

# Package ‘phylogram’

April 13, 2018

**Type** Package

**Title** Dendrograms for Evolutionary Analysis

**Version** 2.0.1

**Author** Shaun Wilkinson [aut, cre]

**Maintainer** Shaun Wilkinson <shaunwilkinson@gmail.com>

**Description** Contains functions for importing and exporting 'dendrogram' objects in parenthetic text format, as well as several functions for command-line tree manipulation.

**License** GPL-3

**LazyData** TRUE

**URL** <http://github.com/shaunwilkinson/phylogram>

**BugReports** <http://github.com/shaunwilkinson/phylogram/issues>

**Encoding** UTF-8

**Depends** R (>= 3.0.0), methods, stats

**Imports** ape (>= 4.0),

**Suggests** knitr, rmarkdown, testthat

**RoxygenNote** 6.0.1

**VignetteBuilder** knitr

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2018-04-13 08:44:44 UTC

## R topics documented:

as.dendrogram.phylo . . . . .	2
as.phylo.dendrogram . . . . .	3
ladder . . . . .	4
phylogram . . . . .	5
prune . . . . .	6

read.dendrogram . . . . .	7
remidpoint . . . . .	8
reposition . . . . .	9
ultrametricize . . . . .	10
write.dendrogram . . . . .	11

<b>Index</b>	<b>12</b>
--------------	-----------

---

as.dendrogram.phylo	<i>Convert "phylo" objects to dendrograms.</i>
---------------------	--

---

## Description

These functions are used for converting dendrograms to "phylo" objects and *vice versa*.

## Usage

```
## S3 method for class 'phylo'
as.dendrogram(object, ...)
```

## Arguments

object	a "phylo" object.
...	further arguments to be passed between methods.

## Details

These functions currently work by temporarily writing a tree to Newick text and then parsing the string using either [read.dendrogram](#) or [read.tree](#). A faster implementation that avoids the transformation to text strings will be available in a future version.

## Value

an object of class "dendrogram".

## Author(s)

Shaun Wilkinson

## Examples

```
newick <- "(A:0.1,B:0.2,(C:0.3,D:0.4):0.5);"
x <- read.dendrogram(text = newick)
y <- as.phylo(x)
z <- as.dendrogram(y)
identical(x, z)
```

---

as.phylo.dendrogram    *Convert dendrogram to "phylo" object.*

---

### Description

Functions for converting dendrograms to "phylo" objects and *vice versa*.

### Usage

```
## S3 method for class 'dendrogram'  
as.phylo(x, ...)
```

### Arguments

x                    a dendrogram object.  
...                   further arguments to be passed between methods.

### Details

These functions currently work by temporarily writing a tree to Newick text and then parsing the string using either [read.dendrogram](#) or [read.tree](#). A faster implementation that avoids the transformation to text strings will be available in a future version.

### Value

an object of class "phylo".

### Author(s)

Shaun Wilkinson

### Examples

```
newick <- "(A:0.1,B:0.2,(C:0.3,D:0.4):0.5);"  
x <- read.dendrogram(text = newick)  
y <- as.phylo(x)  
z <- as.dendrogram(y)  
identical(x, z)
```

---

ladder	<i>Reorder tree branches in ladderized pattern.</i>
--------	---

---

### Description

This function ladderizes the branches of a dendrogram object to aid in visual interpretation.

### Usage

```
ladder(x, decreasing = FALSE)
```

### Arguments

x	an object of class "dendrogram".
decreasing	logical indicating whether the tree should be ladderized upwards or downwards. Defaults to FALSE (downwards).

### Details

This function is the dendrogram analogue of the [ladderize](#) function in the [ape](#) package (Paradis et al 2004, 2012).

### Value

Returns an object of class dendrogram.

### Author(s)

Shaun Wilkinson

### References

Paradis E, Claude J, Strimmer K, (2004) APE: analyses of phylogenetics and evolution in R language. *Bioinformatics* **20**, 289-290.  
Paradis E (2012) Analysis of Phylogenetics and Evolution with R (Second Edition). Springer, New York.

### See Also

The [ladderize](#) function in the [ape](#) package performs a similar operation for objects of class "phylo".

### Examples

```
x <- read.dendrogram(text = "(A:0.1,B:0.2,(C:0.3,D:0.4):0.5);")  
plot(x, horiz = TRUE)  
x <- ladder(x, decreasing = TRUE)  
plot(x, horiz = TRUE)
```

## Description

The phylogram package contains functions for importing and exporting dendrogram objects in the Newick parenthetic text format, as well as several functions for command-line tree manipulation.

## Functions

A brief description of the primary **phylogram** functions are provided with links to their help pages below.

## File import/export

- [read.dendrogram](#) is a text parser that converts parenthetic text (Newick strings) into objects of class "dendrogram"
- [write.dendrogram](#) outputs an object of class "dendrogram" to a text string or file in Newick/New Hampshire format

## Object conversion

- [as.phylo.dendrogram](#) converts a dendrogram to an object of class "phylo" "dendrogram"
- [as.dendrogram.phylo](#) converts a "phylo" object to a dendrogram

## Tree editing and manipulation

- [prune](#) remove branches from a dendrogram object based on regular expression pattern matching
- [ladder](#) reorders the branches of a dendrogram object to aid visualization
- [remidpoint](#) recursively sets "midpoint" and "members" attributes for a nested list/dendrogram object
- [reposition](#) shifts a dendrogram object up or down (or sideways if plotted horizontally)
- [ultrametricize](#) modifies the "height" attributes of the nodes such that all leaves terminate at zero

---

prune

*Remove tree nodes by regular expression pattern matching.*

---

### Description

"prune" takes an object of class "dendrogram" and removes all branches whose branch labels match a given regular expression.

### Usage

```
prune(tree, pattern, invert = FALSE, untag = FALSE, ...)
```

### Arguments

tree	an object of class "dendrogram".
pattern	a regular expression.
invert	logical indicating whether the branches whose labels match the regular expression provided in "pattern" should be discarded and the others kept (FALSE; default) or vice versa. Nodes without "label" attributes are ignored.
untag	logical (used only when invert = TRUE). Indicates whether the specified pattern should be removed from the branch labels in the returned object.
...	further arguments to be passed to <code>grep1</code> and <code>gsub</code> .

### Details

This function recursively tests the "label" attribute of each dendrogram node (including non-leaf inner nodes if applicable) for the specified pattern, removing those that register a positive hit. Note that positive matching inner nodes are removed along with all of their sub-nodes, regardless of whether the "label" attributes of the sub-nodes match the pattern.

### Value

Returns an object of class "dendrogram".

### Author(s)

Shaun Wilkinson

### See Also

The [drop.tip](#) function in the [ape](#) package performs a similar operation for objects of class "phylo". See [regex](#) for help with compiling regular expressions.

## Examples

```
x <- read.dendrogram(text = "(A:0.1,B:0.2,(C:0.3,D:0.4):0.5);")
plot(x, horiz = TRUE)
x <- prune(x, pattern = "^A$")
plot(x, horiz = TRUE)
```

---

read.dendrogram	<i>Read a dendrogram from parenthetic text.</i>
-----------------	---

---

## Description

read.dendrogram parses a text file or character string in Newick (New Hampshire) format and creates an object of class "dendrogram".

## Usage

```
read.dendrogram(file = "", text = NULL, edges = TRUE, ...)
```

## Arguments

file	character string giving a valid path to the file from where to read the data.
text	optional character string in lieu of a "file" argument. If a text argument is provided instead of a file path, the data are read via a text connection.
edges	logical indicating whether edge weights provided in the Newick string should be retained in the returned object (defaults to TRUE).
...	further arguments to be passed to scan.

## Details

There are varying interpretations of the Newick/New Hampshire text format. This function tries to adhere to the Felsenstein standard outlined [here](#). The function supports weighted edges, labels with special metacharacters (enclosed in single quotation marks), comments (enclosed in square brackets; ignored by the parser), multifurcating nodes, and both rooted and unrooted trees. Comments enclosed in square brackets are also discarded. Inner-node labels (for example "(B:6.0,(A:5.0,C:3.0,E:4.0)Ancestor1:5.0,D:1.1)") are also currently ignored; however, the parsing of "label" attributes for non-leaf dendrogram nodes will be available in a future version.

## Value

Returns an object of class "dendrogram".

## Author(s)

Shaun Wilkinson

## References

<http://evolution.genetics.washington.edu/phylip/newicktree.html> [http://evolution.genetics.washington.edu/phylip/newick\\_doc.html](http://evolution.genetics.washington.edu/phylip/newick_doc.html)

## See Also

`write.dendrogram` writes an object of class "dendrogram" to a text string. The `read.tree` function in the `ape` package performs a similar operation for objects of class "phylo" and "multiPhylo".

## Examples

```
newick <- "(A:0.1,B:0.2,(C:0.3,D:0.4):0.5);"  
x <- read.dendrogram(text = newick)  
plot(x, horiz = TRUE)
```

---

remidpoint

*Set dendrogram attributes for a nested list.*

---

## Description

`remidpoint` is a helper function used for manually creating "dendrogram" objects from nested lists. The function recursively assigns the necessary 'midpoint' and 'members' attributes at each node.

## Usage

```
remidpoint(x)
```

## Arguments

`x` a nested list, possibly of class "dendrogram"

## Value

returns a nested list, or an object of class "dendrogram" depending on the class of the input object.

## Author(s)

Shaun Wilkinson



**Examples**

```
## manually create a small dendrogram with three members, A, B and C
x <- list("A", list("B", "C"))
attr(x[[1]], "leaf") <- TRUE
attr(x[[2]][[1]], "leaf") <- TRUE
attr(x[[2]][[2]], "leaf") <- TRUE
attr(x[[1]], "label") <- "A"
attr(x[[2]][[1]], "label") <- "B"
attr(x[[2]][[2]], "label") <- "C"
attr(x, "height") <- 1
attr(x[[1]], "height") <- 0
attr(x[[2]], "height") <- 0.5
attr(x[[2]][[1]], "height") <- 0
attr(x[[2]][[2]], "height") <- 0
x <- remidpoint(x)
class(x) <- "dendrogram"
plot(x, horiz = TRUE)
```

---

reposition

*Reset dendrogram height attributes.*

---

**Description**

reposition is a helper function used for manually creating "dendrogram" objects from nested lists. The function recursively reassigns the 'height' attributes at each node by a given constant.

**Usage**

```
reposition(x, shift = "reset")
```

**Arguments**

x	an object of class "dendrogram".
shift	either the character string "reset" (shift the graph so that the height of the farthest leaf from the root is zero), or a numeric value giving the amount to shift the graph along the primary axis.

**Value**

Returns an object of class "dendrogram".

**Author(s)**

Shaun Wilkinson

**Examples**

```
x <- read.dendrogram(text = "(A:0.1,B:0.2,(C:0.3,D:0.4):0.5);")
plot(x, horiz = TRUE)
x <- reposition(x)
plot(x, horiz = TRUE)
```

---

ultrametricize      *Make dendrogram ultrametric.*

---

**Description**

This is a simple function that sets the 'height' attributes of all leaf nodes to zero to aid visualization.

**Usage**

```
ultrametricize(x)
```

**Arguments**

x                    an object of class "dendrogram".

**Value**

Returns an object of class "dendrogram".

**Author(s)**

Shaun Wilkinson

**Examples**

```
x <- read.dendrogram(text = "(A:0.1,B:0.2,(C:0.3,D:0.4):0.5);")
plot(x, horiz = TRUE)
x <- ultrametricize(x)
plot(x, horiz = TRUE)
```

---

write.dendrogram	<i>Export a dendrogram object to text.</i>
------------------	--

---

### Description

This function writes a dendrogram object to Newick-style parenthetic text.

### Usage

```
write.dendrogram(x, file = "", append = FALSE, edges = TRUE, ...)
```

### Arguments

x	an object of class "dendrogram".
file	a character string naming a file or connection to write the output to. If no file path is specified or file = "" the result is printed to the console.
append	logical indicating whether the output should be appended to the file. If append = FALSE the contents of the file will be overwritten (the default setting).
edges	logical indicating whether edge weights should be included in the output string.
...	further arguments to be passed to format. Used to specify the numbering style of the edge weights (if edges = TRUE).

### See Also

[read.dendrogram](#) to create a "dendrogram" object from a text file. The [write.tree](#) function in the [ape](#) package performs a similar operation for "phylo" and "multiPhylo" objects.

### Examples

```
newick <- "(A:0.1,B:0.2,(C:0.3,D:0.4):0.5);"  
x <- read.dendrogram(text = newick)  
write.dendrogram(x, edges = TRUE)
```

# Index

`ape`, [4](#), [6](#), [8](#), [11](#)

`as.dendrogram.phylo`, [2](#), [5](#)

`as.phylo.dendrogram`, [3](#), [5](#)

`drop.tip`, [6](#)

`ladder`, [4](#), [5](#)

`ladderize`, [4](#)

`phylogram`, [5](#)

`phylogram-package` (phylogram), [5](#)

`prune`, [5](#), [6](#)

`read.dendrogram`, [2](#), [3](#), [5](#), [7](#), [11](#)

`read.tree`, [2](#), [3](#), [8](#)

`regex`, [6](#)

`remidpoint`, [5](#), [8](#)

`reposition`, [5](#), [9](#)

`ultrametricize`, [5](#), [10](#)

`write.dendrogram`, [5](#), [8](#), [11](#)

`write.tree`, [11](#)