

# Package ‘markophylo’

July 5, 2020

**Type** Package

**Title** Markov Chain Models for Phylogenetic Trees

**Version** 1.0.8

**Date** 2020-07-05

**Author** Utkarsh J. Dang and G. Brian Golding

**Maintainer** Utkarsh J. Dang <udang@binghamton.edu>

**Description** Allows for fitting of maximum likelihood models using Markov chains on phylogenetic trees for analysis of discrete character data. Examples of such discrete character data include restriction sites, gene family presence/absence, intron presence/absence, and gene family size data. Hypothesis-driven user-specified substitution rate matrices can be estimated. Allows for biologically realistic models combining constrained substitution rate matrices, site rate variation, site partitioning, branch-specific rates, allowing for non-stationary prior root probabilities, correcting for sampling bias, etc. See Dang and Golding (2016) <doi:10.1093/bioinformatics/btv541> for more details.

**License** GPL (>= 2)

**Imports** Rcpp (>= 0.11.4), ape (>= 3.2), numDeriv (>= 2012.9.1), phangorn (>= 1.99.13)

**LinkingTo** Rcpp, RcppArmadillo

**Suggests** knitr (>= 1.10), testthat (>= 0.9.1), markdown (> 1.0)

**Depends** R (>= 2.10)

**VignetteBuilder** knitr

**LazyData** true

**NeedsCompilation** yes

**Repository** CRAN

**Date/Publication** 2020-07-05 17:40:11 UTC

## R topics documented:

markophylo-package . . . . . 2

estimates	3
estimates_f	8
patterns	11
plotree	12
print.markophylo	13
simdata1	14

<b>Index</b>	<b>15</b>
--------------	-----------

---

markophylo-package	<i>Markov chain models for phylogenetic trees.</i>
--------------------	--

---

## Description

Allows for fitting of maximum likelihood models using Markov chains on phylogenetic trees for analysis of discrete character data. Examples of such discrete character data include restriction sites, gene family presence/absence, intron presence/absence, and gene family size data. Hypothesis-driven user-specified substitution rate matrices can be estimated. Allows for biologically realistic models combining constrained substitution rate matrices, site rate variation, site partitioning, branch-specific rates, allowing for non-stationary prior root probabilities, correcting for sampling bias, etc.

## Details

Package: markophylo  
 Type: Package  
 Version: 1.0.5  
 Date: 2016-01-19  
 License: GPL (>= 2)

Function `estimates` is the main workhorse of the package, allowing users to specify a substitution rate matrix and estimate it. Partition analysis, gamma rate variation, estimating root frequencies of the discrete characters, and specifying clades (or a set of branches) with their own unique rates are all possible.

## Author(s)

Utkarsh J. Dang and G. Brian Golding  
 <udang@mcmaster.ca>

## References

Dirk Eddelbuettel and Conrad Sanderson (2014). RcppArmadillo: Accelerating R with high-performance C++ linear algebra. *Computational Statistics and Data Analysis*, 71, 1054–1063.

Eddelbuettel, Dirk and Romain Francois (2011). Rcpp: Seamless R and C++ Integration. *Journal of Statistical Software*, 40(8), 1–18.

Gilbert, Paul and Ravi Varadhan (2012). numDeriv: Accurate Numerical Derivatives. R package version 2014.2-1.

Harmon Luke J, Jason T Weir, Chad D Brock, Richard E Glor, and Wendell Challenger. 2008. GEIGER: investigating evolutionary radiations. *Bioinformatics* 24:129-131.

Paradis, E., Claude, J. & Strimmer, K. 2004. APE: analyses of phylogenetics and evolution in R language. *Bioinformatics* 20: 289-290. R package version 3.2.

Schliep, K.P. 2011. phangorn: phylogenetic analysis in R. *Bioinformatics*, 27(4) 592-593. R package version 1.99.13.

Yang, Z. (1994). Maximum likelihood phylogenetic estimation from DNA sequences with variable rates over sites: approximate methods. *Journal of Molecular evolution*, 39(3), 306–314.

---

 estimaterrates

---

*Estimate substitution rate matrix.*


---

## Description

Estimate a substitution rate matrix. Users can specify a substitution rate matrix and estimate it. Partition analysis, gamma rate variation, estimating root frequencies of the discrete characters, and specifying clades (or a set of branches) with their own unique rates are also possible. See arguments below. See vignette for detailed examples.

## Usage

```
estimaterrates(userTree = NULL, userPhyl = NULL, matchTipstoData = FALSE,
               unobserved = NULL, alphabet = NULL, modelmat = NULL,
               bgType = "listofnodes", bg = NULL, partition = NULL,
               ratevar = FALSE, nocat = 4, reversible = FALSE, numhessian = TRUE,
               rootprob = NULL, rpvec = NULL, init = 0.9, lowli = 0.001, upli = 100, ...)
```

## Arguments

**userTree** Rooted binary tree of class "phylo". Read in Newick tree using read.tree() from package ape before passing this argument. The branch lengths must be in expected substitutions per site (but see lowli and upli arguments). Trees estimated using MrBayes and BEAST, for instance, yield branch lengths using that scale. If an unrooted tree is available, look into the "root" and "midpoint.root" functions from the APE and phytools packages, respectively.

**userPhyl** A matrix or data frame of phyletic patterns. Rows represent the discrete character patterns and columns represent the taxa. These data can be numeric or character (but not both).

**matchTipstoData**

The default is FALSE, which means that the user must ensure that the ordering of the taxa in the data matrix must match the internal ordering of tip labels of the tree. Set to TRUE, if the column names of the data matrix, i.e., the taxa names, are all present, with the same spelling and notation in the tip labels of the tree provided, in which case, the restriction on ordering is not necessary.

unobserved	A matrix of unobserved phyletic patterns, representing possible sampling or acquisition bias. Each row should be a unique phyletic pattern.
alphabet	The set of discrete characters used. May be integer only or character only.
modelmat	<p>Can be one of two options: pre-built or user-specified. For the pre-built matrices, use:</p> <ul style="list-style-type: none"> <li>• "ER" for an equal rates matrix, e.g., <code>matrix(c(NA, 1, 1, 1, 1, NA, 1, 1, 1, 1, NA, 1, 1, 1, 1, NA), nrow = 4, ncol = 4)</code></li> <li>• "SYM" for a symmetric matrix, e.g., <code>matrix(c(NA, 1, 2, 3, 1, NA, 4, 5, 2, 4, NA, 6, 3, 5, 6, NA), 4, 4)</code></li> <li>• "ARD" for an all rates different matrix, e.g., <code>matrix(c(NA, 1, 2, 3, 4, NA, 5, 6, 7, 8, NA, 9, 10, 11, 12, NA), 4, 4)</code></li> <li>• "GTR" for a general time reversible model. This sets reversible to be TRUE, rootprob to be "maxlik", and the rate matrix supplied to the function is symmetric. The estimated rate matrix can be written as the product of a symmetric matrix multiplied by a diagonal matrix (consisting of the estimated root probabilities).</li> <li>• "BD" for a standard birth death matrix, e.g., <code>matrix(c(NA, 1, 0, 0, 2, NA, 1, 0, 0, 2, NA, 1, 0, 0, 2, NA), 4, 4)</code></li> <li>• "BDER" for a standard birth death matrix, e.g., <code>matrix(c(NA, 1, 0, 0, 1, NA, 1, 0, 0, 1, NA, 1, 0, 0, 1, NA), 4, 4)</code></li> <li>• "BDSYM" for a standard birth death matrix, e.g., <code>matrix(c(NA, 1, 0, 0, 1, NA, 2, 0, 0, 2, NA, 3, 0, 0, 3, NA), 4, 4)</code></li> <li>• "BDARD" for an all rates different birth death matrix, e.g., <code>matrix(c(NA, 1, 0, 0, 4, NA, 2, 0, 0, 5, NA, 3, 0, 0, 6, NA), 4, 4)</code></li> </ul> <p>These pre-built matrices are inspired by the APE and DiscML packages.</p> <p>A square matrix can also be input that consists of integer values denoting the indices of the substitution rates to be estimated. The number of rows and columns corresponds to the total number of discrete states possible in the data. For example, using <code>matrix(c(NA, 1, 2, NA), 2, 2)</code> means that two rates must be estimated corresponding to the entries 1 and 2, respectively. Using <code>matrix(c(NA, 1, 0, NA), 2, 2)</code> means that only one rate must be estimated and the entry 0 corresponds to a substitution that is not permitted (hence, does not need to be estimated). See examples and vignette for more examples.</p>
bgtype	Use this to group branches hypothesized to follow the same rates (but differ from other branches). If clade-specific insertion and deletion rates are required to be estimated, use argument option "ancestornodes". If, on the other hand, a group of branches (not in a clade) are hypothesized to follow the same rates, use argument option "listofnodes".
bg	A vector of nodes should be provided if the "ancestornodes" option was chosen for argument "bgtype". If, on the other hand, "listofnodes" was chosen, a list should be provided with each element of the list being a vector of nodes that limit the branches that follow the same rates. See examples and vignette.
partition	A list of vectors (of sites) subject to different evolutionary constraints. For example, supplying <code>list(c(1:2500), c(2501:5000))</code> means that sites 1 through 2500 follow their own substitution rates distinct from sites 2501 through 5000. These

sites correspond to the rows of the data supplied in userphyl. Partition models can be fitted with a common (or unique) gamma distribution for rate variation over all partitions, and/or common root probabilities can be estimated over all partitions.

ratevar	Default option is FALSE. Specifying "discgamma" implements the discrete gamma approximation model of Yang, 1994. Even if a partition of sites is specified, this option uses the same $\alpha$ parameter for the gamma distribution over all partitions. Specifying "partitionspecificgamma" implements the discrete gamma approximation model in each partition separately. The number of categories can be specified using the "nocat" argument. See examples and vignette.
nocat	The number of categories for the discrete gamma approximation.
reversible	This option forces a model to be reversible, i.e., the flow from state 'a' to 'b' is the same as the flow from state 'b' to 'a'. Only symmetric transition matrices can be specified in modelmat with this option. Inspired by the DiscML package.
numhessian	Set to FALSE if standard errors are not required. This speeds up the algorithm. Default is TRUE. Although the function being used to calculate these errors is reliable, it is rare but possible that the errors are not calculated (due to approximation-associated issues while calculating the Hessian). In this case, try bootstrapping with numhessian=FALSE.
rootprob	Four options are available: "equal", "stationary", "maxlik", and "user". <ul style="list-style-type: none"> <li>• Option "equal" means that all the discrete characters are given equal weight at the root.</li> <li>• Using "stationary" means that the discrete characters are weighted at the root by the stationary frequencies implied by the substitution rate matrix. Note that these can differ based on the partition. In the case that the "bg-type" argument is also provided, an average of the stationary frequencies of all branch groupings is used.</li> <li>• If "maxlik" is supplied, then the root frequencies are also estimated. These do not differ based on the partition.</li> <li>• If "user" is supplied here, a vector of root frequency parameters can be provided to argument "rpvec".</li> </ul>
rpvec	If option "user" is specified for argument "rootprob", supply a vector of the same length as that provided to argument "alphabet", representing the root frequency parameters.
init	Initial value for the rates. The default value is 0.9.
lowli	For finer control of the boundaries of the optimization problem. The default value is 0.001. This usually suffices if the branch lengths are in expected substitutions per site. However, if branch lengths are in different units, this should be changed accordingly.
upli	For finer control of the boundaries of the optimization problem. The default value is 100. This usually suffices if the branch lengths are in expected substitutions per site. However, if branch lengths are in different units, this should be changed accordingly.
...	Passing other arguments to the optimization algorithm "nlminb". For example, control = list(trace = 5) will print progress at every 5th iteration.

**Details**

See vignette for detailed examples.

**Value**

All arguments used while calling the `estimaterrates` function are attached in a list. The following components are also returned in the same list:

<code>call</code>	Function call used.
<code>conv</code>	A vector of convergence indicators for the model run. 0 denotes successful convergence.
<code>time</code>	Time taken in seconds.
<code>tree</code>	The phylogenetic tree used.
<code>bg</code>	List of group of nodes that capture branches that follow unique substitution rates.
<code>results</code>	List of results including the parameter estimates for each unique entry of <code>modemat</code> (excluding zeros), standard errors, number of parameters fit, and AIC and BIC values. Furthermore, details from the optimization routine applied are also available. Use <code>...\$results\$wop\$parsep</code> .
<code>data_red</code>	Unique phyletic patterns observed.
<code>w</code>	List of number of times each gene phyletic pattern was observed.

**Note**

See the vignette for an in-depth look at some examples.

**Author(s)**

Utkarsh J. Dang and G. Brian Golding  
<udang@binghamton.edu>

**References**

Paradis, E., Claude, J. & Strimmer, K. 2004. APE: analyses of phylogenetics and evolution in R language. *Bioinformatics* 20: 289-290. R package version 3.2.

Tane Kim and Weilong Hao 2014. DiscML: An R package for estimating evolutionary rates of discrete characters using maximum likelihood. R package version 1.0.1.

**Examples**

```
library(markophylo)
#####
data(simdata1) #example data for a 2-state continuous time markov chain model.
#Now, plot example tree.
ape::plot.phylo(simdata1$tree, edge.width = 2, show.tip.label = FALSE, no.margin = TRUE)
ape::nodeLabels(frame = "circle", cex = 0.7)
ape::tiplabels(frame = "circle", cex = 0.7)
print(simdata1$Q) #substitution matrix used to simulate example data
```

```

print(table(simdata1$data)) #states and frequencies
model1 <- estimates(userTree = simdata1$tree, userPhyl = simdata1$data,
                    alphabet = c(1, 2), rootprob = "equal",
                    modelmat = matrix(c(NA, 1, 2, NA), 2, 2))

print(model1)
####

#If the data is known to contain sampling bias such that certain phyletic
#patterns are not observed, then these unobserved data can be corrected for
#easily. First, let's create a filtered version of the data following which
#a correction will be applied within the function. Here, any patterns with all
#ones or twos are filtered out.
filterall1 <- which(apply(simdata1$data, MARGIN = 1, FUN =
                          function(x) isTRUE(all.equal(as.vector(x), c(1, 1, 1, 1)))))
filterall2 <- which(apply(simdata1$data, MARGIN = 1, FUN =
                          function(x) isTRUE(all.equal(as.vector(x), c(2, 2, 2, 2)))))
filteredSimdata1 <- simdata1$data[-c(filterall1, filterall2), ]
model1_f_corrected <- estimates(userTree = simdata1$tree, userPhyl = filteredSimdata1,
                               unobserved = matrix(c(1, 1, 1, 1, 2, 2, 2, 2), nrow = 2, byrow = TRUE),
                               alphabet = c(1, 2), rootprob = "equal",
                               modelmat = matrix(c(NA, 1, 2, NA), 2, 2))

print(model1_f_corrected)
#####
data(simdata2)
print(simdata2$Q)
#While simulating the data found in simdata2, the clade with node 7 as its
#most recent common ancestor (MRCA) was constrained to have twice the
#substitution rates as the rest of the branches in the tree.
print(table(simdata2$data))
model2 <- estimates(userTree = simdata2$tree, userPhyl = simdata2$data,
                    alphabet = c(1, 2), bgtype = "ancestornodes", bg = c(7),
                    rootprob = "equal", modelmat = matrix(c(NA, 1, 2, NA), 2, 2))

print(model2)
plotTree(model2, colors=c("blue", "darkgreen"), edge.width = 2, show.tip.label = FALSE,
          no.margin = TRUE)
ape::nodeLabels(frame = "circle", cex = 0.7)
ape::tiplabels(frame = "circle", cex = 0.7)
#####
#Nucleotide data was simulated such that the first half of sites followed
#substitution rates different from the other half of sites. Data was simulated
#in the two partitions with rates 0.33 and 0.99.
data(simdata3)
print(dim(simdata3$data))
print(table(simdata3$data))
model3 <- estimates(userTree = simdata3$tree, userPhyl = simdata3$data,
                    alphabet = c("a", "c", "g", "t"), rootprob = "equal",
                    partition = list(c(1:2500), c(2501:5000)),
                    modelmat = matrix(c(NA, 1, 1, 1, 1, NA, 1, 1,
                                          1, 1, NA, 1, 1, 1, 1, NA), 4, 4))

print(model3)

#More examples in the vignette.

```

---

estimaterrates\_f      *Estimate substitution rate matrix.*

---

### Description

Beta version of code; still in development. More or less identical to [estimaterrates](#). Here, some entries of the substitution rate matrix can be specified by the user to be held fixed during the parameter estimation.

### Usage

```
estimaterrates_f(userTree = NULL, userPhyl = NULL, matchTipstoData = FALSE,
  unobserved = NULL, alphabet = NULL, modelmat = NULL,
  bgtype = "listofnodes", bg = NULL, partition = NULL,
  ratevar = FALSE, nocat = 4, reversible = FALSE, numhessian = TRUE,
  rootprob = NULL, rPvec = NULL, init = 0.9, lowli = 0.001, upli = 100, ...)
```

### Arguments

userTree	Rooted binary tree of class "phylo". Read in Newick tree using read.tree() from package ape before passing this argument. The branch lengths must be in expected substitutions per site (but see lowli and upli arguments). Trees estimated using MrBayes and BEAST, for instance, yield branch lengths using that scale. If an unrooted tree is available, look into the "root" and "midpoint.root" functions from the APE and phytools packages, respectively.
userPhyl	A matrix or data frame of phyletic patterns. Rows represent the discrete character patterns and columns represent the taxa. These data can be numeric or character (but not both).
matchTipstoData	The default is FALSE, which means that the user must ensure that the ordering of the taxa in the data matrix must match the internal ordering of tip labels of the tree. Set to TRUE, if the column names of the data matrix, i.e., the taxa names, are all present, with the same spelling and notation in the tip labels of the tree provided, in which case, the restriction on ordering is not necessary.
unobserved	A matrix of unobserved phyletic patterns, representing possible sampling or acquisition bias. Each row should be a unique phyletic pattern.
alphabet	The set of discrete characters used. May be integer only or character only.
modelmat	Accepts the same arguments as <a href="#">estimaterrates</a> . Here, some entries of the substitution rate matrix can be specified by the user to be held fixed during the parameter estimation. A square matrix can be input that consists of integer values denoting the indices of the substitution rates to be estimated and entries with decimal values (that are not equivalent to an integer in R, i.e., something other than, for example, 1.0, 2.0, etc.) to be held fixed. See example.
bgtype	Use this to group branches hypothesized to follow the same rates (but differ from other branches). If clade-specific insertion and deletion rates are required

to be estimated, use argument option "ancestornodes". If, on the other hand, a group of branches (not in a clade) are hypothesized to follow the same rates, use argument option "listofnodes".

bg	A vector of nodes should be provided if the "ancestornodes" option was chosen for argument "bgtype". If, on the other hand, "listofnodes" was chosen, a list should be provided with each element of the list being a vector of nodes that limit the branches that follow the same rates. See examples and vignette.
partition	A list of vectors (of sites) subject to different evolutionary constraints. For example, supplying list(c(1:2500), c(2501:5000)) means that sites 1 through 2500 follow their own substitution rates distinct from sites 2501 through 5000. These sites correspond to the rows of the data supplied in userphyl. Partition models can be fitted with a common (or unique) gamma distribution for rate variation over all partitions, and/or common root probabilities can be estimated over all partitions.
ratevar	Default option is FALSE. Specifying "discgamma" implements the discrete gamma approximation model of Yang, 1994. Even if a partition of sites is specified, this option uses the same $\alpha$ parameter for the gamma distribution over all partitions. Specifying "partitionspecificgamma" implements the discrete gamma approximation model in each partition separately. The number of categories can be specified using the "nocat" argument. See examples and vignette.
nocat	The number of categories for the discrete gamma approximation.
reversible	This option forces a model to be reversible, i.e., the flow from state 'a' to 'b' is the same as the flow from state 'b' to 'a'. Only symmetric transition matrices can be specified in modelmat with this option. Inspired by the DiscML package.
numhessian	Set to FALSE if standard errors are not required. This speeds up the algorithm. Default is TRUE. Although the function being used to calculate these errors is reliable, it is rare but possible that the errors are not calculated (due to approximation-associated issues while calculating the Hessian). In this case, try bootstrapping with numhessian=FALSE.
rootprob	Four options are available: "equal", "stationary", "maxlik", and "user". <ul style="list-style-type: none"> <li>• Option "equal" means that all the discrete characters are given equal weight at the root.</li> <li>• Using "stationary" means that the discrete characters are weighted at the root by the stationary frequencies implied by the substitution rate matrix. Note that these can differ based on the partition. In the case that the "bg-type" argument is also provided, an average of the stationary frequencies of all branch groupings is used.</li> <li>• If "maxlik" is supplied, then the root frequencies are also estimated. These do not differ based on the partition.</li> <li>• If "user" is supplied here, a vector of root frequency parameters can be provided to argument "rpvec".</li> </ul>
rpvec	If option "user" is specified for argument "rootprob", supply a vector of the same length as that provided to argument "alphabet", representing the root frequency parameters.
init	Initial value for the rates. The default value is 0.9.

lowli	For finer control of the boundaries of the optimization problem. The default value is 0.001. This usually suffices if the branch lengths are in expected substitutions per site. However, if branch lengths are in different units, this should be changed accordingly.
upli	For finer control of the boundaries of the optimization problem. The default value is 100. This usually suffices if the branch lengths are in expected substitutions per site. However, if branch lengths are in different units, this should be changed accordingly.
...	Passing other arguments to the optimization algorithm "nlminb". For example, <code>control = list(trace = 5)</code> will print progress at every 5th iteration.

### Details

See vignette for detailed examples.

### Value

All arguments used while calling the `estimaterrates` function are attached in a list. The following components are also returned in the same list:

call	Function call used.
conv	A vector of convergence indicators for the model run. 0 denotes successful convergence.
time	Time taken in seconds.
tree	The phylogenetic tree used.
bg	List of group of nodes that capture branches that follow unique substitution rates.
results	List of results including the parameter estimates for each unique entry of <code>modemat</code> (excluding zeros), standard errors, number of parameters fit, and AIC and BIC values. Furthermore, details from the optimization routine applied are also available. Use <code>...\$results\$wop\$parsep</code> .
data_red	Unique phyletic patterns observed.
w	List of number of times each gene phyletic pattern was observed.

### Note

Experimental function

### Author(s)

Utkarsh J. Dang and G. Brian Golding  
<udang@binghamton.edu>

### References

- Paradis, E., Claude, J. & Strimmer, K. 2004. APE: analyses of phylogenetics and evolution in R language. *Bioinformatics* 20: 289-290. R package version 3.2.
- Tane Kim and Weilong Hao 2014. DiscML: An R package for estimating evolutionary rates of discrete characters using maximum likelihood. R package version 1.0.1.

**Examples**

```
#####
## Not run:
rm(list=ls())
set.seed(1) #set seed
library(markophylo)
tree1 <- ape::rtree(4, br = runif(6, 0.01, 0.3)) #generate a random 4-taxa tree (6 branches)
bf <- c(0.33, 0.33, 0.34) #vector of equal root frequencies
gf <- 5000 #number of character patterns to be simulated
rootseq <- sample(1:3, gf, replace = TRUE, prob = bf) #simulate the MRCA sequence (1s and 2s).
par1 <- 1.5
par2 <- 3
par3 <- 5
Q_sim <- matrix(c(NA, 0.6, par2, par1, NA, 0.8, par2, par3, NA), 3, 3)
#generated substitution rate matrix
diag(Q_sim) <- -rowSums(Q_sim, na.rm = TRUE)
print(Q_sim)
#simulate data using the geiger package and Q_sim
data_sim <- sapply(1:gf, FUN = function(X) geiger::sim.char(tree1,
Q_sim, model = "discrete", root = rootseq[X]))
simulateddat <- list(data = t(data_sim), Q = Q_sim, tree = tree1)

model1 <- markophylo::estimates_f(usertree = simulateddat$tree,
userphyl = simulateddat$data, alphabet = c(1, 2, 3), rootprob = "equal",
modelmat = matrix(c(NA, 0.6, 2, 1, NA, 0.8, 2, 3, NA), 3, 3))

## End(Not run)
```

---

patterns

*Unique phyletic patterns with counts.*

---

**Description**

Generate a reduced dataset consisting of unique phyletic patterns

**Usage**

```
patterns(x)
```

**Arguments**

x                    A matrix or data frame of phyletic patterns. Rows represent the discrete character patterns and columns represent the taxa. These data can be numeric or character.

**Value**

w                    Counts of unique phyletic patterns.  
databp\_red           Reduced data set corresponding to w.  
names                Column names of x.

**Author(s)**

Utkarsh J. Dang and G. Brian Golding  
<udang@mcmaster.ca>

**See Also**

See also [estimatesrates](#).

**Examples**

```
data(simdata1)
uniq_patt <- patterns(simdata1$data)
print(uniq_patt)
```

---

plottree	<i>Plot the tree used with different hypothesized branch groupings (or clades) following unique rates coloured differently.</i>
----------	---

---

**Description**

Plotting command for use on an object of class "markophylo".

**Usage**

```
plottree(x, colors = NULL, ...)
```

**Arguments**

x	An object of class "markophylo".
colors	Default works well. However, a custom vector of colours can be specified here—should be the same length as length(x\$bg). Note that these colours are used to colour the different branch groupings following their own rates.
...	Any further commands to ape::plot.phylo.

**Author(s)**

Utkarsh J. Dang and G. Brian Golding  
<udang@mcmaster.ca>

**See Also**

See also [estimatesrates](#).

**Examples**

```

data(simdata2)
model2 <- estimatorates(usertree = simdata2$tree, userphyl = simdata2$data,
                        alphabet = c(1, 2), bgtype = "ancestornodes", bg = c(7),
                        rootprob = "equal", modelmat = matrix(c(NA, 1, 2, NA), 2, 2))
plottree(model2, colors=c("blue", "darkgreen"), edge.width = 2, show.tip.label = FALSE,
          no.margin = TRUE)
ape::nodeLabels(frame = "circle", cex = 0.7)
ape::tiplabels(frame = "circle", cex = 0.7)

```

---

```
print.markophylo      Print summary information for the model fit.
```

---

**Description**

Summary command for use on an object of class "markophylo". The estimated substitution rates for each partition specified along with the estimated  $\alpha$ (s) for the gamma distribution (if any) and prior probability of discrete characters at the root (if estimated) are printed. If branch groupings (or clades) were specified, then the rates (and corresponding standard errors) are displayed in a matrix with the columns representing the different branch groupings (ordered by the subsets of  $x$ \$bg where  $x$  is an object of class "idea"). The rows represent the indices of the model matrix.

**Usage**

```
## S3 method for class 'markophylo'
print(x, ...)
```

**Arguments**

<code>x</code>	An object of class "markophylo".
<code>...</code>	Ignore this

**Author(s)**

Utkarsh J. Dang and G. Brian Golding  
<udang@mcmaster.ca>

**See Also**

See also [estimatorates](#).

---

`simdata1`*Simulated data.*

---

**Description**

Five simulated data sets are included.

**Usage**

```
data("simdata1")
data("simdata2")
data("simdata3")
data("simdata4")
data("simdata5")
```

**Format**

Each data set contains a list that comprises a tree (called "tree") and data (called "data") as its components. The tree is in the ape package phylo format. The data component consists of a matrix of discrete character patterns with the different patterns as the rows and the taxa as the columns.

**Details**

These data were simulated with either the geiger or the phangorn packages.

**References**

Schliep K.P. 2011. phangorn: phylogenetic analysis in R. *Bioinformatics*, 27(4) 592–593. R package version 1.99-12.

Harmon Luke J, Jason T Weir, Chad D Brock, Richard E Glor, and Wendell Challenger. 2008. GEIGER: investigating evolutionary radiations. *Bioinformatics* 24:129-131. R package version 2.0.3.

**Examples**

```
data(simdata1)
data(simdata2)
data(simdata3)
data(simdata4)
data(simdata5)
```

# Index

**\* datasets**

simdata1, [14](#)

**\* package**

markophylo-package, [2](#)

estimates, [2](#), [3](#), [8](#), [12](#), [13](#)

estimates\_f, [8](#)

markophylo (markophylo-package), [2](#)

markophylo-package, [2](#)

patterns, [11](#)

plottree, [12](#)

print.markophylo, [13](#)

simdata1, [14](#)

simdata2 (simdata1), [14](#)

simdata3 (simdata1), [14](#)

simdata4 (simdata1), [14](#)

simdata5 (simdata1), [14](#)