Package 'treestartr'

October 14, 2022

Type Package

Title Generate Starting Trees for Combined Molecular, Morphological
and Stratigraphic Data

Version 0.1.0

Description Combine a list of taxa with a phylogeny to generate a starting tree for use in
total evidence dating analyses.

URL https://ropensci.github.io/treeStartR/

BugReports https://github.com/ropensci/treeStartR/issues

License MIT + file LICENSE

Depends R (>= 3.0.0), phytools (>= 0.6-4), ape (>= 5.0)

Suggests knitr, testthat, rmarkdown, covr

Encoding UTF-8

LazyData true

RoxygenNote 6.1.0

NeedsCompilation no

Author April Wright [aut, cre]

Maintainer April Wright <april.wright@selu.edu>

Repository CRAN

Date/Publication 2019-01-09 15:10:03 UTC

R topics documented:

---

absent_tippr	*Add tips to tree via user input*

Description

Add tips not on existing tree to the tree via user input, if they do not have congeners on the tree. This function will generate a tree, with nodes numbered. For each tip to be added, the user will be asked #' to enter the number of the node they would like the tip to subtend.

Usage

```
absent_tippr(tree, absent_list)
```

Arguments

tree	Starting tree; object of type phylo
absent_list	Vector of taxa in the total dataset that are not on the tree

Value

tree Phylo object containing the starting tree, and all tips that were added.

Examples

```
## Not run: new <- absent_tippr(tree, absent_list)
```

---

bears	*bears*

Description

A dataset containing 18 taxa and 62 characters for bears

Usage

```
data(bears)
```

Format

A collection of objects representing function inputs:

tax_frame a two-column dataframe of taxon names and ages

absent_list A list of taxa in the total data set, but are not on the tree

mrca_df a two-column dataframe of taxa to be placed, and a set of taxa that indicate the MRCA which they will subtend

tree A phylogenetic tree to which tips will be added

Source

https://github.com/revbayes/revbayes_tutorial/tree/master/RB_TotalEvidenceDating_FBD_Tutorial/data

---

dataf_parsr	*Detect file format of taxon file*

Description

Detect file format of the total taxon list (molecular, morphological, and stratigraphic) and maximum age of fossil, see RevBayes total -evidence fossil file for an example of this.

Usage

```
dataf_parsr(dataf)
```

Arguments

dataf	A data frame with one column containing the taxon name of tips in phylogenetic tree. Optionally, a second column can indicate the maximum age of the tip. If the tip is extant, use 0.0 as the age. Can be CSV or TSV.

Value

tax_frame Dataframe containing the total set of tips on the tree

Examples

```
## Not run: tax_frame <- dataf_parsr(dataf)
```

---

echo_rb   *Print RevBayes-formatted clade constriants.*

Description

Once a tip has been added to the tree, print the subtree to which the tip was added. The output of
this function is printed for use as a RevBayes clade constraint object.

Usage

```
echo_rb(tree, mrca_list, tip)
```

Arguments

tree	Tree to which the tip will be added; object of type phylo
mrca_list	Vector of taxa. The added tip will subtend the MRCA of these taxa
tip	Taxon which will be added to the tree.

Value

vector A comma-separated list of all the taxa in the subtree to which the tip was added.

Examples

```
 mrca_list <- c("Ursus_arctos", "Ursus_spelaeus", "Ursus_americanus")
clade_constraint <- echo_rb(tree, mrca_list, "Ursus_abstrusus")
```

---

echo_subtree   *Print subtrees with tips added.*

Description

Once a tip has been added to the tree, print the newick-formatted subtree to which the tip was added.

Usage

```
echo_subtree(tree, mrca_list, tip)
```

Arguments

tree	Tree to which the tip will be added; object of type phylo
mrca_list	Vector of taxa. The added tip will subtend the MRCA of these taxa
tip	Taxon which will be added to the tree.

Value

tree subtree to which tip was added

Examples

```
mrca_list <- c("Ursus_arctos", "Ursus_spelaeus", "Ursus_americanus")
tree <- text_placr(tree, mrca_df)
echo_sub <- echo_subtree(tree, mrca_list, "Ursus_abstrusus")
```

---

fileformat   *Utility function to check if file is a csv or tsv.*

Description

Utility function to check if file is a csv or tsv.

Usage

```
fileformat(dataf)
```

Arguments

dataf	Path to input file

Value

None

---

genera_strippr   *Check if tip is on tree*

Description

Test which taxa in the total set of taxa are present on the tree, and which need to be added. The tax_frame object is assumed to be a dataframe, minimally with a column labeled "taxon". If this column does not exist, the first column in the dataframe will be assumed to contain the taxon information.

Usage

```
genera_strippr(tree, tax_frame)
```

Arguments

tree   Starting phylogeny, of type phylo

tax_frame   Total set of taxa on tree, as dataframe.

Value

absent_list of taxa that are present in the total set of trees, but not the starting tree

Examples

```
absent_taxa <- genera_strippr(tree, tax_frame = tax_frame)
```

---

get_found   *Get dataframe of tips that do not have congeners on tree.*

Description

Determine which tips that are not on the tree, and checks if they have congeners on the tree.

Usage

```
get_found(absent_list, tree)
```

Arguments

absent_list   Vector of taxa in the total dataset that are not on the tree

tree   Starting tree; object of type phylo

Value

found_df Dataframe objects expressing the tips that are not on the tree, if they have congeners on the tree

Examples

```
has_congeners <- treestartr:::get_found(absent_list, tree)
```

---

get_lost   *Determine if a tip is present on a tree*

Description

Determine which tips that are not on the tree, puts them in a dataframe if they do not have congeners on the tree

Usage

```
get_lost(absent_list, tree)
```

Arguments

absent_list   Vector of taxa in the total dataset that are not on the tree

tree   Starting tree; object of type phylo

Value

not_found_df Dataframe objects expressing the tips that are not on the tree, and don"t have congeners

Examples

```
no_congeners <- treestartr:::get_lost(absent_list, tree)
```

---

make_absentdf   *Determine which tips are not on the tree, and get their genera.*

Description

Determine which tips are not on the tree, and get their genera.

Usage

```
make_absentdf(absent_list)
```

Arguments

absent_list   Vector of taxa in the total dataset that are not on the tree

Value

absent_df Dataframe objects expressing which tips are in the total set but not the tree, and their
genera

Examples

```
not_present <- treestartr:::make_absentdf(absent_list)
```

---

make_treedf   *Determine which tips are on the tree, and get their genera.*

Description

Determine which tips are on the tree, and get their genera.

Usage

```
make_treedf(tree)
```

Arguments

tree   Starting tree; object of type phylo

Value

tree_df Dataframe objects expressing the tips, and their genera

---

present_tippr   *Add tips to the tree based on taxonomy.*

Description

Add tips not on existing tree to the tree via an MRCA node, if they have congeners. This function
will look to see if any tips on the tree have the same genus as the tip to be added. If there are multiple
members of the genus, the tip will be added subtending the MRCA of all present congeners. If there
is member of the genus, the tip will be added subtending the parent node of the congener.

Usage

```
present_tippr(tree, absent_list, echo_subtrees = NULL,
  echo_revbayes = NULL)
```

Arguments

`tree`	Starting tree; object of type phylo
`absent_list`	Vector of taxa in the total dataset that are not on the tree
`echo_subtrees`	Boolean; Print newick subtree with missing taxa added to screen. Default FALSE.
`echo_revbayes`	Boolean; Print clade constraints with missing taxa added to screen, formatted for RevBayes fossilized birth-death analysis. Default FALSE.

Value

tree. Phylo object containing the starting tree, and all tips that were added.

Examples

```
genera_tree <- present_tippr(tree, absent_list)
```

---

rand_absent_tippr　　　*Random addition of taxa*

Description

Add tips not on existing tree to the tree at random, if they do not have congeners on the tree.

Usage

```
rand_absent_tippr(tree, absent_list, echo_subtrees = NULL,
  echo_revbayes = NULL)
```

Arguments

`tree`	Starting tree; object of type phylo
`absent_list`	Vector of taxa in the total dataset that are not on the tree
`echo_subtrees`	Boolean; Print newick subtree with missing taxa added to screen. Default FALSE.
`echo_revbayes`	Boolean; Print clade constraints with missing taxa added to screen, formatted for RevBayes fossilized birth-death analysis. Default FALSE.

Value

tree Phylo object containing the starting tree, and all tips that were added.

Examples

```
new_tree <- rand_absent_tippr(tree, absent_list)
```

---

taxon_testr *Utility function to check taxon_names in parsed data file*

Description

Utility function to check taxon_names in parsed data file

Usage

```
taxon_testr(tax_frame)
```

Arguments

tax_frame   Dataframe of taxa parsed from datafile.

Value

None

---

text_placr *Add tips to tree via taxon list*

Description

Add tips according to csv or tsv file of taxon names and taxa that form the clade into which you"d like to insert the tip. One column should be called 'taxon', and should contain the taxon to be placed. The other column should be called 'clade' and contain the taxon with which the taxon to be placed will form a group. Each member of the clade will be placed on its own line. An example of this file can be seen in 'inst/extdat a/mrca_df.tsv'

Usage

```
text_placr(tree, mrca_df, echo_subtrees = NULL, echo_revbayes = NULL)
```

Arguments

tree	Starting tree; object of type phylo
mrca_df	Dataframe containing a column of the taxa you'd like to place and one column with the clade into which you'd like to place it
echo_subtrees	Boolean; Print newick subtree with missing taxa added to screen. Default FALSE.
echo_revbayes	Boolean; Print clade constraints with missing taxa added to screen, formatted for RevBayes fossilized birth-death analysis. Default FALSE.

Value

tree Phylo object containing the starting tree, and all tips that were added.

Examples

```
text_placr(tree, mrca_df)
```

| treestartr | *treestartr: A package for generating reasonable starting trees for phylogenetic analysis.* |

Description

treestartr provides multiple ways to add tips to a starting tree

treestartr functions

absent_tippr, echo_rb, echo_subtree, present_tippr, rand_absent_tippr, text_placr

Index