

Package ‘jive’

July 9, 2014

Type Package

Title Analysis of joint intra- and interspecific trait evolution

Version 0.1

Date 2014-06-05

Author Anna Kostikova <anna.kostikova@unil.ch>, Daniele Silvestro <daniele.Silvestro@unil.ch>, Sacha Laurent <Sacha.Laurent@unil.ch>, Martha Serrano <martha.serrano@unil.ch>, Glenn Litsios <glenn.litsios@unil.ch>, Nicolas Salamin <nicolas.salamin@unil.ch>

Maintainer Anna Kostikova <anna.kostikova@unil.ch>

Description macroevolutionary analysis of joint intra- and interspecific variance evolution using comparative phylogenetic models. In addition, it can traitgrams through time based on various comparative models such as Brownian motion, Ornstein-Uhlenbeck, etc.

Depends R (>= 3.0.3)

Imports OUwie,nloptr,ape,MASS,phytools,TeachingDemos,hdrclde,akima,coda,scales,TreeSim,gtools

License GPL-2

R topics documented:

classicalModel	2
jiveMake	2
jiveMCMC	3
jivePlot	4
jiveProc	5
make.hpfun	5
slidingWin	6
Index	7

classicalModel	<i>Simulate trees</i>
----------------	-----------------------

Description

What it does

Usage

```
classicalModel(nb_diversification_shift, nb_mass_extinction, total_nb_species,
               lambda_max, lambda_min, mu_min, survival_rate_min, seed = 1)
```

Arguments

nb_diversification_shift	fff
nb_mass_extinction	fff
total_nb_species	fff
lambda_max	ff
lambda_min	fff
mu_min	ff
survival_rate_min	ff
seed	ff

Details

This function

Examples

```
lol <- classical_model(1, 1, 500, 0.1, 0.01, .01, .2, seed=1506621)
```

jiveMake	<i>Make jive object</i>
----------	-------------------------

Description

Creates a jive object from intraspecific observations and species phylogeny

Usage

```
jiveMake(simmap, traits, model.var = "OU1", model.mean = "BM",
         model.lik = "Multinorm")
```

Arguments

<code>simmap</code>	an object of class "jive" (see details)
<code>traits</code>	name of the output file that will store the log of MCMC chain
<code>model.var</code>	sampling frequency of the MCMC chain (how often chain will be saved into output file)
<code>model.mean</code>	printing frequency of the MCMC chain (how often chain will be printed in the R console)
<code>model.lik</code>	number of classes for thermodynamic integration (see details)

Details

This function creates a jive object needed for [jiveMCMC](#) function. Trait values must be stored as a matrix, where lines are vectors of observations for each species, with NA for no data. Rownames are species names. Phylogenetic tree must be provided as either simmap object (for models with multiple regimes) or as a phylo object (for BM or OU1 models). Rownames and tip labels of a phylogenetic tree should match exactly. There are three models implemented for estimation of species variances evolution - BM, OU1 and OUM. Evolution of species means is only implemented with BM model. Species-specific distribution are models as multivariate normal distribution

Examples

```
my.jive <- jiveMake(phy, traits, model.var="OU1", model.mean="BM", model.lik="Multinorm")
```

jiveMCMC

Jive MCMC

Description

Implements Markov chain Monte Carlo sampling for trait evolutionary models with intraspecific data

Usage

```
jiveMCMC(jive, log.file = "jive_mcmc.log", sampling.freq = 1000,
  print.freq = 1000, ncat = 1, beta.param = 0.3, ngen = 5500000,
  burnin = 0, update.freq = NULL)
```

Arguments

<code>jive</code>	an object of class "jive" (see details)
<code>log.file</code>	name of the output file that will store the log of MCMC chain
<code>sampling.freq</code>	sampling frequency of the MCMC chain (how often chain will be saved into output file)
<code>print.freq</code>	printing frequency of the MCMC chain (how often chain will be printed in the R console)
<code>ncat</code>	number of classes for thermodynamic integration (see details)
<code>beta.param</code>	beta value to define classes for thermodynamic integration (see details)
<code>ngen</code>	number of generation in MCMC chain

burning	a burning phase of MCMC chain (has to be specified for thermodynamic integration)
update.freq	update frequencies for likelihood and prior level parameters

Details

This function runs MCMC sampling on jive object `make.jive`. The jive object contains both the dataset and set of model to be used in MCMC. This function implements both a conventional MCMC and an MCMC with thermodynamic integration. The latter option is turned off by default and can be changed by setting `ncat` to values > 1 . The recommended `ncat` for TI is 10. When setting `ncat` > 1 , make sure to specify `brunin`. Larger `ncat` values will result in longer MCMC running time (?).

Examples

```
jiveMCMC(my.jive, log.file="ny.jive_MCMC.log")
```

jivePlot	<i>Plot jive MCMC object</i>
----------	------------------------------

Description

Creates a jive object from intraspecific observations and species phylogeny

Usage

```
jivePlot(tree, proc.jive, regime, cols = c("blue", "green"),
  cex.label = 0.7, cex.circle = 2, lab.off = 0.025, ladder = TRUE, ...)
```

Arguments

tree	an object of class "jive" (see details)
proc.jive	name of the output file that will store the log of MCMC chain
regime	sampling frequency of the MCMC chain (how often chain will be saved into output file)
cols	printing frequency of the MCMC chain (how often chain will be printed in the R console)
cex.label	number of classes for thermodynamic integration (see details)
cex.circle	
lab.off	
ladder	

Details

This function creates a jive object needed for `jiveMCMC` function. Trait values must be stored as a matrix, where lines are vector of observations for each species, with NA for no data. Rownames are species names. Phylogenetic tree must be provided as either `simmap` object (for models with multiple regimes) or as a `phylo` object (for BM or OU1 models)

Examples

```
jivePlot(phy1, my.l, regime)
```

jiveProc	<i>Process jive MCMC</i>
----------	--------------------------

Description

What it does

Usage

```
jiveProc(log.file = "jive_mcmc_OU1.log", n = 50, stat = jive.mode,
        burning = 0, probHPD = 0.95, verbose = TRUE, ...)
```

Arguments

log.file	log file recorded by jiveMCMC function
n	number of species
stat	which statistics to use to summarize MCMC. By default set to mode for prior level parameters and mean for likelihood level parameters
burning	how much of burning to disregard
probHPD	set HPD intervals
verbose	how much of statistics to return
...	additional parameters that can be passed to HPDinterval function

Details

This function processes the output log file of the jiveMCMC function. It summarizes posterior sample for each variable into summary statistics (e.g. mean, mode, median) and calculates HPD intervals

Examples

```
my.summary <- jiveProc(log.file="OU_log.log", n = 50, verbose=FALSE)
```

make.hpfun	<i>Title</i>
------------	--------------

Description

Short description

Usage

```
make.hpfun(hpf = "Uniform", hp.pars, ...)
```

Arguments

hpf	description
hp.pars	description
...	description

Details

Detailed description

Examples

```
my.hp <- make.hpfun(hpf="Uniform", hp.pars=c(1,2))
```

slidingWin

Proposal functions

Description

Multiplier and sliding window proposals

Usage

```
slidingWin(i, d)
```

Arguments

i	fff
d	fff
...	ff

Details

Make a function to get a hyper prior

Examples

```
sw <- slidingWin(i=2, d=9)
```

Index

`classicalModel`, [2](#)

`jiveMake`, [2](#)

`jiveMCMC`, [3](#), [3](#), [4](#)

`jivePlot`, [4](#)

`jiveProc`, [5](#)

`make.hpfun`, [5](#)

`make.jive`, [4](#)

`slidingWin`, [6](#)