Package 'fancycut'

June 26, 2018

Title A Fancy Version of 'base::cut'

Version 0.1.2

Description Provides the function fancycut() which is like cut() except
you can mix left open and right open intervals with point values,
intervals that are closed on both ends and intervals that are open on both ends.

License CC0

LazyData true

RoxygenNote 6.0.1

Suggests testthat

NeedsCompilation no

Author Adam Rich [aut, cre],
Richie Cotton [ctb]

Maintainer Adam Rich <adamleerich@gmail.com>

Repository CRAN

Date/Publication 2018-06-26 11:58:54 UTC

R topics documented:

`fancycut` *Like* `cut`*, turn a vector of numbers into a factor*

Description

Like `cut`, turn a vector of numbers into a factor

Usage

```
fancycut(x, na.bucket = NA, unmatched.bucket = NA, out.as.factor = TRUE,
  ...)
```

Arguments

x	a numeric vector
na.bucket	what level should NA values be given?
unmatched.bucket	what level should numbers not covered by an interval be given?
out.as.factor	default is TRUE Should the resulting vector be a factor? If FALSE will return a character vector.
...	These take the form `tag = value`. Tags become the bucket names and values the interval definitions.

Examples

```
fancycut(
  x = -10:10,
  Zero = 0,
  Small = '[0,2)',
  Medium = '[2,5]',
  Large = '(5,10]'
)

# The following examples are from Richie Cotton via
# https://www.rdocumentation.org/packages/fancycut/versions/0.1.1/topics/fancycut

# The tag = value syntax is useful.
x <- seq.int(0, 1, 0.25)
fancycut(x, low = '[0, 0.5]', high = '(0.5, 1]')

# Not all the values have to live in a bucket.
x <- seq.int(0, 1, 0.25)
fancycut(x, low = '(0.2, 0.3]', high = '(0.7, 0.8)')

# You can use unmatched.bucket to deal with these other intervals.
x <- seq.int(0, 1, 0.25)
fancycut(x, low = '(0.2, 0.3]', high = '(0.7, 0.8)', unmatched.bucket = 'other')

# To match a specific value, make the lower and upper bound the same number.
x <- seq.int(0, 1, 0.25)
fancycut(x, low = '[0, 0.5)', half = '[0.5,0.5]', high = '(0.5, 1]')

# To match NA values, use na.bucket.
x2 <- c(seq.int(0, 1, 0.25), NA)
fancycut(x2, low = '[0, 0.5)', high = '[0.5, 1]', na.bucket = 'missing')
```

wafflecut *Like* cut*, turn a vector of numbers into a factor*

Description

Like cut, turn a vector of numbers into a factor

Usage

```
wafflecut(x, intervals, buckets = intervals, na.bucket = NA,
  unmatched.bucket = NA, out.as.factor = TRUE)
```

Arguments

x	a numeric vector
intervals	a character vector of intervals
buckets	a character vector of levels for the new factor these have a 1-1 correspondence with intervals
na.bucket	what level should NA values be given?
unmatched.bucket	what level should numbers not covered by an interval be given?
out.as.factor	default is TRUE Should the resulting vector be a factor? If FALSE will return a character vector.

Examples

```
wafflecut(-10:10, c('[0,2)','[2,5)','[5,10]'), c('Small','Medium','Large'))

wafflecut(-10:10, c('[0,0]','(0,2]','(2,5)','[5,10]'), c('Zero','Small','Medium','Large'))

wafflecut(-10:10, c('[0,2)','[2,5)','[5,10]'), c('Small','Medium','Large'))

wafflecut(-10:10, c('[0,0]','[0,2]','(2,5)','[5,10]'), c('Zero','Small','Medium','Large'))


# The following examples are from Richie Cotton via
# https://www.rdocumentation.org/packages/fancycut/versions/0.1.1/topics/fancycut

# Not all the values have to live in a bucket.
x <- seq.int(0, 1, 0.25)
wafflecut(x, c('(0.2, 0.3)', '(0.7, 0.8)'), c('low', 'high'))

# You can use unmatched.bucket to deal with these other intervals.
x <- seq.int(0, 1, 0.25)
wafflecut(x, c('(0.2, 0.3)', '(0.7, 0.8)'), c('low', 'high'), unmatched.bucket = 'other')
```

```
# To match NA values, use na.bucket.
x2 <- c(seq.int(0, 1, 0.25), NA)
wafflecut(x2, c('[0, 0.5)', '[0.5, 1]'), c('low', 'high'), na.bucket = 'missing')
```

Index