CAFE v3.0
Software for **C**omputational **A**nalysis of gene **F**amily **E**volution

Mira V. Han, Gregg W.C. Thomas, and Matthew W. Hahn

August 1, 2013

The purpose of CAFE is to analyze changes in gene family size in a way that accounts for phylogenetic history and provides a statistical foundation for evolutionary inferences. The program uses a birth and death process to model gene gain and loss across a user-specified phylogenetic tree. The distribution of family sizes generated under this model can provide a basis for assessing the significance of the observed family size differences among taxa.

CAFE v3.0 is a major update to CAFE v2.1. This document describes how to download and use CAFE v3.0. Major updates in 3.0 include: 1) the ability to correct for genome assembly and annotation error when analyzing gene family evolution using the **errormodel** command. 2) The ability to estimate separate birth (λ) and death (μ) rates using the **lambdamu** command. 3) The ability to estimate error in an input data set with iterative use of the errormodel command using the accompanying python script **caferror.py**. This version also includes the addition of the **rootdist** command to give the user more control over simulations.

The necessary inputs for CAFE v3.0 are:

1) a **data file** containing gene family sizes for the taxa included in the phylogenetic tree
2) a **Newick formatted phylogenetic tree**, including branch lengths

From the inputs above, CAFE v3.0 will compute

1) the **maximum likelihood value of the birth & death parameter, λ** (or of separate birth and death parameters (λ and μ, respectively)**,** over the whole tree or for user-specified subsets of branches in the tree
2) **ancestral states** of gene family sizes for each node in the phylogenetic tree
3) **p-values** for each gene family describing the likelihood of the observed sizes given average rates of gain and loss
4) **average gene family expansion** along each branch in the tree
5) numbers of **gene families with expansions, contractions, or no change** along each branch in the tree

CITING CAFE

An appropriate citation for use of CAFE 3.0 in published research is:

Han MV, Thomas GWC, Lugo-Martinez J, and Hahn MW (2013) Estimating gene gain and loss rates in the presence of error in genome assembly and annotation using CAFE 3. *Molecular Biology and Evolution*. In press.

Original development of the statistical framework and algorithms implemented in CAFE are published in:

Hahn MW, De Bie T, Stajich JE, Nguyen C, and Cristianini N (2005) Estimating the tempo and mode of gene family evolution from comparative genomic data. *Genome Research.* 15(8):1153-1160

De Bie T, Cristianini N, Demuth JP, and Hahn MW (2006) CAFE: a computational tool for the study of gene family evolution. *Bioinformatics.* 22:1269-1271.

DOWNLOADING

CAFE v3.0 is available from:

http://www.bio.indiana.edu/~hahnlab/Software.html

or **http://sourceforge.net/projects/cafehahnlab/**

CAFE v3.0 is available in compiled versions for Mac OS X and linux (x86_64). Alternatively you may download and compile the source code yourself.

LAUNCHING

CAFE v3.0 is implemented as a shell. The program can be run interactively by simply launching the shell, or the user may execute a shell script that lists a series of CAFE commands saved as a separate text file.

ENTER CAFE SHELL: To run CAFE interactively, launch the shell by typing `cafe` at the UNIX prompt. If all is well the prompt should change to `#`. You may now begin inputting commands. To exit the shell, type `exit`.

EXECUTING CAFE SHELL SCRIPTS: Because many analyses will require a similar series of inputs, you may also run CAFE using a shell script. CAFE scripts should be saved as text files with UNIX line endings. Scripts may be executed from the UNIX prompt or from within the CAFE shell.

Example CAFE script:

```
#!cafe
#version
#date
load -i data/example2.tab -t 10 -l logfile.txt -p 0.05
tree (((chimp:6,human:6):81,(mouse:17,rat:17):70):6,dog:93)
lambda -s -t (((1,1)1,(2,2)2)2,2)
report resultfile
```

In this example, the first line indicates the location of the CAFE shell program. Subsequently, lines beginning with "#" are regarded as comments. Thus, the example above only executes lines 4, 5, 6, and 7. Remember that to run a script you must make the file executable from the UNIX prompt by typing in `chmod +x` *`filename`*. CAFE will automatically exit after the last command in the script is completed, so it is not necessary to specify `exit`.

An example script is available on the CAFE website listed in the DOWNLOADING section above.

COMMANDS

Command	Brief Description
source	run shell script
version	version info
date	date/time
exit	exit CAFE shell
log	log output
load	data file & run parameters
tree	phylogenetic tree
lambda	find/specify birth-death parameter
lambdamu	find/specify separate birth and death parameters
errormodel	apply error-correction to data
noerrormodel	removes applied error models
caferror.py	estimate error in data file
esterror	estimates error matrix from multiple datasets
simerror	simulate/add error to the data based on errormodel
report	report values
genfamily	generate simulated data
rootdist	specify root family size distribution for simulation
lhtest	compare likelihoods of lambda models

 # source `filename`

Load shell script file

 # version

Display the CAFE version number

 # date

Display the current date and time

 # exit

Exit the CAFE shell (`quit` will perform the same action)

 # log `[filename]`

If no filename is given, this command displays the current log file. If a filename is specified, CAFE will create (or overwrite) the log file. Default: stdout (output to screen only). The log file may also be specified in the load command using the `-l` option.

```
# load –i filename [-t integer ] [-l filename ] [-p 0.01 ] [-r 1000 ]
```

-i: DATA FILE: Enter the path to the file containing gene family data. The data file format must be tab delimited with UNIX line endings. Family description may contain spaces (but not tabs). The first line must contain labels in the order: Description, ID, and then the names of each taxon separated by tabs. *If you do not have a Description or ID, CAFE still requires two tabs at the beginning of each line*. The taxon names must be spelled exactly as they are in the **tree structure**. Subsequent lines each correspond to a single gene family. If the data file contains taxa that do not appear in the tree structure they are not considered in the analysis.

Example Data File:

Description	ID	Chimp	Human	Mouse	Rat	Dog
EF 1 ALPHA	ENSF00000000004	5	8	6	12	40
HLA CLASS II	ENSF00000000007	4	4	3	3	3
HLA CLASS I	ENSF00000000014	5	3	5	6	3
RAG 1	ENSF00000000015	1	1	1	1	1
IG HEAVY CHAIN	ENSF00000000020	32	42	51	60	18
ACTIN	ENSF00000000027	27	30	22	28	25
OPSIN	ENSF00000000029	2	2	2	2	2
HEAVY CHAIN	ENSF00000000030	25	25	23	24	18

If the file is loaded correctly, CAFE will output summary information about the current data file to the logfile.

An example data set is available for download from the CAFE website.

-t: The maximum number of CPU threads to be used. Default: 8

-l: LOG FILE: Enter the path and filename where CAFE will write the **main output**. This file will contain a summary of input parameters as well as details of λ searches, including likelihood scores and maximum likelihood values of λ. If the file does not exist, CAFE will create it for you; if the file already exists, CAFE will append results to the previous file. Default: output to screen (no log file created).

-p: P-VALUE THRESHOLD: For each family in the data file, CAFE computes a probability (p-value) of observing the data given the average rate of gain and loss of genes. All else being equal, families with more variance in size are expected to have lower p-values. The p-value threshold allows the user to specify the cutoff for subsequent analyses. Families with p-values larger than the designated threshold will not be included in identification of the most unlikely branch. Default: 0.01.

-r: NUMBER OF RANDOM SAMPLES: To determine the probability of a gene

family with the observed sizes among taxa, CAFE uses a Monte Carlo re-sampling procedure. This option specifies the number of samples CAFE should use to calculate p-values. The tradeoff is between precision and computation time; in most cases 1000 samples should provide reasonable balance. Default: 1000.

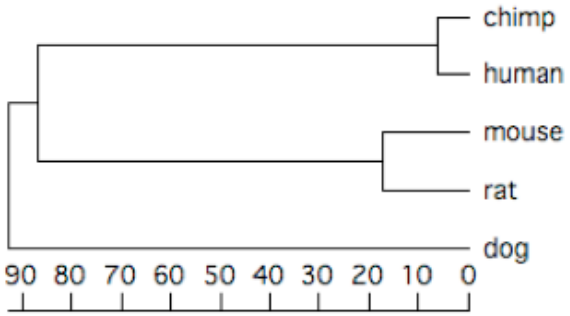
-filter: The birth-death model of CAFE assumes at least one gene in the root of the species tree. This assumption may not be valid for families that were created after the most recent common ancestor of all species. The filter option filters out the families that are inferred (by parsimony) to have no genes in the root node of the species tree.

 # tree tree_structure

TREE STRUCTURE: A Newick formatted tree containing branch lengths and taxon names as they are specified in the input file. Branch lengths should be integer units of time and ultrametric (i.e. the sum of the lengths from root to tip should be the same for all paths). For instructions on converting trees to Newick format visit:
http://evolution.genetics.washington.edu/phylip/newicktree.html.

Warning: There should be no spaces in the tree string and no semicolon at the end of the line.

Example:
In Newick format, the tree diagram below is represented as:
`(((chimp:6,human:6):81,(mouse:17,rat:17):70):6,dog:93)`



 # lambda [-l values | -s | -r start:step:end] [-t lambda_structure]

Warning: The **load** and **tree** commands must be run prior to the **lambda** command.

-l: SPECIFY λ VALUES: This option allows the user to specify the value(s) of λ. If more than one λ is specified, then the –t option must also be used. λ values are specified in the order 1 2 3 … which correspond to the integers specified in the `-t` option. λ values should be separated by spaces.

Warning: 1) The product of λ and the depth of the tree structure should not exceed one (i.e. $\lambda * t < 1$ must be true; where t is the time from the tips to the root). See the section on Known Limitations below for details to diagnose this problem.
2) CAFE will use the last λ value(s) estimated (or user specified) to compute ancestral gene family sizes and to run Monte Carlo simulations.

-s: SEARCH FOR λ VALUE(S): Cafe will search using an optimization algorithm to find the value(s) of λ that maximize the log likelihood of the data for all families. CAFE starts with an intermediate value and then searches iteratively for the best value for λ (or set of λ values if used in conjunction with the `-t` option). Subsequent analyses will automatically use the results from the lambda search.

-r: SEARCH λ IN SPECIFIED RANGE: Returns the likelihood scores for λ values in the user specified range. The format of a range is start:step:end. For example, to see the score distribution with a lambda between 0.003 and 0.005, the range would be 0.003:0.001:0.005. In case of more than one lambda, ranges are separated by a space.

-t: λ STRUCTURE: To investigate whether different parts of the tree are evolving at different rates, the user may specify which branches of the **`tree`** will take the same of different lambda values. Input the same Newick tree structure as in the tree command, but exclude branch lengths and substitute integer values from 1 up to *n* for taxon names (where *n* = the total number of branches on the tree; matching integer values indicate that these branches will take the same value of λ). Default: all branches have the same value for λ.

Example λ structure:
The following lambda structure specifies one λ value for the Human, Chimp, and Ape branches and a second λ value for all other branches based on the tree specified in the **`tree`** command section above.

```
(((1,1)1,(2,2)2)2,2)
```

Warning: CAFE will not always converge to a single optimum with models that contain many parameters. See the Known Limitations section below for details on assessing this problem.

 # lambdamu [-l lambda_list –m mu_list] | -s] [-t lambda_structure] [-eqbg]

-l: SPECIFY λ VALUES: Identical to `-l` for the **lambda** command, but now must be used in conjunction with `-m`.

-m: SPECIFY μ VALUES: Similar to how `-l` works, the user can specify the values of the death rate, μ.

W*arning*: 1) The product of λ or μ and the depth of the tree should not exceed one (i.e. $\lambda * t < 1$ and $\mu * t < 1$ must be true; where t is the time from the tips to the root). See the section on Known Limitations below for details to diagnose this problem.
2) CAFE will use the last λ/μ value(s) estimated (or user specified) to compute ancestral gene family sizes and to run Monte Carlo simulations.

-s: SEARCH FOR λ/μ VALUE(S): Identical to the `-s` option from the **lambda** command, however CAFE v3.0 can now find separate birth (λ) and death (μ) rates if the **lambdamu** command is specified.

-t: λ/μ STRUCTURE: Identical to the `-t` option from the **lambda** command, but used in conjunction with the **lambdamu** command the tree structure can specify branches on which to estimate separate birth (λ) and death (μ) rates. In other words, branches with the same numerical identifier will share λ and μ parameters. See the `-t` section in the **lambda** command for a sample lambda structure. Default: all branches have the same value for λ and μ.

-eqbg: BACKGROUND CONSTRAINT: Only used when the lambda structure is specified with `-t`. This option allows the user to constrain the background rate (those branches with numerical identifier "1") to have equal values of $\lambda = \mu$, while other branches are allowed separate estimates of λ and μ.

 # errormodel [-model filename] [-sp speciesname | -all]

The **errormodel** command allows the user to specify an error distribution. CAFE will correct for this error before calculating ancestral family sizes and estimating λ values. The **errormodel** function is also utilized by **caferror.py** to estimate error in the input data set.

-model: SPECIFY ERROR MODEL FILE: This option allows the user to specify the file

name of the error model file to use in order to correct the input data for errors. The error model file format is as follows:

```
maxcnt: 68
cntdiff -1 0 1
0 0.0 0.8 0.2
1 0.2 0.6 0.2
2 0.2 0.6 0.2
.
.
.
68 0.2 0.6 0.2
```

"maxcnt" is the largest family size observed in the dataset. "cntdiff" defines the error classes for all following rows. Error classes should be space delimited positive or negative integers (and 0) and act as labels for error distributions for each gene family size. The error class with label 0 means that this corresponds to no change in gene family size due to error. After the first two lines, each possible family size in the dataset (size 0 to maxcnt) should have an error distribution defined. Any omitted family size follows the distribution for the previous row. The error distribution for each count should be space delimited probabilities whose columns correspond to the error classes defined in line two. Default: No error model is applied.

Note: 1) Do not specify any negative error correction for family size of 0 as this cannot occur (i.e. there can't be negative gene family sizes).
2) The rows of the error model file must sum to 1.
3) If any gene counts are missing from the error model file, CAFE will assume the same error distribution from the previous line. This can also be used as a shortcut if you know that all of the gene counts are specified with the same error distribution: simply enter the first four lines (maxcnt, cntdiff, *family size*=0,1) into the error model file and CAFE will use the distribution for *family size*=1 as the distribution for all gene family sizes.

-sp: SPECIFY SPECIES: This option is required to specify the species to which the error model will be applied. Species names must be identical to those in the data file and the input tree. The user may specify any combination of species with the same or different errormodel files with separate **`errormodel`** commands, or the user may specify all species with the same errormodel file in one **`errormodel`** command using `-all` as the species option here.

`# noerrormodel` `[-sp species | -all ]`

-sp: REMOVE ERROR MODEL FROM SPECIES: This option will remove a previously

specified error model from the specified `species` and returns it to the default state of no error model. The species name should be identical to those in the data file and input tree. To remove the error models from all species, use `-all`.

```
$ python caferror.py -i [input shell file name ] [-e 0.4 ] [-d output
directory name ] [-l log filename ] [-o output filename ] [-s value ]
```

caferror is a Python script included with the CAFE v3.0 software package that utilizes the **errormodel** command iteratively to estimate error in an input data set with no prior knowledge of the error distribution. **caferror** utilizes the likelihood scores of runs with varying error models to perform a precise grid search of the likelihood surface. The program first estimates average global error across all species in the input phylogeny and then may continue to individual species estimations depending on `-s`.

Warning: 1) Python v2.6 or later must be installed on your machine. The Python interpreter is freely available from www.python.org.
2) **caferror** must be run in the directory in which CAFE is located as it uses that path to run CAFE.

-i: SHELL FILE NAME: The main input of **caferror.py** is a CAFE shell script, as shown above. **caferror** will extract the following information needed from the shell script and use it to run CAFE many times to estimate error: the input gene family file from the **load** command, the **tree** command, and the **lambda** command. **caferror** will not overwrite the input script, but will instead write its own.

-e: INITIAL ERROR VALUE: This is the value with which **caferror** will begin the grid search. This should be a floating point value between 0 and 1. Default: 0.4

-d: DIRECTORY: **caferror** runs CAFE many times, and therefore creates and stores many error model and CAFE log files. All CAFE log files, error model files, and **caferror** output files will be stored in a directory specified with this option. If the directory has not been created, **caferror** will create it automatically. Default: caferror_tmp_dir_x, where x is an integer one higher than the previous default directory.

-l: LOG FILE NAME: **caferror** keeps track of the error estimates and scores in its own log file. This is also where you will find the final error estimates. The user may specify the name of the file with this option. Default: caferrorLog.txt.

-o: OUTPUT FILE: The user may specify a name for the output file with this option. The error estimation algorithms create a curve for

visualization if plotted, and this output file contains two tab-delimited columns consisting of error model and the corresponding score while using that error model. Simply copy/paste these data points into your favorite graphing software to see how **caferror** estimated the error. Default: caferror_default_output.txt.

Note: this only outputs data points for the global error estimation.

-s: SPECIES OPTION: This is a binary option and should be set as either 1 to continue error estimation on individual species after global error estimation has completed or 0 to stop running after global error estimation. Default: 0

 # esterror [-dataerror file1 -datatrue file2] | [-dataerror file1 file2] [-diff number] [-symm] –o outfile

esterror estimates the error matrix from two different gene family datasets. If the user gives two `-dataerror` options, the error is estimated assuming two error-prone measures. If the user gives one `-dataerror` option and one `-datatrue` option, the error is estimated assuming `-datatrue` is the true measure and `-dataerror` is the error prone measure.

-diff: MATRIX SIZE: constrains the error matrix parameters to `number` rows away from the main diagonal. Default is a maximum difference of two counts.

-symm: SYMMETRIC ERROR MATRIX: constrains the error matrix to be symmetrical between the upper triangle and the lower triangle. Default is an asymmetrical model.

-o: OUTFILE NAME: specifies the output file that the estimated error model will be written to. The user can then use the file `outfile` with the command **errormodel** to set the estimated model to the appropriate species.

 # report filename

The **report** command outputs results. Although all analyses must be specified by their own commands, **report** directs the output of CAFE.

`filename`: The file where CAFE will write the main results of gene family analysis. Do not add an extension to the file name.

A description of the main output file follows:

`filename.cafe`: A tab delimited summary of results. Information

provided in `filename.cafe` includes:

`tree`: The current tree

`lambda(s) and likelihood`: The current lambda values set by the **`lambda`** command. It can be either specified by the user (`-l`) or obtained by searching for the maximum likelihood value (`-s`). The likelihood of the data give the current lambda value.

`average expansion`: Mean number of genes gained or lost per family, where "minus" expansion is a net contraction.

`expansion and contractions`: Total count of families that experienced expansions, contractions, or no change along each branch of the species tree.

`List of family and description`.

`List of overall p-value for each family`: The p-values are based on a Monte Carlo re-sampling procedure. To determine the probability of a gene family with the observed sizes among taxa, CAFE will generate the expected distribution of family sizes under the stochastic birth-death model for the tree specified in the load command with the current λ value. Running the simulations uses the most machine resources and thus is the most time intensive step in CAFE. For each family in the data file, CAFE computes a probability (p-value) of observing the data given the average rate of gain and loss of genes. All else being equal, families with more variance in size are expected to have lower p-values.

`List of branch-specific p-values for the significant families`: The branch-specific p-values are obtained by the Viterbi method with the randomly generated likelihood distribution. This method calculates exact p-values for transitions between the parent and child family sizes for all branches of the phylogenetic tree. A low p-value indicates a rapidly evolving branch. This information is reported only for the families with an overall p-value less than the p-value cutoff set with the **`load`** command.

`List of ancestral states for each family`: Reports the maximum likelihood values of the ancestral number of genes at all inner nodes of all gene families.

`# genfamily` `[directory/fileprefix ] [-t integer ]`

The **`genfamily`** command generates simulated data based on the properties

of observed data. These simulated data can be used for many purposes, including generating null distributions of likelihood ratios to assess the significance of multi-parameter models (see **`lhtest`**). CAFE uses the estimated root sizes from the observed data, the most recently specified λ and μ (either by search or user input), and the tree specified in the **tree** command to generate new data sets. Each simulated data file will contain the same number of families and distribution of root sizes as the observed data (unless otherwise specified with the **`rootdist`** command). To specify the data file and the value of λ, the **`load`** and **`lambda`** commands must precede **`genfamily`**.

`directory/fileprefix`: Designates the directory where CAFE will write the simulated data sets. This should be specified relative to the working directory, and must be created prior to running CAFE (i.e. CAFE will not create the directory). Each simulated data set will have the name `fileprefix_#.tab`.

`-t`: NUMBER OF SIMULATED DATASETS: Specifies the number of simulated data set for CAFE to generate.

Example: `genfamily rndtree/rnd –t 100`
This command line will generate the simulated data sets: rnd_1.tab, rnd_2.tab … rnd_100.tab and write them in the "rndtree" directory. These data sets will have the same format as typical CAFE input.

`# rootdist` `[-i filename ]`

Used before the **`genfamily`** command, this command allows the user to set the root family size distribution from which the simulations are run to generate artificial gene family sizes.

`-i`: SPECIFY ROOTDIST INPUT FILE: The **`rootdist`** input file is a text file in the format shown below. The total sum of the frequencies should add up to the number of families the user is aiming to simulate.

`rootdist` file format:

```
Var Freq Max:30
1 10697
2 563
.
.
.
30 1
```

"Var Freq Max" is the maximum root family size desired. Following

that line, the root family size is defined as the first number of each row and the number of families to have that size is the second number in the row. Specifically, the above example would specify root distributions in the following way: 10697 families would have a root size of 1, 563 families would have a root size of 2, and so on.

```
# simerror [-pre gain_diff_1/simerror1 ] [-rep 10 ]
```

Simulates and adds error to gene family sizes according to the error model specified. The error is added to the data read from the gene family input file; these data can either be simulated themselves or come from an external dataset. For this single input dataset, each replicate gets error randomly added to it. The command **load** must have been used before to load the data. The command **errormodel** must have been used before running this command to specify the error to simulate.

-pre: The prefix for the family size files generated by the error simulation

-rep: The number of replicates to be simulated.

```
# lhtest [-d directory ] [-l lambda_seed_value ] [-t lambda_structure ]
      [-o filename]
```

The **lhtest** command computes two likelihood scores for each simulated data set: one based on a model with a single, global λ and one based on a model with multiple λ values. Using simulated data with one λ parameter from **genfamily**, the results of **lhtest** can be used to generate a distribution of likelihood ratios [i.e. LR = 2*(score of global λ model – score of multi- λ model)] under the null hypothesis. This distribution can then be used to assess the significance of models with more than one λ parameter in observed data, by comparing the likelihood ratio to the distribution of ratios generated by analysis of simulated data sets. A multi-parameter model is significantly better than a single-parameter model if the observed LR is greater than 95% of the distribution of simulated LRs.

-d: DIRECTORY NAME: Specifies the directory where CAFE can find the simulated data sets generated by the **genfamily** command.

-l: SPECIFY λ VALUE: For each simulated data set, CAFE will begin the lambda search algorithm at the value specified here. Since CAFE re-estimates lambdas for each simulated data set, specifying a seed value close to the actual lambda used in the simulations may save considerable time.

-t: λ STRUCTURE: Specify the lambda structure as in the **lambda** command above. This command specifies the multi-lambda model to be

estimated.

-o: OUTPUT FILE: The file where CAFE will write the output of **`lhtest`**. The file contains columns for each of the following values: likelihood score for global lambda | estimated global lambda | likelihood score for multiple lambda model | estimate lambda 1 | estimated lambda 2 | … | estimated lambda n.

KNOWN LIMITATIONS

1. Because the random birth and death process assumes that each family has at least one gene at the root of the tree, CAFE will not provide accurate results if gene families are included that are not present in the most recent common ancestor (MRCA) of all taxa included in the tree. For example, even if all included taxa have gene family size = 0, CAFE will assign the MRCA a gene family of size 1, and include the family in estimation of the birth and death rate. This difficulty does not affect analyses containing families that go extinct subsequent to the root node.

2. If a change in gene family size is very large on a single branch, CAFE may fail to provide accurate lambda estimation and/or die during computation. To see if this is a problem, look at the likelihood scores computed during the λ search (reported in the logfile if the job finishes). If ALL scores are "-inf" then there is a problem with large size change giving CAFE a probability = 0. Removing the family with the largest difference in size among species and re-running CAFE should allow λ to be estimated on the remaining data. If the problem persists, remove the family with the next largest difference and proceed in a like manner until CAFE no longer finds families with zero probability. However, if rapidly evolving families are removed, care should be taken in interpretation of the estimated average rate of evolution for the remaining data.

3. If the product of λ and the distance from the tips to the root is greater than 1, then CAFE will not return accurate results. If λ is specified by the user, this problem is seen as @@. If the λ-search option is used, then the value of λ output will be the maximum possible for $\lambda * t < 1$. If this is a problem, CAFE will print a caution message and "@@" will appear before the Newick-formatted tree in the output. ***In our experience, this is the most common error encountered by users.***

4. In very large phylogenetic trees there can be many independent lambda parameters ($2n$-2 in a rooted tree, where n is the number of taxa). CAFE does not always converge to a single global maximum with large numbers of λ

parameters, and therefore can give misleading results. To check for this you should always run the lambda search multiple times to ensure that the same estimated values are found. Also, the likelihood of models with more parameters should always be lower than models with fewer parameters, which may not be true if CAFE has failed to find a global maximum. If CAFE does not converge over multiple runs, then one should reduce the number of parameters estimated and try again.