KHns: Software for Two Tree Tests
Version 1.0

Edward Susko

Department of Mathematics and Statistics, Dalhousie University

Introduction

The main programs, `KHns` and `trees2df`, implement the methods described in Susko (2014); please cite this reference when using the software. The program `KHns` gives results for the KHns test, the conditional chi-square test and, if the degrees of freedom are at most 2, the p-value for the likelihood ratio test using its mixture of chi-squares limiting distribution.

The program `trees2df` calculates the degrees of freedom for a chi-square test based on the two Newick format trees in a file. It is useful in providing the degrees of freedom for a likelihood ratio test, when one wants to obtain the likelihoods for models not currently implemented in `KHns`.

Installation

The program `KHns` is compiled from C and Fortran 77 source code. To install the programs

1. Download and unpack the software:

   ```
   $ tar zxf khnsv1.0.tar.gz
   ```

   This will create a directory `khnsv1.0` that contains the source code.

2. Change directories to `khnsv1.0` and create the main program files `KHns` and `trees2df` with the `make` command.

   ```
   $ cd khnsv1.0
   $ make
   ```

The default installation assumes that the gcc and gfortran compilers are available. To use a different compiler change the variables `CC` and `F77` in `Makefile`.

For `KHns`, in cases where there are a large number of taxa or amino acid data is being considered, it will usually be necessary to use the observed information matrices or Monte Carlo approximations to the expected information matrices utilized in calculations. The Monte Carlo approximations, which are not necessary, assume that the program `seq-gen` (Rambaut and Grassly 1997) is available. This program can be downloaded from the aesthetically pleasing web site

`http://tree.bio.ed.ac.uk/software/seqgen/`

If you only want the trees2df program, you do not need to have a Fortran 77 compiler or seq-gen and can create the desired software with

```
$ make trees2df
```

in place of make in the steps above.

The KHns program utilizes the NNLS routine of Lawson and Hanson (1974) and the L-BFGS-B routines of Byrd et al. (1995) and Morales and Nocedal (2011). The L-BFGS-B routines further uses the LINPACK and BLAS linear algebra routines. These are included in the source code. If you wish to use BLAS libraries optimized for your platform, remove blas.o in Makefile and add a link to the BLAS libraries.

trees2df: Obtaining degrees of freedom

The program trees2df obtains the degrees of freedom for a conditional chi-square test based on the likelihood ratio statistic. It can be run at the command line with

```
$ trees2df [-t 1.0e-6] -n number_of_taxa < intree
```

Here number_of_taxa is the number of taxa considered and must be present. The file intree should contain the Newick format trees, with estimated edge-lengths, that are to be tested. As an example consider the file intree with the following text

```
((2:0.1,3:0.1):0.1,(0:0.1,1:0.1):0.1,(4:0.1,5:0.1):0.1);
((2:0.1,5:0.1):0.1,(0:0.1,4:0.1):0.1,(1:0.1,3:0.1):0.1);
```

The example is solely for illustrative purposes; it is unlikely every edge-length would be estimated as 0.1. The input trees should conform to the Newick standard. A discussion of this standard as implemented in PHYLIP is given at

http://evolution.genetics.washington.edu/phylip/newicktree.html

and a more formal description is available at

http://evolution.genetics.washington.edu/phylip/newick_doc.html

Allowable features of the Newick standard that will likely create difficulties are:

1. Quoted labels.
2. Nested use of the characters '[' and/or ']' in comments. The characters '[' and ']' can only be used to delimit comments and cannot be used within comments.
3. Long leaf labels. A limit of 10 non-null characters is allowed for leaf names.
4. Underscores are not converted to blanks.

Returning to the running example, `trees2df` gives

```
$ trees2df -n 6 < intree
3 3
((4:0.10000,5:0.10000):0.00000,(0:0.10000,1:0.10000):0.00000,(2:0.10000,3:0.10000):0.00000);
```

The first number output (3 in this case) indicates the number of edges that needed to be set to 0 to make the two trees equivalent. Among those edges that needed to be set to 0, the second number indicates the number of these that are estimated as positive in the first tree in `intree`. In the example, all edge-lengths were positive so this number is 3 too. The output tree is the first tree in `intree` with edge lengths set to 0 so that it is a special case of the second tree.

Assume now that in `intree` the first tree was altered to

```
((2:0.1,3:0.1):0.000001,(0:0.1,1:0.1):0.0,(4:0.1,5:0.1):0.1);
```

The resulting output is

```
$ ../trees2df -n 6 -t 1.0e-5 < intree
3 1
((4:0.10000,5:0.10000):0.00000,(0:0.10000,1:0.10000):0.00000,(2:0.10000,3:0.10000):0.00000);
```

The output is the same as before except that it has been indicated that, among those edges that needed to be set to 0 to make the two trees equivalent, only one of the edge-lengths was estimated to be at least as large as 1.0e-5. This tolerance for what constitutes a 'zero' edge-length was set with the `-t 1.0e-5` option and is set to 1.0e-6 by default. If instead it had been set to $1.0e-7$, the first line of output would change to `3 2`, since the edge length 0.000001 is larger than $1.0e-7$.

The second value output, call it ν, is the degrees of freedom that one should use for a conditional chi-square test. If the observed log likelihood ratio statistic is $2dLnL$, the p-value is calculated as the probability that a chi-square random variable with ν degrees of freedom is larger than $2dLnL$. The log likelihood ratio statistic is $2dLnL = 2[l(T_2)-l(T_{2z})]$ where $l(T_2)$ is the log likelihood for Tree 2 with optimal edge lengths and $l(T_{2z})$ is the log likelihood for Tree 3 with optimal edge lengths, but with edge-lengths constrained to 0 to make Tree 1 and 2 equivalent. Since most software does not allow edge-lengths to be constrained to 0, an alternative is to use $2[l(T_2) - l(T_1)]$, where $l(T_1)$ is the log likelihood for Tree 2 with optimal edge lengths. This will give a similar but more conservative result: the p-value will be larger than it would have been using $l(T_{2z})$.

KHns: One-sided tests for two trees

The program `KHns` gives the results of some tests for significant evidence for Tree 1 (the first tree in a file with two Newick format trees), tested against Tree 2 (the second tree). It can be run at the command line with the command

```
KHns controlfile
```

Input

All input to the routine is through a main control file, `controlfile`. The control file is similar in format to the control files used by the programs `baseml` and `codeml` in the PAML package (Yang 1997, 2007). For instance, the `model` variable specifies the substitution model and gives a subset of the models available in PAML, with the same numbering scheme. As an example, consider the following file:

```
model = 4                             * KH model
seqfile = example-six-taxa-02.infile  * sequence data file
kappa = 2                             * kappa parameter
treefile = example-six-taxa-02.tree   * input trees 1 and then 2
outtreefile = outtreefile             * output treefile pseudo-datasets for KHns test
```

As with PAML control files, blank lines are allowed and all text following a '*' till the end of a line is treated as a comment. The word on the left of an equal sign gives a control variable and the word on the right gives the value of that variable. Spaces are required on both side of an equal sign. The order of variables is unimportant. The control variables are as follows. All variables not indicated as optional are required.

`seqfile`: The name of the file containing the sequence data. The file should conform to the requirements of the PHYLIP package (Felsenstein, 1989, 2004). Sequence names should be 10 characters long and padded by blanks. The names should match the names used in the input treefile. Input can be either interleaved or sequential with one caveat: The lines 2 through $m + 2$, where m is the number of taxa, must contain the name of taxa followed by sequence data. For instance the start of a sequence file might be

```
6 3414
Homsa      ANLLLLIVPI LI...
Phovi      INIISLIIPI LL...
...
```

but not

```
6 3414
Homsa
ANLLLLIVPI LI...
Phovi
INIISLIIPI LL...
...
```

which would be allowed under the sequential format by PHYLIP.

Additional information is available at

`http://evolution.genetics.washington.edu/phylip/doc/sequence.html`

`treefile`: The name of the treefile containing the two input trees. The format is the same as for `trees2df`. The p-values reported are for the alternative hypothesis that the first tree in the treefile is the correct tree.

`outtreefile`: An optional file name for the output trees. If present, the trees output to the file are the estimated Trees 1 and then 2 with optimal edge-lengths followed by the estimated Tree 1 but with edges constrained to be 0 to make it a special case of Tree 2.

`nchar`: An optional integer indicating that the model was for nucleotide data (`nchar` = 4) or amino acid data (`nchar` = 20). The default value is 4.

`model`: An integer code for the substitution model. For nucleotide data (`nchar` = 4), the models currently implemented are

`model`	Model
0	JC
2	F81
3	F84
4	HKY
7	GTR

and for amino acid data (`nchar` = 20) the models currently implemented are

`model`	Model
0	Poisson
1	Proportional
2	Empirical
3	Empirical+F
8	REVaa

The documentation for the PAML package gives a good description of the models listed and can fit all of them.

The GTR and REVaa models refer to the most general time-reversible models in the nucleotide and amino acid case, respectively. The Poisson and Proportional models are the analogues of the JC and F81 models for amino acid data. The Poisson and Proportional models have substitution probabilities

$$P_{ij}(t) = \begin{cases} \pi_j + (1 - \pi_j)\exp[-\mu t] & \text{if } i = j \\ \pi_j - \pi_j \exp[-\mu t] & \text{otherwise} \end{cases}$$

where $\mu = [\sum \pi_i(1-\pi_i)]^{-1}$ and π_j gives the stationary of the jth amino acid. In the Poisson model, the frequencies are all 1/20

When `model` = 2 or 3 an empirical model is fit. The model is specified by the variable `aaRatefile`. When `model` = 2, the stationary frequencies are the stationary frequencies of the specified empirical model.

`Qfile:` Only required for the general time reversible model, GTR or REVaa (`model` = 7, `nchar` = 4 or `model` = 8, `nchar` = 20). The name of a file containing the entries of the rate matrix separated by blanks.

`aaRatefile`: Only required for empirical amino acid models (`model` = 2 or 3 and `nchar` = 20). The name of the empirical model to fit. The models currently implemented are

`dayhoff.dat`	Dayhoff or PAM	Dayhoff et al. (1978)
`jones.dat`	JTT	Jones et al. (1992)
`wag.dat`	WAG	Whelan and Goldman (2001)
`mtREV24.dat`	mtREV	Adachi and Hasegawa (1996)
`lg.dat`	LG	Le and Gascuel (2008)

The naming scheme was chosen to be consistent with PAML. However, `aaRatefile` is not actually the name of file, rather it identifies a model.

`kappa` or `ttratio:` One of these is required for the F84 and HKY models (`model` = 3 or 4 and `nchar` = 4). A real number giving the κ parameter for the model. The F84 model has a rate matrix proportional to

$$\begin{bmatrix} \cdot & \pi_C & (1+\kappa/\pi_R)\pi_G & \pi_T \\ \pi_A & \cdot & \pi_G & (1+\kappa/\pi_Y)\pi_T \\ (1+\kappa/\pi_R)\pi_A & \pi_C & \cdot & \pi_T \\ \pi_A & (1+\kappa/\pi_Y)\pi_C & \pi_G & \cdot \end{bmatrix}$$

where $\pi_R = \pi_A + \pi_G$ and $\pi_Y = \pi_C + \pi_T$. The HKY model has a rate matrix proportional to

$$\begin{bmatrix} \cdot & \pi_C & \kappa\pi_G & \pi_T \\ \pi_A & \cdot & \pi_G & \kappa\pi_T \\ \kappa\pi_A & \pi_C & \cdot & \pi_T \\ \pi_A & \kappa\pi_C & \pi_G & \cdot \end{bmatrix}$$

The transition-transversion ratio (`ttratio`) is related to the κ parameter in the F84 model through

$$R = \kappa \times \frac{\pi_A\pi_G/\pi_R + \pi_C\pi_T/\pi_Y}{\pi_R\pi_Y} + \frac{\pi_A\pi_G + \pi_C\pi_T}{\pi_R\pi_Y}$$

where $\pi_R = \pi_A + \pi_G$ and $\pi_Y = \pi_C + \pi_T$. For the HKY model the relationship is

$$R = \kappa \times \frac{\pi_A \pi_G + \pi_C \pi_T}{\pi_R \pi_Y}$$

`alpha`: Only required if a gamma rates-across-sites model (Yang 1994) is desired. A real value giving the shape parameter of the gamma distribution. This is used as an initial value for estimation of α.

`ncatG`: Only used if a discrete gamma rates-across-sites model was fit. Optionally, an integer giving the number of categories to use in the discrete approximation. The default is 4. The discrete gamma approximation used is the same as the default of PAML 4.2; the representative rate is the conditional mean for the class.

`tinfo`: Optionally one of 0, 1 or 2. The default is 0.

The KHns and mixture of chi-square tests use expected information matrix calculations that can require substantial calculation when either the number of taxa is large or amino acid data is considered. Expected information matrices can be approximated through simulation. The value `tinfo`=0 indicates that calculation should be exact and the value `tinfo`=1 indicates that simulation should be used for calculation. The value `tinfo`=1 assumes that the program `seq-gen` is available.

An alternative is to use the observed information matrix. Where the expected information matrix is the expected value of the second derivative matrix of the log likelihood multiplied by -1, the observed information is the actual second derivative matrix of the log likelihood multiplied by -1. Theory indicates that, assuming the generating model is correct, the observed and expected information matrix should be approximately the same with large numbers of sites and the use of either will yield first-order correct adjusted BP values. To use the observed information indicate option `tinfo=2`.

`tol`: The tolerance for declaring an edge-length to be estimated as 0. The default is 1.0e-6. If an estimated edge-length in tree 1 is less than `tol` it will be treated as 0. The routines for estimation do not allow edge-lengths less than 1.0e-11 and, due to numerical imprecision, might not give an edge-lengths as small as this when in fact an estimated edge length should be 0.

Output

The output (to the screen or stdout) gives the results of the tests. The results output depend on the number of edge-lengths that need to be set to 0 to make the two trees equivalent. In all cases, the result of the KHns test are output. If only one or two edge-length needs to be set to zero, the result of a mixture test is given. In the case that two or more edge-lengths are set to 0, the result of conditional chi-square test is given as well.

As an example, consider the control file `controlfile`:

```
model = 4                            * KH model
seqfile = example-six-taxa-02.infile * sequence data file
alpha = -1                           * no RAS estimation
kappa = 2                            * kappa parameter
treefile = example-six-taxa-02.tree  * input trees 1 and then 2
outtreefile = outtreefile            * output treefile
```

The output is

```
$ KHns controlfile
Log likelihoods for Trees 1,2 and the collapsed tree
-3968.888929 -3969.221180 -3969.221180
Number of zero-length edges in the collapsed tree = 2
Number of estimated non-zero edges = 1
Conditional chi-square p-value = 0.414975
Mixture p-value = 0.371117
KHns p-value = 0.265200
```

indicating that the log likelihood for the Trees 1 and 2 with optimal edge-lengths are -3968.89 and -3969.22. When edge-lengths in Tree 1 are constrained to be 0 to make it equivalent to Tree 2, the log likelihood is also -3969.22. This is because of estimated edge-lengths for tree 2 that were effectively zero, as can be seen by considering the estimated second tree in the output tree file `outtreefile` below. Two edge-lengths needed to be set to zero in Tree 1 to make it equivalent to Tree 2 and one of these was estimated to be positive. Thus the degrees of freedom for the conditional chi-square test is 1 and its p-value turned out to be 0.41. Using the same test statistic but a mixture of chi-squares to calculate the p-value of 0.37. The p-value with a mixture of chi-squares is more appropriate in this case but not available when the number of collapsed edges is more than 2. The KHns p-value turned out to be 0.27 in this case.

The `outtreefile` indicates the output tree file and gives, respectively, the estimated Trees 1 and then 2 with optimal edge-lengths and finally the estimated Tree 1 but with collapsed edges constrained to be 0.

```
((4:0.08306,5:0.09143):0.10487,(0:0.12458,1:0.09841):0.00000,(2:0.09420,3:0.08315):0.00166);
((1:0.09874,3:0.08438):0.00000,(0:0.12478,2:0.09543):0.00000,(4:0.08306,5:0.09143):0.10520);
((4:0.08306,5:0.09143):0.10519,(0:0.12478,1:0.09874):0.00000,(2:0.09543,3:0.08438):0.00000);
```

References

Byrd, R.H., Lu, P., Nocedal, J. and Zhu, C. (1995). A limited memory algorithm for bound constrained optimization. SIAM J. Sci. Comput. 16, 5, 1190-1208.

Lawson, C.L. and R.J. Hanson. (1974). Solving least squares problems. Prentice-Hall, NJ.

Morales, J.L. and Nocedal, J. (2011). Remark on “Algorithm 778: L-BFGS-B: Fortran Subroutines for Large-Scaled Bound Constrained Optimization” ACM Trans. Math. Soft. 38, No. 1. Article 7.

Rambaut, A. and Grassly, N. C. (1997). Seq-Gen: An application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees. Comput. Appl. Biosci. 13: 235-238.

Susko, E. (2014). Tests for Two Trees using Likelihood Methods. Mol. Biol. Evol. 31:1029–1039.

Yang, Z. (2007). PAML 4: a program for phylogenetic analysis by maximum likelihood. Mol. Biol. Evol. 24:1586–1591.

Yang, Z. (1997). PAML: a program for phylogenetic analysis by maximum likelihood. Comput. Appl. Biosci. 13:555–556.