NSGTR-BH

by

Liwen Zou

Bioinformatics Research Center

Department of Genetics, North Carolina State University

October. 2011

Declaimer

We implemented a software named NSGTR-BH for estimating the estimates of permutations and nonstationary edge lengths proposed in Zou et al. (2011b). This is a free software. We have done our best to design, implement and test. However there is no guarantee that it will deliver everything right. **It is your own risk to use it.** Please read this manual carefully before using it.

Introduction

In Zou et al. (2011a), we have shown that the identifiability issue exists in the BH estimates. Non-identified estimates of a BH model lead to difficulties to interpret the frequencies of character bases at internal nodes and the edge lengths along all edges. In Zou et al. (2011b), we proposed non-stationary GTR models fitting in the estimates of the BH model along edges. By fitting non-stationary GTR models along edges, we developed an ad hoc algorithm implemented in this application, which estimates the permutations of estimates of the BH model iteratively. We also proposed a simplified formula of (2.3) in Minin and Suchard (2008) that can be used to estimate edge lengths corresponding to the joint probability estimates of any non-stationary models along edges. We have implemented the edge length calculation formula, equation (5) in Zou et al. (2011b), in this application

When doing estimation, without knowing the evolutionary directions, both of two evolutionary directions along an edge are considered. For each of two evolutionary directions, non-stationary GTR models are fit for

all possible permutations. Each of edges which connect two internal nodes has 576 possible permutations; each of edges which connect an internal node and a terminal node has 24 possible permutations. In our approach, we fit non-stationary GTR models for all of 576 permutations of internal edges and 24 permutations for edges connecting the internal and terminal nodes. After fitting non-stationary GTR models, our approach performs an ad hoc algorithm and iteratively find the estimated permutations by minimizing the sums of squares of differences between estimates of joint probability matrices of a BH model and estimates of using fitted non-stationary GTR models as shown in (3) in Zou et al. (2011b). Taking estimates of permutations, our application estimates edge lengths which can be interpreted as the expected number of substitutions.

When estimating permutations, our application only takes the input file which has the format of output file from the implementation of Jayaswal et al. (2005) in the following link http://www.la-press.com/estimation-of-phylogeny-using-a-general-markov-model-article-a186. Our application was developed using Java language. The algebra calculations are done by calling functions in the Lapack library through JNI.

Installing the NSGTR-BH

Computing environment

The application was tested on Linux OS Systems CentOS version 5 and Ubuntu 11.1. Java jre and jdk, C/C++ and FORTRAN compilers are necessary for running and compiling the application and shared libraries. Java jdk that includes Java jre can be download from http://www.oracle.com/technetwork/java/javase/downloads/index.html. For C/C++ and FORTRAN compilers, GNU compilers can be download from http://gcc.gnu.org/; Intel compilers can be downloaded from http://software.intel.com/en-us/articles/non-commercial-software-download/.

Note:

Please refer to the License agreements when using both GNU and Intel compilers.

Where to download

The package is available at the following link.

www.mscs.dal.ca/~tsusko/

The package

In the package, it contains

- the Java source code of NSGTR-BH application is in "Java" folder;
- the interface functions of calling Lapack library are in "interfaceFunctions" folder;
- the Lapack functions:
  - the Lapack functions used in this application extracted from Lapack library are in "fortranFunctionLapack" folder;
  - we also provide the most current version of Lapack library that can be downloaded separately for your convenience. If you prefer downloading original Lapack package by yourself, you can get it from http://www.netlib.org/lapack/.
- the manual is in "manual" folder.

After downloading, unpack the package as the following.

- Change the working directory to where the NSGTR-BH stored. For instance, if NSGTR-BH is in the directory "/home/myName/", type the following command at prompt

  cd /home/myName/

- At prompt, type in the command "tar -vxf nsgtrBH.tar". The unpacked NSGTR-BH will be in the folder named as "nsgtrBH" inside working directory.
- If you download "lapack-3.3.1.tar", change the directory to where "lapack-3.3.1.tar" is saved and do the same thing as for "nsgtrBH.tar".

How to compile Java code

Before compiling, please make sure that Java jdk 4.0 or higher version has been installed on your machine.

- Change the working directory to the folder "Java". For instance, if "Java" folder has the path "/home/myName/NSGTR-BH/Java", type the following command to change the working directory.

  cd /home/myName/NSGTR-BH/Java

- At prompt, type the following command for compiling java code.

  javac *.java

Create a dynamic library for Lapack functions

- If they haven't been installed, install GNU C/C++ and FORTRAN compilers or INTEL C/C++ and FORTRAN compilers and GNU "make".
  Note: The "makefile"s provided by this package are based on GNU compiler. If using INTEL compilers, "LOADER" AND "LOADERC" have to be changed accordingly. Please refer to as the manual of INTEL compilers to make changes properly.
  - Compile the Lapack library
    * Change working directory to "lapack-3.3.1".
    * Type command "make" at prompt. "make" will automatically run "Makefile" and compile both Lapack and Blas libraries. When it is done, you should be able to get two shared libraries files liblapack.so and libblas.so in "lapack-3.3.1". If you download the Lapack library by yourself, please replace "Makefile" and "make.inc" files in your downloaded Lapack by files with the same names "Makefile" and "make.inc" in "lapack-3.3.1" folder of this package.
  - Compile functions extracted from Lapack library.

* Change working directory to “fortranFunctionLapack”.
* Type command “make” at prompt.
* When it is done, you should be able to get a shared library named “liblapack.so” in the directory of “interfaceFunctions”.

There is no necessary to compile both of them. You only need to choose one of above options and a shared library for algebra calculation functions.

- Compile interface functions.
  - Change working directory to “interfaceFunctions”.
  - In the Makefile, this default setting is to use the shared library compiled from extracted functions from the Lapack library. If you are using the liblapack.so by compiling the Lapack library, please comment out the default “LAPACKPATH = -L“../interfaceFunctions”” by putting a “#” sign in front of this term, and select the one “LAPACKPATH = -L“pathTolapack-3.3.1/”” by removing the “#”sign in front of this term and change “pathTolapack-3.3.1” to the path where “lapack-3.3.1” is saved.
  - Find out the paths for “jni.h” and other header files used by “jni.h”, and change “JNIPATH” accordingly.
  - Type command “make” at prompt.
  - When it is done, you should be able to obtain a library file “libcallingLapackInterface.so”.

Set up environment variables

Add the path of the directory which contains “libcallingLapackInterface.so” into the environment variable “LD_LIBRARY_PATH”. For instance, if using BASH shell, and if assuming that “libcallingLapackInterface.so” is in the folder of “/home/myHome/NSGTR-BH/interfaceFunctions”, add the following line into .bashrc file.

LD_LIBRARY_PATH=/home/myHome/NSGTR-BH/interfaceFunctions;$LD_LIBRARY_PATH

export LD_LIBRARY_PATH

Add the path of the "Java" directory into the environment variable "PATH". For instance, if using BASH shell, and if assuming that "Java" folder has the path "/home/myHome/NSGTR-BH/Java", add the following line into .bashrc file.

PATH=/home/myHome/NSGTR-BH/Java;$PATH

export PATH

Estimating permutations and edge lengths for the BH estimates

Input data

NSGTR-BH takes the estimate of the BH models and the tree file when estimating BH estimates as the input files. Using inputted data, NSGTR-BH estimates the permutations and edge lengths.

Notes:

We assume that you have run the implementation of Jayaswal et al. (2005) with the data you are interested and have had the estimates of the BH model handy. Among the result files of Jayaswal et al. (2005), the file named as "divMatOutput.txt" contains estimates of joint probability matrices along edges of a BH model in the ORDER corresponding to the tree topology in the tree file. PLEASE DO NOT EDIT THIS FILE.

Options for Estimates

NSGTR-BH provides several choices for estimates for your convenience. You may select the rooting positions from a list of available nodes along the topology, or you may want NSGTR-BH estimating a root for you. You can also choose the outputs by only selecting the joint probability matrices along edges corresponding to

estimates of permutations or the estimates of the NSGTR models along edges corresponding to the estimates of the permutations or both of them. A summary of the options is listed below.

Rooting position:

1 Best estimated root;

Selecting this option, NSGTR-BH will estimate the root position. The root position selected by NSGTR-BH has the minimum SS over all possible rooting positions.

2 Selecting root position/positions from a list of candidate nodes.

This choice allows users to select multiple rooting positions from all internal and terminal nodes in the tree file. Then NSGTR-BH will estimate permutations, edge lengths and parameters of the NSGTR models for all of selected rooting positions.

Output selections:

1 Joint probability matrices along edges

This option writes the estimates of joint probability matrices along edges based on estimates of permutations and root positions either specified by users or the best root estimated by NSGTR-BH into a file named "bestFitJointProbabilityMatrices.txt".

2 NSGTR model parameters along edges

This option writes the estimates of the fitted NSGTR model parameters along edges corresponding to the estimates of permutations and root positions either specified by users or the best root estimated by NSGTR-BH into a file named "bestFitNSGTRParameters.txt".

3 Both.

Both of files in option 1 and 2 are generated.

It doesn't matter which output option has been chosen. The estimated edge lengths corresponding to selected rooting positions will be written into a Newick format file named "tree.txt".

Output Files

Based on your choices, NSGTR-BH may provide four output files.

- "tree.jpg": the visualization of the input tree. This visualization is useful when selecting rooting positions from a list of nodes. NSGTR-BH automatically generates this file as soon as it gets the input tree file.
- "tree.txt": the estimated edge lengths in the Newick format. In this file, it provides a list of tree presentations in the Newick format based on the choices of rooting positions. Each rooting position has an entry in this file.
- "bestFitJointProbabilityMatrices.txt": the estimates of joint probability matrices along edges based on the rooting position. NSGTR-BH provides a set of joint probability matrices for all 2*m - 3 edges for each rooting position, where m is the number of taxa in the data. The outputs of joint probability matrices for each rooting position are separated by the title of the rooting position.
- "bestFitNSGTRParameters.txt": the estimates of parameters of nonstationary GTR models along edges. For each rooting position, 2*m - 3 nonstationary GTR models are reported. For each of nonstationary GTR model, NSGTR-BH gives the estimates of instantaneous rates and stationary frequencies. The outputs of estimates of parameters of nonstationary GTR models for each rooting position are separated by the title of the rooting position.

Running NSGTR-BH

Note:

- **In the "example" directory, it contains the samples of joint probability matrices and a tree file. You may want to run NSGTR-BH with those files before running your own data.**
- **Please make sure that all environment variables have been set up properly before running NSGTR-BH. Please refer to as the section of "Set up environment variables" for the details.**

NSGTR-BH designs the interface as close as to the implementation of Jayaswal et al. (2005) for your convenience. Here is an example about how to use NSGTR-BH.

Assume that the data files are in the directory "/tmp/NSGTR-BH". The input joint probability matrices are in the file "divMatOutput.txt"; the input tree is in the file "inputTree.txt". The screen snapshot in Figure 1 shows the steps of running NSGTR-BH and it should be easy to following. When running NSGTR-BH, it may be useful if paying a little more attention to the following points.

- NSGTR-BH only takes one set of the BH estimates and one tree in one run. It won't handle multiple trees and sets of the BH estimates in a run.
- Please give the full path of input files as shown in Figure 1.
- Again, **Please do not edit the file "divMatOutput.txt".**

Notes:

This manual was written for the user who has some knowledge about the basic Linux commands. If you need more help, please feel free to contact the author by email at zoul@mathstat.dal.ca. We appreciate for any feedback.

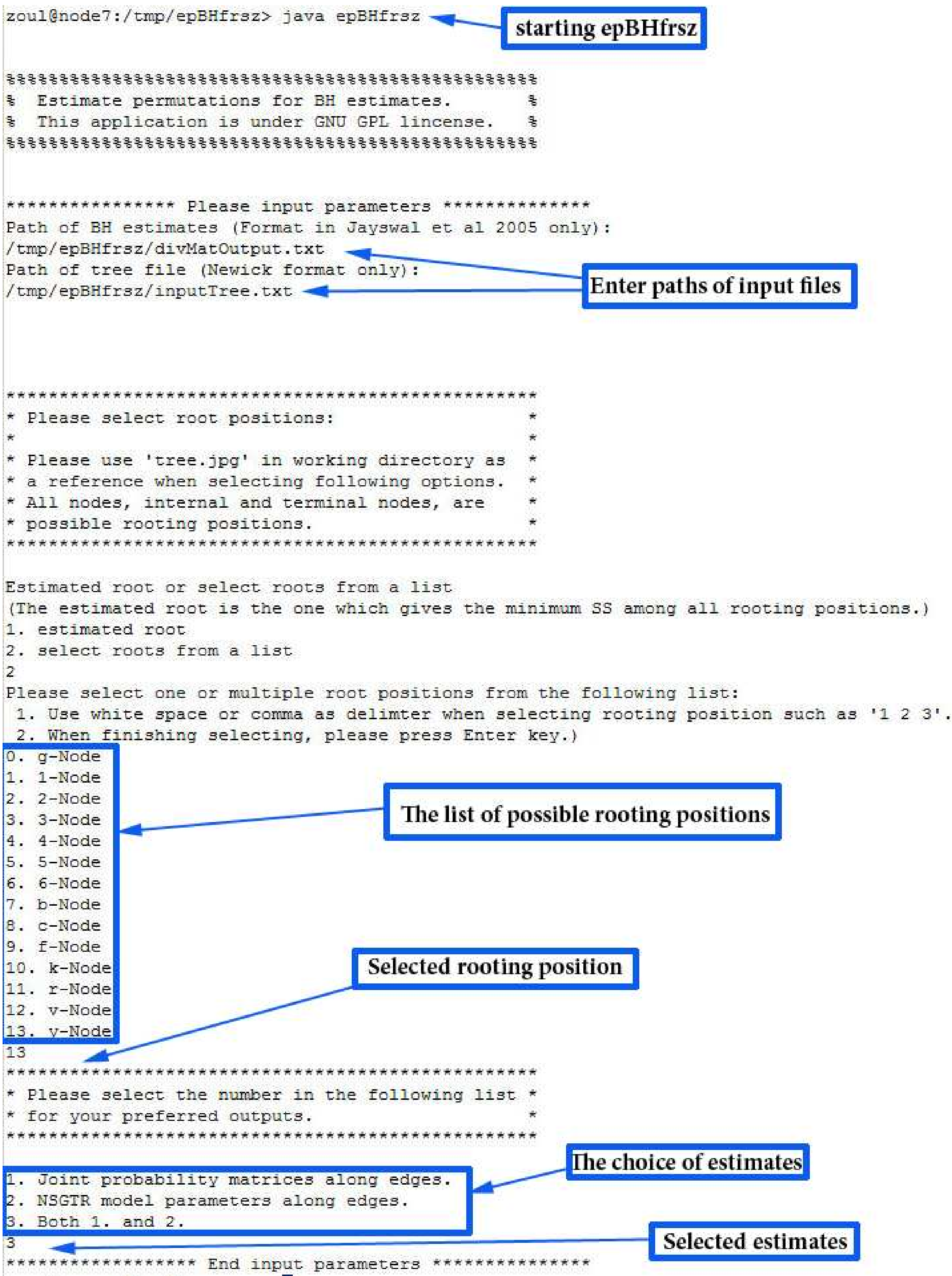

Figure 1: Screen Snapshot of an example of running NSGTR-BH. The input joint probability file has the path "/tmp/NSGTR-BH/divMatOutput.txt"; the input tree file has the path "/tmp/NSGTR-BH/inputTree.txt". The rooting position is selected as node y which is indixed as "13" in the list of candidates. The selection of estimates is to estimate both joint probability matrices and the parameters of the NSGTR models.

References

Jayaswal, V., L. Jermiin, and J. Robinson. 2005. Estimation of phylogeny using a general Markov model. Evolutionary Bioinformatics Online 1:62–80.

Minin, V. N. and M. A. Suchard. 2008. Fast, accurate and simulation-free stochastic mapping. Philosophical transactions of the Royal Society of London.Series B, Biological sciences 363:3985–3995.

Zou, L., C. Field, E. Susko, and A. Roger. 2011a. The Barry and Hartigan general Markov model suffers from statistical non-identifiability. Systematic Biology doi:10.1093/sysbio/syr034 .

Zou, L., E. Susko, C. Field, and A. J. Roger. 2011b. Fitting Nonstationary General-time-reversible Models to Obtain Edge-lengths and Frequencies for the Barry-Hartigan Model. Under Review .