Package 'visATC'

September 23, 2026

Type Package

Title Visualise the Anatomical Therapeutic Chemical (ATC) Hierarchy

Version 1.0.0

Description Visualisation and subsetting of the World Health Organisation Anatomical Therapeutic Chemical (ATC) classification system.

License GPL (>= 3)

Encoding UTF-8

LazyData true

Imports tidyr, dplyr, rlang, igraph, methods, plotly, graphlayouts

RoxygenNote 7.3.2

Suggests knitr, rmarkdown

Depends R (>= 4.1.0)

URL https://jnm212.github.io/visATC/

NeedsCompilation no

Author J Matthews [aut, cre, cph]

Maintainer J Matthews <jnmatt212@gmail.com>

Repository CRAN

Date/Publication 2026-09-23 04:00:02 UTC

Contents

`ATCdata` *nodes of the ATC hierarchy*

Description

This contains nodes of the ATC hierarchy at all 5 levels (e.g. A, A01, A01A ...), and associated descriptive text (eg. for A: ALIMENTARY TRACT AND METABOLISM), used here for visualisation.

Usage

```
ATCdata
```

Format

A named character vector for each node, where each entry is an ATC code and the name is a textual description

Source

The ATC data can be perused at <https://atcddd.fhi.no/atc_ddd_index/>.

This data is from the repository at <https://github.com/fabkury/atcd> and this release is at <https://github.com/fabkury/atcd/re

The first two columns from the .csv file were selected to produce the named character vector

---

`atctree` *constructor for atctree objects*

Description

constructor for atctree objects

Usage

```
atctree(
  whichlevs = NULL,
  schema = c("none", "full", "anatomical", "therapeutic", "chemical")
)
```

Arguments

`whichlevs`	subset of levels to use from integers 1:5
`schema`	a conceptual subset of the ATC hierarchy, or all of it ('full'), or something bespoke ('none')

Details

either schema or whichlevs should be specified; (e.g. ‘schema=full‘ corresponds to ‘whichlevs=1:5‘)

Value

an object of class atctree

Examples

```
h <- atctree(schema="anatomical")
# tree will contain levels 1 and 5
h <- atctree(whichlevs=c(1,2,5))
# bespoke structure
```

---

atctree-class *class to represent a tree from the ATC hierarchy*

Description

class to represent a tree from the ATC hierarchy

Slots

`node` node id

`pnode` parent node id

`lab` ATC code

`plab` parent ATC code

`text` ATC textual info

`lev` level of each node (root node is 0)

`schema` one of ’none’, ’full’, ’anatomical’, ’therapeutic’, ’chemical’

`whichlevs` subset of integers 1:5; should correspond to schema

`Nnode` number of nodes in the tree excluding root node

`Nlev` number of levels in the tree excluding root node

cutting,atctree-method *cutting from an ATC tree*

Description

Cut out a branch with just the descendants of the specified node.

This will include the direct parent of the specified node as information - this is also useful for grafting.

Usage

```
## S4 method for signature 'atctree'
cutting(h, nodelabel = NULL)
```

Arguments

h	an object of class atctree
nodelabel	identify node of interest

Value

returns a (smaller) object of class atctree

Examples

```
h <- atctree(schema="therapeutic")
# Suppose we are interested in a branch of this tree; take a cutting and plot it:
plot(cutting(h, "P01"))

# a different look:
plot(cutting(h, "P01"),  text_angle=0, circle=TRUE)
```

drugnames *display drug name for ATC codes*

Description

display drug name for ATC codes

Usage

```
drugnames(x)
```

Arguments

x	vector of ATC codes

Value

character vector of drug names

---

parents *show the parents of the specified node(s)*

Description

show the parents of the specified node(s)

Usage

```
parents(x)
```

Arguments

x	a node label e.g. A01AX

Value

character vector

---

plot,atctree-method *plot the tree as a network graph*

Description

plot the tree as a network graph

Usage

```
## S4 method for signature 'atctree'
plot(
  x,
  circle = NULL,
  ATC_text = NULL,
  leaf_label = NULL,
  stem_label = NULL,
  text_size = 10,
  text_angle = 45,
  width = 500,
  height = 500,
  hover_msg = TRUE
)
```

Arguments

x	an object of class atctree
circle	whether or not to arrange graph over a circle
ATC_text	whether or not to label with ATC textual info
leaf_label	whether or not leaf nodes will be labelled
stem_label	whether or not stem nodes will be labelled
text_size	annotation text size
text_angle	annotation text angle
width	graph width
height	graph height
hover_msg	whether to remind the user that they can hover over nodes

Details

Each node is labelled with its ATC code, and by hovering the cursor over any node the ATC desciption is displayed. The ATC tree can be displayed as a network in layers or on a circle (with radii corresponding to ATC level). It is evident that a plot the full ATC tree is too cluttered. Some subsetting (link), pruning (link) or cutting (link) of the tree is likely to help visualisation.

Value

No return value, called for side effects

Examples

```
# create an ATC tree and plot the whole thing:
h <- atctree(schema="full")
# you could plot this with plot(h) but it takes a while
# large network so not easy to read

# take cutting above node B ; plot all its descendants:
plot(cutting(h,"B"))

# plot siblings of node A03
plot(h["A03",0])

# plot node B down to its grandchildren:
plot(h["B",-2])
# or use a circular layout
plot(h["B",-2], circle=TRUE, stem_label=TRUE)
```

pruning,atctree-method *prune an ATC tree*

Description

Pruning is specifying the removal of some unwanted branch of the tree (in contrast to taking a cutting, where it is the branch which is kept).

Usage

```
## S4 method for signature 'atctree'
pruning(h, nodelabels = NULL)
```

Arguments

h	an object of class atctree
nodelabels	character vector of nodes of interest

Value

a (smaller) object of class atctree

Examples

```
library(visATC)
h1 <- atctree(whichlevs=1:3)
h1A <- cutting(h1, "C")
# make a subtree cut at node C
plot(h1A, circle=TRUE)
# suppose we decide not to display some nodes:
plot(pruning(h1A, c("C01", "C05", "C10")), circle=TRUE)
```

show,atctree-method *summarise atctree object*

Description

shows schema, number of nodes and levels and a cross-table, excluding the root node.

Usage

```
## S4 method for signature 'atctree'
show(object)
```

Arguments

object	of class acttree

Value

prints summary to console

Examples

```
h <- atctree(whichlevs=1:4)
(h)
```

---

treelevs,atctree-method

helper functions for atctrees

Description

treelevs: return levels within tree (excluding 0)

leaves: return labels of terminal elements (leaves) of tree

nlevs: number of levels in tree (excluding 0)

Usage

```
## S4 method for signature 'atctree'
treelevs(h)

## S4 method for signature 'atctree'
leaves(h)

## S4 method for signature 'atctree'
nlevs(h)
```

Arguments

h	an object of class atctree

Value

a vector of the level of each node

vector of labels of the leaf nodes

vector of number of levels in the tree

`[,atctree-method` *find a subset of the tree*

Description

find a subset of the tree

Usage

```
## S4 method for signature 'atctree'
x[i, j = 0]
```

Arguments

x	object of class atctree
i	label (ATC code) of focal node
j	generation (integer). 0: siblings, -1: children, -2: grandchildren, 1: parents etc

Details

The subset method for objects of class atctree uses a focal node and a specification of how many generations to accrue. The focal node is specified by an ATC code at any of the five levels (e.g. i="A01"). The generation is specified by integer j, signed positive for ancestors and negative for descendants.

Value

an object of class atctree

Examples

```
h <- atctree(schema="full")
#more useful to use the full tree with this approach
# plot the siblings of C01:
plot(h["C01",0], ATC_text=TRUE, stem_label=TRUE, leaf_label=TRUE)
# plot descendents of P01 (as far as grandchildren):
plot(h["P01",-2], stem_label=TRUE, leaf_label=TRUE)
# plot descendents of P01 (as far as great grandchildren):
plot(h["P01",-3], stem_label=TRUE, leaf_label=TRUE)
```

Index