Description of options

BRLENS = <u>NO</u> | YES

Use BRLENS=YES to save branch lengths as character weights; a weighted parsimony analysis using these weights will regenerate the original tree and branch lengths.

FILE = *tree-file-name*

Specifies a name for the tree file. If you do not explicitly specify a file name, a default file name will be used.

REPLACE = YES | <u>NO</u>
APPEND = YES | <u>NO</u>

If FILE = *tree-file-name* option is specified and the *tree-file-name* already exists, you will be prompted for confirmation that the existing file should be replaced. Explicit specification of REPLACE = YES suppresses this prompt; the existing file will be quietly overwritten by the new data. Alternatively, you may specify APPEND = YES, in which case a new TREES block will be concatenated to the end of an existing file.

FROM = *starting-tree-number*

If nonzero, *starting-tree-number* specifies the number of the first tree to save.

TO = *ending-tree-number*

If nonzero, *ending-tree-number* specifies the number of the last tree to save.

MPRSETS

Use the MPRSETS command to request output of possible character-state assignments (MPR-sets) for one or more characters on one or more trees, by superimposing the possible character-states for each node on a diagram of the tree.

Syntax

MPRSETS character-list [/*options*];

The *character-list* specifies the character(s) for which possible character-state assignments are shown, and consists of one or more character numbers, character names, or character-set names.

Menu equivalent

Trees > Show Reconstructions ...

Available options

- TREES = *tree-list*
- TCOMPRESS = YES | NO

Description of options

TREES = *tree-list*

The *tree-list* specifies the tree numbers for which possible character-state assignments are to be shown. If this is the first MPRSETS command and you do not specify a tree list, reconstructions are shown for the first tree only.

TCOMPRESS = YES | NO

See "Options And Subcommands Affecting Multiple Commands" on page 30 for details.

If no characters are specified for either a RECONSTRUCT (page 107) or MPRSETS command, the characters are taken to be those plotted in the last invocation of either of these commands. For example, RECONSTRUCT 1 3 5 7; MPRSETS; will cause both commands to output information for characters 1, 3, 5, and 7.

NJ

Use the NJ command to calculate a tree using the neighbor-joining method.

Syntax

NJ [*options*];

Menu equivalent

Analysis > Neighbor Joining/UPGMA ...

Available options

- ENFORCE = YES | <u>NO</u>
- CONSTRAINTS = *constraint-name*
- BRLENS = YES | <u>NO</u>
- BREAKTIES = <u>SYSTEMATIC</u> | RANDOM
- TIESEED = *integer-value*
- SHOWTREE = <u>YES</u> | NO
- TREEFILE = *tree-file-name*
- *REPLACE = YES | <u>NO</u>
- *APPEND = YES | <u>NO</u>

*Option is nonpersistent

Description of options

BRLENS = YES | <u>NO</u>

> If BRLENS is specified, a list of branch lengths will be output to the display buffer and tree descriptions will include branch lengths. BRLENS = NO reverses the effect of a previous BRLENS specification.

BREAKTIES = <u>SYSTEMATIC</u> | RANDOM

> Unless BREAKTIES = RANDOM, ties encountered during tree construction are broken systematically according to the input order of the taxa.

TIESEED = *integer-value*

> If BREAKTIES = RANDOM, PAUP* references the system clock to initialize the seed used to generate pseudorandom numbers for breaking ties found during tree construction. An explicit seed may be specified to override the system clock default. (See "INITSEEDS = 0 | 1" on page 116 for more details.)

SHOWTREE = <u>YES</u> | NO

> If SHOWTREE = YES, the neighbor-joining tree is printed to the display buffer.

TREEFILE = *tree-file-name*

> If TREEFILE is specified, a description of the trees found by the neighbor-joining method is output to a file in the NEXUS tree format. If BRLENS = YES the tree description in the tree-file will contain branch length information. The *tree-file-name* must follow the conventions described under"Input/Output files" on page 9.

REPLACE = YES | <u>NO</u>
APPEND = YES | <u>NO</u>

> If the specified file already exists, you will be prompted for confirmation before the existing file is replaced. Use REPLACE = YES to suppress this prompt; the existing file will be quietly overwritten by the new data. Alternatively, you may specify APPEND, in which case a new TREES block will be concatenated to the end of an existing file.

See "Options And Subcommands Affecting Multiple Commands" on page 30 for a description of the following options:

ENFORCE = YES | <u>NO</u>
CONSTRAINTS = *constraint-name*

OUTGROUP

Use the OUTGROUP command to assign one or more taxa to the outgroup.

Syntax

OUTGROUP *taxon-list* [/ONLY];

Unless /ONLY is specified, taxa specified in the taxon-list are simply added to the current outgroup. If taxa have already been assigned to the outgroup and you want only those taxa specified in taxon-list to remain in the outgroup, specify /ONLY; any taxon that is not explicitly specified in the list will be transferred to the ingroup.

Menu equivalent

Data > Define Outgroup ...

PAIRDIFF

Use the PAIRDIFF command to show nucleotide-pair frequencies.

Syntax

PAIRDIFF [*taxon-list*] [/*options*];

Menu equivalent

Data > Pairwise Base Differences ...

Available options

- SHORTFMT = YES | <u>NO</u>
- LONGFMT = <u>YES</u> | NO

Description of options

SHORTFMT = YES | <u>NO</u>

If SHORTFMT is specified, then nucleotide-pair frequencies, proportion of sites differing, and apparent transition/transversion ratio is given on a single line for each sequence pair.

LONGFMT = <u>YES</u> | NO

If LONGFMT is specified, then a 4 X 4 matrix of nucleotide-pair frequencies, proportion of sites differing, and apparent transition/transversion ratio is given for each sequence pair both before and after adjustments are made for gaps/ambiguities.

PERMUTE

Use the PERMUTE command to perform Archie-Faith-Cranston randomization tests.

Syntax

PERMUTE [*options*][/*heuristic-search-options* | *branch-and-bound-search-options*];

Menu equivalent

Analysis > Permutation Tests ...

Available options

- TEST = <u>PTP</u> | TPTP | COMPARE2
- PSEED = *integer-value*
- NREPS = *integer-value*
- SEARCH = <u>HEURISTIC</u> | BANDB
- RANDOMIZE = <u>ALL</u> | INGROUP
- CONVERSE = YES | <u>NO</u>
- CONSTRAINTS = *constraint-name*
- FIRST = *constraint-tree-name*
- SECOND = *constraint-tree-name*

Description of options

TEST = <u>PTP</u> | TPTP | COMPARE2

TEST = PTP does a simple permutation test for the existence of phylogenetic structure. TEST = TPTP does a "topology-dependent" PTP test; comparing the best unconstrained tree to threes forces to be compatible with a constraint tree. If TEST = COMPARE2, then the difference between two topologies is compared to a null distribution

of differences.

PSEED = *integer-value*

> By default, PAUP* references the system clock to initialize the seed used to generate sequences of pseudorandom numbers for randomizing characters. An explicit seed may be specified to override the system clock default. (See "INITSEEDS = 0 | 1" on page 116 for more details.)

NREPS = *integer-value*

> Use NREPS to select the number of times the original data set is permuted for the null distribution of tree scores or tree score differences.

SEARCH = HEURISTIC | BANDB

> Use SEARCH to select a tree searching strategy to be used on each permutation of the original data set.

RANDOMIZE = ALL | INGROUP

> By default PAUP* randomizes the character states of all taxa used in the analyses. If RANDOMIZE = INGROUP, then only the character states of the ingroup taxa are randomized.

FIRST = *constraint-tree-name*
SECOND = *constraint-tree-name*

> If TEST = COMPARE2, use the FIRST and SECOND options to name the constraint trees to be compared.

CONSTRAINTS = *constraint-name*

> Specifies the constraint tree to be used for the T-PTP test.

CONVERSE = YES | NO

> See "Options And Subcommands Affecting Multiple Commands" on page 30 for a description of this option.

PSCORES

Use the PSCORES command to request a listing of tree lengths and/or fit measures for one or more trees.

Syntax

PSCORES [*tree-list*] [/*options*];

If no *tree-list* is specified, ALL is assumed. By default, only tree lengths (for single characters, overall, or both) are output.

Menu equivalent

Trees > Tree Scores > Parsimony ...

Available options

- SINGLE = <u>NO</u> | VAR | ALL
- RANGE = YES | <u>NO</u>
- TOTAL = <u>YES</u> | NO
- TL = <u>YES</u> | NO
- CI = YES | <u>NO</u>
- RI = YES | <u>NO</u>
- RC = YES | <u>NO</u>
- HI = YES | <u>NO</u>
- GFIT = YES | <u>NO</u>
- KHTEST = YES | <u>NO</u>
- NONPARAMTEST = YES | <u>NO</u>
- TESTDETAILS = YES | <u>NO</u>
- *SORTTREES = YES | <u>NO</u>
- SCOREFILE = *score-file-name*
- *APPEND = YES | <u>NO</u>
- *REPLACE = YES | <u>NO</u>
- *DISPLAYOUT = YES | <u>NO</u>

*Option is nonpersistent

Description of options

SINGLE = ALL | VAR | <u>NO</u>

The SINGLE option specifies the type of single-character output. If SINGLE = ALL, lengths (or fit measures) are output for all trees specified by *tree-list.* If SINGLE = VAR, lengths (or fit measures) are output only for those characters whose lengths vary among the trees specified by tree-list. If SINGLE = NO, no single-character tree lengths (or fit measures) are output.

RANGE = YES | <u>NO</u>

If RANGE is specified, only the minimum and maximum tree lengths (or best and worst fit measures) are output for each character.

TOTAL = <u>YES</u> | NO

If TOTAL = NO is specified, overall tree lengths and fit measures are not output.

TL = <u>YES</u> | NO

> Unless TL = No is specified, tree lengths are output. The default is to show tree lengths and no other fit measures.

CI = YES | <u>NO</u>

> Use CI to request output of consistency indices.

RI = YES | <u>NO</u>

> Use RI to request output of retention indices.

RC = YES | <u>NO</u>

> Use RC to request output of rescaled consistency indices.

HI = YES | <u>NO</u>

> Use HI to request output of homoplasy indices.

GFIT = { YES | <u>NO</u> }

> Use GFIT to request output of Goloboff-fits.

KHTEST = YES | <u>NO</u>

> Use KHTEST to request that the Kishino-Hasegawa test be used to compare pairs of topologies.

NONPARAMTEST = YES | <u>NO</u>

> Use NONPARAMTEST to request that the Templeton (Wilcoxon signed-rank) and winning-sites (sign) tests be used to compare pairs of topologies.

> > TESTDETAILS = YES | <u>NO</u>

> > > If NONPARAMTEST = YES, then use TESTDETAILS to display a detailed summary of the Templeton (Wilcoxon signed-rank) test calculations.

SORTTREES = YES | <u>NO</u>

> When multiple trees are evaluated under the PSCORES command SORTTREES = YES will sort tree scores so that they are displayed in ascending order. Note, SORTTREES = YES may noticably increase the time required to display tree scores.

SCOREFILE = *score-file-name*

> If SCOREFILE is specified, parsimony scores for each tree evaluated are saved to a text file. The *score-file-name* must follow the conventions described under "Input/Output files" on page 9.

> > APPEND = YES | <u>NO</u>
> > REPLACE = YES | <u>NO</u>

> > > If the *score-file-name* already exists, you will be prompted for confirmation before the existing file is replaced. Use REPLACE

= YES to suppress this prompt; the existing file will be quietly overwritten by the new data. Alternatively, you may specify APPEND, in which case a new TREES block will be concatenated to the end of an existing file.

DISPLAYOUT = YES | <u>No</u>

Set DISPLAY = YES if you wish to suppress output to the display buffer. The DISPLAYOUT option must be used in conjunction with the SCOREFILE option.

PSET
_____

Use the PSET command to set options for parsimony analysis.

Syntax

PSET [*options*];

Menu equivalent

Analysis > Parsimony Settings ...

Available options

- COLLAPSE = NO | <u>MAXBRLEN</u> | MINBRLEN | AMBEQUAL
- MSTAXA = <u>UNCERTAIN</u> | POLYMORPH | VARIABLE
- OPT = <u>ACCTRAN</u> | DELTRAN | MINF
- STEPMATRIX = OBSONLY | ALLSTATES | <u>THREEPLUS1</u>
- INCLUDEANC YES | <u>No</u>
- ANCSTATES = *ancstates-name*
- GK = *integer-value*
- GOLOBOFF = YES | <u>No</u>
- GUNINF = <u>EXCLUDE</u> | INCLUDE
- GPEEWEE = YES | <u>No</u>
- MINFORFIT = <u>MINLENGTH</u> | RANGE
- GAPMODE = <u>MISSING</u> | NEWSTATE

Description of options

COLLAPSE = NO | <u>MAXBRLEN</u> | MINBRLEN | AMBEQUAL

Use COLLAPSE to specify the criterion for collapsing branch lengths. MAXBRLEN is the default method used in PAUP* and the method used in PAUP version 3 (and Hennig86). MINBRLN collapses a branch if it is possible for it to have zero length. AMBEQUAL collapses a branch if the MPRset of the two incident nodes are identical. These latter two methods were introduced in Goloboff's NONA program.

We do not recommend their use, but they are available for those who wish to use them.

MSTAXA = <u>UNCERTAIN</u> | POLYMORPH | VARIABLE

If one or more taxa are coded as having multiple states use MSTAXA to specify how those characters are treated. If MSTAXA = VARIABLE the punctuation used to enclose multistate characters is respected. More specifically, characters enclosed by "{}" are treated as variable and characters enclosed by "()" are treated as polymorphic.

> MINFORFIT = <u>MINLENGTH</u> | RANGE
>
> If MSTAXA is specified, then use MINFORFIT to specify the "minimum" values used for calculating CI, RI, and RC indices. You may specify minimum-possible single-character lengths (MINLENGTH) or character "ranges" (RANGE) be used.

STEPMATRIX = OBSONLY | ALLSTATES | <u>THREEPLUS1</u>

By default PAUP* allows assignment of states not observed in terminal taxa to internal nodes but only those states that can be identified as potential shortcuts by the "3+1" test (STEPMATRIX = THREEPLUS1). If STEPMATRIX = ALLSTATES, then any possible character state may be assigned to an internal node. If STEPMATRIX = OBSONLY, then internal-node state assignments are limited to states observed in the terminal data.

INCLUDEANC = YES | <u>NO</u>

Use INCLUDEANC to include a hypothetical taxon possessing putative ancestral states for all characters in the searches. This taxon specifies the root of the tree.

GOLOBOFF = YES | <u>NO</u>

Use GOLOBOFF to specify the Goloboff-fit criterion.

> GK = *integer-value*
>
> Use GK to specify the concavity parameter in Goloboff's implied weights method. Note that this value corresponds to the definition of K given by Goloboff (1993) and not the CONC parameter in Goloboff's Pee-Wee program; which is equal to K+1. The default value for this option is 2.
>
> GUNINF = <u>EXCLUDE</u> | INCLUDE
>
> If GUNINF = INCLUDE, then fits for "uninformative" characters are included.
>
> GPEEWEE = YES | <u>NO</u>
>
> Although it is not recommended, you may specify GPEEWEE to emulate results obtained by the program Pee -Wee written by P. Goloboff. Calculations are performed using integer arithmetic with weights ranging from 0 to 100, and fractional values are truncated rather than rounded.

GAPMODE = <u>MISSING</u> | NEWSTATE

> If GAPMODE = MISSING, gap characters in sequence data are treated as "missing." If GAPMODE = NEWSTATE, gap characters are treated as a fifth base or 21st amino acid.

See "Options And Subcommands Affecting Multiple Commands" on page 30 for a description of the following options:

ANCSTATES = *ancstates-name*
OPT = <u>ACCTRAN</u> | DELTRAN | MINF

PUZZLE

Use the PUZZLE command to find a tree by the quartet puzzling method. This method is currently supported only for the maximum likelihood criterion.

Syntax

PUZZLE [*options*];

Menu equivalent

Analysis > Quartet Puzzling ...

Available options

- NPUZZLES = *integer-value*
- SEED = *integer-value*
- SHOWTREE = <u>YES</u> | NO
- GRPFREQ = <u>YES</u> | NO
- CUTOFFPCT = *integer-value*
- USEAPPROXL = <u>YES</u> | NO
- TREEFILE = *tree-file-name*
- *REPLACE = YES | <u>NO</u>
- *APPEND = YES | <u>NO</u>
- DSTATUS = *integer-value* | NONE
- MINMEMREQ = YES | <u>NO</u>

*Option is nonpersistent

Description of options

NPUZZLES = *integer-value*

> Use NPUZZLES to specify the number of puzzling steps. The default value for this option is 1000

SEED = *integer-value*

By default, PAUP* references the system clock to initialize the seed used to generate sequences of pseudorandom numbers used to determine the order of sequence insertion during the "puzzling" step. An explicit seed may be specified to override the system clock default. (See "INITSEEDS = 0 | 1" on page 116 for more details.)

SHOWTREE = <u>YES</u> | NO

Use SHOWTREE to print the tree found by quartet-puzzling to the display buffer.

GRPFREQ = <u>YES</u> | NO

Use GRPFREQ to show all observed bipartitions found during the "puzzling" step and the frequency at which they occurred.

CUTOFFPCT = *integer-value*

Use CUTOFFPCT to specify the minimum puzzle partition frequency to be displayed in the partition frequency table. Partitions occurring at a frequency below this value are not shown.

USEAPPROXL = <u>YES</u> | NO

If USEAPPROXL = YES, then approximate rather than "exact" likelihoods will be used.

TREEFILE = *tree-file-name*

If TREEFILE is specified, a description of the tree found at each puzzling step is output to a file in the NEXUS tree format. The *tree-file-name* must follow the conventions described under *"Input/Output files" on page 9*

REPLACE = YES | <u>NO</u>
APPEND = YES | <u>NO</u>

If the specified file already exists, you will be prompted for confirmation that the existing file should be replaced. Explicit specification of REPLACE suppresses this prompt and the existing file will be quietly overwritten by the new data. Alternatively, you may specify APPEND, in which case a new TREES block will be concatenated to the end of an existing file.

DSTATUS = *integer-value* | <u>NONE</u>

See "Options And Subcommands Affecting Multiple Commands" on page 30 for a description of this option.

MINMEMREQ = YES | <u>NO</u>

Specifying MINMEMREQ = YES minimizes memory requirements but also can increase the amount of time required to complete an analysis.

QUIT

The QUIT command causes PAUP* to terminate. Unless you specify the option WARNTSAVE = NO, PAUP* will prompt you before quitting if there are unsaved trees in memory. WARNTSAVE is the only option under this command.

Syntax

QUIT [*options*];

Menu equivalent

File > Quit

Available options

- WARNTSAVE = <u>YES</u> | NO

RANDTREES

Use the RANDTREES command to randomly sample trees from the set of all possible trees and compute their scores under the current optimality criterion. The results are shown in the form of a frequency distribution of tree scores.

Syntax

RANDTREES [*options*];

Menu equivalent

Analysis > Evaluate Random Trees ...

Available options

- NREPS = *integer-value*
- SEED = *integer-value*
- FDTYPE = <u>BARCHART</u> | HISTOGRAM
- NCLASSES = *integer-value*
- FDFILE = *frequency-distribution-file-name*
- *REPLACE = YES | <u>NO</u>

*Option is nonpersistent

Description of options

NREPS = *integer-value*

integer-value specifies the number of random trees to be evaluated. The default is NREPS =1000.

SEED = *integer-value*

By default, PAUP* references the system clock to initialize the seed used to generate sequences of pseudorandom numbers used to randomly sample tree topologies. An explicit seed may be specified to override the system clock default. (See "INITSEEDS = 0 | 1" on page 116 for more details.)

FDTYPE = <u>BARCHART</u> | HISTOGRAM

The default FDTYPE specifies that the number of trees at each tree score is output as a BARCHART. If FDTYPE = HISTOGRAM, tree scores are pooled into classes, the size of which is specified by the NCLASSES option.

NCLASSES = *integer-value*

Integer-value specifies the number of class intervals for the frequency distribution of tree scores. By default, NCLASSES=20, so that adjacent tree scores are pooled into 20 classes of tree scores.

FDFILE = *frequency-distribution-file-name*

If FDFILE is specified, data for the frequency distribution of tree scores are saved to a text file for input to other programs. The FDFILE option is not persistent. *Frequency-distribution-file-name* specifies the name of the file to receive the frequency distribution. (See "Input/Output files" on page 9 for more details).

REPLACE = YES | <u>NO</u>

If FDFILE is used and the specified file already exists you will be prompted for confirmation that the existing file should be replaced. REPLACE suppresses this prompt and the existing file will be quietly overwritten by the new data.

RATESET

Use the RATESET command to define a "rate set".

Syntax

RATESET *rate-set-name* = *rateset-definition* ;

Example 1

In the following example, the RATESET command is used to define three gene fragments according to their position in the data matrix.

```
rateset genes=1:1-300, 2:301-600, 3:601-700;
```

Example 2

In the following example, the RATESET command is used to defined codon positions base on previously defined character sets (see the "CharSet" command described on page 23).

```
charset 1stpos = 2-457\3 660-896\3;
charset 2ndpos = 3-457\3 661-896\3;
charset 3rdpos = 4-457\3 662-.\3;
rateset codons=1:1stpos, 2:2ndpos, 3:3rdpos;
```

RECONSTRUCT

Use the RECONSTRUCT command to request output of character-state reconstructions for one or more characters on one or more trees. The reconstruction is shown by superimposing the character-states assigned to each node on a plot of the tree. Reconstructions are available under either the parsimony or likelihood criterion.

Syntax

RECONSTRUCT [*character-list*] [/*options*];

The *character-list* specifies the character(s) for which reconstructions are shown, and consists of one or more character numbers, character names, or character-set names.

Menu equivalent

Trees > Show Reconstructions ...

Available options

- TREES = *tree-list*
- TCOMPRESS = YES | NO

Description of options

TREES = *tree-list*

The *tree-list* specifies the tree numbers for which reconstructions are to be shown. If this is the first RECONSTRUCT command and you do not specify a tree list, reconstructions are shown for the first tree only. If other options follow the TREES option you must the put the *tree-list* list in double quotes.

TCOMPRESS = YES | No

See "Options And Subcommands Affecting Multiple Commands" on page 30 for details.

If no characters are specified for either a RECONSTRUCT or MPRSETS (page 94) command, the characters are taken to be those plotted in the last invocation of either of these commands. For example, `reconstruct 1 3 5 7; mprets;` will cause both commands to output information for characters 1, 3, 5, and 7.

RESTORE

The RESTORE command is synonymous with the UNDELETE command. See "Undelete" on page 132 for a description of available options.

REVFILTER

Use the REVFILTER command to "reverse" the effect of the current filter. All trees that are currently hidden by the filter will become visible, and all trees that were previously visible will be hidden. There are no options.

Syntax

REVFILTER;

Menu equivalent

Trees > Filter > Reverse Filter

REWEIGHT

Use the REWEIGHT command to assign weights to the characters based on their fit to the trees currently in memory.

Syntax

REWEIGHT [*options*];

Menu equivalent

Data > Reweight Characters ...

Available options

- INDEX = <u>RC</u> | CI | RI
- FIT = <u>MAXIMUM</u> | MINIMUM | MEAN
- BASEWT = *integer-value*
- TRUNCATE = YES | <u>NO</u>
- MINFORFIT = <u>RANGE</u> | MINLENGTH

Description of options

BASEWT = *integer-value*

> Base-weight specifies the maximum possible weight that a character can be assigned, corresponding to an index value (see below) of 1. Weights are scaled from 0 to this value. The default is BASEWT=1.

INDEX = RC | CI | RI

> The INDEX option is used to specify which fit measure to use when calculating the new character weights (RC = rescaled consistency index, CI = consistency index, RI = retention index).

FIT = MAXIMUM | MINIMUM | MEAN

> The FIT option is used to specify whether the new weights are based on the maximum, minimum, or mean of the fit values for each character over all of the trees in memory.

TRUNCATE = YES | NO

> Because PAUP* uses integers to represent weights, they are ordinarily scaled to the nearest integer by rounding. Use the TRUNCATE option if you want to simply discard the fractional part (e.g., 8.7 goes to 8 rather than 9). This option is primarily intended to duplicate the weights obtained using the Hennig86 program (Farris, 1988).

MINFORFIT = RANGE | MINLENGTH

> If the data matrix contains polymorphic or uncertain characters, use MINFORFIT to specify the "minimum" values used for calculating CI, RI, and RC indices. You may specify minimum-possible single-character lengths (MINLENGTH) or character "ranges" (RANGE) be used.

ROOTTREES

Use the ROOTTREES command to convert all trees in memory from an unrooted to a rooted representation. Trees are rooted according to the currently specified outgroup.

Syntax

> ROOTTREES [*options*];

Menu equivalent

> Trees > Root Trees

Available options

> • OUTROOT = POLYTOMY | PARAPHYL | MONOPHYL

- METHOD = <u>OUTGROUP</u> | LUNDBERG | MIDPOINT
- USERBRLENS = YES | <u>NO</u>

Description of options

OUTROOT = <u>POLYTOMY</u> | PARAPHYL | MONOPHYL

See "Options And Subcommands Affecting Multiple Commands" on page 30 for details.

METHOD = <u>OUTGROUP</u> | LUNDBERG | MIDPOINT

The METHOD option is used to specify how unrooted trees are to be rooted prior to output. You can choose OUTGROUP rooting, using whichever outgroup you have selected; MIDPOINT rooting, which roots the tree at its midpoint; or LUNDBERG rooting, which requires that a previous ANCSTATES command has been issued. By default, OUTGROUP rooting is in effect.

USERBRLENS = YES | <u>NO</u>

If METHOD = MIDPOINT, then you may specify user-defined branch lengths.

SAVEASSUM

Use the SAVEASSUM command to save the current character-type, character-weight, character-exclusion, ancestral-states, taxon-deletion, and outgroup status to a file in ASSUMPTIONS and a PAUP block. You can restore the settings in effect at the time the SAVEASSUM command was issued simply by executing the commands stored in the file (use the "Execute" command described on page 60 for this purpose).

Syntax

SAVEASSUM [*options*];

Menu equivalent

Data > Save Current Assumptions ...

Available options

- FILE = *settings-file-name*
- REPLACE = YES | <u>NO</u>

*Option is nonpersistent

Description of options

FILE = *settings-file-name*

The FILE option is used to specify the name and location of the file

with the current assumptions settings. The *settings-file-name* must follow the conventions described under "Input/Output files" on page 9.

REPLACE = YES | <u>NO</u>

By default, if the specified file already exists, you will be asked if you want to replace it . To suppress this warning, specify REPLACE; in this case the file will be quietly overwritten.

SAVEDIST

Use the SAVEDIST command to save distance matrix to a file.

Syntax

SAVEDIST [*options*];

Menu equivalent

File > Save Distances To File ...

Available options

- FORMAT = <u>ONECOLUMN</u> | TABTEXT | NEXUS | PHYLIP
- FILE = *export-file-name*
- TRIANGLE = <u>LOWER</u> | UPPER | BOTH
- DIAGONAL = YES | <u>NO</u>
- UNDEFINED = <u>TWICEMAX</u> | ASTERISK
- NDECIMALS = *integer-value*
- *REPLACE = YES | <u>NO</u>
- *APPEND = YES | <u>NO</u>

*Option is nonpersistent

Description of options

FORMAT = <u>ONECOLUMN</u> | TABTEXT | NEXUS | PHYLIP

Use FORMAT to request that pairwise distances be output as tab-delimited text either as a full matrix format (TABTEXT) or in a single-column (ONECOLUMN) useful for saturation plots, etc.

TRIANGLE = <u>LOWER</u> | UPPER | BOTH

If any FORMAT other than ONECOLUMN is selected than the distances matrix may be output as a lower half triangle (LOWER), an upper half triangle (UPPER), or full matrix (BOTH).

DIAGONAL = YES | <u>NO</u>

Use the DIAGONAL option to include the diagonal values if relevant.

UNDEFINED = <u>TWICEMAX</u> | ASTERISK

Use UNDEFINED to specify whether undefined distances are output as twice the maximum defined distance (TWICEMAX) or as an asterisk (ASTERISK).

NDECIMALS = *integer-value*

Use NDECIMALS to specify the maximum number of decimals allowed for pairwise distances.

FILE = *export-file-name*

Use the FILE to output pairwise distances to a file in one of two formats. The *export-file-name* must follow the file-name conventions described "Input/Output files" on page 9.

REPLACE = YES | <u>NO</u>
APPEND = YES | <u>NO</u>

If FILE = *export-file-name* option is specified and the *export-file-name* already exists, you will be prompted for confirmation that the existing file should be replaced. Explicit specification of REPLACE = YES suppresses this prompt; the existing file will be quietly overwritten by the new data. Alternatively, you may specify APPEND = YES, in which case new distances in the specified format will be concatenated to the end of an existing file.

SAVETREES

Use the SAVETREES command to write trees currently in memory to a file as a NEXUS-format TREES block or as a treefile accepted by another program.

Syntax

SAVETREES [*options*];

Menu equivalent

Trees > Save Trees to File ...

Available options

- FILE = *tree-file-name*
- FORMAT = <u>NEXUS</u> | ALTNEXUS | FREQPARS | PHYLIP | HENNIG
- ROOT = YES | <u>NO</u>
- BRLENS = YES | <u>NO</u>
- SAVEBOOTP = <u>NO</u> | BRLENS | NODELABELS | BOTH

- MAXDECIMALS = *integer-value*
- FROM = *integer-value*
- TO = *integer-value*
- *REPLACE = YES | NO
- *APPEND = YES | NO

*Option is nonpersistent

Description of options

FORMAT = <u>NEXUS</u> | ALTNEX | FREQPARS | PHYLIP | HENNIG

Specifies the type of tree-file to produce. NEXUS requests a file containing the standard NEXUS TREES block using a translation table (which greatly reduces the amount of disk space required to store the trees). ALTNEX also specifies a NEXUS TREES block, but no translation table is used (the full taxon names are included in each tree description). FREQPARS requests a treefile for the FREQPARS program described by Swofford and Berlocher (1987). PHYLIP requests a treefile for input to version 3.4 of Felsenstein's (1991) PHYLIP package. HENNIG requests a treefile for version 1.5 of Farris's (1988) Hennig86 program.

BRLENS = YES | <u>NO</u>

If BRLENS is specified, tree descriptions will include branch lengths if the program corresponding to the FORMAT setting supports them. BRLENS = NO reverses the effect of a previous BRLENS specification. Branch lengths are calculated according to the current optimality criterion. If the SAVETREE command is given following the BOOTSTRAP (page 39) or JACKKNIFE command (page 77), branch lengths will represent bootstrap support values. To save branch lengths for individual bootstrap or jackknife replicate trees use the TREEFILE and BRLENS options under the BOOTSTRAP (page 39) or JACKKNIFE (page 77) commands.

SAVEBOOTP = NO | <u>BRLENS</u> | NODELABELS | BOTH

Use SAVEBOOTP to include bootstrap or jackknife proportions as branch lengths (BRLENS), internal node labels (NODELABELS), or both (BOTH). Note, branch lengths saved as node labels are only recognized by other programs.

MAXDECIMALS = *integer-value*

If branch lengths will be included in tree descriptions, use MAXDECIMALS to specify the maximum number of decimals stored in each tree description.

FILE = *tree-file-name*

Specifies a name for the tree file. If you do not explicitly specify a file name, a default file name will be used.

REPLACE = YES | <u>NO</u>
APPEND = YES | <u>NO</u>

> If FILE = *tree-file-name* option is specified and the *tree-file-name* already exists, you will be prompted for confirmation that the existing file should be replaced. Explicit specification of REPLACE = YES suppresses this prompt; the existing file will be quietly overwritten by the new data. Alternatively, you may specify APPEND = YES, in which case a new TREES block will be concatenated to the end of an existing file.

FROM = *integer-value*

> If nonzero, *starting-tree-number* specifies the number of the first tree to save.

TO = *integer-value*

> If nonzero, *ending-tree-number* specifies the number of the last tree to save.

ROOT = YES | <u>NO</u>

> If the current trees in memory are unrooted and ROOT is specified, trees are rooted (using the rooting options currently in effect) before they are saved. Note that the process of rooting the trees slows down the saving operation considerably. If you are saving the trees only with the intention of rereading them into PAUP*, there is no need to root the trees. However, if you are exporting the trees to another program, rooting them may be desirable.

> The ROOT option has no effect if the trees in memory are already rooted. ROOT = NO can be used to reverse the effect of a previous ROOT specification.

SET

The SET command is used to set a variety of options whose scope extends beyond single commands.

Syntax

SET [*options*];

Menu equivalent

Options

Available options

- ROOT = <u>OUTGROUP</u> | LUNDBERG | MIDPOINT
- INITSEEDS = <u>0</u> | 1
- MONITOR = <u>YES</u> | NO

- SEMIGRAPH = YES | <u>NO</u>
- MAXTREES = *integer-value*
- INCREASE = NO | <u>PROMPT</u> | AUTO
- AUTOINC = *integer-value*
- SHOWEXCLUDED = YES | <u>NO</u>
- ALLOWPUNCT = YES | <u>NO</u>
- CMLABELS = YES | <u>NO</u>
- CMSHOWEQ = <u>YES</u> | NO
- CMCOLWID = *integer-value*
- CMCSTATUS = YES | <u>NO</u>
- CONSTRAINTS = *constraint-name*
- ERRORBEEP = <u>YES</u> | NO
- QUERYBEEP = YES | <u>NO</u>
- KEYBEEP = YES | <u>NO</u>
- NOTIFYBEEP = <u>YES</u> | NO
- ERRORSTOP = <u>YES</u> | NO
- AUTOCLOSE = YES | <u>NO</u>
- TORDER = <u>STANDARD</u> | RIGHT | LEFT | ALPHABET
- CHECKEVTS = <u>YES</u> | NO
- TCOMPRESS = YES | <u>NO</u>
- SHOWTAXNUM = YES | <u>NO</u>
- BACKGROUND = <u>YES</u> | NO
- STATUS = <u>YES</u> | NO
- OUTROOT = <u>POLYTOMY</u> | PARAPHYL | MONOPHYL
- VISNOTIF = NONE | <u>SHOWALERT</u> | FLASHONLY
- FLUSHLOG = YES | <u>NO</u>
- ALLOWEND = YES | <u>NO</u>
- ALLDIGLAB = PROHIBIT | <u>WARN</u> | NOWARN
- CRITERION = <u>PARSIMONY</u> | LIKELIHOOD | DISTANCE
- DSTATUS = *integer-value* | NONE
- DEFAULTMODE = YES | <u>NO</u>
- STOREBRLENS = YES | <u>NO</u>
- STORETREEWTS = YES | <u>NO</u>
- WARNRESET = <u>YES</u> | NO
- WARNTREE = <u>YES</u> | NO
- WARNTSAVE = <u>YES</u> | NO
- WARNBLKNAME = <u>YES</u> | NO
- WARNROOT = <u>YES</u> | NO

- WARNREDEF = <u>YES</u> | NO
- SHOWABBREV = YES | <u>NO</u>
- PAUSE = <u>NO</u> | SILENT | BEEP | MSG
- TAXLABELS = FULL | <u>TRUNCATE</u>
- DROPMODE = EDIT | <u>CONDITIONAL</u> | EXECUTE

Description of options

MONITOR = <u>YES</u> | NO

> MONITOR = NO suppresses output to the "main display" (window or terminal screen) and is useful when you want to send output to the log file and/or printer only. MONITOR = YES reactivates the main display.

> At least one output destination must be active at all times. Consequently, if no log file is active or the "echo to printer" (ECHO) option is not set, output will be sent to the main display even if MONITOR = NO has been requested.

INITSEEDS = <u>0</u> | 1

> A few commands used in PAUP* require sequences of "random" numbers (i.e., BOOTSTRAP, JACKKNIFE, HOMPART, PERMUTE, RANDTREES, PUZZLE, and SURFCHECK). Other command also use "random" numbers, but only when certain options are specified (i.e., GENERATETREES, HSEARCH, NJ, and STARDECOMP). PAUP* uses a linear congruential method starting with an integer between 1 and 2,147,483,646 to seed a random number generator. By default INITSEEDS = 0, instructs PAUP* to obtain the initial seed from the system clock. User may also set INITSEEDS=1, in which case all seeds are initialized to 1. If you do not specify a starting seed for subsequent runs, the seed defaults to the next number in the random number sequence initiated during the previous run. Users are always given the option to specify a value for the seed on any command that accepts seed settings, thus overriding the PAUP* default of referencing the system clock. This INITSEEDS option ONLY applies if a seed value has not been initialized during the current run of the program. You can reset the sequence using a clock-based seed value by specifying a seed value of 0 on any command that accepts seed settings.

ALLOWPUNCT = YES | <u>NO</u>

> Unless ALLOWPUNCT is specified, taxon and character names in input data matrices must conform to the NEXUS specification; namely they cannot contain special punctuation characters like parentheses, hyphens, etc. (see "Identifiers" on page 6). If ALLOWPUNCT = YES is set prior to processing of a DATA or CHARACTERS block, then taxon and character names are allowed to contain these special characters. This option applies only to the DATA and CHARACTERS blocks; names containing special characters must be enclosed within single quotes in other contexts. This option allows backward compatibility with files prepared for PAUP 3.1;

new data files should always conform to the NEXUS standard to prevent incompatibilities with other programs.

SEMIGRAPH = YES | <u>NO</u>

PAUP* uses special characters in its internal font to draw trees and other items. On the IBM-PC, these characters are nonstandard "high ASCII" characters. On the Macintosh, these characters are neither in the standard 128 ASCII characters nor in the set of special characters normally included with Macintosh fonts. Thus, although the trees look nice when drawn in the main display window, they may not look right when printed on some printers. Therefore, PAUP* ordinarily translates these "semigraphics" characters to standard ASCII substitutes when output is directed to a printer, file, or document window. If you want to override this behavior, specify SEMIGRAPH. (E.g., many IBM-PC printers can print the high ASCII characters, and the Apple LaserWriter can create a "bit map" version of PAUP*'s internal "PAUPMonaco" font).

This option is relevant only for DOS, Windows, and Macintosh interfaces.

AUTOCLOSE = YES | <u>NO</u>

After a search has ended the status window will remain in the foreground until the user closes it. Use AUTOCLOSE = YES to close the status window automatically after a run has ended.

NOTIFYBEEP = <u>YES</u> | NO

Unless NOTIFYBEEP = NO, a "beep" is sounded at the end of a search.

TORDER = <u>STANDARD</u> | RIGHT | LEFT | ALPHABET

Specifies the convention used to "order" the tree.

FLUSHLOG = YES | <u>NO</u>

Specification of FLUSHLOG causes the file's buffer to be flushed after every line of output. Ordinarily, this degrades system performance and is not recommended. However, there may be situations in which immediate flushing is useful.

DROPMODE = EDIT | <u>CONDITIONAL</u> | EXECUTE

The DROPMODE option allows the user to always edit (EDIT) or always execute (EXECUTE) when a PAUP* document is dragged onto the application icon or PAUP*'s main window. Otherwise, under the default setting (DROPMODE=CONDITIONAL) the file will be executed if no active data file has been set. If a data file is already active, the file will be edited. The DROPMODE option is only available to the Macintosh and Windows interfaces.

ALLOWEND = <u>YES</u> | NO

The NEXUS convention is to terminate blocks with END; however, earlier versions of PAUP used ENDBLOCK. Unless ALLOWEND = NO, PAUP* will accommodate both character strings for block

termination.

ALLDIGLAB = PROHIBIT | <u>WARN</u> | NOWARN

By default, PAUP* issues a warning if all digit taxa or character labels are included. Use ALLDIGLAB to prohibit the use of all-digit labels (PROHIBIT) or to allow all-digit labels and suppress warnings (NOWARN). See "Identifiers" on page 6 for a description of valid taxa and character labels.

CRITERION = <u>PARSIMONY</u> | LIKELIHOOD | DISTANCE

Use CRITERION to selection from the three optimality criterian.

DEFAULTMODE = YES | <u>NO</u>

By default when option settings or other information is required, a dialog box will be opened. If DEFAULTMODE = YES, then PAUP* will take a default action. Because the default action will not be appropriate in every situation, this option is not recommended unless you know what affect the default action will have on your analyses.

STOREBRLENS = YES | <u>NO</u>

If branch lengths are present in the TREES block, then STOREBRLENS = YES stores branch lengths for subsequent use within the program. Otherwise, the branch lengths are discarded.

STORETREEWTS = YES | <u>NO</u>

If tree weights are present in the TREES block, then STORETREEWTS = YES stores tree weights. When tree weights are the reciprocal of the number of trees found in either a BOOTSTRAP or JACKKNIFE replicate, this option allows the combination of bootstrap results from runs performed at different times or on different machines and the recovery of results obtained prior to a system crash. (See "USETREEWTS = YES | NO" on page 46 for more details)

SHOWABBREV = YES | <u>NO</u>

Specifies whether help shows command and option names so as to indicate the shortest acceptable abbreviation.

TAXLABELS = FULL | <u>TRUNCATE</u>

If TAXLABELS = FULL, then the full 127 characters of long taxon labels will be output to the display buffer (although they will still be truncated somewhat if the total output width cannot accommodate the long names). Otherwise, taxon names are truncated to 16 characters.

ERRORSTOP = <u>YES</u> | NO

Ordinarily, PAUP* stops processing an input file when unrecognized commands, OPTIONS-command keywords, or formats are encountered. If ERRORSTOP = NO is specified, a warning message is issued and processing is allowed to continue.

WARNRESET = <u>YES</u> | NO

> Ordinarily, PAUP* issues a warning message when an input file containing a DATA block is executed and a DATA block has already been processed. Specification of WARNRESET = NO suppresses this warning.

WARNTREE = <u>YES</u> | NO

> Use WARNTREE to issue a warning about treefile operations that could delete unsaved trees.

WARNTSAVE = <u>YES</u> | NO

> Use WARNTSAVE to issue a warning before quitting if there are unsaved trees in memory.

WARNBLKNAME = <u>YES</u> | NO

> Use WARNBLKNAME to issue a warning when unrecognized blocknames are included in a data set.

WARNROOT = <u>YES</u> | NO

> Use WARNROOT to issue a warning about the rooting status of user-input trees.

WARNREDEF = <u>YES</u> | NO

> Use WARNREDEF to issue a warning about redefining names for sets, constraints, etc.

PAUSE = <u>NO</u> | SILENT | BEEP | MSG

> By default, output is displayed to the display buffer without pause. Alternatively, PAUP* will display one screen of output at a time either pausing silently (SILENT), issuing a warning beep (BEEP), or printing a message to the screen (MSG) for the user press return key before displaying the next screen.

SHOWTAXNUM = YES | <u>NO</u>

> When SHOWTAXNUM is set to YES trees are displayed with both taxa labels and taxa numbers. The number are based on the order in which the taxa appear in the CHARACTERS or DATA blocks.

The following three options affect the setting of the maximum number of trees that PAUP* can store at any given time:

MAXTREES = *integer-value*

> The MAXTREES parameter specifies the maximum number of trees that can be saved. Setting MAXTREES to a large value will reduce the likelihood that the tree buffer will become full during a search or tree-file operation, at the expense of a larger chunk of memory being tied up and therefore unavailable for other purposes.
>
> Ordinarily, if the number of trees found during a search reaches the value of MAXTREES, you will be given a chance to increase MAXTREES

before proceeding. This behavior can be altered using the INCREASE option (see below). MAXTREES is initially set to 100.

The MAXTREES option apply to the entire search in random-addition-sequence searches. If you would like to impose a maximum tree limit for each addition-sequence replicate then you must use the NCHUCK and CHUSKSORE options under the HSEARCH command (page 71). For replicated searches such as the bootstrap (page 39), the jackknife (page 77), and the partition-homogeneity test (page 70) the limits are always applied separately for each search.

INCREASE = <u>PROMPT</u> | AUTO | NO

The setting of the INCREASE option determines the action taken by PAUP* if the limit on the number of trees that can be stored (=MAXTREES, see above) is reached during a search or a tree-file operation. If INCREASE = PROMPT, you will be given the opportunity to increase MAXTREES. If INCREASE = AUTO, MAXTREES will automatically be increased by a number of trees equal to the current AUTOINC setting (see below). If INCREASE = NO, MAXTREES will not be increased, and no prompt will be issued. In this case, a "tree-buffer overflow" occurs which can affect the effectiveness of the search in progress. (The tree-buffer overflow condition will be documented in the output.)

AUTOINC = *integer-value*

The AUTOINC value species the number of trees by which MAXTREES is increased when the number of trees saved reaches MAXTREES and the INCREASE = AUTO option is in effect. AUTOINC is initially set to 100.

Example 1

In the following example, a maximum tree limit of 100 for each random-addition-sequence replicate is set. The NCHUCH and CHUCKSCORE options of the HSEARCH command (page 71) are used to set the limit for each replicate search.

```
set maxtrees=1000 increase=no;
hsearch addseq=random nreps=10 nchuck=100
chuckscore=1;
```

The following three options specify whether PAUP* sounds a "beep" when various kinds of errors occur:

ERRORBEEP = <u>YES</u> | NO

Ordinarily, PAUP* beeps to alert you that an error message has been issued. Specify ERRORBEEP = NO to suppress these beeps.

QUERYBEEP = YES | <u>NO</u>

Ordinarily, PAUP* beeps to alert you when it stops for your input before it can continue a process. Specify QUERYBEEP = NO to suppress these beeps.

KEYBEEP = YES | <u>NO</u>

>   Ordinarily, PAUP* beeps when you type a key that is invalid in the
>   current context. Specify KEYBEEP= NO to suppress these beeps.

The following three options are specific to the Macintosh version.

BACKGROUND = <u>YES</u> | NO

>   Ordinarily, PAUP* continues processing when it is moved to the
>   background. Specify BACKGROUND = NO to suppress background
>   processing, thereby giving more time to the foreground application.

CHECKEVTS = <u>YES</u> | NO

>   CHECKEVTS = NO disables "event-checking," causing all mouse clicks
>   an key presses to be ignored. Speed of PAUP* searches is improved
>   somewhat, but it will not be possible to stop the search (without
>   restarting the computer) or to switch to a different application under
>   MultiFinder.

VISNOTIF = NONE | <u>SHOWALERT</u> | FLASHONLY

>   When a background job ends an alert box will be sent to the
>   foreground and the PAUP* icon on the task-menu will flash. Use
>   VISNOTIF to suppress the alert box (FLASHONLY) or both of the alert
>   box and the flashing PAUP* icon (NONE).

The following options are described in "Options And Subcommands
Affecting Multiple Commands" command described on page 30:

> CONSTRAINTS = *CONSTRAINT-TREE-NAME*
> STATUS = <u>YES</u> | NO
> DSTATUS = *integer-value* | NONE
> ROOT = <u>OUTGROUP</u> | LUNDBERG | MIDPOINT
> OUTROOT = <u>POLYTOMY</u> | PARAPHYL | MONOPHYL
> TCOMPRESS = YES | <u>NO</u>
> SHOWEXCLUDED = YES | <u>NO</u>
> CMSHOWEQ = <u>YES</u> | NO
> CMLABELS = YES | <u>NO</u>
> CMCSTATUS = YES | <u>NO</u>
> CMCOLWID = *integer-value*

SHOWANC

Use the SHOWANC command to request a listing of the ancestral character-
states currently in effect. No options are available.

Syntax

> SHOWANC ;

Menu equivalent

Data > Show Other > AncStates

SHOWCHARPARTS

Use the SHOWCHARPARTS command to show one or all character-partition definitions.

Syntax

SHOWCHARPARTS [*partition-name*];

If no *partition-name* is specified, all defined partitions are output.

Menu equivalent

Data > Show Other > CharPartitions

SHOWCONSTR

Use the SHOWCONSTR command to show one or more constraint-tree definitions.

Syntax

SHOWCONSTR [{*constraint-name*|ALL}];

If you do not specify a name (or ALL), the current default constraint tree will be shown. Specify ALL to show all constraint trees that have been defined.

Menu equivalent

Analysis > Show Constraints

SHOWDIST

Use the SHOWDIST command to output a matrix of "distances" between taxa. There are no options.

Syntax

SHOWDIST;

Menu equivalent

Data > Show Pairwise Distances

Use the DSET command (page 55) to specify the distance measure to be calculated.

SHOWMATRIX

Use the SHOWMATRIX command to list the current data matrix.

Syntax

SHOWMATRIX [*options*];

Available options

- SHOWEXCLUDED = YES | <u>NO</u>
- CMLABELS = YES | <u>NO</u>
- CMSHOWEQ = <u>YES</u> | NO
- CMCOLWID = *integer-value*
- CMCSTATUS = YES | <u>NO</u>

The options available here are described under "Options And Subcommands Affecting Multiple Commands" on page 30.

SHOWRATESETS

Use the SHOWRATESETS command to show one or all rate-set definitions.

Syntax

SHOWRATESETS [*rate-set-name*];

Menu equivalent

Data > Show Other > RateSets

SHOWTAXPARTS

Use the SHOWTAXPARTS command to show one or all taxon-partition definitions.

Syntax

SHOWTAXPARTS [*partition-name*];

Menu equivalent

Data > Show Other > TaxPartitions

SHOWTREES

Use the SHOWTREES command to request a diagram of one or more trees with no other information (see also the DESCRIBE command).

Syntax

SHOWTREES [*tree-list*] [*/options*];

Menu equivalent

Trees > Show Trees ...

Available options

- TCOMPRESS = YES | <u>NO</u>
- SHOWTAXNUM = YES | <u>NO</u>
- ROOT = <u>OUTGROUP</u> | LUNDBERG | MIDPOINT
- OUTROOT = <u>POLYTOMY</u> | PARAPHYL | MONOPHYL
- USERBRLENS = YES | <u>NO</u>

Description of options

USERBRLENS = YES | <u>NO</u>

If USERBRLENS = YES, then user-supplied branch lengths are used.

SHOWTAXNUM = YES | <u>NO</u>

When SHOWTAXNUM is set to YES trees are displayed with both taxa labels and taxa numbers. The number are based on the order in which the taxa appear in the CHARACTERS or DATA blocks.

See "Options And Subcommands Affecting Multiple Commands" on page 30 for a description of the following options:

TCOMPRESS = YES | <u>NO</u>
ROOT = <u>OUTGROUP</u> | LUNDBERG | MIDPOINT
OUTROOT = <u>POLYTOMY</u> | PARAPHYL | MONOPHYL

SHOWUSERTYPE

Use the SHOWUSERTYPE command to show all user-defined character types. There are no options for the command.

Syntax

SHOWUSERTYPE [*user-type-name*];

Menu equivalent

Data > Show Other > UserTypes

SORTTREES

Use the SORTTREES command to sort trees that are currently in memory according to the current optimality criterion.

Syntax

SORTTREES;

Menu equivalent

Trees > Sort Trees

STARDECOMP

Use the STARDECOMP command to find a tree by star-decomposition search.

Syntax

STARDECOMP [*options*];

Menu equivalent

Analysis > Star Decomposition Search ...

Available options

- BREAKTIES = <u>SYSTEMATIC</u> | RANDOM
- TIESEED = *integer-value*
- ENFORCE = YES | <u>NO</u>
- CONSTRAINTS = *constraint-name*

Description of options

BREAKTIES = <u>SYSTEMATIC</u> | RANDOM

Unless BREAKTIES = RANDOM, ties encountered during tree construction are broken systematically according to the input order of the taxa.

TIESEED = *integer-value*

If BREAKTIES = RANDOM, PAUP* references the system clock to initialize the seed used to generate pseudorandom numbers for breaking ties found during tree construction. An explicit seed may be specified to override the system clock default. (See "INITSEEDS =

0|1" on page 116 for more details.)

See "Options And Subcommands Affecting Multiple Commands" on page 30 for details.

> ENFORCE = <u>YES</u> | NO
> CONSTRAINTS = *constraint-name*

SURFCHECK

Use the SURFCHECK command to check likelihood surface for multiple peaks.

Syntax

> SURFCHECK [*options*];

Available options

- NREPS = *integer-value*
- SEED = *integer-value*
- TREENUM = *integer-value*
- BRLENS = YES | <u>NO</u>

Description of options

NREPS = *integer-value*

Use NREPS to specify the number of passes ...

SEED = *integer-value*

PAUP* references the system clock to initialize the seed used to generate pseudorandom numbers for sampling "random" branch lengths. An explicit seed may be specified to override the system clock default. (See "INITSEEDS = 0|1" on page 116 for more details.)

TREENUM = *integer-value*

Use TREENUM to specify the tree for which the likelihood surface is checked.

BRLENS = { YES | NO }

If BRLENS = YES, then user-supplied branch lengths are used.

TAXPARTITION

Define a partition of the taxa.

Syntax

TAXPARTITION *partition-name* = *partition-definition* ;

The TAXPARTITION command is ordinarily issued from within the SETS block (page 22). However, you may also issue it from the command line or from within a PAUP block. TAXPARTITIONS are need for "convexity" constraints.

TAXSET

Use the TAXSET command to define a "taxon set." Taxon sets are simply groups of taxa that can be referred to by a single name in other commands.

Syntax

TAXSET *taxon-set-name* = *taxon-list* ;

The taxon-set-name must not be identical to any of the original taxon names. The TAXSET command is ordinarily issued from within the SETS block (page 22). However, you may also issue it from the command line or from within a PAUP block.

TIME

Use the TIME command to output the current time and date.

Syntax

TIME ;

TONEXUS

Use the TONEXUS command to convert a file to NEXUS format.

Syntax

TONEXUS [*options*] ;

Menu equivalent

File > Import Data ...

Available options

- FORMAT = <u>PHYLIP</u> | HENNIG86 | TABTEXT | TEXT | GCG | MEGA | PIR | FREQPARS
- FROMFILE = *foreign-file-name*

- TOFILE = *NEXUS-file-name*
- DATATYPE = <u>NUCLEOTIDE</u> | PROTEIN | RESTSITE | STANDARD | DISTANCE | DNA
- INTERLEAVE = YES | <u>NO</u>
- GAPSYMBOL = *single-character-value*
- MISSSYMBOL = *single-character-value*
- IDENTSYMBOL = *single-character-value*
- *REPLACE = YES | <u>NO</u>

*Option is nonpersistent

Description of options

FORMAT = <u>PHYLIP</u> | HENNIG86 | TABTEXT | TEXT | GCG | MEGA | PIR | FREQPARS }

Use FORMAT to specify the import file-type. You may choose from the following formats:

1. PHYLIP (Felsenstein)

2. Hennig68 (Farris)

3. GCG MSF

4. MEGA (Kumar, Tamura, and Nei)

4. NBRF-PIR

5. Text (tab or space delimited)

6. FREQPARS (Swofford and Berlocher)

FROMFILE = *foreign-file-name*

Use FROMFILE to specify the *foreign-file-name* to be imported.

TOFILE = *NEXUS-file-name*

Use TOFILE to specify the *NEXUS-file-name* used to save the NEXUS-formatted data set.

DATATYPE = <u>NUCLEOTIDE</u> | PROTEIN | RESTSITE | STANDARD | DISTANCE | DNA

If FORMAT = PHYLIP | TABTEXT | TEXT, then use DATATYPE to specify the type of data being imported. If FORMAT = MEGA, then DATATYPE = NUCLEOTIDE | PROTEIN | DISTANCE may be selected. Note that PAUP* currently does not support distance formats.

INTERLEAVE = YES | <u>NO</u>

Use INTERLEAVE if FORMAT = PHYLIP and the data set in the PHYLIP file are interleaved.

GAPSYMBOL = *single-character-value*

If FORMAT = TABTEXT | TEXT | MEGA, then a *single-character-value* may

be entered to represent gaps.

MISSSYMBOL = *single-character-value*

If FORMAT = TABTEXT | TEXT | MEGA, then a *single-character-value* may be entered to represent missing data.

IDENTSYMBOL = *single-character-value*

If FORMAT = TABTEXT | TEXT | MEGA, then a *single-character-value* may be entered to represent an identity character. This is equivalent to what PAUP* refers to as MATCHCHAR (see page 13).

REPLACE = YES | <u>No</u>

If the *NEXUS-file-name* used under the TOFILE option already exists you will be prompted for confirmation that the existing file should be replaced. REPLACE suppresses this prompt; the existing file will be quietly overwritten by the new data.

TREEDIST

Use the TREEDIST command to request output of a matrix of tree-to-tree distances computed according to the symmetric-difference or "partition" metric, or one of two metrics based on agreement subtrees.

Syntax

TREEDIST [*options*];

Menu equivalent

Trees > Tree-to-Tree Distances ...

Available options

- METRIC = SYMDIFF | AGD1 | AGREEMENT
- FROMTREE = *integer-value*
- FD = <u>YES</u> | NO
- SHOWALL = <u>YES</u> | NO
- FILE = *export-file-name*
- *REPLACE = YES | <u>NO</u>
- *APPEND = YES | <u>NO</u>

*Option is nonpersistent

Descriptions of options

METRIC = <u>SYMDIFF</u> | AGD1 | AGREEMENT

Use METRIC to specific the metric used to calculate tree-to-tree

distances. By default, the symmetric-difference metric (SYMDIFF) (Penny and Hendy, 1985) is used; however, you may also choose between two agreement-subtree metrics d (AGREEMENT) and d1 (AGD1) (Goddard et al. 1994).

FROMTREE = *integer-value*

If FROMTREE is specified, then the tree given by the *integer-value* will be compared to all other trees retained in memory. The default (FROMTREE = 0) specifies that all pairwise tree comparison be made.

FD = YES | NO

By default, the frequency distribution of tree-to-tree distances is output. Use FD=NO to suppress this output, or FD to reverse the effect of a previous FD=NO specification.

SHOWALL = YES | NO

By default, a matrix of the tree-to-tree distances for all comparisons performed is output. Use SHOWALL=NO to suppress the output, or SHOWALL to reverse the effect of a previous SHOWALL=NO specification.

FILE = *export-file-name*

Use the FILE option to save the tree-to-tree distances (as specified under the METRIC option) to a file.

REPLACE = YES | NO
APPEND = YES | NO

If FILE = *export-file-name* option is specified and the *export-file-name* already exists, you will be prompted for confirmation that the existing file should be replaced. Explicit specification of REPLACE = YES suppresses this prompt and the existing file will be quietly overwritten by the new data. Alternatively, you may specify APPEND = YES, in which case a new TREES block will be concatenated to the end of an existing file.

TREEINFO

Use the TREEINFO command to obtain information on the status of trees currently in memory. No options are available.

Syntax

TREEINFO ;

Menu equivalent

Trees > Tree Info > Brief Status

TREEWTS

Use the TREEWTS command to show current tree weights or specify ne tree weights.

Syntax

TREEWTS {SHOW|*list-of-tree-weights*};

Menu equivalent

Trees > Tree Info > Tree Weights

TSTATUS

Use the TSTATUS command to obtain information on which taxa, if any, are deleted and which taxa have been assigned to the outgroup.

Syntax

TSTATUS [*options*];

Menu equivalent

Data > Show Taxon Status ...

Available options

- FULL = <u>NO</u>|YES
- DELETED = <u>SHOW</u>|HIDE

Description of options

FULL = <u>NO</u>|YES

By default (FULL = NO) PAUP* briefly summarizes the number of taxa, in the data matrix, deleted, and assigned to the outgroup. If FULL = YES one line per taxon is given.

DELETED = <u>SHOW</u>|HIDE

Unless DELETED = HIDE, TSTATUS will display information for all of the taxa included in the DATA (PAGE 18) or TAXA (PAGE 10) block.

TYPESET

The TYPESET command, used to define a type set, is ordinarily issued from within the ASSUMPTIONS block. However, you may also issue it from the command line or from within a PAUP block. See "Commands used in the ASSUMPTIONS Block" on page 19 for the description of this command.

UNDELETE

Use the UNDELETE command to restore previously deleted taxa (or delete taxa not in list).

Syntax

UNDELETE *taxon-list* [/[ONLY] [CLEARTREES] [*options*]];

Unless /ONLY is specified, taxa specified in the taxon list are simply removed from the set of currently deleted taxa. Specify /ONLY to delete, in addition, any taxon not explicitly specified in the list.

Menu equivalent

Data > Delete-Restore Taxa ...

Available options

- *PRUNE = YES | NO
- *CLEARTREES = YES | NO

Description of options

The following two options are ignored unless the "/Only" option is also requested, in which case the effect of the command may be a deletion of currently nondeleted taxa.

PRUNE = YES | NO

If PRUNE = YES is specified, newly deleted taxa will be removed ("pruned") from the trees currently in memory, which otherwise remain unmodified.

CLEARTREES = YES | NO

If CLEARTREES is specified, any trees currently in memory are simply deleted.

UPGMA

Use the UPGMA command to calculate a UPGMA tree.

Syntax

UPGMA [*options*];

Menu equivalent

Analysis > Neighbor Joining/UPGMA ...

Available options

- BRLENS = YES | <u>NO</u>
- BREAKTIES = <u>SYSTEMATIC</u> | RANDOM
- TIESEED = *integer-value*
- SHOWTREE = <u>YES</u> | NO
- TREEFILE = *tree-file-name*
- *REPLACE = YES | NO
- *APPEND = YES | NO

*Option is nonpersistent

Description of options

BRLENS = YES | <u>NO</u>

If BRLENS = YES is specified, tree descriptions will include branch lengths. BRLENS = NO reverses the effect of a previous BRLENS specification.

BREAKTIES = <u>SYSTEMATIC</u> | RANDOM

Unless BREAKTIES = RANDOM, ties encountered during tree construction are broken systematically according to the input order of the taxa.

TIESEED = *integer-value*

If BREAKTIES = RANDOM, PAUP* references the system clock to initialize the seed used to generate pseudorandom numbers for breaking ties found during tree construction. An explicit seed may be specified to override the system clock default. (See "INITSEEDS = 0 | 1" on page 116 for more details.)

SHOWTREE = <u>YES</u> | NO

If SHOWTREE = YES, the UPGMA tree is printed to the display buffer

TREEFILE = *tree-file-name*

If TREEFILE is specified, a description of the UPGMA tree is output to a file in the NEXUS tree format. The *tree-file-name* must follow the conventions described under *"Input/Output files" on page 9*

REPLACE = YES | <u>NO</u>
APPEND = YES | <u>NO</u>

If the specified file already exists, you will be prompted for confirmation that the existing file should be replaced. Explicit specification of REPLACE suppresses this prompt and the existing file will be quietly overwritten by the new data. Alternatively, you may specify APPEND, in which case a new TREES block will be concatenated to the end of an existing file.

USERTREE

Use the USERTREE command to input a single user-defined tree. Ordinarily, you should use a TREES block (see "Commands used in the TREES Block" on page 25) to input one or more user-defined trees. This command merely provides a mechanism for quickly specifying a user-defined tree from the command line, which may be useful in certain situations.

USERTYPE

The USERTYPE command, used to define a user-defined character type, is ordinarily issued from within the ASSUMPTIONS block. However, you may also issue it from the command line or from within a PAUP block. See "Commands used in the ASSUMPTIONS Block" on page 19 for the description of this command.

WEIGHTS

Use the WEIGHTS command to assign weights to one or more characters.

Syntax

> WEIGHTS *character-weight*:*character-list* [,*character-weight*:*character-list*]...;

Each *character-weight* must be a valid character weight. Each *character-list* consists of one or more character numbers, character names, or character-set names (see "Character lists" on page 8 for details). The characters specified by character-list are assigned the immediately preceding character-weight. Any number of character-weight:character-list pairs, separated by commas, may be specified.

Menu equivalent

Data > Set Character Weights ...

WTS

The WTS command is a synonym for the WEIGHTS command. For some reason, I have a hard time typing "weights."

WTSET

The WTSET command, used to define a weight set, is ordinarily issued from within the ASSUMPTIONS block. However, you may also issue it from the command line or from within a PAUP block. See "Commands used in the ASSUMPTIONS Block" on page 19 for the description of this command.

REFERENCES

Adams, E. N., III. 1972. Consensus techniques and the comparison of taxonomic trees. Systematic Zoology 21:390-397.

Adams, E. N., III. 1986. N-trees as nestings: complexity, similarity, and consensus. Journal of Classification 3:299-317.

Archie, J. W. 1989. Homoplasy excess ratios: new indices for measuring levels of homoplasy in phylogenetic systematics and a critique of the consistency index. Systematic Zoology 38:253-269.

Bremer, K. 1990. Combinable component consensus. Cladistics 6:369-372.

Camin, J. H. and R. R. Sokal. 1965. A method for deducing branching sequences in phylogeny. Evolution 19:311-326.

Cavalli-Sforza, L. L. and A. W. F. Edwards. 1967. Phylogenetic analysis: Models and estimation procedures. Evolution 32:550-570.

Constantinescu, M. and D. Sankoff. 1986. Tree enumeration modulo a consensus. Journal of Classification 3:349-356.

DeBry, R. W. and N. A. Slade. 1985. Cladistic analysis of restriction endonuclease cleavage maps within a maximum-likelihood framework. Systematic Zoology 34:21-34.

Donoghue, M. J. and W. P. Maddison. 1986. Polarity assessment in phylogenetic systematics: a response to Meacham. Taxon 35:534-545.

Faith, D. P. 1991. Cladistic permutation tests for monophyly and nonmonophyly. Systematic Zoology 40:366-375.

Farris, J. S. 1969. A successive approximations approach to character weighting. Systematic Zoology 18:374-385.

Farris, J. S. 1970. Methods for computing Wagner trees. Systematic Zoology 19:83-92.

Farris, J. S. 1972. Estimating phylogenetic trees from distance matrices. American Naturalist 106:645-668.

Farris, J. S. 1977. Phylogenetic analysis under Dollo's Law. Systematic Zoology 26:77-88.

Farris, J. S. 1982. Outgroups and parsimony. Systematic Zoology 31:328-334.

Farris, J. S. 1988. Hennig86, version 1.5. Distributed by the author, Port Jefferson Station, N. Y.

Farris, J. S. 1989a. The retention index and homoplasy excess. Systematic Zool-

ogy 38:406-407.

Farris, J. S. 1989b. The retention index and the rescaled consistency index. Cladistics 5:417-419.

Farris, J. S., M. Källersjö, A. G. Kluge, and C. Bult. 1995. Testing significance of incongruence. Cladistics 10:315-319.

Felsenstein, J. 1978a. Cases in which parsimony and compatibility methods will be positively misleading. Systematic Zoology 27:401-410.

Felsenstein, J. 1978b. The number of evolutionary trees. Systematic Zoology 27:27-33.

Felsenstein, J. 1981. Evolutionary trees from DNA sequences: A maximum likelihood approach. J. Mol. Evol. 17:368-376.

Felsenstein, J. 1984. Distance methods for inferring phylogenies: A justification. Evolution 38:16-24.

Felsenstein, J. 1984b. The statistical approach to inferring phylogeny and what it tells us about parsimony and compatibility. Pages 169-191 in T. Duncan and T. F. Stuessy (ed.), Cladistics: Perspectives on the Reconstruction of Evolutionary History (Columbia University Press: New York).

Felsenstein, J. 1985. Confidence limits on phylogenies: An approach using the bootstrap. Evolution 39:783-791.

Felsenstein, J. 1991. PHYLIP (Phylogeny Inference Package), version 3.4. Distributed by the author, Univ. of Washington, Seattle, Washington.

Fishman, G. S. and L. R. Moore. 1982. A statistical evaluation of multiplicative congruential random number generators with modulus 231 - 1. Journal of the American Statistical Association 77:129-136.

Fitch, W. M. 1971. Toward defining the course of evolution: minimal change for a specific tree topology. Systematic Zoology 20:406-416.

Fitch, W. M. and E. Margoliash. 1967. Construction of phylogenetic trees. Science 155:279-284.

Funk, V. A. and D. R. Brooks. 1990. Phylogenetic Systematics as the Basis of Comparative Biology. (Smithsonian Institution Press: Washington, D. C.).

Goloboff, P. A. 1993. Estimating character weights during tree search. Cladistics 9:83-91.

Goddard, W., E. Kubicka, G. Kubicki, and F. R. McMorris. 1994. The agreement metric for labeled binary trees. Mathematical Bioscience 123:215-226.

Goldman, N., J. P. Anderson, and A. G. Rodrigo. 2000. Likelihood-based tests of topologies in phylogenetics. Systematic Biology 49:652-670.

Gould, R. 1988. Graph Theory. (Benjamin/Cummings: Menlo Park, California).

Harary, F. 1969. Graph Theory. (Addison-Wesley: Reading, Massachusetts).

Hasegawa, M., H. Kishino, and T. Yano. 1985. Dating the human-ape split by a molecular clock of mitochondrial DNA. Journal of Molecular Evolution 22: 160-174.

Hendy, M. D. and D. Penny. 1982. Branch and bound algorithms to determine minimal evolutionary trees. Mathematical Biosciences 59:277-290.

Hendy, M. D., M. A. Steel, D. Penny, and I. M. Henderson. 1988. Families of trees and consensus. Pages 355-362 in H. H. Bock (ed.), Classification and Related Methods of Data Analysis (Elsevier: Amsterdam).

Hennig, W. 1966. Phylogenetic Systematics. (University of Illinois Press: Urbana, Illinois).

Hillis, D. M. 1987. Molecular versus morphological approaches to systematics. Annual Review of Ecology and Systematics 18:23-42.

Hillis, D. M. 1991. Discriminating between phylogenetic signal and random noise in DNA sequences. Pages 278-294 in M. M. Miyamoto and J. Cracraft (ed.), Phylogenetic Analysis of DNA Sequences (Oxford University Press: New York, N. Y.).

Hillis, D. M. and J. P. Huelsenbeck. 1992. Signal, noise, and reliability in molecular phylogenetic analyses. Journal of Heredity 83:189-195.

Holmquist, R., M. M. Miyamoto, and M. Goodman. 1988. Analysis of higher-primate phylogeny from transversion differences in nuclear and mitochondrial DNA by Lake's methods of evolutionary parsimony and operator metrics. Molecular Biology and Evolution 5:217-236.

Huelsenbeck, J. P. 1991. Tree-length distribution skewness: an indicator of phylogenetic information. Systematic Zoology 40:257-270.

Jukes, T. H., and C. R. Cantor. 1969. Evolution of protein molecules. p. 21-132. In Mammalian protein metabolism. H. N. Munro (ed.). Academic Press, New York.

Källersjö, M., J. S. Farris, A. G. Kluge, and C. Bult. 1992. Skewness and permutation. Cladistics 8:275-287.

Kluge, A. G. and J. S. Farris. 1969. Quantitative phyletics and the evolution of anurans. Systematic Zoology 18:1-32.

Kimura, M. 1980. A simple method for estimating evolutionary rate of base substitution through comparative studies of nucleotide sequences. Journal of Molecular Evolution 16: 111-120.

Kimura, M. 1981. Estimation of evolutionary distances between homologous nucleotide sequenes. Proc. Natl. Acad. Sci. USA 78:454-458.

Kishino, H. and M. Hasegawa. 1989. Evaluation of the maximum likelihood estimate of the evolutionary tree topologies from DNA sequence data, and the

branching order in Hominoidea. Journal of Molecular Evolution 29:170-179.

Kishino, H., T. Miyata, M. Hasegawa. 1990. Maximum likelihood inference of protein phylogeny and the origin of chloroplasts. Journal of Molecular Evolution 30: 151-160.

Lake, J. A. 1987a. Determining evolutionary distances from highly diverged nucleic acid sequences: operator metrics. Journal of Molecular Evolution 26:59-73.

Lake, J. A. 1987b. A rate-independent technique for analysis of nucleic acid sequences: evolutionary parsimony. Journal of Molecular Evolution 4:167-191.

Lake, J. A. 1994. Reconstructing evolutionary trees from DNA and protein sequecnes: Paralinear distances. Proc. Natl. Acad. Sci. USA 91:1455-1459.

Lanave, C., G. Preparata, C. Saccone and G. Serio. 1984. A new method for calculating evolutionary substitution rates. J. Mol. Evol. 20:86-93.

Lockhart, P. J., M.A. Steel, M.D. Hendy and D. Penny. 1994. Recovering evolutionary trees under a more realistic model of sequence evolution. Mol. Biol. Evol. 11:605-612.

Lundberg, J. G. 1972. Wagner networks and ancestors. Systematic Zoology 21:398-413.

Maddison, D. R. 1991. The discovery and importance of multiple islands of most-parsimonious trees. Systematic Zoology 40:315-328.

Maddison, W. P. 1989. Reconstructing character evolution on polytomous cladograms. Cladistics 5:365-377.

Maddison, W. P. In press. Missing data versus missing characters in phylogenetic analysis. Systematic Biology

Maddison, W. P., M. J. Donoghue, and D. R. Maddison. 1984. Outgroup analysis and parsimony. Systematic Zoology 33:83-103.

Maddison, W. P. and D. R. Maddison. 1992. MacClade: Analysis of Phylogeny and Character Evolution, version 3.0. Sinauer Associates, Sunderland, Massachusetts.

Margush, T. and F. R. McMorris. 1981. Consensus n-trees. Bulletin of Mathematical Biology 43:239-244.

Meacham, C. A. 1984. The role of hypothesized direction of characters in the estimation of evolutionary history. Taxon 33:26-38.

Meacham, C. A. 1986. More about directed characters: a reply to Donoghue and Maddison. Taxon 35:538-540.

Miyamoto, M. M. 1985. Consensus cladograms and general classifications. Cla-

distics 1:186-189.

Nei, M. and W. H. Li. 1979. Mathematical model for studying genetic variation in terms of restriction endonucleases. Proc. Natl. Acad. Sci. USA 76:5269-5273.

Page, R. D. M. 1989. Comments on component-compatibility in historical biogeography. Cladistics 5:167-182.

Page, R. D. M. 1993. COMPONENT: Tree comparsion software for Microsoft Windows, version 2.0. Natural History Museum, London.

Penny, D. and M. D. Hendy. 1985. The use of tree comparison metrics. Systematic Zoology 34:75-82.

Rodriguez, R., J. L. Oliver, A. Marin and J. R. Medina. 1990. The general stochastic model of nucleotide substitution. J. Theor. Bio. 142:485-501.

Robinson, D. F. and L. R. Foulds. 1981. Comparison of phylogenetic trees. Mathematical Biosciences 53:131-147.

Rohlf, F. J. 1982. Consensus indices for comparing classifications. Mathematical Biosciences 59:131-144.

Sanderson, M. J. 1989. Confidence limits on phylogenies: the bootstrap revisited. Cladistics 5:113-129.

Sankoff, D. and R. J. Cedergren. 1983. Simultaneous comparison of three or more sequences related by a tree. Pages 253-263 in D. Sankoff and J. B. Kruskal (ed.), Time Warps, String Edits, and Macromolecules: the Theory and Practice of Sequence Comparison (Addison-Wesley: Reading, Mass.).

Sankoff, D., R. J. Cedergren, and W. McKay. 1982. A strategy for sequence phylogeny research. Nucleic Acids Research 10:421-431.

Sankoff, D. and P. Rousseau. 1975. Locating the vertices of a Steiner tree in an arbitrary metric space. Mathematical Programming 9:240-246.

Shimodaira, H. and M. Hasegawa. 1999. Multiple Comparisons of Log-Likelihoods with Applications to Phylogenetic Inference. Molecular Biology and Evolution 16:1114-1116.

Sokal, R. R. and F. J. Rohlf. 1981. Taxonomic congruence in the Leptopodomorpha re-examined. Systematic Zoology 30:309-325.

Steel, M. 1994. Recovering a tree from the Markov leaf colourations it generates under a Markov model. Appl. Math. Lett. 7:19-23 (also published as May 1993, Research Rep. 103, Mathematics Dept., University of Christchurch, NZ).

Swofford, D. L. 1991. When are phylogeny estimates from morphological and molecular data incongruent? Pages 295-333 in M. M. Miyamoto and J. Cracraft (ed.), Phylogenetic Analysis of DNA Sequences (Oxford University Press: New York, N. Y.).

Swofford, D. L. and S. H. Berlocher. 1987. Inferring evolutionary trees from gene frequency data under the principle of maximum parsimony. Systematic Zoology 36:293-325.

Swofford, D. L. and W. P. Maddison. 1987. Reconstructing ancestral character states under Wagner parsimony. Mathematical Biosciences 87:199-229.

Swofford, D. L. and W. P. Maddison. 1992. Parsimony, character-state reconstructions, and evolutionary inferences. Pages 186-223 in R. L. Mayden (ed.), Systematics, Historical Ecology, and North American Freshwater Fishes (Stanford University Press: Stanford).

Swofford, D. L. and G. J. Olsen. 1990. Phylogeny reconstruction. Pages 411-501 in D. M. Hillis and C. Moritz (ed.), Molecular Systematics (Sinauer Associates: Sunderland, Massachusetts).

Swofford, D. L., G. J. Olsen, P. J. Waddell, and D. M. Hillis. 1996. Phylogenetic inference. Pages 407-514 in D. M. Hillis, C. Moritz, and B. K. Mable (ed.), Molecular Systematics, 2nd edition. (Sinauer Associates: Sunderland, Massachusetts).

Templeton, A. R. 1983a. Convergent evolution and non-parametric inferences from restriction fragment and DNA sequence data. Pages 151-179 in B. Weir (ed.), Statistical Analysis of DNA Sequence Data (Marcel Dekker: New York).

Templeton, A. R. 1983b. Phylogenetic inference from restriction endonuclease cleavage site maps with particular reference to the evolution of humans and apes. Evolution 37:221-244.

Tamura, K. and M. Nei. 1993. Estimation of the number of nucleotide substitutions in the control region of mitochondrial DNA in humans and chimpanzees. Molecular Biology and Evolution. 10:512-526.

Upholt, W. B. 1977. Estimation of DNA sequence divergence from comparison of restriction endonuclease digests. Nucl. Acids Res. 4:1257-1265.

Wiley, E. O. 1981. Phylogenetics. The Theory and Practice of Phylogenetic Systematics. (Wiley and Sons: New York).

Wiley, E. O., D. Siegel-Causey, D. R. Brooks, and V. A. Funk. 1991. The Compleat Cladist. A Primer of Phylogenetic Procedures. (University of Kansas Museum of Natural History Special Publ. No. 19: Lawrence, Kansas).