Package 'hqmisc'

July 22, 2025

Version 0.2-1

Date 2022-05-09

Encoding UTF-8

Title Miscellaneous Convenience Functions and Dataset

Maintainer Hugo Quene <h.quene@uu.nl>

Depends R (>= 3.0.0), stats

Imports graphics

Description Miscellaneous convenience functions and wrapper functions
to convert frequencies between Hz, semitones, mel and Bark,
to create a matrix of dummy columns from a factor,
to determine whether x lies in range [a,b],
and to add a bracketed line to an existing plot.
This package also contains an example data set of a stratified sample
of 80 talkers of Dutch.

License GPL (>= 2)

NeedsCompilation no

Author Hugo Quene [aut, cre] (ORCID: <https://orcid.org/0000-0001-7988-1346>)

Repository CRAN

Date/Publication 2022-05-09 17:00:02 UTC

Contents

as.dummies *Create matrix of dummy vectors from single factor*

Description

Create a matrix of dummy vectors (consisting of 0 and 1 only) from factor `x`. This may be useful if you want to use dummy predictors in regression analysis.

Usage

```
as.dummies ( x )
```

Arguments

`x`	input factor. If the argument is not a factor, the function stops with a message.

Details

Levels of the input factor correspond with columns in the output matrix. The integer value in cell [i,j] of the returned matrix indicates whether (value=1) or not (value=0) the i'th observation in the input factor *x* has the value of the j'th level of that factor.

`as.dummies(x)` is similar to `model.matrix(~-1+x)`.

Value

Matrix of integer values 0 or 1. The number of rows of the returned matrix equals the length of the input factor, and the number of columns equals the number of levels of the input factor.

See Also

`levels`, `matrix`, `model.matrix`..

Examples

```
xf <- factor( rep(letters[1:3],each=2) )
as.dummies(xf)
```

bracket *Add rectangular bracket to plot*

Description

Add a rectangular bracket to the most recent plot. This bracket may be used to loosely connect points or bars in a plot, to attract the viewer's attention to these connected points or bars.

Usage

```
bracket( x0, y0, x1=x0, y1=y0,
         offset=1, length=offset/2,
         side=1, col="grey", ... )
```

Arguments

`x0, y0`	coordinates of points **from** which to draw bracket.
`x1, y1`	coordinates of points **to** which to draw bracket.
`offset`	offset or distance of bracket from specified coordinates; the direction of the offset depends on '`side`'. The offset must be specified in data units of the appropriate dimension of the plot.
`length`	the length of the whiskers of the bracket.
`side`	the side, relative to the specified coordinates, on which to draw the bracket; 1=below, 2=left, 3=above, 4=right.
`col`	the color in which to draw the bracket.
`...`	other graphical parameters to be passed on (see par), such as `lty`, `lwd`.

Details

If '`offset`' is positive (the default), then positive values of '`length`' (the default) result in inside whiskers, towards the specified coordinates, and negative values of '`length`' result in outside whiskers, away from the specified coordinates.

The value for '`side`' must be specified as an integer number; values other than 1:4 will be ignored.

Value

None; a bracket is added to the existing plot.

See Also

`segments`, `par`.

Examples

```
plot( 1:9, 1:9 )
bracket( x0=4, x1=6, y0=3, side=1 )
bracket( x0=3, y0=4, y1=6, side=2, offset=1, length=1, lty=2 )
bracket( x0=4, x1=6, y0=7, side=3, col="red" )
# negative length: whiskers outside
bracket( x0=7, y0=4, y1=6, side=4, offset=0, length=-1 )
```

---

f2bark *Convert between frequency in Hz and in Bark*

Description

Convert between frequency in Hz and in psychoacoustic Bark units

Usage

```
f2bark( hz )
bark2f( bark )
```

Arguments

hz	a frequency in Hertz (cycles per second).
bark	a frequency in Bark units.

Details

These functions convert between the linear Hertz and psychoacoustic Bark scales. The functions use Formula (6) of Traunmüller (1990).

Value

Transformed value(s) in Bark or Hertz units.

References

Traunmüller, H. (1990). Analytical expressions for the tonotopic sensory scale. *Journal of the Acoustical Society of America*, 88(1), 97-100.

See Also

f2st, st2f, f2mel, mel2f.

Examples

```
f2bark( c(1,3,5)*500 )
bark2f( 12 )
```

f2mel *Convert between frequency in Hz and in mel units*

Description

Convert between frequency in Hz and in psychoacoustic mel units.

Usage

```
f2mel( hz )
mel2f( mel )
```

Arguments

hz	a frequency in Hertz (cycles per second).
mel	a frequency in mel units.

Details

These functions convert between the linear Hertz and psychoacoustic Mel scales. The functions use formulas provided by Rietveld & Van Heuven (2009).

A frequency of 1000 Hz equals 1000 mel. For frequencies below 1000 Hz the mel scale is approximately linear; for frequencies above 1000 Hz the mel scale is approximately logarithmic.

Value

Transformed value(s) in mel or hertz units.

References

Rietveld, A. C. M., & Van Heuven, V. J. (2009). *Algemene Fonetiek* (3rd ed.). Bussum: Coutinho.

See Also

f2st, st2f, f2bark, bark2f.

Examples

```
f2mel( seq( 250, 1000, by= 250) )
f2mel( seq(1000, 3000, by=1000) )
mel2f( seq( 500, 2500, by= 500) )
```

`f2st` *Convert between frequency in Hz and in semitones*

Description

Convert between frequency in Hz and in semitones, relative to baseline frequency in Hz.

Usage

```
f2st( hz, base=50 )
st2f( st, base=50 )
```

Arguments

`hz` a frequency in Hertz (cycles per second).

`st` a frequency in semitones relative to the base frequency.

`base` the baseline frequency (in Hertz units) relative to which semitones are expressed.

Details

These functions convert between the linear Hertz and logarithmic semitone scales. A doubling of the frequency in Hz equals 12 semitones or 1 octave.

Function `f2st` was based on function `h2st` by Mark Liberman
(last seen at http://ldc.upenn.edu/myl/llog/semitones.R but no longer available online).
Function `st2f` was based on a conversion script by Jan Roelof de Pijper
(last seen at http://users.abo.fi/jtuomain/speech/semitone.html but no longer available online).

For frequencies in Hz of musical notes, see e.g. `http://www.phy.mtu.edu/~suits/notefreqs.html`.

Value

Transformed value(s) in semitones or in Hertz (cycles per second).

See Also

`f2bark`, `bark2f`, `f2mel`, `mel2f`.

Examples

```
# distance in semitones between target and baseline frequencies in Hz.
f2st( 440, base=110 )

# frequency in Hz of a tone that is +24 semitones distant
# from the baseline, i.e. a distance of +two octaves.
st2f( 24 )
```

`is.inrange` *Check whether argument is within range*

Description

Check whether argument *x* is within *range*.

Usage

```
is.inrange( x, range=c(0,1) )
```

Arguments

`x`	numerical value(s) to be evaluated.
`range`	list of two numbers defining the range used in the evaluation of *x*.

Details

The return value indicates whether *x* is an element of *[range[1], range[2]]*.

Note that the range used in the evaluation includes both endpoints of the specified range.

Value

Logical.

See Also

`range`.

Examples

```
is.inrange(1/2)
is.inrange(pi, range=c(0,pi))
```

`talkers` *Data set of talkers of Dutch from the Netherlands.*

Description

This data set gives metadata (id, sex, age, region of origin) and speech characteristics (average syllable duration, average phrase length) for a stratified sample of 80 talkers of Dutch from the Netherlands.

Usage

```
data(talkers)
```

Format

A data frame with 80 observations on the following 6 variables.

`id` identifier code (from data source, see Source)

`sex` sex (0=female, 1=male)

`age` age (in years)

`region` region of origin (a factor with levels `M`=Mid, `N`=North, `S`=South, or `W`=West)

`syldur` average duration of syllables, or seconds per syllable (in seconds, excluding pause time, 1/(articulation rate))

`nsyl` average number of syllables per phrase, or average phrase length in syllables

Details

Talkers grew up in their region of origin, and have lived and worked there as teachers of Dutch Language and Literature in secondary education. Talkers with ages between 41 and 45 were not included in this study. The sample is stratified by *sex*, *region*, and (*age*>41) (see Examples).

Speech data were collected from (and averaged over) a recorded interview lasting about 15 minutes. The talker and the interviewer only spoke Standard Dutch during the interview.

One talker (`id` 117) spoke remarkably slower than all others, yielding a very high *syldur*.

The West region is commonly regarded as the linguistic center of the Netherlands. Each of the four *region*s has a distinct variety of Dutch. The variety of the West region is closest to the Standard Dutch spoken in the Netherlands.

Speech recordings and metadata were collected in 1999.

Source

`http://tla.mpi.nl/resources/data-archive/`, Corpus of Spoken Dutch

References

Oostdijk, N. (2000). The Spoken Dutch Corpus: Overview and first evaluation. In M. Gravilidou, G. Carayannis, S. Markantonatou, S. Piperidis & G. Stainhaouer (Eds.), *Proceedings of the Second International Conference on Language Resources and Evaluation* (Vol. 2, pp. 887-894).

Adank, P., van Hout, R., & van de Velde, H. (2007). An acoustic description of the vowels of northern and southern Standard Dutch II: Regional varieties. *Journal of the Acoustical Society of America*, 121(2), 1130-1141.

Quené, H. (2008). Multilevel modeling of between-speaker and within-speaker variation in spontaneous speech tempo. *Journal of the Acoustical Society of America*, 123(2), 1104-1113.

Examples

```
data(talkers)
str(talkers)
pairs( talkers[,2:6] )
with( talkers, table( sex, region, I(age>41) ) )
```

Index