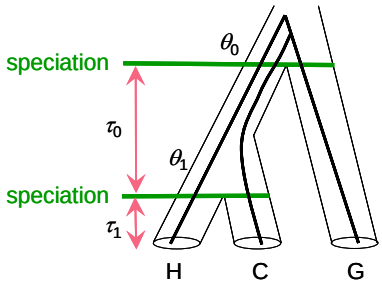
Ne3sML ver. 1.1

Introduction



The C program Ne3sML.c and the Mathematica packages (Ne3slnL.m or Ne3slnL.Rates.m), working together, implement the likelihood method of Yang (2002) for estimating the ancestral population sizes and species divergence times. The method works for three species, with one sequence from each species. The species tree is assumed known. The four parameters involved in the model are θ_0, θ_1, τ_0, τ_1. See figure. The data are counts of site patterns at multiple loci for three species. It is essential that you read Yang (2002) and use the included files to duplicate the calculations in that paper.

You must have Mathematica v3 or later installed to run the program. Windows executables are included. The source files should be compilable for MACs and UNIX as well, if you have a C compiler that is compatible with the Mathlink library. However, I have never tested the program on those systems. If you want to run this program on MACs or UNIX, you should look at the notes below in the section on Compiling programs. If you do not have Mathematica installed, you will get an error message "Unable to Locate Component", saying something like ML32I2.dll was not found.

The C program reads in the data and then runs an iteration routine to optimize the log likelihood function. If there are L loci in the data, each calculation of the log likelihood requires calculation of $2L$ two-dimensional integrals, which is achieved numerically by calling Mathematica (using the Mathematica program Ne3slnL.m). I wasted several months of time to learn how to link the C program with Mathematica. Let's hope you are lucky and can achieve it sooner.

How to Run the Program

Unpack the zip file into a folder, say, Ne3sML. Open a Command Prompt window and cd to that folder. dir to see the list of files. You run the program by typing (case-insensitive):

```
Ne3sML -linkmode launch
```

Then you will see a message "Please locate and select MathKernel" and a window will pop up. You have to walk around your folder tree to find MathKernel.exe (typically on drive C:\Program Files\Wolfram Research\Mathematica\3.0\MathKernel.exe). Click to highlight the file and then click on Open.

Next the program calls Mathematica to calculate 2.2 + 3.3, and if you see the following

```
2.2000 + 3.3000 = 5.5000
```

Things look good. The program then reads the data and asks you to input four values as starting values for the ML iteration. Type, say, 0.5 0.5 0.5 0.5 and hit Enter.

The parameters are in the order θ_0, θ_1, τ_0, τ_1, as defined in Yang (2002). In the program they are multiplied by 100 for numerical stability. So you have to divide the final estimates by 100 afterwards.

The iteration uses the same routine as baseml or codeml, and the iteration process is recorded in the file rub. After the iteration, the lnL and MLEs are printed on the screen. For the test data set, the results are as follows (from my new notebook with an intel centrino at 1.6GHz).

```
lnL  = -3099.411263
MLEs:    0.30572   0.09900   0.10894   0.51938

Time used: 00:12:32
```

The program also automatically calculates the SEs of parameters and the variance-covariance matrix by approximating the curvature of the log likelihood surface by the difference method. However, this calculation may not be reliable. For the example data set, I was unable to get reliable SEs, as the results from different runs are not similar enough for them to be usable.

Data File Format

The input data file name is Ne3sML.dat. The filename is hard coded in the C program, but you can prepare your data and then save the file using this name. You can print out the copy included in the package. The first number, 53, is the number of loci. Then for each locus, there are five counts of sites; see Table 1 of Yang (2002).

I used baseml to generate the site-pattern counts from the original sequence alignments. See Section below.

Compiling the Program

If you have MS VC++6 and want to compile the program yourself, you will have to copy two files, mathlink.h and ml32i1m.lib, from your Mathematica Installation into the Ne3sML\src\ folder. To find those two files, you go down the folder tree (based on my 3.0 installation): C: - Program Files - Wolfram Research - Mathematica - 3.0 - AddOns - Mathlink - DeveloperKits - Windows - CompilerAdditions - mldev32. Then look at the include and lib folders for the two files.

Then use the following command to compile, or you can make up a Makefile yourself. Afterwards move Ne3sML.exe from Ne3sML\src to Ne3sML\ where the data file resides.

```
cl -ML -O2 -W3 Ne3sML.c minsub2.c -link ml32i1m.lib
del *.obj
move Ne3sML.exe ..
```

Accounting for variable rates among loci

This section is added during the upgrade from v1.0 to v1.1 The update adds the option of specifying relative rates for loci, that is, to duplicate the results under "Variable rates among sites" in table 2 of Yang (2002). The following command should duplicate those results (wait for ~1 hour for the computation to finish).

```
Ne3sML -linkmode launch Ne3slnL.rates.m
```

The last command-line argument, Ne3slnL.rates.m, is the Mathematica package to be used in the calculation. The default is Ne3slnL.m, which assumes one rate for all loci.

To analyze your own data, open the file Ne3slnL.rates.m in a text editor and edit the following lines. The numbers below are the relative rates in table 1 of Yang (2002). Please replace those rates using the relative rates for your loci but follow the format exactly. Be extremely careful when you edit those

lines. The spaces do not matter, but please do not remove or add braces {} or commas by mistake. Note that the numbers must be separated by commas. You should prepare as many relative rates as the number of loci in your data. Note that the file is a plain text file.

```
rates={1.01,2.236,0.696,0.959,0.996,1.476,0.858,1.138,0.939,0.767,1.071,0.675,
   0.993,1.01,0.875,0.773,1.006,0.838,0.854,0.875,0.821,0.554,0.892,1.084,
   1.175,0.983,0.763,0.936,0.963,1.034,1.635,1.094,0.827,0.625,1.402,0.591,
   0.868,1.351,1.256,0.571,0.811,1.51,1.321,1.077,1.01,1.128,0.878,0.74,
   0.942,1.756,0.966,0.733,0.659};
```

To reiterate, note two things when you want to incorporate rates for loci. First, you should prepare the rates yourself and input the rates in the file Ne3slnL.rates.m by editing the lines copied above. Second you use the following command to run the program.

```
Ne3sML -linkmode launch Ne3slnL.rates.m
```

Other things are the same as before.

Collecting Site-Pattern Counts from Sequences

The Seqs2Counts program in the ChenLi3s\ folder reads sequences and collects site-pattern counts into a file named rst, which you rename as Ne3sML.dat to use with Ne3sML. Seqs2Counts is baseml, renamed after I added a few lines of code to print out the counts into rst. Look at ChenLiData3s.nuc, which has the sequence alignments for the 53 loci, one after another. Suppose the species tree is ((species1, species2), species3). Then the order of sequences must be species1, species2, species3 at each and every locus. In the example, the order of sequences is H, C, G at every locus, with the species tree ((H,C),G).

Seqs2Counts is controled by baseml.ctl, which you can look at. You can change seqfile, outfile, ndata. You should not change the other variables (model, RateAncestor, cleandata). Note that cleandata = 1, which means that sites with alignment gaps and undetermined nucleotides are all deleted. You can look at mlb, rst, and ignore the other files generated in the run. Run the program by

```
Seqs2Counts
```

History

version 1.0: 28 June 2003
version 1.1, 11 February 2005: added back variable rates among loci (by including the file Ne3slnL.Rates.m).

Reference

Yang Z (2002) Likelihood and Bayes estimation of ancestral population sizes in Hominoids using data from multiple loci. Genetics 162:1811-1823.