Package ‘TESS’

October 28, 2015

Type Package
Title Diversification Rate Estimation and Fast Simulation of Reconstructed Phylogenetic Trees under Tree-Wide Time-Heterogeneous Birth-Death Processes Including Mass-Extinction Events
Version 2.1.0
Date 2015-10-23
Author Sebastian Hoehna and Michael R. May
Maintainer Sebastian Hoehna <Sebastian.Hoehna@gmail.com>
Description Simulation of reconstructed phylogenetic trees under tree-wide time-heterogeneous birth-death processes and estimation of diversification parameters under the same model. Speciation and extinction rates can be any function of time and mass-extinction events at specific times can be provided. Trees can be simulated either conditioned on the number of species, the time of the process, or both. Additionally, the likelihood equations are implemented for convenience and can be used for Maximum Likelihood (ML) estimation and Bayesian inference.
License GPL-3
Imports Rcpp (>= 0.11.0)
LinkingTo Rcpp
Suggests knitr
VignetteBuilder knitr
NeedsCompilation yes
Repository CRAN
Date/Publication 2015-10-28 08:44:56

R topics documented:

TESS-package .......................................................... 2
cettiidae .................................................................. 3
conifers ................................................................. 4
**TESS-package**

Diversification rate estimation and fast simulation of reconstructed phylogenetic trees under tree-wide time-heterogeneous birth-death processes including mass-extinction events

**Description**

Simulation of reconstructed phylogenetic trees under tree-wide time-heterogeneous birth-death processes and estimation of parameters under the same model. Speciation and extinction rates can be any function of time and mass-extinction events at specific times can be provided. Trees can be simulated either conditioned on the number of species, the time of the process, or both. Additionally, the likelihood equations are implemented for convenience and can be used for Maximum Likelihood (ML) estimation and Bayesian inference.

**Details**

- **Package:** TESS
- **Type:** Package
- **Version:** 2.1.0
- **Date:** 2015-10-23
- **License:** GPL-3
- **LazyLoad:** yes
**Author(s)**
Sebastian Hoehna and Michael R. May
Maintainer: Sebastian Hoehna <Sebastian.Hoehna@gmail.com>

**References**


S. Hoehna: The time-dependent reconstructed evolutionary process with a key-role for mass-extinction events. 2015, Journal of Theoretical Biology, 380, 321-331.


MR May, S. Hoehna, and BR Moore: A Bayesian approach for detecting mass-extinction events when rates of lineage diversification vary. 2015, Systematic Biology

**See Also**
ape coda

---

**Description**

This phylogeny describes the species relationship and divergence times of the bird family Cettiidae, published in Alstroem et al. (2011).

**Usage**
data(cettiidae)

**Format**
The phylogeny is stored as an object of class "phylo". The structure is described in the help page of the function read.tree of the package ape.

**Source**
Examples

# load the data
data(conifers)

# save the old plotting settings
op <- par()

# set the new plotting settings
par(cex = 0.3)

# plot the phylogeny
plot(conifers)

# restore the plotting settings
par(op)

---

Conifer phylogeny from Leslie et al. (2012)

Description

This phylogeny describes the species relationships and divergence times of the plant order Conifera, published in Leslie et al. (2012).

Usage

data(conifers)

Format

The phylogeny is stored as an object of class "phylo". The structure is described in the help page of the function `read.tree` of the package `ape`.

Source


Examples

# load the tree
data(conifers)

# save the settings of the plotting device
op <- par()

# set the line width for plotting the branches
par(cex = 0.3)
# plot the phylogenetic tree
plot(conifers)

# restore the settings of the device
par(op)

mammalia

Dated family level mammalian phylogeny from Meredith et al. (2011): Impacts of the cretaceous terrestrial revolution and kpg extinction on mammal diversification.

Description
This phylogeny describes the species relationship and divergence times of the class Mammalia with 1-3 species included per family, published in Meredith et al. (2011).

Usage
data(mammalia)

Format
The phylogeny is stored as an object of class "phylo". The structure is described in the help page of the function read.tree of the package ape.

Source

Examples
# load the data
data(mammalia)

# safe the current settings of the plotting device
op <- par()

# set the line width for drawing thinner lines for the branches
par(cex = 0.3)

# plot the mammalian phylogeny
plot(mammalia)

# restore the settings of the device
par(op)
tess.analysis: Diversification rate estimation under an episodic birth-death process including mass-extinction events.

Description

tess.analysis estimates diversification rates under an episodic birth-death process including mass-extinction events. The method uses a reversible-jump MCMC algorithm to estimate the number, timing and magnitude of rate-shifts and mass-extinction events. It is possible to fix either number of events and provide specific values that will be used. We assume a Poison process for the number of events and a lognormal distribution with fixed, but specified, hyper-parameters for the speciation and extinction rate; and an independent Poison process for the number of mass-extinction events where each survival probability follows a Beta distribution with fixed hyper-parameters.

The MCMC algorithm can be run either for a specified number of iterations, until a time limit in seconds has been reached, or until the effective sample size (ESS) has reached a given threshold. Once the first of these requirements are met TESS will stop the analysis. Internally we use scaling and sliding proposals to change the parameter values during the MCMC and a birth-move and death-move to add/remove events (rate-shifts or mass-extinction events).

The results of the MCMC run are stored within a directory that is specified by the user. Several files will be generated containing the sampled parameter values. To summarize the output see tess.process.output(...) and tess.plot.output(...).

Usage

tess.analysis( tree,
  initialSpeciationRate,
  initialExtinctionRate,
  empiricalHyperPriors = TRUE,
  empiricalHyperPriorInflation = 10.0,
  empiricalHyperPriorForm = c("lognormal","normal","gamma"),
  speciationRatePriorMean = 0.0,
  speciationRatePriorStdDev = 1.0,
  extinctionRatePriorMean = 0.0,
  extinctionRatePriorStdDev = 1.0,
  initialSpeciationRateChangeTime = c(),
  initialExtinctionRateChangeTime = c(),
  estimateNumberRateChanges = TRUE,
  numExpectedRateChanges = 2,
  samplingProbability = 1,
  missingSpecies = c(),
  timesMissingSpecies = c(),
  tInitialMassExtinction = c(),
  pInitialMassExtinction = c(),
  pMassExtinctionPriorShape1 = 5,
  pMassExtinctionPriorShape2 = 95,
  estimateMassExtinctionTimes = TRUE,
numExpectedMassExtinctions = 2,
estimateNumberMassExtinctions = TRUE,
MRCA = TRUE,
CONDITION = "survival",
BURNIN = 10000,
MAX_ITERATIONS = 200000,
THINNING = 100,
OPTIMIZATION_FREQUENCY = 500,
CONVERGENCE_FREQUENCY = 1000,
MAX_TIME = Inf, MIN_ESS = 500,
ADAPTIVE = TRUE,
dir = "",
priorOnly = FALSE,
verbose = TRUE)

Arguments

tree
The tree in 'phylo' format.

initialSpeciationRate
The initial value of the speciation rate when the MCMC is started. This can either be a single number or a vector of rates per interval.

initialExtinctionRate
The initial value of the extinction rate when the MCMC is started. This can either be a single number or a vector of rates per interval.

empiricalHyperPriors
Should we estimate the hyper-parameters empirically?

empiricalHyperPriorInflation
The scaling factor of the variance for the empirical hyperpriors.

empiricalHyperPriorForm
The possible empirical hyper prior distributions; either lognormal, normal or gamma

speciationRatePriorMean
The mean of the log-normal prior distribution for the speciation rate.

speciationRatePriorStDev
The standard deviation of the log-normal prior distribution for the speciation rate.

extinctionRatePriorMean
The mean of the log-normal prior distribution for the extinction rate.

extinctionRatePriorStDev
The standard deviation of the log-normal prior distribution for the extinction rate.

initialSpeciationRateChangeTime
The initial value of the time points when speciation rate-shifts occur. The number of time-shifts needs to be one smaller than the number of initial speciation rates.
initialExtinctionRateChangeTime
   The initial value of the time points when extinction rate-shifts occur. The number of time-shifts needs to be one smaller than the number of initial extinction rates.

estimateNumberOfRateChanges
   Do we estimate the number of rate shifts? Default is true.

numExpectedRateChanges
   Expected number of rate changes which follow a Poisson process. The default gives 0.5 probability on 0 shifts.

samplingProbability
   The extant taxa sampling probability at the present time.

missingSpecies
   The number of species missed which originated in a given time interval (empirical taxon sampling).

timesMissingSpecies
   The times intervals of the missing species (empirical taxon sampling).

tInitialMassExtinction
   The initial value of the vector of times of the mass-extinction events. This is used as initial values for the MCMC.

pInitialMassExtinction
   The initial value of the vector of survival probabilities of the mass-extinction events. This is used as initial values for the MCMC.

pMassExtinctionPriorShape1
   The alpha (first shape) parameter of the Beta prior distribution for the survival probability of a mass-extinction event.

pMassExtinctionPriorShape2
   The beta (second shape) parameter of the Beta prior distribution for the survival probability of a mass-extinction event.

estimateMassExtinctionTimes
   Do we estimate the times of mass-extinction events? Default is true.

numExpectedMassExtinctions
   Expected number of mass-extinction events which follow a Poisson process. The default gives 0.5 probability on 0 events.

estimateNumberOfMassExtinctions
   Do we estimate the number of mass-extinction events? Default is true.

MRCA
   Does the process start with the most recent common ancestor? If not, the tree must have a root edge!

CONDITION
   Do we condition the process on timelsurvivalltaxa?

BURNIN
   The length of the burnin period.

MAX_ITERATIONS
   The maximum number of iteration of the MCMC. The default is 200000.

THINNING
   The frequency how often samples are recorded during the MCMC. The default is every 100 iterations.

OPTIMIZATION_FREQUENCY
   The frequency how often the MCMC moves are optimized. The default is every 500 iterations.
CONVERGENCE_FREQUENCY
The frequency how often we check for convergence? The default is every 1000 iterations.

MAX_TIME
The maximum time the MCMC is allowed to run in seconds. The default is Inf

MIN_ESS
The minimum number of effective samples (ESS) to assume convergence. The default is 500

ADAPTIVE
Do we use auto-tuning of the MCMC moves? The default is TRUE (recommended).

dir
The subdirectory in which the output will be stored. The default is the present directory ("")

priorOnly
Do we sample from the prior only? The default is FALSE

verbose
Do you want detailed output?

Value
There is no return value because all the results are stored into files.

Author(s)
Sebastian Hoehna

References
S. Hoehna: The time-dependent reconstructed evolutionary process with a key-role for mass-extinction events. 2015, Journal of Theoretical Biology, 380, 321-331.


MR May, S. Hoehna, and BR Moore: A Bayesian approach for detecting mass-extinction events when rates of lineage diversification vary. 2015, Systematic Biology

Examples

# we load the conifers as the test data set
data(conifers)

# for the conifers we know what the total number of species is
total <- 630
# thus, we can compute what the sampling fraction is
rho <- (conifers$Nnode+1)/total

# next, we specify the prior mean and standard deviation
# for the speciation and extinction rate
mu_lambda = 0.15
std_lambda = 0.02
mu_mu = 0.09
std_mu = 0.02
# now we can run the entire analysis.
# note that a full analyses should be run much longer

tess.analysis(
    tree=conifers,
    initialSpeciationRate=exp(mu_lambda),
    initialExtinctionRate=exp(mu_mu),
    empiricalHyperPriors = FALSE,
    speciationRatePriorMean = mu_lambda,
    speciationRatePriorStDev = std_lambda,
    extinctionRatePriorMean = mu_mu,
    extinctionRatePriorStDev = std_mu,
    numExpectedRateChanges = 2,
    samplingProbability = rho,
    numExpectedMassExtinctions = 2,
    BURNIN = 100,
    MAX_ITERATIONS = 200,
    THINNING = 10,
    dir = "analysis_conifer"
)

# You may want to look into the vignette for a more detailed description
# of the features for an analysis.
# also have a look at the functions tess.process.output and tess.plot.output

tess.likelihood
tess.likelihood: Probability density of a tree under a tree-wide
time-dependent birth-death process

Description
tess.likelihood computes the probability of a reconstructed phylogenetic tree under time-dependent
diversification rates. The rates may be any positive function of time or a constant. Additionally,
mass-extinction event can be provided and a uniform taxon sampling probability. You have several
options for the start of the process (origin vs MRCA) and the condition of the process (time, survival
or taxa; note that survival and taxa implicitly condition on the time too!). See equation (5) in Hoehna
(2013) for more information. Note that constant rates lead to much faster computations. The
likelihood can be computed for incompletely sampled trees if you give a sampling probability !=
1.0. You have two options for the sampling strategy: uniform|diversified. The detailed description
of these can be found in the references. More information can be obtained in the vignette about how
to apply this likelihood function.

Usage
tess.likelihood(times,
    lambda,
    mu,
    massExtinctionTimes=c(),
    massExtinctionSurvivalProbabilities=c(),
    missingSpecies = c(),
    timesMissingSpecies = c(),

    empiricalHyperPriors = FALSE,
    speciationRatePriorMean = mu_lambda,
    speciationRatePriorStDev = std_lambda,
    extinctionRatePriorMean = mu_mu,
    extinctionRatePriorStDev = std_mu,
    numExpectedRateChanges = 2,
    samplingProbability = rho,
    numExpectedMassExtinctions = 2,
    BURNIN = 100,
    MAX_ITERATIONS = 200,
    THINNING = 10,
    dir = "analysis_conifer"
)
samplingProbability=1.0, samplingStrategy="uniform", MRCA=TRUE, CONDITION="survival", log=TRUE)

**Arguments**

- **times**
  The branching times of the phylogeny.

- **lambda**
  The speciation rate function or constant.

- **mu**
  The extinction rate function or constant.

- **massExtinctionTimes**
  The set of mass-extinction times after the start of the process.

- **massExtinctionSurvivalProbabilities**
  The set of survival probabilities for each speciation event. The set must have the same length as the set of mass-extinction times.

- **missingSpecies**
  The number of species missed which originated in a given time interval (empirical taxon sampling).

- **timesMissingSpecies**
  The times intervals of the missing species (empirical taxon sampling).

- **samplingProbability**
  The probability for a species to be included in the sample.

- **samplingStrategy**
  The strategy how samples were obtained. Options are: uniform|diversified|age.

- **MRCA**
  Does the process start with the most recent common ancestor? If not, the tree must have a root edge!

- **CONDITION**
  do we condition the process on time|survival|taxa?

- **log**
  Should we log-transform the likelihood?

**Value**

Returns the (log) probability of the tree, i.e. the likelihood of the parameters given the tree.

**Author(s)**

Sebastian Hoehna

**References**


S. Hoehna: The time-dependent reconstructed evolutionary process with a key-role for mass-extinction events. 2015, Journal of Theoretical Biology, 380, 321-331.
Examples

```r
# load a test data set
data(cettiidae)

data

# convert the phylogeny into the branching times
times <- as.numeric(branching.times(cettiidae))

# construct speciation and extinction rate function that resemble the rate-shift
# any other function could be used too
l <- function(x) { if (x > 0.5 || x < 0.3) { return(1) } else { return(0.5) } }
e <- function(x) { if (x > 0.5 || x < 0.3) { return(0.95) } else { return(0.5) } }

# now compute the likelihood for the tree
tess.likelihood(times, l, e, MRCA=TRUE, log=TRUE)

# a second approach is the specific episodic birth-death process likelihood function
# we need to give the rates for each episode and the end time of the episodes
# you should see that both are equivalent in this setting
# the function approach is more general but also slower.
tess.likelihood.rateshift(times,
lambda=c(2,1,2),
mu=c(0.95,0.5,0.95),
rateChangeTimesLambda=c(0.3,0.5),
rateChangeTimesMu=c(0.3,0.5),
MRCA=TRUE,
log=TRUE)
```

tess.likelihood.rateshift

*tess.likelihood.rateshift:* Probability density of a tree under a tree-wide time-dependent birth-death-shift process

Description
tess.likelihood.rateshift computes the probability of a reconstructed phylogenetic tree under a rate-shift model. The rates are piecewise constant. Additionally, mass-extinction event can be provided and a uniform taxon sampling probability. You have several options for the start of the process (origin vs MRCA) and the condition of the process (time, survival or taxa; note that survival and taxa implicitly condition on the time too!). See equation (5) in the manuscript for more information. Note that constant rates lead to much faster computations. The likelihood can be computed for incompletely sampled trees. You need to give a sampling probability != 1.0. You have three options for the sampling strategy: uniform|diversified|age. The detailed description of these can be found in the references. More information can be obtained in the vignette about how to apply this likelihood function.
Usage

tess.likelihood.rateshift( times,
   lambda,
   mu,
   rateChangeTimesLambda = c(),
   rateChangeTimesMu = c(),
   massExtinctionTimes = c(),
   massExtinctionSurvivalProbabilities = c(),
   missingSpecies = c(),
   timesMissingSpecies = c(),
   samplingStrategy = "uniform",
   samplingProbability = 1,
   MRCA = TRUE,
   CONDITION = "survival",
   log = TRUE)

Arguments

times The branching times of the tree.
lambda The speciation rate as a vector representing the rate for each time interval.
mu The extinction rate as a vector representing the rate for each time interval.
rateChangeTimesLambda The times of the rate-shifts for the speciation rate.
rateChangeTimesMu The times of the rate-shifts for the extinction rate.
massExtinctionTimes The set of mass-extinction times after the start of the process.
massExtinctionSurvivalProbabilities The set of survival probabilities for each speciation event. The set must have the same length as the set of mass-extinction times.
missingSpecies The number of species missed which originated in a given time interval (empirical taxon sampling).
timesMissingSpecies The times intervals of the missing species (empirical taxon sampling).
samplingStrategy The strategy how samples were obtained. Options are: uniform|diversified|age.
samplingProbability The probability for a species to be included in the sample.
MRCA Does the process start with the most recent common ancestor? If not, the tree must have a root edge!
CONDITION do we condition the process on timesurvivalltaxa?
log should the likelihood be in log-scale?

Value

Returns the (log) probability of the tree, i.e., the likelihood of the parameters given the tree.
Author(s)
Sebastian Hoehna

References
S. Hoehna: The time-dependent reconstructed evolutionary process with a key-role for mass-extinction events. 2015, Journal of Theoretical Biology, 380, 321-331.

Examples

# load a test data set
data(cettiidae)

# convert the phylogeny into the branching times
times <- as.numeric(branching.times(cettiidae))

# construct speciation and extinction rate function that resemble the rate-shift
# any other function could be used too
l <- function(x) { if (x > 0.5 || x < 0.3) { return (1) } else { return (2) } }
e <- function(x) { if (x > 0.5 || x < 0.3) { return (0.95) } else { return (0.5) } }

# now compute the likelihood for the tree
tess.likelihood(times,l,e,MRCA=TRUE,log=TRUE)

# a second approach is the specific episodic birth-death process likelihood function
# we need to give the rates for each episode and the end time of the episodes
# you should see that both are equivalent in this setting
# the function approach is more general but also slower.
tess.likelihood.rateshift(times,
lambda=c(2,1,2),
mu=c(0.95,0.5,0.95),
rateChangeTimesLambda=c(0.3,0.5),
rateChangeTimesMu=c(0.3,0.5),
MRCA=TRUE,
log=TRUE)

Description
tess.mcmc constructs a Markov chain Monte Carlo sampler (MCMC) by implementing a general Metropolis-Hastings algorithm. Any model can be used where the likelihood is known and thus can be passed in as an argument. The parameters have to be continuous. Proposals are taken from a normal distribution centered around the current value. The variance of the new proposed values is initialized with 1 but can be automatically optimized when using the option adaptive = TRUE. The
algorithm creates samples from the posterior probability distribution and returns these a CODA mcmc object. More information can be obtained in the vignette about how to apply this method.

Usage

tess.mcmc(likelihoodFunction, priors, parameters, logTransforms, delta, iterations, burnin=round(iterations/3), thinning=1, adaptive=TRUE, verbose=FALSE)

Arguments

  likelihoodFunction
    The log-likelihood function which will be called internally by likelihoodFunction(parameters).
  priors
    A list of functions of the log-prior-densities of each parameter.
  parameters
    The initial parameter value list.
  logTransforms
    A vector of booleans telling if log-transform for the parameters should be used (e.g. for rates).
  delta
    The variance of new proposed values.
  iterations
    The number of iterations for the MCMC.
  burnin
    The number of iterations to burn before starting the MCMC.
  thinning
    The frequency of taking a sample of the parameters.
  adaptive
    Should we use adaptive MCMC?
  verbose
    Do you want detailed information during the run?

Value

Returns the posterior samples for the parameters.

Author(s)

  Sebastian Hoehna

References


Examples

  # load in a test data set
data(cettiidae)

  # convert the phylogeny into the branching times
times <- as.numeric(branching.times(cettiidae))
# specify a likelihood function that takes in a vector of parameters
likelihood <- function(params) {
  # We use the parameters as diversification rate and turnover rate.
  # Thus we need to transform first
  d <- params[2]

  lnl <- tess.likelihood(times,b,d,samplingProbability=1.0,log=TRUE)
  return(lnl)
}

# specify a the prior functions
prior.diversification <- function(x) { dexp(x,rate=0.1,log=TRUE) }
prior.turnover <- function(x) { dexp(x,rate=0.1,log=TRUE) }
priors <- c(prior.diversification,prior.turnover)

# Note, the number of iterations and the burnin is too small here
# and should be adapted for real analyses
samples <- tess.mcmc( likelihood,
  priors,
  runif(2,0,1),
  logTransforms=c(TRUE,TRUE),
  delta=c(0.1,0.1),
  iterations=100,
  burnin=20)

# now summarize and visualize the results
# plot(samples)
summary(samples)
colMeans(samples)

tess.nTaxa.expected
tess.nTaxa.expected: The expected number of taxa at present of a tree under a global, time-dependent birth-death process ( E[ N(T) ] )

Description
tess.nTaxa.expected computes the expected number of taxa at the present time T (the process start at time s and times increases until the present) under time-dependent. The rates may be any positive function of time or a constant. Additionally, mass-extinction event can be provided and a uniform taxon sampling probability. You have several options for the start of the process (origin vs MRCA). One important feature is that you can compute the expected number of taxa under the reconstructed process, that is, only lineages that survive until the present.

Usage
tess.nTaxa.expected( begin,
tess.nTaxa.expected

```r
  t, 
  end, 
  lambda, 
  mu, 
  massExtinctionTimes=c(), 
  massExtinctionSurvivalProbabilities=c(), 
  samplingProbability=1.0, 
  MRCA=TRUE, 
  reconstructed=FALSE)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>begin</code></td>
<td>The time when the process starts.</td>
</tr>
<tr>
<td><code>t</code></td>
<td>The time at which we want to know the expected number of lineages (could be equal to <code>end</code>).</td>
</tr>
<tr>
<td><code>end</code></td>
<td>The time when the process end (e.g. the present).</td>
</tr>
<tr>
<td><code>lambda</code></td>
<td>The speciation rate function or constant.</td>
</tr>
<tr>
<td><code>mu</code></td>
<td>The extinction rate function or constant.</td>
</tr>
<tr>
<td><code>massExtinctionTimes</code></td>
<td>The set of mass-extinction times after the start of the process.</td>
</tr>
<tr>
<td><code>massExtinctionSurvivalProbabilities</code></td>
<td>The set of survival probabilities for each speciation event. The set must have the same length as the set of mass-extinction times.</td>
</tr>
<tr>
<td><code>samplingProbability</code></td>
<td>The probability for a species to be included in the sample.</td>
</tr>
<tr>
<td><code>MRCA</code></td>
<td>Does the process start with the most recent common ancestor? If not, the tree must have a root edge!</td>
</tr>
<tr>
<td><code>reconstructed</code></td>
<td>Are we computing the expected number of lineage at time <code>t</code> in the reconstructed process?</td>
</tr>
</tbody>
</table>

**Value**

Returns the expected number of taxa.

**Author(s)**

Sebastian Hoehna

**References**

Examples

```r
# create the time-dependent speciation and extinction rate functions
# here we use episodic functions
l <- function(x) { if (x > 0.5 || x < 0.3) { return (1) } else { return (2) } }
e <- function(x) { if (x > 0.5 || x < 0.3) { return (0.95) } else { return (0.5) } }

# now we can compute the expected number of taxa at time t
# note that we compute here the actual diversity at time t
# if you set reconstructed=TRUE, then you get the expected
# number of lineages that will survive until the present
tess.nTaxa.expected(begin=0,t=2,end=5,1,e,MRCA=TRUE)
```

tess.pathSampling  
*tess.pathSampling*: Marginal likelihood estimation via Path-Sampling.

Description  
tess.pathSampling uses a power posterior series and path-sampling to estimate the marginal likelihood of a model. This is a very general implementation of this algorithm which can be applied basically to any model. More information can be obtained in the vignette about how to apply this method.

Usage  
`tess.pathSampling(likelihoodFunction,priorFunction,parameters,logTransforms,
iterations,burnin=round(iterations/3),K=50)`

Arguments  
- **likelihoodFunction**: The log-likelihood function which will be called internally by `likelihoodFunction(parameters)`.  
- **priorFunction**: A list of functions of the log-prior-densities of each parameter.  
- **parameters**: The initial parameter value list.  
- **logTransforms**: A vector of booleans telling if log-transform for the parameters should be used (e.g. for rates).  
- **iterations**: The number of iterations for the MCMC.  
- **burnin**: The number of iterations to burn before starting the MCMC.  
- **K**: The number of stepping stones.

Value  
Returns the posterior samples for the parameters.
Author(s)

Sebastian Hoehna

References


Baele et al., 2012: Improving the accuracy of demographic and molecular clock model comparison while accommodating phylogenetic uncertainty

Baele et al., 2013: Accurate Model Selection of Relaxed Molecular Clocks in Bayesian Phylogenetics

Examples

```r
# load a test data set
data(cettiidae)
# convert the phylogeny into the branching times
times <- as.numeric(branching.times(cettiidae))

# construct a likelihood function taking in a vector of parameters
likelihood <- function(params) {
  # We use the parameters as diversification rate and turnover rate.
  # Thus we need to transform first
  d <- params[2]

  lnl <- tess.likelihood(times,b,d,samplingProbability=1.0,log=TRUE)
  return(lnl)
}

# next, create the prior density functions
prior_diversification <- function(x) { dexp(x,rate=0.1,log=TRUE) }
prior_turnover <- function(x) { dexp(x,rate=0.1,log=TRUE) }
priors <- c(prior_diversification,prior_turnover)

# Note, the number of iterations, the burnin
# and the number of stepping stones is too small here
# and should be adapted for real analyses
marginalLikelihood <- tess.pathSampling(likelihood,
priors,
runif(2,0,1),
c(TRUE,TRUE),
10,
10,
K=4)
```
tess.plot.multichain.diagnostics

tess.plot.multichain.diagnostics: Plotting the mcmc diagnostics of a episodic diversification rate analysis with mass-extinction events.

Description

tess.plot.multichain.diagnostics plots MCMC diagnostics for the output generated by a tess.process.output(...) command. For more examples see the vignette.

Usage

tess.plot.multichain.diagnostics(outputs,
  parameters=c("speciation rates",
                "speciation shift times",
                "extinction rates",
                "extinction shift times",
                "net-diversification rates",
                "relative-extinction rates",
                "mass extinction times"),
  diagnostics="Gelman-Rubin",
  gelman.crit=1.05,
  xlab="million years ago",
  col=NULL,
  xaxt="n",
  yaxt="s",
  pch=19,
  ...
)

Arguments

outputs The processed output for plotting.
parameters Which parameters to diagnose. See details for a complete description.
diagnostics Which diagnostics to use. Currently the only option is "Rubin-Gelman".
gelman.crit The critical value above which a Rubin-Gelman statistic is considered a failure.
xlab The label of the x-axis. By default, millions of years.
col Colors used for printing. Must be of same length as fig.types.
xaxt The type of x-axis to plot. By default, no x-axis is plotted (recommended).
yaxt The type of y-axis to plot.
pch The type of points to draw (if points are drawn).
... Arguments delegated to plot()
Details

This function generates visual summaries of multi-chain MCMC diagnostics for the CoMET analysis in the output object. The argument parameters specifies the aspects of the model to summarize. Valid options are:

- speciation rates: Plots the interval-specific speciation rates.
- speciation shift times: Plots the posterior probability of at least one speciation-rate shift for each interval.
- extinction rates: Plots the interval-specific extinction rates.
- extinction shift times: Plots the posterior probability of at least one extinction-rate shift for each interval.
- net-diversification rates: Plots the interval-specific net-diversification rates.
- relative-extinction rates: Plots the interval-specific relative-extinction rates.
- mass extinction times: Plots the posterior probability of at least one mass-extinction event for each interval.

Author(s)

Michael R. May

Examples

```r
# Load the data, compute the sampling fraction rho
data(conifers)
totalConiferSpecies <- 630
sampledConiferSpecies <- conifers$Nnode+1
rho <- sampledConiferSpecies / totalConiferSpecies

# Run a tess analysis
tess.analysis(tree = conifers,
              initialSpeciationRate=c(1.0),
              initialExtinctionRate=c(0.5),
              empiricalHyperPriors = FALSE,
              numExpectedRateChanges = 2,
              numExpectedMassExtinctions = 2,
              samplingProbability = rho,
              MAX ITERATIONS = 200,
              BURNIN = 100,
              dir = "/run_1")

tess.analysis(tree = conifers,
              initialSpeciationRate=c(1.0),
              initialExtinctionRate=c(0.5),
              empiricalHyperPriors = FALSE,
              numExpectedRateChanges = 2,
              numExpectedMassExtinctions = 2,
              samplingProbability = rho,
              MAX ITERATIONS = 200,
              BURNIN = 100,
```
dir = "/run_2"

# Process the output
coniferOutput_1 <- tess.process.output(dir="/run_1",
numExpectedRateChanges=2,
numExpectedMassExtinctions=2)

coniferOutput_2 <- tess.process.output(dir="/run_2",
numExpectedRateChanges=2,
numExpectedMassExtinctions=2)

# Plot the output
outputs <- list(coniferOutput_1, coniferOutput_2)
tess.plot.multichain.diagnostics(outputs)

tess.plot.output  

**tess.plot.output**: Plotting the output of a diversification rate estimation including mass-extinction events.

**Description**

tess.output.summary plots the output generated by a tess.process.output(...) command. More specifically, you can plot the speciation, extinction, diversification and relative extinction rate over time, as well as the probability and Bayes factor for the timing of rate shifts and mass-extinction events. For more examples see the vignette.

**Usage**

tess.plot.output(output,
fig.types=c("speciation rates",
"speciation shift times",
"speciation Bayes factors",
"extinction rates",
"extinction shift times",
"extinction Bayes factors",
"net-diversification rates",
"relative-extinction rates",
"mass extinction times",
"mass extinction Bayes factors"),
xlab="million years ago",
col=NULL,
col.alpha=50,
xaxt="n",
yaxt="s",
pch=19,
plot.tree=FALSE,
...)

tess.plot.output

Arguments

- **output**: The processed output for plotting.
- **fig.types**: Which aspects of the model to visualize. See details for a complete description.
- **xlab**: The label of the x-axis. By default, millions of years.
- **col**: Colors used for printing. Must be of same length as fig.types.
- **col.alpha**: Alpha channel parameter for credible intervals.
- **xaxt**: The type of x-axis to plot. By default, no x-axis is plotted (recommended).
- **yaxt**: The type of y-axis to plot.
- **pch**: The type of points to draw (if points are drawn).
- **plot.tree**: Are we plotting the tree too?
- **...**: Arguments delegated to plot()

Details

This function generates visual summaries of the CoMET analysis in the output object. The argument fig.types specifies the aspects of the model to summarize. Valid options are:

- **speciation rates**: Plots the interval-specific speciation rates.
- **speciation shift times**: Plots the posterior probability of at least one speciation-rate shift for each interval.
- **speciation Bayes factors**: Plots the Bayes factor support for at least one speciation-rate shift for each interval (as 2 ln BF).
- **extinction rates**: Plots the interval-specific extinction rates.
- **extinction shift times**: Plots the posterior probability of at least one extinction-rate shift for each interval.
- **extinction Bayes factors**: Plots the Bayes factor support for at least one extinction-rate shift for each interval (as 2 ln BF).
- **net-diversification rates**: Plots the interval-specific net-diversification rates.
- **relative-extinction rates**: Plots the interval-specific relative-extinction rates.
- **mass extinction times**: Plots the posterior probability of at least one mass-extinction event for each interval.
- **mass extinction Bayes factors**: Plots the Bayes factor support for at least one mass-extinction event for each interval (as 2 ln BF).

Author(s)

Michael R. May
Examples

# Load the data, compute the sampling fraction rho
data(conifers)
totalConiferSpecies <- 630
sampledConiferSpecies <- conifers$Nnode+1
rho <- sampledConiferSpecies / totalConiferSpecies

# Run a tess analysis
tess.analysis(tree = conifers,
    initialSpeciationRate=c(1.0),
    initialExtinctionRate=c(0.5),
    empiricalHyperPriors = FALSE,
    numExpectedRateChanges = 2,
    numExpectedMassExtinctions = 2,
    samplingProbability = rho,
    MAX_ITERATIONS = 200,
    BURNIN = 100)

# Process the output
coniferOutput <- tess.process.output(dir=getwd(),
    numExpectedRateChanges=2,
    numExpectedMassExtinctions=2)

# Plot the output
tess.plot.output(coniferOutput)

tess.plot.singlechain.diagnostics

tess.plot.mcmc.diagnostics: Plotting the single chain mcmc diagnostics of a episodic diversification rate analysis with mass-extinction events.

Description

tess.plot.singlechain.diagnostics plots MCMC diagnostics for the output generated by a tess.process.output(...) command. For more examples see the vignette.

Usage

tess.plot.singlechain.diagnostics(output,
    parameters=c("speciation rates",
                "speciation shift times",
                "extinction rates",
                "extinction shift times",
                "net-diversification rates",
                "relative-extinction rates",
                "mass extinction times"),
    diagnostics=c("ESS","geweke"),
    ess.crit=c(100,200),
geweke.crit=0.05,
correction="bonferroni",
xlab="million years ago",
col=NULL,
xaxt="n",
yaxt="s",
pch=19,
...)

Arguments

output The processed output for plotting.
parameters Which parameters to diagnose. See details for a complete description.
diagnostics Which diagnostics to use. Options are "ESS" and "geweke".
ess.crit Two values which correspond to low ESS threshold and acceptable ESS threshold. Default values are 100 and 200.
geweke.crit The p-value cutoff for Geweke’s diagnostic. Default is the canonical 0.05.
correction What type of multiple-correction method to use. Options are "bonferroni" and "sidak".
xlab The label of the x-axis. By default, millions of years.
col Colors used for printing. Must be of same length as fig.types.
xaxt The type of x-axis to plot. By default, no x-axis is plotted (recommended).
yaxt The type of y-axis to plot.
pch The type of points to draw (if points are drawn).
... Arguments delegated to plot()

Details

This function generates visual summaries of single-chain MCMC diagnostics for the CoMET analysis in the output object. The argument parameters specifies the aspects of the model to summarize. Valid options are:

- speciation rates: Plots the interval-specific speciation rates.
- speciation shift times: Plots the posterior probability of at least one speciation-rate shift for each interval.
- extinction rates: Plots the interval-specific extinction rates.
- extinction shift times: Plots the posterior probability of at least one extinction-rate shift for each interval.
- net-diversification rates: Plots the interval-specific net-diversification rates.
- relative-extinction rates: Plots the interval-specific relative-extinction rates.
- mass extinction times: Plots the posterior probability of at least one mass-extinction event for each interval.
Author(s)

Michael R. May

Examples

# Load the data, compute the sampling fraction rho
data(conifers)
totalConiferSpecies <- 630
sampledConiferSpecies <- conifers$Nnode+1
rho <- sampledConiferSpecies / totalConiferSpecies

# Run a tess analysis
tess.analysis(tree = conifers,
    initialSpeciationRate=c(1.0),
    initialExtinctionRate=c(0.5),
    empiricalHyperPriors = FALSE,
    numExpectedRateChanges = 2,
    numExpectedMassExtinctions = 2,
    samplingProbability = rho,
    MAX_ITERATIONS = 200,
    BURNIN = 100)

# Process the output
crderOutput <- tess.process.output(dir=getwd(),
    numExpectedRateChanges=2,
    numExpectedMassExtinctions=2)

# Plot the output
tess.plot.singlechain.diagnostics(crderOutput)

tess.PosteriorPrediction

Description

tess.PosteriorPrediction calls the simulation function exactly once for each sampled parameter combination. In that way, posterior predictive simulations can be obtained which then in turn can be used to compute summary statistics based on these posterior predictive simulations. For more information see the vignette.

Usage
tess.PosteriorPrediction(simulationFunction,parameters,burnin)
Arguments

simulationFunction
   The simulation function which will be called internally by simulationFunction(parameters).

parameters
   A matrix of parameters where the rows represent samples of parameters and the
column the different parameters.

burnin
   The fraction of samples to be discarded as burnin. This is 0.25 by default.

Value

Returns samples simulated from the posterior predictive distribution.

Author(s)

Sebastian Hoehna

References

S. Hoehna: Fast simulation of reconstructed phylogenies under global, time-dependent birth-death
processes. 2013, Bioinformatics, 29:1367-1374

Examples

# We first run an MCMC to obtain samples from the posterior distribution
# and then simulate the posterior predictive distribution.

data(cettiidae)
times <- as.numeric(branching.times(cettiidae))

# The log-likelihood function
likelihood <- function(params) {
   # We use the parameters as diversification rate and turnover rate.
   # Thus we need to transform first
   d <- params[2]

   lnl <- tess.likelihood(times,b,d,samplingProbability=1.0,log=TRUE)
   return (lnl)
}

prior_diversification <- function(x) { dexp(x,rate=0.1,log=TRUE) }
prior_turnover <- function(x) { dexp(x,rate=0.1,log=TRUE) }
priors <- c(prior_diversification,prior_turnover)

# Note, the number of iterations and the burnin is too small here
# and should be adapted for real analyses
samples <- tess.mcmc(likelihood,priors,c(1,0.1),c(TRUE,TRUE),c(0.1,0.1),10,10)

tmrca <- max(branching.times(cettiidae))
# The simulation function
sim <- function(params) {
}
# We use the parameters as diversification rate and turnover rate.
# Thus we need to transform first
d <- params[2]

tree <- tess.sim.age(n=1, age=tmrca,b,d,samplingProbability=1.0)[[1]]
return (tree)
}
trees <- tess.PosteriorPrediction(sim,samples)

# compute the posterior predictive test statistic
ppt <- tess.PosteriorPredictiveTest(trees,cettiidae,gammaStat)
# get the p-value of the observed test-statistic
ppt[[2]]

---

tess.PosteriorPredictiveTest

tess.PosteriorPredictiveTest: Approximation of the posterior predictive distribution.

Description
tess.PosteriorPredictiveTest computes the values of the statistic for the posterior predictive simulations and computes the p-value for the observed statistic.

Usage
tess.PosteriorPredictiveTest(samples,observation,statistic)

Arguments

- samples: Samples from the posterior predictive distribution.
- observation: The observed value.
- statistic: The function that computes the statistic.

Value

Returns a list of the statistic for each sample.

Author(s)

Sebastian Hoehna

References

Examples

# We first run an MCMC to obtain samples from the posterior distribution
# and then simulate the posterior predictive distribution.

# The bird phylogeny as the test data set
data(cettiidae)
times <- as.numeric(branching.times(cettiidae))

# The log-likelihood function
likelihood <- function(params) {
  # We use the parameters as diversification rate and turnover rate.
  # Thus we need to transform first
  d <- params[2]

  lnl <- tess.likelihood(times, b, d, samplingProbability=1.0, log=TRUE)
  return(lnl)
}
prior_diversification <- function(x) { dexp(x, rate=0.1, log=TRUE) }
prior_turnover <- function(x) { dexp(x, rate=0.1, log=TRUE) }
priors <- c(prior_diversification,prior_turnover)

# Note, the number of iterations and the burnin is too small here
# and should be adapted for real analyses
samples <- tess.mcmc(likelihood,priors,c(1,0.1),c(TRUE,TRUE),c(0.1,0.1),10,10)

tmrca <- max(branching.times(cettiidae))

# The simulation function
sim <- function(params) {
  # We use the parameters as diversification rate and turnover rate.
  # Thus we need to transform first
  d <- params[2]

  # We need trees with at least three tips for the gamma-statistics
  repeat {
    tree <- tess.sim.age(n=1, age=tmrca, b, d, samplingProbability=1.0, MRCA=TRUE)[[1]]
    if (tree$Nnode > 1) break
  }
  return (tree)
}

# simulate trees from the posterior predictive distribution
trees <- tess.PosteriorPrediction(sim, samples)

# compute the posterior predictive test statistic
ppt <- tess.PosteriorPredictiveTest(trees, cettiidae, gammaStat)
# get the p-value of the observed test-statistic
ppt[[2]]
tess.process.output: Summarizing the output of a diversification rate estimation including mass-extinction events. See the tess.analysis function for more information on how such output is generated and the tess.plot.output how the output can be visualized. Also have a look at the vignette for more in detail description and examples.

Description

tess.process.output summarizes the output generated by a tess.analysis(...) run.

Usage

tess.process.output(dir,  
  tree=NULL,  
  numExpectedRateChanges=2,  
  numExpectedMassExtinctions=2,  
  burnin=0.25,  
  numIntervals=100,  
  criticalBayesFactors=c(2,6,10))

Arguments

dir The directory from which the CoMET output will be read.
tree The tree analyzed with CoMET in phylo format. By default, looks for a tree in the target directory.
numExpectedRateChanges The number of expected diversification-rate changes.
numExpectedMassExtinctions The number of expected mass-extinction events.
burnin The fraction of samples that will be discarded as burnin.
numIntervals The number of discrete intervals in which to break the tree.
criticalBayesFactors The Bayes factor thresholds to use to assess significance of events.

Details

The output of a CoMET analysis is stored in a directory with different files containing the MCMC samples from the posterior distribution. For example, the tess.analysis function stores the times and survival probabilities of the mass-extinction events in a file. This function, converts the output by counting the number of events that fall into a given time-bin. This pre-processing of the output simplifies the plotting.
**Value**

This function returns a list with the following elements:

- `posterior`: An object of class `mcmc` that contains the trace of the model’s posterior probability.

- `numSpeciationCategories`: An object of class `mcmc` that contains samples from the posterior distribution of the number of speciation categories (minimum 1, since this includes the initial speciation rate).

- `numExtinctionCategories`: An object of class `mcmc` that contains samples from the posterior distribution of the number of extinction categories (minimum 1, since this includes the initial extinction rate).

- `numMassExtinctions`: An object of class `mcmc` that contains samples from the posterior distribution of the number of mass-extinction events.

- `speciation rates`: An object of class `mcmc` that contains speciation rates sampled from the posterior distribution for each of `numIntervals` discrete time intervals. Rows correspond to samples from the posterior distribution, columns correspond to intervals.

- `speciation change times`: An object of class `mcmc` that contains speciation-rate-change events sampled from the posterior distribution for each of `numIntervals` discrete time intervals. A value of 1 indicates an event was contained in the interval, 0 that no event was contained in the interval. Rows correspond to samples from the posterior distribution, columns correspond to intervals.

- `speciation Bayes factors`: A vector of class `numeric` that contains the Bayes factor support for there being a speciation-rate-change event for each of `numIntervals` discrete time intervals. The `ith` element corresponds to the Bayes factor support for an event in the `ith` interval.

- `speciationRateChangeCriticalPosteriorProbabilities`: A vector of posterior probabilities that correspond to critical Bayes factor thresholds (specified by the argument `criticalBayesFactors`). Element `i` is the posterior probability of a speciation-rate-change event in an interval needed to produce Bayes factor support of `criticalBayesFactors[i]`.

- `extinction rates`: An object of class `mcmc` that contains extinction rates sampled from the posterior distribution for each of `numIntervals` discrete time intervals. Rows correspond to samples from the posterior distribution, columns correspond to intervals.

- `extinction change times`: An object of class `mcmc` that contains extinction-rate-change events sampled from the posterior distribution for each of `numIntervals` discrete time intervals. A value of 1 indicates an event was contained in the interval, 0 that no event was contained in the interval. Rows correspond to samples from the posterior distribution, columns correspond to intervals.
extinction Bayes factors
A vector of class ‘numeric’ that contains the Bayes factor support for there being a extinction-rate-change event for each of numIntervals discrete time intervals. The ith element corresponds to the Bayes factor support for an event in the ith interval.

extinctionRateChangeCriticalPosteriorProbabilities
A vector of posterior probabilities that correspond to critical Bayes factor thresholds (specified by the argument criticalBayesFactors). Element i is the posterior probability of an extinction-rate-change event in an interval needed to produce Bayes factor support of criticalBayesFactors[i].

net-diversification rates
An object of class 'mcmc' that contains net-diversification (speciation - extinction) rates sampled from the posterior distribution for each of numIntervals discrete time intervals. Rows correspond to samples from the posterior distribution, columns correspond to intervals.

relative-extinction rates
An object of class 'mcmc' that contains relative-extinction (extinction / speciation) rates sampled from the posterior distribution for each of numIntervals discrete time intervals. Rows correspond to samples from the posterior distribution, columns correspond to intervals.

mass extinction times
An object of class 'mcmc' that contains mass-extinction events sampled from the posterior distribution for each of numIntervals discrete time intervals. A value of 1 indicates an event was contained in the interval, 0 that no event was contained in the interval. Rows correspond to samples from the posterior distribution, columns correspond to intervals.

mass extinction Bayes factors
A vector of class ‘numeric’ that contains the Bayes factor support for there being a mass-extinction event for each of numIntervals discrete time intervals. The ith element corresponds to the Bayes factor support for an event in the ith interval.

massExtinctionCriticalPosteriorProbabilities
A vector of posterior probabilities that correspond to critical Bayes factor thresholds (specified by the argument criticalBayesFactors). Element i is the posterior probability of a mass-extinction event in an interval needed to produce Bayes factor support of criticalBayesFactors[i].

criticalBayesFactors
The critical Bayes factor values used for the Bayes factor tests (default 2 ln BF = \{2,6,10\}).

tree
The tree analyzed with CoMET (just in case).

intervals
The discrete intervals used to compute the interval-specific parameters.

Author(s)
Michael R. May
tess.sim.age

Examples

# Load the data, compute the sampling fraction rho
data(conifers)
totalConiferSpecies <- 630
sampledConiferSpecies <- conifers$Node+1
rho <- sampledConiferSpecies / totalConiferSpecies

# Run a tess analysis
tess.analysis(tree = conifers,
               initialSpeciationRate=c(1.0),
               initialExtinctionRate=c(0.5),
               empiricalHyperPriors = FALSE,
               numExpectedRateChanges = 2,
               numExpectedMassExtinctions = 2,
               samplingProbability = rho,
               MAX_ITERATIONS = 200,
               BURNIN=100)

# Process the output
coniferOutput <- tess.process.output(dir=getwd(),
                                      numExpectedRateChanges=2,
                                      numExpectedMassExtinctions=2)

# Plot the output
tess.plot.output(coniferOutput)

tess.sim.age
tess.sim.age: Simulate a reconstructed tree for a given age under a global, time-dependent birth-death process.

Description

tess.sim.age simulates a reconstructed phylogenetic tree under a global, time-dependent birth-death process conditioned on the age of the tree. The rates may be any positive function of time or a constant. The process starts at time 0 and goes forward in time, hence the rates and events should be interpreted in the time after the origin. Additionally, mass-extinction event can be provided and a uniform taxon sampling probability. It is possible to start either with the origin (1 species) or with the most recent common ancestor (2 species).

Usage

tess.sim.age(n, age, lambda, mu, massExtinctionTimes = c(),
             massExtinctionSurvivalProbabilities = c(), samplingProbability = 1,
             samplingStrategy = "uniform", maxTaxa = Inf, MRCA = TRUE)

Arguments

n Number of simulations.
The age of the tree, i.e. the time to simulate.

*lambda*  
The speciation rate function or constant.

*mu*  
The extinction rate function or constant.

*massExtinctionTimes*  
The set of mass-extinction times after the start of the process.

*massExtinctionSurvivalProbabilities*  
The set of survival probabilities for each speciation event. The set must have the same length as the set of mass-extinction times.

*samplingProbability*  
The probability for a species to be included in the sample.

*samplingStrategy*  
The strategy how samples were obtained. Options are: uniform|diversified.

*maxTaxa*  
The maximum number of possible taxa. If by chance a higher number is simulated, than simply `ntaxa=maxTaxa`. This is useful when too large trees should be simulated because this takes too much time and memory.

*MRCA*  
Does the process start with the most recent common ancestor?

**Value**

Returns a set of trees in 'phylo' format.

**Author(s)**

Sebastian Hoehna

**References**


**Examples**

```r
l <- function(x) { if (x > 0.5 || x < 0.3) { return(1) } else { return(2) } }

e <- function(x) { if (x > 0.5 || x < 0.3) { return(0.95) } else { return(0.5) } }

tess.sim.age(n=1, age=1, l, e, MRCA=TRUE)

# simulation under constant rates

tess.sim.age(n=1, age=1,2,1,0, MRCA=TRUE)
```
tess.sim.taxa

**Description**

tess.sim.taxa simulates a reconstructed phylogenetic tree under a global, time-dependent birth-death process conditioned on the number of taxa sampled. The rates may be any positive function of time or a constant. The process starts at time 0 and goes forward in time, hence the rates and events should be interpreted in the time after the origin. Additionally, mass-extinction event can be provided and a uniform taxon sampling probability. It is possible to start either with the origin (1 species) or with the most recent common ancestor (2 species).

**Usage**

```r
tess.sim.taxa(n,ntaxa,max,lambda,mu, massExtinctionTimes = c(),
  massExtinctionSurvivalProbabilities = c(), samplingProbability = 1,
  samplingStrategy = "uniform", SURVIVAL = TRUE, MRCA = TRUE, t_crit = c())
```

**Arguments**

- **n** Number of simulations.
- **ntaxa** Number of species sampled.
- **max** Maximum time/height of the tree.
- **lambda** The speciation rate function or constant.
- **mu** The extinction rate function or constant.
- **massExtinctionTimes** The set of mass-extinction times after the start of the process.
- **massExtinctionSurvivalProbabilities** The set of survival probabilities for each speciation event. The set must have the same length as the set of mass-extinction times.
- **samplingProbability** The probability for a species to be included in the sample.
- **samplingStrategy** The strategy how samples were obtained. Options are: uniform|diversified.
- **SURVIVAL** Do you want to condition on survival of the process?
- **MRCA** Does the process start with the most recent common ancestor?
- **t_crit** The critical time points when a jump in the rate function occurs. Only a help for the numerical integration routine.

**Value**

Returns a tree in ‘phylo’ format.
**Author(s)**

Sebastian Hoehna

**References**


**Examples**

\[
\begin{align*}
1 & \leftarrow \text{function}(x) \{ \text{if} \ (x > 0.5 \ | | x < 0.3) \ {\text{return} (1)} \ \text{else} \ {\text{return} (2)} \} \\
\text{e} & \leftarrow \text{function}(x) \{ \text{if} \ (x > 0.5 \ | | x < 0.3) \ {\text{return} (0.95)} \ \text{else} \ {\text{return} (0.5)} \}
\end{align*}
\]

```r
\text{tess.sim.taxa(age)}(n=1, nTaxa=10, max=10, l, e, MRCA=TRUE)

# simulation under constant rates
\text{tess.sim.taxa(n=1, nTaxa=10, max=10, 2.0, 1.0, MRCA=TRUE)}
```

**tess.sim.taxa.age**

**tess.sim.taxa.age**: Simulate a reconstructed tree for a given age and number of taxa under a global, time-dependent birth-death process.

**Description**

tess.sim.taxa.age simulates a reconstructed phylogenetic tree under a global, time-dependent birth-death process conditioned on the age of the tree and number of taxa sampled. The rates may be any positive function of time or a constant. The process starts at time 0 and goes forward in time, hence the rates and events should be interpreted in the time after the origin. Additionally, mass-extinction event can be provided and a uniform taxon sampling probability. It is possible to start either with the origin (1 species) or with the most recent common ancestor (2 species).

**Usage**

tess.sim.taxa.age(n, nTaxa, age, lambda, mu, massExtinctionTimes = c(),
massExtinctionSurvivalProbabilities = c(), samplingProbability = 1,
samplingStrategy = "uniform", MRCA = TRUE)

**Arguments**

- **n**: Number of simulations.
- **nTaxa**: Number of species sampled.
- **age**: The age of the tree, i.e. the time to simulate.
- **lambda**: The speciation rate function or constant.
- **mu**: The extinction rate function or constant.
**tess.steppingStoneSampling**

massExtinctionTimes
The set of mass-extinction times after the start of the process.

massExtinctionSurvivalProbabilities
The set of survival probabilities for each speciation event. The set must have the same length as the set of mass-extinction times.

samplingProbability
The probability for a species to be included in the sample.

samplingStrategy
The strategy how samples were obtained. Options are: uniform|diversified.

MRCA
Does the process start with the most recent common ancestor?

**Value**
Returns a tree in 'phylo' format.

**Author(s)**
Sebastian Hoehna

**References**

**Examples**

```r
l <- function(x) { if (x > 0.5 || x < 0.3) { return(1) } else { return(2) } }
e <- function(x) { if (x > 0.5 || x < 0.3) { return(0.95) } else { return(0.5) } }
tess.sim.taxa.age(n=1,1,e,nTaxa=10,age=1,MRCA=TRUE)
# simulation under constant rates
  tess.sim.taxa.age(n=1,2.0,1.0,nTaxa=10,age=1,MRCA=TRUE)
```

**tess.steppingStoneSampling**


**Description**
tess.steppingStoneSampling uses a power posterior series and stepping-stone-sampling to estimate the marginal likelihood of a model.
Usage
tess.steppingStoneSampling(likelihoodFunction, priors, parameters, logTransforms, iterations, burnin=round(iterations/3), K=50)

Arguments

likelihoodFunction
The log-likelihood function which will be called internally by likelihoodFunction(parameters).
priors
A list of functions of the log-prior-densities of each parameter.
parameters
The initial parameter value list.
logTransforms
A vector of booleans telling if log-transform for the parameters should be used (e.g. for rates).
iterations
The number of iterations for the MCMC.
burnin
The number of iterations to burn before starting the MCMC.
K
The number of stepping stones.

Value
Returns the posterior samples for the parameters.

Author(s)
Sebastian Hoehna

References
Xie et al., 2011: Improving marginal likelihood estimation for Bayesian phylogenetic model selection Baele et al., 2012: Improving the accuracy of demographic and molecular clock model comparison while accommodating phylogenetic uncertainty Baele et al., 2013: Accurate Model Selection of Relaxed Molecular Clocks in Bayesian Phylogenetics

Examples

data(cettiidae)
times <- as.numeric( branching.times(cettiidae) )

likelihood <- function(params) {
  # We use the parameters as diversification rate and turnover rate.
  # Thus we need to transform first
  d <- params[2]

  lnl <- tess.likelihood(times, b, d, samplingProbability=1.0, log=TRUE)
  return (lnl)
}
prior_diversification <- function(x) { dexp(x, rate=0.1, log=TRUE) }
prior_turnover <- function(x) { dexp(x, rate=0.1, log=TRUE) }
priors <- c(prior_diversification, prior_turnover)

# Note, the number of iterations, the burnin
# and the number of stepping stones is too small here
# and should be adapted for real analyses
marginalLikelihood <- tess.steppingStoneSampling( likelihood,
  priors,
  runif(2,0,1),
  c(TRUE,TRUE),
  10,
  10,
  K=4)
Index

*Topic datagen
  tess.sim.age, 33
  tess.sim.taxa, 35
  tess.sim.taxa.age, 36

*Topic datasets
  cettiidae, 3
  conifers, 4
  mammalia, 5

*Topic htest
  tess.analysis, 6
  tess.likelihood, 10
  tess.likelihood.rateshift, 12
  tess.mcmc, 14
  tess.plot.pathSampling, 18
  tess.plot.multichain.diagnostics, 20
  tess.plot.output, 22
  tess.plot.singlechain.diagnostics, 24
  tess.PosteriorPrediction, 26
  tess.PosteriorPredictiveTest, 28
  tess.process.output, 30
  tess.steppingStoneSampling, 37

*Topic models
  tess.analysis, 6
  tess.likelihood, 10
  tess.likelihood.rateshift, 12
  tess.nTaxa.expected, 16
  tess.plot.multichain.diagnostics, 20
  tess.plot.output, 22
  tess.plot.singlechain.diagnostics, 24
  tess.process.output, 30
  tess.sim.age, 33
  tess.sim.taxa, 35
  tess.sim.taxa.age, 36
  tess.steppingStoneSampling, 37
  globalBiDe.analysis(tess.analysis), 6
  globalBiDe.output.summary(tess.process.output), 30
  mammalia, 5
  read.tree, 3–5
  TESS (TESS-package), 2
  TESS-package, 2
  tess.analysis, 6
  tess.likelihood, 10
  tess.likelihood.rateshift, 12
  tess.mcmc, 14
  tess.nTaxa.expected, 16
  tess.plot.pathSampling, 18
  tess.plot.multichain.diagnostics, 20
  tess.plot.output, 22
  tess.plot.singlechain.diagnostics, 24
  tess.PosteriorPrediction, 26
  tess.PosteriorPredictiveTest, 28
  tess.process.output, 30
  tess.sim.age, 33
  tess.sim.taxa, 35
  tess.sim.taxa.age, 36
  tess.steppingStoneSampling, 37

ape, 3–5

cettiidae, 3
conifers, 4