Package: bppr (via r-universe)

November 1, 2024

Type Package

Title An R package for BPP

Version 0.6.3

Author Mario dos Reis

Maintainer Mario dos Reis <mariodosreis@gmail.com>

Description Functions to work with the multi-species coalescent program BPP, for example, functions to calibrate BPP trees to geological time.

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.2.0

Repository https://phylotastic.r-universe.dev

RemoteUrl https://github.com/dosreislab/bppr

RemoteRef HEAD

RemoteSha e64568115157c967ccbbe84bde8a50eefd845a63

Contents

bayes.factors *Calculate Bayes factors and posterior model probabilities*

Description

Calculate Bayes factors and posterior model probabilities

Usage

```
bayes.factors(..., prior = NULL, boot = TRUE, n = 1000, prob = 0.95)
```

Arguments

...	list of marginal likelihood objects, see details
prior	numeric, the prior model probabilities
boot	logical, whether to perform parametric boostrap of probabilities
n	numeric, number of bootstrap samples
prob	numeric, the probability used to calculate the boostrap CI

Details

Input is a list of marginal likelihood objects, with each object generated by either `stepping.stones()` or `gauss.quad()`. If `boot = TRUE`, parametric bootstrap is performed by assuming the log-marginal likelihood estimates are normally distributed with standard deviation equal to the standard error. The re-sampled `n` marginal log-likelihoods are used to estimate re-sampled posterior probabilities, and to calculate an equal-tail bootstrap confidence interval for these.

Note that the length of `prior` should be the same as the number of models being compared. The `prior` is rescaled so that `sum(prior) == 1`.

Value

A list with elements `bf`, the Bayes factors; `pr`, the posterior model probabilities; `prior` the prior model probabilities and, if `boot = TRUE`, `pr.ci` the equal-tail bootstrap confidence interval.

Author(s)

Mario dos Reis

Examples

```
# See Table 5 in dos Reis et al. (2018, Syst. Biol., 67: 594-615)
# Bayesian selection of relaxed clock models for the 1st and 2nd sites
# of mitochondrial protein-coding genes of primates
# Models: strick clock, independent-rates, and autocorrelated-rates
sc <- list(); sc$logml <- -16519.03; sc$se <- .01
ir <- list(); ir$logml <- -16480.58; ir$se <- .063
ar <- list(); ar$logml <- -16477.82; ar$se <- .035
```

```
bayes.factors(sc, ir, ar)
bayes.factors(sc, ir, ar, prior=c(.25,.5,.25))
bayes.factors(sc, ir, ar, prior=c(0,1,0))
```

---

gauss.quad *Estimate marginal likelihood by thermodynamic integration*

Description

Estimate marginal likelihood by thermodynamic integration and Gauss-Legendre quadrature from a sample of n power posterior MCMC chains sampled with mcmctree (or bpp).

Usage

```
gauss.quad(mcmcf = "mcmc.txt", betaf = "beta.txt")
```

Arguments

mcmcf	character, mcmc output file name
betaf	character, file with beta values

Details

The MCMC samples should be stored in a directory structure created by make.bfctlf with method = "gauss-quad". The function will read the stored log-likelihood values and calculate the log-marginal likelihood.

Numerical integration is done using Gauss-Legendre quadrature. See Rannala and Yang (2017) for details (also dos Reis et al. 2017, Appendix 2).

Value

A list with components logml, the log-marginal likelihood estimate; se, the standard error of the estimate; mean.logl, the mean of log-likelihood values sampled for each beta; and b, the beta values used.

Author(s)

Mario dos Reis

References

Rannala B and Yang Z. (2017) Efficient Bayesian species tree inference under the multispecies coalescent. *Systematic Biology* 66: 823-842.

dos Reis et al. (2017) Using phylogenomic data to explore the effects of relaxed clocks and calibration strategies on divergence time estimation: Primates as a test case. *bioRxiv*

See Also

make.bfctlf to prepare directories and mcmctree or bpp control files to calculate the power posterior.

---

hominids *A BPP A00 MCMC sample for an hominid phylogeny*

Description

This dataset contains the results from the BPP A00 analysis of hominid evolution from Angelis and dos Reis (2015).

Usage

```
hominids
```

Format

hominids is a list with elements mcmc, a dataframe with 20,000 rows and 8 columns, and tree, an object of class phylo from the ape package.

mcmc is a posterior sample from a BPP A00 MCMC analysis containing the relative divergence times (tau's) and nucleotide diversities (theta's) for the four species ape (hominid) phylogeny.

tree contains the phylogeny with node ages given as the posterior means of the tau's in mcmc.

Source

K. Angelis and M. dos Reis (2015) *The impact of ancestral population size and incomplete lineage sorting on Bayesian estimation of species divergence times.* Curr. Zool., 61: 874–885.

See Also

microcebus

---

make.beta *Make beta values for marginal likelihood calculation*

Description

Make appropriate beta values

Usage

```
make.beta(n, method = c("step-stones", "gauss-quad"), a = 5)
```

Arguments

n	numeric, number of beta points
method	character, the method to choose the beta points, see details
a	numeric, exponent for stepping stones beta generation, see details

Details

If method = "step-stones", the beta values are given by the formula

$$\beta_i = \left(\frac{i-1}{n}\right)^a.$$

Values of a between 5 to 8 appear appropriate. Large a values produce beta values close to zero.

If method = "gauss-quad", the beta values are calculated according to the n Gauss-Legendre quadrature rule (see Rannala and Yang, 2017).

Value

Numeric vector with n beta values

Author(s)

Mario dos Reis

References

Rannala B and Yang Z (2017) Efficient Bayesian species tree inference under the multispecies coalescent. *Systematic Biology*, 66: 823–842.

See Also

The generated beta values are suitable input for make.bfctlf.

---

make.bfctlf — *Prepare mcmctree or bpp control files for marginal likelihood calculation*

Description

Prepare mcmctree or bpp control files for marginal likelihood calculation

Usage

```
make.bfctlf(beta, ctlf = "bpp.ctl", betaf = "beta.txt")
```

Arguments

beta	numeric vector of beta values
ctlf	character, mcmctree or bpp control file template
betaf	character, file onto which to write selected beta values

Details

This function generates a set of n directories each containing a modified ctlf control file with the appropriate beta value to run mcmctree (or bpp) to obtain MCMC samples under the required power-posterior distribution. For the general theory of marginal likelihood calculation with power posteriors see Yang (2014).

The beta values are printed to betaf.

Author(s)

Mario dos Reis

References

Yang Z (2014) *Molecular Evolution: A Statistical Approach*. Oxford University Press. Pages 256–260.

See Also

make.beta, stepping.stones

---

mcmc.summary *MCMC summary*

Description

Create an MCMC summary from a BPP or MCMCTree analysis

Usage

```
mcmc.summary(mcmc, prob = 0.95)
```

Arguments

mcmc	Data frame with the MCMC output of BPP or MCMCTree
prob	Numeric, probability for credibility interval calculation

Details

mcmc should contain the output (say from file mcmc.txt) generated by a BPP A00 or MCMCtree analysis. The function will calculate the posterior (prior) means, the equal-tail credibility interval (CI), and the highest posterior (prior) density (HPD) CI. prob is used to calculate the CIs. For example, if prob = 95 95

Value

A list with elements `means`, `eq.ci`, and `hpd.ci` containing the posterior (or prior) means, equal tail CI and HPD CI.

Author(s)

Mario dos Reis

Examples

```
## Not run:
mcmc.summary(hominids$mcmc[,-1])

## End(Not run)
```

`mcmc2densitree` *Plot a densi-tree from an MCMC sample*

Description

Plot a densi-tree from an MCMC sample from a BPP or MCMCTree analysis

Usage

```
mcmc2densitree(
  tree,
  mcmc,
  time.name,
  thin,
  col = "blue",
  alpha = 1,
  y.offset = 0,
  pfract = 0.1,
  plot.labels = TRUE,
  axis = TRUE,
  add = FALSE,
  tip.ages = NULL
)
```

Arguments

`tree`	an object of class phylo.
`mcmc`	data frame with an MCMC sample from MCMCTree or a BPP A00 analysis.
`time.name`	character vector of length one.
`thin`	numeric, the fraction of MCMC samples to keep.

col	character, the color for branches.
alpha	numeric, between 0 and 1, the branch color transparency.
y.offset	numeric, the vertical offset for plotting the tree.
pfract	numeric, how much of the plotting space to used for plotting the tip labels. If pfrac = 1, the same amount of space is used for the tree and the labels. Use large values if your tip labels are long.
plot.labels	logical, whether to plot the tip labels. Ignored if add = TRUE.
axis	logical, whether to plot the x axis.
add	logical, if TRUE add the trees to an existing plot, otherwise create a new plot.
tip.ages	numeric, the ages of the tips, with the most recent tip having age zero, and the oldest tip having the largest age. If NULL, tips are assumed to have all age zero.

Details

The function will reduce the MCMC sample to `dim(mcmc)[1] * thin` observations. Then the node ages in each observarion are used to plot each tree in the sample. For a tree with `s` species. The y coordinates of the tips are given by `0:(s - 1) + y.offset`.

The `tree` must be rooted, strictly bifurcating, and be the same tree used to genarate the BPP (A00) or MCMCTree MCMC samples.

Author(s)

Mario dos Reis

Examples

```
data(microcebus)
mcmc2densitree(microcebus$tree, microcebus$mcmc, time.name="tau_", thin=0.05,
 alpha=0.01, col="blue")
 title(xlab="Distance (substitutions per site)")

data(hominids)
# Calibrate the hominid phylogeny with a uniform fossil calibration of
# between 6.5 to 10 Ma for the human-chimp divergence, and plot the
# calibrated sample
calmsc <- msc2time.t(mcmc=hominids$mcmc, node="7humanchimp", calf=runif,
  min=6.5, max=10)
mcmc2densitree(hominids$tree, calmsc, "t_", thin=0.05, alpha=0.01)
title(xlab="Divergence time (Ma)")
```

mcmc2multiphylo *Convert an MCMC sample from BPP or MCMCTree to a list of trees*

Description

Convert an MCMC sample from BPP or MCMCTree to a list of trees

Usage

```
mcmc2multiphylo(tree, mcmc, time.name, thin)
```

Arguments

tree	an object of class phylo
mcmc	data frame with an MCMC sample from MCMCTree or a BPP A00 analysis
time.name	character vector of length one
thin	numeric, the fraction of MCMC samples to keep

Details

tree must be rooted and strictly bifurcating, and it must match the tree used by BPP or MCMCTree to obtain the MCMC sample. The function uses the node ages in mcmc to calculate branch lengths and generate a list of trees (with the same topology as tree), one tree per (thinned) MCMC sample. The tips of the phylogeny are assumed to have age zero.

Value

An object of class multiPhylo (i.e., a list of trees).

Author(s)

Mario dos Reis

Examples

```
data(microcebus)
# convert a BPP A00 MCMC sample of Microcebus spp. to a list of trees
mtts <- mcmc2multiphylo(microcebus$tree, microcebus$mcmc, "tau_", thin=0.01)
length(mtts)

data(hominids)
# Calibrate the hominid phylogeny with a uniform fossil calibration of
# between 6.5 to 10 Ma for the human-chimp divergence.
calmsc <- msc2time.t(mcmc=hominids$mcmc, node="7humanchimp", calf=runif,
                     min=6.5, max=10)
# convert the time-calibrated MCMC sample to a list of trees
htts <- mcmc2multiphylo(hominids$tree, calmsc, "t_", thin=0.01)
htts[[1]]
```

```
## Not run:
# If you have the ape package installed, you can output the trees in Newick
ape::write.tree(htts[1:5])

# The trees are suitable for plotting with the phangorn package
# Relative node ages (tau's):
mcon <- microcebus$tree$tip.label
phangorn::densiTree(mtts, col="blue", alpha=0.04, cons=mcon, label.offset=.01)

# Absolute node ages (in millions of years):
hcon <- hominids$tree$tip.label
phangorn::densiTree(htts, col="blue", alpha=0.04, cons=hcon, label.offset=.01)

## End(Not run)
```

`microcebus` *A BPP A00 MCMC sample for a mouse lemur phylogeny*

Description

This dataset contains the results from the BPP A00 analysis of mouse lemur evolution in Madagascar from Yoder et al. (2016).

Usage

```
microcebus
```

Format

`microcebus` is a list with elements `mcmc`, a dataframe with 20,000 rows and 12 columns, and `tree`, an object of class `phylo` from the `ape` package.

`mcmc` is a posterior sample from a BPP A00 MCMC analysis containing the relative divergence times (tau's) and nucleotide diversities (theta's) for the six species mouse lemur (*Microcebus* spp) phylogeny.

`tree` contains the phylogeny with node ages given as the posterior means of the tau's in `mcmc`.

Source

A. D. Yoder, C. R. Campbell, M. B. Blanco, M. dos Reis, J. U. Ganzhorn, S. M. Goodman, K. E. Hunnicutt, P. A. Larsen, P. M. Kappeler, R. M. Rasoloarison, J. M. Ralison, D. L. Swofford, and D. W. Weisrock. (2016) *Geogenetic patterns in mouse lemurs (genus Microcebus) reveal the ghosts of Madagascar's forests past.* Proc. Nat. Acad. Sci. USA., 113: 8049–8056.

See Also

`hominids`

`msc2time` *Time-calibrate a multi-species phylogeny*

Description

Calibrate a BPP A00 MCMC sample from a multi-species coalescent phylogeny to absolute time using a fossil calibration or a prior on the molecular rate.

Usage

```
msc2time.t(mcmc, node.name, calf, ...)

msc2time.r(mcmc, u.mean, u.sd, g.mean, g.sd)
```

Arguments

`mcmc`	A data frame containing the MCMC output of a BPP A00 analysis
`node.name`	A character vector of length one with the name of the node to which the calibration will be applied
`calf`	A calibration function to generate random numbers
`...`	Further parameters passed to `calf`
`u.mean`	Numeric vector of length one with the mean for the per-generation molecular rate calibration
`u.sd`	Numeric vector of length one with the SD for for the per-generation molecular rate calibration
`g.mean`	Numeric vector of length one with the mean for the generation time calibration
`g.sd`	Numeric vector of length one with the SD for the generation time calibration

Details

`msc2time.t` will calibarte a BPP A00 analysis to geological time using a fossil calibration and `msc2time.r` will do the same but using a prior on the rate.

`msc2time.t` will obtain a sample of times from the random distribution in `calf`. Suitable choices for `calf` are `runif`, `rgamma`, and `rslnorm`. The sampled times are then used to calculate the molecular rate, and then re-scale the relative times (tau's) for the other nodes in `mcmc` to geological time.

`u.mean` and `u.sd`, and `g.mean` and `g.sd`, are used to construct gamma density calibrations for the per-generation molecular rate and generation time respectively. The gamma density with mean m and s.d. s has shape $a = (m/s)^2$ and rate $b = m/s^2$.

In `msc2time.r`, the gamma densities are used to obtain random samples of the per-generation rate and generation time. From these the molecular rate per absolute time unit is calculated, and then used to convert the relative times (tau's) to absolute divergence times. The relative population sizes (theta's) are converted to effective population sizes in number of individuals.

Angelis and dos Reis (2015) give the random sampling procedure used in these functions. Yoder et al. (2016) gives an example of calibrating a mouse lemur phylogeny using a prior on the rate. The BPP A00 analysis is described in Yang (2015).

Value

A data frame with a posterior sample of the calibrated times and molecular rate, and additionally, in the case of `msc2time.r`, the population sizes.

Author(s)

Mario dos Reis

References

K. Angelis and M. dos Reis (2015) *The impact of ancestral population size and incomplete lineage sorting on Bayesian estimation of species divergence times.* Curr. Zool., 61: 874–885.

Z. Yang (2015) *The BPP program for species tree estimation and species delimitation.* Curr. Zool., 61: 854–865.

A. D. Yoder, C. R. Campbell, M. B. Blanco, M. dos Reis, J. U. Ganzhorn, S. M. Goodman, K. E. Hunnicutt, P. A. Larsen, P. M. Kappeler, R. M. Rasoloarison, J. M. Ralison, D. L. Swofford, and D. W. Weisrock. (2016) *Geogenetic patterns in mouse lemurs (genus Microcebus) reveal the ghosts of Madagascar's forests past.* Proc. Nat. Acad. Sci. USA., 113: 8049–8056.

Examples

```
data(hominids)

# Calibrate the hominid phylogeny with a uniform fossil calibration of
# between 6.5 to 10 Ma for the human-chimp divergence.
calmsc <- msc2time.t(mcmc=hominids$mcmc, node="7humanchimp", calf=runif,
  min=6.5, max=10)

# posterior age of human-chimp (this is the same as the calibration)
plot(density(calmsc$t_7humanchimp, adj=.1), xlab="Time (Ma)",
  main="Human-chimp age")
rug(calmsc$t_7humanchimp)

## Not run:
# calculate posterior summary (requires CODA package)
mcmc.summary(calmsc)

## End(Not run)

# Calibrate the hominid phylogeny using a shifted-lognormal fossil calibration
# with minimum at 6.5 Ma for the human-chimp divergence.
calmsc <- msc2time.t(mcmc=hominids$mcmc, node="7humanchimp", calf=rslnorm,
  shift=6.5, sdlog=.5)
plot(density(calmsc$t_7humanchimp, adj=.1), xlab="Time (Ma)",
  main="Human-chimp age")
rug(calmsc$t_7humanchimp)

data(microcebus)

# Calibrate the Microcebus phylogeny to absoluate divergence times using a
```

```
# prior on the per-generation rate and generation time
calmsc <- msc2time.r(mcmc=microcebus$mcmc, u.mean=8.7e-9, u.sd=1.65e-9,
  g.mean=3.75, g.sd=0.375)

# posterior age of the phylogeny's root (in thousans of years)
plot(density(calmsc$t_7OLMXRB / 1e3, adj=.1), xlab="Time (Ka)",
  main="Root age (Microcebus)")
rug(calmsc$t_7OLMXRB / 1e3)

# Posterior of the ancestral effective population at the root (in
# thousands of individuals)
plot(density(calmsc$Ne_7OLMXRB / 1e3, adj=.1),
  xlab="Ne (x 10^3 individuals)", main = "Ne at root (Microcebus)")
rug(calmsc$Ne_7OLMXRB / 1e3)
```

`ShiftedLognormal` *The Shifted Log-normal Distribution*

Description

Density, distribution and quantile functions, and random number generation for the shifted log-normal distribution.

Usage

```
dslnorm(x, shift, meanlog = 0, sdlog = 1, log = FALSE)

pslnorm(q, shift, meanlog = 0, sdlog = 1, lower.tail = TRUE, log.p = FALSE)

qslnorm(p, shift, meanlog = 0, sdlog = 1, lower.tail = TRUE, log.p = FALSE)

rslnorm(n, shift, meanlog = 0, sdlog = 1)
```

Arguments

`x`, `q`	vector of quantiles.
`shift`	vector of shifts.
`meanlog`, `sdlog`	mean and standard deviation of the distribution on the log scale with default values of 0 and 1 respectively.
`log`, `log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are $P[X \leq x]$, otherwise, $P[X > x]$.
`p`	vector of probabilities.
`n`	number of observations. If `length(n) > 1`, the length is taken to be the number required.

Details

Let Y have a log-normal distribution with parameters μ (`meanlog`) and σ (`sdlog`). Then $X = Y + s$ has a shifted log-normal distribution with shift s (`shift`), mean $E(X) = exp(\mu + 1/2\sigma^2) + s$ and variance $Var(X) = exp(2 * \mu + \sigma^2) * (exp(\sigma^2) - 1)$.

Note `[dpqr]slnorm` are wrappers for the corresponding `[dpqr]lnorm` functions.

Value

`dslnorm` gives the density, `pslnorm` gives the distribution function, `qslnorm` gives the quantile function, and `rslnorm` generates random deviates.

The length of the result is determined by `n` for `rlnorm`, and is the maximum of the lengths of the numerical arguments for the other functions.

The numerical arguments other than `n` are recycled to the length of the result. Only the first elements of the logical arguments are used.

See Also

Lognormal

Examples

```
curve(dslnorm(x, shift=6.5), from=0, to=15, n=1e3)
rr <- rslnorm(1e3, shift=6.5)
lines(density(rr, adj=.1), lty=2)

all.equal (qslnorm(c(.025, .9), shift=6.5) - 6.5, qlnorm(c(.025, .9)))
all.equal (pslnorm(10, shift=6.5), plnorm(10 - 6.5))
```

---

stepping.stones *Estimate marginal likelihood by stepping stones*

Description

Estimate the marginal likelihood using the stepping stones method from a sample of `n` power posterior MCMC chains sampled with mcmctree (or bpp).

Usage

```
stepping.stones(mcmcf = "mcmc.txt", betaf = "beta.txt")
```

Arguments

`mcmcf`	character, mcmc output file name
`betaf`	character, file with beta values

Details

The MCMC samples should be stored in a directory structure created by `make.bfctlf` with `method = "step-stones"`. The function will read the stored log-likelihood values and calculate the log-marginal likelihood.

An approximation based on the Delta method is used to calculate the standard error (see Xie et al. 2011). Warnings are given if the approximation appears unreliable.

Value

A list with components `logml`, the log-marginal likelihood estimate; `se`, the standard error of the estimate; `mean.logl`, the mean of log-likelihood values sampled for each beta; and `b`, the beta values used.

Author(s)

Mario dos Reis

References

Xie et al. (2011) Improving marginal likelihood estimation for Bayesian phylogenetic model selection. *Systematic Biology*, 60: 150–160.

See Also

`make.bfctlf` to prepare directories and mcmctree or bpp control files to calculate the power posterior.

Index