

Package ‘dscore’

December 30, 2019

Type Package

Title D-score for Child Development

Version 1.0.0

Description The dscore package provides tools for calculating the D-score, a quantitative measure of child development. The main function converts milestone scores from various instruments into the D-score and the DAZ (D-score adjusted for age). The user needs to map the variable names of each milestone into the 9-position GSED convention. The packages contains tools that assist in this translation.

Depends R (>= 3.5)

Imports dplyr (>= 0.8.2), Rcpp, stats, stringr, tidyr (>= 1.0.0)

LinkingTo Rcpp, RcppArmadillo

Suggests ggplot2, kableExtra, knitr, lme4, rmarkdown, testthat

Encoding UTF-8

License GPL-3 + file LICENSE

LazyData TRUE

VignetteBuilder knitr

RoxygenNote 7.0.2

NeedsCompilation yes

URL <https://github.com/stefvanbuuren/dscore>,
<https://stefvanbuuren.name/dscore/>,
<https://stefvanbuuren.name/dbook1/>

BugReports <https://github.com/stefvanbuuren/dscore/issues>

Roxygen list(markdown = TRUE)

Author Stef van Buuren [cre, aut],
Iris Eekhout [aut],
Arjan Huizing [aut]

Maintainer Stef van Buuren <stef.vanbuuren@tno.nl>

R topics documented:

dscore-package	2
builtin_itembank	4
builtin_itemtable	4
builtin_references	5
calculate_posterior	6
count_mu_dutch	7
count_mu_gcdg	8
daz	8
decompose_itemnames	9
dscore	10
get_age_equivalent	13
get_itemnames	14
get_itemtable	16
get_labels	17
get_reference	17
get_tau	18
milestones	19
normalize	20
posterior	20
rename_gcdg_gsed	21
sort_itemnames	22
Index	23

dscore-package	<i>dscore: D-score for Child Development</i>
----------------	--

Description

The dscore package implements several tools needed to calculate the D-score, a numerical score that measures generic development in children.

User functions

The available functions are:

Function	Description
<code>get_itemnames()</code>	Extract item names from an itemtable
<code>order_itemnames()</code>	Order item names
<code>sort_itemnames()</code>	Sort item names
<code>decompose_itemnames()</code>	Get four components from itemname
<code>get_itemtable()</code>	Get a subset from the itemtable
<code>get_labels()</code>	Get labels for items
<code>rename_gcdg_gsed()</code>	Rename gcdg into gsed lexicon
<code>dscore()</code>	Estimate D-score and DAZ
<code>dscore_posterior()</code>	Calculate full posterior of D-score
<code>get_tau()</code>	Get difficulty parameters from item bank

<code>daz()</code>	Transform to age-adjusted standardized D-score
<code>zad()</code>	Inverse of <code>daz()</code>
<code>get_reference()</code>	Get D-score age-reference

Built-in data

The package contains the following built-in data:

Data	Description
<code>builtin_itembank()</code>	A <code>data.frame</code> containing the difficulty estimates of items according to final Rasch models.
<code>builtin_itemtable()</code>	A <code>data.frame</code> containing names and descriptions of items from 22 instruments.
<code>builtin_references()</code>	A <code>data.frame</code> with LMS reference values used to transform from D-score to DAZ, DAZ to D
<code>milestones()</code>	A small demo dataset with PASS/FAIL responses from 27 preterms, measured at various ages and 2.5 years.

Note

This study was supported by the Bill & Melinda Gates Foundation. The contents are the sole responsibility of the authors and may not necessarily represent the official views of the Bill & Melinda Gates Foundation or other agencies that may have supported the primary data studies used in the present study.

The authors wish to recognize the principal investigators and their study team members for their generous contribution of the data that made this tool possible and the members of the Ki team who directly or indirectly contributed to the study: Amina Abubakar, Claudia R. Lindgren Alves, Orazio Attanasio, Maureen M. Black, Maria Caridad Araujo, Susan M. Chang-Lopez, Gary L. Darmstadt, Bernice M. Doove, Wafaie Fawzi, Lia C.H. Fernald, Günther Fink, Emanuela Galasso, Melissa Gladstone, Sally M. Grantham-McGregor, Cristina Gutierrez de Pineres, Pamela Jervis, Jena Derakhshani Hamadani, Charlotte Hanlon, Simone M. Karam, Gillian Lancaster, Betzy Lozoff, Gareth McCray, Jeffrey R Measelle, Girmay Medhin, Ana M. B. Menezes, Lauren Pisani, Helen Pitchik, Muneera Rasheed, Lisy Ratsifandrihamanana, Sarah Reynolds, Linda Richter, Marta Rubio-Codina, Norbert Schady, Limbika Sengani, Chris Sudfeld, Marcus Waldman, Susan P. Walker, Ann M. Weber and Aisha K. Yousafzai.

References

- Jacobusse, G., S. van Buuren, and P.H. Verkerk. 2006. "An Interval Scale for Development of Children Aged 0-2 Years." *Statistics in Medicine* 25 (13): 2272–83. [pdf](#)
- Van Buuren S (2014). Growth charts of human development. *Stat Methods Med Res*, 23(4), 346-368. [pdf](#)
- Weber AM, Rubio-Codina M, Walker SP, van Buuren S, Eekhout I, Grantham-McGregor S, Caridad Araujo M, Chang SM, Fernald LCH, Hamadani JD, Hanlon A, Karam SM, Lozoff B, Ratsifandrihamanana L, Richter L, Black MM (2019). The D-score: a metric for interpreting the early development of infants and toddlers across global settings. *BMJ Global Health*, *BMJ Global Health* 4: e001724. [pdf](#).
- GSED team (Maureen Black, Kieran Bromley, Vanessa Cavallera (lead author), Jorge Cuartas, Tarun Dua (corresponding author), Iris Eekhout, Gunther Fink, Melissa Gladstone, Katelyn Hepworth, Magdalena Janus, Patricia Kariger, Gillian Lancaster, Dana McCoy, Gareth McCray, Abbie Raikes, Marta Rubio-Codina, Stef van Buuren, Marcus Waldman, Susan Walker and Ann Weber). 2019. "The Global Scale for Early Development (GSED)." *Early Childhood Matters*. [link](#)

builtin_itembank	<i>Built-in itembank</i>
------------------	--------------------------

Description

A data frame with administrative information per item. Includes only items that are part of a Rasch model. See [builtin_itemtable\(\)](#) for an overview of all currently defined items.

Usage

```
builtin_itembank
```

Format

A data.frame with variables:

Name	Label
key	String indicating a specific Rasch model, either "gsed", "gcdg" or "dutch"
item	Item name, gsed lexicon
tau	Difficulty estimate
instrument	Instrument code, 3 letters
domain	Domain code, 2 letter
mode	Mode code, 1 letter
number	Item number, 3 numbers
label	Label (English)

See Also

[dscore\(\)](#), [get_tau\(\)](#), [builtin_itemtable\(\)](#)

Examples

```
head(builtin_itembank)
```

builtin_itemtable	<i>Global Scale for Early Development - itemtable</i>
-------------------	---

Description

The built-in variable named itemtable contains descriptions of all items found in the gsed data.

Usage

```
builtin_itemtable
```

Format

A data.frame with variables:

Name	Label
item	Item name, gsed lexicon
instrument	Instrument code, 3 letters
domain	Domain code, 2 letter
mode	Mode code, 1 letter
number	Item number, 3 numbers
equate	Equate group
label	Label (English)

Details

Data are collected by the members of the Global Scale for Early Development (GSED) group. The itemtable is created by \\data-raw\\R\\import_itemtable.R.

Author(s)

Compiled by Stef van Buuren

builtin_references	<i>Age-conditional reference distribution of D-score</i>
--------------------	--

Description

A data frame containing the age-dependent distribution of the D-score for children aged 0-5 years. The distribution is modelled after the LMS distribution (Cole & Green, 1992), and is equal for both boys and girls. The LMS values can be used to graph reference charts and to calculate age-conditional Z-scores, also known as DAZ.

Usage

```
builtin_references
```

Format

A data.frame with 265 rows and 17 variables:

Name	Label
pop	Population, either "dutch" or "gcdg"
age	Decimal age in years
mu	M-curve, median D-score, P50
sigma	S-curve, spread expressed as coefficient of variation
nu	L-curve, the lambda coefficient of the LMS model for skewness
P3	P3 percentile
P10	P10 percentile
P25	P25 percentile
P50	P50 percentile
P75	P75 percentile

P90	P90 percentile
P97	P97 percentile
SDM2	-2SD centile
SDM1	-1SD centile
SD0	0SD centile, median
SDP1	+1SD centile
SDP2	+2SD centile

Details

The "dutch" references were calculated from the SMOCC data, and cover age range 0-2.5 years (van Buuren, 2014). The "gcdg" references were calculated from the 15 cohorts of the GCDG-study, and cover age range 0-5 years (Weber, 2019).

References

Cole TJ, Green PJ (1992). Smoothing reference centile curves: The LMS method and penalized likelihood. *Statistics in Medicine*, 11(10), 1305-1319.

Van Buuren S (2014). Growth charts of human development. *Stat Methods Med Res*, 23(4), 346-368. [pdf](#)

Weber AM, Rubio-Codina M, Walker SP, van Buuren S, Eekhout I, Grantham-McGregor S, Caridad Araujo M, Chang SM, Fernald LCH, Hamadani JD, Hanlon A, Karam SM, Lozoff B, Ratsifandrihamanana L, Richter L, Black MM (2019). The D-score: a metric for interpreting the early development of infants and toddlers across global settings. *BMJ Global Health*, *BMJ Global Health* 4: e001724. [pdf](#).

See Also

[dscore\(\)](#)

Examples

```
head(builtin_references)
```

calculate_posterior	<i>Calculate posterior of ability</i>
---------------------	---------------------------------------

Description

Calculate posterior of ability

Usage

```
calculate_posterior(scores, tau, qp, mu, sd)
```

Arguments

scores	A vector with PASS/FAIL observations. Scores are coded numerically as pass = 1 and fail = 0.
tau	A vector containing the item difficulties for the item scores in scores estimated from the Rasch model in the preferred metric/scale.
qp	Numeric vector of equally spaced quadrature points.
mu	Numeric scalar. The mean of the prior.
sd	Numeric scalar. Standard deviation of the prior.

Value

A list with three elements:

Name	Label
eap	Mean of the posterior
gp	Vector of quadrature points
posterior	Vector with posterior distribution.

Since dscore V40.1 the function does not return the "start" element.

Author(s)

Stef van Buuren, Arjan Huizing, 2020

count_mu_dutch	<i>Median of Dutch references</i>
----------------	-----------------------------------

Description

Returns the age-interpolated median of the Dutch references. Thw working range is 0-3 years.

Usage

```
count_mu_dutch(t)
```

Arguments

t	Decimal age, numeric vector
---	-----------------------------

Value

A vector of length `length(t)` with the median of the dutch references.

Note

Internal function. Called by `dscore()`

Examples

```
dscore:::count_mu_dutch(0:2)
```

count_mu_gcdg	<i>Median of gcdg references</i>
---------------	----------------------------------

Description

Returns the age-interpolated median of the GCDG references (Weber et al, 2019). The working range is 0-4 years.

Usage

```
count_mu_gcdg(t)
```

Arguments

t	Decimal age, numeric vector
---	-----------------------------

Value

A vector of length `length(t)` with the median of the dutch references.

Note

Internal function. Called by `dscore()`

Examples

```
dscore:::count_mu_gcdg(0:2)
```

daz	<i>D-score standard deviation score: DAZ</i>
-----	--

Description

The `daz()` function calculated the "Development for Age Z-score". The DAZ represents a child's D-score after adjusting for age by an external age-conditional reference. The `zad()` is the inverse of `daz()`: Given age and the Z-score, it finds the raw D-score.

Usage

```
daz(d, x = as.numeric(names(d)), reference = get_reference(), dec = 3)
```

```
zad(z, x = as.numeric(names(z)), reference = get_reference(), dec = 2)
```

Arguments

d	Vector of D-scores
x	Vector of ages (decimal age)
reference	A <code>data.frame</code> with the LMS reference values. The default uses the <code>get_reference()</code> function. This selects a subset of rows from the <code>builtin_references</code> using its default <code>pop</code> argument.
dec	The number of decimals (default <code>dec = 3</code>).
z	Vector of standard deviation scores (DAZ)

Value

The `daz()` function return a named vector with Z-scores with `length(d)` elements

The `zad()` function returns a vector with D-scores with `length(z)` elements.

Author(s)

Stef van Buuren 2020

References

Cole TJ, Green PJ (1992). Smoothing reference centile curves: The LMS method and penalized likelihood. *Statistics in Medicine*, 11(10), 1305-1319.

See Also

[dscore\(\)](#)

Examples

```
# using gcdg-reference
daz(d = c(35, 50), x = c(0.5, 1.0))

# using Dutch reference
daz(d = c(35, 50), x = c(0.5, 1.0), reference = get_reference("dutch"))
# population median at ages 0.5, 1 and 2 years, gcdg reference
zad(z = rep(0, 3), x = c(0.5, 1, 2))

# population median at ages 0.5, 1 and 2 years, dutch reference
zad(z = rep(0, 3), x = c(0.5, 1, 2), reference = get_reference("dutch"))

# percentiles of D-score reference
g <- expand.grid(age = seq(0.1, 2, 0.1), p = c(0.1, 0.5, 0.9))
d <- zad(z = qnorm(g$p), x = g$age)
matplot(
  x = matrix(g$age, ncol = 3), y = matrix(d, ncol = 3), type = "l",
  lty = 1, col = "blue", xlab = "Age (years)", ylab = "D-score"
)
```

decompose_itemnames	<i>Decomposes item names into their four components</i>
---------------------	---

Description

This utility function decomposes item names into components: instrument, domain, mode and number

Usage

```
decompose_itemnames(x)
```

Arguments

`x` A character vector containing item names (gcdg lexicon)

Details

The gsed-naming convention is as follows. Position 1-3 codes the instrument, position 4-5 codes the domain, position 6 codes direct/caregiver/message, positions 7-9 is a item sequence number.

Value

A data.frame with length(x) rows and four columns, named: instrument, domain, mode, and number.

Author(s)

Stef van Buuren

References

https://docs.google.com/spreadsheets/d/1zLsSW9CzqshL8ubb7K5R9987jF4YGDVAW_NBw1hR2aQ/edit#gid=0

See Also

[sort_itemnames\(\)](#)

Examples

```
itemnames <- c("aqigmc028", "grihsd219", "", "by1mdd157", "mdsgmd006")
decompose_itemnames(itemnames)
```

dscore	<i>D-score estimation</i>
--------	---------------------------

Description

The function dscore() function estimates the D-score, a numeric score that measures child development, from PASS/FAIL observations on milestones.

Usage

```
dscore(
  data,
  items = names(data),
  xname = "age",
  xunit = c("decimal", "days", "months"),
  key = "gsed",
  itembank = dscore::builtin_itembank,
  metric = c("dscore", "logit"),
  prior_mean = ".gcdg",
  prior_sd = NULL,
  transform = NULL,
  qp = -10:100,
  population = key,
  dec = 3L
)
```

```

dscore_posterior(
  data,
  items = names(data),
  xname = "age",
  xunit = c("decimal", "days", "months"),
  key = "gsed",
  itembank = dscore::builtin_itembank,
  metric = c("dscore", "logit"),
  prior_mean = ".gcdg",
  prior_sd = NULL,
  transform = NULL,
  qp = -10:100,
  population = key,
  dec = 3L
)

```

Arguments

<code>data</code>	A <code>data.frame</code> with the data. A row collects all observations made on a child on a set of milestones administered at a given age. The function calculates a D-score for each row. Different rows correspond to different children or different ages.
<code>items</code>	A character vector containing names of items to be included into the D-score calculation. Milestone scores are coded numerically as 1 (pass) and 0 (fail). By default, D-score calculation is done on all items found in the data that have a difficulty parameter under the specified key.
<code>xname</code>	A string with the name of the age variable in data. The default is "age".
<code>xunit</code>	A string specifying the unit in which age is measured (either "decimal", "days" or "months"). The default ("decimal") means decimal age in years.
<code>key</code>	A string that sets the key, the set of difficulty estimates from a fitted Rasch model. The built-in keys are: "gsed" (default), "gcdg", and "dutch". Use <code>key = ""</code> to use all item names, which should only be done if there are no duplicate itemnames.
<code>itembank</code>	A <code>data.frame</code> with columns <code>key</code> , <code>item</code> , <code>tau</code> , <code>instrument</code> , <code>domain</code> , <code>mode</code> , <code>number</code> and <code>label</code> . Only columns <code>item</code> and <code>tau</code> are required. The function uses <code>dscore::builtin_itembank</code> by default.
<code>metric</code>	A string, either "dscore" (default) or "logit", signalling the metric in which ability is estimated.
<code>prior_mean</code>	A string specifying a column name in data with the mean of the prior for the D-score calculation. The default <code>prior_mean = ".gcdg"</code> calculates an age-dependent prior mean internally according to function <code>dscore::count_mu_gcdg()</code> . The choice <code>prior_mean = ".dutch"</code> calculates <code>prior_mean</code> from the Count model coded in <code>dscore::count_mu_dutch()</code> .
<code>prior_sd</code>	A string specifying a column name in data with the standard deviation of the prior for the D-score calculation. If not specified, the standard deviation is taken as 5.
<code>transform</code>	Vector of length 2, signalling the intercept and slope respectively of the linear transform that converts an observation in the logit scale to the the D-score scale. Only needed if <code>metric == "logit"</code> .

qp	Numeric vector of equally spaced quadrature points. This vector should span the range of all D-score values. The default (qp = -10:100) is suitable for age range 0-4 years.
population	A string describing the population. Currently supported are "dutch" and "gcdg" (default).
dec	Integer specifying the number of decimals for rounding the ability estimates and the DAZ. The default is dec = 3.

Details

The algorithm is based on the method by Bock and Mislevy (1982). The method uses Bayes rule to update a prior ability into a posterior ability.

The item names should correspond to the "gsed" lexicon.

The built-in itembank (object `builtin_itembank()`) supports keys "gsed" (default), "gcdg" and "dutch". A key is defined by the set of estimated item difficulties.

Key	Model	Quadrature	Instruments	Direct/Caregiver	Reference
"dutch"	75_0	-10:80	1	direct	Van Buuren, 2014/2020
"gcdg"	565_18	-10:100	14	direct	Weber, 2019
"gsed"	807_17	-10:100	20	mixed	GSED Team, 2019

As a general rule, one should only compare D-scores that are calculated using the same key and the same set of quadrature points. For calculating D-scores on new data, the advice is to use the most general key, "gsed".

The default starting prior is a mean calculated from a so-called "Count model" that describes mean D-score as a function of age. The Count models are stored as internal functions `dscore:::count_mu_gcdg()` (default) and `dscore:::count_mu_dutch()`. The spread of the starting prior is 5 D-score points around this mean D-score, which corresponds to approximately twice the normal spread of child of a given age. The starting prior is thus somewhat informative for low numbers of valid items, and uninformative for large number of items (say >10 items).

Value

The `dscore()` function returns a `data.frame` with `nrow(data)` rows and the following columns:

Name	Label
a	Decimal age
n	Number of items with valid (0/1) data
p	Percentage of passed milestones
d	Ability estimate, mean of posterior
sem	Standard error of measurement, standard deviation of the posterior
daz	D-score corrected for age, calculated in Z-scale

The `dscore_posterior()` function returns a numeric matrix with `nrow(data)` rows and `length(qp)` columns with the density at each quadrature point. The vector represents the full posterior ability distribution. If no valid responses were obtained, `dscore_posterior()` returns the prior.

Author(s)

Stef van Buuren, Iris Eekhout, Arjan Huizing (2020)

References

Bock DD, Mislevy RJ (1982). Adaptive EAP Estimation of Ability in a Microcomputer Environment. *Applied Psychological Measurement*, 6(4), 431-444.

Van Buuren S (2014). Growth charts of human development. *Stat Methods Med Res*, 23(4), 346-368. [pdf](#)

Van Buuren S, Dusseldorp E, Doove B (2017). D-scores and references for ages 2-4 years. The Netherlands. In preparation, 2017-11-27.

Weber AM, Rubio-Codina M, Walker SP, van Buuren S, Eekhout I, Grantham-McGregor S, Caridad Araujo M, Chang SM, Fernald LCH, Hamadani JD, Hanlon A, Karam SM, Lozoff B, Ratsifandrihamanana L, Richter L, Black MM (2019). The D-score: a metric for interpreting the early development of infants and toddlers across global settings. *BMJ Global Health*, BMJ Global Health 4: e001724. [pdf](#).

See Also

[get_tau\(\)](#), [builtin_itembank\(\)](#), [posterior\(\)](#), [builtin_references\(\)](#)

Examples

```
data <- data.frame(
  age = rep(round(21 / 365.25, 4), 10),
  ddifmd001 = c(NA, NA, 0, 0, 0, 1, 0, 1, 1, 1),
  ddicmm029 = c(NA, NA, NA, 0, 1, 0, 1, 0, 1, 1),
  ddigmd053 = c(NA, 0, 0, 1, 0, 0, 1, 1, 0, 1)
)
items <- names(data)[2:4]

# third item is not part of default key
get_tau(items)

# calculate D-score
dscore(data)

# calculate full posterior
p <- dscore_posterior(data)

# plot posterior for row 7
plot(x = -10:100, y = p[7, ], type = "l", xlab = "D-score",
     ylab = "Density", xlim = c(0, 30))
```

get_age_equivalent

Get age equivalents of items that have a difficulty estimate

Description

This function calculates the ages at which a certain percent in the reference population passes the items.

Usage

```
get_age_equivalent(
  items,
  pct = c(10, 50, 90),
  key = "gsed",
  itembank = dscore::builtin_itembank,
  population = key,
  xunit = c("decimal", "days", "months"),
  dec = 3L
)
```

Arguments

items	A character vector containing names of items to be included into the D-score calculation. Milestone scores are coded numerically as 1 (pass) and 0 (fail). By default, D-score calculation is done on all items found in the data that have a difficulty parameter under the specified key.
pct	Numeric vector with requested percentiles (0-100). The default is <code>pct = c(10, 50, 90)</code> .
key	A string that sets the key, the set of difficulty estimates from a fitted Rasch model. The built-in keys are: "gsed" (default), "gcdg", and "dutch".
itembank	A <code>data.frame</code> with columns named <code>key</code> , <code>item</code> and <code>tau</code> . The function uses <code>dscore::builtin_itembank</code> by default.
population	A string describing the population. Currently supported are "dutch" and "gcdg" (default).
xunit	A string specifying the unit in which age is measured (either "decimal", "days" or "months"). The default ("decimal") means decimal age in years.
dec	Integer specifying the number of decimals for rounding the ability estimates and the DAZ. The default is <code>dec = 3</code> .

Value

Tibble with four columns: `item`, `d` (*D*-score), `pct` (percentile), and `a` (age-equivalent, in `xunit` units).

Examples

```
get_age_equivalent(c("ddicmm030", "ddicmm050"), key = "dutch")
```

get_itemnames	<i>Extract item names</i>
---------------	---------------------------

Description

The `get_itemnames()` function matches names against the 9-code template. This is useful for quickly selecting names of items from a larger set of names.

Usage

```
get_itemnames(
  x,
  instrument = NULL,
  domain = NULL,
  mode = NULL,
  number = NULL,
  strict = FALSE,
  itemtable = NULL
)
```

Arguments

x	A character vector, data.frame or an object of class lean. If not specified, the function will return all item names in itemtable.
instrument	A character vector with 3-position codes of instruments that should match. The default instrument = NULL allows for all instruments.
domain	A character vector with 2-position codes of domains that should match. The default instrument = NULL allows for all domains.
mode	A character vector with 1-position codes of the mode of administration. The default mode = NULL allows for all modes.
number	A numeric or character vector with item numbers. The default number = NULL allows for all numbers.
strict	A logical specifying whether the resulting item names must conform to one of the built-in names. The default is strict = FALSE.
itemtable	A data.frame set up according to the same structure as builtin_itemtable() . If not specified, the builtin_itemtable is used.

Details

The gsed-naming convention is as follows. Position 1-3 codes the instrument, position 4-5 codes the domain, position 6 codes direct/caregiver/message, positions 7-9 is a item sequence number.

Value

A vector with names of items

Author(s)

Stef van Buuren 2020

See Also

[sort_itemnames\(\)](#)

Examples

```
itemnames <- c("aqigmc028", "grihsd219", "", "age", "mdsgmd999")

# filter out impossible names
get_itemnames(itemnames)
get_itemnames(itemnames, strict = TRUE)
```

```
# only items from specific instruments
get_itemnames(itemnames, instrument = c("aqi", "mds"))
get_itemnames(itemnames, instrument = c("aqi", "mds"), strict = TRUE)

# get all items from the se domain of iyo instrument
get_itemnames(domain = "se", instrument = "iyo")

# get all item from the se domain with direct assessment mode
get_itemnames(domain = "se", mode = "d")

# get all item numbers 70 and 73 from gm domain
get_itemnames(number = c(70, 73), domain = "gm")
```

get_itemtable

Get a subset of items from the itemtable

Description

The `builtin_itemtable` object in the `dscore` package contains basic meta-information about items: a name, the equate group, the instrument, the domain and the item label. The `get_itemtable()` function returns a subset of items in the itemtable.

Usage

```
get_itemtable(items = NULL, itemtable = NULL)
```

Arguments

<code>items</code>	A logical or character vector of item names to return. The default (NULL) returns all items.
<code>itemtable</code>	A data.frame set up according to the same structure as <code>builtin_itemtable()</code> . If not specified, the <code>builtin_itemtable</code> is used.

Value

A data.frame with seven columns.

See Also

[get_labels\(\)](#), [get_itemnames\(\)](#)

get_labels	<i>Get labels for items</i>
------------	-----------------------------

Description

The `get_labels()` function obtains the item labels for a specified set of items.

Usage

```
get_labels(items = NULL, trim = NULL, itemtable = NULL)
```

Arguments

<code>items</code>	A character vector of item names to return. The default (NULL) returns the labels of all items.
<code>trim</code>	The maximum number of characters in the label. The default <code>trim = NULL</code> does not trim labels.
<code>itemtable</code>	A data.frame set up according to the same structure as <code>builtin_itemtable()</code> . If not specified, the <code>builtin_itemtable</code> is used.

Value

A named character vector with item labels.

See Also

`builtin_itemtable()`, `get_itemnames()`

Examples

```
# get labels of first two Macarthur items
get_labels(get_itemnames(instrument = "mac", number = 1:2), trim = 40)
```

get_reference	<i>Get D-score reference</i>
---------------	------------------------------

Description

The `get_reference()` function selects the D-score reference distribution.

Usage

```
get_reference(population = "gcdg", references = dscore::builtin_references)
```

Arguments

<code>population</code>	A string describing the population. Currently supported are "dutch" and "gcdg" (default).
<code>references</code>	A data.frame with the same structure as <code>builtin_references</code> . The default is to use <code>builtin_references</code> .

Value

A `data.frame` with the LMS reference values.

Note

No references for population "gsed" exist. The function will silently rewrite `population = "gsed"` into `population = "gsed"`.

The "dutch" reference was published in Van Buuren (2014)

The "gcdg" was calculated from 15 cohorts with direct observations (Weber, 2019).

References

Van Buuren S (2014). Growth charts of human development. *Stat Methods Med Res*, 23(4), 346-368.

Weber AM, Rubio-Codina M, Walker SP, van Buuren S, Eekhout I, Grantham-McGregor S, Caridad Araujo M, Chang SM, Fernald LCH, Hamadani JD, Hanlon A, Karam SM, Lozoff B, Ratsifandrihamanana L, Richter L, Black MM (2019). The D-score: a metric for interpreting the early development of infants and toddlers across global settings. *BMJ Global Health*, *BMJ Global Health* 4: e001724. <https://gh.bmj.com/content/bmjgh/4/6/e001724.full.pdf>.

See Also

`builtin_references()`

get_tau

Obtain difficulty parameters from item bank

Description

Searches the item bank for matching items, and returns the difficulty estimates. Matching is done by item name. Comparisons are done in lower case.

Usage

```
get_tau(items, key = "gsed", itembank = dscore::builtin_itembank)
```

Arguments

items A character vector containing names of items to be included into the D-score calculation. Milestone scores are coded numerically as 1 (pass) and 0 (fail). By default, D-score calculation is done on all items found in the data that have a difficulty parameter under the specified key.

key A string that sets the key, the set of difficulty estimates from a fitted Rasch model. The built-in keys are: "gsed" (default), "gcdg", and "dutch". Use `key = ""` to use all item names, which should only be done if there are no duplicate itemnames.

itembank A `data.frame` with columns `key`, `item`, `tau`, `instrument`, `domain`, `mode`, `number` and `label`. Only columns `item` and `tau` are required. The function uses `dscore::builtin_itembank` by default.

Value

A named vector with the difficulty estimate per item with `length(items)` elements.

Author(s)

Stef van Buuren 2020

See Also

[builtin_itembank\(\)](#), [dscore\(\)](#)

Examples

```
# difficulty levels in the GHAP lexicon
get_tau(items = c("ddifmd001", "DDigmd052", "xyz"))
```

milestones	<i>Outcomes on developmental milestones for preterm-born children</i>
------------	---

Description

A demo dataset with developmental scores at the item level for a set of 27 preterm children.

Usage

```
milestones
```

Format

A `data.frame` with 100 rows and 62 variables:

Name	Label
<code>id</code>	Integer, child ID
<code>agedays</code>	Integer, age in days
<code>age</code>	Numeric, decimal age in years
<code>sex</code>	Character, "male", "female"
<code>gagebrth</code>	Integer, gestational age in days
<code>ddifmd001</code>	Integer, Fixates eyes: 1 = yes, 0 = no
...	and so on..

See Also

[dscore\(\)](#)

Examples

```
head(milestones)
```

normalize	<i>Normalize distribution</i>
-----------	-------------------------------

Description

Normalizes the distribution so that the total mass equals 1.

Usage

```
normalize(d, qp)
```

Arguments

d	A vector with <code>length(qp)</code> elements representing the unscaled density at each quadrature point.
qp	Vector of equally spaced quadrature points.

Value

A vector of `length(d)` elements with the prior density estimate at each quadrature point.

Note

: Internal function

Examples

```
dscore:::normalize(c(5, 10, 5), qp = c(0, 1, 2))

sum(dscore:::normalize(rnorm(5), qp = 1:5))
```

posterior	<i>Calculate posterior for one item given score, difficulty and prior</i>
-----------	---

Description

Calculate posterior for one item given score, difficulty and prior

Usage

```
posterior(score, tau, prior, qp)
```

Arguments

score	Integer, either 0 (fail) and 1 (pass)
tau	Numeric, difficulty parameter
prior	Vector of prior values on quadrature points qp
qp	vector of equally spaced quadrature points

Details

This function assumes that the difficulties have been estimated by a binary Rasch model (e.g. by `sirt::rasch.pairwise.itemcluster()`).

Value

A vector of length `length(prior)`

Note

: Internal function

Author(s)

Stef van Buuren, Arjan Huizing, 2020

See Also

`dscore()`, `sirt::rasch.pairwise.itemcluster()`

rename_gcdg_gsed	<i>Rename items from gcdg into gsed lexicon</i>
------------------	---

Description

Function `rename_gcdg_gsed()` translates item names in the gcdg lexicon to item names in the gsed lexicon.

Usage

```
rename_gcdg_gsed(x, copy = TRUE)
```

Arguments

x	A character vector containing item names in the gcdg lexicon
copy	A logical indicating whether any unmatched names should be copied (<code>copy = TRUE</code>) or set to an empty string.

Details

The gsed-naming convention is as follows. Position 1-3 codes the instrument, position 4-5 codes the domain, position 6 codes direct/caregiver/message, positions 7-9 is a item sequence number.

The function currently support ASQ-I (aqi), Barrera-Moncade (bar), Batelle (bat), Bayley I (by1), Bayley II (by2), Bayley III (by3), Dutch Development Instrument (ddi), Denver (den), Griffith (gri), MacArthur (mac), WHO milestones (mds), Mullen (mul), pegboard (peg), South African Griffith (sgr), Stanford Binet (sbi), Tepsí (tep), Vineland (vin).

In cases where the domain of the items isn't clear (vin, bar), the domain is coded as 'xx'.

Value

A character vector of length `length(x)` with gcdg item names replaced by gsed item name.

Author(s)

Iris Eekhout, Stef van Buuren

References

https://docs.google.com/spreadsheets/d/1zLsSW9CzqshL8ubb7K5R9987jF4YGDVAW_NBw1hR2aQ/edit#gid=0

Examples

```
from <- c(
  "ag28", "gh2_19", "a14ps4", "b1m157", "mil6",
  "bm19", "a16fm4", "n22", "ag9", "gh6_5"
)
to <- rename_gcdg_gsed(from, copy = FALSE)
to
```

sort_itemnames

Sorts item names according to user-specified priority

Description

This function sorts the item names according to instrument, domain, mode and number. The user can specify the sorting order.

Usage

```
sort_itemnames(x, order = "idnm")

order_itemnames(x, order = "idnm")
```

Arguments

x	A character vector containing item names (gsed lexicon)
order	A four-letter string specifying the sorting order. The four letters are: i for instrument, d for domain, m for mode and n for number. The default is "idnm".

Value

sort_itemnames() return a character vector with length(x) sorted elements. order_itemnames() return an integer vector of length length(x) with positions of the sorted elements.

Author(s)

Stef van Buuren

See Also

[decompose_itemnames\(\)](#)

Examples

```
itemnames <- c("aqigmc028", "grihsd219", "", "by1mdd157", "mdsgmd006")
decompose_itemnames(itemnames)
```

Index

*Topic **datasets**

- builtin_itembank, [4](#)
- builtin_itemtable, [4](#)
- builtin_references, [5](#)
- milestones, [19](#)

- builtin_itembank, [4](#)
- builtin_itembank(), [3](#), [12](#), [13](#), [19](#)
- builtin_itemtable, [4](#)
- builtin_itemtable(), [3](#), [4](#), [15–17](#)
- builtin_references, [5](#)
- builtin_references(), [3](#), [13](#), [18](#)

- calculate_posterior, [6](#)
- count_mu_dutch, [7](#)
- count_mu_gcdg, [8](#)

- daz, [8](#)
- daz(), [3](#)
- decompose_itemnames, [9](#)
- decompose_itemnames(), [2](#), [22](#)
- dscore, [10](#)
- dscore(), [2](#), [4](#), [6](#), [9](#), [19](#), [21](#)
- dscore-package, [2](#)
- dscore_posterior (dscore), [10](#)
- dscore_posterior(), [2](#)

- get_age_equivalent, [13](#)
- get_itemnames, [14](#)
- get_itemnames(), [2](#), [16](#), [17](#)
- get_itemtable, [16](#)
- get_itemtable(), [2](#)
- get_labels, [17](#)
- get_labels(), [2](#), [16](#)
- get_reference, [17](#)
- get_reference(), [3](#)
- get_tau, [18](#)
- get_tau(), [2](#), [4](#), [13](#)

- milestones, [19](#)
- milestones(), [3](#)

- normalize, [20](#)

- order_itemnames (sort_itemnames), [22](#)

- order_itemnames(), [2](#)

- posterior, [20](#)
- posterior(), [13](#)

- rename_gcdg_gsed, [21](#)
- rename_gcdg_gsed(), [2](#)

- sirt::rasch.pairwise.itemcluster(), [21](#)
- sort_itemnames, [22](#)
- sort_itemnames(), [2](#), [10](#), [15](#)

- zad (daz), [8](#)
- zad(), [3](#)