Package 'eq5dsuite'

May 14, 2026

Type Package

Title Handling and Analysing EQ-5d Data

Version 2.0.0

Maintainer Kim Rand <krand@mathsinhealth.com>

Description The EQ-5D is a widely-
used standarized instrument for measuring Health Related Quality Of Life (HRQOL),
developed by the EuroQol group <https:
//euroqol.org/>. It assesses five dimensions; mobility, self-care,
usual activities, pain/discomfort, and anxiety/depression, using either a three-level (EQ-5D-
3L) or five-level (EQ-5D-5L) scale.
Scores from these dimensions are commonly converted into a single utility index using country-
specific value sets,
which are critical in clinical and economic evaluations of healthcare and in population health sur-
veys.
The eq5dsuite package enables users to calculate utility index values for the EQ-5D instruments,
including crosswalk utilities using the original crosswalk devel-
oped by van Hout et al. (2012) <doi:10.1016/j.jval.2012.02.008>
(mapping EQ-5D-5L responses to EQ-5D-
3L index values), or the recently developed reverse crosswalk
by van Hout et al. (2021) <doi:10.1016/j.jval.2021.03.009> (mapping EQ-5D-3L responses
to EQ-5D-5L index values). Users are allowed to add and/or remove user-defined value sets.
Additionally, the package provides tools to analyze EQ-5D data according to the recommended
guidelines outlined in ``Methods for Analyzing and Reporting EQ-
5D data'' by Devlin et al. (2020) <doi:10.1007/978-3-030-47622-9>.

License GPL (>= 2)

Encoding UTF-8

LazyData true

RoxygenNote 8.0.0

Depends R (>= 3.5)

Imports curl, ggplot2, moments, RColorBrewer, rlang, scales, rappdirs

Suggests shiny, DT, bslib, readxl, spelling, testthat (>= 3.0.0),
withr, knitr, rmarkdown

VignetteBuilder knitr

Config/testthat/edition 3

Language en-US

NeedsCompilation no

Author Kim Rand [aut, cre] (ORCID: <https://orcid.org/0000-0001-7692-4099>),
Oliver Rivero-Arias [aut] (ORCID:
<https://orcid.org/0000-0003-2233-6544>),
Iryna Schlackow [aut] (ORCID: <https://orcid.org/0000-0002-4154-1431>),
Anabel Estévez-Carrillo [aut] (ORCID:
<https://orcid.org/0000-0001-8778-5055>)

Repository CRAN

Date/Publication 2026-05-14 10:30:02 UTC

Contents

`eq5dsuite-package` *eq5dsuite: Standardised Tools for EQ-5D Analysis*

Description

The **eq5dsuite** package provides a comprehensive set of functions for calculating EQ-5D preference-based values and analysing EQ-5D data following the recommendations of Devlin et al. (2020).

The package supports three EQ-5D instruments:

- EQ-5D-3L
- EQ-5D-5L
- EQ-5D-Y-3L

Value calculation

Functions for calculating EQ-5D preference-based values:

- `eq5d` — Generic calculator for all instruments
- `eq5d3l` — EQ-5D-3L value calculation
- `eq5d5l` — EQ-5D-5L value calculation
- `eq5dy3l` — EQ-5D-Y-3L value calculation
- `eqxw` — Original crosswalk method (van Hout et al., 2012)
- `eqxwr` — Reverse crosswalk method (van Hout and Shaw, 2021)
- `eqxw_UK` — UK-specific NICE-recommended mapping (Hernandez Alava et al., 2023)

Value set management

Functions for managing country-specific value sets:

- `eqvs_display` — List all available value sets
- `eqvs_add` — Add a custom value set
- `eqvs_drop` — Remove a custom value set
- `eqvs_load` — Load a saved custom value set into the current session
- `update_value_sets` — Check for and install new value sets from the online repository

Profile analysis

Functions for analysing EQ-5D health state profiles, including cross-sectional summaries, longitudinal change analyses, and severity metrics:

- `eq5d_profile_level_summary` — Cross-sectional frequency summary by dimension
- `eq5d_profile_level_summary_by_group` — Stratified frequency summary by subgroup
- `eq5d_profile_top_states` — Most frequently observed health state profiles
- `eq5d_profile_change_summary` — Dimension-level frequencies at two time points

- eq5d_profile_dimension_change_table — Changes in levels per dimension as percentages of total and of type of change
- eq5d_profile_pchc_table — Paretian Classification of Health Change table
- eq5d_profile_pchc_with_no_problems_table — PCHC table with no-problems category
- eq5d_profile_pchc_by_group_plot — PCHC bar chart by subgroup
- eq5d_profile_better_dimensions_by_group_plot — Dimensions improved among Better patients
- eq5d_profile_worse_dimensions_by_group_plot — Dimensions worsened among Worse patients
- eq5d_profile_mixed_dimensions_by_group_plot — Dimensions changed among Mixed patients
- eq5d_profile_health_profile_grid — Health Profile Grid of individual transitions
- eq5d_profile_lss_utility_summary — Descriptive statistics by Level Sum Score
- eq5d_profile_lss_utility_plot — EQ-5D values plotted against LSS
- eq5d_profile_lfs_distribution — Level Frequency Score distribution table
- eq5d_profile_lfs_mean_utility — Mean EQ-5D values by LFS category
- eq5d_profile_lfs_utility_summary — Descriptive statistics by Level Frequency Score
- eq5d_profile_lfs_utility_plot — EQ-5D values plotted against LFS
- eq5d_profile_density_curve — Health State Density Curve

EQ-5D value analysis

Functions for analysing EQ-5D preference-based values across time points, subgroups, and population norms:

- eq5d_utility_summary — Descriptive statistics at each time point
- eq5d_utility_summary_by_group — Descriptive statistics stratified by subgroup
- eq5d_utility_norms_comparison — Comparison with population norms by age and sex
- eq5d_utility_over_time_plot — Mean values with confidence intervals over time
- eq5d_utility_by_group_plot — Mean values with confidence intervals by subgroup
- eq5d_utility_change_by_group_plot — Longitudinal change by subgroup
- eq5d_utility_distribution_plot — Histogram of EQ-5D value distribution
- eq5d_utility_vas_scatter_plot — Scatter plot of EQ-5D values against EQ-VAS

EQ-VAS analysis

Functions for analysing EQ-VAS self-rated health scores:

- eq5d_vas_summary — Descriptive statistics at each time point
- eq5d_vas_distribution_table — Frequency distribution in pre-defined ranges
- eq5d_vas_histogram — Histogram of EQ-VAS scores
- eq5d_vas_grouped_distribution_plot — Bar chart of grouped EQ-VAS frequencies

Helper functions

Utility functions for preparing EQ-5D data:

- `toEQ5Dindex` — Combine dimension columns into a 5-digit profile code
- `toEQ5Ddims` — Split a 5-digit profile code into separate dimension columns
- `make_all_EQ_indexes` — Generate all valid EQ-5D state codes as a vector
- `make_all_EQ_states` — Generate all valid EQ-5D states as a data frame
- `make_dummies` — Create dummy variables from EQ-5D dimension columns

Interactive application

A Shiny application providing point-and-click access to the same analytical workflow without requiring R coding:

- `run_app` — Launch the Shiny application

Cross-platform suite

eq5dsuite is part of a cross-platform suite also available in Stata and Excel:

- Full suite: https://github.com/MathsInHealth/eq5dsuite
- R package: https://github.com/MathsInHealth/eq5dsuite-r
- Value sets repository: https://github.com/MathsInHealth/eq5dsuite-value-sets

Author(s)

Anabel Estévez-Carrillo, Oliver Rivero-Arias, Iryna Schlackow, Kim Rand

Maintainer: Anabel Estévez-Carrillo <aestevez@mathsinhealth.com>

References

Devlin N, Parkin D, Janssen B (2020). *Methods for Analysing and Reporting EQ-5D Data*. Springer, Cham. doi:10.1007/9783030476229

van Hout B, Janssen MF, Feng YS, et al. (2012). Interim scoring for the EQ-5D-5L: mapping the EQ-5D-5L to EQ-5D-3L value sets. *Value in Health*, 15(5), 708–715. doi:10.1016/j.jval.2012.02.008

`.add_utility` *Add utility values to a data frame*

Description

This function adds utility values to a data frame based on a specified version of EQ-5D and a country name.

Usage

```
.add_utility(df, eq5d_version, country)
```

Arguments

`df`	A data frame containing the state data. The state must be included in the data frame as a character vector under the column named ‘state‘.
`eq5d_version`	A character string specifying the version of EQ-5D, i.e. 3L or 5L.
`country`	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

A data frame with an additional column named ‘utility‘ containing the calculated utility values. If the input country name is not found in the country_codes dataset, a list of available codes is printed, and subsequentyl an error message is displayed and the function stops.

Examples

```
df <- data.frame(state = c("11111", "11123", "32541"))
.add_utility(df, "5L", "DK")
```

`.check_uniqueness` *Check the uniqueness of groups This function takes a data frame ‘df‘ and a vector of columns ‘group_by‘, and checks whether the combinations of values in the columns specified by ‘group_by‘ are unique. If the combinations are not unique, a warning message is printed.*

Description

Check the uniqueness of groups This function takes a data frame ‘df‘ and a vector of columns ‘group_by‘, and checks whether the combinations of values in the columns specified by ‘group_by‘ are unique. If the combinations are not unique, a warning message is printed.

Usage

```
.check_uniqueness(df, group_by)
```

Arguments

df	A data frame.
group_by	A character vector of column names in ‘df‘ that specify the groups to check for uniqueness.

Value

No return value, called for side effects: it will stop with an error if any group combinations are not unique.

Examples

```
df <- data.frame(id = c(1, 1, 1, 1, 2, 2),
                 fu = rep(c("baseline", "follow-up"), 3),
                 value = rnorm(6))
.check_uniqueness(df, c("id", "fu"))
```

.EQxwrprob .EQxwrprob

Description

Takes a matrix of parameters for reverse crosswalk model, returns 243 x 25 matrix of state/level transition probabilities.

Usage

```
.EQxwrprob(par = NULL)
```

Arguments

par	Matrix of model parameters

Value

An 243 * 25 matrix with probabilities for state level transitions.

.freqtab *Helper function for frequency of levels by dimensions tables*

Description

Helper function for frequency of levels by dimensions tables

Usage

```
.freqtab(
  df,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL,
  eq5d_version = NULL,
  add_summary_problems_change = TRUE
)
```

Arguments

df	Data frame with the EQ-5D and follow-up columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_fu	Character string for the follow-up column. If NULL, no grouping is used, and the table reports for the total population.
levels_fu	Character vector containing the order of the values in the follow-up column. If NULL (default value), the levels will be ordered in the order of appearance in df.
eq5d_version	Version of the EQ-5D instrument
add_summary_problems_change	If set to false, the resulting dataframe does not include a row on problems change.

Value

Summary data frame.

.gen_colours *Generate colours for PCHC figures*

Description

This internal function generates a vector of colours based on the specified base colour. Currently only green and orange colours are implemented. The wrapper is used in Figures 2.2-2.4.

Usage

```
.gen_colours(col, n)
```

Arguments

col	A character string specifying the base colour. Only "green" or "orange" is accepted.
n	A positive integer specifying the number of colours to generate.

Value

A vector of colours generated based on the specified base colour and number of colours.

Examples

```
# generate 10 colours for base colour "green"
.gen_colours("green", 10)
# generate 7 colours for base colour "orange"
.gen_colours("orange", 7)
```

.getmode *Get the mode of a vector.*

Description

This function calculates the mode of a numeric or character vector. If there are multiple modes, the first one is returned. The code is taken from an R help page.

Usage

```
.getmode(v)
```

Arguments

v	A numeric or character vector.

Value

The mode of ‘v‘.

Examples

```
.getmode(c(1, 2, 3, 3))
.getmode(c("a", "b", "b", "c"))
```

.get_lfs *Calculate the Level Frequency Score (LFS)*

Description

This function calculates the Level Frequency Score (LFS) for a given EQ-5D state and a specified version of EQ-5D. If at least one domain contains a missing entry, the whole LFS is set to be NA.

Usage

```
.get_lfs(s, eq5d_version)
```

Arguments

s	A character vector representing the EQ-5D state, e.g. 11123.
eq5d_version	A character string specifying the version of EQ-5D, i.e. 3L or 5L.

Value

A character vector representing the calculated LFS.

Examples

```
.get_lfs("333", "3L") # returns 003
.get_lfs("333", "5L") # returns 00300
.get_lfs("12345", "5L") # returns 11111
```

.get_names *Replace NULL names with default values*

Description

This function takes in a list of parameters, which would be column names of the input data frame, and checks if they are null. Any nulls are replaced with default values, and the updated list of parameters is returned.

Usage

```
.get_names(df = NULL, ...)
```

Arguments

df	a data frame; only used/supplied if levels_fu needs to be defined
...	a list of parameters consisting of any/all of ‘names_eq5d‘, ‘name_fu‘, ‘levels_fu‘, ‘eq5d_version‘, and ‘name_vas‘.

Value

a list of parameters with null entries replaced with default values.

Examples

```
.get_names(names_eq5d = c("mo", "sc", "ua", "pd", "ad"))
.get_names(names_eq5d = NULL, eq5d_version = NULL, name_vas = NULL)
```

.modify_ggplot_theme *Modify ggplot2 theme*

Description

Modify ggplot2 theme

Usage

```
.modify_ggplot_theme(p)
```

Arguments

p	ggplot2 plot

Value

ggplot2 plot with modified theme

.pchc *Wrapper to determine Paretian Classification of Health Change*

Description

This internal function determines Paretian Classification of Health Change (PCHC) for each combination of the variables specified in the 'group_by' argument. It is used in the code for eq5d_profile_pchc_table, eq5d_profile_pchc_with_no_problems_table, eq5d_profile_dimension_change_table, and the eq5d_profile_*_by_group_plo functions. An EQ-5D health state is deemed to be 'better' than another if it is better on at least one dimension and is no worse on any other dimension. An EQ-5D health state is deemed to be 'worse' than another if it is worse in at least one dimension and is no better in any other dimension.

Usage

```
.pchc(df, level_fu_1, add_noprobs = FALSE)
```

Arguments

df	A data frame with EQ-5D states and follow-up variable. The dataset is assumed to be have been ordered correctly.
level_fu_1	Value of the first (i.e. earliest) follow-up. Would normally be defined as levels_fu[1].
add_noprobs	Logical value indicating whether to include a separate classification for those without problems (default is FALSE)

Value

A data frame with PCHC value for each combination of the grouping variables. If 'add_noprobs' is TRUE, a separate classification for those without problems is also included.

Examples

```
df <- data.frame(id = c(1, 1, 2, 2),
                 fu = c(1, 2, 1, 2),
                 mo = c(1, 1, 1, 1),
                 sc = c(1, 1, 5, 1),
                 ua = c(1, 1, 4, 3),
                 pd = c(1, 1, 1, 3),
                 ad = c(1, 1, 1, 1))
.pchc(df, level_fu_1 = 1, add_noprobs = TRUE)
```

.pchctab *.pchctab: Changes in health according to the PCHC (Paretian Classification of Health Change)*

Description

.pchctab: Changes in health according to the PCHC (Paretian Classification of Health Change)

Usage

```
.pchctab(
  df,
  name_id,
  name_groupvar,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL,
  add_noprobs = FALSE
)
```

Arguments

df	Data frame with the EQ-5D, grouping, id and follow-up columns
name_id	Character string for the patient id column
name_groupvar	Character string for the grouping column
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_fu	Character string for the follow-up column
levels_fu	Character vector containing the order of the values in the follow-up column. If NULL (default value), the levels will be ordered in the order of appearance in df.
add_noprobs	if set to TRUE, level corresponding to "no problems" will be added to the table

Value

Summary data frame

Examples

```
.pchctab(df = example_data,
  name_id = "id",
  name_groupvar = "procedure",
  name_fu = "time",
  levels_fu = c('Pre-op', 'Post-op')
)
```

.pchc_plot_by_dim *Wrapper to generate Paretian Classification of Health Change plot by dimension*

Description

This internal function plots Paretian Classification of Health Change (PCHC) by dimension. The input is a data frame containing the information to plot, and the plot will contain bars representing the proportion of the total data that falls into each dimension, stacked by covariate. The wrapper is used in Figures 2.2-2.4.

Usage

```
.pchc_plot_by_dim(plot_data, ylab, title, cols, text_rotate = FALSE)
```

Arguments

plot_data	A data frame containing information to plot, with columns for name (the dimensions to plot), p (the proportion of the total data falling into each dimension), and fu (the follow-up).
ylab	The label for the y-axis.

title	The plot title.
cols	A vector of colors to use for the bars.
text_rotate	A logical indicating whether to rotate the text labels for the bars.

Value

A ggplot object containing the PCHC plot.

Examples

```
df <- data.frame(
  name = rep(c("Dim1", "Dim2"), each = 2),
  p = c(0.6, 0.4, 0.7, 0.3),
  groupvar = rep(c("Group A", "Group B"), 2)
)
colors <- c("Group A" = "#1b9e77", "Group B" = "#d95f02")
.pchc_plot_by_dim(df, ylab = "Proportion", title = "Example Plot", cols = colors)
```

.prep_eq5d *Data checking/preparation: EQ-5D variables*

Description

This function prepares a data frame for analysis by extracting, processing, and adding columns for EQ-5D variables, including state, LSS (Level Sum Score), LFS (Level Frequency Score) and utility.

Usage

```
.prep_eq5d(
  df,
  names,
  add_state = FALSE,
  add_lss = FALSE,
  add_lfs = FALSE,
  add_utility = FALSE,
  eq5d_version = NULL,
  country = NULL
)
```

Arguments

df	a data frame of EQ-5D scores
names	character vector of length 5 with names of EQ-5D variables in the data frame. The variables should be in an integer format.
add_state	logical indicating whether the EQ-5D state should be added
add_lss	logical indicating whether the LSS (Level Sum Score) should be added

add_lfs	logical indicating whether the LFS (Level Frequency Score) should be added
add_utility	logical indicating whether the utility should be added
eq5d_version	character indicating the version of the EQ-5D questionnaire to use (either "3L" or "5L")
country	character indicating the country to retrieve the quality of life score for

Value

a modified data frame with EQ-5D domain columns renamed to default names, and, if necessary, with added columns for state, LSS, LFS, and/or utility. If any of the checks fail (e.g. EQ-5D columns are not in an integer format), an error message is displayed and the function is stopping.

Examples

```
set.seed(1234)
df <- data.frame(mo = sample(1:5, 3), sc = sample(1:5, 3),
  ua = sample(1:5, 3), pd = sample(1:5, 3), ad = sample(1:5, 3))
.prep_eq5d(df, names = c("mo", "sc", "ua", "pd", "ad"),
  add_state = TRUE, add_lss = TRUE)
.prep_eq5d(df, names = c("mo", "sc", "ua", "pd", "ad"),
  add_state = TRUE, add_lss = TRUE, add_lfs = TRUE, add_utility = TRUE,
  eq5d_version = "5L", country = "ES")
```

.prep_fu *Data checking/preparation: follow-up variable*

Description

This function prepares the follow-up (FU) variable for analysis by giving it a default name ('fu') and factorising

Usage

```
.prep_fu(df, name = NULL, levels = NULL)
```

Arguments

df	A data frame.
name	Column name in the data frame that contains follow-up information.
levels	Levels to factorise the FU variable into.

Value

A data frame with the follow-up variable renamed as "fu" and factorised.

Examples

```
df <- data.frame(id = c(1, 1, 2, 2),
  visit = c("baseline", "follow-up", "baseline", "follow-up"))
.prep_fu(df = df, name = "visit", levels = c("baseline", "follow-up"))
```

`.prep_vas` *Data checking/preparation: VAS variable*

Description

The function prepares the data for VAS (Visual Analogue Scale) analyses.

Usage

```
.prep_vas(df, name)
```

Arguments

`df`	A data frame.
`name`	Column name in the data frame that holds the VAS score. The column can only contain integers or NAs

Value

A modified data frame with the VAS score renamed to "vas". If any checks fail (e.g. column is not numeric), an error message is displayed and the function is stopping.

Examples

```
df <- data.frame(vas_score = c(20, 50, 80, NA, 100))
.prep_vas(df = df, name = "vas_score")
df <- data.frame(vas_score = c(20.5, 50, 80, NA, 100))
.prep_vas(df = df, name = "vas_score")
```

`.pstate3t5` *.pstate3t5*

Description

Takes a N x 25 matrix with probabilities per level/dimension, and creates an N * 3125 matrix with probabilities per state

Usage

```
.pstate3t5(PPP)
```

Arguments

PPP	N x 25 matrix with probabilities per level/dimension created by EQrxwprobs

Value

An N * 3125 matrix with probabilities per state

.pstate5t3 *.pstate5t3*

Description

Takes a 15 x 5 matrix with probabilities per level/dimension, and creates an 3125x243 matrix with probabilities per state

Usage

```
.pstate5t3(probs = .EQxwprob)
```

Arguments

probs	15 x 5 matrix with probabilities per level/dimension, typically saved in .EQxwprob

Value

An 3125x243 matrix with probabilities per state

.summary_cts_by_fu *Wrapper to summarise a continuous variable by follow-up (FU)*

Description

This function summarizes a continuous variable for each follow-up (FU) and calculates various statistics such as mean, standard deviation, median, mode, kurtosis, skewness, minimum, maximum, range, and number of observations. It also reports the total sample size and the number (and proportion) of missing values for each FU. The input ‘df‘ must contain an ordered FU variable and the continuous variable of interest. The name of the continuous variable must be specified using ‘name_v‘. The wrapper is used in Table 3.1 (for VAS) or Table 4.2 (for EQ-5D utility)

Usage

```
.summary_cts_by_fu(df, name_v)
```

Arguments

df	A data frame containing the FU and continuous variable of interest. The dataset must contain an ordered ‘fu‘ variable.
name_v	A character string with the name of the continuous variable in ‘df‘ to be summarised.

Value

Data frame with one row for each statistic and one column for each FU.

Examples

```
df <- data.frame(fu = c(1,1,2,2,3,3),
                 vas = c(7,8,9,NA,7,6))
.summary_cts_by_fu(df, name_v = "vas")
```

.summary_mean_ci *Wrapper to calculate summary mean with 95% confidence interval*

Description

This internal function calculates summary mean and 95% confidence interval of the utility variable, which can also be grouped. The function is used in Figures 4.2-4.4.

Usage

```
.summary_mean_ci(df, group_by)
```

Arguments

df	A data frame containing a ‘utility‘ column.
group_by	A character vector of column names to group by.

Value

A data frame with the mean, lower bound, and upper bound of the 95

Examples

```
df <- data.frame(group = c("A", "A", "B", "B"),
                 utility = c(0.5, 0.7, 0.8, 0.9))
.summary_mean_ci(df, group_by = "group")
```

.summary_table_2_1 *Wrapper for the repetitive code in function_table_2_1. Data frame summary*

Description

This internal function summarises a data frame by grouping it based on the variables specified in the 'group_by' argument and calculates the frequency of each group. The output is used in Table 2.1

Usage

```
.summary_table_2_1(df, group_by)
```

Arguments

df	A data frame
group_by	A character vector of variables in 'df' to group by. Should contain 'eq5d' and 'fu'.

Value

A summarised data frame with groups defined by 'eq5d' and 'fu' variables, the count of observations in each group, and the frequency of each group.

Examples

```
set.seed(1234)
df <- data.frame(eq5d = rep(rnorm(5), 2),
                 fu = rep(c(1, 0, 1, 0, 1), 2))
.summary_table_2_1(df, c("eq5d", "fu"))
```

.summary_table_4_3 *Summary wrapper for Table 4.3*

Description

This internal function creates a summary of the data frame for Table 4.3. It groups the data by the variables specified in 'group_by' and calculates various summary statistics.

Usage

```
.summary_table_4_3(df, group_by)
```

Arguments

df	A data frame.
group_by	A character vector of names of variables by which to group the data.

Value

A data frame with the summary statistics.

Examples

```
df <- data.frame(group = c("A", "A", "B", "B"),
                 utility = c(0.5, 0.7, 0.8, 0.9))
.summary_table_4_3(df, group_by = "group")
```

.summary_table_4_4 *Summary wrapper for Table 4.4*

Description

This internal function creates a summary of the data frame for Table 4.4. It groups the data by the variables specified in ‘group_by‘ and calculates various summary statistics.

Usage

```
.summary_table_4_4(df, group_by)
```

Arguments

df	A data frame.
group_by	A character vector of names of variables by which to group the data.

Value

A data frame with the summary statistics.

Examples

```
df <- data.frame(group = c("A", "A", "B", "B"),
                 utility = c(0.5, 0.7, 0.8, 0.9))
.summary_table_4_4(df, group_by = "group")
```

eq5d *eq5d*

Description

Get EQ-5D index values for the -3L, -5L, crosswalk (-3L value set applied to -5L health states), reverse crosswealk (-5L value set applied to -3L health states), and -Y-3L

Usage

```
eq5d(
  x,
  country = NULL,
  version = "5L",
  dim.names = c("mo", "sc", "ua", "pd", "ad")
)
```

Arguments

x	A vector of 5-digit EQ-5D-3L state indexes or a matrix/data.frame with columns corresponding to EQ-5D state dimensions
country	String vector indicating country names or ISO3166 Alpha 2 / 3 country codes.
version	String indicating which version to use. Options are '5L' (default), '3L', 'xw', 'xwr', and 'Y3L'.
dim.names	A vector of dimension names to identify dimension columns.

Value

A vector of values or data.frame with one column for each value set requested.

Examples

```
# US -3L value set
eq5d(c(11111, 12321, 32123, 33333), 'US', '3L')
# Danish and US -5L value sets applied to -3L descriptives, i.e. reverse crosswalk
eq5d(make_all_EQ_states('3L'), c('DK', 'US'), 'XWR')
# US -5L value set
eq5d(c(11111, 12321, 32153, 55555), 'US', '5L')
```

eq5d3l *eq5d3l*

Description

Get EQ-5D-3L index values from individual responses to the five dimensions of the EQ-5D-3L.

Usage

```
eq5d3l(x, country = NULL, dim.names = c("mo", "sc", "ua", "pd", "ad"))
```

Arguments

x	A vector of 5-digit EQ-5D-3L state indexes, or a matrix/data.frame with columns corresponding to the EQ-5D-3L dimensions.
country	String vector indicating country names or ISO3166 Alpha 2 / 3 country codes.
dim.names	A character vector specifying the names of the EQ-5D-3L dimensions. Default is ‘c("mo", "sc", "ua", "pd", "ad")‘.

Value

A numeric vector of EQ-5D-3L values, or a data.frame with one column for each requested value set.

Examples

```
# Example 1: utility values from EQ-5D-3L profile codes
eq5d3l(c(11111, 12321, 32123, 33333), country = "US")

# Example 2: request multiple value sets
eq5d3l(make_all_EQ_states("3L"), country = c("DK", "CA"))

# Example 3: use a data.frame with dimension columns
df3l <- data.frame(
  mo = c(1, 2, 3),
  sc = c(1, 2, 2),
  ua = c(1, 3, 1),
  pd = c(2, 2, 3),
  ad = c(1, 1, 2)
)
eq5d3l(df3l, country = "US")

# Example 4: use custom dimension column names
df3l_named <- data.frame(
  mobility = c(1, 2, 3),
  self_care = c(1, 2, 2),
  usual_activities = c(1, 3, 1),
  pain_discomfort = c(2, 2, 3),
  anxiety_depression = c(1, 1, 2)
```

```
)
eq5d3l(
  df3l_named,
  country = "US",
  dim.names = c(
    "mobility", "self_care", "usual_activities",
    "pain_discomfort", "anxiety_depression"
  )
)
```

---

eq5d5l *eq5d5l*

Description

Get EQ-5D-5L index values from individual responses to the five dimensions of the EQ-5D-5L.

Usage

```
eq5d5l(x, country = NULL, dim.names = c("mo", "sc", "ua", "pd", "ad"))
```

Arguments

x	A vector of 5-digit EQ-5D-5L state indexes, or a matrix/data.frame with columns corresponding to the EQ-5D-5L dimensions.
country	String vector indicating country names or ISO3166 Alpha 2 / 3 country codes.
dim.names	A character vector specifying the names of the EQ-5D-5L dimensions. Default is ‘c("mo", "sc", "ua", "pd", "ad")‘.

Value

A numeric vector of EQ-5D-5L values, or a data.frame with one column for each requested value set.

Examples

```
# Example 1: utility values from EQ-5D-5L profile codes
eq5d5l(c(11111, 12321, 32423, 55555), country = "IT")

# Example 2: request multiple value sets
eq5d5l(make_all_EQ_states("5L"), country = c("ES", "DE"))

# Example 3: use a data.frame with dimension columns
df5l <- data.frame(
  mo = c(1, 2, 5),
  sc = c(1, 2, 4),
  ua = c(1, 3, 3),
  pd = c(2, 4, 2),
  ad = c(1, 5, 1)
```

```
)
eq5d5l(df5l, country = "ES")

# Example 4: use custom dimension column names from a real-world style dataset
df5l_named <- data.frame(
  mobility = c(1, 5, 3),
  self_care = c(2, 4, 2),
  usual_activities = c(3, 3, 1),
  pain_discomfort = c(4, 2, 2),
  anxiety_depression = c(5, 1, 3)
)
eq5d5l(
  df5l_named,
  country = "ES",
  dim.names = c(
    "mobility", "self_care", "usual_activities",
    "pain_discomfort", "anxiety_depression"
  )
)
```

`eq5dy3l` *eq5dy3l*

Description

Get EQ-5D-Y-3L index values from individual responses to the five dimensions of the EQ-5D-Y-3L.

Usage

```
eq5dy3l(x, country = NULL, dim.names = c("mo", "sc", "ua", "pd", "ad"))
```

Arguments

`x`	A vector of 5-digit EQ-5D-Y-3L state indexes, or a matrix/data.frame with columns corresponding to the EQ-5D-Y-3L dimensions.
`country`	String vector indicating country names or ISO3166 Alpha 2 / 3 country codes.
`dim.names`	A character vector specifying the names of the EQ-5D-Y-3L dimensions. Default is ‘c("mo", "sc", "ua", "pd", "ad")‘.

Value

A numeric vector of EQ-5D-Y-3L values, or a data.frame with one column for each requested value set.

Examples

```
# Example 1: utility values from EQ-5D-Y-3L profile codes
eq5dy3l(x = c(11111, 12321, 33333), country = "SI")

# Example 2: request multiple value sets
eq5dy3l(make_all_EQ_states("3L"), country = c("ES", "DE"))

# Example 3: use a data.frame with dimension columns
dfy3l <- data.frame(
  mo = c(1, 2, 3),
  sc = c(1, 1, 2),
  ua = c(1, 2, 3),
  pd = c(2, 2, 3),
  ad = c(1, 3, 2)
)
eq5dy3l(dfy3l, country = "SI")

# Example 4: use custom dimension column names
dfy3l_named <- data.frame(
  mobility = c(1, 2, 3),
  self_care = c(1, 1, 2),
  usual_activities = c(1, 2, 3),
  pain_discomfort = c(2, 2, 3),
  anxiety_depression = c(1, 3, 2)
)
eq5dy3l(
  dfy3l_named,
  country = "SI",
  dim.names = c(
    "mobility", "self_care", "usual_activities",
    "pain_discomfort", "anxiety_depression"
  )
)
```

`eq5d_profile_better_dimensions_by_group_plot`

eq5d_profile_better_dimensions_by_group_plot: Percentage of Respondents Who Improved in Each EQ-5D Dimension, by Group This function calculates how many respondents improved in each dimension between two time points and summarizes the results for each group. The, it prodcuces a dimension-focused chart illustrating improvement percentages by dimension.

Description

eq5d_profile_better_dimensions_by_group_plot: Percentage of Respondents Who Improved in Each EQ-5D Dimension, by Group This function calculates how many respondents improved in each dimension between two time points and summarizes the results for each group. The, it prodcuces a dimension-focused chart illustrating improvement percentages by dimension.

Usage

```
eq5d_profile_better_dimensions_by_group_plot(
  df,
  name_id,
  name_groupvar = NULL,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL
)
```

Arguments

df	Data frame containing EQ-5D columns, a grouping variable, an ID column, and a follow-up column
name_id	Character string for the patient ID column
name_groupvar	Character string for the grouping column (e.g. procedure). If NULL (default), the analysis is performed on the full population.
names_eq5d	Character vector of EQ-5D dimension names
name_fu	Character string for the follow-up column
levels_fu	Character vector of length 2, specifying the order of the follow-up levels (e.g. c("Pre-op","Post-op"))

Value

A list containing:

plot_data	A data frame of improvements by group and dimension
p	A ggplot2 object produced by '.pchc_plot_by_dim()'

Examples

```
result <- eq5d_profile_better_dimensions_by_group_plot(
  df = example_data,
  name_id = "id",
  name_groupvar = "procedure",
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  name_fu = "time",
  levels_fu = c("Pre-op","Post-op")
)
result$p
result$plot_data
```

eq5d_profile_change_summary *eq5d_profile_change_summary: Frequency of levels by dimensions, by follow-up*

Description

eq5d_profile_change_summary: Frequency of levels by dimensions, by follow-up

Usage

```
eq5d_profile_change_summary(
  df,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL,
  eq5d_version = NULL
)
```

Arguments

df	Data frame with the EQ-5D and follow-up columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_fu	Character string for the follow-up column. If NULL, the function will check if there is a column named "follow-up" or "fu", in which case the first of those will be used.
levels_fu	Character vector containing the order of the values in the follow-up column. If NULL (default value), the levels will be ordered in the order of appearance in df.
eq5d_version	Version of the EQ-5D instrument

Value

Summary data frame.

Examples

```
eq5d_profile_change_summary(
  df = example_data,
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  name_fu = "time",
  levels_fu = c("Pre-op" , "Post-op"),
  eq5d_version = "3L"
)
```

`eq5d_profile_density_curve` *eq5d_profile_density_curve: Generate a Health State Density Curve (HSDC) for EQ-5D Data*

Description

This function calculates and plots the Health State Density Curve (HSDC) for a given EQ-5D dataset. It concatenates dimension values to form health state profiles, filters out invalid states based on the specified EQ-5D version, then computes the cumulative distribution of profiles (profiles vs. observations). A diagonal reference line indicates a perfectly even distribution. The function also calculates the Health State Density Index (HSDI), representing how sharply the observed distribution deviates from the diagonal.

Usage

```
eq5d_profile_density_curve(df, names_eq5d, eq5d_version)
```

Arguments

`df`	Data frame with the EQ-5D columns
`names_eq5d`	Character vector of column names for the EQ-5D dimensions
`eq5d_version`	Version of the EQ-5D instrument

Value

A list containing:

`plot_data`	A data frame with the cumulative distribution of profiles
`p`	A ggplot2 object showing the Health State Density Index

Examples

```
figure <- eq5d_profile_density_curve(
            df = example_data,
            names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
            eq5d_version = "3L"
          )
figure$plot_data
figure$p
```

`eq5d_profile_dimension_change_table` *eq5d_profile_dimension_change_table: Changes in levels in each dimension, percentages of total and of type of change*

Description

eq5d_profile_dimension_change_table: Changes in levels in each dimension, percentages of total and of type of change

Usage

```
eq5d_profile_dimension_change_table(
  df,
  name_id,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL
)
```

Arguments

`df`	Data frame with the EQ-5D, id and follow-up columns
`name_id`	Character string for the patient id column
`names_eq5d`	Character vector of column names for the EQ-5D dimensions
`name_fu`	Character string for the follow-up column
`levels_fu`	Character vector containing the order of the values in the follow-up column. If NULL (default value), the levels will be ordered in the order of appearance in df.

Value

Summary data frame

Examples

```
eq5d_profile_dimension_change_table(
  df = example_data,
  name_id = "id",
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  name_fu = "time",
  levels_fu = c("Pre-op" , "Post-op")
)
```

`eq5d_profile_health_profile_grid` *eq5d_profile_health_profile_grid: Health Profile Grid (HPG) for Two Time Points*

Description

This function creates a Health Profile Grid (HPG) for EQ-5D data, plotting each individual's change in health states (ranked from best to worst) between two time points. A diagonal reference line indicates no change; points above the line reflect improvement, and points below indicate deterioration.

Usage

```
eq5d_profile_health_profile_grid(
  df,
  names_eq5d,
  name_fu,
  levels_fu = NULL,
  name_id,
  eq5d_version,
  country
)
```

Arguments

`df`	A data frame containing EQ-5D columns, a grouping variable, an ID column, and a follow-up column
`names_eq5d`	A character vector of EQ-5D dimension names
`name_fu`	A character string for the follow-up column
`levels_fu`	A character vector of length 2, specifying the order of the follow-up levels (e.g., c("Pre-op","Post-op"))
`name_id`	A character string for the patient ID column
`eq5d_version`	Version of the EQ-5D instrument
`country`	A character string representing the name of the country.

Value

A list with components:

`plot_data`	The plot data with ranks and classification.
`p`	A `ggplot2` object displaying the HPG scatter plot.

Examples

```
tmp <- eq5d_profile_health_profile_grid(
           df = example_data,
           names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
           name_fu = "time",
           levels_fu = c("Pre-op", "Post-op"),
           name_id = "id",
           eq5d_version = "3L",
           country = "UK"
       )
```

---

eq5d_profile_level_summary

eq5d_profile_level_summary: Frequency of levels by dimensions, cross-sectional

---

Description

eq5d_profile_level_summary: Frequency of levels by dimensions, cross-sectional

Usage

```
eq5d_profile_level_summary(df, names_eq5d = NULL, eq5d_version = NULL)
```

Arguments

df	Data frame with the EQ-5D and follow-up columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
eq5d_version	Version of the EQ-5D instrument

Value

Summary data frame.

Examples

```
eq5d_profile_level_summary(
 df = example_data[example_data$time == "Pre-op",],
 names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
 eq5d_version = "3L"
)
```

eq5d_profile_level_summary_by_group *eq5d_profile_level_summary_by_group: Frequency of levels by dimensions, separated by category*

Description

eq5d_profile_level_summary_by_group: Frequency of levels by dimensions, separated by category

Usage

```
eq5d_profile_level_summary_by_group(
  df,
  names_eq5d = NULL,
  name_cat = NULL,
  levels_cat = NULL,
  eq5d_version = NULL
)
```

Arguments

df	Data frame with the EQ-5D and follow-up columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_cat	Character string for the category column. If NULL, no grouping is used, and the table reports for the total population, i.e. equal to table 1.1.1.
levels_cat	Character vector containing the order of the values in the category column, if the wish is to have these presented in a particular order. If NULL (default value), unless the variable is a factor, the levels will be ordered in the order of appearance in df.
eq5d_version	Version of the EQ-5D instrument

Value

Summary data frame.

Examples

```
eq5d_profile_level_summary_by_group(
 df = example_data[example_data$time == "Pre-op",],
 names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
 name_cat = "procedure",
 levels_cat = c("Hip Replacement", "Knee Replacement"),
 eq5d_version = "3L"
)
```

`eq5d_profile_lfs_distribution`

eq5d_profile_lfs_distribution: Distribution of the EQ-5D states by LFS (Level Frequency Score)

Description

eq5d_profile_lfs_distribution: Distribution of the EQ-5D states by LFS (Level Frequency Score)

Usage

```
eq5d_profile_lfs_distribution(df, names_eq5d = NULL, eq5d_version = NULL)
```

Arguments

`df`	Data frame with the EQ-5D columns
`names_eq5d`	Character vector of column names for the EQ-5D dimensions
`eq5d_version`	Version of the EQ-5D instrument

Value

Summary data frame

Examples

```
eq5d_profile_lfs_distribution(
  example_data,
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  eq5d_version = "3L"
)
```

`eq5d_profile_lfs_mean_utility`

eq5d_profile_lfs_mean_utility: Number of observations in the LFS (Level Frequency Score) according to the EQ-5D values

---

Description

eq5d_profile_lfs_mean_utility: Number of observations in the LFS (Level Frequency Score) according to the EQ-5D values

Usage

```
eq5d_profile_lfs_mean_utility(
  df,
  names_eq5d = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

df	Data frame with the EQ-5D columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
eq5d_version	Version of the EQ-5D instrument
country	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary data frame

Examples

```
eq5d_profile_lfs_mean_utility(
  example_data,
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  eq5d_version = "3L",
  country = "UK"
)
```

---

eq5d_profile_lfs_utility_plot

eq5d_profile_lfs_utility_plot: EQ-5D values plotted against LFS

Description

eq5d_profile_lfs_utility_plot: EQ-5D values plotted against LFS

Usage

```
eq5d_profile_lfs_utility_plot(
  df,
  names_eq5d = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

df	Data frame with the EQ-5D columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
eq5d_version	Version of the EQ-5D instrument
country	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary plot and data used for plotting

Examples

```
tmp <- eq5d_profile_lfs_utility_plot(
 example_data,
 names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
 eq5d_version = "3L",
 country = "UK"
)
tmp$p
tmp$plot_data
```

---

eq5d_profile_lfs_utility_summary

eq5d_profile_lfs_utility_summary: Summary statistics of EQ-5D values by LFS (Level Frequency Score)

Description

eq5d_profile_lfs_utility_summary: Summary statistics of EQ-5D values by LFS (Level Frequency Score)

Usage

```
eq5d_profile_lfs_utility_summary(
  df,
  names_eq5d = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

df	Data frame with the EQ-5D columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
eq5d_version	Version of the EQ-5D instrument
country	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary data frame

Examples

```
eq5d_profile_lfs_utility_summary(
  example_data,
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  eq5d_version = "3L",
  country = "UK"
)
```

---

eq5d_profile_lss_utility_plot

eq5d_profile_lss_utility_plot: EQ-5D values plotted against LSS

Description

eq5d_profile_lss_utility_plot: EQ-5D values plotted against LSS

Usage

```
eq5d_profile_lss_utility_plot(
  df,
  names_eq5d = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

df	Data frame with the EQ-5D columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
eq5d_version	Version of the EQ-5D instrument
country	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary plot and data used for plotting

Examples

```
df <- data.frame(make_all_EQ_states(version = "5L"))
tmp <- eq5d_profile_lss_utility_plot(
 df,
 names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
 eq5d_version = "5L",
 country = "US"
)
tmp$p
tmp$plot_data
```

`eq5d_profile_lss_utility_summary`

eq5d_profile_lss_utility_summary: Summary statistics for the EQ-5D values by all the different LSSs (Level Sum Scores)

Description

eq5d_profile_lss_utility_summary: Summary statistics for the EQ-5D values by all the different LSSs (Level Sum Scores)

Usage

```
eq5d_profile_lss_utility_summary(
  df,
  names_eq5d = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

`df`	Data frame with the EQ-5D columns
`names_eq5d`	Character vector of column names for the EQ-5D dimensions
`eq5d_version`	Version of the EQ-5D instrument
`country`	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary data frame

Examples

```
df <- data.frame(make_all_EQ_states(version = "5L"))
eq5d_profile_lss_utility_summary(
  df,
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  eq5d_version = "3L",
  country = "US"
)
```

`eq5d_profile_mixed_dimensions_by_group_plot`

eq5d_profile_mixed_dimensions_by_group_plot: Percentage of Respondents Who Had a Mixed Change Overall, by Dimension Improved or Worsened, Grouped by Procedure (or Other Grouping)

Description

This function focuses on patients classified as having "Mixed change" overall (i.e., some dimensions improved, others worsened). It then examines which dimensions improved vs. worsened for each subject. Results are summarized by a grouping variable (e.g., procedure) and time points. The final output is a table plus a ggplot object.

Usage

```
eq5d_profile_mixed_dimensions_by_group_plot(
  df,
  name_id,
  name_groupvar = NULL,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL
)
```

Arguments

`df`	Data frame containing columns for EQ-5D dimensions, a grouping variable, a patient ID, and a follow-up variable
`name_id`	Character string indicating the patient ID column
`name_groupvar`	Character string for the grouping column (e.g. "procedure"). If NULL (default), the analysis is performed on the full population.
`names_eq5d`	Character vector naming the EQ-5D dimensions (e.g. c("mo","sc","ua","pd","ad"))
`name_fu`	Character string for the follow-up column (e.g. "time")
`levels_fu`	Character vector of length 2 specifying the time order (e.g. c("Pre-op","Post-op"))

Value

A list with two elements:

plot_data	A wide-format data frame of dimension-specific improvements/worsenings for "Mixed change"
p	A ggplot2 object showing a dimension-level bar chart from .pchc_plot_by_dim

Examples

```
result <- eq5d_profile_mixed_dimensions_by_group_plot(
  df = example_data,
  name_id = "id",
  name_groupvar = "procedure",
  names_eq5d = c("mo","sc","ua","pd","ad"),
  name_fu = "time",
  levels_fu = c("Pre-op","Post-op")
)
result$plot_data
result$p
```

---

eq5d_profile_pchc_by_group_plot

eq5d_profile_pchc_by_group_plot: Paretian Classification of Health Change (PCHC) by Group This function computes PCHC categories between two time points for each subject, stratifies them by a grouping variable, and produces a single bar chart with side-by-side bars showing the distribution of PCHC categories.

Description

eq5d_profile_pchc_by_group_plot: Paretian Classification of Health Change (PCHC) by Group This function computes PCHC categories between two time points for each subject, stratifies them by a grouping variable, and produces a single bar chart with side-by-side bars showing the distribution of PCHC categories.

Usage

```
eq5d_profile_pchc_by_group_plot(
  df,
  name_id,
  name_groupvar = NULL,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL
)
```

Arguments

df	Data frame containing EQ-5D dimensions, a grouping variable, patient ID, and follow-up columns
name_id	Character string for the patient ID column
name_groupvar	Character string for the grouping column (e.g., procedure). If NULL (default), the analysis is performed on the full population.
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_fu	Character string for the follow-up column
levels_fu	Character vector of length 2 indicating the order of follow-up time points (e.g., c("Pre-op", "Post-op"))

Value

A list with two elements:

plot_data	A tibble of PCHC percentages by group
p	A ggplot2 object showing a bar chart with side-by-side bars for each PCHC category

Examples

```
result <- eq5d_profile_pchc_by_group_plot(
  df = example_data,
  name_id = "id",
  name_groupvar = "procedure",
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  name_fu = "time",
  levels_fu = c("Pre-op", "Post-op")
)
result$p        # shows the plot
result$plot_data  # shows the summary table
```

---

eq5d_profile_pchc_table

eq5d_profile_pchc_table: Changes in health according to the PCHC (Paretian Classification of Health Change)

Description

eq5d_profile_pchc_table: Changes in health according to the PCHC (Paretian Classification of Health Change)

Usage

```
eq5d_profile_pchc_table(
  df,
  name_id,
  name_groupvar = NULL,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL
)
```

Arguments

df	Data frame with the EQ-5D, grouping, id and follow-up columns
name_id	Character string for the patient id column
name_groupvar	Character string for the grouping column. If NULL (default), the analysis is performed on the full population.
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_fu	Character string for the follow-up column
levels_fu	Character vector containing the order of the values in the follow-up column. If NULL (default value), the levels will be ordered in the order of appearance in df.

Value

Summary data frame

Examples

```
eq5d_profile_pchc_table(
  df = example_data,
  name_id = "id",
  name_groupvar = "procedure",
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  name_fu = "time",
  levels_fu = c("Pre-op" , "Post-op")
)
```

---

eq5d_profile_pchc_with_no_problems_table

eq5d_profile_pchc_with_no_problems_table: Changes in health according to the PCHC, taking account of those with no problems

---

Description

eq5d_profile_pchc_with_no_problems_table: Changes in health according to the PCHC, taking account of those with no problems

Usage

```
eq5d_profile_pchc_with_no_problems_table(
  df,
  name_id,
  name_groupvar = NULL,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL
)
```

Arguments

df	Data frame with the EQ-5D, grouping, id and follow-up columns
name_id	Character string for the patient id column
name_groupvar	Character string for the grouping column. If NULL (default), the analysis is performed on the full population.
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_fu	Character string for the follow-up column
levels_fu	Character vector containing the order of the values in the follow-up column. If NULL (default value), the levels will be ordered in the order of appearance in df.

Value

Summary data frame

Examples

```
eq5d_profile_pchc_with_no_problems_table(
  df = example_data,
  name_id = "id",
  name_groupvar = "procedure",
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  name_fu = "time",
  levels_fu = c("Pre-op" , "Post-op")
)
```

---

eq5d_profile_top_states

eq5d_profile_top_states: Prevalence of the 10 most frequently observed self-reported health states

---

Description

eq5d_profile_top_states: Prevalence of the 10 most frequently observed self-reported health states

Usage

```
eq5d_profile_top_states(df, names_eq5d = NULL, eq5d_version = NULL, n = 10)
```

Arguments

df	Data frame with the EQ-5D columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
eq5d_version	Version of the EQ-5D instrument
n	Number of most frequently observed states to display (default 10)

Value

Summary data frame

Examples

```
eq5d_profile_top_states(
  df = example_data[example_data$time == "Pre-op",],
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  eq5d_version = "3L",
  n = 10
)
```

`eq5d_profile_worse_dimensions_by_group_plot`

This function identifies respondents with a "Worsen" PCHC state (i.e., overall health state got worse between levels_fu[1] and levels_fu[2]), checks dimension-specific changes (e.g., mo_diff < 0), and summarizes by a grouping variable (e.g., procedure) and time points. It returns a data table and a ggplot object.

Description

This function identifies respondents with a "Worsen" PCHC state (i.e., overall health state got worse between levels_fu[1] and levels_fu[2]), checks dimension-specific changes (e.g., mo_diff < 0), and summarizes by a grouping variable (e.g., procedure) and time points. It returns a data table and a ggplot object.

Usage

```
eq5d_profile_worse_dimensions_by_group_plot(
  df,
  name_id,
  name_groupvar = NULL,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL
)
```

Arguments

df	A data frame containing EQ-5D columns, a grouping variable, an ID column, and a follow-up column
name_id	A character string for the patient ID column
name_groupvar	A character string for the grouping column (e.g., procedure). If NULL (default), the analysis is performed on the full population.
names_eq5d	A character vector of EQ-5D dimension names
name_fu	A character string for the follow-up column
levels_fu	A character vector of length 2, specifying the order of the follow-up levels (e.g., c("Pre-op","Post-op"))

Value

A list containing:

plot_data	A data frame of "Worsen" percentages by group and dimension
p	A ggplot2 object produced by '.pchc_plot_by_dim()'

Examples

```
result <- eq5d_profile_worse_dimensions_by_group_plot(
  df = example_data,
  name_id = "id",
  name_groupvar = "procedure",
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  name_fu = "time",
  levels_fu = c("Pre-op","Post-op")
)
result$p        # shows the plot
result$plot_data  # shows the summary table
```

---

eq5d_utility_by_group_plot

eq5d_utility_by_group_plot: Mean EQ-5D values and 95% confidence intervals: all vs by groupvar

---

Description

eq5d_utility_by_group_plot: Mean EQ-5D values and 95% confidence intervals: all vs by groupvar

Usage

```
eq5d_utility_by_group_plot(
  df,
  names_eq5d = NULL,
  name_groupvar = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

df	Data frame with the EQ-5D and grouping columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_groupvar	Character string for the grouping column. If NULL (default), the analysis is performed on the full population.
eq5d_version	Version of the EQ-5D instrument
country	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary plot and data used for plotting

Examples

```
tmp <- eq5d_utility_by_group_plot(
 example_data,
 names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
 name_groupvar = "procedure",
 eq5d_version = "3L",
 country = "UK"
)
tmp$p
tmp$plot_data
```

---

eq5d_utility_change_by_group_plot

eq5d_utility_change_by_group_plot: EQ-5D values: smoothed lines and confidence intervals by groupvar

---

Description

eq5d_utility_change_by_group_plot: EQ-5D values: smoothed lines and confidence intervals by groupvar

Usage

```
eq5d_utility_change_by_group_plot(
  df,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL,
  name_groupvar = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

df	Data frame with the EQ-5D, follow-up and grouping columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_fu	Character string for the follow-up column
levels_fu	Character vector containing the order of the values in the follow-up column. If NULL (default value), the levels will be ordered in the order of appearance in df.
name_groupvar	Character string for the grouping column. If NULL (default), the analysis is performed on the full population.
eq5d_version	Version of the EQ-5D instrument
country	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary plot and data used for plotting

Examples

```
tmp <- eq5d_utility_change_by_group_plot(
 example_data,
 names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
 name_fu = "time",
 levels_fu = c('Pre-op', 'Post-op'),
 name_groupvar = "procedure",
 eq5d_version = "3L",
 country = "UK"
)
tmp$p
tmp$plot_data
```

`eq5d_utility_distribution_plot` *eq5d_utility_distribution_plot: EQ-5D values: smoothed lines and confidence intervals by groupvar*

Description

eq5d_utility_distribution_plot: EQ-5D values: smoothed lines and confidence intervals by groupvar

Usage

```
eq5d_utility_distribution_plot(
  df,
  names_eq5d = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

`df`	Data frame with the EQ-5D columns
`names_eq5d`	Character vector of column names for the EQ-5D dimensions
`eq5d_version`	Version of the EQ-5D instrument
`country`	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary plot and data used for plotting

Examples

```
tmp <- eq5d_utility_distribution_plot(
 example_data,
 names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
 eq5d_version = "3L",
 country = "UK"
)
tmp$p
tmp$plot_data
```

eq5d_utility_norms_comparison
eq5d_utility_norms_comparison:EQ-5D values: by age and groupvar

Description

eq5d_utility_norms_comparison:EQ-5D values: by age and groupvar

Usage

```
eq5d_utility_norms_comparison(
  df,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL,
  name_groupvar = NULL,
  name_age,
  eq5d_version = NULL,
  country
)
```

Arguments

df	Data frame with the EQ-5D, age, follow-up and grouping columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_fu	Character string for the follow-up column
levels_fu	Character vector containing the order of the values in the follow-up column. If NULL (default value), the levels will be ordered in the order of appearance in df.
name_groupvar	Character string for the grouping column. If NULL (default), the analysis is performed on the full population.
name_age	Character string for the age column
eq5d_version	Version of the EQ-5D instrument
country	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary data frame

Examples

```
example_data$ageband <- factor(
  example_data$ageband,
  levels = c("20 to 29", "30 to 39", "40 to 49", "50 to 59", "60 to 69", "70 to 79", "80 to 89")
)
example_data <- example_data[example_data$gender %in% c("Male", "Female"),]
eq5d_utility_norms_comparison(
  example_data,
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  name_fu = "time",
  levels_fu = c('Pre-op', 'Post-op'),
  name_groupvar = "gender",
  name_age = "ageband",
  eq5d_version = "3L",
  country = "UK"
)
```

---

eq5d_utility_over_time_plot

eq5d_utility_over_time_plot: EQ-5D values by timepoints: mean values and 95% confidence intervals

---

Description

eq5d_utility_over_time_plot: EQ-5D values by timepoints: mean values and 95% confidence intervals

Usage

```
eq5d_utility_over_time_plot(
  df,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

df	Data frame with the VAS columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_fu	Character string for the follow-up column
levels_fu	Character vector containing the order of the values in the follow-up column. If NULL (default value), the levels will be ordered in the order of appearance in df.
eq5d_version	Version of the EQ-5D instrument

country	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary plot and data used for plotting

Examples

```
tmp <- eq5d_utility_over_time_plot(
 example_data,
 names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
 name_fu = "time",
 levels_fu = c('Pre-op', 'Post-op'),
 eq5d_version = "3L",
 country = "UK"
)
tmp$p
tmp$plot_data
```

---

eq5d_utility_summary *eq5d_utility_summary: EQ-5D values: by timepoints*

Description

eq5d_utility_summary: EQ-5D values: by timepoints

Usage

```
eq5d_utility_summary(
  df,
  names_eq5d = NULL,
  name_fu = NULL,
  levels_fu = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

df	Data frame with the EQ-5D and follow-up columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_fu	Character string for the follow-up column
levels_fu	Character vector containing the order of the values in the follow-up column. If NULL (default value), the levels will be ordered in the order of appearance in df.
eq5d_version	Version of the EQ-5D instrument
country	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary data frame

Examples

```
eq5d_utility_summary(
  example_data,
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  name_fu = "time",
  levels_fu = c('Pre-op', 'Post-op'),
  eq5d_version = "3L",
  country = "UK"
)
```

---

eq5d_utility_summary_by_group

eq5d_utility_summary_by_group:EQ-5D values: by groupvar

Description

eq5d_utility_summary_by_group:EQ-5D values: by groupvar

Usage

```
eq5d_utility_summary_by_group(
  df,
  names_eq5d = NULL,
  name_groupvar = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

df	Data frame with the EQ-5D, follow-up and grouping columns
names_eq5d	Character vector of column names for the EQ-5D dimensions
name_groupvar	Character string for the grouping column. If NULL (default), the analysis is performed on the full population.
eq5d_version	Version of the EQ-5D instrument
country	A character string representing the name of the country.

Value

Summary data frame

Examples

```
eq5d_utility_summary_by_group(
  example_data,
  names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
  name_groupvar = "procedure",
  eq5d_version = "3L",
  country = "UK"
)
```

`eq5d_utility_vas_scatter_plot` *eq5d_utility_vas_scatter_plot: EQ-5D values: smoothed lines and confidence intervals by groupvar*

Description

eq5d_utility_vas_scatter_plot: EQ-5D values: smoothed lines and confidence intervals by groupvar

Usage

```
eq5d_utility_vas_scatter_plot(
  df,
  names_eq5d = NULL,
  name_vas = NULL,
  eq5d_version = NULL,
  country
)
```

Arguments

`df`	Data frame with the EQ-5D columns
`names_eq5d`	Character vector of column names for the EQ-5D dimensions
`name_vas`	Character string for the VAS column
`eq5d_version`	Version of the EQ-5D instrument
`country`	A character string representing the name of the country. This could be in a 2-letter format, full name or short name, as specified in the country_codes datasets.

Value

Summary plot and data used for plotting

Examples

```
tmp <- eq5d_utility_vas_scatter_plot(
   example_data,
   names_eq5d = c("mo", "sc", "ua", "pd", "ad"),
   name_vas = "vas",
   eq5d_version = "3L",
   country = "UK"
 )
tmp$p
tmp$plot_data
```

`eq5d_vas_distribution_table`
eq5d_vas_distribution_table: EQ VAS Scores frequency of mid-points

---

Description

eq5d_vas_distribution_table: EQ VAS Scores frequency of mid-points

Usage

```
eq5d_vas_distribution_table(df, name_vas = NULL, add_na_total = TRUE)
```

Arguments

`df`	Data frame with the VAS column
`name_vas`	Character string for the VAS column
`add_na_total`	Logical, whether to add summary of the missing, and across the Total, data

Value

Summary data frame

Examples

```
eq5d_vas_distribution_table(
  example_data,
  name_vas = 'vas',
  add_na_total =  TRUE
  )
```

eq5d_vas_grouped_distribution_plot

eq5d_vas_grouped_distribution_plot: Mid-point EQ VAS scores

Description

eq5d_vas_grouped_distribution_plot: Mid-point EQ VAS scores

Usage

```
eq5d_vas_grouped_distribution_plot(df, name_vas = NULL)
```

Arguments

df	Data frame with the VAS column
name_vas	Character string for the VAS column

Value

Summary plot and data used for plotting

Examples

```
tmp <- eq5d_vas_grouped_distribution_plot(example_data, name_vas = 'vas')
tmp$p
tmp$plot_data
```

eq5d_vas_histogram

eq5d_vas_histogram: EQ VAS scores

Description

eq5d_vas_histogram: EQ VAS scores

Usage

```
eq5d_vas_histogram(df, name_vas = NULL)
```

Arguments

df	Data frame with the VAS column
name_vas	Character string for the VAS column

Value

Summary plot and data used for plotting

Examples

```
tmp <- eq5d_vas_histogram(example_data, name_vas = 'vas')
tmp$p
tmp$plot_data
```

`eq5d_vas_summary` *eq5d_vas_summary: EQ VAS Score by timepoints*

Description

eq5d_vas_summary: EQ VAS Score by timepoints

Usage

```
eq5d_vas_summary(df, name_vas = NULL, name_fu = NULL, levels_fu = NULL)
```

Arguments

`df`	Data frame with the VAS and the follow-up columns
`name_vas`	Character string for the VAS column
`name_fu`	Character string for the follow-up column
`levels_fu`	Character vector containing the order of the values in the follow-up column.

Value

Summary data frame

Examples

```
eq5d_vas_summary(
  example_data,
  name_vas = 'vas',
  name_fu = 'time',
  levels_fu = c('Pre-op', 'Post-op')
)
```

eqvs_add *eqvs_add*

Description

Add user-defined EQ-5D value set and corresponding crosswalk option.

Usage

```
eqvs_add(
  df,
  version = "5L",
  country = NULL,
  countryCode = NULL,
  VSCode = NULL,
  description = NULL,
  saveOption = 1,
  savePath = NULL
)
```

Arguments

df	A data.frame or file name pointing to csv file. The contents of the data.frame or csv file should be exactly two columns: state, containing a list of all 3125 (for 5L) or 243 (for 3L) EQ-5D health state vectors, and a column of corresponding utility values, with a suitable name.
version	Version of the EQ-5D instrument. Can take values 5L (default) or 3L.
country	Optional string. If not NULL, will be used as a country description for the user-defined value set.
countryCode	Optional string. If not NULL, will be used as the two-digit code for the value set. Must be different from any existing national value set code.
VSCode	Optional string. If not NULL, will be used as the three-digit code for the value set. Must be different from any existing national value set code.
description	Optional string. If not NULL, will be used as a descriptive text for the user-defined value set.
saveOption	Integer indicating how the cache data should be saved. 1: Do not save (default), 2: Save in package folder, 3: Save in another path.
savePath	A path where the cache data should be saved when ‘saveOption‘ is 3. Please use ‘eqvs_load‘ to load it in your next session.

Value

True/False, indicating success or error.

Examples

```
# make nonsense value set
new_df <- data.frame(state = make_all_EQ_indexes(), TEST = runif(3125))
# Add as value set for Fantasia
eqvs_add(
   new_df,
   version = "5L",
   country = 'Fantasia',
   countryCode = "MyCountry",
   VSCode = "FAN",
   saveOption = 1
)
eq5d5l(55555,country = "FAN")
```

`eqvs_display` *eqvs_display*

Description

Display available value sets, which can also be used as (reverse) crosswalks. Built-in value sets are shown first, followed by any user-defined value sets added via `eqvs_add()`.

Usage

```
eqvs_display(version = "5L", return_df = FALSE, show_citation = FALSE)
```

Arguments

`version`	Version of the EQ-5D instrument. One of `"3L"`, `"5L"` (default), or `"Y3L"`.
`return_df`	Logical. If `TRUE`, returns a data.frame with all columns (including `citation` if present). Defaults to `FALSE`.
`show_citation`	Logical. If `TRUE`, prints the full AMA citation for each value set after the summary table. Defaults to `FALSE`.

Value

`NULL` invisibly (default), or a data.frame when `return_df = TRUE`.

Examples

```
# Display available EQ-5D-5L value sets.
eqvs_display(version = "5L")
```

eqvs_drop *eqvs_drop*

Description

Drop user-defined EQ-5D value set to reverse crosswalk options.

Usage

```
eqvs_drop(country = NULL, version = "5L", saveOption = 1, savePath = NULL)
```

Arguments

country	Optional string. If NULL, a list of current user-defined value sets will be provided for selection. If set, and matching an existing user-defined value set, a prompt will be given as to whether the value set should be deleted.
version	Version of the EQ-5D instrument. Can take values 5L (default) or 3L.
saveOption	Integer indicating how the cache data should be saved. 1: Do not save (default), 2: Save in package folder, 3: Save in another path.
savePath	A path where the cache data should be saved when ‘saveOption‘ is 3. Please use ‘eqvs_load‘ to load it in your next session.

Value

True/False, indicating success or error.

Examples

```
# make nonsense value set
new_df <- data.frame(state = make_all_EQ_indexes(), TEST = runif(3125))
# Add as value set for Fantasia
eqvs_add(
 new_df,
 version = "5L",
 country = 'Fantasia',
 countryCode = "MyCountry",
 VSCode = "FAN",
 saveOption = 1
)
# Test the new value set
eq5d5l(55555,country = "FAN")
# Drop value set for Fantasia
eqvs_drop(country = 'FAN', saveOption = 1)
```

eqvs_load *eqvs_load*

Description

Load cache data from a specified path.

Usage

```
eqvs_load(loadPath)
```

Arguments

loadPath	The path from which to load the cache data.

Value

TRUE if loading is successful, FALSE otherwise.

eqxw *eqxw*

Description

Get crosswalk values

Usage

```
eqxw(x, country = NULL, dim.names = c("mo", "sc", "ua", "pd", "ad"))
```

Arguments

x	A vector of 5-digit EQ-5D-5L state indexes or a matrix/data.frame with columns corresponding to EQ-5D state dimensions
country	String vector indicating country names or ISO3166 Alpha 2 / 3 country codes.
dim.names	A vector of dimension names to identify dimension columns

Value

A vector of reverse crosswalk values or data.frame with one column per reverse crosswalk set requested.

Examples

```
eqxw(c(11111, 12521, 32123, 55555), 'US')
eqxw(make_all_EQ_states('5L'), c('DK', 'US'))
```

eqxwr *eqxwr*

Description

Get reverse crosswalk values

Usage

```
eqxwr(x, country = NULL, dim.names = c("mo", "sc", "ua", "pd", "ad"))
```

Arguments

x	A vector of 5-digit EQ-5D-3L state indexes or a matrix/data.frame with columns corresponding to EQ-5D state dimensions
country	String vector indicating country names or ISO3166 Alpha 2 / 3 country codes.
dim.names	A vector of dimension names to identify dimension columns

Value

A vector of reverse crosswalk values or data.frame with one column per reverse crosswalk set requested.

Examples

```
eqxwr(c(11111, 12321, 32123, 33333), 'US')
eqxwr(make_all_EQ_states('3L'), c('DK', 'US'))
```

eqxw_UK *eqxw_UK*

Description

Crosswalks EQ-5D-5L responses to EQ-5D-3L utilities using NICE's mapping.

Usage

```
eqxw_UK(x, age, male, dim.names = c("mo", "sc", "ua", "pd", "ad"), bwidth = 0)
```

Arguments

x	A vector of 5-digit EQ-5D-5L states (domain scores) or a summary score.
age	A numeric vector or column name (if 'x' is a data frame). Can be either: (1) a numeric age between 18 and 100, which will be automatically grouped into NICE-defined age bands (18-35, 35-45, 45-55, 55-65, +65), or (2) a factor/character/numeric vector already representing the NICE age bands with values 1-5 indicating age bands (18-35, 35-45, 45-55, 55-65, +65).
male	A numeric vector (1=male, 0=female) or column name indicating gender.
dim.names	A vector of dimension names for EQ-5D states (default: c("mo", "sc", "ua", "pd", "ad")).
bwidth	Numeric. Bandwidth for kernel smoothing when using summary scores.

Value

A vector or data frame with crosswalked EQ-5D-3L utilities.

Examples

```
eqxw_UK(c(11111, 12345, 32423, 55555), age = c(30, 40, 55, 70), male = c(1, 0, 1, 0))
```

---

example_data *example_data*

Description

A dataset containing patient-level data in a long format.

Usage

```
data(example_data)
```

Format

A data frame with 10000 rows and 14 variables:

`id` double Patient id

`time` character Follow-up (Pre-op / Post-op)

`mo` double EQ-5D-5L Mobility dimension

`sc` double EQ-5D-5L Self-care dimension

`ua` double EQ-5D-5L Usual activities dimension

`pd` double EQ-5D-5L Pain / discomfort dimension

`ad` double EQ-5D-5L Anxiety/depression dimension

`vas` double Value of the VAS scale measurememnt

`providercode` character Provider code

`procedure` character Type of surgery

`year` character Year of intervention

`ageband` character Age in pre-defined ranges

`gender` character Patient's gender (Female / Male)

`make_all_EQ_indexes` *make_all_EQ_indexes*

Description

Make a vector containing all 5-digit EQ-5D indexes for -3L or -5L version.

Usage

```
make_all_EQ_indexes(
  version = "5L",
  dim.names = c("mo", "sc", "ua", "pd", "ad")
)
```

Arguments

`version`	Either "3L" or "5L", to signify whether 243 or 3125 states should be generated
`dim.names`	A vector of dimension names to be used as names for output columns.

Value

A vector with 5-digit state indexes for all 243 (-3L) or 3125 (-5L) EQ-5D health states

Examples

```
make_all_EQ_indexes('3L')
```

`make_all_EQ_states` *make_all_EQ_states*

Description

Make a data.frame with all health states defined by dimensions

Usage

```
make_all_EQ_states(
  version = "5L",
  dim.names = c("mo", "sc", "ua", "pd", "ad"),
  append_index = FALSE
)
```

Arguments

version	Either "3L" or "5L", to signify whether 243 or 3125 states should be generated
dim.names	A vector of dimension names to be used as names for output columns.
append_index	Boolean to indicate whether a column of 5-digit EQ-5D health state indexes should be added to output.

Value

A data.frame with 5 columns and 243 (-3L) or 3125 (-5L) health states

Examples

```
make_all_EQ_states('3L')
```

---

make_dummies *EQ_dummies*

Description

Make a data.frame of all EQ-5D dummies relevant for e.g. regression modeling.

Usage

```
make_dummies(
  df,
  version = "5L",
  dim.names = c("mo", "sc", "ua", "pd", "ad"),
  drop_level_1 = TRUE,
  add_intercept = FALSE,
  incremental = FALSE,
  prepend = NULL,
  append = NULL,
  return_df = TRUE
)
```

Arguments

df	data.frame containing EQ-5D health states.
version	Either "3L" or "5L", to signify EQ-5D instrument version
dim.names	A vector of dimension names to be used as names for output columns.
drop_level_1	If set to FALSE, dummies for level 1 will be included. Defaults to TRUE.
add_intercept	If set to TRUE, a column containing 1s will be appended. Defaults to FALSE.
incremental	If set to TRUE, incremental dummies will be produced (e.g. MO = 3 will give mo2 = 1, mo3 = 1). Defaults to FALSE.
prepend	Optional string to be prepended to column names.
append	Optional string to be appended to column names.
return_df	If set to TRUE, data.frame is returned, otherwise matrix. Defaults to TRUE.

Value

A data.frame of dummy variables

Examples

```
make_dummies(make_all_EQ_states('3L'), '3L')

make_dummies(df = make_all_EQ_states('3L'),
             version =  '3L',
             incremental = TRUE,
             add_intercept = TRUE,
             prepend = "d_")
```

---

run_app *Launch the eq5dsuite Shiny Application*

Description

Opens an interactive Shiny application for uploading, processing, analysing, and exporting EQ-5D data using the eq5dsuite package.

Usage

```
run_app(...)
```

Arguments

`...`	Additional arguments passed to `runApp`, such as `port` or `launch.browser`.

Value

Called for its side effect of launching a Shiny application. Returns invisibly.

Examples

```
## Not run:
  eq5dsuite::run_app()

## End(Not run)
```

toEQ5Ddims *toEQ5Ddims*

Description

Generate dimension vectors based on state index

Usage

```
toEQ5Ddims(x, dim.names = c("mo", "sc", "ua", "pd", "ad"))
```

Arguments

x	A vector of 5-digit EQ-5D state indexes.
dim.names	A vector of dimension names to be used as names for output columns.

Value

A data.frame with 5 columns, one for each EQ-5D dimension, with names from dim.names argument.

Examples

```
toEQ5Ddims(c(12345, 54321, 12321))
```

toEQ5Dindex *Convert EQ-5D dimension scores to a five-digit profile index*

Description

Convert EQ-5D dimension scores to a five-digit profile index

Usage

```
toEQ5Dindex(
  x,
  dim.names = c("mo", "sc", "ua", "pd", "ad"),
  na.rm = FALSE,
  quiet = FALSE
)
```

Arguments

x	A data.frame, matrix, or named numeric vector of EQ-5D dimension scores. Each dimension must contain integer values (typically 1–3 or 1–5).
dim.names	Character vector of length 5 giving the dimension names, in the conventional MO, SC, UA, PD, AD order.
na.rm	Logical. If FALSE (default), any NA in a row produces NA in the output. If TRUE, NA dimensions are treated as 0 — use with care, as this silently changes the index value.
quiet	Logical. Suppress informational messages about assumed column / name order. Default FALSE so existing scripts see the same messages; set TRUE inside pipelines.

Value

An integer vector with one element per row (data.frame/matrix input) or a single integer (vector input). Works inside `dplyr::mutate()` without `rowwise()`.

Examples

```
# Named vector — scalar usage unchanged
toEQ5Dindex(c(mo=1, sc=2, ua=3, pd=1, ad=2))
```

---

update_value_sets *Update eq5dsuite value sets*

Description

Connects to the eq5dsuite value sets repository and checks whether new value sets are available for any of the three EQ-5D instruments. Optionally drop or rename existing value sets before checking for updates.

Usage

```
update_value_sets(
  versions = c("3L", "5L", "Y3L"),
  ask = TRUE,
  drop = NULL,
  rename = NULL
)
```

Arguments

versions	Character vector. Which instruments to check. Defaults to c("3L", "5L", "Y3L").
ask	Logical. Whether to ask for confirmation before installing, dropping, or renaming. Defaults to TRUE.

drop	Character vector of VS_codes to remove, in the format "VS_code:version" e.g. c("NL:3L", "US:5L"). Defaults to NULL (no removals).
rename	Named character vector of renames in the format c("old_VS_code:version" = "new_VS_code") e.g. c("NL:3L" = "NL_2006"). Defaults to NULL (no renames).

Details

This function requires an internet connection for checking and installing new value sets. Drop and rename operations work offline.

Value

Invisibly returns a list with elements `checked`, `new`, `installed`, `dropped`, and `renamed`.

Examples

```
## Not run:
# Check for new value sets interactively
update_value_sets()

# Check specific instrument only
update_value_sets(versions = "5L")

# Drop a value set
update_value_sets(drop = "NL:3L")

# Rename a value set
update_value_sets(rename = c("NL:3L" = "NL_2006"))

# Drop and rename in one call
update_value_sets(
  drop   = "DE:3L",
  rename = c("NL:3L" = "NL_2006", "SI:3L" = "SI_TTO")
)

# Non-interactive update
update_value_sets(ask = FALSE)

## End(Not run)
```

Index