Package ‘cloneRate’

September 22, 2023

Type Package

Title Estimate Growth Rates from Phylogenetic Trees

Version 0.2.3

Description Quickly estimate the net growth rate of a population or clone whose growth can be approximated by a birth-death branching process. Input should be phylogenetic tree(s) of clone(s) with edge lengths corresponding to either time or mutations. Based on coalescent results in Johnson et al. (2023) <doi:10.1093/bioinformatics/btad561>. Simulation techniques as well as growth rate methods build on prior work from Lambert A. (2018) <doi:10.1016/j.tpb.2018.04.005> and Stadler T. (2009) <doi:10.1016/j.jtbi.2009.07.018>.

License MIT + file LICENSE

NeedsCompilation yes

URL https://github.com/bdj34/cloneRate,
https://bdj34.github.io/cloneRate/

BugReports https://github.com/bdj34/cloneRate/issues

Depends R (>= 3.6.0)

Imports ape (>= 4.0), methods, Rcpp (>= 0.12.0), Rmpfr (>= 0.8), rstan (>= 2.18.1), rstantools (>= 2.3.1)

Suggests car, covr, ggplot2, ggsurvfit, knitr, parallel, rmarkdown, survival, testthat (>= 3.0.0)

LinkingTo BH (>= 1.66.0), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.3.0), RcppParallel (>= 5.0.1), rstan (>= 2.18.1), StanHeaders (>= 2.18.0)

VignetteBuilder knitr

Config/testthat/edition 3

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Biarch true
SystemRequirements  GNU make

Author  Brian Johnson [cre, aut, cph] (<https://orcid.org/0000-0002-9477-4677>),
        Yubo Shuai [aut, cph],
        Jason Schweinsberg [aut, cph] (<https://orcid.org/0000-0002-4152-0059>),
        Kit Curtius [aut, cph] (<https://orcid.org/0000-0002-2678-0960>)

Maintainer  Brian Johnson <brian.d.johnson97@gmail.com>

Repository  CRAN

Date/Publication  2023-09-22 15:40:02 UTC

R topics documented:

cloneRate-package .............................................  2
birthDeathMCMC .............................................  3
coal_to_tree .............................................  4
exampleMutTrees .............................................  4
exampleUltraTrees ...........................................  5
internalLengths ...........................................  6
longitudinalData ...........................................  7
maxLikelihood .............................................  8
realCloneData .............................................  8
sharedMuts ................................................  9
simMut ..................................................  10
simUltra ..................................................  11
siteFrequency ............................................  12
ultra2mut ................................................  13

Index  14

cloneRate-package  The 'cloneRate' package.

Description

Quickly estimate the net growth rate of a population or clone whose growth can be approximated by a birth-death branching process.

References

Johnson et al. 2023 Bioinformatics. doi:10.1093/bioinformatics/btad561
birthDeathMCMC

Growth rate estimate using MCMC

**Description**

Uses Rstan and the No U-turn sampler to approximate the growth rate using the likelihood from Stadler 2009 "On incomplete sampling under birth–death models and connections to the sampling-based coalescent"

**Usage**

```r
birthDeathMCMC(
  tree,
  maxGrowthRate = 4,
  alpha = 0.05,
  verbose = TRUE,
  nChains = 4,
  nCores = 1,
  chainLength = 2000
)
```

**Arguments**

- `tree`: An ultrametric tree subset to include only the clone of interest. Alternatively, a list with several such trees.
- `maxGrowthRate`: Sets upper bound on birth rate. Default is 4 but this will depend on the nature of the data.
- `alpha`: Used for calculation of confidence intervals. 1-alpha confidence intervals used with default of alpha = 0.05 (95 percent confidence intervals).
- `verbose`: TRUE or FALSE, should the Rstan MCMC intermediate output and progress be printed?
- `nChains`: Number of chains to run in MCMC. Default is 4.
- `nCores`: Number of cores to perform MCMC. Default is 1, but chains can be run in parallel.
- `chainLength`: Number of iterations for each chain in MCMC. Default is 2000 (1000 warm-up + 1000 sampling), increase if stan tells you to.

**Value**

A dataframe including the net growth rate estimate, confidence intervals, and other important details (clone age estimate, runtime, n, etc.)

**See Also**

- `internalLengths`
- `maxLikelihood` which use alternative methods for growth rate estimation from an ultrametric tree.
Examples

def <- birthDeathMCMC(cloneRate::exampleUltraTrees[[1]])

c coal_to_tree  Generate tree from coalescence times

description generates a tree from a vector of coalescence times by randomly merging lineages.

Usage

c coal_to_tree(coal_times)

Arguments

c coal_times A numeric vector of coalescence times

Value

An ape object of class "phylo" representing the ultrametric phylogenetic tree with edge lengths in units of time.

Examples

# Generate an ape phylo tree with n tips from a vector of n-1 coalescence times
randomCoalTimes <- c(9.3, 7.8, 10.15, 11.23, 9.4, 8.8, 10.01, 13)
tree <- coal_to_tree(randomCoalTimes)

Description

Set of 100 mutation based trees reconstructed from the distribution of a sample of n=100 tips. All trees have a net growth rate of 1 with birth rates between 1 and 2 (sampled from a uniform distribution). Death rates are equal to the chosen birth rate minus 1. Tree reconstruction uses the exact distribution of coalescence times described in "The coalescent of a sample from a binary branching process", Lambert A., Theor. Pop. Bio. 2018. Tree construction and formatting uses ape R package ape::rcoal(). We then change the edge lengths from time-based to mutation-based by drawing from a poisson distribution with mean equal to edge length (in units of time) multiplied by the mutation rate, nu, which is drawn from a uniform distribution between 10 and 20 mutations per year.
Usage

data(exampleMutTrees)

Format

A list of objects of class phylo

edge A matrix of edge connections which reconstruct the tree.

data.exampleMutTrees

edge.length A numeric vector of the branch lengths of the connections in edge matrix. Units are
mutations.

tip.label A character vector containing the (arbitrary in this case) labels for the 100 tips/samples of
the tree.

Nnode Integer number of internal nodes of the tree

params data.frame containing info on the params used to generate the tree See ape package for
details on class phylo objects.

References

This data set was created for the cloneRate package using coalescent theory approaches described

Examples

# Plot first of 100 trees
ap::plot.phylo(cloneRate::exampleMutTrees[[1]],
  direction = "downwards", show.tip.label = FALSE
)

Description

Set of 100 time-based ultrametric trees reconstructed from the distribution of a sample of n=100
tips. All trees have a net growth rate of 1 with birth rates between 1 and 2 (sampled from a uniform
distribution). Death rates are equal to the chosen birth rate minus 1. Tree reconstruction uses the
exact distribution of coalescence times described in "The coalescent of a sample from a binary
R package ape::rcoal().

Usage

data(exampleUltraTrees)
Format

A list of objects of class phylo

edge A matrix of edge connections which reconstruct the tree.

degree A numeric vector of the branch lengths of the connections in edge matrix. Units are years.

tip.label A character vector containing the (arbitrary in this case) labels for the 100 tips/samples of the tree.

Nnode Integer number of internal nodes of the tree

params data.frame containing info on the params used to generate the tree See ape package for details on class phylo objects.

References

This data set was created for the cloneRate package using coalescent theory approaches described in "The coalescent of a sample from a binary branching process", Lambert A., Theor. Pop. Bio. 2018.

Examples

# Plot first of 100 trees
ape::plot.phylo(cloneRate::exampleUltraTrees[[1]],
    direction = "downwards", show.tip.label = FALSE
)

internalLengths Growth rate estimate using the sum of internal lengths

Description

internalLengths() provides an estimate for the net growth rate of the clone with confidence bounds, using the internal lengths method.

Usage

internalLengths(tree, alpha = 0.05)

Arguments

tree An ultrametric tree subset to include only the clone of interest. Alternatively, a list with several such trees.

alpha Used for calculation of confidence intervals. 1-alpha confidence intervals used with default of alpha = 0.05 (95 percent confidence intervals)
longitudinalData

Value

A dataframe including the net growth rate estimate, the sum of internal lengths and other important
details (clone age estimate, runtime, n, etc.)

See Also

maxLikelihood(), sharedMuts() for other growth rate methods.

Examples

internalLengths(cloneRate::exampleUltraTrees[[1]])

longitudinalData Longitudinal validation data

Description

For three individuals with clonal expansions that can be estimated using our methods, we have
longitudinal data to orthogonally validate these estimates, which is included here. Additionally, for
13 clones with a driver gene matching a driver gene in the single cell data, but without a match to a
specific clone, we include this longitudinal data as well.

Usage

longitudinalData

Format

A data.frame containing all the information needed

Sample.ID  The individual’s ID
Age  Individual’s age at the various sampling times
VAF  The variant allele frequency at the various sampling times for the clone of interest
Gene  Gene or genes with mutation that identifies the clone
Protein  Protein affected by the mutation
cellType  The type of cells used for sequencing
cloneName  The name we use for the clone to match to single cell data, if applicable.

References

These datasets were generated and annotated in: Williams et al. 2022 Fabre et al. 2022

Examples

# Plot longitudinal data from PD9478
library(ggplot2)
ggplot(longitudinalData[longitudinalData$Sample.ID == "PD9478", ] ) +
geom_point(aes(x = Age, y = VAF))
maxLikelihood

**Growth rate estimate using Maximum Likelihood**

**Description**

Uses the approximation that coalescence times $H_i$ are equal to $a+b*U_i$ to find $a$ and $b$. $b$ is equal to $1/r$, where $r$ is the net growth rate.

**Usage**

```r
maxLikelihood(tree, alpha = 0.05)
```

**Arguments**

- `tree`: An ultrametric tree subset to include only the clone of interest. Alternatively, a list with several such trees.
- `alpha`: Used for calculation of confidence intervals. 1-alpha confidence intervals used with default of alpha = 0.05 (95 percent confidence intervals)

**Value**

A dataframe including the net growth rate estimate, confidence intervals, and other important details (clone age estimate, runtime, n, etc.)

**See Also**

`internalLengths` which uses an alternative method for growth rate estimation from an ultrametric tree.

**Examples**

```r
df <- maxLikelihood(cloneRate::exampleUltraTrees[[1]])
```

---

realCloneData

**Real clone data from human blood**

**Description**

42 clones (39 distinct) from 32 individual donors, 13 of whom have a diagnosis of Myeloproliferative Neoplasm

**Usage**

```r
data(realCloneData)
```
**sharedMuts**

**Format**

A list of containing one list with the full ultrametric trees from 30 of the 32 individual donors (the two from Van Egeren are not included), and one list containing the 42 clone trees. In three cases, there are two timepoints from the same clone, and these are separate phylo objects. Each list contains a tree as a class phylo object. See ape package documentation for details on class phylo objects. Names of each phylo object (tree) in the list matches the naming used in the sources and also includes driver, age, and clone number.

**References**

These datasets were generated and annotated in: Williams et al. 2022 Mitchell et al. 2022 Fabre et al. 2022 Van Egeren et al. 2021

**Examples**

```r
# Plot full reconstructed tree from donor PD34493
ape::plot.phylo(cloneRate::realCloneData[["fullTrees"]][["PD34493"]],
               direction = "downwards", show.tip.label = FALSE
```

**sharedMuts**

*Growth rate estimate using the sum of shared mutations assuming a mutation tree*

**Description**

sharedMuts() provides an estimate for the net growth rate of the clone with confidence bounds, using the shared mutations method.

**Usage**

```r
sharedMuts(tree, nu = NULL, alpha = 0.05)
```

**Arguments**

- **tree**
  - A non-ultrametric ape tree subset to include only the clone of interest
- **nu**
  - The mutation rate. If none given, sharedMuts() will first look for a nu column in a metadata data.frame of the tree, and then look for a nu in the tree itself. Will throw error if no nu given or found.
- **alpha**
  - Used for calculation of confidence intervals. 1-alpha confidence intervals used with default of alpha = 0.05 (95 percent confidence intervals)

**Value**

A dataframe including the net growth rate estimate, the sum of internal lengths and other important details (clone age estimate, runtime, n, etc.)
simMut

Simulate mutation-based birth and death branching trees

Description

Generates a sampled tree (or a list of many sampled trees) from a supercritical (birth rate > death rate) birth and death branching process according to the coalescent point process described in "Lambert, A. The coalescent of a sample from a binary branching process. (2018)." Edge lengths will be in units of mutations, assuming poissonian mutation accumulation. Essentially a wrapper combining simUltra() and ultra2mut() functions into one step.

Usage

```r
simMut(
  a,  # Birth rate
  b,  # Death rate
  cloneAge,  # Clone age. Make sure it's same time units as birth and death rates
  n,  # Number of samples/tips of the tree to be returned
  nu,  # Mutation rate in units of mutations per unit time. Make sure time units are consistent with birth and death rates and cloneAge
  nTrees = 1,  # Integer indicating the number of trees to generate. Default is 1.
  precBits = 1000,  # Rmpfr param for handling high precision numbers. Needed to draw coalescence times.
  addStem = FALSE,  # Boolean indicating whether to add stem to tree preceding first split/coalescence
  nCores = 1  # Integer indicating the number of cores to use if parallel pkg is installed. Default is 1.
)
```

Arguments

- `a`: Birth rate
- `b`: Death rate
- `cloneAge`: Clone age. Make sure it's same time units as birth and death rates
- `n`: Number of samples/tips of the tree to be returned
- `nu`: Mutation rate in units of mutations per unit time. Make sure time units are consistent with birth and death rates and `cloneAge`
- `nTrees`: Integer indicating the number of trees to generate. Default is 1.
- `precBits`: Rmpfr param for handling high precision numbers. Needed to draw coalescence times.
- `addStem`: Boolean indicating whether to add stem to tree preceding first split/coalescence
- `nCores`: Integer indicating the number of cores to use if parallel pkg is installed. Default is 1.

See Also

- `internalLengths()` which is the ultrametric/time-based analogue

Examples

```r
sharedMuts(cloneRate::exampleMutTrees[[1]])
```

Value

An ape object of class "phylo" representing the ultrametric phylogenetic tree with edge lengths in units of time. Tree metadata is located in the 'metadata' data.frame included in each "phylo" object. If 'nTrees' param is greater than 1, simUltra returns a list of objects of such objects of class "phylo".

Examples

# Generate a single mutation-based tree with a specified mutation rate
tree <- simMut(a = 1, b = 0.5, cloneAge = 40, n = 50, nu = 10)

# Generate a list of mutation-based trees with a range of mutation rates
tree_list <- simMut(
a = 1, b = 0.5, cloneAge = 40, n = 50,
     nu = stats::runif(n = 3, min = 10, max = 20), nTrees = 3
)

---

**simUltra**  
*
Simulate ultrametric birth and death branching trees

**Description**

Generates a sampled tree (or a list of many sampled trees) from a supercritical (birth rate > death rate) birth and death branching process according to the coalescent point process described in "Lambert, A. The coalescent of a sample from a binary branching process. (2018)."

**Usage**

```
simUltra(
a,  
b,  
cloneAge,  
n,  
nTrees = 1,  
precBits = 1000,  
addStem = FALSE,  
nCores = 1  
)
```

**Arguments**

- **a**: Birth rate or vector of birth rates of length 'nTrees'
- **b**: Death rate or vector of death rates of length 'nTrees'
- **cloneAge**: Clone age or vector of clone ages of length 'nTrees'. Make sure it's same time units as birth and death rates
- **n**: Number of samples/tips of the tree to be returned. Can be a vector of length 'nTrees' as well.
Value

An ape object of class "phylo" representing the ultrametric phylogenetic tree with edge lengths in units of time. Tree metadata is located in the 'metadata' data.frame included in each "phylo" object. If 'nTrees' param is greater than 1, simUltra returns a list of objects of such objects of class "phylo".

Examples

# Generate a single tree
tree <- simUltra(a = 1, b = 0.5, cloneAge = 20, n = 50)

# Generate a list of trees
tree_list <- simUltra(a = 1, b = 0.5, cloneAge = 20, n = 50, nTrees = 3)

---

siteFrequency

Get site frequency spectrum of a tree

Description

siteFrequency() calculates the site frequency in units of time or mutations, as well as a normalized frequency.

Usage

siteFrequency(tree, includeStem = FALSE)

Arguments

tree An ultrametric or mutation-based tree subset to include only the clone of interest. Alternatively, a list with several such trees.

includeStem Boolean indicating whether we should count the stem of the tree as contributing to the site frequency distribution. Default is FALSE.

Value

A data.frame with three columns: the number of descendant cells, site frequency in units of time or mutations, and normalized site frequency. If a list of trees is input, output will be a list of such data.frames.
ultra2mut

See Also

`internalLengths()` and `sharedMuts()` which use the sum of edge lengths ancestral to between 2 and n-1 tips to calculate a growth rate.

Examples

```r
# Get site frequency of a single tree
example.df <- siteFrequency(exampleUltraTrees[[1]])

# Get site frequency of a list of trees
example.list <- siteFrequency(exampleMutTrees)
```

---

ultra2mut

*Add poissonian mutations to an ultrametric tree(s)*

Description

Takes an ultrametric tree of class "phylo" (or a list of such trees) and draws new edge lengths in units of mutations, with the mean of each new edge length equal to the old edge length multiplied by the mutation rate. Mutation rate can be set or drawn from a uniform distribution.

Usage

```r
ultra2mut(tree, nu)
```

Arguments

- `tree`: A single tree or list of trees of class "phylo", with edge lengths in units of time.
- `nu`: Mutation rate in units of mutations per unit time. Can also be a vector of mutation rates with length equal to the number of input trees. Make sure time units are consistent in nu and tree$edge.length.

Value

An ape object of class "phylo" representing the phylogenetic tree with edge lengths in units of mutations. Value of mutation rate will be added to 'metadata' data.frame of output tree if such a data.frame exists in the input tree. Otherwise, mutation rate value will be added to "phylo" object directly. If input is a list of trees, ultra2mut() will return a list of such "phylo" objects.

Examples

```r
# Convert the time-based, ultrametric example trees into mutation-based trees
mutTrees <- ultra2mut(exampleUltraTrees,
    nu = stats::runif(n = length(exampleUltraTrees), min = 10, max = 20)
)
```
Index

* datasets
  exampleMutTrees, 4
  exampleUltraTrees, 5
  realCloneData, 8
* longitudinal
  longitudinalData, 7

ape::rcoal(), 4, 5

birthDeathMCMC, 3

cloneRate (cloneRate-package), 2
cloneRate-package, 2
could_to_tree, 4

eapleMutTrees, 4
eapleUltraTrees, 5

internalLengths, 3, 6, 8
internalLengths(), 10, 13

longitudinalData, 7

maxLikelihood, 3, 8
maxLikelihood(), 7

realCloneData, 8

sharedMuts, 9
sharedMuts(), 7, 13
simMut, 10
simUltra, 11
siteFrequency, 12

ultra2mut, 13