

# Package: whomds (via r-universe)

September 14, 2024

**Type** Package

**Title** Calculate Results from WHO Model Disability Survey Data

**Version** 1.1.1.9000

**Description** The Model Disability Survey (MDS)

<<https://www.who.int/activities/collection-of-data-on-disability>>  
is a World Health Organization (WHO) general population survey instrument to assess the distribution of disability within a country or region, grounded in the International Classification of Functioning, Disability and Health  
<<https://www.who.int/standards/classifications/international-classification-of-functioning-disability-and-health>>. This package provides fit-for-purpose functions for calculating and presenting the results from this survey, as used by the WHO. The package primarily provides functions for implementing Rasch Analysis (see Andrich (2011) <[doi:10.1586/erp.11.59](https://doi.org/10.1586/erp.11.59)>) to calculate a metric scale for disability.

**License** GPL-3

**Encoding** UTF-8

**LazyData** true

**Imports** colorspace, dplyr, eRm, ggraph, ggplot2, GPArotation, grDevices, igraph, nFactors, plyr, polycor, purrr, RColorBrewer, readr, rlang, scales, srvyr, stringr, TAM, tibble, tidygraph, tidyr, WrightMap

**Depends** R (>= 3.6.0), psych

**BugReports** <https://github.com/lindsayevanslee/whomds/issues>

**RoxygenNote** 7.2.0

**Suggests** covr, httr, knitr, lubridate, rmarkdown, testthat

**VignetteBuilder** knitr

**URL** <https://github.com/lindsayevanslee/whomds>

**Repository** <https://lindsayevanslee.r-universe.dev>

**RemoteUrl** <https://github.com/lindsayevanslee/whomds>

**RemoteRef** HEAD

**RemoteSha** 4c3afa13ae56d87dd4e089e270765b38c17c570f

## Contents

df_adults . . . . .	3
df_children . . . . .	6
fig_density . . . . .	8
fig_dist . . . . .	9
fig_LID . . . . .	11
fig_poppyramid . . . . .	12
helper_checkrow . . . . .	13
helper_indicator . . . . .	14
helper_installation . . . . .	15
helper_palette . . . . .	15
helper_rowSums . . . . .	16
helper_varlist . . . . .	17
rasch_df_nest . . . . .	17
rasch_DIF . . . . .	18
rasch_drop . . . . .	20
rasch_factor . . . . .	21
rasch_mds . . . . .	22
rasch_mds_children . . . . .	24
rasch_model . . . . .	26
rasch_model_children . . . . .	28
rasch_quality_children . . . . .	28
rasch_quality_children_print . . . . .	29
rasch_rawscore . . . . .	30
rasch_recode . . . . .	31
rasch_rescale . . . . .	32
rasch_rescale_children . . . . .	33
rasch_split . . . . .	34
rasch_split_age . . . . .	35
rasch_testlet . . . . .	36
SepRel_1.0.1 . . . . .	37
table_basicstats . . . . .	38
table_unweightedpctn . . . . .	39
table_weightedpct . . . . .	40

**Index**

**42**

---

`df_adults`*Example of WHO Model Disability Survey data for adults*

---

## Description

An dummy data set of data from the WHO Model Disability Survey. All survey variables and demographic characteristics (except work status) are randomly generated. Responses to the survey questions and work status are from randomly selected rows of the 2014 Chilean implementation of the MDS, ENDIS II (2014).

## Usage

`df_adults`

## Format

A tibble with 2500 rows and 90 variables:

**HHID** household number

**strata** survey strata

**PSU** PSU id

**weight** survey weights

**sex** sex, Male or Female

**age** age in years

**age\_cat** age category, one of "18-24", "25-39", "40-64", or "64-100"

**edu\_cat** highest level of education attained

**work\_cat** whether respondent is currently working, binary

**disability\_score** metric scale of performance, from Rasch analysis

**disability\_cat** performance level, as defined by WHO cutoffs of *disability\_score* *Functioning section: On a scale from 1 (none) to 5 (extreme), in the last 30 days, taking into account any help or support you receive, how much has of a problem has it been for you to...*

**F1** ...stand up from sitting?

**F2** ...stand for long periods of time, for example 30 minutes?

**F3** ...leave the house?

**F4** ...walk short distances, such as a street block or 100 meters?

**F5** ...walk 10 blocks or a kilometer?

**F6** ...do vigorous activities, such as play football, lift heavy objects, ride a bike or run?

**F7** ...get where you want to go?

**F8** ...manipulate small objects or opening a container?

**F9** ...lift a 2L full bottle of water from your waist to eye level?

**F10** ...wash or dress yourself?

- F11** ...feed yourself?
- F12** ...use the bathroom?
- F13** ...cut your toenails?
- F14** ...take care of your health, for example exercise, eat well or take your medications?
- F15** ...see objects at a distance?
- F16** ...see objects at arms length?
- F17** ...hear a conversation in a quiet room?
- F18** ...hear a conversation in a loud room?
- F19** ...feel pain?
- F20** ...sleep?
- F21** ...feel tired or not have enough energy?
- F22** ...feel short of breath?
- F23** ...cough or wheeze?
- F24** ...feel sad, down or depressed?
- F25** ...feel worried, nervous or anxious?
- F26** ...get along with people close to you, including your family and friends?
- F27** ...get along with people you don't know?
- F28** ...make new friends or maintain your friendships?
- F29** ...have intimate relationships?
- F30** ...manage stress?
- F31** ...cope with everything you have to do?
- F32** ...be understood in your usual language?
- F33** ...understand others in your usual language?
- F34** ...forget things?
- F35** ...remember the important things you have to do day-to-day?
- F36** ...find solutions to your day-to-day problems?
- F37** ...complete household tasks, like sweeping, cooking, arranging the house or taking out the trash?
- F38** ...manage the money you have?
- F39** ...do things for relaxation?
- F40** ...participate in community activities?
- F41** ...participate in local or national politics or civil society?
- F42** ...take care of others?
- F43** ...get a job?
- F44** ...obtain a higher education?
- F45** ...use public transportation?
- F46** ...get things done in your job (if not currently working, NA)?

- F47** ...get things done at your school (if not currently studying, NA)? *Capacity section: On a scale from 1 (none) to 5 (extreme), in the last 30 days, withing taking into account any type of help or support, due to your health how much difficulty have you had with...*
- C2** ...seeing, without contact lenses or glasses?
- C3** ...hearing, without hearing aids?
- C4** ...walking or climbing steps?
- C5** ...remembering or concentrating?
- C6** ...washing or dressing?
- C7** ...communicating in your usual language?
- C8** ...feeding yourself?
- C9** ...using the bathroom?
- C10** ...waking up and getting out of bed?
- C11** ...going out to the street?
- C12** ...doing shopping or going to the doctor?
- C13** ...manipulating small objects or opening a container?
- C14** ...sleeping?
- C15** ...breathing?
- C16** ...doing household tasks like sweeping, cooking, arranging the house or taking out the trash?
- C17** ...taking care of others?
- C18** ...participating in community activities?
- C19** ...feeling sad, down or depressed?
- C20** ...feeling worried, nervous or anxious?
- C21** ...getting along with people close to you, including your family and friends?
- C22** ...coping with everything you have to do?
- C23** ...feeling pain?
- C24** ...getting things done in your job (if not currently working, NA)?
- C25** ...getting things done at your school (if not currently studying, NA)? *Environmental factors section: On a scale from 1 (very easy) to 5 (very hard), to what extent...*
- EF1** ...do health facilities you need regularly make it easy or hard for you to use them?
- EF2** ...do places where you socialize and engage in community activities make it easy or hard for you to do this?
- EF3** ...do the shops, banks and post office in your neighbourhood make it easy or hard for you to use them?
- EF4** ...do your regular places of worship make it easy or hard for you to worship?
- EF5** ...does the transportation you need or want to use make it easy or hard for you to use it?
- EF6** ...does your dwelling (including the toilet and all rooms) make it easy or hard for you to live there?
- EF7** ...do the temperature, terrain, and climate of the place you usually live make it easy or hard for you to live there?

- EF8** ...does the lighting in your surroundings make it easy or hard for you to live there?  
**EF9** ...does the noise in your surroundings make it easy or hard for you to live there?  
**EF10** ...do the crowds in your surroundings make it easy or hard for you to live there?  
**EF11** ...does your workplace make it easy or hard for you to work or learn (if not currently working, NA)?  
**EF12** ...does your educational institution make it easy or hard for you to work or learn (if not currently studying, NA)?

---

df\_children

*Example of WHO Model Disability Survey data for children*


---

### Description

An dummy data set of data from the WHO Model Disability Survey for children. All survey variables and demographic characteristics (except age variables) are randomly generated. Responses to the survey questions and age variables are from randomly selected rows of the 2014 Chilean implementation of the MDS, ENDIS II (2014).

### Usage

```
df_children
```

### Format

A tibble with 2500 rows and 42 variables:

**HHID** household number

**strata** survey strata

**PSU** PSU id

**weight** survey weights

**sex** sex, Male or Female

**age** age in years

**age\_cat** age category, one of "Age2to4", "Age5to9", or "Age10to17" *Functioning section: On a scale from 1 (none) to 5 (extreme), in the last 30 days, taking into account any help or support the child receives, compared with children of the same age, how much has of a problem it been for the child to...*

**child1** ...walk?

**child2** ...manipulat small objects or opening a container?

**child3** ...see things from a long distance?

**child4** ...hear?

**child5** ...feel pain?

**child6** ...not have enough energy?

- child7** ...have too much energy
- child8** ...feel short of breath?
- child9** ...feel sad, down or depressed?
- child10** ...feel worried, nervous or anxious?
- child11** ...(for children aged 5 to 17) wash or dress him- or herself?
- child12** ...(for children aged 2 to 4) bite or hit other children or adults?
- child13** ...(for children aged 5 to 17) control his or her own behavior?
- child14** ...(for children aged 5 to 17) get along with children of the same age?
- child15** ...(for children aged 2 to 4) understand what you say to him or her?
- child16** (for children aged 2 to 4) How much of a problem has it been for you to understand what the child says?
- child17** ...(for children aged 5 to 17) understand other people?
- child18** ...(for children aged 5 to 17) be understood?
- child19** ...(for children aged 2 to 3) learn the names of household objects?
- child20** ...(for children aged 3 to 17) learn to do new things?
- child21** ...(for children aged 5 to 17) complete a task?
- child22** ...(for children aged 5 to 17) make changes to his or her routine?
- child23** ...(for children aged 5 to 17) do homework as requested at school?
- child24** ...(for children aged 2 to 5) play with toys or domestic objects?
- child25** ...(for children aged 2 to 12) play with other children?
- child26** ...(for children aged 13 to 17) do activities with other children?
- child27** ...(for children aged 5 to 17) participate in community activities? *Capacity section: On a scale from 1 (none) to 5 (extreme), in the last 30 days, without taking into account any help or support the child receives, how much has difficulty has the child had with...*
- child28** ...seeing without glasses?
- child29** ...hearing without hearing aids?
- child30** ...walking?
- child31** ...understanding you or others?
- child32** ...learning?
- child33** ...controlling his or her behavior?
- child34** ...completing a task?
- child35** ...getting along with other children?

---

fig\_density                      *Plot a density of a score*

---

## Description

Plot a density of a score

## Usage

```
fig_density(
  df,
  score,
  var_color = NULL,
  var_facet = NULL,
  cutoffs = NULL,
  x_lab = "Score",
  pal = "Paired",
  adjust = 2,
  size = 1.5
)
```

## Arguments

df	a data frame where each row is an individual, containing at least a score column (between 0 and 100)
score	a string (length 1) of the column name for the score variable to print the distribution of
var_color	a string (length 1) of the column name for the variable to set color of density lines by. Default is NULL.
var_facet	a string (length 1) of the column name for the variable to create a <code>ggplot2::facet_grid()</code> with. Default is NULL.
cutoffs	a numeric vector of the cut-offs for the score categorization. Default is NULL.
x_lab	a string (length 1) of x-axis label. Default is "Score".
pal	a string specifying either a manual color to use for the color aesthetic, a character vector explicitly specifying the colors to use for the color scale, or as the name of a palette to pass to <code>RColorBrewer::brewer.pal()</code> with the name of the color palette to use for the color scale. Default is "Paired"
adjust	a numeric value to pass to <code>adjust</code> argument of <code>ggplot2::geom_density()</code> . Default is 2.
size	a numeric value to pass to <code>size</code> argument of <code>ggplot2::geom_density()</code> . Default is 1.5.



## Details

Plots a histogram of a score that ranges between 0 and 100, with the fill determined by some set categorization of the score. This is the function used to plot the distributions of disability scores resulting from the WHO Model Disability Survey.

## Value

A density figure

## See Also

Other figure functions: [fig\\_LID\(\)](#), [fig\\_dist\(\)](#), [fig\\_poppyramid\(\)](#)

## Examples

```
fig_density(df_adults, score = "disability_score", cutoffs = c(19.1, 34.4, 49.6),
x_lab = "Disability score")
fig_density(df_adults, score = "disability_score", var_color = "sex",
cutoffs = c(19.1, 34.4, 49.6), x_lab = "Disability score")
fig_density(df_adults, score = "disability_score", var_color = "sex",
var_facet = "age_cat", cutoffs = c(19.1, 34.4, 49.6), x_lab = "Disability score")
```

---

fig\_dist

*Plot a distribution of a score*

---

## Description

Plot a distribution of a score

## Usage

```
fig_dist(
  df,
  score,
  score_cat,
  cutoffs,
  x_lab = "Score",
  y_max = NULL,
  pcent = FALSE,
  pal = "Blues",
  binwidth = 5
)
```

**Arguments**

df	a data frame where each row is an individual, containing at least a score column (between 0 and 100) and a categorization of that score
score	a string (length 1) of the column name for the score variable to print the distribution of
score_cat	a string (length 1) of the column name for the categorization of the score variable
cutoffs	a numeric vector of the cut-offs for the score categorization
x_lab	a string (length 1) of x-axis label. Default is "Score".
y_max	a numeric value of the maximum limit on the y-axis. Default is NULL to use default value from geom_histogram()
pcent	a logical value determining whether or not to display the distribution as percentages or frequency. Default is FALSE, to display as frequency.
pal	a string to pass to RColorBrewer::brewer.pal() with the name of the color palette to use
binwidth	a numeric value giving the width of the bins in the histogram. Default is 5.

**Details**

Plots a histogram of a score that ranges between 0 and 100, with the fill determined by some set categorization of the score. This is the function used to plot the distributions of disability scores resulting from the WHO Model Disability Survey.

**Value**

A score distribution figure with fill based on categorization of the score

**See Also**

Other figure functions: [fig\\_LID\(\)](#), [fig\\_density\(\)](#), [fig\\_poppyramid\(\)](#)

**Examples**

```
fig_dist(df_adults, score = "disability_score", score_cat = "disability_cat",
cutoffs = c(19.1, 34.4, 49.6), x_lab = "Disability score")
fig_dist(df_adults, score = "disability_score", score_cat = "disability_cat",
cutoffs = c(19.1, 34.4, 49.6), x_lab = "Disability score", y_max = 2000)
fig_dist(df_adults, score = "disability_score", score_cat = "disability_cat",
cutoffs = c(19.1, 34.4, 49.6), x_lab = "Disability score", y_max = 0.2, pcent=TRUE)
```

---

`fig_LID`*Print a graph showing significant correlations between survey items*

---

## Description

Print a graph showing significant correlations between survey items

## Usage

```
fig_LID(  
  LIDforgraph,  
  LIDcutoff = 0.2,  
  path_output,  
  extra_file_label = NULL,  
  vertex_print_grey = NULL  
)
```

## Arguments

`LIDforgraph` a square matrix of item correlations

`LIDcutoff` a numeric value between 0 and 1 for the cut-off for significant correlation

`path_output` a string with the path to the output folder

`extra_file_label`  
a string to tack on to the end of names of files outputted and the title of the plot.  
Default is NULL.

`vertex_print_grey`  
a character vector with the names of vertices to print in "lightgrey", with all others printed in "skyblue". If left as default NULL, all vertices will be printed in "lightgrey".

## Details

This function could be applied to visualize any kind of correlations. But within the context of the Rasch Analysis used for the WHO Model Disability Survey, the residual correlations are used when analyzing item dependence.

## Value

Returns a ggplot graph showing the items with correlation  $\geq$  LIDcutoff and prints a csv of the corresponding correlations

## See Also

Other figure functions: [fig\\_density\(\)](#), [fig\\_dist\(\)](#), [fig\\_poppyramid\(\)](#)

---

fig_poppyramid	<i>Print a population pyramid</i>
----------------	-----------------------------------

---

**Description**

Print a population pyramid

**Usage**

```
fig_poppyramid(  
  df,  
  var_age,  
  var_sex,  
  x_axis = c("n", "pct"),  
  age_plus = 100,  
  age_by = 5  
)
```

**Arguments**

df	a data frame of individual survey data, where each row is an individual
var_age	a string (length 1) of the name of the age column
var_sex	a string (length 1) of the name of the sex column
x_axis	a string (length 1) indicating whether to use absolute numbers or sample percentage on the x-axis. Choices are "n" (default) or "pct".
age_plus	a numeric value (length 1) indicating the age that is the first value of the oldest age group. Default is 100, for the last age group to be 100+
age_by	a numeric value (length 1) indicating the width of each age group, in years. Default is 5.

**Details**

A population pyramid gives an image of the age and sex distribution of a population.

The function divides the var\_age variable into intervals of width age\_by.

**Value**

A population pyramid figure

**See Also**

Other figure functions: [fig\\_LID\(\)](#), [fig\\_density\(\)](#), [fig\\_dist\(\)](#)

**Examples**

```
fig_poppyramid(df_adults, "age", "sex")
```

---

helper_checkrow	<i>Check a vector for particular values</i>
-----------------	---

---

### Description

Check a vector for particular values

### Usage

```
helper_checkrow(  
  row,  
  check_type = c("all_equal", "all_not_equal", "any_in"),  
  check_value,  
  na_rm = TRUE  
)
```

### Arguments

row	a vector of data
check_type	a string indicating the type of check to do on row. Options are "all_equal", "all_not_equal", and "any_in". A value of "all_equal" will check that all values in row equal check_value. A value of "all_not_equal" will check that all values of row are not equal to check_value. A value of "any_in" will check if any of the values in check_value are in row.
check_value	a vector of values to check against
na_rm	logical indicating whether or not to perform check after removing NAs, passed to argument na.rm of any() or all(). Default is TRUE.

### Details

If all values of row are NA, then an NA is returned.

### Value

a logical vector with the result of the check

### See Also

Other helper functions: [helper\\_palette\(\)](#), [helper\\_varslist\(\)](#)

---

helper\_indicator      *Create indicators from data frame*

---

### Description

Create indicators from data frame

### Usage

```
helper_indicator(  
  df,  
  vars_indicators,  
  mapvalues_from,  
  mapvalues_to,  
  make_factor = TRUE  
)
```

### Arguments

df                    a data frame

vars\_indicators      a character vector of the variables from df to create indicators for

mapvalues\_from      vector to pass to `plyr::mapvalues()` argument from

mapvalues\_to        vector to pass to `plyr::mapvalues()` argument to

make\_factor         a logical indicating whether resulting indicators should be factors. Default is TRUE.

### Value

a data frame with new columns that are the indicators from `vars_indicators`, with the same names pasted with `"_ind"`.

### Examples

```
helper_indicator(df = df_adults,  
  vars_indicators = c("EF1", "EF2", "EF3"),  
  mapvalues_from = 1:5,  
  mapvalues_to = c(0,0,0,1,1))
```

---

helper\_installation     *Check installation of whomds is the most updated*

---

**Description**

Compares build date of installed package against the date of the last commit from GitHub (<https://github.com/lindsayevanslee/whomds>)

**Usage**

```
helper_installation()
```

**Value**

Prints a message stating whether or not installed package is same as most updated version from Github

---

helper\_palette             *Color palette for the MDS*

---

**Description**

Color palette for the MDS

**Usage**

```
helper_palette(  
  n,  
  h = c(-100, 100),  
  c. = c(60, 100),  
  l = c(15, 95),  
  power = c(2, 0.9),  
  fixup = TRUE,  
  gamma = NULL,  
  alpha = 1,  
  ...  
)
```

**Arguments**

n	the number of colors ( $\geq 1$ ) to be in the palette.
h	hue value in the HCL or HSV color description, has to be in [0, 360] for HCL and in [0, 1] for HSV colors.
c.	chroma value in the HCL color description.
l	luminance value in the HCL color description.

power	control parameter determining how chroma and luminance should be increased (1 = linear, 2 = quadratic, etc.).
fixup	logical. Should the color be corrected to a valid RGB value before correction?
gamma	Deprecated.
alpha	numeric vector of values in the range [0, 1] for alpha transparency channel (0 means transparent and 1 means opaque).
...	Other arguments passed to <a href="#">hex</a> .

**Details**

Generated with `colorspace::choose_palette()`

**Value**

n hex codes from the color palette

**See Also**

Other helper functions: [helper\\_checkrow\(\)](#), [helper\\_varslis\(\)](#)

---

helper_rowSums	<i>Perform row sum</i>
----------------	------------------------

---

**Description**

Perform row sum

**Usage**

```
helper_rowSums(x, allNA0 = TRUE, ...)
```

**Arguments**

x	an array of two or more dimensions, containing numeric, complex, integer or logical values, or a numeric data frame
allNA0	logical indicating that if a whole row is NA to give the row sum as NA. Only works for two dimensional x and if <code>na.rm</code> is TRUE.
...	other arguments to pass to <code>base::rowSums()</code>

**Details**

Essentially equivalent to `base::rowSums()` except with the addition of the `allNA0` argument.

**Value**

A numeric or complex array of suitable size, or a vector if the result is one-dimensional.



**Examples**

```
x <- data.frame(v1 = c(NA,1:4), v2 = c(NA, 2:5), v3 = c(NA, 1:2, NA, 3))
helper_rowSums(x, na.rm = TRUE, allNA0 = TRUE)
helper_rowSums(x, na.rm = TRUE, allNA0 = FALSE)
```

---

helper_varslist	<i>Convert list to vector with unique elements</i>
-----------------	--

---

**Description**

Convert list to vector with unique elements

**Usage**

```
helper_varslist(vars_list)
```

**Arguments**

vars\_list      a list of character vectors

**Value**

a character vector of all unique items from vars\_list

**See Also**

Other helper functions: [helper\\_checkrow\(\)](#), [helper\\_palette\(\)](#)

---

rasch_df_nest	<i>Nest data by age group</i>
---------------	-------------------------------

---

**Description**

Nest data by age group

**Usage**

```
rasch_df_nest(df, vars_group, vars_id)
```

**Arguments**

df              a data frame of individual survey data, where each row is an individual  
vars\_group      a string with the column name identifying grouping variable  
vars\_id         a string with column name uniquely identifying individuals

**Value**

nested tibble with two columns: the age group given by `vars_group` and `df_split` with the split data

**See Also**

Other rasch functions: `rasch_DIF()`, `rasch_drop()`, `rasch_factor()`, `rasch_mds_children()`, `rasch_mds()`, `rasch_model_children()`, `rasch_model()`, `rasch_quality_children_print()`, `rasch_quality_children()`, `rasch_rawscore()`, `rasch_recode()`, `rasch_rescale_children()`, `rasch_rescale()`, `rasch_split_age()`, `rasch_split()`, `rasch_testlet()`

Other children analysis functions: `rasch_drop()`, `rasch_mds_children()`, `rasch_model_children()`, `rasch_quality_children_print()`, `rasch_quality_children()`, `rasch_recode()`, `rasch_rescale_children()`, `rasch_split_age()`, `rasch_split()`, `rasch_testlet()`

---

rasch_DIF	<i>Perform analysis of Differential Item Functioning (DIF) for Rasch Model</i>
-----------	--

---

**Description**

Perform analysis of Differential Item Functioning (DIF) for Rasch Model

**Usage**

```
rasch_DIF(
  df,
  vars_metric,
  vars_DIF,
  residuals_PCM,
  split_strategy = NULL,
  print_results = FALSE,
  path_output = NULL,
  breaks = 6
)
```

**Arguments**

<code>df</code>	a data frame of individual survey data, where each row is an individual
<code>vars_metric</code>	a character vector of items to use in the Rasch Analysis
<code>vars_DIF</code>	a string with the column names to use for analyzing differential item functioning (DIF). Default is NULL, to skip analysis of DIF.
<code>residuals_PCM</code>	a matrix giving the residuals of the person parameters from the Rasch Model. Row names are the numbers of the people and the columns are for each variable.

<code>split_strategy</code>	a named list giving the strategy to take for splitting variables by categories, passed to <code>rasch_split()</code> . One element of the list per variable to split by. Each element of the list must be a character vector of column names to split. The names of the list are the variables to split each group of variables by. Default is <code>NULL</code> , to not split items.
<code>print_results</code>	a logical value indicating whether or not to print various files displaying results from the Rasch Model. Default is <code>FALSE</code> , to not print the files.
<code>path_output</code>	a string with the path to the output folder. Default is <code>NULL</code> .
<code>breaks</code>	a numeric value giving the number of class intervals. Default is 6.

### Details

Differential Item Functioning (DIF) refers to the circumstance when different groups in a sample respond to items in different ways. For instance, DIF would be observed if men and women had different patterns of responses to a set of survey questions. DIF can cause poor fit for the Rasch Model, and therefore should be analyzed. This function uses ANOVA to find DIF by the variables supplied and by a generated class interval.

### Value

a list with results from the DIF analysis:

<code>df_DIF_class</code>	the person residuals from the Rasch Model, the assigned class intervals, and the variables used for DIF analysis
<code>tab_aov_DIF</code>	the results of the ANOVA used to analyze DIF
<code>DIF_results</code>	string of variables that exhibit some form of DIF

### Note

Currently the calculation of the class intervals is quite slow. Reducing the number of breaks can improve speed.

### See Also

Other rasch functions: [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

 rasch\_drop

*Drop items from a Rasch Analysis*


---

**Description**

Drop items from a Rasch Analysis

**Usage**

```
rasch_drop(vars_metric, drop_vars, max_values)
```

**Arguments**

vars_metric	a character vector of items to use in the Rasch Analysis
drop_vars	a character vector of column names to drop from the Rasch Analysis. Default is NULL, to not drop items.
max_values	a tibble with two columns, var equivalent to vars_metric and max_val with their corresponding maximum possible values

**Details**

Dropping variables might be desirable if one finds that particular items are causing a lot of problems for the fit of a Rasch Model.

**Value**

a named list with:

vars_metric	new vars_metric after dropping the desired variables
max_values	new max_values after dropping the desired variables

**See Also**

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

Other children analysis functions: [rasch\\_df\\_nest\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

rasch_factor	<i>Calculate a factor analysis for a Rasch Model</i>
--------------	--

---

**Description**

Calculate a factor analysis for a Rasch Model

**Usage**

```
rasch_factor(df, vars_metric, print_results = FALSE, path_output = NULL)
```

**Arguments**

df	a data frame of individual survey data, where each row is an individual
vars_metric	a character vector of items to use in the Rasch Analysis
print_results	a logical value indicating whether or not to print various files displaying results from the Rasch Model. Default is FALSE, to not print the files.
path_output	a string with the path to the output folder. Default is NULL.

**Details**

Unidimensionality of the data is one of the core assumptions of the Rasch Model. This function performs the factor analysis to assess the unidimensionality of the data.

**Value**

a named list with results from the factor analysis for a Rasch Model:

cor_poly	the matrix of polychoric correlations
eigenvalues	the eigenvalues
parallel_analysis	permutation parallel analysis distribution
results_scree	results of a scree analysis
n_group_factors	number of factors from the parallel analysis in the scree analysis
fa_onefactor	results from factor analysis with one factor
fa_resid	local dependency based on polychoric correlations of the items

**See Also**

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

rasch_mds	<i>Top-level function to perform Rasch Analysis on WHO Model Disability Survey data</i>
-----------	---

---

## Description

Top-level function to perform Rasch Analysis on WHO Model Disability Survey data

## Usage

```

rasch_mds(
  df,
  vars_metric,
  vars_id,
  vars_DIF = NULL,
  resp_opts = 1:5,
  max_NA = 2,
  print_results = FALSE,
  path_parent = NULL,
  model_name = NULL,
  testlet_strategy = NULL,
  recode_strategy = NULL,
  drop_vars = NULL,
  split_strategy = NULL,
  comment = NULL
)

```

## Arguments

df	a data frame of individual survey data, where each row is an individual
vars_metric	a character vector of items to use in the Rasch Analysis
vars_id	a string with column name uniquely identifying individuals
vars_DIF	a string with the column names to use for analyzing differential item functioning (DIF). Default is NULL, to skip analysis of DIF.
resp_opts	a numeric vector of possible response options for vars_metric. Must begin with 1. Default is 1:5
max_NA	a numeric value for the maximum number of NAs allowed per individual among vars_metric. Default is 2.
print_results	a logical value indicating whether or not to print various files displaying results from the Rasch Model. Default is FALSE, to not print the files.
path_parent	a string with the path to the folder where results from multiple models will be outputted. Default is NULL
model_name	a string with a name for the model, which is used to create a new folder for model output. Default is NULL.

<code>testlet_strategy</code>	a list giving the strategy to take for creating testlets, passed to <code>rasch_testlet()</code> . One element of the list per testlet to create. Each element of the list must be a character vector of column names to use for the testlet. Optionally, name the element of the list to give the name of the new testlet. Otherwise, the new testlet will be the original column names separated by "_". Default is NULL, to not create testlets.
<code>recode_strategy</code>	a named list giving the strategy to take for recoding variables, passed to <code>rasch_recode()</code> . One element of the list per recode strategy. Each element of the list is a numeric vector giving the new values to map the variables to. The names of the list are the groups of column names to use for each recoding strategy, separated only by ",". Default is NULL, to not recode items.
<code>drop_vars</code>	a character vector of column names to drop from the Rasch Analysis. Default is NULL, to not drop items.
<code>split_strategy</code>	a named list giving the strategy to take for splitting variables by categories, passed to <code>rasch_split()</code> . One element of the list per variable to split by. Each element of the list must be a character vector of column names to split. The names of the list are the variables to split each group of variables by. Default is NULL, to not split items.
<code>comment</code>	a string giving a comment describing the analysis, printed to a txt file. Default is NULL, to not print a comment.

## Details

This function combines all of the separate analyses of model fit necessary to assess the quality of the Rasch Model. It is designed to require minimal intervention from the user. Users wishing to have more control over the analysis can use the other Rasch functions in this package separately.

## Value

a named list with:

<code>df</code>	a tibble with new columns representing the original person abilities ( <code>person_pars</code> ) and the rescaled person abilities ( <code>rescaled</code> )
<code>vars_metric</code>	a character vector with the variables used in the metric after all adjustments
<code>df_results</code>	a tibble of one row with key results of the model

If `print_results` is TRUE, prints files to the working directory with the results of the Rasch Model.

## See Also

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

rasch_mds_children	<i>Top-level function to perform Rasch Analysis on WHO Model Disability Survey data for children</i>
--------------------	--

---

### Description

Top-level function to perform Rasch Analysis on WHO Model Disability Survey data for children

### Usage

```
rasch_mds_children(
  df,
  vars_id,
  vars_group,
  vars_metric_common,
  vars_metric_grouped = NULL,
  TAM_model = "PCM2",
  vars_DIF = NULL,
  resp_opts = 1:5,
  has_at_least_one = 4:5,
  max_NA = 2,
  print_results = FALSE,
  path_parent = NULL,
  model_name = NULL,
  testlet_strategy = NULL,
  recode_strategy = NULL,
  drop_vars = NULL,
  split_strategy = NULL,
  comment = NULL
)
```

### Arguments

df	a data frame of individual survey data, where each row is an individual
vars_id	a string with column name uniquely identifying individuals
vars_group	a string with the column name identifying grouping variable
vars_metric_common	a character vector the common items among all individuals
vars_metric_grouped	a named list of character vectors with the items to use in the Rasch Analysis per group. The list should have names corresponding to the different groups, and contain character vectors of the corresponding items for each group.
TAM_model	a string with the type of IRT model to use, passed to irtmodel argument of TAM::tam(). Default is "PCM2"



vars_DIF	Currently does nothing. In the future, a string with the column names to use for analyzing differential item functioning (DIF). Default is NULL, to skip analysis of DIF.
resp_opts	a numeric vector of possible response options for vars_metric. Must begin with 1. Default is 1:5
has_at_least_one	a numeric vector with the response options that a respondent must have at least one of in order to be included in the metric calculation. See details for more information.
max_NA	a numeric value for the maximum number of NAs allowed per individual among vars_metric_common and the relevant items from vars_metric_grouped. Default is 2.
print_results	a logical value indicating whether or not to print various files displaying results from the Rasch Model. Default is FALSE, to not print the files.
path_parent	a string with the path to the folder where results from multiple models will be outputted. Default is NULL
model_name	a string with a name for the model, which is used to create a new folder for model output. Default is NULL.
testlet_strategy	a list giving the strategy to take for creating testlets, passed to rasch_testlet(). One element of the list per testlet to create. Each element of the list must be a character vector of column names to use for the testlet. Optionally, name the element of the list to give the name of the new testlet. Otherwise, the new testlet will be the original column names separated by "_". Default is NULL, to not create testlets.
recode_strategy	a named list giving the strategy to take for recoding variables, passed to rasch_recode(). One element of the list per recode strategy. Each element of the list is a numeric vector giving the new values to map the variables to. The names of the list are the groups of column names to use for each recoding strategy, separated only by ",". Default is NULL, to not recode items.
drop_vars	a character vector of column names to drop from the Rasch Analysis. Default is NULL, to not drop items.
split_strategy	a named list giving the strategy to take for splitting variables by categories, passed to rasch_split(). One element of the list per variable to split by. Each element of the list must be a character vector of column names to split. The names of the list are the variables to split each group of variables by. Default is NULL, to not split items.
comment	a string giving a comment describing the analysis, printed to a txt file. Default is NULL, to not print a comment.

## Details

This function combines all of the separate analyses of model fit necessary to assess the quality of the Rasch Model. It is designed to require minimal intervention from the user. Users wishing to have more control over the analysis can use the other Rasch functions in this package separately.

Often Rasch Analysis of children data is more difficult because of the extreme skewness of the responses. For this reason, it is often advisable to build a scale only with the respondents on the more severe end of the disability continuum. By specifying `has_at_least_one`, the function will remove all children from the sample who do endorse an answer of any of `has_at_least_one` in at least one `vars_metric`. The scores created can be reunited with the excluded children post-hoc.

### Value

a tibble with new columns representing the original person abilities (`person_pars`) and the rescaled person abilities (`rescaled`).

If `print_results` is TRUE, prints files to the working directory with the results of the Rasch Model.

### See Also

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

Other children analysis functions: [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

rasch\_model

*Run the Rasch Model and print diagnostic results*

---

### Description

Run the Rasch Model and print diagnostic results

### Usage

```
rasch_model(
  df,
  vars_metric,
  vars_id,
  print_results = FALSE,
  path_output = NULL,
  LIDcutoff = 0.2
)
```

### Arguments

<code>df</code>	a data frame of individual survey data, where each row is an individual
<code>vars_metric</code>	a character vector of items to use in the Rasch Analysis
<code>vars_id</code>	a string with column name uniquely identifying individuals

<code>print_results</code>	a logical value indicating whether or not to print various files displaying results from the Rasch Model. Default is FALSE, to not print the files.
<code>path_output</code>	a string with the path to the output folder. Default is NULL.
<code>LIDcutoff</code>	either a numeric value between 0 and 1 indicating the cut-off for significant local item dependence, or the string "christensen" to use the cut-off suggested by Christensen et al. 2017 (see reference). If "christensen" cut-off fails, defaults to 0.2.

### Details

The Rasch Model is calculated using the function `eRm : :PCM()`.

### Value

a list with results from the Rasch Model:

<code>model</code>	the results from the Rasch Model
<code>df_score</code>	a tibble with the items used in the analysis and the person abilities
<code>thresholds</code>	the item thresholds (i.e., crossings)
<code>person_parameters</code>	person abilities
<code>PSI</code>	the person-separation index
<code>item_fit</code>	infit and outfit statistics per item
<code>residuals_PCM</code>	the standardized person residuals
<code>LID</code>	matrix with the item residual correlations
<code>targeting</code>	a matrix with information on the targeting of the model
<code>fit_results</code>	a string with results of the item fit
<code>LID_results</code>	a string with results of the local item dependency
<code>disordered_results</code>	a string listing items with disordered thresholds

### References

Christensen, K. B., Makransky, G., & Horton, M. (2017). Critical Values for Yen's Q 3 : Identification of Local Dependence in the Rasch Model Using Residual Correlations. *Applied Psychological Measurement*, 41(3), 178-194. doi:10.1177/0146621616677520

### See Also

Other rasch functions: `rasch_DIF()`, `rasch_df_nest()`, `rasch_drop()`, `rasch_factor()`, `rasch_mds_children()`, `rasch_mds()`, `rasch_model_children()`, `rasch_quality_children_print()`, `rasch_quality_children()`, `rasch_rawscore()`, `rasch_recode()`, `rasch_rescale_children()`, `rasch_rescale()`, `rasch_split_age()`, `rasch_split()`, `rasch_testlet()`

---

rasch\_model\_children *Run the multigroup and anchored Rasch Model*

---

### Description

Run the multigroup and anchored Rasch Model

### Usage

```
rasch_model_children(df, df_nest, vars_metric, vars_group, TAM_model)
```

### Arguments

df	a data frame of individual survey data, where each row is an individual
df_nest	a nested tibble that contains the column df_split with the data split by the categories in the column vars_group
vars_metric	a character vector of items to use in the Rasch Analysis
vars_group	a string with the column name identifying grouping variable
TAM_model	a string with the type of IRT model to use, passed to irtmodel argument of TAM::tam(). Default is "PCM2"

### Value

a nested tibble with new columns with the Rasch Models calculated with the TAM package

### See Also

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

Other children analysis functions: [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

rasch\_quality\_children

*Calculate quality of multigroup and anchored Rasch Models*

---

### Description

Calculate quality of multigroup and anchored Rasch Models

**Usage**

```
rasch_quality_children(df_nest, vars_metric)
```

**Arguments**

`df_nest` a nested tibble that contains the column `df_split` with the data split by the categories in the column `vars_group`

`vars_metric` a character vector of items to use in the Rasch Analysis

**Value**

a nested tibble with new columns with information about model quality

**See Also**

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

Other children analysis functions: [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

rasch\_quality\_children\_print

*Print results of analysis of Rasch Model quality*

---

**Description**

Print results of analysis of Rasch Model quality

**Usage**

```
rasch_quality_children_print(  
  df_nest,  
  vars_metric,  
  vars_group,  
  TAM_model,  
  LIDcutoff = 0.2,  
  path_output  
)
```

**Arguments**

df_nest	a nested tibble that contains the column df_split with the data split by the categories in the column vars_group
vars_metric	a character vector of items to use in the Rasch Analysis
vars_group	a string with the column name identifying grouping variable
TAM_model	a string with the type of IRT model to use, passed to irtmodel argument of TAM: :tam(). Default is "PCM2"
LIDcutoff	a numeric value between 0 and 1 for the cut-off for significant correlation
path_output	a string with the path to the output folder

**Value**

does not return anything to the environment, but prints files to the folder specified in path\_output

**See Also**

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

Other children analysis functions: [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

rasch_rawscore	<i>Add the raw scores to the data and artificial individuals attaining the minimum and/or maximum</i>
----------------	---

---

**Description**

Add the raw scores to the data and artificial individuals attaining the minimum and/or maximum

**Usage**

```
rasch_rawscore(df, vars_metric, vars_id, max_values)
```

**Arguments**

df	a data frame of individual survey data, where each row is an individual
vars_metric	a character vector of items to use in the Rasch Analysis
vars_id	a string with column name uniquely identifying individuals
max_values	a tibble with two columns, var equivalent to vars_metric and max_val with their corresponding maximum possible values

**Value**

a tibble with a new column `RawScore` with the raw sum score of `vars_metric` for each individual, and artificial rows with individuals that attain the minimum and/or maximum if either is not attained in `df`. The artificial maximum row has value "MAX" in the `vars_id` column, and likewise the artificial minimum row has the value "MIN" in this column.

**See Also**

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

rasch_recode	<i>Recode survey items for use in Rasch Analysis</i>
--------------	--

---

**Description**

Recode survey items for use in Rasch Analysis

**Usage**

```
rasch_recode(df, vars_metric, recode_strategy, max_values)
```

**Arguments**

<code>df</code>	a data frame of individual survey data, where each row is an individual
<code>vars_metric</code>	a character vector of items to use in the Rasch Analysis
<code>recode_strategy</code>	a named list giving the strategy to take for recoding variables, passed to <code>rasch_recode()</code> . One element of the list per recode strategy. Each element of the list is a numeric vector giving the new values to map the variables to. The names of the list are the groups of column names to use for each recoding strategy, separated only by ",". Default is NULL, to not recode items.
<code>max_values</code>	a tibble with two columns, <code>var</code> equivalent to <code>vars_metric</code> and <code>max_val</code> with their corresponding maximum possible values

**Value**

a named list with:

<code>df</code>	new <code>df</code> after recoding the desired variables
<code>max_values</code>	new <code>max_values</code> after recoding the desired variables

**See Also**

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

Other children analysis functions: [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

rasch_rescale	<i>Rescale score from Rasch Analysis to range from 0 to 100</i>
---------------	---

---

**Description**

Rescale score from Rasch Analysis to range from 0 to 100

**Usage**

```
rasch_rescale(df, df_score, vars_id)
```

**Arguments**

df	a data frame of individual survey data, where each row is an individual
df_score	a tibble resulting from <a href="#">rasch_model()</a> with the person abilities from the Rasch Model
vars_id	a string with column name uniquely identifying individuals

**Value**

a tibble with the left join between df and df\_score and new column "rescaled" with the rescaled person abilities, ranging from 0 to 100, and filter out any rows with an artificial minimum or maximum

**See Also**

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)



---

`rasch_rescale_children`*Rescale score from Rasch Analysis for children to range from 0 to 100*

---

### Description

Rescale score from Rasch Analysis for children to range from 0 to 100

### Usage

```
rasch_rescale_children(df, df_nest, vars_group, vars_id)
```

### Arguments

<code>df</code>	a data frame of individual survey data, where each row is an individual
<code>df_nest</code>	a nested tibble that contains the column <code>df_split</code> with the data split by the categories in the column <code>vars_group</code>
<code>vars_group</code>	a string with the column name identifying grouping variable
<code>vars_id</code>	a string with column name uniquely identifying individuals

### Value

a tibble with the data `df` or unnested `df_nest` and new columns "person\_pars" and "rescaled" with the original and rescaled person abilities, ranging from 0 to 100, and filter out any rows with an artificial minimum or maximum

### See Also

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

Other children analysis functions: [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

 rasch\_split

*Split survey items by categories for a Rasch Model*


---

**Description**

Split survey items by categories for a Rasch Model

**Usage**

```
rasch_split(df, vars_metric, split_strategy, max_values)
```

**Arguments**

df	a data frame of individual survey data, where each row is an individual
vars_metric	a character vector of items to use in the Rasch Analysis
split_strategy	a named list giving the strategy to take for splitting variables by categories, passed to <code>rasch_split()</code> . One element of the list per variable to split by. Each element of the list must be a character vector of column names to split. The names of the list are the variables to split each group of variables by. Default is NULL, to not split items.
max_values	a tibble with two columns, var equivalent to <code>vars_metric</code> and <code>max_val</code> with their corresponding maximum possible values

**Details**

If significant differential item functioning (DIF) is observed, it may be desirable to split variables based on the characteristic for which DIF is observed. For example, if men and women have significantly different patterns of responses to items, then it may be desirable to split items by sex. This function performs that variable splitting.

**Value**

a named list with:

df	new df after splitting the desired variables
vars_metric	new vars_metric after splitting the desired variables
max_values	new max_values after splitting the desired variables

**See Also**

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_testlet\(\)](#)

Other children analysis functions: [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_testlet\(\)](#)

---

rasch_split_age	<i>Split all survey items by age category for a Rasch Model if they are not discrete</i>
-----------------	--

---

**Description**

Split all survey items by age category for a Rasch Model if they are not discrete

**Usage**

```
rasch_split_age(df, vars_group, vars_metric, vars_id, max_values)
```

**Arguments**

df	a data frame of individual survey data, where each row is an individual
vars_group	a string with the column name identifying grouping variable
vars_metric	a character vector of items to use in the Rasch Analysis
vars_id	a string with column name uniquely identifying individuals
max_values	a tibble with two columns, var equivalent to vars_metric and max_val with their corresponding maximum possible values

**Value**

a named list with:

df	new df after splitting the variables
vars_metric	new vars_metric after splitting the variables
max_values	new max_values after splitting the variables

**See Also**

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

Other children analysis functions: [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_split\(\)](#), [rasch\\_testlet\(\)](#)

---

<code>rasch_testlet</code>	<i>Create testlets of survey items for a Rasch Model</i>
----------------------------	--

---

### Description

Create testlets of survey items for a Rasch Model

### Usage

```
rasch_testlet(df, vars_metric, testlet_strategy, max_values, resp_opts)
```

### Arguments

<code>df</code>	a data frame of individual survey data, where each row is an individual
<code>vars_metric</code>	a character vector of items to use in the Rasch Analysis
<code>testlet_strategy</code>	a list giving the strategy to take for creating testlets, passed to <code>rasch_testlet()</code> . One element of the list per testlet to create. Each element of the list must be a character vector of column names to use for the testlet. Optionally, name the element of the list to give the name of the new testlet. Otherwise, the new testlet will be the original column names separated by "_". Default is NULL, to not create testlets.
<code>max_values</code>	a tibble with two columns, var equivalent to <code>vars_metric</code> and <code>max_val</code> with their corresponding maximum possible values
<code>resp_opts</code>	a numeric vector of possible response options for <code>vars_metric</code> . Must begin with 1. Default is 1:5

### Details

If high local item dependence is observed (i.e., residual correlation) is observed between items, it may be desirable to combine them into a testlet. This code creates the testlets as desired.

### Value

a named list with:

<code>df</code>	new <code>df</code> after creating desired testlets
<code>vars_metric</code>	new <code>vars_metric</code> after creating desired testlets
<code>testlet_strategy</code>	new <code>testlet_strategy</code> after creating desired testlets
<code>max_values</code>	new <code>max_values</code> after creating desired testlets

**See Also**

Other rasch functions: [rasch\\_DIF\(\)](#), [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_factor\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_mds\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_model\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_rawscore\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_rescale\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#)

Other children analysis functions: [rasch\\_df\\_nest\(\)](#), [rasch\\_drop\(\)](#), [rasch\\_mds\\_children\(\)](#), [rasch\\_model\\_children\(\)](#), [rasch\\_quality\\_children\\_print\(\)](#), [rasch\\_quality\\_children\(\)](#), [rasch\\_recode\(\)](#), [rasch\\_rescale\\_children\(\)](#), [rasch\\_split\\_age\(\)](#), [rasch\\_split\(\)](#)

---

 SepRel\_1.0.1

*Separation Reliability: Person Separation Reliability*


---

**Description**

Copied from eRm::SepRel(), v1.0-1: This function calculates the proportion of person variance that is not due to error. The concept of person separation reliability is very similar to reliability indices such as Cronbach's alpha.

**Usage**

```
SepRel_1.0.1(pobject)
```

**Arguments**

pobject            Object of class ppar (see ?eRm::person.parameter).

**Details**

See full documentation at <https://www.rdocumentation.org/packages/eRm/versions/1.0-1/topics/Separation%20Reliability>

**Value**

SepRel returns a list object of class eRm\_SepRel containing:

sep.rel	the person separation reliability,
SSD.PS	the squared standard deviation (i.e., total person variability),
MSE	the mean square measurement error (i.e., model error variance).

**Author(s)**

Original code by Adrian Brügger (<Adrian.Bruegger@imu.unibe.ch>), adapted by Marco J. Maier in package eRm v1.0-1

**References**

Wright, B.D., and Stone, M.H. (1999). Measurement essentials. Wide Range Inc., Wilmington. (<https://www.rasch.org/measess/me-all.pdf> 28Mb).

---

table_basicstats	<i>Compute basic statistics of the number of members per group per household</i>
------------------	--

---

**Description**

Compute basic statistics of the number of members per group per household

**Usage**

```
table_basicstats(df, hh_id, group_by_var)
```

**Arguments**

df	a data frame of household data where the rows represent members of the households in the sample
hh_id	string (length 1) indicating the name of the variable in df uniquely identifying households
group_by_var	string (length 1) to pass to group_by_at() with name of variable in df to group results by

**Value**

A tibble with rows for each level of group\_by\_var and "Total" and columns for the Mean (SD), Median and Range of the number of people in each group per household.

**Note**

Includes a call to tidyr::complete(), which causes the function to be a bit slow.

**See Also**

Other table functions: [table\\_unweightedpctn\(\)](#), [table\\_weightedpct\(\)](#)

**Examples**

```
#create dummy table of household data, where each row represents one member
df_hh <- data.frame(HHID = sample(
  x = 1:300,
  size = 1000,
  replace = TRUE
),
age_cat = ordered(sample(
  x = c("18-24", "25-39", "40-64", "64-100"),
  size = 1000,
  replace = TRUE
)))

table_basicstats(df_hh, "HHID", "age_cat")
```

---

table_unweightedpctn	<i>Compute unweighted percent and N for multiple variables, disaggregated</i>
----------------------	---

---

### Description

Compute unweighted percent and N for multiple variables, disaggregated

### Usage

```
table_unweightedpctn(
  df,
  vars_demo,
  group_by_var = NULL,
  spread_by_group_by_var = FALSE,
  group_by_var_sums_to_100 = FALSE,
  add_totals = FALSE
)
```

### Arguments

df	a data frame of individual survey data, where each row is an individual
vars_demo	a character vector of names of variables to calculate percent and N for
group_by_var	a string (length 1) with the name of the variable from df to disaggregate by
spread_by_group_by_var	logical determining whether to pass group_by_var to <code>tidyr::pivot_wider()</code> to give a wide-format tab. Default is FALSE.
group_by_var_sums_to_100	logical determining whether percentages sum to 100 along the margin of group_by_var, if applicable. Default is FALSE.
add_totals	logical determining whether to create total rows or columns (as appropriate) that demonstrate the margin that sums to 100. Default is FALSE.

### Value

A tibble with percent and N for each level of each variable in vars\_demo

### See Also

Other table functions: [table\\_basicstats\(\)](#), [table\\_weightedpct\(\)](#)

### Examples

```
table_unweightedpctn(df_adults, vars_demo = c("sex", "age_cat", "work_cat", "edu_cat"))
table_unweightedpctn(df_adults, vars_demo = c("sex", "age_cat", "work_cat", "edu_cat"),
  group_by_var = "disability_cat")
table_unweightedpctn(df_adults, vars_demo = c("sex", "age_cat", "work_cat", "edu_cat"),
  group_by_var = "disability_cat", spread_by_group_by_var = TRUE)
```

---

table_weightedpct	<i>Calculate table of percentages or N of response distribution for survey items, survey weighted, disaggregated</i>
-------------------	--

---

### Description

Calculate table of percentages or N of response distribution for survey items, survey weighted, disaggregated

### Usage

```
table_weightedpct(
  df,
  vars_ids,
  vars_strata,
  vars_weights,
  formula_vars,
  ...,
  formula_vars_levels = 0:1,
  by_vars = NULL,
  pct = TRUE,
  willfilter = NULL,
  add_totals = FALSE,
  spread_key = NULL,
  spread_value = "prop",
  arrange_vars = NULL,
  include_SE = FALSE
)
```

### Arguments

df	a data frame of individual survey data, where each row is an individual
vars_ids	a character vector of cluster ids, passed to <code>srvyr::as_survey_design()</code>
vars_strata	a character vector of strata ids, passed to <code>srvyr::as_survey_design()</code>
vars_weights	a character vector of survey weight ids, passed to <code>srvyr::as_survey_design()</code>
formula_vars	a character vector of variables to calculate the percentages of each level for
...	captures expressions to pass to <code>dplyr::filter()</code> or <code>dplyr::transmute()</code> , depending on the value of argument <code>willfilter</code> . See Details.
formula_vars_levels	a vector of the levels of the the <code>formula_vars</code>
by_vars	a character vector of variables to disaggregate results by. Default is NULL for no disaggregation. The columns listed must not include NAs.
pct	a logical variable indicating whether or not to calculate weighted percentages. Default is TRUE for weighted percentages. Set to FALSE for weighted N.



willfilter	a logical variable that tells the function whether or not to filter or transmute the data. Leave as default NULL to not filter or transmute. Set as TRUE to filter and FALSE to transmute. See Details.
add_totals	logical determining whether to create total rows or columns (as appropriate) that demonstrate the margin that sums to 100. Default is FALSE.
spread_key	a string with variable name to pass to names_from argument of tidyr::pivot_wider(). Default is NULL.
spread_value	a string with variable name to pass to values_from argument of tidyr::pivot_wider(). Default is "prop" (the column of percentages created within the function)
arrange_vars	a character vector with variables to pass to dplyr::arrange(). Default is NULL.
include_SE	a logical variable indicating whether to include the standard errors in the table. Default is FALSE. Currently does not work when adding totals, spreading or transmuting.

### Details

If willfilter is NULL, the table is not filtered or transmuted. If willfilter is TRUE, the table is filtered before it is spread or arranged. If willfilter is FALSE, the table is transmuted after the spread and/or arrange. "..." captures the non-standard evaluation expressions (NSE) to pass to dplyr::filter or dplyr::transmute().

The function performs the following actions with the table after results are calculated in the following order (if applicable): filter, add totals, spread, arrange, transmute

### Value

a tibble of weighted response percentages or N's

### See Also

See vignette("programming", package = "dplyr") for more about non-standard evaluation (NSE)

Other table functions: [table\\_basicstats\(\)](#), [table\\_unweightedpctn\(\)](#)

### Examples

```
table_weightedpct(df_adults,
  vars_ids = c("HHID", "PSU"),
  vars_strata = "strata",
  vars_weights = "weight",
  formula_vars = paste0("EF", 1:10),
  formula_vars_levels = 1:5,
  by_vars = "sex")
```

# Index

- \* **children analysis functions**
  - rasch\_df\_nest, 17
  - rasch\_drop, 20
  - rasch\_mds\_children, 24
  - rasch\_model\_children, 28
  - rasch\_quality\_children, 28
  - rasch\_quality\_children\_print, 29
  - rasch\_recode, 31
  - rasch\_rescale\_children, 33
  - rasch\_split, 34
  - rasch\_split\_age, 35
  - rasch\_testlet, 36
- \* **datasets**
  - df\_adults, 3
  - df\_children, 6
- \* **figure functions**
  - fig\_density, 8
  - fig\_dist, 9
  - fig\_LID, 11
  - fig\_poppyramid, 12
- \* **helper functions**
  - helper\_checkrow, 13
  - helper\_palette, 15
  - helper\_varslst, 17
- \* **rasch functions**
  - rasch\_df\_nest, 17
  - rasch\_DIF, 18
  - rasch\_drop, 20
  - rasch\_factor, 21
  - rasch\_mds, 22
  - rasch\_mds\_children, 24
  - rasch\_model, 26
  - rasch\_model\_children, 28
  - rasch\_quality\_children, 28
  - rasch\_quality\_children\_print, 29
  - rasch\_rawscore, 30
  - rasch\_recode, 31
  - rasch\_rescale, 32
  - rasch\_rescale\_children, 33
  - rasch\_split, 34
  - rasch\_split\_age, 35
  - rasch\_testlet, 36
- \* **table functions**
  - table\_basicstats, 38
  - table\_unweightedpctn, 39
  - table\_weightedpct, 40
- df\_adults, 3
- df\_children, 6
- fig\_density, 8, 10–12
- fig\_dist, 9, 9, 11, 12
- fig\_LID, 9, 10, 11, 12
- fig\_poppyramid, 9–11, 12
- helper\_checkrow, 13, 16, 17
- helper\_indicator, 14
- helper\_installation, 15
- helper\_palette, 13, 15, 17
- helper\_rowSums, 16
- helper\_varslst, 13, 16, 17
- hex, 16
- rasch\_df\_nest, 17, 19–21, 23, 26–35, 37
- rasch\_DIF, 18, 18, 20, 21, 23, 26–35, 37
- rasch\_drop, 18, 19, 20, 21, 23, 26–35, 37
- rasch\_factor, 18–20, 21, 23, 26–35, 37
- rasch\_mds, 18–21, 22, 26–35, 37
- rasch\_mds\_children, 18–21, 23, 24, 27–35, 37
- rasch\_model, 18–21, 23, 26, 26, 28–35, 37
- rasch\_model\_children, 18–21, 23, 26, 27, 28, 29–35, 37
- rasch\_quality\_children, 18–21, 23, 26–28, 28, 30–35, 37
- rasch\_quality\_children\_print, 18–21, 23, 26–29, 29, 31–35, 37
- rasch\_rawscore, 18–21, 23, 26–30, 30, 32–35, 37

rasch\_recode, *18–21, 23, 26–31, 31, 32–35, 37*  
rasch\_rescale, *18–21, 23, 26–32, 32, 33–35, 37*  
rasch\_rescale\_children, *18–21, 23, 26–32, 33, 34, 35, 37*  
rasch\_split, *18–21, 23, 26–33, 34, 35, 37*  
rasch\_split\_age, *18–21, 23, 26–34, 35, 37*  
rasch\_testlet, *18–21, 23, 26–35, 36*  
  
SepRel\_1.0.1, *37*  
  
table\_basicstats, *38, 39, 41*  
table\_unweightedpctn, *38, 39, 41*  
table\_weightedpct, *38, 39, 40*