Package ‘TESS’

February 19, 2015

Type Package

Title Fast simulation of reconstructed phylogenetic trees under
time-dependent birth-death processes

Version 1.2.1

Date 2013-03-27

Author Sebastian Hoehna

Maintainer Sebastian Hoehna <hoehna@math.nus.se>

Depends ape, coda, deSolve

Description TESS is a package for simulation of reconstructed
phylogenetic trees under global, time-dependent birth-death
processes. 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

LazyLoad yes

NeedsCompilation no

Repository CRAN

Date/Publication 2013-04-22 18:06:47

R topics documented:

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>TESS-package</td>
<td>2</td>
</tr>
<tr>
<td>cettiidae</td>
<td>3</td>
</tr>
<tr>
<td>globalBiDe.likelihood</td>
<td>3</td>
</tr>
<tr>
<td>globalBiDe.nTaxa.expected</td>
<td>5</td>
</tr>
<tr>
<td>mammalia</td>
<td>6</td>
</tr>
<tr>
<td>sim.globalBiDe.age</td>
<td>7</td>
</tr>
<tr>
<td>sim.globalBiDe.taxa</td>
<td>8</td>
</tr>
</tbody>
</table>
TESS-package

Sim.globalBiDe.taxa.age ......................................................... 10
tess.mcmc ................................................................. 11
tess PosteriorPrediction .................................................... 13
tess PosteriorPredictiveTest ................................................ 14
tess steppingStoneSampling ................................................. 16

Index ................................................. 18

Package: TESS
Type: Package
Version: 1.2.1
Date: 2013-03-27
License: GPL-3
Lazy Load: yes

Description

TESS is a package for simulation of reconstructed phylogenetic trees under global, time-dependent birth-death processes. 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

Author(s)

Sebastian Hoehna
Maintainer: Sebastian Hoehna <hoehna@math.su.se>

References


See Also

ape
Cettiidae phylogeny from Alstroem et al. (2011)

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" which structure is described in the help page of the function `read.tree` of the package `ape`.

Source


Examples

data(cettiidae)
op <- par()
par(cex = 0.3)
plot(cettiidae)
par(op)

---

globalBiDe.likelihood

globalBiDe.likelihood: Probability density of a tree under a global, time-dependent birth-death process

Description

globalBiDe.likelihood computes the probability of a reconstructed phylogenetic tree under time-dependent. The rates may be any positive function of time or a constant. Additionally, mass-extinction event can be provided and a random 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: random/diversified/age. The detailed description of these can be found in the references.
Usage

```
globalBiDe.likelihood(tree, lambda, mu, massExtinctionTimes = c(),
massExtinctionSurvivalProbabilities = c(), t.crit = c(),
samplingProbability = 1, samplingStrategy = "random", MRCA = TRUE,
CONDITION = "survival", log = TRUE)
```

Arguments

tree The tree in 'phylo' format.
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.
t.crit Set of times when a critical change in the rate function happens, such as a jump in a stepwise discrete function.
samplingProbability The probability for a species to be included in the sample.
samplingStrategy The strategy how samples were obtained. Options are: random|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 log-likelihood?

Value

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

Author(s)

Sebastian Hoehna

References

Examples

data(cettiidae)

1 <- 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) } }

globalBiDe.likelihood(cettiidae,1,e,MRCA=TRUE,log=TRUE)

# constant rate likelihood function
globalBiDe.likelihood(cettiidae,2.0,1.0,MRCA=TRUE,log=TRUE)

# constant rate pure birth likelihood function
globalBiDe.likelihood(cettiidae,2.0,0.0,MRCA=TRUE,log=TRUE)

---

globalBiDe.nTaxa.expected

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

Description

globalBiDe.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 random taxon sampling probability. You have several options for the start of the process (origin vs MRCA). See equation (??) in the manuscript for more information. Note that constant rates (scalars) lead to much faster computations.

Usage

globalBiDe.nTaxa.expected(t_low,t_high,lambda,mu,massExtinctionTimes=c(),
massExtinctionSurvivalProbabilities=c(),samplingProbability=1.0,MRCA=TRUE)

Arguments

t_low The time when the process starts.
t_high The time when the process end (e.g. the present).
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.
Sampling Probability

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!

Value

Returns the expected number of taxa.

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) } }
globalBiDe.nTaxa.expected(0.5,l,e,MRCA=TRUE)
```

---

**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" which structure is described in the help page of the function `read.tree` of the package `ape`. 
**Source**


**Examples**

```r
data(mammalia)
op <- par()
par(cex = 0.3)
plot(mammalia)
par(op)
```

**Description**

`sim.globalBiDe.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 random 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
sim.globalBiDe.age(n, age, lambda, mu, massExtinctionTimes = c(),
                   massExtinctionSurvivalProbabilities = c(), samplingProbability = 1,
                   samplingStrategy = "random", MRCA = TRUE)
```

**Arguments**

- `n` Number of simulations.
- `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.
- `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: random|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) } }

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

# simulation under constant rates
sim.globalBiDe.age(n=1, age=2, 0, 1, 0, MRCA=TRUE)
```

---

**sim.globalBiDe.taxa**

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

**Description**

sim.globalBiDe.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 random 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
sim.globalBiDe.taxa(n, nTaxa, max, lambda, mu, massExtinctionTimes = c(),
                    massExtinctionSurvivalProbabilities = c(), samplingProbability = 1,
                    samplingStrategy = "random", 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: random|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

```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) } }

sim.globalBiDe.taxa(n=1, nTaxa=10, max=10, l, e, MRCA=TRUE)

# simulation under constant rates
sim.globalBiDe.taxa(n=1, nTaxa=10, max=10, l, e, MRCA=TRUE)
```
Description

_sim.globalBiDe.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 random 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
sim.globalBiDe.taxa.age(n, nTaxa, age, lambda, mu, massExtinctionTimes = c(), massExtinctionSurvivalProbabilities = c(), samplingProbability = 1, samplingStrategy = "random", 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.
- `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: random|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

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) } }
sim.globalBiDe.taxa.age(n=1,1,e,nTaxa=10,age=1,MRCA=TRUE)

# simulation under constant rates
sim.globalBiDe.taxa.age(n=1,2,0.1,0,nTaxa=10,age=1,MRCA=TRUE)

tess.mcmc

tess.mcmc: Metropolis-Hastings algorithm.

description

tess.mcmc constructs a Markov chain Monte Carlo algorithm (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.

Usage

tess.mcmc(likelihoodfunction,priors,parameters,logtransforms,delta, iterations,burnin=round(iterations/3),thining=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.
tess.mcmc

burnin: The number of iterations to burn before starting the MCMC.
thin: 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

data(cettiidae)

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

  lnl <- globalBiDe.likelihood(cettiidae,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,runif(2,0,1),logTransforms=c(TRUE,TRUE),delta=c(0.1,0.1),iterations=100,plot=samples)
sample(summary(samples))
colMeans(samples)
tess.PosteriorPrediction

**tess.PosteriorPrediction: Approximation of the posterior predictive distribution.**

**Description**

`tess.PosteriorPrediction` calls the simulation function exactly once for each sampled parameter combination. In that way, posterior predictive simulations can be obtained.

**Usage**

`tess.PosteriorPrediction(simulationFunction, parameters)`

**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.

**Value**

Returns samples simulated from the posterior predictive distribution.

**Author(s)**

Sebastian Hoehna

**References**


**Examples**

```r
# 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)

# 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 <- globalBiDe.likelihood(cettiidae, b, d, samplingProbability=1.0, log=TRUE)
}
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)

# 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 <- globalBiDe.likelihood(cettiidae, 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 <- sim.globalBiDe.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.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)

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

  lnl <- globalBiDe.likelihood(cettiidae,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
marginallikelihood <- tess.steppingStoneSampling(likelihood,priors,runif(2,0,1),c(TRUE,TRUE),10,10,K=4)
Index

*Topic **datagen**
  sim.globalBiDe.age, 7
  sim.globalBiDe.taxa, 8
  sim.globalBiDe.taxa.age, 10

*Topic **datasets**
  cettiidae, 3
  mammalia, 6

*Topic **htest**
  globalBiDe.likelihood, 3
  tess.mcmc, 11
  tess.PosteriorPrediction, 13
  tess.PosteriorPredictiveTest, 14
  tess.steppingStoneSampling, 16

*Topic **models**
  globalBiDe.likelihood, 3
  globalBiDe.nTaxa.expected, 5
  ape, 3, 6
  cettiidae, 3
  globalBiDe.likelihood, 3
  globalBiDe.nTaxa.expected, 5
  mammalia, 6
  read.tree, 3, 6
  sim.globalBiDe.age, 7
  sim.globalBiDe.taxa, 8
  sim.globalBiDe.taxa.age, 10
  TESS (TESS-package), 2
  TESS-package, 2
  tess.mcmc, 11
  tess.PosteriorPrediction, 13
  tess.PosteriorPredictiveTest, 14
  tess.steppingStoneSampling, 16