Package: tidytree (via r-universe)

October 25, 2024

Title A Tidy Tool for Phylogenetic Tree Data Manipulation

Version 0.4.6.002

Description Phylogenetic tree generally contains multiple components including node, edge, branch and associated data. 'tidytree' provides an approach to convert tree object to tidy data frame as well as provides tidy interfaces to manipulate tree data.

Depends R (>= 3.4.0)

Imports ape, dplyr, lazyeval, magrittr, methods, rlang, tibble, tidyr, tidyselect, yulab.utils (>= 0.1.5), pillar, cli

Suggests knitr, rmarkdown, prettydoc, testthat, utils

VignetteBuilder knitr

ByteCompile true

License Artistic-2.0

URL `https://www.amazon.com/Integration-Manipulation-Visualization-Phylogenetic-Computational-ebook/dp/B0B5NLZR1Z/`

BugReports `https://github.com/YuLab-SMU/tidytree/issues`

Encoding UTF-8

RoxygenNote 7.3.2

Roxygen list(markdown = TRUE)

Repository https://yulab-smu.r-universe.dev

RemoteUrl https://github.com/yulab-smu/tidytree

RemoteRef HEAD

RemoteSha 1115d91fe184856b7d6f72ebb79fa29308dd7b60

Contents

`ancestor` *ancestor*

Description

access ancestor data

Usage

```
ancestor(.data, .node, ...)

## S3 method for class 'tbl_tree'
ancestor(.data, .node, ...)
```

Arguments

`.data`	phylo or tbl_tree object
`.node`	node number
`...`	additional parameters

Value

ancestor data

Author(s)

Guangchuang Yu

Examples

```
library(ape)
tree <- rtree(4)
x <- as_tibble(tree)
ancestor(x, 3)
```

`as.treedata` *as.treedata*

Description

convert a tree object to treedata object

Usage

```
as.treedata(tree, ...)

## S3 method for class 'tbl_tree'
as.treedata(tree, ...)
```

Arguments

`tree`	tree object
`...`	additional parameters

Value

treedata object

Examples

```
library(ape)
set.seed(2017)
tree <- rtree(4)
d <- tibble(label = paste0('t', 1:4),
           trait = rnorm(4))
x <- as_tibble(tree)
full_join(x, d, by = 'label') %>% as.treedata
```

`as.ultrametric` *as.ultrametric*

Description

as.ultrametric

Usage

```
as.ultrametric(tree, ...)
```

Arguments

`tree`	tree object
`...`	additional parameters

Value

treedata or phylo object

`child` *child*

Description

access child data

Usage

```
child(.data, .node, ...)

## S3 method for class 'tbl_tree'
child(.data, .node, ...)
```

Arguments

`.data`	phylo or tbl_tree object
`.node`	node number
`...`	additional parameters

Value

child data

Author(s)

Guangchuang Yu

Examples

```
library(ape)
tree <- rtree(4)
x <- as_tibble(tree)
child(x, 4)
```

`drop.tip` *drop.tip method*

Description

drop.tip method

Usage

```
drop.tip(object, tip, ...)

keep.tip(object, tip, ...)

## S4 method for signature 'treedata'
drop.tip(object, tip, ...)

## S4 method for signature 'phylo'
drop.tip(object, tip, ...)

## S4 method for signature 'treedata'
keep.tip(object, tip, ...)

## S4 method for signature 'phylo'
keep.tip(object, tip, ...)
```

Arguments

`object`	A treedata or phylo object
`tip`	a vector of mode numeric or character specifying the tips to delete
`...`	additional parameters

Value

updated object

Author(s)

Casey Dunn `http://dunnlab.org` and Guangchuang Yu `https://guangchuangyu.github.io`

Source

drop.tip for phylo object is a wrapper method of ape::drop.tip from the ape package. The documentation you should read for the drop.tip function can be found here: drop.tip

See Also

drop.tip

Examples

```
library(tidytree)
set.seed(123)
tr <- ape::rtree(6)
da <- data.frame(id=tip.label(tr), value = letters[seq_len(6)])
trda <- tr %>% dplyr::left_join(da, by = c('label'='id'))
tr1 <- drop.tip(tr, c("t2", "t1"))
tr2 <- keep.tip(tr, c("t2", "t1"))
```

`get.data` *get.data method*

Description

get.data method

get.data method

Usage

```
get.data(object, ...)

## S4 method for signature 'treedata'
get.data(object)
```

Arguments

`object`	`treedata` object
`...`	additional parameter

Value

associated data of phylogeny

`get.fields` *get.fields method*

Description

get.fields method

Usage

```
get.fields(object, ...)

## S4 method for signature 'treedata'
get.fields(object)
```

Arguments

`object`	`treedata` object
`...`	additional parameter

Value

available annotation variables

`get.treetext` *get.treetext method*

Description

access tree text (newick text) from tree object

Usage

```
get.treetext(object, ...)

## S4 method for signature 'treedata'
get.treetext(object)
```

Arguments

`object`	treedata object
`...`	additional parameter

Value

phylo object

`getNodeNum` *getNodeNum*

Description

calculate total number of nodes

Usage

```
getNodeNum(tree)
```

Arguments

`tree`	tree object

Value

number

Author(s)

Guangchuang Yu

Examples

```
getNodeNum(rtree(30))
```

`get_tree_data` *get_tree_data*

Description

get associated data stored in treedata object

Usage

```
get_tree_data(tree_object)
```

Arguments

`tree_object`	a `treedata` object

Value

tbl_df

Author(s)

guangchuang yu

`groupClade` *groupClade*

Description

grouping clades

Usage

```
groupClade(.data, .node, group_name = "group", overlap = "overwrite", ...)
```

Arguments

`.data`	tree object (phylo, treedata, tbl_tree, ggtree etc.)
`.node`	selected nodes
`group_name`	character the name of the group cluster, default is `group`.
`overlap`	character one of `overwrite`,`origin` and `abandon`, default is `overwrite`.
`...`	additional parameter

Value

updated tree with group information or group index

Author(s)

Guangchuang Yu

`groupOTU` *groupOTU*

Description

grouping OTUs

Usage

```
groupOTU(.data, .node, group_name = "group", ...)
```

Arguments

`.data`	tree object (phylo, treedata, tbl_tree, ggtree etc.)
`.node`	selected nodes
`group_name`	character the name of the group cluster, default is `group`.
`...`	additional parameter

Value

updated tree with group information or group index

Author(s)

Guangchuang Yu

`isTip` *isTip*

Description

whether the node is a tip

Usage

```
isTip(.data, .node, ...)

## S3 method for class 'tbl_tree'
isTip(.data, .node, ...)

## S3 method for class 'phylo'
isTip(.data, .node, ...)

## S3 method for class 'treedata'
isTip(.data, .node, ...)
```

Arguments

`.data`	phylo, treedata or tbl_tree object
`.node`	node number
`...`	additional parameters

Value

logical value

Author(s)

Guangchuang Yu

`MRCA` *MRCA*

Description

access most recent common ancestor data

Usage

```
MRCA(.data, ...)
```

Arguments

`.data`	phylo or tbl_tree object
`...`	additional parameters

Value

MRCA data

Author(s)

Guangchuang Yu

`Nnode.treedata` *Nnode*

Description

number of nodes

Usage

```
## S3 method for class 'treedata'
Nnode(phy, internal.only = TRUE, ...)
```

Arguments

`phy`	treedata object
`internal.only`	whether only count internal nodes
`...`	additional parameters

Value

number of nodes

Author(s)

Guangchuang Yu

Examples

```
Nnode(rtree(30))
```

`node.label` *extract the node label of phylo, treedata or tbl_tree*

Description

extract the node label of phylo, treedata or tbl_tree

Usage

```
node.label(x, node = "internal", ...)
```

Arguments

`x`	object, should be one of `treedata`,`phylo` or `tbl_tree`.
`node`	character, to extract which type node label, default is `internal`, should be one of `internal`, `external`, `all`, `tip`.
`...`	additional parameters.

Value

label character vector.

`nodeid` *nodeid*

Description

convert tree label to internal node number

Usage

```
nodeid(tree, label)
```

Arguments

`tree`	tree object
`label`	tip/node label(s)

Value

node number

Author(s)

Guangchuang Yu

`nodelab` *nodelab*

Description

convert internal node number tip/node label

Usage

```
nodelab(tree, id)
```

Arguments

`tree`	tree object
`id`	node number

Value

tip/node label(s)

Author(s)

Guangchuang Yu

`offspring` *offspring*

Description

access offspring data

Usage

```
offspring(.data, .node, tiponly, self_include, ...)

## S3 method for class 'tbl_tree'
offspring(.data, .node, tiponly = FALSE, self_include = FALSE, ...)
```

Arguments

`.data`	phylo or tbl_tree object
`.node`	node number
`tiponly`	whether only return tip nodes
`self_include`	whether include the input node, only applicable for tiponly = FALSE
`...`	additional parameters

Value

offspring data

Author(s)

Guangchuang Yu

Examples

```
library(ape)
tree <- rtree(4)
x <- as_tibble(tree)
offspring(x, 4)
```

`parent` *parent*

Description

access parent data

Usage

```
parent(.data, .node, ...)

## S3 method for class 'tbl_tree'
parent(.data, .node, ...)
```

Arguments

`.data`	phylo or tbl_tree object
`.node`	node number
`...`	additional parameters

Value

parent data

Author(s)

Guangchuang Yu

Examples

```
library(ape)
tree <- rtree(4)
x <- as_tibble(tree)
parent(x, 2)
```

`root.treedata` *root*

Description

re-root a tree

Usage

```
## S3 method for class 'treedata'
root(phy, outgroup, node = NULL, edgelabel = TRUE, ...)
```

Arguments

`phy`	tree object
`outgroup`	a vector of mode numeric or character specifying the new outgroup
`node`	node to reroot
`edgelabel`	a logical value specifying whether to treat node labels as edge labels and thus eventually switching them so that they are associated with the correct edges.
`...`	additional parameters passed to ape::root.phylo

Value

rerooted treedata

`rootnode` *rootnode*

Description

access root node data

Usage

```
rootnode(.data, ...)
```

Arguments

`.data`	phylo or tbl_tree object
`...`	additional parameters

Value

root node data

Author(s)

Guangchuang Yu

`show` *show method*

Description

show method for `treedata` instance

Usage

```
show(object)
```

Arguments

`object`	`treedata` object

Value

print info

Author(s)

Guangchuang Yu `https://guangchuangyu.github.io`

`sibling` *sibling*

Description

access sibling data

Usage

```
sibling(.data, ...)
```

Arguments

`.data`	phylo or tbl_tree object
`...`	additional parameters

Value

sibling

Author(s)

Guangchuang Yu

`td-label-assign` *the tip or internal node label assign of tbl_tree phylo and treedata*

Description

the tip or internal node label assign of tbl_tree phylo and treedata

Usage

```
tip.label(x) <- value

node.label(x) <- value

## S3 replacement method for class 'phylo'
node.label(x) <- value

## S3 replacement method for class 'treedata'
node.label(x) <- value

## S3 replacement method for class 'tbl_tree'
node.label(x) <- value
```

```
## S3 replacement method for class 'phylo'
tip.label(x) <- value

## S3 replacement method for class 'treedata'
tip.label(x) <- value

## S3 replacement method for class 'tbl_tree'
tip.label(x) <- value
```

Arguments

`x`	object, should be one of `tbl_tree`, `phylo` or `treedata`
`value`	character, the character vector

`tip.label` *extract the tip label of phylo treedata or tbl_tree*

Description

extract the tip label of phylo treedata or tbl_tree

Usage

```
tip.label(x, ...)
```

Arguments

`x`	object, should be one of `treedata`,`phylo` or `tbl_tree`.
`...`	additional parameters.

`treedata` *treedata*

Description

treedata object contructor

Usage

```
treedata(...)
```

Arguments

`...`	parameters

Value

treedata object

Author(s)

guangchuang yu

`treedata-class` *Class "treedata" This class stores phylogenetic tree with associated data*

Description

Class "treedata" This class stores phylogenetic tree with associated data

Slots

`file` tree file

`treetext` newick tree string

`phylo` phylo object for tree structure

`data` associated data

`extraInfo` extra information, reserve for merge_tree

`tip_seq` tip sequences

`anc_seq` ancestral sequences

`seq_type` sequence type, one of NT or AA

`tipseq_file` tip sequence file

`ancseq_file` ancestral sequence file

`info` extra information, e.g. metadata, software version etc.

Author(s)

Guangchuang Yu `https://guangchuangyu.github.io`

`tree_subset` *Subset tree objects by related nodes*

Description

This function allows for a tree object to be subset by specifying a node and returns all related nodes within a selected number of levels

Usage

```
tree_subset(
  tree,
  node,
  levels_back = 5,
  group_node = TRUE,
  group_name = "group",
  root_edge = TRUE
)

## S3 method for class 'phylo'
tree_subset(
  tree,
  node,
  levels_back = 5,
  group_node = TRUE,
  group_name = "group",
  root_edge = TRUE
)

## S3 method for class 'treedata'
tree_subset(
  tree,
  node,
  levels_back = 5,
  group_node = TRUE,
  group_name = "group",
  root_edge = TRUE
)
```

Arguments

`tree`	a tree object of class phylo
`node`	either a tip label or a node number for the given tree that will be the focus of the subsetted tree
`levels_back`	a number specifying how many nodes back from the selected node the subsetted tree should include

`group_node`	whether add grouping information of selected node
`group_name`	group name (default 'group') for storing grouping information if group_node = TRUE
`root_edge`	If TRUE (by default), set root.edge to path length of orginal root to the root of subset tree

Details

This function will take a tree and a specified node from that tree and subset the tree showing all relatives back to a specified number of nodes. This function allows for a combination of `ancestor` and `offspring` to return a subsetted tree that is of class phylo. This allows for easy graphing of the tree with `ggtree`

Examples

```
set.seed(123)
tree <- ape::rtree(6)
sub_tree <- tree_subset(tree, node = "t1", levels_back = 2)
```

Index