

Package ‘treeman’

August 29, 2016

Type Package

Title Phylogenetic Tree Manipulation Class and Methods

Version 1.0

Date 2016-04-16

Author D.J. Bennett

Maintainer D.J. Bennett <dominic.john.bennett@gmail.com>

Description S4 class and methods for efficient phylogenetic tree manipulation
for simulating evolution, running phylogenetic statistics and plotting.

License GPL-2

Depends R (>= 3.2.4), methods

Imports plyr

Suggests testthat

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2016-04-17 19:50:17

R topics documented:

treeman-package	3
addTip	3
as-TreeMen	4
calcDstBLD	5
calcDstMtrx	6
calcDstRF	6
calcDstTrp	7
calcFrPrp	8
calcNdBlnc	9
calcNdsBlnc	10
calcOvrlp	10
calcPhyDv	11

cTrees	12
getNdAge	13
getNdKids	13
getNdLng	14
getNdPrid	15
getNdPtid	15
getNdsAge	16
getNdsKids	17
getNdsLng	17
getNdSlt	18
getNdsPrid	19
getNdsPtid	19
getNdsSlt	20
getNdsSstr	21
getNdSstr	22
getOtgrp	22
getPath	23
getPrnt	24
getSpnAge	24
getSpnsAge	25
getSubtree	26
getTxnoms	26
mammals	27
Node-class	28
pinTips	29
randTree	30
readTree	30
rmTip	31
setAge	32
setNdID	32
setNdOther	33
setNdsID	34
setNdsOther	35
setNdSpn	35
setNdsSpn	36
setPD	37
setTol	38
TreeMan-class	38
TreeMen-class	41
viz	42
writeTree	42

treeman-package *Phylogenetic Tree Manipulation*

Description

Manipulate phylogenetic trees in R simply, intuitively and efficiently with a list-based tree structure.

Details

Package: treeman
Type: Package
Version: 1.0
Date: 2015-12-02
License: GPL-2

Author(s)

D.J. Bennett

Maintainer: D.J. Bennett <dominic.john.bennett@gmail.com>

References

No key references

See Also

<https://github.com/DomBennett/treeman/wiki>

Examples

```
library(treeman)
tree <- randTree(100)
print(tree)
```

addTip *Add tip to a tree*

Description

Returns a tree with a tip ID added

Usage

```
addTip(tree, tid, sid, start, end, pid = paste0("p_", tid))
```

Arguments

tree	TreeMan object
tid	tip ID
sid	ID of node that will become new tip's sister
start	start time
end	end time
pid	parent ID (default is 'p_' + tid)

Details

User must provide a new tip ID, the ID of a node which will become the new tip's sister, a start time point to specify when the new branch will start in time and, an end time point (0 for extant tips).

See Also

[rmTip](https://github.com/DomBennett/treeman/wiki/manip-methods), <https://github.com/DomBennett/treeman/wiki/manip-methods>

Examples

```
library(treeman)
tree <- randTree(10)
# add a new tip to the branch preceding t1
# calculate the span and find a point in that time frame for start
t1_spn <- getSpnAge(tree, 't1')
start <- runif(max=t1_spn[1, 'start'], min=t1_spn[1, 'end'], n=1)
end <- runif(max=start, min=0, n=1)
tree <- addTip(tree, tid='t11', sid='t1', start=start, end=end)
```

as-TreeMen

Convert list to a TreeMen

Description

Return a TreeMen object from a list of TreeMans

See Also

[TreeMen-class](#)

Examples

```
library(treeman)
trees <- list('tree_1'=randTree(10), 'tree_2'=randTree(10))
trees <- as(trees, 'TreeMen')
```

calcDstBLD	<i>Calculate the BLD between two trees</i>
------------	--

Description

Returns the branch length distance between two trees.

Usage

```
calcDstBLD(tree_1, tree_2, nrmlsd = FALSE, ...)
```

Arguments

tree_1	TreeMan object
tree_2	TreeMan object
nrmlsd	Boolean, should returned value be between 0 and 1? Default, FALSE.
...	plyr arguments

Details

BLD is the Robinson-Foulds distance weighted by branch length. Instead of summing the differences in partitions between the two trees, the metric takes the square root of the squared difference in branch lengths. Parallelizable.

References

Kuhner, M. K. and Felsenstein, J. (1994) Simulation comparison of phylogeny algorithms under equal and unequal evolutionary rates. *Molecular Biology and Evolution*, 11, 459-468.

See Also

`calcDstTrp`, `calcDstRF` <https://github.com/DomBennett/treeman/wiki/calc-methods>

Examples

```
library(treeman)
tree_1 <- randTree(10)
tree_2 <- randTree(10)
calcDstBLD(tree_1, tree_2)
```

calcDstMtrx	<i>Calculate the distance matrix</i>
-------------	--------------------------------------

Description

Returns a distance matrix for specified ids of a tree.

Usage

```
calcDstMtrx(tree, ids)
```

Arguments

tree	TreeMan object
ids	IDs of nodes/tips

Details

The distance between every id in the tree is calculated by summing the lengths of the branches that connect them. This can be useful for testing the distances between trees, checking for evolutionary isolated tips etc.

See Also

[calcDstBLD](#), [calcDstRF](#), [calcDstTrp](#) <https://github.com/DomBennett/treeman/wiki/calc-methods>

Examples

```
# checking the distance between two trees
library(treeman)
tree_1 <- randTree(10)
tree_2 <- randTree(10)
dmat1 <- calcDstMtrx(tree_1, tree_1['tips'])
dmat2 <- calcDstMtrx(tree_2, tree_2['tips'])
mdl <- cor.test(x=dmat1, y=dmat2)
as.numeric(1 - mdl$estimate) # 1 - Pearson's r
```

calcDstRF	<i>Calculate the Robinson-Foulds distance between two trees</i>
-----------	---

Description

Returns the Robinson-Foulds distance between two trees.

Usage

```
calcDstRF(tree_1, tree_2, nrmlsd = FALSE, ...)
```

Arguments

tree_1	TreeMan object
tree_2	TreeMan object
nrmlsd	Boolean, should returned value be between 0 and 1? Default, FALSE.
...	plyr arguments

Details

RF distance is calculated as the sum of partitions in one tree that are not shared by the other. The maximum number of split differences is the total number of nodes in both trees (excluding the roots). Parallelizable.

References

Robinson, D. R.; Foulds, L. R. (1981). "Comparison of phylogenetic trees". *Mathematical Bio-sciences* 53: 131-147.

See Also

[calcDstBLD](https://github.com/DomBennett/treeman/wiki/calc-methods), [calcDstTrp](https://github.com/DomBennett/treeman/wiki/calc-methods) <https://github.com/DomBennett/treeman/wiki/calc-methods>

Examples

```
library(treeman)
tree_1 <- randTree(10)
tree_2 <- randTree(10)
calcDstRF(tree_1, tree_2)
```

calcDstTrp

Calculate the triplet distance between two trees

Description

Returns the triplet distance between two trees.

Usage

```
calcDstTrp(tree_1, tree_2, nrmlsd = FALSE, ...)
```

Arguments

tree_1	TreeMan object
tree_2	TreeMan object
nrmlsd	Boolean, should returned value be between 0 and 1? Default, FALSE.
...	plyr arguments

Details

The triplet distance is calculated as the sum of different outgroups among every triplet of tips between the two trees. Normalisation is performed by dividing the resulting number by the total number of triplets shared between the two trees. The triplet distance is calculated only for shared tips between the two trees. Parallelizable.

References

Critchlow DE, Pearl DK, Qian C. (1996) The Triples Distance for rooted bifurcating phylogenetic trees. *Systematic Biology*, 45, 323-34.

See Also

calcDstBLD, calcDstRF <https://github.com/DomBennett/treeman/wiki/calc-methods>

Examples

```
library(treeman)
tree_1 <- randTree(10)
tree_2 <- randTree(10)
calcDstTrp(tree_1, tree_2)
```

calcFrPrp

Calculate evolutionary distinctness

Description

Returns the evolutionary distinctness of ids using the fair proportion metric.

Usage

```
calcFrPrp(tree, tids, ...)
```

Arguments

tree	TreeMan object
tids	tip IDs
...	plyr arguments

Details

The fair proportion metric calculates the evolutionary distinctness of tips in a tree through summing the total amount of branch length each tip represents, where each branch in the tree is evenly divided between all descendants. Parallelizable.

References

Isaac, N.J.B., Turvey, S.T., Collen, B., Waterman, C. and Baillie, J.E.M. (2007). Mammals on the EDGE: conservation priorities based on threat and phylogeny. PLoS ONE, 2, e296.

See Also

[calcPhyDv](https://github.com/DomBennett/treeman/wiki/calc-methods) <https://github.com/DomBennett/treeman/wiki/calc-methods>

Examples

```
library(treeman)
tree <- randTree(10)
calcFrPrp(tree, tree['tips'])
```

calcNdBlnc	<i>Calculate the balance of a node</i>
------------	--

Description

Returns the balance of a node.

Usage

```
calcNdBlnc(tree, nid)
```

Arguments

tree	TreeMan object
nid	node id

Details

Balance is calculated as the absolute difference between the number of descendents of the two bifurcating edges of a node and the expected value for a balanced tree. NA is returned if the node is polytomous or a tip.

See Also

[calcNdsBlnc](https://github.com/DomBennett/treeman/wiki/calc-methods), <https://github.com/DomBennett/treeman/wiki/calc-methods>

Examples

```
library(treeman)
tree <- randTree(10)
calcNdBlnc(tree, nid=tree['root']) # root balance
```

`calcNdsBlnc`*Calculate the balances of all nodes*

Description

Returns the absolute differences in number of descendants for bifurcating branches of every node

Usage

```
calcNdsBlnc(tree, nids, ...)
```

Arguments

<code>tree</code>	TreeMan object
<code>nids</code>	node ids
<code>...</code>	plyr arguments

Details

Runs `calcNdBlnc()` across all node IDs. NA is returned if the node is polytomous. Parallelizable.

See Also

`calcNdBlnc`, <https://github.com/DomBennett/treeman/wiki/calc-methods>

Examples

```
library(treeman)
tree <- randTree(10)
calcNdsBlnc(tree, nids=tree['nds'])
```

`calcOvrlp`*Calculate phylogenetic overlap*

Description

Returns the sum of branch lengths represented by `ids_1` and `ids_2` for a tree.

Usage

```
calcOvrlp(tree, ids_1, ids_2, nrmlsd = FALSE, ...)
```

Arguments

tree	TreeMan object
ids_1	tip ids of community 1
ids_2	tip ids of community 2
nrmlsd	Boolean, should returned value be between 0 and 1? Default, FALSE.
...	plyr arguments

Details

Use this to calculate the sum of branch lengths that are represented between two communities. This measure is also known as the unique fraction. It can be used to measure concepts of phylogenetic turnover. Parallelizable.

References

Lozupone, C., & Knight, R. (2005). UniFrac: a new phylogenetic method for comparing microbial communities. *Applied and Environmental Microbiology*, 71(12), 8228-35.

See Also

calcPhyDv <https://github.com/DomBennett/treeman/wiki/calc-methods>

Examples

```
library(treeman)
tree <- randTree(10)
ids_1 <- sample(tree['tips'], 5)
ids_2 <- sample(tree['tips'], 5)
calcOvrlp(tree, ids_1, ids_2)
```

calcPhyDv

Calculate phylogenetic diversity

Description

Returns the phylogenetic diversity of a tree for the tips specified.

Usage

```
calcPhyDv(tree, tids, ...)
```

Arguments

tree	TreeMan object
tids	tip ids
...	plyr arguments

Details

Faith's phylogenetic diversity is calculated as the sum of all connected branches for specified tips in a tree. It can be used to investigate how biodiversity as measured by the phylogeny changes. Parallelizable.

References

Faith, D. (1992). Conservation evaluation and phylogenetic diversity. *Biological Conservation*, 61, 1-10.

See Also

`calcFrPrp`, `calcOvrIp` <https://github.com/DomBennett/treeman/wiki/calc-methods>

Examples

```
library(treeman)
tree <- randTree(10)
calcPhyDv(tree, tree['tips'])
```

cTrees

cTrees

Description

Return TreeMen of concatenated trees.

Usage

```
cTrees(x, ...)
```

Arguments

<code>x</code>	TreeMan or TreeMen objects
<code>...</code>	more TreeMan or TreeMen objects

Details

Concatenate trees into single TreeMen object.

See Also

[TreeMen-class](#), [TreeMan-class](#), [as-TreeMen](#)

Examples

```
library(treeman)
trees <- cTrees(randTree(10), randTree(10))
```

`getNdAge`*Get age*

Description

Return the root to tip distance for id.

Usage

```
getNdAge(tree, id)
```

Arguments

tree	TreeMan object
id	node id

Details

Returns a numeric.

See Also

[getNdsAge](#), [getSpnAge](#), [getSpnsAge](#), [getPrnt](#), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
data(mammals)
# when did apes emerge?
# get parent id for all apes
prnt_id <- getPrnt(mammals, ids=c('Homo_sapiens', 'Hylobates_concolor'))
getNdAge(mammals, id=prnt_id)
```

`getNdKids`*Get children IDs*

Description

Return the node ids of all tips that descend from node.

Usage

```
getNdKids(tree, id)
```

Arguments

tree	TreeMan object
id	node id

Details

Returns a vector

See Also

[getNdsKids](https://github.com/DomBennett/treeman/wiki/get-methods), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
# everyone descends from root
getNdKids(tree, id=tree['root'])
```

getNdLng

Get lineage

Description

Return unique taxonyms for connecting id to root.

Usage

```
getNdLng(tree, id)
```

Arguments

tree	TreeMan object
id	node id

Details

Returns a vector.

See Also

[getNdsLng](https://github.com/DomBennett/treeman/wiki/get-methods), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
data(mammals)
# return human lineage
getNdLng(mammals, id='Homo_sapiens')
```

getNdPrid *Get pre-nodes to root*

Description

Return node ids for connecting id to root.

Usage

```
getNdPrid(tree, id)
```

Arguments

tree	TreeMan object
id	node id

Details

Returns a vector.

See Also

[getNdsPrid](#), [getNdPtId](#), [getNdsPtId](#), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
# get all nodes to root
getNdPrid(tree, id='t1')
```

getNdPtId *Get post-nodes to tips*

Description

Return node ids for connecting id to kids.

Usage

```
getNdPtId(tree, id)
```

Arguments

tree	TreeMan object
id	node id

Details

Returns a vector.

See Also

[getNdsPtid](#), [getNdPrid](#), [getNdsPrid](#), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
# get all nodes from root to tip
getNdPtid(tree, id='n1')
```

getNdsAge

Get ages for multiple nodes

Description

Return the root to tip distances for ids.

Usage

```
getNdsAge(tree, ids, ...)
```

Arguments

tree	TreeMan object
ids	vector of node ids
...	plyr arguments

Details

Returns a vector, parallelizable.

See Also

[getNdAge](#), [getSpnAge](#), [getSpnsAge](#), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
getNdsAge(tree, ids=tree['nds'])
```

getNdsKids *Get children IDs for multiple nodes*

Description

Return the node ids of all tips that descend from each node in ids.

Usage

```
getNdsKids(tree, ids, ...)
```

Arguments

tree	TreeMan object
ids	vector of node ids
...	plyr arguments

Details

Returns a list, parallelizable.

See Also

[getNdKids](https://github.com/DomBennett/treeman/wiki/get-methods), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
getNdsKids(tree, id=tree['nds'])
```

getNdsLng *Get lineage for multiple nodes*

Description

Return unique taxonyms for connecting ids to root.

Usage

```
getNdsLng(tree, ids, ...)
```

Arguments

tree	TreeMan object
ids	vector of node ids
...	plyr arguments

Details

Returns a list, parallelizable.

See Also

[getNdLng](https://github.com/DomBennett/treeman/wiki/get-methods), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
data(mammals)
# return human and gorilla lineages
getNdLng(mammals, id=c('Homo_sapiens', 'Gorilla_gorilla'))
```

getNdSlt

Get a node slot

Description

Returns the value of named slot.

Usage

```
getNdSlt(tree, slt_nm, id)
```

Arguments

tree	TreeMan object
slt_nm	slot name
id	node id

Details

Returned object depends on name, either character, vector or numeric.

See Also

[getNdsSlt](https://github.com/DomBennett/treeman/wiki/get-methods), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
getNdSlt(tree, slt_nm='spn', id='t1') # return span of t1
```

getNdsPrid	<i>Get pre-nodes to root for multiple nodes</i>
------------	---

Description

Return node ids for connecting ids to root.

Usage

```
getNdsPrid(tree, ids, ...)
```

Arguments

tree	TreeMan object
ids	vector of node ids
...	plyr arguments

Details

Returns a list, parallizable.

See Also

[getNdPrid](#), [getNdPtid](#), [getNdsPtid](#), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
# get all nodes to root
getNdsPrid(tree, ids=tree['tips'])
```

getNdsPtid	<i>Get post-nodes to tips for multiple nodes</i>
------------	--

Description

Return node ids for connecting ids to kids.

Usage

```
getNdsPtid(tree, ids, ...)
```

Arguments

tree	TreeMan object
ids	vector of node ids
...	plyr arguments

Details

Returns a list, parallizable.

See Also

[getNdPtid](#), [getNdPrid](#), [getNdsPrid](#), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
# get all nodes to tip for all nodes
getNdsPtid(tree, ids=tree['nds'])
```

getNdsSlT

Get a node slot for multiple nodes

Description

Returns the value of named slot.

Usage

```
getNdsSlT(tree, slt_nm, ids, ...)
```

Arguments

tree	TreeMan object
slt_nm	slot name
ids	vector of node ids
...	plyr arguments

Details

Returned object depends on name, either character, vector or numeric. Parallelizable.

See Also

[getNdSlT](#), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
getNdsSstr(tree, slt_nm='spn', ids=tree['tips']) # return spans of all tips
```

getNdsSstr

Get sister id

Description

Returns the ids of the sister(s) of nd ids given.

Usage

```
getNdsSstr(tree, ids, ...)
```

Arguments

tree	TreeMan object
ids	nd ids
...	plyr arguments

Details

An error is raised if there is no sister (e.g. for the root). There can be more than one sister if tree is polytomous. Parallelizable.

See Also

[getNdSstr](https://github.com/DomBennett/treeman/wiki/get-methods), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
getNdsSstr(tree, ids=tree['tips'])
```

getNdSstr	<i>Get sister id</i>
-----------	----------------------

Description

Returns the id of the sister(s) of node id given.

Usage

```
getNdSstr(tree, id)
```

Arguments

tree	TreeMan object
id	node id

Details

An error is raised if there is no sister (e.g. for the root). There can be more than one sister if tree is polytomous.

See Also

[getNdsSstr](https://github.com/DomBennett/treeman/wiki/get-methods), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
getNdSstr(tree, id='t1')
```

getOtgrp	<i>Get outgroup</i>
----------	---------------------

Description

Return the outgroup based on a tree and a vector of IDs.

Usage

```
getOtgrp(tree, ids)
```

Arguments

tree	TreeMan object
ids	vector of node ids

Details

Returns a id, character. If there are multiple possible outgroups, returns NULL.

See Also

<https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
data(mammals)
# orangutan is an outgroup wrt humans and chimps
getOtgrp(mammals, ids=c('Homo_sapiens', 'Pan_troglodytes', 'Pongo_pygmaeus'))
```

getPath	<i>Get path between nodes</i>
---------	-------------------------------

Description

Return node ids for connecting from to to.

Usage

```
getPath(tree, from, to)
```

Arguments

tree	TreeMan object
from	starting node id
to	ending node id

Details

Returns a vector, first id is from to to.

See Also

<https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
data(mammals)
# what's the phylogenetic distance from humans to gorillas?
ape_id <- getPrnt(mammals, ids=c('Homo_sapiens', 'Hylobates_concolor'))
pth <- getPath(mammals, from='Homo_sapiens', to='Gorilla_gorilla')
sum(getNdsSlt(mammals, ids=pth, slt_nm='spn'))
```

getPrnt	<i>Get parent</i>
---------	-------------------

Description

Return parental (most recent common ancestor) node id for ids.

Usage

```
getPrnt(tree, ids)
```

Arguments

tree	TreeMan object
ids	vector of node ids

Details

Returns a character.

See Also

[getSubtree](https://github.com/DomBennett/treeman/wiki/get-methods), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
data(mammals)
# choosing ids from the two main branches of apes allows to find the parent for all apes
ape_id <- getPrnt(mammals, ids=c('Homo_sapiens', 'Hylobates_concolor'))
```

getSpnAge	<i>Get age range</i>
-----------	----------------------

Description

Return start and end root to tip distances for id.

Usage

```
getSpnAge(tree, id)
```

Arguments

tree	TreeMan object
id	node id

Details

Returns a dataframe.

See Also

[getNdAge](#), [getNdsAge](#), [getSpnsAge](#), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
data(mammals)
getSpnAge(mammals, id='Homo_sapiens')
```

getSpnsAge	<i>Get age ranges for multiple nodes</i>
------------	--

Description

Return start and end root to tip distances for ids.

Usage

```
getSpnsAge(tree, ids, ...)
```

Arguments

tree	TreeMan object
ids	vector of node ids
...	plyr arguments

Details

Returns a dataframe, parallelizable.

See Also

[getNdAge](#), [getNdsAge](#), [getSpnAge](#), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
tree <- randTree(10)
# all nodes but root
ids <- tree['nds'][tree['nds'] != tree['root']]
getSpnsAge(tree, ids=ids)
```

`getSubtree`*Get subtree*

Description

Return tree descending from id.

Usage

```
getSubtree(tree, id)
```

Arguments

<code>tree</code>	TreeMan object
<code>id</code>	node id

Details

Returns a TreeMan, parallelizable.

See Also

`getPrnt`, <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
data(mammals)
# get tree of apes
ape_id <- getPrnt(mammals, ids=c('Homo_sapiens', 'Hylobates_concolor'))
apes <- getSubtree(mammals, id=ape_id)
```

`getTxnyms`*Get node id(s) for txonyms*

Description

Returns the node ids of nodes with given taxonyms.

Usage

```
getTxnyms(tree, txnyms, ...)
```

Arguments

tree	TreeMan object
txnyms	vector of taxonomic names
...	plyr arguments

Details

Returns a list, parallelizable.

See Also

[getNdLng](#), [getNdsLng](#), <https://github.com/DomBennett/treeman/wiki/get-methods>

Examples

```
library(treeman)
data(mammals)
homo_ids <- getTxnyms(mammals, txnyms='Homo')
```

mammals

Phylogenetic tree of Mammalia

Description

Bininda-Emonds et al. (2007)'s Mammalian supertree augmented with taxonomic information. Here used for testing and demonstrating treeman functions.

Usage

```
data(mammals)
```

Format

mammals is a TreeMan object

Source

Bininda-Emonds et al. (2007) The Delayed Rise of Present-Day Mammals. *Nature*, 446(7135): 507-512

Examples

```
data(mammals) # load object
print(mammals)
print(mammals[['Homo_sapiens']])
```

Node-class

*Node-class***Description**

The Node is an S4 class used for displaying node information. It is only generated when a user implements the `[[]]` on a tree.

Usage

```
## S4 method for signature 'Node'
as.character(x)

## S4 method for signature 'Node'
show(object)

## S4 method for signature 'Node'
print(x)

## S4 method for signature 'Node,character'
x[i]
```

Arguments

x	Node object
object	Node object
i	slot name

Slots

```
id unique ID for node in tree['ndlst']
spn length of preceding branch
prid parent node ID
ptid child node ID
kids descending tip IDs
nkids number of descending tip IDs
txnym list of associated taxonyms
pd total branch length represented by node
prdst total branch length of connected prids
age age of node in tree
root T/F root node?
tip T/F tip node?
```

See Also

[cTrees](#)

pinTips	<i>Pin tips to a tree</i>
---------	---------------------------

Description

Returns a tree with new tips added based on given lineages and time points

Usage

```
pinTips(tree, tids, lngs, ends, ...)
```

Arguments

tree	TreeMan object
tids	new tip ids
lngs	list of vectors of the lineages of each tid
ends	end time points for each tid
...	plyr arguments

Details

User must provide a vector of new tip IDs, a list of the ranked lineages for these IDs (in ascending order) and a vector of end time points for each new ID (0s for extant tips). The function expects the given tree to be taxonomically informed; the `txnym` slot for every node should have a taxonomic label. The function takes the lineage and tries to randomly add the new tip at the lowest point in the taxonomic rank before the end time point. Parallelizable.

See Also

<https://github.com/DomBennett/treeman/wiki/manip-methods>

Examples

```
# see https://github.com/DomBennett/treeman/wiki/Pinning-tips for a detailed example
```

randTree	<i>Generate a random tree</i>
----------	-------------------------------

Description

Returns a random TreeMan tree with n tips.

Usage

```
randTree(n)
```

Arguments

n	number of tips, integer, must be greater than 1
---	---

Details

Equivalent to ape's `rtree()` but returns a TreeMan tree. Tree is always rooted and bifurcating.

See Also

[TreeMan-class](#)

Examples

```
library(treeman)
tree <- randTree(5)
```

readTree	<i>Read a Newick tree</i>
----------	---------------------------

Description

Return a TreeMan or TreeMen object from a Newick treefile

Usage

```
readTree(file = NULL, text = NULL, ...)
```

Arguments

file	file path
text	Newick character string
...	plyr arguments

Details

Read a single or multiple trees from a file, or a text string. Parallelizable.

See Also

[writeTree](#), [randTree](#), https://en.wikipedia.org/wiki/Newick_format

Examples

```
library(treeman)
tree <- readTree(text="((A:1.0,B:1.0):1.0,(C:1.0,D:1.0):1.0);")
```

rmTip	<i>Remove tip from a tree</i>
-------	-------------------------------

Description

Returns a tree with a tip ID remove

Usage

```
rmTip(tree, tid, drp_intrnl = TRUE)
```

Arguments

tree	TreeMan object
tid	tip ID
drp_intrnl	Boolean, drop internal branches, default FALSE

Details

Removes a tip in a tree. Set `drp_intrnl` to FALSE to convert internal nodes into new tips.

See Also

[addTip](#), <https://github.com/DomBennett/treeman/wiki/manip-methods>

Examples

```
library(treeman)
tree <- randTree(10)
tree <- rmTip(tree, 't1')
```

setAge	<i>Set the age of a tree</i>
--------	------------------------------

Description

Return a tree with the age altered.

Usage

```
setAge(tree, val)
```

Arguments

tree	TreeMan object
val	new age

Details

Use this function to change the age of a tree. For example, you might want to convert the tree so that its age equals 1. This function will achieve that by modifying every branch, while maintaining their relative lengths.

See Also

[setPD](https://github.com/DomBennett/treeman/wiki/set-methods) <https://github.com/DomBennett/treeman/wiki/set-methods>

Examples

```
library(treeman)
tree <- randTree(10)
tree <- setAge(tree, val=1)
(tree['age'])
```

setNdID	<i>Set the ID of a node</i>
---------	-----------------------------

Description

Return a tree with the ID of a node altered.

Usage

```
setNdID(tree, id, val)
```


Arguments

tree	TreeMan object
id	id to be changed
val	new id

Details

IDs cannot be changed directly for the TreeMan class. To change an ID use this function. Warning: all IDs must be unique, avoid spaces in IDs.

See Also

[setNdsID](https://github.com/DomBennett/treeman/wiki/set-methods) <https://github.com/DomBennett/treeman/wiki/set-methods>

Examples

```
library(treeman)
tree <- randTree(10)
tree <- setNdID(tree, 't1', 'heffalump')
```

setNdOther	<i>Set a user defined slot</i>
------------	--------------------------------

Description

Return a tree with a user defined slot for node ID.

Usage

```
setNdOther(tree, id, val, slt_nm)
```

Arguments

tree	TreeMan object
id	id of the node
val	data for slot
slt_nm	slot name

Details

A user can specify new slots in a tree. Add a new slot with this function by providing a node ID, a value for the new slot and a unique new slot name. Slot names must not be default TreeMan names. The new value can be any data type.

See Also

[setNdsOther](https://github.com/DomBennett/treeman/wiki/set-methods) <https://github.com/DomBennett/treeman/wiki/set-methods>

Examples

```
library(treeman)
tree <- randTree(10)
tree <- setNdOther(tree, 't1', 1, 'binary_val')
(getNdSlt(tree, id='t1', slt_nm='binary_val'))
```

setNdsID

Set the IDs of multiple nodes

Description

Return a tree with the IDs of nodes altered.

Usage

```
setNdsID(tree, ids, vals, ...)
```

Arguments

tree	TreeMan object
ids	ids to be changed
vals	new ids
...	plyr arguments

Details

Runs `setNdID()` over multiple nodes. Warning: all IDs must be unique, avoid spaces in IDs. Parellizable.

See Also

`setNdID` <https://github.com/DomBennett/treeman/wiki/set-methods>

Examples

```
library(treeman)
tree <- randTree(10)
new_ids <- paste0('heffalump_', 1:tree['ntips'])
tree <- setNdsID(tree, tree['tips'], new_ids)
```

setNdsOther	<i>Set a user defined slot for multiple nodes</i>
-------------	---

Description

Return a tree with a user defined slot for node IDs.

Usage

```
setNdsOther(tree, ids, vals, slt_nm, ...)
```

Arguments

tree	TreeMan object
ids	id sof the nodes
vals	data for slot
slt_nm	slot name
...	plyr arguments

Details

Runs setNdOther() over multiple nodes. Parellizable.

See Also

[setNdOther](https://github.com/DomBennett/treeman/wiki/set-methods) <https://github.com/DomBennett/treeman/wiki/set-methods>

Examples

```
library(treeman)
tree <- randTree(10)
vals <- sample(0:1, size=tree['nall'], replace=TRUE)
tree <- setNdsOther(tree, tree['all'], vals, 'binary_val')
(getNdsSlt(tree, ids=tree['all'], slt_nm='binary_val'))
```

setNdSpn	<i>Set the branch length of a specific node</i>
----------	---

Description

Return a tree with the span of a node altered.

Usage

```
setNdSpn(tree, id, val, ...)
```

Arguments

tree	TreeMan object
id	id of node whose preceding edge is to be changed
val	new span
...	plyr arguments

Details

Takes a tree, a node ID and a new value for the node's preceding branch length (span). Parallelizable.

See Also

[setNdsSpn](https://github.com/DomBennett/treeman/wiki/set-methods) <https://github.com/DomBennett/treeman/wiki/set-methods>

Examples

```
library(treeman)
tree <- randTree(10)
viz(tree)
tree <- setNdsSpn(tree, id='t1', val=100)
viz(tree)
```

setNdsSpn

Set the branch lengths of specific nodes

Description

Return a tree with the span of a node altered.

Usage

```
setNdsSpn(tree, ids, vals, ...)
```

Arguments

tree	TreeMan object
ids	ids of nodes whose preceding edges are to be changed
vals	new spans
...	plyr arguments

Details

Runs setNdsSpn over multiple nodes. Parallelizable.

See Also

[setNdsSpn](https://github.com/DomBennett/treeman/wiki/set-methods) <https://github.com/DomBennett/treeman/wiki/set-methods>

Examples

```
library(treeman)
tree <- randTree(10)
# make tree taxonomic
tree <- setNdsSpn(tree, ids=tree['all'], vals=1)
```

setPD	<i>Set the phylogenetic diversity</i>
-------	---------------------------------------

Description

Return a tree with the phylogenetic diversity altered.

Usage

```
setPD(tree, val)
```

Arguments

tree	TreeMan object
val	new phylogenetic diversity

Details

Use this function to convert the phylogenetic diversity of a tree. For example, you might want to convert the tree so the sum of all branches is 1. This function will achieve that by modifying every branch, while maintaining their relative lengths.

See Also

[setAge](https://github.com/DomBennett/treeman/wiki/set-methods) <https://github.com/DomBennett/treeman/wiki/set-methods>

Examples

```
library(treeman)
tree <- randTree(10)
tree <- setPD(tree, val=1)
(tree['pd'])
```

setTol	<i>Set the extinction tolerance</i>
--------	-------------------------------------

Description

Return a tree with the tolerance altered.

Usage

```
setTol(tree, tol)
```

Arguments

tree	TreeMan object
tol	new tolerance

Details

Extant tips are determined by how close they are to zero. By default this value is 1e-8. Using this function to change the tolerance will alter the ext and exc slots.

See Also

[setNdsSpn](https://github.com/DomBennett/treeman/wiki/set-methods) <https://github.com/DomBennett/treeman/wiki/set-methods>

Examples

```
library(treeman)
tree <- randTree(10)
tree <- setTol(tree, 10)
print(tree)
```

TreeMan-class	<i>TreeMan-class</i>
---------------	----------------------

Description

S4 class for representing phylogenetic trees as a list of nodes.

Usage

```

## S4 method for signature 'TreeMan,character'
x[[i]]

## S4 method for signature 'TreeMan,character'
x[i]

## S4 method for signature 'TreeMan'
as.character(x)

## S4 method for signature 'TreeMan'
show(object)

## S4 method for signature 'TreeMan'
str(object, max.level = 2L, ...)

## S4 method for signature 'TreeMan'
print(x)

## S4 method for signature 'TreeMan'
cTrees(x, ...)

```

Arguments

x	TreeMan object
i	node ID or slot name
object	TreeMan object
max.level	str() maximum number of levels to show
...	additional tree objects

Details

A TreeMan object holds a list of nodes. The idea of the TreeMan class is to make adding and removing nodes as similar as possible to adding and removing elements in a list. Note that internal nodes and tips are both considered nodes. Trees can be polytomous but not unrooted.

Each node within the TreeMan nd1st contains the following data slots:

- id: character string for the node ID
- txnym: name of taxonomic clade (optional)
- spn: length of the preceding branch
- prid: IDs of the preceding nodes to the root
- ptid: IDs of the immediately connecting nodes
- kids: descending tip IDs
- pd: phylogenetic diversity represented by node
- prdst: pre distance(distance to root if rooted or most distal tip if unrooted)

These data slots are updated whenever a node is modified, added or removed.
See below in 'Examples' for these methods in use.

Slots

ndlst list of nodes
nds vector of node ids that are internal nodes
nnds numeric of number of internal nodes in tree
tips vector of node ids that are tips
ntips numeric of number of internal nodes in tree
all vector of all node ids
nall numeric of number of all nodes in tree
age numeric of max root to tip distance
pd numeric of total branch length of tree
ext vector of node ids of all tips with 0 age
exc vector of node ids of all tips with age > 0
wspn logical, do nodes have spans
ultr logical, do all tips end at 0
ply logical, is tree bifurcating
tol numeric of tolerance for determining extant
root character of node id of root, if no root then empty character

See Also

[randTree](#), [Node-class](#), [viz](#)

Examples

```
library(treeman)
# Generate random tree
tree <- randTree(10)
# Print to get basic stats
print(tree)
# Currently available methods
tree['tips'] # return all tips IDs
tree['nds'] # return all internal node IDs
tree['ntips'] # count all tips
tree['nnds'] # count all internal nodes
tree['root'] # identify root node
tree[['t1']] # return t1 node object
tree['pd'] # return phylogenetic diversity
tree['age'] # return age of tree
tree['ultr'] # is ultrametric?
tree['ply'] # is polytomous?
tree['ext'] # return all extant tip IDs
tree['exc'] # return all extinct tip IDs
```



```

tree <- setTol(tree, 10) # reset tolerance, default 1e-8
# now tol is higher more tips will be classed as extant
tree['ext']
# Because all nodes are lists with metadata we can readily
# get specific information on nodes of interest
nd <- tree[['n2']]
print(nd)
# And then use the same syntax for the tree
nd['age'] # .... nkids, pd, etc.

```

TreeMen-class

TreeMen-class

Description

S4 class for multiple phylogenetic trees

Usage

```

## S4 method for signature 'TreeMen'
cTrees(x, ...)

## S4 method for signature 'TreeMen,ANY'
x[[i]]

## S4 method for signature 'TreeMen,character'
x[i]

## S4 method for signature 'TreeMen'
as.character(x)

## S4 method for signature 'TreeMen'
show(object)

## S4 method for signature 'TreeMen'
str(object, max.level = 2L, ...)

## S4 method for signature 'TreeMen'
print(x)

```

Arguments

x	TreeMen object
...	additional tree objects
i	tree index (integer or character)
object	TreeMen object
max.level	str() maximum level

Slots

treelist list of TreeMan objects
 ntips sum of tips per tree
 ntrees total number of trees

See Also

[cTrees](#)

viz	<i>Method viz</i>
-----	-------------------

Description

Crude plot of TreeMan objects

Usage

```
viz(tree, taxonyms = FALSE)

## S4 method for signature 'TreeMan'
viz(tree, taxonyms = FALSE)
```

Arguments

tree	TreeMan object
taxonyms	Boolean, show taxonyms rather than IDs?

See Also

[TreeMan-class](#)

writeTree	<i>Write a Newick tree</i>
-----------	----------------------------

Description

Creates a Newick tree from a TreeMan object.

Usage

```
writeTree(tree, file, ndLabels = function(nd) { return(NULL) })
```

Arguments

tree	TreeMan object
file	file path
ndLabels	node label function

Details

The ndLabels argument can be used to add a user defined node label in the Newick tree. It should take only 1 argument, nd, the node represented as a list. It should only return a single character value that can be added to a newick string.

See Also

[readTree](#), [randTree](#), https://en.wikipedia.org/wiki/Newick_format

Examples

```
library(treeman)
tree <- randTree(10)
ndLabels <- function(n) {
  paste0(n[['id']], '_ndlabel')
}
writeTree(tree, file='example.tre', ndLabels)
file.remove('example.tre')
```

Index

- *Topic **datasets**
 - mammals, [27](#)
- *Topic **evolution**
 - treeman-package, [3](#)
- *Topic **macroevolution**
 - treeman-package, [3](#)
- *Topic **phylogeny**
 - treeman-package, [3](#)
- *Topic **tree simulation**
 - treeman-package, [3](#)
- [,Node,character-method (Node-class), [28](#)
- [,TreeMan,character-method (TreeMan-class), [38](#)
- [,TreeMen,character-method (TreeMen-class), [41](#)
- [[,TreeMan,character-method (TreeMan-class), [38](#)
- [[,TreeMen,ANY-method (TreeMen-class), [41](#)

- addTip, [3](#), [31](#)
- as-TreeMen, [4](#)
- as.character,Node-method (Node-class), [28](#)
- as.character,TreeMan-method (TreeMan-class), [38](#)
- as.character,TreeMen-method (TreeMen-class), [41](#)

- calcDstBLD, [5](#), [6–8](#)
- calcDstMtrx, [6](#)
- calcDstRF, [5](#), [6](#), [6](#), [8](#)
- calcDstTrp, [5–7](#), [7](#)
- calcFrPrp, [8](#), [12](#)
- calcNdBlnc, [9](#), [10](#)
- calcNdsBlnc, [9](#), [10](#)
- calcOvr1p, [10](#), [12](#)
- calcPhyDv, [9](#), [11](#), [11](#)
- cTrees, [12](#), [28](#), [42](#)

- cTrees,TreeMan-method (TreeMan-class), [38](#)
- cTrees,TreeMen-method (TreeMen-class), [41](#)

- getNdAge, [13](#), [16](#), [25](#)
- getNdKids, [13](#), [17](#)
- getNdLng, [14](#), [18](#), [27](#)
- getNdPrid, [15](#), [16](#), [19](#), [20](#)
- getNdPtid, [15](#), [15](#), [19](#), [20](#)
- getNdsAge, [13](#), [16](#), [25](#)
- getNdsKids, [14](#), [17](#)
- getNdsLng, [14](#), [17](#), [27](#)
- getNdSlt, [18](#), [20](#)
- getNdsPrid, [15](#), [16](#), [19](#), [20](#)
- getNdsPtid, [15](#), [16](#), [19](#), [19](#)
- getNdsSlt, [18](#), [20](#)
- getNdsSstr, [21](#), [22](#)
- getNdSstr, [21](#), [22](#)
- getOtgrp, [22](#)
- getPath, [23](#)
- getPrnt, [13](#), [24](#), [26](#)
- getSpnAge, [13](#), [16](#), [24](#), [25](#)
- getSpnsAge, [13](#), [16](#), [25](#), [25](#)
- getSubtree, [24](#), [26](#)
- getTxnmys, [26](#)

- mammals, [27](#)

- Node-class, [28](#)
- Node-method (Node-class), [28](#)

- pinTips, [29](#)
- print,Node-method (Node-class), [28](#)
- print,TreeMan-method (TreeMan-class), [38](#)
- print,TreeMen-method (TreeMen-class), [41](#)

- randTree, [30](#), [31](#), [40](#), [43](#)
- readTree, [30](#), [43](#)
- rmTip, [4](#), [31](#)

setAge, [32](#), [37](#)
setNdID, [32](#), [34](#)
setNdOther, [33](#), [35](#)
setNdsID, [33](#), [34](#)
setNdsOther, [33](#), [35](#)
setNdSpn, [35](#), [37](#)
setNdsSpn, [36](#), [36](#), [38](#)
setPD, [32](#), [37](#)
setTol, [38](#)
show, Node-method (Node-class), [28](#)
show, TreeMan-method (TreeMan-class), [38](#)
show, TreeMen-method (TreeMen-class), [41](#)
str, TreeMan-method (TreeMan-class), [38](#)
str, TreeMen-method (TreeMen-class), [41](#)

treeman (treeman-package), [3](#)
TreeMan-class, [38](#)
TreeMan-method (TreeMan-class), [38](#)
treeman-package, [3](#)
TreeMen-class, [41](#)
TreeMen-method (TreeMen-class), [41](#)

viz, [40](#), [42](#)
viz, TreeMan-method (viz), [42](#)

writeTree, [31](#), [42](#)