

Package ‘bGMYC4’

September 4, 2026

Type Package

Title Bayesian General Mixed Yule-Coalescent Model for Species Delimitation

Version 4.1.6

Description Implements a Bayesian version of the general mixed Yule-coalescent model for species delimitation. It uses Markov Chain Monte Carlo simulation to evaluate the posterior distribution of the model and supports flexible prior specification on model parameters.

License GPL (>= 3)

Encoding UTF-8

LazyData true

Depends R (>= 4.0.0)

Imports ape (>= 5.0), future, future.apply

Suggests mcmcse, plotly, base64enc, testthat (>= 3.0.0), knitr, rmarkdown, htmltools, htmlwidgets, treeio, ggtree, dplyr

VignetteBuilder knitr, rmarkdown

URL <https://github.com/DmitryKarabanov/bGMYC4>

BugReports <https://github.com/DmitryKarabanov/bGMYC4/issues>

RoxygenNote 8.0.0

NeedsCompilation no

Author Noah Reid [aut, cph] (Original author of bGMYC),
Dmitry Karabanov [aut, cre] (ORCID:
<<https://orcid.org/0000-0001-6008-7441>>)

Maintainer Dmitry Karabanov <dk@ibiw.ru>

Repository CRAN

Date/Publication 2026-09-04 20:30:02 UTC

Contents

bGMYC	2
bgmyc.dataprep	3
bgmyc.gibbs	4
bgmyc.lik	6
bgmyc.multiphylo	7
bgmyc.point	9
bgmyc.prior	10
bgmyc.singlephy	11
bgmyc.spec	13
checkrates	14
east10	15
gmyc.5parameter.lik	16
plot.bgmycprobmata	16
plot.bgmycrates	17
plot.interactive.probmata	18
plot.multibgmyc	19
plot.singlebgmyc	20
spec.probmata	21
Index	22

bGMYC	<i>A Bayesian MCMC implementation of the general mixed Yule-coalescent model for species delimitation.</i>
-------	--

Description

This package implements a Bayesian version of the general mixed Yule-coalescent model for species delimitation introduced by Pons et al. 2006. It uses Markov Chain Monte Carlo simulation to evaluate the posterior distribution of the model and uses a flexible method of specifying prior distributions on the model parameters. The primary strength of this method over the pre-existing implementation is that it allows the use of multiple trees sampled from an MCMC analysis, eliminating the conditioning of results on a single point estimate of the phylogenetic tree, which often is associated with high uncertainty.

Details

Package:	bGMYC4
Type:	Package
Version:	4.1.0
Date:	2026-05-01
License:	GPL (>= 3)
LazyData:	true

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

References

Pons et al. 2006 Reid and Carstens 2012

Examples

```
# For user data, ape tree format is required.
# Easiest to input a nexus tree file using ape function "read.nexus".
# In this case, load test data, an ape multiPhylo object.
data(east10)

# Two primary functions for running analysis:
# bgmyc.singlephy and bgmyc.multiphylo.
# Run bGMYC on a single tree (phylo object).
result.single <- bgmyc.singlephy(phylo = east10[[1]], mcmc = 100, burnin = 1, thinning = 10)

# Visualize MCMC output.
plot(result.single)

# Run bGMYC on multiple trees (multiphylo object).
# Wrapped in \donttest{} to bypass CRAN check parallel worker limits.

result.multi <- bgmyc.multiphylo(multiphylo = east10[1:2], mcmc = 100, burnin = 1, thinning = 10)

# For multiple trees, visualize MCMC output OR pool samples
# from across trees into a single matrix using plot=FALSE.
plot(result.multi, plot = FALSE) -> parameter.matrix

# Get the posterior probabilities of each possible species cluster.
bgmyc.spec(res = result.multi) -> result.spec

# Produce a sequence by sequence matrix of probabilities of conspecificity.
spec.probmat(result.multi) -> result.probmat

# Plot the matrix of conspecificity probabilities.
# Provide a single tree to order the matrix.
plot(result.probmat, east10[[1]])
```

bgmyc.dataprep

Prepares a tree for analysis.

Description

Takes an ape "phylo" object and turns it into input for downstream functions.

Usage

```
bgmyc.dataprep(tr)
```

Arguments

tr an ape "phylo" object.

Value

A list with the following components:

mrca.nodes	list of MRCA node indices for each threshold
nod.types	list of node type vectors for each threshold
n	list of the number of coalescent entities for each threshold
list.s.nod	list of matrices indicating node states
list.i.mat	list of interval matrices used in likelihood calculation
internod	vector of internode distances
tree	the input phylogenetic tree

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

bgmyc.gibbs

MCMC sampler

Description

This function samples from the GMYC model using output from bgmyc.dataprep as data.

Usage

```
bgmyc.gibbs(data, m, burnin = 1, thinning = 1,
             py1, py2, pc1, pc2, t1, t2,
             scale = c(20, 10, 5),
             start = c(1, 0.5, 50),
             likelihood, prior)
```

Arguments

data	output from bgmyc.dataprep
m	number of samples to take from the Markov Chain
burnin	the number of samples to discard as burn-in
thinning	the interval at which samples are retained from the Markov Chain

py1	governs the prior on the Yule (speciation) rate change parameter. using the default prior distribution, this is the lower bound of a uniform distribution. this can be the most influential prior of the three. rate change is parameterized as n^{py} where n is the number of lineages in a waiting interval. if there are 50 sequences in an analysis and the Yule rate change parameter is 2, this allows for a potential 50-fold increase in speciation rate. this unrealistic parameter value can cause the threshold between Yule and Coalescent process to be difficult to distinguish. are more reasonable upper bound for the prior would probably be less than 1.5 (a potential 7-fold increase). Or you could modify the prior function to use a different distribution entirely.
py2	governs the prior on the Yule rate change parameter. using the default prior distribution, this is the upper bound of a uniform distribution.
pc1	governs the prior on the coalescent rate change parameter. using the default prior distribution, this is the lower bound of a uniform distribution. rate change is parameterized as $(n(n-1))^{pc}$ where n is the number of lineages in a waiting interval (see Pons et al. 2006). In principle pc can be interpreted as change in effective population size ($pc < 1$ decline, $pc > 1$ growth) but because identical haplotypes must be excluded from this analysis an accurate biological interpretation is not possible.
pc2	governs the prior on the coalescent rate change parameter. using the default prior distribution, this is the upper bound of a uniform distribution.
t1	governs the prior on the threshold parameter. the lower bound of a uniform distribution. the bounds of this uniform distribution should not be below 1 or greater than the number of unique haplotypes in the analysis.
t2	governs the prior on the threshold parameter. the upper bound of a uniform distribution
scale	a vector of scale parameters governing the proposal distributions for the markov chain. the first to are the Yule and coalescent rate change parameters. increasing them makes the proposals more conservative. the third is the threshold parameter. increasing it makes the proposals more liberal.
start	a vector of starting parameters in the same order as the scale parameters, py, pc, t. t may need to be set so that it is not impossible given the dataset.
likelihood	a function that calculates the likelihood of the data given the model and parameters. in this case, bgmyc.lik is the only function provided that will work.
prior	a function that returns the prior probability of each parameter value. in this case bgmyc.prior is the only function provided. that function could in principle be easily modified and a new one used here. e.g., replace the uniform distribution density functions (dunif) with gamma or lognormal distributions (dgamma, dlnorm).

Details

this function is called by bgmyc.singlephy and bgmyc.multiphylo, so there probably isn't any reason to use it on its own.

Value

Returns a list with two primary elements.

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

Examples

```
#prepare a tree for analysis:
data(east10)
bgmyc.dataprep(east10[[1]])->input

# analyze the tree
output <- bgmyc.gibbs(input, 100, burnin = 1, thinning = 1,
                      py1 = 0, py2 = 1.5, pc1 = 0, pc2 = 2,
                      t1 = 2, t2 = 75, scale = c(20, 10, 5),
                      start = c(1, 0.5, 50),
                      likelihood = bgmyc.lik,
                      prior = bgmyc.prior)
```

bgmyc.lik

GMYC likelihood function

Description

Returns the log likelihood of the model given the data and parameter values.

Usage

```
bgmyc.lik(params, data)
```

Arguments

params	a vector of parameter values to be evaluated in the order: py, pc, t
data	an output from gmyc.dataprep

Details

there probably isn't any reason to use this function on its own, but it can be optimized using "optim"

Value

returns the log likelihood of the model given data and parameter values

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

bgmyc.multiphylo *Multi-tree GMYC analysis*

Description

conducts MCMC analysis for multiple trees.

Usage

```
bgmyc.multiphylo(multiphylo, mcmc, burnin, thinning,
                 py1 = 0, py2 = 2, pc1 = 0, pc2 = 2,
                 t1 = 2, t2 = 51, scale = c(20, 10, 5),
                 start = c(1, 0.5, 50),
                 sampler = bgmyc.gibbs,
                 likelihood = bgmyc.lik, prior = bgmyc.prior)
```

Arguments

<code>multiphylo</code>	an ape "multiphylo" object
<code>mcmc</code>	number of samples to take from the Markov Chain
<code>burnin</code>	the number of samples to discard as burn-in
<code>thinning</code>	the interval at which samples are retained from the Markov Chain
<code>py1</code>	governs the prior on the Yule (speciation) rate change parameter. using the default prior distribution, this is the lower bound of a uniform distribution. this can be the most influential prior of the three. rate change is parameterized as n^{py} where n is the number of lineages in a waiting interval (see Pons et al. 2006). if there are 50 sequences in an analysis and the Yule rate change parameter is 2, this allows for a potential 50-fold increase in speciation rate. this unrealistic parameter value can cause the threshold between Yule and Coalescent process to be difficult to distinguish. are more reasonable upper bound for the prior would probably be less than 1.5 (a potential 7-fold increase). Or you could modify the prior function to use a different distribution entirely.
<code>py2</code>	governs the prior on the Yule rate change parameter. using the default prior distribution, this is the upper bound of a uniform distribution.
<code>pc1</code>	governs the prior on the coalescent rate change parameter. using the default prior distribution, this is the lower bound of a uniform distribution. rate change is parameterized as $(n(n-1))^{pc}$ where n is the number of lineages in a waiting interval (see Pons et al. 2006). In principle pc can be interpreted as change in effective population size ($pc < 1$ decline, $pc > 1$ growth) but because identical haplotypes must be excluded from this analysis an accurate biological interpretation is not possible.
<code>pc2</code>	governs the prior on the coalescent rate change parameter. using the default prior distribution, this is the upper bound of a uniform distribution.

<code>t1</code>	governs the prior on the threshold parameter. the lower bound of a uniform distribution. the bounds of this uniform distribution should not be below 1 or greater than the number of unique haplotypes in the analysis.
<code>t2</code>	governs the prior on the threshold parameter. the upper bound of a uniform distribution.
<code>scale</code>	a vector of scale parameters governing the proposal distributions for the markov chain. the first to are the Yule and coalescent rate change parameters. increasing them makes the proposals more conservative. the third is the threshold parameter. increasing it makes the proposals more liberal.
<code>start</code>	a vector of starting parameters in the same order as the scale parameters, <code>py</code> , <code>pc</code> , <code>t</code> . <code>t</code> may need to be set so that it is not impossible given the dataset.
<code>sampler</code>	an MCMC sampler function. the only function provided is <code>bgmyc.gibbs</code> .
<code>likelihood</code>	a function that calculates the likelihood of the data given the model and parameters. in this case, <code>bgmyc.lik</code> is the only function provided that will work.
<code>prior</code>	a function that returns the prior probability of each parameter value. in this case <code>bgmyc.prior</code> is the only function provided. that function could in principle be easily modified and a new one used here. e.g., replace the uniform distribution density functions (<code>dunif</code>) with gamma or lognormal distributions (<code>dgamma</code> , <code>dlnorm</code>).

Details

conducts Markov Chain Monte Carlo sampling for the GMYC model for each of the trees in a "multiphylo" object. MCMC output should be checked for non-stationary behavior using `plot(output)`. A matrix containing MCMC samples pooled across all trees can be obtained using `plot(output, plot=FALSE)`

Value

a list of class "multibgmyc" containing a series of objects of class "singlebgmyc" each of which contains MCMC samples from each tree.

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

Examples

```
# Load test data
data(east10)

# Single tree analysis
result.single <- bgmyc.singlephy(
  phylo = east10[[1]],
  mcmc = 100, burnin = 1, thinning = 10
)
plot(result.single)
```

```
# Multi-tree analysis (explicitly 2 trees to satisfy CRAN parallel limits)
result.multi <- bgmyc.multiphylo(
  multiphylo = east10[1:2],
  mcmc = 100, burnin = 1, thinning = 10
)

# Post-processing
probmata <- spec.probmata(result.multi)
```

bgmyc.point	<i>Returns a point estimate of species limits based on a posterior probability threshold input by the user.</i>
-------------	---

Description

This function processes output from a bgmyc analysis and produces a single point estimate of species limits based on a posterior probability threshold.

Usage

```
bgmyc.point(probmata, ppcutoff)
```

Arguments

probmata	output from function spec.probmata of class "bgmycprobmata"
ppcutoff	a posterior probability threshold for lumping samples into species. e.g. if 0.05 were selected, all individuals having greater than 0.05 posterior probability of conspecificity will be lumped into species returned.

Value

Returns an R list where each entry gives the members of a species that meets the threshold. It is worth noting that sometimes, because of uncertainty in the phylogenetic tree, not all individuals in a "species" resulting from a given threshold will have greater than the specified probability of conspecificity. They can all be connected, however, through individuals whose probability of conspecificity is greater than the threshold.

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

bgmyc.prior

a function to return the prior probability of a model.

Description

Given model parameter values and parameters governing the prior distribution for each, this function returns prior probability of the model.

Usage

```
bgmyc.prior(params, py1, py2, pc1, pc2, t1, t2)
```

Arguments

params	parameter values for which the function returns prior probabilities. in the order: py, pc, t
py1	governs the prior on the Yule (speciation) rate change parameter. using the default prior distribution, this is the lower bound of a uniform distribution. this can be the most influential prior of the three. rate change is parameterized as n^{py} where n is the number of lineages in a waiting interval (see Pons et al. 2006). if there are 50 sequences in an analysis and the Yule rate change parameter is 2, this allows for a potential 50-fold increase in speciation rate. this unrealistic parameter value can cause the threshold between Yule and Coalescent process to be difficult to distinguish. are more reasonable upper bound for the prior would probably be less than 1.5 (a potential 7-fold increase). Or you could modify the prior function to use a different distribution entirely.
py2	governs the prior on the Yule rate change parameter. using the default prior distribution, this is the upper bound of a uniform distribution.
pc1	governs the prior on the coalescent rate change parameter. using the default prior distribution, this is the lower bound of a uniform distribution. rate change is parameterized as $(n(n-1))^{pc}$ where n is the number of lineages in a waiting interval (see Pons et al. 2006). In principle pc can be interpreted as change in effective population size ($pc < 1$ decline, $pc > 1$ growth) but because identical haplotypes must be excluded from this analysis an accurate biological interpretation is not possible.
pc2	governs the prior on the coalescent rate change parameter. using the default prior distribution, this is the upper bound of a uniform distribution.
t1	governs the prior on the threshold parameter. the lower bound of a uniform distribution. the bounds of this uniform distribution should not be below 1 or greater than the number of unique haplotypes in the analysis.
t2	governs the prior on the threshold parameter. the upper bound of a uniform distribution

Details

this function is used by bgmyc.gibbs, bgmyc.singlephy and bgmyc.multiphylo to get model prior probabilities in the MCMC. each parameter prior is a uniform distribution whose parameters are set as above. this function can be easily modified to use either gamma or lognormal priors and should fit smoothly into the other existing functions.

Value

returns a log prior probability

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

bgmyc.singlephy	<i>Single tree GMYC analysis.</i>
-----------------	-----------------------------------

Description

Single tree Bayesian GMYC analysis using MCMC sampling.

Usage

```
bgmyc.singlephy(phylo, mcmc, burnin, thinning, py1 = 0, py2 = 2, pc1 = 0, pc2 = 2,
  t1 = 2, t2 = 51, scale = c(20, 10, 5), start = c(1, 0.5, 50),
  sampler = bgmyc.gibbs, likelihood = bgmyc.lik, prior = bgmyc.prior)
```

Arguments

phylo	an ape "phylo" object
mcmc	number of samples to take from the Markov Chain
burnin	the number of samples to discard as burn-in
thinning	the interval at which samples are retained from the Markov Chain
py1	governs the prior on the Yule (speciation) rate change parameter. using the default prior distribution, this is the lower bound of a uniform distribution. this can be the most influential prior of the three. rate change is parameterized as n^{py1} where n is the number of lineages in a waiting interval (see Pons et al. 2006). if there are 50 sequences in an analysis and the Yule rate change parameter is 2, this allows for a potential 50-fold increase in speciation rate. this unrealistic parameter value can cause the threshold between Yule and Coalescent process to be difficult to distinguish. are more reasonable upper bound for the prior would probably be less than 1.5 (a potential 7-fold increase). Or you could modify the prior function to use a different distribution entirely.
py2	governs the prior on the Yule rate change parameter. using the default prior distribution, this is the upper bound of a uniform distribution.

<code>pc1</code>	governs the prior on the coalescent rate change parameter. using the default prior distribution, this is the lower bound of a uniform distribution. rate change is parameterized as $(n(n-1))^{pc}$ where n is the number of lineages in a waiting interval (see Pons et al. 2006). In principle pc can be interpreted as change in effective population size ($pc < 1$ decline, $pc > 1$ growth) but because identical haplotypes must be excluded from this analysis an accurate biological interpretation is not possible.
<code>pc2</code>	governs the prior on the coalescent rate change parameter. using the default prior distribution, this is the upper bound of a uniform distribution.
<code>t1</code>	governs the prior on the threshold parameter. the lower bound of a uniform distribution. the bounds of this uniform distribution should not be below 1 or greater than the number of unique haplotypes in the analysis.
<code>t2</code>	governs the prior on the threshold parameter. the upper bound of a uniform distribution
<code>scale</code>	a vector of scale parameters governing the proposal distributions for the markov chain. the first to are the Yule and coalescent rate change parameters. increasing them makes the proposals more conservative. the third is the threshold parameter. increasing it makes the proposals more liberal.
<code>start</code>	a vector of starting parameters in the same order as the scale parameters, <code>py</code> , <code>pc</code> , <code>t</code> . <code>t</code> may need to be set so that it is not impossible given the dataset.
<code>sampler</code>	an MCMC sampler function. the only function provided is <code>bgmyc.gibbs</code> .
<code>likelihood</code>	a function that calculates the likelihood of the data given the model and parameters. in this case, <code>bgmyc.lik</code> is the only function provided that will work.
<code>prior</code>	a function that returns the prior probability of each parameter value. in this case <code>bgmyc.prior</code> is the only function provided. that function could in principle be easily modified and a new one used here. e.g., replace the uniform distribution density functions (<code>dunif</code>) with gamma or lognormal distributions (<code>dgamma</code> , <code>dlnorm</code>).

Details

conducts Markov Chain Monte Carlo sampling for the GMYC model an ape "phylo" object. MCMC output should be checked for non-stationary behavior using `plot(output)`.

Value

Returns a list with two primary elements.

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

bgmyc.spec	<i>bGMYC output processing.</i>
------------	---------------------------------

Description

Processes output from bgmyc.singlephy or bgmyc.multiphylo.

Usage

```
bgmyc.spec(res, filename = NULL, cmatrix = NULL)
```

Arguments

res	output from bgmyc.singlephy or bgmyc.multiphylo
filename	if given, a file that contains all species sampled in the MCMC and their posterior probabilities
cmatrix	this option may not work at the moment. a matrix of tree leaves by sampling localities. if given, returns a matrix of species by sampling localities suitable for community ecological analysis

Details

this function outputs a list with one or two components. the first is a data frame containing all species sampled in the MCMC and their posterior probabilities (also optionally output to a text file if "filename" is given) and the second is an optional collection of matrices if "cmatrix" is specified. cmatrix should be a tree leaves by sampling locality matrix and if given, a collection of species by sampling locality matrices are returned that represent the posterior distribution of species occurrences.

Value

A list with one or two components:

specprobs	a data frame containing all species sampled in the MCMC and their posterior probabilities
cmatrix	(optional) a collection of species by sampling locality matrices if cmatrix argument is specified

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

checkrates	<i>bGMYC output processing.</i>
------------	---------------------------------

Description

Processes output from `bgmyc.singlephy` or `bgmyc.multiphylo`.

Usage

```
checkrates(result)
```

Arguments

<code>result</code>	output from <code>bgmyc.singlephy</code> or <code>bgmyc.multiphylo</code>
---------------------	---

Details

This function takes the results of a `bgmyc.singlephy` or `bgmyc.multiphylo` run and outputs a matrix of all parameters, including the branching rates for the Yule and coalescent processes (this is the only place to get those last two). The branching rates are may be useful in evaluating the reliability of the model output (although I haven't tested this extensively). The GMYC is based on the idea that coalescent and speciation events happen at drastically different (perhaps on around an order of magnitude different) rates and thus can be differentiated in a phylogenetic tree. If the rates are similar, then the threshold point in the model might be relatively indistinct and the results unreliable. Comparison of the rates might help identify this.

Value

An object of class "bgmycrates": a matrix with columns:

<code>p.div</code>	Yule rate change parameter
<code>p.coal</code>	coalescent rate change parameter
<code>threshold</code>	threshold parameter (number of species)
<code>lambda.div</code>	Yule branching rate
<code>lambda.coal</code>	coalescent branching rate
<code>lambda.div.mod</code>	modified Yule branching rate

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

east10	<i>10 trees sampled from a posterior distribution. based on simulated data.</i>
--------	---

Description

A multiPhylo object containing 10 trees sampled from the posterior distribution of a simulated dataset.

Usage

```
data(east10)
```

Format

The format is: Class "multiPhylo" List of 10 \$:List of 3 ..\$ edge : int [1:428, 1:2] 216 217 218 219 219 218 220 220 217 221\$ edge.length: num [1:428] 9.53e-04 6.80e-02 8.75e-05 6.01e-03 6.01e-03\$ Nnode : int 214 ..- attr(*, "class")= chr "phylo" \$:List of 3 ..\$ edge : int [1:428, 1:2] 216 217 218 218 219 219 217 216 220 221\$ edge.length: num [1:428] 7.07e-02 1.23e-05 4.38e-03 2.15e-03 2.23e-03\$ Nnode : int 214 ..- attr(*, "class")= chr "phylo" \$:List of 3 ..\$ edge : int [1:428, 1:2] 216 217 218 219 220 221 222 222 221 220\$ edge.length: num [1:428] 0.00086 0.02685 0.0232 0.02339 0.00457\$ Nnode : int 214 ..- attr(*, "class")= chr "phylo" \$:List of 3 ..\$ edge : int [1:428, 1:2] 216 217 218 219 220 221 222 222 221 220\$ edge.length: num [1:428] 0.00652 0.01154 0.02245 0.02312 0.0024\$ Nnode : int 214 ..- attr(*, "class")= chr "phylo" \$:List of 3 ..\$ edge : int [1:428, 1:2] 216 217 217 218 219 219 218 216 220 221\$ edge.length: num [1:428] 0.087911 0.005809 0.001012 0.000921 0.003876\$ Nnode : int 214 ..- attr(*, "class")= chr "phylo" \$:List of 3 ..\$ edge : int [1:428, 1:2] 216 217 218 219 220 221 221 222 222 220\$ edge.length: num [1:428] 0.02239 0.02818 0.04245 0.01276 0.00154\$ Nnode : int 214 ..- attr(*, "class")= chr "phylo" \$:List of 3 ..\$ edge : int [1:428, 1:2] 216 217 218 219 220 220 221 221 222 222\$ edge.length: num [1:428] 0.0026 0.04503 0.02079 0.00618 0.00526\$ Nnode : int 214 ..- attr(*, "class")= chr "phylo" \$:List of 3 ..\$ edge : int [1:428, 1:2] 216 217 218 219 220 221 221 222 222 220\$ edge.length: num [1:428] 0.00762 0.02619 0.01898 0.01848 0.00591\$ Nnode : int 214 ..- attr(*, "class")= chr "phylo" \$:List of 3 ..\$ edge : int [1:428, 1:2] 216 217 217 218 218 219 219 216 220 221\$ edge.length: num [1:428] 0.07977 0.00602 0.00214 0.00388 0.00139\$ Nnode : int 214 ..- attr(*, "class")= chr "phylo" \$:List of 3 ..\$ edge : int [1:428, 1:2] 216 217 218 219 220 221 222 223 223 224\$ edge.length: num [1:428] 0.0009 0.01292 0.01373 0.02408 0.00663\$ Nnode : int 214 ..- attr(*, "class")= chr "phylo" - attr(*, "TipLabel")= chr [1:215] "JP_133_isolate_133_1" "JP_137_isolate_137_1" "JP_137_isolate_137_2" "JP_137_isolate_137_3" ...

Examples

```
data(east10)
## maybe str(east10) ; plot(east10) ...
```

gmyc.5parameter.lik *5 parameter likelihood function.*

Description

Returns the likelihood of the GMYC model if all 5 parameters are free to vary.

Usage

```
gmyc.5parameter.lik(params, data)
```

Arguments

params	a vector of five the five parameters in the model in the order y, c, py, pc, t
data	output from gmyc.dataprep

Details

The Pons et al. implementation of the GMYC (and this one) treat the model as if it had 3 parameters, a Yule rate change parameter, a coalescent rate change parameter and a threshold parameter. The Yule and coalescent rates are fixed to an ML estimator at each possible threshold and not integrated over. Allowing all 5 parameters to vary freely in some cases yields a flatter likelihood surface which in some cases (in simulation) has a peak very far away from the true value and very different from the 3 parameter implementation. Using the ML estimator for the rate parameters yields empirically better performance so we use it in our paper (Reid and Carstens 2012), but it took some time to figure that out. We include this function here for anyone who feels like messing with it.

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

plot.bgmycprobmata *Plots the matrix of probability of conspecificity.*

Description

Visualizes the conspecificity probability matrix alongside the phylogenetic tree. Enhanced version with customizable color palettes and export options.

Usage

```
## S3 method for class 'bgmycprobmata'
plot(x, tree, palette = "classic", save_pdf = NULL, legend_cex = 0.5, ...)
```

Arguments

x	A matrix of conspecificity probabilities (output of spec.probmat)
tree	A phylogenetic tree (phylo object), must match the matrix dimensions
palette	Character; color palette (default: "classic")
save_pdf	Character or NULL; output PDF path
legend_cex	Numeric; legend text size (default: 0.5)
...	Additional graphical parameters (currently unused)

Details

A plotting function that visualizes the matrix of probabilities of conspecificity.

Value

None; produces a plot as a side effect

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

Examples

```
data(east10)
set.seed(1)
result <- bgmyc.singlephy(east10[[1]], mcmc = 1000, burnin = 200, thinning = 10)
prob_mat <- spec.probmat(result)
p <- plot.interactive.probmat(prob_mat, east10[[1]])
```

plot.bgmycrates	<i>Plots relative coalescent and Yule rates for model fit evaluation.</i>
-----------------	---

Description

Plots relative coalescent and Yule rates for model fit evaluation.

Usage

```
## S3 method for class 'bgmycrates'
plot(x, ...)
```

Arguments

x	output from checkrates function.
...	additional arguments passed to plot.

Details

plots relative coalescent and Yule rates for model fit evaluation

Value

No return value, called for side effects (produces a 2x2 panel of diagnostic plots).

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

plot.interactive.probmat

Interactive Probability Matrix Plot for bGMYC Results

Description

Creates an interactive HTML visualization of conspecificity probability matrix from bGMYC analysis, with optional posterior-colored phylogenetic tree. Supports two modes: standard (PNG-based tree overlay) and custom (vector-based subplot with interactive branches).

Usage

```
## S3 method for class 'interactive.probmat'
plot(
  x,
  tree,
  palette = "green",
  show_tree = TRUE,
  tree_width = 0.3,
  width = 1400,
  height = 1000,
  save_html = NULL,
  custom_mode = TRUE,
  posterior_col = NULL,
  branch_gradient = NULL,
  tip_label_step = NULL,
  treeio_tree = NULL,
  ...
)
```

Arguments

x	A probability matrix (output from spec.probmat).
tree	A phylogenetic tree of class phylo.
palette	Color palette for heatmap: "green", "viridis", "RdYIBu", or "classic". Default: "green".

show_tree	Logical: whether to display phylogenetic tree alongside matrix. Default: TRUE.
tree_width	Relative width of tree panel (0.1–0.5) when show_tree = TRUE. Default: 0.3.
width	Plot width in pixels. Default: 1400.
height	Plot height in pixels. Default: 1000.
save_html	Optional path to save interactive plot as self-contained HTML file.
custom_mode	Logical: enable advanced vector-based visualization with posterior-colored branches? Default: FALSE (backward compatible). TRUE for publication-quality output with interactive tree.
posterior_col	Optional name of column containing posterior probabilities in tree annotations. If NULL, auto-detects from c("posterior", "prob", "Posterior", "PROB").
branch_gradient	Optional list of color gradient control points for branch coloring. Format: list(c(position, R, G, B), ...). Default: red→orange→yellow→lightgreen→green ramp.
tip_label_step	Step for sparse tip labeling (e.g., 40 = label every 40th tip). If NULL, auto-calculated as max(1, ceiling(n_tips / 40)).
treeio_tree	Optional pre-loaded treedata object (from treeio::read.beast). Avoids re-reading file when using custom_mode = TRUE.
...	Additional arguments passed to plotting functions.

Value

An invisible plotly object. If save_html is specified, also saves to file.

Author(s)

Dmitry Karabanov <dk@ibiw.ru>

Examples

```
data(east10)
set.seed(1)
result <- bgmyc.singlephy(east10[[1]], mcmc = 1000, burnin = 200, thinning = 10)
prob_mat <- spec.probmat(result)
p <- plot.interactive.probmat(prob_mat, east10[[1]])
```

plot.multibgmyc

Plot MCMC output.

Description

Plots pooled MCMC output from multiple trees.

Usage

```
## S3 method for class 'multibgmyc'
plot(x, plot = TRUE, ...)
```

Arguments

x	object of class "multibgmyc", output from bgmyc.multiphylo.
plot	logical. If TRUE, produces trace plots. If FALSE, returns a matrix of pooled parameters.
...	additional arguments passed to plot.

Value

If plot = TRUE: no return value, called for side effects (produces a 2x2 panel of trace plots). If plot = FALSE: a matrix of pooled MCMC parameters from all trees.

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

plot.singlebgmyc	<i>Plot MCMC output.</i>
------------------	--------------------------

Description

Plots trace plots of MCMC parameters for a single tree analysis.

Usage

```
## S3 method for class 'singlebgmyc'
plot(x, burnin = 0, thinning = 1, ...)
```

Arguments

x	object of class "singlebgmyc", output from bgmyc.singlephy.
burnin	the number of samples to discard as burn-in.
thinning	the interval at which samples are retained from the Markov Chain.
...	additional arguments passed to plot.

Value

No return value, called for side effects (produces a 2x2 panel of trace plots).

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

spec.probmat	<i>Return matrix of probability of conspecificity.</i>
--------------	--

Description

This function processes output from bgmyc analysis and yields a matrix of the probability of conspecificity for each pair of leaves in the given distribution of trees.

Usage

```
spec.probmat(res)
```

Arguments

res	output from function bgmyc.spec
-----	---------------------------------

Value

A matrix of tree leaves by tree leaves whose values are the probability of conspecificity.

Author(s)

Noah M. Reid Maintainer: Dmitry Karabanov <dk@ibiw.ru>

Index

- * **aplot**
 - plot.interactive.probmat, [18](#)
- * **datasets**
 - east10, [15](#)
- * **hplot**
 - plot.interactive.probmat, [18](#)
- * **package**
 - bGMYC, [2](#)

bGMYC, [2](#)
bgmyc.dataprep, [3](#)
bgmyc.gibbs, [4](#)
bgmyc.lik, [6](#)
bgmyc.multiphylo, [7](#)
bgmyc.point, [9](#)
bgmyc.prior, [10](#)
bgmyc.singlephy, [11](#)
bgmyc.spec, [13](#)

checkrates, [14](#)

east10, [15](#)

gmyc.5parameter.lik, [16](#)

plot.bgmycprobmat, [16](#)
plot.bgmycrates, [17](#)
plot.interactive.probmat, [18](#)
plot.multibgmyc, [19](#)
plot.singlebgmyc, [20](#)

spec.probmat, [21](#)