PROGRAM_NAME Release v1.0.0

(program name acronym spelled out)

Description

PROGRAM_NAME is a simulation / optimization procedure that estimates evolutionary distance between two sequences, mean of indel length, and rate of insertion and rate of deletion per substitution, given raw alignment sequences in a FASTA file format.

Citations

Please cite the application as:

Khare, A. PROGRAM_NAME: Simulation / optimizaiton procedure to infer estimates of evolutionary distance, mean of indel length, and rate of insertion and rate of deletion per substitution.

Getting Started

In order to run PROGRAM_NAME, you need to install R on your computer. Download the most recent version of R from http://www.r-project.org/ and follow installation instructions. The library "optparse" also needs to be downloaded once R is installed. To install this package, run the command as follows:

```
install.packages("optparse", repose="http://R-Forge.R-Project.org")
```

You will also needs to download and install Ngila Release-### or greater from http://scit.us/projects/ngila/

You will also need to download and install Dawg Release-### or greater form http://scit.us/projects/dawg/

After downloading the PROGRAM_NAME package, unzip hte file folder and store in your desktop or in your preferred directory. The forlder contains the following application and documentation files:

Optimize.R	PROGRAM_NAME application
ngila_wrapper.cpp	wrapper to use Ngila to make initial biased estimates
regression.RData	data file in R containing linear regression models
Makefile	file for ease of usage to compile ngila_wrapper.cpp
check.fasta	example file to test usage of PROGRAM_NAME
manual.pdf	this manual

Compiling the code

PROGRAM_NAME runs from the command line. To open the command prompt from Windows, you can use the start menu (e.g. Start -> All Programs -> Accessories -> Command Prompt). Change the directory to the location of the files, e.g.

```
cd %USERPROFILE%\Desktop\PROGRAM_NAME
```

Compile the ngila_wrapper.cpp by typing:

```
make
```

This should produce the object file ngila_wrapper.o and executable NgilaWrapper. If at any time you would like to remove the object file and the executable, type the following command:

```
make clean
```

It should be noted however, that if you want to use the PROGRAM_NAME properly, ngila_wrapper.cpp should be compiled correctly first.

Usage Example

I will walk you through an example of how to utilize PROGRAM_NAME on let's say, a linux environment.

The first step of utilizing the software is to make sure a proper input file is present. PROGRAM_NAME accepts input files in the FASTA file format. More information can be found in the link below regarding the FASTA file format:

```
http://prodata.swmed.edu/mummals/info/fasta_format_file_example.htm
```

The file check.fasta that came with the download package contains 100 repetitions of alignment sequences between taxons "A_%r" and "C_%r". Once we have the FASTA file that we want to analyze, the next part is to properly input the file into PROGRAM_NAME.

Help for the command line arguments of Optimize.R can be accessed by typing:

```
./Optimize.R –h
```

Here are the command line arguments that exist for Optimize.R:

-f Name of input file (must be in FASTA format) [default = NULL]

-c Number of cycles [default = 15]

This is the number of times PROGRAM_NAME iterates before widening the prediction interval. The higher the number of cycles, the more accurate the estimates are, however, the cost is processing time. Once the cycle limit is reached, the number of cycles is increased by a factor of 1.5 (rounded to nearest integer) along with the widening of the prediction interval. It is advised that the number of cycles not be set to less than 6.

-t Threshold value for parameter estimate t [default = 1×10^{-5}]

This is the threshold value for parameter estimate t that is used by PROGRAM_NAME to produce an estimate for parameter t. Until the cost value for parameter t estimate does not reach below the set threshold value along with the other two estimates, PROGRAM_NAME continues to search. The lower the value is, the higher the processing time; but higher the accuracy of estimates.

-m Threshold value for parameter estimate m [default = 1×10^{-3}]

This is the threshold value for parameter estimate m that is used by PROGRAM_NAME to produce an estimate for parameter m. Until the cost value for parameter m estimate does not reach below the set threshold value along with the other two estimates, PROGRAM_NAME continues to search. The lower the value is, the higher the processing time; but higher the accuracy of estimates.

-l Threshold value for parameter estimate l [default = 8×10^{-5}]

This is the threshold value for parameter estimate l that is used by PROGRAM_NAME to produce an estimate for parameter l. Until the cost value for parameter l estimate does not reach below the set threshold value along with the other two estimates, PROGRAM_NAME continues to search. The lower the value is, the higher the processing time; but higher the accuracy of estimates.

-q disable all warning messages [default = F]

This parameter should be set to “T” if one wants to disable all warning messages. Only arguments “T” or “F” are accepted.

-o Name of output file in [default = NULL]

-h help option

Now that we have an input file and an understanding of the valid command line arguments for PROGRAM_NAME, let us analyze check.fasta via this command:

```
./Optimize.R -f check.fasta -c 4 -q T -o results.txt
```

Here we have specified the input file name via the –f argument, set the number of cycles to 4 via the –c argument, turned off all warning messages via –q argument, and defined the name of the output file (results.txt).

As the PROGRAM_NAME goes through each cycle, it will display the results of that particular cycle as shown below:

```
==========================================================

cycle number:  3  of  4

[t estimate, cost]: [ 0.09973145 ,  9.404474e-05 ]
[m estimate, cost]: [ 8.177963 ,  0.001184498 ]
[l estimate, cost]: [ 0.02448441 ,  3.990776e-07 ]

==========================================================
```

The part highlighted in green displays the cycle at which PROGRAM_NAME just finished. The region highlighted in yellow displays the particular estimate for each parameter along with its associated cost value. If PROGRAM_NAME is unable to make an estimated within the defined number of cycles, the following output will be seen:

```
the prediction intervals have been widened
number of cycles has been updated to:  6
```

Here we can see that the prediction interval widened and the number of cycles was increase to 1.5*4 to the nearest integer. Once the program is able to make the final estimates, the following output will be seen on the console:

```
==========================================================

The Simulation is Complete

Estimated value for Parameter t:  0.09792204
Estimated value for Parameter m:  8.214338
Estimated value for Parameter l:  0.02464569

Duration of simulation procedure (s) :

   user  system elapsed
616.025  36.256 653.830

==========================================================
```

The final output message displays the final parameter estimates (in green) and the total time it took for PROGRAM_NAME to complete the optimization task (in yellow). Because we also defined an output file, this is what is seen in results.txt:

Parameter t	Parameter m	Parameter l
0.09792204	8.214338	0.1250199

Contact

Any questions or concerns or reporting of bugs should be directed to Akash Khare <akhare4@asu.edu>