

USER GUIDE - BH PROGRAM

Version 1.0

Created By: Vivek Jayaswal

Table of Contents

Java Programs	
Installing Java	3
Maximum Likelihood Programs	3
• Compiling Programs	4
• Input	4
• Output	5
• Executing Tree Traversal Program	6
○ Input	6
○ Output	6
Divergence Matrices Programs	7
• Compiling	7
• Input	7
• Output	8
R Programs	
Installing R	8
Computing Branch Lengths	8
References	8

1. JAVA PROGRAMS

1.1 Installing Java

Java™ can be downloaded from the website <http://www.java.sun.com>. The programs were developed using JDK 1.4 and can be compiled using either JDK1.4 or JDK1.5. There are several IDEs (**I**ntegrated **D**evelopment **E**nvironment) available for developing/ modifying Java programs e.g. JCreator, NetBeans and Eclipse. An advanced user may use any of these IDE's for viewing and/ or modifying the existing BH programs.

1.2 Maximum Likelihood Programs

This sub-section provides some information about the various methods implemented in Java and can be skipped by a user not interested in the actual implementation of the program.

Table 1 provides a brief description of the programs required for computing Maximum-Likelihood using Barry and Hartigan's algorithm (Statistical Science-1987, 2:191-210). Some of the methods and program names are based on the terminology used by Barry and Hartigan and users are encouraged to read the journal article cited above. For a detailed description of the methods defined in each of these programs, users should read the relevant HTML files .e.g. to obtain information about the methods defined in the program BranchDetails.java, go to <directory>/HTML Files/BranchDetails/index.html

TABLE 1 List of programs for computing maximum-likelihood

Program Name	Brief Description
BranchDetails.java	This program contains methods for storing and retrieving a 4*4 matrix of joint probability distribution values along each edge of the phylogenetic tree
InternalNodeRiXi.java	This program contains methods for storing and retrieving RiXi values for each internal node for all 4 nucleotide types
InternalNodeSiXi.java	This program contains methods for storing and retrieving SiXi values for each internal node for all 4 nucleotide types
MaximumAverageLikelihood.java	This program contains functions for reading a set of DNA sequences, initializing the joint probability distribution values along each branch, computing log likelihood based on Q-values and updating Q-values
NodeStructure.java	This program contains methods for storing and retrieving parent node value, child node value and node type (Leaf/ Internal)
NewickTreeTraversal.java	This program contains the methods for constructing a phylogenetic tree from a Newick tree representation
QmatrixGenerator.java	This program is used to generate random Q-matrices. The matrix generated could be either symmetric or non-symmetric

DisplayTree.java

This program is used to generate a JPG file corresponding to the user-specified Newick Tree. It is useful for identifying the nodes (terminal/internal) linked to a particular internal node

1.2.1 Compiling Programs

1. Download the file BH.zip
2. Unzip the file to obtain the following folders:
 - a. HTML Files: This folder contains the HTML files corresponding to the different java classes and provides information about the various methods implemented in each class
 - b. Java: This folder contains the source code for the java classes described in Table 1
 - c. R: This folder contains the source code for the program to be implemented in R (a Statistical Software Package)
 - d. Sample Files: This folder contains the example input and output files
3. Go to the command prompt. Windows users can click on Start -> Run and then type cmd to go to the command prompt
4. Type **set classpath=%classpath%;C:/BH/Java** at the command prompt. For Unix users the command would be **export CLASSPATH=\$CLASSPATH:/BH/Java**
5. Change the present working directory to BH/Java e.g. Windows users can type **cd C:/BH/Java**
6. Type **javac MaximumAverageLikelihood.java** to compile the java programs
7. Type **java MaximumAverageLikelihood** to execute the program and obtain **log likelihood** value

1.2.2 Input

Table 2 describes the input parameters required by the MaximumAverageLikelihood program.

TABLE 2 Input parameters for the program MaximumAverageLikelihood

Parameter Name	Brief Description	Sample Value
1. No of sites	Number of nucleotide sites per sequence	1206
2. Directory of sequence files	Name and path of the input sequence file. It should be a PHYLIP file with sequences being sequential and not interleaved	C:/javapgms/Input/ND7.txt
3. Options for Q-matrices: 1 -> Default, 2 -> User File, 3 -> Random	<p>1: Uses a Q-matrix with the diagonal elements being 1/8 and off-diagonal elements being 1/24 for each edge</p> <p>2: Q-values initialization file: Name and complete path of the user-defined Q-matrices file. A blank line should separate the Q-matrices. Each line should have 4 values separated by “/” character (generated by pressing the “TAB” key). The Q-matrices are read from the source file in a pre-determined</p>	

order which can be identified by running the program NewickTreeTraversal (refer section 1.2.4)

3: Random: Randomly generate Q-matrices for each edge. The Q-matrices could be either symmetric or non-symmetric. The Q-values generated are stored in a file **“InitMatrices.txt”** in the same folder as the destination file

4. Output file (name and path)	Name and path of the output file that stores log likelihood value at convergence and Q-matrices along each edge	C:/javapgms/Output/out.txt
5. Tree Representation	Newick tree representation	(Or((Mc,Gb)(Hm(Gr(Ch,Bo))))))
6. Display phylogenetic tree (0 -> No, 1 -> Yes)	Obtain a pictorial representation of the gene tree in JPG format. The file is created in the same folder as the output file	
7. Specify Iteration cut-off value/ Number of iterations (1/2)	<p>1: Specify a cut-off value (e.g. 0.0000025): The log likelihood value at convergence may vary slightly based on the cut-off value chosen. Usually this value should be less than or equal to 0.0000025</p> <p>such that $\sum_{i=1}^4 \sum_{j=1}^4 [Q^{\text{new}}(i,j) - Q^{\text{old}}(i,j)]^2 \leq \text{cut-off value}$</p> <p>where, Q^{new} = new joint probability distribution matrix Q^{old} = old joint probability distribution matrix $Q(i,j)$ = $(i,j)^{\text{th}}$ element of the Q-matrix</p> <p>2: Number of iteration: Specify the number of iterations e.g. 10</p>	

1.2.3 Output

The converged log likelihood value is displayed at the command prompt (figure 1) and the graphical representation of the phylogenetic tree is saved as a jpg file called tree.jpg. In the jpg file, each node name (internal/ leaf) is shortened and only the first two characters are displayed e.g. Orangutan is shortened to Or in the jpg file.

```

----- INPUT PARAMETERS -----
No of sites: 1206
Directory of sequence files: H:\My_Documents\UserGuide-publication\ND7.txt
Options for Q-matrices: 1 -> Default, 2-> User File, 3-> Random
1
Output file (name and path): H:\My_Documents\UserGuide-publication\out.txt
Tree Representation: (Orangutan((Macaque,Gibbon)(Human,(Gorilla,(Chimpanzee,Bonobo))))))
Display phylogenetic tree (0 -> No, 1 -> Yes): 1
Specify Iteration cut-off value/ Number of iterations (1/2): 2
Number of iterations: 10
Program Started -> Wed Jul 20 06:57:56 GMT 2005
Modified number of sites = 1206
Sequences read
Program Completed -> Wed Jul 20 06:57:58 GMT 2005
Log Likelihood Value = -3539.283688199969

```

Figure 1 Console Output for the program MaximumAverageLikelihood

The program MaximumAverageLikelihood generates the following output files:

1. divMatInput.txt: This file contains the Q-matrices at convergence and acts as an input file for generating divergence matrices
2. User-specified output file: This is the output file specified by the user. It contains the log likelihood values and the Q-matrices for all iterations. If the user selects option “2” for input parameter number 6, only those Q-matrices that have changed since the previous iteration are stored. A Q-matrix is assumed to have changed if

$$\sum_{i=1}^4 \sum_{j=1}^4 [Q^{\text{new}}(i,j) - Q^{\text{old}}(i,j)]^2 > \text{cut-off value}$$

For each Q-matrix, there is a description is of the form “Probability Distribution Values b/w parent node and child node (xx-Node)” immediately above it. The link between xx-node and its immediate parent node determines the edge for which a given Q-matrix is defined. The parent node can be identified using the graphics file tree.jpg. The rows of a Q-matrix correspond to the parent node and the columns to the child node (xx-node). The first row (or column) represents the values for base A, the second row (or column) represents the values for base C, the third row (or column) represents the values for base G and the fourth row (or column) represents the values for base T (figure 2).

Probability Distribution Values b/w parent node and child node (1-Node)

```
0.29 0.00 0.00 0.00
0.00 0.28 0.00 0.01
0.00 0.00 0.12 0.00
0.00 0.01 0.00 0.27
```

		Child Node (1-Node)			
		A	C	G	T
Parent of 1-Node	A	0.29	0.00	0.00	0.00
	C	0.00	0.28	0.00	0.01
	G	0.00	0.00	0.12	0.00
	T	0.00	0.01	0.00	0.27

(a) Q-matrix stored in output file

(b) Interpretation of Q-matrix shown in (a)

Figure 2 Reading and interpreting a Q-matrix

1.2.4 Executing TreeTraversal Program

The program NewickTreeTraversal should be run by the user to determine the order in which the Q-matrices initialization file is read. The program is executed by typing *java NewickTreeTraversal* at the command prompt.

Input

The input parameters required by the NewickTreeTraversal program are specified in table 3.

TABLE 3 Input parameters for the program NewickTreeTraversal

Parameter Name	Brief Description	Sample Value
1. Newick Tree Representation	Newick format tree	(Or((Mc,Gb)(Hm(Gr(Ch,Bo))))))
2. JPG file path	Name and path of JPG file	C:/javapgms/Output/tree.jpg

Output

The exact order in which the edges are read from the Q-matrices initialization file is displayed at command prompt (figure 3). The links between the various nodes can be viewed by opening the JPG file created by the NewickTreeTraversal program.

```

----- INPUT PARAMETERS -----
Newick Tree Representation: <Orangutan<<Macaque,Gibbon>><Human,<Gorilla,<Chimpanzee,Bonobo>>>>
JPG file path: H:\My Documents\UserGuide-publication\1.jpg
----- Order of reading Q-matrices -----
Edge defined by parent node = 5-Node and child node = 1-Node
Edge defined by parent node = 3-Node and child node = 2-Node
Edge defined by parent node = 4-Node and child node = 3-Node
Edge defined by parent node = 5-Node and child node = 4-Node
Edge defined by parent node = Orangutan-Node and child node = 5-Node
Edge defined by parent node = 2-Node and child node = Bonobo-Node
Edge defined by parent node = 2-Node and child node = Chimpanzee-Node
Edge defined by parent node = 1-Node and child node = Gibbon-Node
Edge defined by parent node = 3-Node and child node = Gorilla-Node
Edge defined by parent node = 4-Node and child node = Human-Node
Edge defined by parent node = 1-Node and child node = Macaque-Node

```

Figure 3 Console Output for the program NewickTreeTraversal

1.3 Divergence Matrices Programs

Table 4 provides a brief description of the programs required for computing Divergence Matrices. For a detailed description of the methods defined in each of these programs, users should read the relevant HTML files

TABLE 4 List of programs for computing divergence matrices

Program Name	Brief Description
ComputeFMatrices.java	This program contains the methods for computing F-matrices (divergence matrices) for all pairs of leaf nodes of a given phylogenetic tree. The rows correspond to values of the leaf node that appears first in the output file
DivergenceMatrices.java	This class contains the methods for constructing a phylogenetic tree, determining its internal nodes and leaf nodes, and calling methods for computation of F matrices (divergence matrices)

1.3.1 Compiling Programs

- 1 Type **javac DivergenceMatrices.java** to compile the java programs
- 2 Type **java DivergenceMatrices** to execute the program

1.3.2 Input

The input parameters required by the DivergenceMatrices program are specified in table 5.

TABLE 5 Input parameters for the program DivergenceMatrices

Parameter Name	Brief Description	Sample Value
1. Enter tree topology (Newick Tree)	Tree topology in Newick format	(Or((Mc,Gb)(Hm(Gr(Ch,Bo))))))
2. Enter divergence matrix input file	• Name and path of divergence matrix source file (divMatInput.txt) created in section 1.2.3	C:/javapgms/Output/divMatInput.txt

1.3.3 Output

The divergence matrices for all pairs of leaf nodes are saved in a file called divMatOutput.txt in the same folder that contains the divMatInput.txt file. For each leaf node pair, rows correspond to the first leaf node and columns correspond to the second leaf node. The first row (or column) represents the values for base A, the second row (or column) represents the values for base C, the third row (or column) represents the values for base G and the fourth row (or column) represents the values for base T.

2. R PROGRAMS

2.1 Installing R

R can be downloaded and installed from the CRAN website <http://cran.mirrors.pair.com>.

2.2 Computing Branch Lengths

Download the file getLength.R and load it in R. This program takes the joint probability distribution matrix (Q-matrix) for an edge as input and returns the average branch length as output. The Q-matrix along an edge can be obtained from the “User-specified output file” (refer section 1.2.3) under the section “FINAL Q-MATRICES”.

Example: If qVal is a 4*4 matrix of joint probability distribution values, type the command **get.length(qVal)** inside R to obtain the branch length.

3. REFERENCES

Ababneh, F., L. S. Jermin, C. Ma, and J. Robinson. 2006. Matched-pairs tests of homogeneity with applications to homologous nucleotide sequences. *Bioinf.* (in press).

Barry D, Hartigan JA. 1987a. Statistical analysis of hominoid molecular evolution. *Stat Sci*, 2:191-210.

Rodríguez F, Oliver JL, Marin A, et al 1990. The general stochastic model of nucleotide substitution. *J Theor Biol*, 142: 485-501.