Package 'dscore'

January 20, 2023

Type Package

Title D-Score for Child Development

Version 1.8.0

Description The D-score is a quantitative measure of child development.
The D-score follows the Rasch model. See Jacobusse, van Buuren and
Verkerk (2006) <doi:10.1002/sim.2351>. The user can convert
milestone scores from many assessment instruments into the D-score
and the DAZ (D-score adjusted for age). Several tools assist in
mapping milestone names into the 9-position Global Scale of Early
Development (GSED) convention. Supports calculation of the D-score
using 'dutch' <doi:10.1177/0962280212473300>,
'gcdg' <doi:10.1136/bmjgh-2019-001724> and 'gsed' conversion keys.
The user can calculate DAZ using 'phase1' (default), 'gcdg' and 'dutch'
age-conditional references.

Depends R (>= 3.5)

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

LinkingTo Rcpp, RcppArmadillo

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

Encoding UTF-8

License AGPL-3

LazyData TRUE

VignetteBuilder knitr

NeedsCompilation yes

URL https://github.com/d-score/dscore, https://d-score.org/dscore/,
https://d-score.org/dbook1/

BugReports https://github.com/d-score/dscore/issues

Copyright Stef van Buuren, Iris Eekhout, Arjan Huizing

RoxygenNote 7.2.3

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

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

Repository CRAN

Date/Publication 2023-01-20 13:00:02 UTC

R **topics documented:**

---

dscore-package   *dscore: D-score for Child Development*

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
get_itemnames()	Extract item names from an itemtable
order_itemnames()	Order item names
sort_itemnames()	Sort item names
decompose_itemnames()	Get four components from itemname
get_itemtable()	Get a subset from the itemtable
get_labels()	Get labels for items
rename_gcdg_gsed()	Rename gcdg into gsed lexicon
dscore()	Estimate D-score and DAZ
dscore_posterior()	Calculate full posterior of D-score
get_tau()	Get difficulty parameters from item bank
daz()	Transform to age-adjusted standardized D-score
zad()	Inverse of daz()
get_reference()	Get D-score age-reference
get_age_equivalent()	Translate difficulty to age

Built-in data

The package contains the following built-in data:

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

Note

zoff, 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. `https://stefvanbuuren.name/publication/jacobusse-2006/`

Van Buuren S (2014). Growth charts of human development. Stat Methods Med Res, 23(4), 346-368. `https://stefvanbuuren.name/publication/van-buuren-2014-gc/`

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`.

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*. `https://earlychildhoodmatters.online/2019/the-global-scale-for-early-development-gsed/`

---

builtin_itembank *Built-in itembank*

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 (the key)
item	Item name, gsed lexicon
tau	Difficulty estimate
label	Label (English)
instrument	Instrument code
domain	Domain code

mode	Administration mode
number	Item number

Details

In general, one can only compare D-score calculated with the same key. The current recommendation for new projects is to choose key gsed2212.

Note

Last update:

- Dec 01, 2022 - Overwrite labels of gto by correct item order.
- Dec 05, 2022 - Adds key gsed2212, adding instruments gl1 and gs1, and defining correct order for gto
- Jan 05, 2023 - Adds instrument gh1 to key gsed2212

See Also

dscore(), get_tau(), builtin_itemtable()

Examples

```
head(builtin_itembank)
```

---

builtin_itemtable   *Global Scale for Early Development - itemtable*

Description

The built-in variable named builtin_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
equate	Equate group
label	Label (English)

Details

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

Last update:

- May 30, 2022 - added gto (LF) and gpa (SF) items
- June 1, 2022 - added seven gsd items
- Nov 24, 2022 - Added instruments gs1, gs2
- Dec 01, 2022 - Labels of gto replaced by correct order. This change invalidates any analyses done on LF done after May 30, 2022 !!!
- Dec 05, 2022 - Redefines gs1 and instrument for Phase 2, removes gs2 (139) Adds gl1 (Long Form Phase 2 items 155)
- Jan 05, 2023 - Adds 55 items from GSED-HH

Author(s)

Compiled by Stef van Buuren

---

builtin_references   *Age-conditional reference distribution of D-score*

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) or BCT model (Stasinopoulos & Rigby, 2022) and is equal for both boys and girls. The LMS/BCT 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", "gcdg" or "phase1"
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/BCT model for skewness
tau	Kurtosis parameter in the BCT model
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). The "phase1" references were calculated from the GSED Phase 1 validation data (GSED-BGD, GSED-PAK, GSED-TZA) cover age range 2w-3.5 years. The age range 3.5-5 yrs is linearly extrapolated and are only indicative.

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. https://stefvanbuuren.name/publication/van-buuren-2014-gc/

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

Stasinopoulos M, Rigby R (2022). gamlss.dist: Distributions for Generalized Additive Models for Location Scale and Shape, R package version 6.0-3, https://CRAN.R-project.org/package=gamlss.dist

See Also

dscore()

Examples

```
head(builtin_references)
```

---

calculate_posterior   *Calculate posterior of ability*

Description

If the tauj is not within the range rello - relhi from the dynamic EAP, the procedure ignores the score of item j.

Usage

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

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.
relhi	Positive numeric scalar. Upper end of the relevance interval
rello	Negative numeric scalar. Lower end of the relevance interval

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   *Median of Dutch references*

Description

Returns the age-interpolated median of the Dutch references (van Buuren 2014). The working range is 0-3 years. This function should be called when the key = "dutch".

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   *Median of GCDG references*

Description

Returns the age-interpolated median of the GCDG references (Weber et al, 2019). The working range is 0-4 years. This function should be called when the key = "gsed" or key = "gcdg".

Usage

```
count_mu_gcdg(t)
```

Arguments

t   Decimal age, numeric vector

Value

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

Note

Internal function. Called by dscore()

Examples

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

---

count_mu_phase1 *Median of phase1 references*

Description

Returns the age-interpolated median of the phase1 references based on LF & SF in GSED-BGD, GSED-PAK, GSED-TZA.

Usage

```
count_mu_phase1(t)
```

Arguments

t   Decimal age, numeric vector

Details

The interpolation is done in two rounds. First round: Calculate D-scores using .gcdg prior-mean, calculate reference, estimate round 1 parameters used in this function. Round 2: Calculate D-score using round 1 estimates as the prior mean (most differences are within 0.1 D-score points), recalculate references, estimate round 2 parameters used in this function.

Round 1: Count model: <= 9MN: 21.3449 + 26.4916 t + 7.0251(t + 0.2) Count model: > 9Mn & <= 3.5 YR: 14.69947 - 12.18636 t + 69.11675(t + 0.92) Linear model: > 3.5 YRS: 61.40956 + 3.80904 t

Round 2: Count model: < 9MND: 20.5883 + 27.3376 t + 6.4254(t + 0.2) Count model: > 9MND & < 3.5 YR: 14.63748 - 12.11774 t + 69.05463(t + 0.92) Linear model: > 3.5 YRS: 61.37967 + 3.83513 t

The working range is 0-3.5 years. After the age of 3.5 years, the function will increase at an arbitrary rate of 3.8 D-score points per year. This function is intended to be called when key = "gsed2212", key = "gsed2208" or key = "293_0".

Value

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

Note

Internal function. Called by dscore()

Author(s)

Stef van Buuren, on behalf of GSED project

Examples

```
dscore:::count_mu_phase1(0:5)
```

---

daz   *D-score standard deviation score: DAZ*

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 `data.frame` with the LMS reference values. The default uses the `get_reference()` function. This selects a subset of rows from the `builtin_references` using its default pop argument.
dec	The number of decimals (default dec = 3).
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   *Decomposes item names into their four components*

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/](https://docs.google.com/spreadsheets/d/1zLsSW9CzqshL8ubb7K5R9987jF4YGDVAW_NBw1hR2aQ/edit#gid=0)
[edit#gid=0](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	*D-score estimation*

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 = NULL,
  itembank = dscore::builtin_itembank,
  metric = c("dscore", "logit"),
  prior_mean = NULL,
  prior_sd = NULL,
  transform = NULL,
  qp = -10:100,
  population = NULL,
  dec = c(2L, 3L),
  relevance = c(-Inf, Inf)
)

dscore_posterior(
  data,
  items = names(data),
  xname = "age",
  xunit = c("decimal", "days", "months"),
  key = NULL,
  itembank = dscore::builtin_itembank,
```

```
  metric = c("dscore", "logit"),
  prior_mean = NULL,
  prior_sd = NULL,
  transform = NULL,
  qp = -10:100,
  population = NULL,
  dec = c(2L, 3L),
  relevance = c(-Inf, Inf)
)
```

Arguments

data	A data.frame 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.
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.
xname	A string with the name of the age variable in data. The default is "age".
xunit	A string specifying the unit in which age is measured (either "decimal", "days" or "months"). The default ("decimal") means decimal age in years.
key	A string that selects a subset in the itembank that makes up the key, the set of difficulty estimates from a fitted Rasch model. The built-in keys are: "gsed2212" (default), "gsed2208" (deprecated), "gsed2206" (deprecated), "gsed1912", "lf2206", "sf2206", "gcdg", and "dutch". Since version 1.5.0, the key = "gsed" selects the latest key starting with the string "gsed". Use key = "" to use all item names, which should only be done if there are no duplicate itemnames in the itembank.
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.
metric	A string, either "dscore" (default) or "logit", signalling the metric in which ability is estimated.
prior_mean	A string specifying where the mean of the prior for the D-score calculation should come from. It could be a column name in data (when you want your own prior for every row), but normally this is one of the keywords ".dutch", ".gcdg" or ".phase1". The default depends on the key. If key == "dutch" then prior_mean = ".dutch". The choice prior_mean = ".dutch" calculates prior_mean from the Count model coded in dscore:::count_mu_dutch(). If key is #' "gcdg", "gsed1912", "gsed2206", "lf2206" or "sf2206" then prior_mean = ".gcdg". This setting calculates an age-dependent prior mean internally according to function dscore:::count_mu_gcdg(). In other cases, prior_mean = ".phase1" which uses the function dscore:::count_mu_phase1(). Normally, you should not touch this parameter, but feel free to use prior_mean to override the automatic choices.

prior_sd   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 for every row.

transform   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 metric == "logit".

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 "phase1" (default), "dutch", "gcdg".

dec   A vector of two integers specifying the number of decimals for rounding the D-score and DAZ, respectively. The default is dec = c(2L, 3L).

relevance   A numeric vector of length with the lower and upper bounds of the relevance interval. The procedure calculates a dynamic EAP for each item. If the difficulty level (tau) of the next item is outside the relevance interval around EAP, the procedure ignore the score on the item. The default is c(-Inf, +Inf) does not ignore scores.

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.

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
"gsed1912"	807_17	-10:100	20	mixed	GSED Team, 2019
"gsed2206"	818_17	-10:100	22	mixed	GSED Team, 2022
"gsed2208"	818_6	-10:100	22	mixed	GSED Team, 2022
"gsed2212"	818_6	-10:100	22	mixed	GSED Team, 2022
"lf2206"	155_0	-10:100	1	direct	GSED Team, 2022
"sf2206"	139_0	-10:100	1	caregiver	GSED Team, 2022

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 default, which currently links to "gsed2212".

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_phase1(), dscore:::count_mu_gcdg() 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 1.5 to 2 times 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 (2022)

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. https://stefvanbuuren.name/publication/van-buuren-2014-gc/

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

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 = NULL,
  itembank = dscore::builtin_itembank,
  population = NULL,
  xunit = c("decimal", "days", "months")
)
```

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 pct = c(10, 50, 90).
key	A string that selects a subset in the itembank that makes up the key, the set of difficulty estimates from a fitted Rasch model. The built-in keys are: "gsed2212" (default), "gsed2208" (deprecated), "gsed2206" (deprecated), "gsed1912", "lf2206", "sf2206", "gcdg", and "dutch". Since version 1.5.0, the key = "gsed" selects the latest key starting with the string "gsed". Use key = "" to use all item names, which should only be done if there are no duplicate itemnames in the itembank.
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.

population	A string describing the population. Currently supported are "phase1" (default), "dutch", "gcdg".
xunit	A string specifying the unit in which age is measured (either "decimal", "days" or "months"). The default ("decimal") means decimal age in years.

Details

The function internally defines a scale factor given the key.

Value

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

Examples

```
get_age_equivalent(c("gpagmc018", "gtogmd026", "ddicmm050"))
```

---

get_itemnames	*Extract item names*

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,
  order = "idnm"
)
```

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.
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".

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, and the item label. The `get_itemtable()` function returns a subset of `items` in the itemtable.

Usage

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

Arguments

items
: A logical or character vector of item names to return. The default (NULL) returns all items.

itemtable
: A `data.frame` set up according to the same structure as [builtin_itemtable()](). If not specified, the `builtin_itemtable` is used. If `itemtable = ""`, then a dynamic item table is created from any specified item names.

decompose
: If `TRUE`, the function adds four columns: `instrument`, `domain`, `mode` and `number`.

Value

A `data.frame` with seven columns.

See Also

[get_labels()](), [get_itemnames()]()

Examples

```
head(get_itemtable(), 3)
get_itemtable(LETTERS[1:3], "")
```

---

get_labels *Get labels for items*

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

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

Value

A named character vector with length(items) elements with item labels, in the same order as in items.

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	*Get D-score reference*

Description

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

Usage

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

Arguments

population	A string describing the population. Currently supported are "dutch", "gcdg" and "phase1".
references	A data.frame with the same structure as builtin_references. The default is to use builtin_references.

Value

A data.frame with the LMS reference values.

Note

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

The "dutch" reference was published in Van Buuren (2014) The "gcdg" was calculated from 15 cohorts with direct observations (Weber, 2019). The "phase1" references were calculated from the GSED Phase 1 validation data (GSED-BGD, GSED-PAK, GSED-TZA) cover age range 2w-3.5 years. The age range 3.5-5 yrs is linearly extrapolated and are only indicative.

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 = NULL, 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 selects a subset in the itembank that makes up the key, the set of difficulty estimates from a fitted Rasch model. The built-in keys are: "gsed2212" (default), "gsed2208" (deprecated), "gsed2206" (deprecated), "gsed1912", "lf2206", "sf2206", "gcdg", and "dutch". Since version 1.5.0, the key = "gsed" selects the latest key starting with the string "gsed". Use key = "" to use all item names, which should only be done if there are no duplicate itemnames in the itembank.

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"))
```

---

gsample   *Sample of 10 children from the GSED Phase 1 study*

Description

A demo dataset with developmental scores at the item level for 10 random children from the GSED
Phase 1 data.

Usage

```
gsample
```

Format

A data.frame with 10 rows and 295 variables:

Name	Label
id	Integer, child ID
agedays	Integer, age in days
gpalac001	Integer, Cry when hungry...: 1 = yes, 0 = no, NA = not administered
gpalac002	Integer, Look at/focus...: 1 = yes, 0 = no, NA = not administered
...	and so on..

There are 138 gpa items (item gpamoc008 (clench fists) removed) from GSED SF and and 155 gto
items from GSED LF.

See Also

[dscore()](#)

Examples

```
head(gsample)
```

---

milestones   *Outcomes on developmental milestones for preterm-born children*

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

See Also

[dscore()](#)

Examples

```
head(milestones)
```

---

normalize   *Normalize distribution*

Description

Normalizes the distribution so that the total mass equals 1.

Usage

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

Arguments

d	A vector with `length(qp)` 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 quadature 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	*Calculate posterior for one item given score, difficulty and prior*

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 `rasch.pairwise.itemcluster()` of the `sirt` package.

Value

A vector of length `length(prior)`

Note

: Internal function

Author(s)

Stef van Buuren, Arjan Huizing, 2020

See Also

[dscore()]

---

rename_gcdg_gsed   *Rename items from gcdg into gsed lexicon*

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 unmatches names should be copied (copy = TRUE) 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), Tepsi (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
```

---

sample_hf   *Sample of 10 children from GSED HH*

Description

A demo dataset with developmental scores at the item level for 10 random children from the GSED Phase 1 data.

Usage

```
sample_hf
```

Format

A data.frame with 10 rows and 57 variables:

Name	Label
subjid	Integer, child ID
agedays	Integer, age in days
hf001	Integer, ...: 1 = yes, 0 = no, NA = not administered
hf002	Integer, ...: 1 = yes, 0 = no, NA = not administered
...	and so on..

Sample data for 55 gpa items forming GSED HH V1

See Also

[dscore()](#)

Examples

```
head(sample_hf)
```

---

sample_lf   *Sample of 10 children from gto (LF)*

Description

A demo dataset with developmental scores at the item level for 10 random children from the GSED
Phase 1 data.

Usage

```
sample_lf
```

Format

A `data.frame` with 10 rows and 157 variables:

	Name	Label
	subjid	Integer, child ID
	agedays	Integer, age in days
	lf001	Integer, ...: 1 = yes, 0 = no, NA = not administered
	lf002	Integer, ...: 1 = yes, 0 = no, NA = not administered
	...	and so on..

Sample data for 155 `gto` items from GSED SF

See Also

[dscore()]

Examples

```
head(sample_lf)
```

---

sample_sf   *Sample of 10 children from gpa (SF)*

Description

A demo dataset with developmental scores at the item level for 10 random children from the GSED
Phase 1 data.

Usage

```
sample_sf
```

Format

A `data.frame` with 10 rows and 141 variables:

Name	Label
subjid	Integer, child ID
agedays	Integer, age in days
sf001	Integer, Cry when hungry...: 1 = yes, 0 = no, NA = not administered
sf002	Integer, Look at/focus...: 1 = yes, 0 = no, NA = not administered
...	and so on..

Sample data for 139 gpa items from GSED SF

See Also

dscore()

Examples

```
head(sample_sf)
```

---

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")
sort_itemnames(itemnames)
```

Index