

Package ‘menura’

May 22, 2019

Title Fitting (Non)-Gaussian diffusion models to phylogenies

Version 0.4.4

Author Simone P Blomberg

Description Fits user-defined stochastic diffusion models to univariate trait data on a phylogeny. Also fits three canned models: Ornstein Uhlenbeck (OU; Gaussian), Cox, Ingersoll, Ross (CIR; Non-Gaussian) and an OU-like model with a Beta stationary distribution (Beta, Non-Gaussian). Models are fitted and parameters are estimated using a Data Augmentation - Metropolis Hastings algorithm. Output can be analysed using the 'coda' package, 'tidybayes' package or other appropriate packages for Bayesian analysis and visualisation. Menura is the genus name for the Australian Superb Lyrebird (*Menura novaehollandiae*), known for being the world's largest passerine, its elaborate tail and its talent for mimicry. We wrote menura in order to study the behaviour of SDE models on phylogenies. Thus, menura is EXPERIMENTAL SOFTWARE. If you are unsure whether menura is appropriate for your own analyses, it probably isn't.

Maintainer Simone Blomberg <s.blomberg1@uq.edu.au>

Depends R (>= 3.2.0)

License GPL-3

Encoding UTF-8

LazyData true

Imports ape, sde, stats, graphics

NeedsCompilation no

Suggests knitr, rmarkdown

VignetteBuilder knitr

RoxygenNote 6.0.1.9000

R topics documented:

fit_model	2
fossil_id	4
phylo_sde	5
tree_logL	5

Index	7
-------	---

fit_model

*fit_model***Description**

Bayesian Estimator of Parameters of Univariate Diffusion Model for Continuous Trait Evolution

Usage

```
fit_model(tr, tipdata, rt_value = mean(tipdata), model = "OU",
  priors = list(alpha = list(df = function(x, a = 1, b = 125, log_scale =
    TRUE) {      dunif(x, min = a, max = b, log = log_scale) }, rf = function(n, a
    = 1, b = 125) {      runif(n, min = a, max = b) })), mu = list(df = function(x,
    a = 0, b = 20, log_scale = TRUE) {      dnorm(x, mean = a, sd = b, log =
    log_scale) }, rf = function(n, a = 0, b = 20) {      rnorm(n, mean = a, sd = b)
    })), sigma = list(df = function(x, a = 1, b = 225, log_scale = TRUE) {
    dunif(x, min = a, max = b, log = log_scale) }, rf = function(n, a = 1, b =
    225) {      runif(n, min = a, max = b) })), proposals = list(alpha = list(df
    = function(n, alpha, gamma = 0.5, log_scale = TRUE) {      dlnorm(n, meanlog =
    log(alpha), sdlog = gamma, log = log_scale) }, rf = function(n, alpha, gamma =
    0.5) {      rlnorm(n, meanlog = log(alpha), sdlog = gamma) })), mu = list(df =
    function(n, mu, gamma = 0.5, log_scale = TRUE) {      dnorm(n, mean = mu, sd =
    gamma, log = log_scale) }, rf = function(n, mu, gamma = 0.5) {      rnorm(n,
    mean = mu, sd = gamma) })), sigma = list(df = function(n, sigma, gamma = 0.5,
    log_scale = TRUE) {      dlnorm(n, meanlog = log(sigma), sdlog = gamma, log =
    log_scale) }, rf = function(n, sigma, gamma = 0.5) {      rlnorm(n, meanlog =
    log(sigma), sdlog = gamma) })), mcmc_type = "DA", alpha = NULL,
  mu = NULL, sigma = NULL, N = 1000, init_method = "sim",
  update_method = "subtree", iters = 5000, method = "euler",
  fossils = NULL, ...)
```

Arguments

tr	Single evolutionary tree as an object of the "phylo" class in the ape package.
tipdata	A numeric vector containing tip values in the same order as the tip labels in tr\$tip.label.
rt_value	Value at the root of tr.
model	Either a list containing drift and diffusion coefficients in quote format as functions of alpha, mu and sigma, or a string ("OU", "CIR", or "Beta") specifying the diffusion process. See Details.
priors	A list of lists containing functions for prior distributions of the model parameters. Use of the default priors is not recommended.
proposals	A list of lists containing functions for proposal distributions of the model parameters.
mcmc_type	Type of MCMC algorithm, "DA" or "Fuchs"

alpha	Set to NULL if alpha is to be estimated, otherwise set to a numeric value or a numeric vector specifying the value of the parameter for all the branches/edges. In the latter case, the values must be specifying in the same order as the edges in the tr object.
mu	As alpha.
sigma	As alpha.
N	Data augmentation frequency.
init_method	Method for initial data imputation. Currently only the "sim" option is available.
update_method	Method for data imputation during the MCMC. Option "subtree" will only update part of the tree at each iteration, where as "tree" will update the whole tree. See Details.
iters	Number of MCMC iterations.
method	Numerical approximation method, "euler" or "milstein."
fossils	A numeric vector containing the tip values for every fossil added to the tree.
...	Not used.

Details

This function estimates posterior distributions for evolutionary models of continuous traits in a phylogeny. The evolutionary processes considered here belong to a class of diffusion processes which are typically given as solutions to the stochastic differential equations of the form given by

$$dX_t = a(X_t, \alpha, \mu)dt + b(X_t, \sigma)dW_t, \quad X_0 = x_0$$

where X_t denotes the state variable, t the time, a the drift function and b the diffusion function are known in parametric form from which α , μ , and σ are the parameters, and W_t is Brownian motion. The value of X_t at time t_0 , X_0 , is independent of W_t .

Given root and tip values, the tree, and drift and diffusion functions, Markov Chain Monte Carlo (MCMC) estimates of the parameters are obtained. The parameter to be estimated is assumed to be the same for all the branches of the tree; however, the rest can be allowed to vary. Due to the low frequency nature of the data, the MCMC method used assumes intermediate data are missing. Thus, during each step of the MCMC iterations, missing data and the model parameters are imputed.

If the diffusion process is Ornstein-Uhlenbeck (OU), Cox-Ingersoll-Ross (CIR), or Beta (Beta) this can be specified by setting the model to "OU", "CIR", or "Beta", respectively. If the diffusion process is not one of these, the estimation of model parameters can be done by specifying drift and diffusion coefficients in a list assigned to model. In this case, the list object model must include : 1) d and s which are functions of time (t), space (x) and a vector of the parameter values (θ) consisting of α , μ and σ , 2) drift coefficients as a list containing a quote and 3) a diffusion coefficient as a list object which includes diffusion as the diffusion coefficients and x as the first derivatives of diffusion coefficients.

The Ornstein-Uhlenbeck (OU) model is given by

$$dX_t = \alpha(\mu - X_t)dt + \sigma dW_t,$$

with $X_0 = x_0 > 0$, W_t is the Brownian process, α , μ , and σ are the model parameters where α and σ are positive values.

The Cox-Ingersoll-Ross (CIR) model is given by

$$dX_t = \alpha(\mu - X_t)dt + \sigma \sqrt{X_t}dW_t,$$

with $X_0 = x_0 > 0$, where W_t is the Brownian process, α , μ , and σ are the model parameters which are all positive values. If the model == "CIR" is specified, then the parameter estimation is done by using transformation $Y = \sqrt{X}$ of Ito diffusion process.

The Beta model is given by

$$dX_t = \alpha(\mu - X_t)dt + \sigma \sqrt{X_t(1 - X_t)}dW_t,$$

with $X_0 = x_0 > 0$, W_t is the Brownian process, α , μ , and σ are the model parameters where α and σ are positive values. If the model == "Beta" is specified, then the parameter estimation is done by using transformation $Y = 2 \sin^{-1}(\sqrt{X})$ of Ito diffusion process.

See Also

[fossil_id](#)

fossil_id

fossil_id

Description

Create a vector of the location of the fossils in a tree

Usage

```
fossil_id(ftr)
```

Arguments

ftr Single evolutionary tree as an object of the "phylo" class in the ape package.

Details

This function locates the tip values of fossils in the tree. It is primarily used in other functions of menura to handle fossils data but can be useful to locate the fossils in the tree.

phylo_sde

phylo_sde

Description

Simulate a Cox-Ingersoll-Ross Diffusion Process in the Tree of Life

Usage

```
phylo_sde(tr, rt_value, N, theta, model, method, fossils = NULL, ...)
```

Arguments

<code>tr</code>	An object of class <code>phylo</code> from the <code>ape</code> package. In this version the CIR process parameters <code>alpha</code> , <code>mu</code> , and <code>sigma</code> for each branch are within vectors in the same order as the edge (branch) labelling.
<code>rt_value</code>	Value at the root of <code>tr</code> .
<code>N</code>	Data imputation frequency.
<code>theta</code>	Matrix of parameter values for each edge of the tree.
<code>model</code>	A list containing drift, diffusion, and the partial differentiation of diffusion as quoted expressions using <code>method quote</code> . For the Euler scheme the drift coefficient as <code>drift</code> , the diffusion coefficient as <code>diffusion</code> , and the partial differentiation of diffusion by <code>x</code> as <code>dx_diffusion</code> is required. See the Examples.
<code>method</code>	Specified as either "euler" or "milstein."
<code>fossils</code>	A numeric vector containing the tip values for every fossil added to the tree.
<code>...</code>	Not used.

Value

A list of time series projects for each simulated path equal to the length of the number of branches in the `tr` object.

tree_logL

tree_logL

Description

Calculate the loglikelihood of a Cox-Ingersoll-Ross Diffusion Process in the Tree of Life

Usage

```
tree_logL(tr, tipdata, lst, alpha, mu, sigma, model, method, fossils = NULL,
...)
```

Arguments

tr	An object of class phylo from the ape package. In this version the CIR process parameters alpha, mu, and sigma for each branch are within vectors in the same order as the edge (branch) labelling.
tipdata	A numeric vector containing tip values in the same order as the tip labels in tr\$tip.label.
lst	A list of time series projects for each simulated path equal to the length of the number of branches in the tr object.
alpha	Set to NULL if alpha is to be estimated, otherwise set to a numeric value or a numeric vector specifying the value of the parameter for all the branches/edges. In the latter case, the values must be specifying in the same order as the edges in the tr object.
mu	As alpha.
sigma	As alpha.
model	A list containing drift, diffusion, and the partial differentiation of diffusion as quoted expressions using method quote. For the Euler scheme the drift coefficient as drift, the diffusion coefficient as diffusion, and the partial differentiation of diffusion by x as dx_diffusion is required.
method	Specified as either "euler" or "milstein."
fossils	A numeric vector containing the tip values for every fossil added to the tree.
...	Not used.

Details

This function calculates the log likelihood.

Value

logL, an integer.

See Also

[fossil_id](#)

Index

fit_model, [2](#)
fossil_id, [4](#), [4](#), [6](#)
phylo_sde, [5](#)
tree_logL, [5](#)