

Package ‘BMhyd’

August 6, 2015

Version 1.2-8

Date 2015-08-05

Type Package

Title PCM for Hybridization

Author Dwueng-

Chwuan Jhwueng <djhwueng@umail.iu.edu> and Brian O'Meara <omeara.brian@gmail.com>

Maintainer Dwueng-Chwuan Jhwueng <djhwueng@umail.iu.edu>

Imports corpcor, numDeriv, geiger, ape, TreeSim, phytools, phylobase,
mvtnorm, grDevices, graphics, methods, stats

Description The BMhyd package analyzes the phenotypic evolution of species of hybrid origin on a phylogenetic network. This package can detect the hybrid vigor effect, a burst of variation at formation, and the relative portion of heritability from its parents. Parameters are estimated by maximum likelihood. Users need to enter a comparative data set, a phylogeny, and information on gene flow leading to hybrids.

License GPL (>= 2)

NeedsCompilation no

Repository CRAN

Date/Publication 2015-08-06 07:15:38

R topics documented:

AdaptiveConfidenceIntervalSampling	2
AdjustForDet	4
AICc	5
AkaikeWeight	6
AttachHybridsToDonor	7
BMhyd	8
CalculateLikelihood	10
cichlid	12
ContourFromAdaptiveSampling	12
DetPass	13
GenerateValues	14

GetAncestor	16
GetClade	17
GetMeansModified	18
GetVModified	19
LumpIntoClades	20
nicotiana	21
PlotNetwork	22
Plot_ConvexHull	23
SimulateNetwork	24
SimulateTipData	26
Index	27

AdaptiveConfidenceIntervalSampling

Confidence interval under adaptive cluster sampling technique

Description

This function uses adaptive cluster sampling technique to generate confidence interval for parameters of interest.

Usage

```
AdaptiveConfidenceIntervalSampling(par, fn, lower=-Inf, upper=Inf, desired.delta = 2,
n.points=5000, verbose=TRUE, ...)
```

Arguments

par	parameter of interest for sampling in the model.
fn	the negative log-likelihood function.
lower	the lower bound for the values allowed for sampling.
upper	the upper bound for the values allowed for sampling.
desired.delta	a numeric value with a default value of 2 for the criterions that the desired log-likelihood shall no more than 2 unit away from the maximum.
n.points	number of required points for calculating likeelihood value.
verbose	whether to print detailed information during the run.
...	further arguments passed(see details).

Details

This function starts with a given parameters values, then the points are generated by the function `GeneratedValues` and the likelihood value is calculated and sorted. To ensure the sampling is widely enough, we restrict the `width.ratio` (defined by the ratio of the sampled range over the total range of the points) to be less than 0.5. The default number of simulated point is set to 5000 and the desired intervals are determined with the end points that within the value of `desired.delt`. For the further argument passed, users shall specify the tree in phylo class, the structure of gene flow, the desired parameters for generated values, and whether is allow the extrapolation when the likelihood function is difficult to calculate.

Value

a data frame where the first column contains the calculated likelihood values and the rest of columns are the grid points of the parameters generated under the adaptive sampling technique.

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2014. *Studying trait evolution in hybrid species on phylogenetic networks*. Submitted.

Seber G.A.F., Salehi M.M. 2013. Adaptive sampling designs: Inference for spatial and clustered population. Springer.

Examples

```
library(corpcor)
#assign the number of non hybrid taxa
ntax.nonhybrid<-4
#assign the number of hybrid
ntax.hybrid<-1
#simulate the network
network<-SimulateNetwork(ntax.nonhybrid=ntax.nonhybrid, ntax.hybrid=ntax.hybrid,
flow.proportion=0.5, origin.type='clade', birth = 1, death = 0.5, sample.f = 0.5,
tree.height = 1, allow.ghost=FALSE)

#generate the tip data
tips<-rnorm(ntax.nonhybrid+ntax.hybrid)
names(tips)<-paste("t", (1:(ntax.nonhybrid+ntax.hybrid)), sep="")

#set free parameters
free.parameters<-rep(TRUE, 5)
names(free.parameters) <- c("sigma.sq", "mu", "bt", "vh", "SE")

#the best parameter values for the confidence intervals
par<-c(0.01,1,1,0,0)

#Simulate 100 samples
```

```

interval.results <- AdaptiveConfidenceIntervalSampling(par, fn=CalculateLikelihood,
lower=c(0, -Inf, 0, 0, 0)[which(free.parameters)], n.points=100,data=tips,
phy=network$phy, flow=network$flow, actual.params =
free.parameters[which(free.parameters)], allow.extrapolation=TRUE)

#show the results
interval.results.in <- interval.results[which(interval.results[,1] -
min(interval.results[,1])<=2),]
interval.results.out <- interval.results[which(interval.results[,1] -
min(interval.results[,1])>2),]
interval.results.in
interval.results.out

```

AdjustForDet

Adjust the phylogeny

Description

A function adjusts the phylogeny when the determinant of the variance covariance matrix of the the phylogeny is negative.

Usage

```
AdjustForDet(phy, max.attempts=100)
```

Arguments

phy an object of class 'phylo'
max.attempts the maximum number of attempts to adjust the phylogeny. The default is 100.

Details

This function first calculates the determinant of the phylogeny using the function DetPass. When the determinant of the variance covariance matrix for the phylogeny is negative, this function will adjust the phylogeny by slightly lengthening the edge lengths of the tree. This procedure repeats until a good adjusted phylogeny is returned.

Value

an object of class 'phylo'

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Studying trait evolution in hybrid species on phylogenetic networks*. Submitted.

Examples

```
library(ape)
#simulate a tree of 5 taxa
phy<-rtree(5)
#check and adjust the phylogeny
AdjustForDet(phy,max.attempts=100)
```

AICc

AICc

Description

Calculates the second order Akaike's information criterion score for models of interest.

Usage

AICc(n, k, LogLik)

Arguments

n	number of taxa for the given phylogenetic tree. It represents the sample size(the number of species on the tip of phylogeny).
k	number of free parameters in the model
LogLik	the minimum of the negative log-likelihood value obtained by optimizing the likelihood function.

Details

'AICc' is a function to compute the AICc values and is valid to select among different models. $AICc = 2 * n * k / (n - k - 1) - 2 \log L$ where L is the maximum likelihood for the model.

Value

The AICc values.

Author(s)

Dwueng-Chwuan Jhwueng

References

Burnham, K.P., and D.R. Anderson. 2004. *Model selection and inference: a practical information-theoretic approach*. Sec. Ed. Springer, New York.

Examples

```
#assign the size
n<-5
#assign the number of parameter
k<-3
#assign the negative log likelihood value.
LogLik<- -2
#compute the AICc score
AICc(n,k,LogLik)
# result AICc value of 26.
```

AkaikeWeight

Akaike weight

Description

Calculate Akaike weight for the models

Usage

```
AkaikeWeight(Delta.AICc.Array)
```

Arguments

Delta.AICc.Array

Delta.AICc.Array is defined as the difference between the AICc value and the minimum AICc value among candidate models.

Details

For n models of interest, the Akaike weight for the i th model is defined as $w = \exp(-0.5 * \Delta AICc_i) / \sum_i \exp(-0.5 * \Delta AICc_i)$ where $\Delta AICc_i$ is the difference of the AICc value between the i th model and the best model. The weights can be used in model averaging in advanced.

Value

Akaike weights.

Author(s)

Dwueng-Chwuan Jhwueng

References

Burham, K.P., Anderson, D.R. (2002) Model selection and multimodel inference: a practical information-theoretic approach. Second edition. Springer. New York.

Examples

```
#simulate 4 AICc values for 4 models.
AICc_Array<-rnorm(4, mean=10,sd=1)
#calculate the delta AICc
Delta.AICc.Array<-AICc_Array-min(AICc_Array)
#calculate the Akaike weight
AkaikeWeight(Delta.AICc.Array)
```

AttachHybridsToDonor *Attach hybrids to the donor species*

Description

This function attaches the hybrid to the donor species.

Usage

```
AttachHybridsToDonor(phy, flow, suffix="_DUPLICATE")
```

Arguments

phy	a phylogeny of phylo class.
flow	structure of gene flow.
suffix	a duplicated species that used for hybrid information in advanced.

Details

This function attaches hybrid to the donor and return the new phylogeny. It first identifies the flow clades using the function `LumpIntoClades`. Then for each clade containing the flow, it searches the donor taxa and then attach the hybrid into the clade ancestor. For empirical analysis, users shall have the correct format for the flow structure to ensure the hybrid information.

Value

a merged tree that identifies the hybrid species from the original tree.

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Trait evolution on phylogenetic networks*. Submitted.

Examples

```
#set the number of non hybrid species
ntax.nonhybrid<-4
#set the number of hybrid species
ntax.hybrid<-1
#simulate a network
network<-SimulateNetwork(ntax.nonhybrid=ntax.nonhybrid, ntax.hybrid=ntax.hybrid,
flow.proportion=0.5, origin.type='clade', birth = 1, death = 0.5, sample.f = 0.5,
tree.height = 1, allow.ghost=FALSE)
#show the tree.
network$phy
#show the flow information
network$flow
#Attach hybrid species to the donor and show the new tree.
#The name of the hybrid is displayed with suffix "_DUPLICATE"
AttachHybridsToDonor(network$phy, network$flow, suffix="_DUPLICATE")
```

BMhyd

*Comparative method for studying hybridization using Brownian motion
for trait evolution*

Description

This function fits the Brownian motion model of continuous character to investigate hybrid species through the hybrid vigor β , and the variation at the burst of hybridization v_H . Measurement error SE is also considered as well as the parameters including the over all mean μ and the overall variance σ^2 in the typical Brownian motion model.

Usage

```
BMhyd(data, phy, flow, opt.method="Nelder-Mead", models=c(1,2,3,4), verbose=TRUE,
get.se=TRUE, plot.se=TRUE, store.sims=FALSE, precision=2, auto.adjust=FALSE,
likelihood.precision=0.001, allow.extrapolation=FALSE)
```

Arguments

data	continuous trait data containing species information in vector format
phy	a tree in phylo class
flow	a struture of gene flow
opt.method	the method for used for optimization. The default is Nelder-Mead
models	the model used for analysis (see details)

verbose	a TRUE/FALSE argument to start optimization
get.se	a TRUE/FALSE argument estimation for doing simulation to estimate parameter uncertainty (see details)
plot.se	a TRUE/FALSE argument for output the uncertainty plot for the model (see details)
store.sims	a TRUE/FALSE argument to record the the parameter estimates and relevant values.
precision	a numeric value to present the cutoff at which the user thinks the estimates become unreliable due to ill conditioned matrix.
auto.adjust	a TRUE/FALSE argument to adjust the the phylogeny
likelihood.precision	a numerical value used for verifying the convergent of the estimation.
allow.extrapolation	a TRUE/FALSE argument. If TRUE, the VCV matrix was ill-conditioned, so used splines to estimate its likelihood.

Details

Function **BMhyd** fits likelihood models for continuous characters. It incorporates phylogenetic tree, structure of gene flow and comparative data. The full likelihood model includes several parameters: the ancestral state μ , the overall variation σ^2 , the hybrid vigor β , the hybrid burst variation at formation v_H and the measurement error SE. The structure of gene flow is a five column table where the first and the second column contain the donor and recipient information. The third column is the information about the heritability factor m which is a fraction of the recipient trait that comes from the source. The fourth column, recipient time, records time from the root of the recipient that counting forward from the root when the gene flow happened from the donor. The fifth column, recipient time, records time from the root of the recipient that counting forward from the root when the gene flow happened from the donor. For detail modeling, see the manuscript in Jhwueng and O'Meara (2015). The function allows some fixed values of parameters and treat others as free parameters: model 1 fixes β at 1 but allow v_H to vary; model 2 allows β to vary but fixes v_H at 0; model 3 fixes β at 1 and v_H at 0; and model 4 allows both to vary. **BMhyd** fits the model through maximum likelihood technique, it returns MLEs. When setting the `get.se` to TRUE, the **BMhyd** will do simulation to estimate parameter uncertainty and the confidence intervals by adaptive confidence interval sampling method(see manuscript in Jhwueng and O'Meara). It setting `plot.se` to be TRUE, **BMhyd** will be saved the CI in PDF file once the simulations are finished. Model averaged parameter estimates are calculated by the Akaike weight. As an input, **BMhyd** requires a phylogenetic tree of the phylo class, a structure of gene flow and a comparative data. Currently the method is develop for univariate analysis where the comparative data includes a single trait for analyses.

Value

A summarized table including the type of model, the corresponding number of parameter, the parameter estimates, the likelihood values, the upper bound and lower bound of the parameters, and the Akaike weights for model averaging.

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Studying trait evolution in hybrid species on phylogenetic networks*. Submitted.

Burnham, K.P., and D.R. Anderson. 2004. *Model selection and inference: a practical information-theoretic approach*. Sec. Ed. Springer, New York.

Examples

```
#set up the number of non hybrid
ntax.nonhybrid<-2
#set up the number of non hybrid
ntax.hybrid<-1
#simulate a network
network<-SimulateNetwork(ntax.nonhybrid=ntax.nonhybrid, ntax.hybrid=ntax.hybrid,
flow.proportion=0.5, origin.type='clade', birth = 1, death = 0.5, sample.f = 0.5,
tree.height = 1, allow.ghost=FALSE)
#simulate the tips data
tips<-rnorm(ntax.nonhybrid+ntax.hybrid)
names(tips)<-paste("t", (1:(ntax.nonhybrid+ntax.hybrid)), sep="")
#run the analysis uses model 3

BMhyd(tips,network$phy,network$flow, opt.method="Nelder-Mead", models=3, verbose=TRUE,
get.se=FALSE, plot.se=FALSE, store.sims=FALSE, precision=2, auto.adjust=FALSE,
likelihood.precision=0.001, allow.extrapolation=FALSE)
```

CalculateLikelihood *Calculate the likelihood value for the model*

Description

This function calculates the likelihood value for the model

Usage

```
CalculateLikelihood(x, data, phy, flow, actual.params, precision=2,
proportion.mix.with.diag=0, allow.extrapolation=FALSE)
```

Arguments

x	parameter of interest
data	the trait values
phy	a tree of phylo class
flow	strcuture of gene flow
actual.params	the free parameters for hybridization
precision	a number to verify the condition of the variance covariation for the network model
proportion.mix.with.diag	the proportion value that applies to the diagonal of the vcv matrix
allow.extrapolation	a TRUE/FALSE argument

Details

This function calculates the likelihood value. As described in the argument, the x contains the parameter of interests including over all mean μ , the rate of evolution σ^2 , the measurement error SE and the hybrid vigor β and the variation at the burst of hybridization v_H . Names of the taxa in data vector and the phylogeny must be match for further analysis, otherwise it would terminate immediately. The negative log likelihood function is calculated and a reasonable likelihood value is retruned by checking the variance covariance matrix where a precision number is used for verifying the log conditioned number of the variance covaraince of the network model. When the matrix is ill conditioned, we appraopriately adjust the matrix by the shrink the value of the off diagonal matrix using a proportion array. Then the likelihood is calculated using a spline approximation.

Value

negative log likelihood value

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2014. *Studying trait evolution in hybrid species on phylogenetic networks*. Submitted.

Examples

```
#set the number of hybrid
ntax.nonhybrid<-5
#set the number of hybrid
ntax.hybrid<-1
#simulate a network
network<-SimulateNetwork(ntax.nonhybrid=ntax.nonhybrid, ntax.hybrid=ntax.hybrid,
flow.proportion=0.5, origin.type='clade', birth = 1, death = 0.5, sample.f = 0.5,
tree.height = 1, allow.ghost=FALSE)
```

```

sigma.sq <- 0.01;mu <- 1;SE <- 0
#simulatedata
data<-rnorm(ntax.nonhybrid+ntax.hybrid)
names(data)<-paste("t", (1:(ntax.nonhybrid+ntax.hybrid)), sep="")
#calculate the likelihood value
CalculateLikelihood(c(sigma.sq,mu,SE), data, network$phy, network$flow, "vh",
precision=2, proportion.mix.with.diag=0, allow.extrapolation=TRUE)

```

cichlid

Cichlid data from Kobmuller et al. 2007

Description

This gives the phylogeny, structure of gene flow and comparative data for extant species from Kobmuller et al. 2007.

Usage

```
cichlid
```

Format

a list with the phylogeny (phy), gene flow (flow), and comparative data (data) for cichlid species.

Source

Kobmuller et al. 2007

References

Kobmuller, S., N. Duftner, K. M. Sefc, M. Aibara, M. Stipacek, M. Blanc, B. Egger, and C. Sturmbauer. 2007. Reticulate phylogeny of gastropod-shell-breeding cichlids from Lake Tanganyika: the result of repeated introgressive hybridization. *BMC Evolutionary Biology* 7:7.

ContourFromAdaptiveSampling

Contour maps

Description

This function generates the contour maps for a pair of parameters of interest.

Usage

```
ContourFromAdaptiveSampling(sims, params.of.interest=NULL)
```

Arguments

`sims` a data frame with the simulation results
`params.of.interest` name of parameter that is of interest to be plotted

Details

This function generates the contour plots for the parameter of interest. It uses the simulated data from generating the confidence interval under adaptive sampling technique where the likelihood value and parameters values are stored. Then it uses the function `Plot_ConvexHull` to generate the contour maps.

Value

It returns the plot of the contour map for a pair of parameters.

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Studying trait evolution in hybrid species on phylogenetic networks*. Submitted.

Examples

```
#simulate a data frame for two parameters
sims<-matrix(rnorm(300),ncol=3)
sims<-as.data.frame(sims)
colnames(sims)<-c("neglnL","param1","param2")
#plot the contour map for the parameters
ContourFromAdaptiveSampling(sims, params.of.interest=NULL)
```

DetPass

determinant of the matrix

Description

Calculate the determinant of the matrix and check whether it is positive.

Usage

`DetPass(phy)`

Arguments

phy an object of class 'phylo'

Details

This function first calculates variance covariance matrix C for the tree. It then calculates the determinant of the C, $0.0001 * C$, and $1000 * C$ and verifies whether all the three determinants are positive and finite. If yes, a TRUE value is return and the C matrix is good for further use.

Value

a TRUE/FALSE value.

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Studying trait evolution in hybrid species on phylogenetic networks*. Submitted.

Examples

```
library(ape)
##simulate a tree of 5 taxa
phy<-rtree(5)
#check whether the determiniant is positive
DetPass(phy)
```

GenerateValues

Generate parameter values

Description

This function generate new parameter values

Usage

```
GenerateValues(par, lower, upper, max.tries=100, expand.prob=0, examined.max,
examined.min)
```

Arguments

par	parameters of interest
lower	the actual lower bound for the parameters
upper	the actual upper bound for the parameters
max.tries	maximum number of attempts to generate the parameter value. The default is set to 100.
expand.prob	a probability value to expand.
examined.max	the allowed maximum for the parameters values.
examined.min	the allowed minimum for the parameters values

Details

This function generates new parameter values using the uniform distribution with the interval (L, U) where for each of the parameter of interest, the lower bound of the interval is set to $L = \max(\text{lower}, 0.9 * \text{examined.min})$ and the upper bound of the interval is set to $U = \min(\text{upper}, 1.1 * \text{examined.max})$.

Value

The generated parameter values.

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2014. *Studying trait evolution in hybrid species on phylogenetic networks*. Submitted.

Seber G.A.F., Salehi M.M. 2013. Adaptive sampling designs: Inference for spatial and clustered population. Springer.

Examples

```
#parameter of interests
mu<-1;sigma.sq<-0.1;bt<-1;vH<-0;SE<-0
#parameters of interest
par<-c(mu,sigma.sq,bt,vH,SE)
#the lower bound
lower=c(-2,0,-10,0,0)
#the upper bound
upper<-c(2,1,10,100,100)
#the examined minimum value
examined.min<-c(-1.8,0.001,-9.8,0,0)
#the examined maximum value
examined.max<-c(1.90,1.1,10.2,100,100)
#simulate points
sim.points<-GenerateValues(par, lower, upper, max.tries=100, expand.prob=0,
```

```
examined.max, examined.min)
names(sim.points)<-c("mu", "sigma.sq", "bt", "vH", "SE")
#show result
sim.points
```

GetAncestor

Identify the ancestor

Description

Given the descendent node for the tree, the function will return its ancestor node.

Usage

```
GetAncestor(phy, node)
```

Arguments

phy	an object of class 'phylo'.
node	descendant node.

Value

the ancestor node

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Studying trait evolution in hybrid species on phylogenetic networks*. Submitted.

Examples

```
library(ape)
#simulate a tree of 3 taxa
phy<-rtree(3)
#plot the tree
plot(phy)
#descendant node
node<-1
#get the ancestor node, it will return 5.
GetAncestor(phy,node)
```

GetClade	<i>Get clade from the tree</i>
----------	--------------------------------

Description

Search and report the nodes of the tree that have the desired clade size.

Usage

```
GetClade(phy, clade.size)
```

Arguments

phy	an object of class 'phylo'.
clade.size	size of the clade.

Details

This function uses the 'phylo' class where the edges(ancestor-decedant relationship) of the tree are used to identify the interior node with desired number of descendent (clade.size) on the tips. It applies the function `findMRCA` in *phytools* to search the ancestor.

Value

the interior node that has descendants of size clade.size on the tip.

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2014. *Studying trait evolution in hybrid species on phylogenetic networks*. Submitted.

Examples

```
library(ape)
library(phytools)
#simulate a tree
phy<-rtree(3)
#plot the tree
plot(phy)
#set up the clade size
clade.size<-2
#search the nodes that has desired clade size
GetClade(phy,clade.size) #node 5 will be return
```

GetMeansModified *The species means for the network model*

Description

This function returns the species mean for the network model. For the non hybrid species, it returns as a parameter μ and for hybrid species it returns $\mu + \log \beta$.

Usage

```
GetMeansModified(x, phy, flow, actual.params)
```

Arguments

x	vector of the parameters: including rate of evolution σ^2 , overall mean μ , measurement error SE, hybrid vigor β , variation at the burst of hybridization v_H .
phy	a class of phylo tree.
flow	a flow structure of gene flow.
actual.params	The parameters that are related to the hybridization including the hybrid vigor β and variation at the burst of hybridization v_H .

Value

the species mean

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Trait evolution on phylogenetic networks*. Submitted.

Examples

```
#number of nonhybrd species
ntax.nonhybrid<-5
#number of hybrid species
ntax.hybrid<-2
#simulate network
network<-SimulateNetwork(ntax.nonhybrid=ntax.nonhybrid, ntax.hybrid=ntax.hybrid,
flow.proportion=0.5, origin.type='clade', birth = 1, death = 0.5, sample.f = 0.5,
tree.height = 1, allow.ghost=FALSE)
#set the parameter values
sigma.sq <- 0.01;mu <- 1;SE <- 0;bt<-12;vh<-0
par<-c(sigma.sq,mu,SE,bt,vh)
names(par)<-c("sigma.sq","mu","SE","bt","vh")
```

```

actual.params<-12
names(actual.params)<-"bt"
#calculate the species means
GetMeansModified(par, network$phy, network$flow,actual.params)

```

GetVModified

Variance covariance matrix for the network model

Description

A function returns the variance covariance function for the network model under the Brownian motion for trait evolution.

Usage

```
GetVModified(x, phy, flow, actual.params)
```

Arguments

x	vector of the parameters: including rate of evolution σ^2 , overall mean μ , measurement error SE, hybrid vigor β , and variation at the burst of hybridization v_H .
phy	a class of phylo tree.
flow	a structure of gene flow.
actual.params	The parameters that are related to the hybridization including the hybrid vigor β and variation at the burst of hybridization v_H .

Details

This function calculates the variance covariance matrix under the network model. The function first reads the tree in phylo class and converts the variance covariance matrix using the Brownian motion model. It then uses the structure of the gene flow to identify the hybrid species, parents and the direction of the flow. The covariance between the hybrid and non hybrid species is the weighted sum of the covariance from evolution along the tree plus evolution along the migration path. For the detail model description, please see Jhwueng and O'Meara 2015.

Value

the variance covariance matrix of size equal to the number of species on the tip of the phylogeny.

Author(s)

Brian O'Meara, Dwieng-Chwan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Trait evolution on phylogenetic networks*. Submitted.

Examples

```
#set the number of non hybrid species
ntax.nonhybrid<-4
#set the number of hybrid species
ntax.hybrid<-1
#simulate a network
network<-SimulateNetwork(ntax.nonhybrid=ntax.nonhybrid, ntax.hybrid=ntax.hybrid,
flow.proportion=0.5, origin.type='clade', birth = 1, death = 0.5, sample.f = 0.5,
tree.height = 1, allow.ghost=FALSE)
#assign the parameter values
sigma.sq <- 0.01;mu <- 1;SE <- 0
#get the variance covariance matrix
GetVModified(c(sigma.sq,mu,SE), network$phy, network$flow, "bt")
```

LumpIntoClades

Lump into clades

Description

This function Lump the hybrid species into a clades

Usage

```
LumpIntoClades(phy, flow)
```

Arguments

phy	a phylogeny of phylo class
flow	structure of gene flow

Details

The function summaries the flow structure into the clade where the donor clades contribute more than one recipient clades.

Value

a flow table that contains the donor clade, recipient clade, proportion of the gene flow,time from root donor, and time from root recipient.

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Trait evolution on phylogenetic networks*. Submitted.

Examples

```
#set the number of non hybrid species
ntax.nonhybrid<-5
#set the number of hybrid species
ntax.hybrid<-2
#simulate a network
network<-SimulateNetwork(ntax.nonhybrid=ntax.nonhybrid, ntax.hybrid=ntax.hybrid,
flow.proportion=0.5, origin.type='clade', birth = 1, death = 0.5, sample.f = 0.5,
tree.height = 1, allow.ghost=FALSE)
#show the flow
network$flow
#show the new flow table
LumpIntoClades(network$phy, network$flow)
```

nicotiana

nicotiana data

Description

The phylogeny, gene flow, and comparative data for nicotiana.

Usage

nicotiana

Format

a list with the phylogeny (phy), gene flow (flow), and comparative data (data) for Nicotiana species.

Source

Chase et al. (2003), Clarkson et al. (2005), Komori et al. (2000)

References

Chase M.W., Knapp S., Cox A.V., Clarkson J.J., Butsko Y., Joseph J., Savolainen V., and Parokony A.S. 2003. Molecular systematics, GISH and the origin of hybrid taxa in Nicotiana(Solanaceae). *Annals of Botany* 92: 107-127.

Clarkson J.J., Lim K.Y., Kovarik A., Chase M.W., Knapp S. and Leitch A.R. 2005. Long-term genome diploidization I allopolyploid Nicotiana section Repandae(Solanaceae). *New Phytologist* 168:241-252.

Komori T., Myers P.N., Yamada S., Kubo T., and Imaseki H. 2000. Comparative study of the Nicotiana species with respect to water deficit tolerance during early growth. *Euphytica* 116:121-130.

 PlotNetwork

Phylogenetic Network Plot

Description

This function plots the phylogenetic network with required number of taxa . The hybrids with gene flow direction.

Usage

```
PlotNetwork(phy, flow, col.non="black", col.hybrid="red", col.donor="blue",
name.padding=1.5, cex=1, xlab="", bty="n", head.length=0.2, edge.width=2,
col.tree="darkgray", col.arrow="red", arrow.width=1, ...)
```

Arguments

phy	a class of phylo tree
flow	a flow structure of gene flow
col.non	color of non hybrid taxa
col.hybrid	color of hybrid taxa
col.donor	color of the donor
name.padding	the size that allowed for the plot in the x axis
cex	number indicating the amount by which plotting text and symbols should be scaled relative to the default. 1=default, 1.5 is 50 percent larger, 0.5 is 50 percent smaller, etc.
xlab	X axis label using font and character expansion par("font.lab") and color par("col.lab").
bty	the type of box to be drawn around the legend. The allowed values are "o" (the default) and "n".
head.length	length of the head.
edge.width	width of the edges.
col.tree	color of the tree.
col.arrow	color of gene flow arrow.
arrow.width	width of the arrow.
...	other arguments

Details

This function generates plots for network models. It first draws the phylogenetic tree for the given species. Then uses the gene flow data to draw the arrow from the donor species to recipient species. The hybrid taxa are shown in red with the gene flow direction from the donor species (colored in blue).

Value

plot of the network that involves the phylogeny and the gene flow direction.

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Trait evolution on phylogenetic networks*. Submitted.

Examples

```
#set the number of non hybrid species
ntax.nonhybrid<-10
#set the number of hybrid species
ntax.hybrid<-3
#simulate the network with desired species
network<-SimulateNetwork(ntax.nonhybrid=ntax.nonhybrid, ntax.hybrid=ntax.hybrid,
flow.proportion=0.5, origin.type='clade', birth = 1, death = 0.5, sample.f = 0.5,
tree.height = 1, allow.ghost=FALSE)
#plot the network
PlotNetwork(network$phy, network$flow)
```

Plot_ConvexHull	<i>Convex hull</i>
-----------------	--------------------

Description

Plotting the convex hull for given 2D datasets.

Usage

```
Plot_ConvexHull(xcoord, ycoord, lcolor)
```

Arguments

xcoord	coordinate for the horizontal axis.
ycoord	coordinate for the vertical axis.
lcolor	color of the convex hull.

Details

This function uses the base function *chull()*, which takes x,y coordinates of data and outputs a vector(hpts variable) of integers that indicate the points in dataset that comprise the convex hull around data. It then uses the base function *lines* to draw the convex hull using the x,y coordinates stored in hpts.

Value

The convex hull plot for a set of 2D points.

Author(s)

Ken Takagi

References

<https://chitchatr.wordpress.com/2011/12/30/convex-hull-around-scatter-plot-in-r/>

Examples

```
# Create a set of random data to plot convex hull around
x<-rnorm(100,0.8,0.3)
y<- rnorm(100,0.8,0.3)
#get max and min of all x and y data for nice plotting
xrange<-range(x)
yrange<-range(y)
#plot it up!
plot(x,y,type="p",pch=1,col='black',xlim=c(xrange),ylim=c(yrange))
Plot_ConvexHull(xcoord=x,ycoord=y,lcolor='red')
```

SimulateNetwork

Simulate phylogenetic network

Description

Simulate network with desired number of taxa and number of hybrid.

Usage

```
SimulateNetwork(ntax.nonhybrid=100, ntax.hybrid=10, flow.proportion=0.5,
origin.type=c("clade", "individual"), birth = 1, death = 0.5, sample.f = 0.5,
tree.height = 1, allow.ghost=FALSE)
```

Arguments

ntax.nonhybrid	number of non hybrid taxa.
ntax.hybrid	number of hybrid taxa.
flow.proportion	the gene flow proportion from the parent.
origin.type	the type where the hybrids were formed. clade: hybrid formed and then speciate; individual: hybrid species formed and evolved without speciation.
birth	birth rate for the tree speciation.

death	death rate (extinction) of the tree.
sample.f	sampling frequency.
tree.height	the height of the tree.
allow.ghost	allows ghost lineage persists hybridization for a while and then goes extinct. Otherwise, hybridization event will be between coeval with extant descendants.

Details

This function generates tree with relevant hybridization information (the structure of gene flow information). The gene flow must happen on its stem edge, which starts at shortest from root and goes up to longest from root. Gene flow cannot go back in time. In general, gene flow can go forward in time via ghost lineages. If no ghost lineages, then there must be temporal overlap between the donor and recipient lineages. So the tipward end of the donor edge must be later than the rootward end of the recipient edge. When the non of hybr is 1 and clade sampling, this will do individual sampling instead.

Value

a list that contains

phy	a birth death tree with number of taxa of ntax.nonhybrid + ntax.hybrid
flow	The flow structure of the gene flow

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Trait evolution on phylogenetic networks*. Submitted.

Examples

```
library(TreeSim)
#set up the number for non hybrid species
non.hybrid <- 3
#set up the number for hybrid species
hybrid <- 1
#set up the gene flow porportion
flow<- 0.5
#set pu the hybridization type to original
origins<- "individual"
#start to simulate the network
network<-SimulateNetwork(ntax.nonhybrid=non.hybrid, ntax.hybrid=hybrid,
flow.proportion=flow, origin.type="individual", birth = 1, death = 0.5,
sample.f = 0.5, tree.height = 1, allow.ghost=FALSE)
#print out the result
(network)
```

SimulateTipData *Simulate data on a network*

Description

Simulate data on a network given its flow parameters

Usage

```
SimulateTipData(phy, flow, params, suffix="_DUPLICATE")
```

Arguments

phy	phylo object
flow	data.frame of the flow parameters
params	named vector of BMhyd parameter values
suffix	taxa are duplicated on the network; this suffix distinguishes them

Details

This function simulates tip data on a network, under a BMhyd model.

Value

a named vector of tip values

Author(s)

Brian O'Meara, Dwueng-Chwuan Jhwueng.

References

Jhwueng D.C. and O'Meara B.C. 2015. *Trait evolution on phylogenetic networks*. Submitted.

Examples

```
#add example later. Perhaps using cichlid
```

Index

*Topic **datasets**

cichlid, [12](#)
nicotiana, [21](#)

AdaptiveConfidenceIntervalSampling, [2](#)

AdjustForDet, [4](#)

AICc, [5](#)

AkaikeWeight, [6](#)

AttachHybridsToDonor, [7](#)

BMhyd, [8](#)

CalculateLikelihood, [10](#)

cichlid, [12](#)

ContourFromAdaptiveSampling, [12](#)

DetPass, [13](#)

GenerateValues, [14](#)

GetAncestor, [16](#)

GetClade, [17](#)

GetMeansModified, [18](#)

GetVModified, [19](#)

LumpIntoClades, [20](#)

nicotiana, [21](#)

Plot_ConvexHull, [23](#)

PlotNetwork, [22](#)

SimulateNetwork, [24](#)

SimulateTipData, [26](#)