

Package ‘youthu’

August 8, 2021

Title Youth Outcomes to Health Utility

Version 0.0.0.9095

Description Tools for mapping measures routinely collected in youth mental health services to Quality Adjusted Life Years (QALYs). Part of the First Bounce model of primary youth mental health services. This development version of the youthu package has been made available as part of the process of testing and documenting the package. If you have any questions, please contact the authors (matthew.hamilton@orygen.org.au). The documentation for this package has been automatically generated by the ready4fun package and is therefore quite rudimentary. Further human authored documentation will follow in 2021.

License GPL-3 + file LICENSE

URL <https://ready4-dev.github.io/youthu/>,
<https://github.com/ready4-dev/youthu>, <https://www.ready4-dev.com/>

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.1.1

Imports assertthat,
BCEA,
boot,
dataverse (>= 0.3.9),
dplyr,
knitr,
knitrBootstrap,
lifecycle,
lubridate,
magrittr,
MatchIt,
methods,
purrr,
ready4fun (>= 0.0.0.9298),
ready4show (>= 0.0.0.9038),
ready4use (>= 0.0.0.9133),
rlang,
stats,

stringr,
 testthat,
 tibble,
 tidyr,
 tidyselect,
 truncnorm,
 TTU (\geq 0.0.0.9317),
 utils,
 youthvars (\geq 0.0.0.9064)

VignetteBuilder knitr

Depends R (\geq 2.10)

Suggests rmarkdown

Remotes ready4-dev/ready4show,
 ready4-dev/ready4use,
 ready4-dev/youthvars,
 ready4-dev/TTU,
 iqss/dataverse-client-r,
 ready4-dev/ready4fun

R topics documented:

youthu-package	3
abbreviations_lup	4
add_change_in_ds_var	4
add_costs_by_tmpt	5
add_costs_from_gamma_dstr	6
add_dates_from_dstr	6
add_diffs_by_group_and_tmpt	7
add_qalys	9
add_qalys_to_ds	10
add_utl_predn	11
fns_dmt_tb	12
fn_type_lup_tb	12
get_dv_dss_mdls_smrys	13
get_dv_ds_publication	14
get_dv_mdls_smrys	14
get_filtered_ttus_dss	15
get_mdls_lup	15
get_mdls_ctlg_url	16
get_mdls_ds_url	17
get_mdls_from_dv	17
get_mdls_metadata	18
get_mdls_smrys	19
get_model	19
get_predictors_lup	20
get_tmfn_from_lup	21
get_ttus_ds_smrys	21
get_ttus_dv_dss	22
get_ttus_dv_predrs	22
make_balanced_fake_ds	23

make_costs_vec_from_gamma_dstr	24
make_cst_efns_smry	24
make_fake_ds_one	25
make_fake_ds_two	26
make_fake_trial_ds	26
make_hlth_ec_smry	27
make_matched_ds	28
make_matched_ds_spine	29
make_predn_metadata_ls	30
make_sngl_grp_ds	31
predict_from_md1_coefs	32
transform_ds_for_cmprsn	32
update_col_with_diff	33
update_multpl_cols_with_diffs	34

Index 35

youthu-package	<i>youthu: Youth Outcomes to Health Utility</i>
----------------	---

Description

Tools for mapping measures routinely collected in youth mental health services to Quality Adjusted Life Years (QALYs). Part of the First Bounce model of primary youth mental health services. This development version of the youthu package has been made available as part of the process of testing and documenting the package. If you have any questions, please contact the authors (matthew.hamilton@orygen.org.au). The documentation for this package has been automatically generated by the ready4fun package and is therefore quite rudimentary. Further human authored documentation will follow in 2021.

Details

To learn more about youthu, start with the vignettes: `browseVignettes(package = "youthu")`

Author(s)

Maintainer: Matthew Hamilton <matthew.hamilton@orygen.org.au> ([ORCID](#))

Authors:

- Caroline Gao <caroline.gao@orygen.org.au> ([ORCID](#))

Other contributors:

- Orygen [copyright holder, funder]
- Headspace [funder]
- National Health and Medical Research Council [funder]

See Also

Useful links:

- <https://ready4-dev.github.io/youthu/>
- <https://github.com/ready4-dev/youthu>
- <https://www.ready4-dev.com/>

abbreviations_lup *Common abbreviations lookup table*

Description

A lookup table for abbreviations commonly used in object names in the youthupackage.

Usage

```
abbreviations_lup
```

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 503 rows and 3 columns.

Details

A tibble

short_name_chr Short name (a character vector)

long_name_chr Long name (a character vector)

plural_lgl Plural (a logical vector)

Source

<https://doi.org/10.7910/DVN/2Y9VF9>

add_change_in_ds_var *Add change in dataset variable*

Description

`add_change_in_ds_var()` is an Add function that updates an object by adding data to that object. Specifically, this function implements an algorithm to add change in dataset variable. Function argument `ds_tb` specifies the object to be updated. The function returns Updated dataset (a tibble).

Usage

```
add_change_in_ds_var(  
  ds_tb,  
  id_var_nm_1L_chr = "fkClientID",  
  round_var_nm_1L_chr = "round",  
  round_bl_val_1L_chr = "Baseline",  
  change_var_nm_1L_chr,  
  var_nm_1L_chr,  
  arrange_by_id_lgl = T  
)
```

Arguments

ds_tb	Dataset (a tibble)
id_var_nm_1L_chr	Identity variable name (a character vector of length one), Default: 'fkClientID'
round_var_nm_1L_chr	Round variable name (a character vector of length one), Default: 'round'
round_bl_val_1L_chr	Round baseline value (a character vector of length one), Default: 'Baseline'
change_var_nm_1L_chr	Change variable name (a character vector of length one)
var_nm_1L_chr	Variable name (a character vector of length one)
arrange_by_id_lgl	Arrange by identity (a logical vector), Default: T

Value

Updated dataset (a tibble)

add_costs_by_tmpt	<i>Add costs by time point</i>
-------------------	--------------------------------

Description

add_costs_by_tmpt() is an Add function that updates an object by adding data to that object. Specifically, this function implements an algorithm to add costs by time point. Function argument ds_tb specifies the object to be updated. The function returns Updated dataset (a tibble).

Usage

```
add_costs_by_tmpt(
  ds_tb,
  round_var_nm_1L_chr,
  round_lvls_chr = c("Baseline", "Follow-up"),
  costs_mean_dbl,
  costs_sd_dbl,
  extra_cost_args_ls = list(costs_var_nm_1L_chr = "costs_dbl"),
  fn = add_costs_from_gamma_dstr
)
```

Arguments

ds_tb	Dataset (a tibble)
round_var_nm_1L_chr	Round variable name (a character vector of length one)
round_lvls_chr	Round levels (a character vector), Default: c("Baseline", "Follow-up")
costs_mean_dbl	Costs mean (a double vector)
costs_sd_dbl	Costs standard deviation (a double vector)
extra_cost_args_ls	Extra cost arguments (a list), Default: list(costs_var_nm_1L_chr = "costs_dbl")
fn	Function (a function), Default: add_costs_from_gamma_dstr

Value

Updated dataset (a tibble)

add_costs_from_gamma_dstr *Add costs from gamma distribution*

Description

add_costs_from_gamma_dstr() is an Add function that updates an object by adding data to that object. Specifically, this function implements an algorithm to add costs from gamma distribution. Function argument ds_tb specifies the object to be updated. The function returns Updated dataset (a tibble).

Usage

```
add_costs_from_gamma_dstr(
  ds_tb,
  costs_mean_dbl,
  costs_sd_dbl,
  costs_var_nm_1L_chr = "costs_dbl"
)
```

Arguments

ds_tb Dataset (a tibble)

costs_mean_dbl Costs mean (a double vector)

costs_sd_dbl Costs standard deviation (a double vector)

costs_var_nm_1L_chr
 Costs variable name (a character vector of length one), Default: 'costs_dbl'

Value

Updated dataset (a tibble)

add_dates_from_dstr *Add dates from distribution*

Description

add_dates_from_dstr() is an Add function that updates an object by adding data to that object. Specifically, this function implements an algorithm to add dates from distribution. Function argument ds_tb specifies the object to be updated. The function returns Updated dataset (a tibble).

Usage

```
add_dates_from_dstr(  
  ds_tb,  
  bl_start_date_dtm,  
  bl_end_date_dtm,  
  duration_args_ls,  
  duration_fn = stats::rnorm,  
  date_var_nm_1L_chr = "date_psx",  
  id_var_nm_1L_chr = "fkClientID",  
  round_var_nm_1L_chr = "round",  
  round_bl_val_1L_chr = "Baseline",  
  origin_1L_chr = "1970-01-01"  
)
```

Arguments

ds_tb	Dataset (a tibble)
bl_start_date_dtm	Baseline start date (a date vector)
bl_end_date_dtm	Baseline end date (a date vector)
duration_args_ls	Duration arguments (a list)
duration_fn	Duration (a function), Default: stats::rnorm
date_var_nm_1L_chr	Date variable name (a character vector of length one), Default: 'date_psx'
id_var_nm_1L_chr	Identity variable name (a character vector of length one), Default: 'fkClientID'
round_var_nm_1L_chr	Round variable name (a character vector of length one), Default: 'round'
round_bl_val_1L_chr	Round baseline value (a character vector of length one), Default: 'Baseline'
origin_1L_chr	Origin (a character vector of length one), Default: '1970-01-01'

Value

Updated dataset (a tibble)

add_diffs_by_group_and_tmpt

Add differences by group and time point

Description

add_diffs_by_group_and_tmpt() is an Add function that updates an object by adding data to that object. Specifically, this function implements an algorithm to add differences by group and time point. Function argument ds_tb specifies the object to be updated. The function returns Updated dataset (a tibble).

Usage

```

add_diffs_by_group_and_tmpt(
  ds_tb = trial_ds_tb,
  cmprsn_var_nm_1L_chr = "study_arm_chr",
  cmprsn_group_match_val_chr = c("Intervention"),
  round_var_nm_1L_chr = "round",
  timepoint_match_val_1L_chr = "Follow-up",
  match_idx_var_nm_1L_chr = "match_idx_int",
  var_nms_chr,
  fns_ls,
  abs_mean_diff_dbl,
  diff_sd_dbl,
  multiplier_dbl,
  min_dbl,
  max_dbl,
  integer_lgl
)

```

Arguments

ds_tb	Dataset (a tibble), Default: trial_ds_tb
cmprsn_var_nm_1L_chr	Comparison variable name (a character vector of length one), Default: 'study_arm_chr'
cmprsn_group_match_val_chr	Comparison group match value (a character vector), Default: c("Intervention")
round_var_nm_1L_chr	Round variable name (a character vector of length one), Default: 'round'
timepoint_match_val_1L_chr	Timepoint match value (a character vector of length one), Default: 'Follow-up'
match_idx_var_nm_1L_chr	Match index variable name (a character vector of length one), Default: 'match_idx_int'
var_nms_chr	Variable names (a character vector)
fns_ls	Functions (a list)
abs_mean_diff_dbl	Absolute mean difference (a double vector)
diff_sd_dbl	Difference standard deviation (a double vector)
multiplier_dbl	Multiplier (a double vector)
min_dbl	Minimum (a double vector)
max_dbl	Maximum (a double vector)
integer_lgl	Integer (a logical vector)

Value

Updated dataset (a tibble)

 add_qalys

Add Quality Adjusted Life Years

Description

add_qalys() is an Add function that updates an object by adding data to that object. Specifically, this function implements an algorithm to add quality adjusted life years. Function argument ds_tb specifies the object to be updated. The function returns Updated dataset (a tibble).

Usage

```
add_qalys(
  ds_tb,
  cmprsn_var_nm_1L_chr = "study_arm_chr",
  duration_var_nm_1L_chr = "duration_prd",
  id_var_nm_1L_chr = "fkClientID",
  match_idx_var_nm_1L_chr = "match_idx_int",
  msrmt_date_var_nm_1L_chr = "date_dtm",
  qalys_var_nm_1L_chr = "qalys_dbl",
  round_var_nm_1L_chr = "round",
  round_bl_val_1L_chr = "Baseline",
  utl_change_var_nm_1L_chr = "utl_change_dbl",
  utl_var_nm_1L_chr = "utility_dbl",
  reshape_1L_lgl = T
)
```

Arguments

ds_tb	Dataset (a tibble)
cmprsn_var_nm_1L_chr	Comparison variable name (a character vector of length one), Default: 'study_arm_chr'
duration_var_nm_1L_chr	Duration variable name (a character vector of length one), Default: 'duration_prd'
id_var_nm_1L_chr	Identity variable name (a character vector of length one), Default: 'fkClientID'
match_idx_var_nm_1L_chr	Match index variable name (a character vector of length one), Default: 'match_idx_int'
msrmt_date_var_nm_1L_chr	Measurement date variable name (a character vector of length one), Default: 'date_dtm'
qalys_var_nm_1L_chr	Quality Adjusted Life Years variable name (a character vector of length one), Default: 'qalys_dbl'
round_var_nm_1L_chr	Round variable name (a character vector of length one), Default: 'round'
round_bl_val_1L_chr	Round baseline value (a character vector of length one), Default: 'Baseline'
utl_change_var_nm_1L_chr	Utility change variable name (a character vector of length one), Default: 'utl_change_dbl'

util_var_nm_1L_chr

Utility variable name (a character vector of length one), Default: 'utility_dbl'

reshape_1L_lgl Reshape (a logical vector of length one), Default: T

Value

Updated dataset (a tibble)

add_qalys_to_ds	<i>Add Quality Adjusted Life Years to dataset</i>
-----------------	---

Description

add_qalys_to_ds() is an Add function that updates an object by adding data to that object. Specifically, this function implements an algorithm to add quality adjusted life years to dataset. Function argument ds_tb specifies the object to be updated. The function returns Dataset (a tibble).

Usage

```
add_qalys_to_ds(
  ds_tb,
  predn_ds_ls,
  include_predrs_1L_lgl = T,
  reshape_1L_lgl = T
)
```

Arguments

ds_tb Dataset (a tibble)

predn_ds_ls Prediction dataset (a list)

include_predrs_1L_lgl
Include predictors (a logical vector of length one), Default: T

reshape_1L_lgl Reshape (a logical vector of length one), Default: T

Value

Dataset (a tibble)

add_utl_predn	<i>Add utility prediction</i>
---------------	-------------------------------

Description

add_utl_predn() is an Add function that updates an object by adding data to that object. Specifically, this function implements an algorithm to add utility prediction. Function argument data_tb specifies the object to be updated. The function returns Updated (a tibble).

Usage

```
add_utl_predn(  
  data_tb,  
  predn_ds_ls,  
  deterministic_1L_lgl = T,  
  force_min_max_1L_lgl = T,  
  key_1L_chr = NULL,  
  make_from_tbl_1L_lgl = T,  
  model_md1 = NULL,  
  new_data_is_1L_chr = "Simulated",  
  server_1L_chr = "dataverse.harvard.edu",  
  utl_cls_fn = NULL  
)
```

Arguments

data_tb	Data (a tibble)
predn_ds_ls	Prediction dataset (a list)
deterministic_1L_lgl	Deterministic (a logical vector of length one), Default: T
force_min_max_1L_lgl	Force minimum maximum (a logical vector of length one), Default: T
key_1L_chr	Key (a character vector of length one), Default: NULL
make_from_tbl_1L_lgl	Make from table (a logical vector of length one), Default: T
model_md1	Model (a model), Default: NULL
new_data_is_1L_chr	New data is (a character vector of length one), Default: 'Simulated'
server_1L_chr	Server (a character vector of length one), Default: 'dataverse.harvard.edu'
utl_cls_fn	Utility class (a function), Default: NULL

Value

Updated (a tibble)

 fns_dmt_tb

youthu function documentation table

Description

Meta-data on each youthu function used to create package documentation

Usage

fns_dmt_tb

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 39 rows and 10 columns.

Details

A tibble

fns_chr Functions (a character vector)

title_chr Title (a character vector)

desc_chr Description (a character vector)

details_chr Details (a character vector)

inc_for_main_user_lgl Include for main user (a logical vector)

output_chr Output (a character vector)

example_lgl Example (a logical vector)

args_ls Arguments (a list)

file_nm_chr File name (a character vector)

file_pfx_chr File prefix (a character vector)

Source

<https://ready4-dev.github.io/youthu/>

 fn_type_lup_tb

Function type lookup table

Description

A lookup table to find descriptions for different types of functions used within the youthu package suite.

Usage

fn_type_lup_tb

Format

An object of class `tbl_df` (inherits from `tbl`, `data.frame`) with 46 rows and 6 columns.

Details

A tibble

fn_type_nm_chr Function type name (a character vector)

fn_type_desc_chr Function type description (a character vector)

first_arg_desc_chr First argument description (a character vector)

second_arg_desc_chr Second argument description (a character vector)

is_generic_lgl Is generic (a logical vector)

is_method_lgl Is method (a logical vector)

Source

<https://doi.org/10.7910/DVN/2Y9VF9>

get_dv_dss_md1_smrys *Get dataverse datasets model summaries*

Description

`get_dv_dss_md1_smrys()` is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get dataverse datasets model summaries. Function argument `ids_chr` specifies the where to look for the required object. The function returns Dataverse datasets model summaries (a list).

Usage

```
get_dv_dss_md1_smrys(  
  ids_chr,  
  server_1L_chr = "dataverse.harvard.edu",  
  key_1L_chr = NULL  
)
```

Arguments

`ids_chr` Identities (a character vector)

`server_1L_chr` Server (a character vector of length one), Default: 'dataverse.harvard.edu'

`key_1L_chr` Key (a character vector of length one), Default: NULL

Value

Dataverse datasets model summaries (a list)

get_dv_ds_publication *Get dataverse dataset publication*

Description

get_dv_ds_publication() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get dataverse dataset publication. Function argument ds_url_1L_chr specifies the where to look for the required object. The function returns Doi url (a character vector of length one).

Usage

```
get_dv_ds_publication(
  ds_url_1L_chr,
  server_1L_chr = "dataverse.harvard.edu",
  key_1L_chr = NULL
)
```

Arguments

ds_url_1L_chr Dataset url (a character vector of length one)
 server_1L_chr Server (a character vector of length one), Default: 'dataverse.harvard.edu'
 key_1L_chr Key (a character vector of length one), Default: NULL

Value

Doi url (a character vector of length one)

get_dv_md1_smrys *Get dataverse model summarys*

Description

get_dv_md1_smrys() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get dataverse model summarys. Function argument mdls_lup specifies the where to look for the required object. The function is called for its side effects and does not return a value.

Usage

```
get_dv_md1_smrys(mdls_lup, md1_nms_chr = NULL)
```

Arguments

mdls_lup Models (a lookup table)
 md1_nms_chr Model names (a character vector), Default: NULL

Value

Dataverse model (summarys)

get_filtered_ttu_dss *Get filtered ttu datasets*

Description

get_filtered_ttu_dss() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get filtered ttu datasets. Function argument ttu_dv_dss_tb specifies the where to look for the required object. The function returns Ttu dataverse datasets (a tibble).

Usage

```
get_filtered_ttu_dss(
  ttu_dv_dss_tb = NULL,
  mdl_predrs_in_ds_chr = NULL,
  utility_type_chr = NULL,
  ttu_dv_nms_chr = "TTU",
  server_1L_chr = "dataverse.harvard.edu",
  key_1L_chr = NULL
)
```

Arguments

ttu_dv_dss_tb Ttu dataverse datasets (a tibble), Default: NULL
mdl_predrs_in_ds_chr
 Model predictors in dataset (a character vector), Default: NULL
utility_type_chr
 Utility type (a character vector), Default: NULL
ttu_dv_nms_chr Ttu dataverse names (a character vector), Default: 'TTU'
server_1L_chr Server (a character vector of length one), Default: 'dataverse.harvard.edu'
key_1L_chr Key (a character vector of length one), Default: NULL

Value

Ttu dataverse datasets (a tibble)

get_mdls_lup *Get models*

Description

get_mdls_lup() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get models lookup table. Function argument ttu_dv_dss_tb specifies the where to look for the required object. The function returns Models (a lookup table).

Usage

```
get_mdls_lup(
  ttu_dv_dss_tb = NULL,
  mdl_predrs_in_ds_chr = NULL,
  utility_type_chr = NULL,
  ttu_dv_nms_chr = "TTU",
  server_1L_chr = "dataverse.harvard.edu",
  key_1L_chr = NULL
)
```

Arguments

ttu_dv_dss_tb Ttu dataverse datasets (a tibble), Default: NULL
 mdl_predrs_in_ds_chr Model predictors in dataset (a character vector), Default: NULL
 utility_type_chr Utility type (a character vector), Default: NULL
 ttu_dv_nms_chr Ttu dataverse names (a character vector), Default: 'TTU'
 server_1L_chr Server (a character vector of length one), Default: 'dataverse.harvard.edu'
 key_1L_chr Key (a character vector of length one), Default: NULL

Value

Models (a lookup table)

get_md1_ctlg_url	<i>Get model ctlg url</i>
------------------	---------------------------

Description

get_md1_ctlg_url() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get model ctlg url. Function argument mdls_lup specifies the where to look for the required object. The function is called for its side effects and does not return a value.

Usage

```
get_md1_ctlg_url(
  mdls_lup,
  mdl_nm_1L_chr,
  server_1L_chr = "dataverse.harvard.edu",
  key_1L_chr = NULL
)
```

Arguments

mdls_lup Models (a lookup table)
 mdl_nm_1L_chr Model name (a character vector of length one)
 server_1L_chr Server (a character vector of length one), Default: 'dataverse.harvard.edu'
 key_1L_chr Key (a character vector of length one), Default: NULL

Value

NA ()

get_md1_ds_url	<i>Get model dataset url</i>
----------------	------------------------------

Description

get_md1_ds_url() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get model dataset url. Function argument mdls_lup specifies the where to look for the required object. The function is called for its side effects and does not return a value.

Usage

```
get_md1_ds_url(mdls_lup, mdl_nm_1L_chr)
```

Arguments

mdls_lup	Models (a lookup table)
mdl_nm_1L_chr	Model name (a character vector of length one)

Value

NA ()

get_md1_from_dv	<i>Get model from dataverse</i>
-----------------	---------------------------------

Description

get_md1_from_dv() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get model from dataverse. Function argument mdl_nm_1L_chr specifies the where to look for the required object. The function returns Model (a model).

Usage

```
get_md1_from_dv(  
  mdl_nm_1L_chr,  
  dv_ds_nm_1L_chr = "https://doi.org/10.7910/DVN/JC6PTV",  
  server_1L_chr = "dataverse.harvard.edu",  
  key_1L_chr = NULL  
)
```

Arguments

mdl_nm_1L_chr Model name (a character vector of length one)
 dv_ds_nm_1L_chr Dataverse dataset name (a character vector of length one), Default: 'https://doi.org/10.7910/DVN/JC6
 server_1L_chr Server (a character vector of length one), Default: 'dataverse.harvard.edu'
 key_1L_chr Key (a character vector of length one), Default: NULL

Value

Model (a model)

get_md1_metadata	<i>Get model metadata</i>
------------------	---------------------------

Description

get_md1_metadata() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get model metadata. Function argument mdls_lup specifies the where to look for the required object. The function returns Ingredients (a list).

Usage

```
get_md1_metadata(
  mdls_lup,
  mdl_nm_1L_chr,
  server_1L_chr = "dataverse.harvard.edu",
  key_1L_chr = NULL
)
```

Arguments

mdls_lup Models (a lookup table)
 mdl_nm_1L_chr Model name (a character vector of length one)
 server_1L_chr Server (a character vector of length one), Default: 'dataverse.harvard.edu'
 key_1L_chr Key (a character vector of length one), Default: NULL

Value

Ingredients (a list)

get_md1_smrys	<i>Get model summaries</i>
---------------	----------------------------

Description

get_md1_smrys() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get model summaries. Function argument ingredients_ls specifies the where to look for the required object. The function returns Models summary (a list).

Usage

```
get_md1_smrys(ingredients_ls, mdl_nms_chr = NULL)
```

Arguments

ingredients_ls Ingredients (a list)
 mdl_nms_chr Model names (a character vector), Default: NULL

Value

Models summary (a list)

get_model	<i>Get model</i>
-----------	------------------

Description

get_model() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get model. Function argument mdls_lup specifies the where to look for the required object. The function returns Model (a model).

Usage

```
get_model(  
  mdls_lup,  
  mdl_nm_1L_chr,  
  make_from_tbl_1L_lgl = T,  
  mdl_meta_data_ls = NULL,  
  server_1L_chr = "dataverse.harvard.edu",  
  key_1L_chr = NULL  
)
```

Arguments

mdl_s_lup	Models (a lookup table)
mdl_nm_1L_chr	Model name (a character vector of length one)
make_from_tbl_1L_lgl	Make from table (a logical vector of length one), Default: T
mdl_meta_data_ls	Model meta data (a list), Default: NULL
server_1L_chr	Server (a character vector of length one), Default: 'dataverse.harvard.edu'
key_1L_chr	Key (a character vector of length one), Default: NULL

Value

Model (a model)

get_predictors_lup	<i>Get predictors</i>
--------------------	-----------------------

Description

get_predictors_lup() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get predictors lookup table. Function argument mdl_meta_data_ls specifies the where to look for the required object. The function returns Predictors (a tibble).

Usage

```
get_predictors_lup(
  mdl_meta_data_ls = NULL,
  mdl_s_lup = NULL,
  mdl_nm_1L_chr = NULL,
  outp_is_abbrvs_tb = F,
  server_1L_chr = "dataverse.harvard.edu",
  key_1L_chr = NULL
)
```

Arguments

mdl_meta_data_ls	Model meta data (a list), Default: NULL
mdl_s_lup	Models (a lookup table), Default: NULL
mdl_nm_1L_chr	Model name (a character vector of length one), Default: NULL
outp_is_abbrvs_tb	Output is abbrvs (a tibble), Default: F
server_1L_chr	Server (a character vector of length one), Default: 'dataverse.harvard.edu'
key_1L_chr	Key (a character vector of length one), Default: NULL

Value

Predictors (a tibble)

get_tfmn_from_lup *Get transformation from*

Description

get_tfmn_from_lup() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get transformation from lookup table. Function argument mdl_nm_1L_chr specifies the where to look for the required object. The function returns Transformation (a character vector of length one).

Usage

```
get_tfmn_from_lup(mdl_nm_1L_chr, mdl_s_lup = NULL)
```

Arguments

mdl_nm_1L_chr Model name (a character vector of length one)
mdl_s_lup Models (a lookup table), Default: NULL

Value

Transformation (a character vector of length one)

get_ttu_ds_smrys *Get ttu dataset summaries*

Description

get_ttu_ds_smrys() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get ttu dataset summaries. Function argument ttu_dv_nm_1L_chr specifies the where to look for the required object. The function returns Dataverse datasets model summaries (a list).

Usage

```
get_ttu_ds_smrys(  
  ttu_dv_nm_1L_chr = "TTU",  
  server_1L_chr = "dataverse.harvard.edu",  
  key_1L_chr = NULL,  
  reference_int = NULL  
)
```

Arguments

ttu_dv_nm_1L_chr Tu dataverse name (a character vector of length one), Default: 'TTU'
server_1L_chr Server (a character vector of length one), Default: 'dataverse.harvard.edu'
key_1L_chr Key (a character vector of length one), Default: NULL
reference_int Reference (an integer vector), Default: NULL

Value

Dataverse datasets model summarys (a list)

get_ttu_dv_dss	<i>Get ttu dataverse datasets</i>
----------------	-----------------------------------

Description

get_ttu_dv_dss() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get ttu dataverse datasets. Function argument ttu_dv_nms_chr specifies the where to look for the required object. The function returns Ttu dataverse datasets (a tibble).

Usage

```
get_ttu_dv_dss(
  ttu_dv_nms_chr = "TTU",
  server_1L_chr = "dataverse.harvard.edu",
  key_1L_chr = NULL
)
```

Arguments

ttu_dv_nms_chr Ttu dataverse names (a character vector), Default: 'TTU'
 server_1L_chr Server (a character vector of length one), Default: 'dataverse.harvard.edu'
 key_1L_chr Key (a character vector of length one), Default: NULL

Value

Ttu dataverse datasets (a tibble)

get_ttu_dv_predrs	<i>Get ttu dataverse predictors</i>
-------------------	-------------------------------------

Description

get_ttu_dv_predrs() is a Get function that retrieves a pre-existing data object from memory, local file system or online repository. Specifically, this function implements an algorithm to get ttu dataverse predictors. Function argument ttu_dv_dss_tb specifies the where to look for the required object. The function returns Predictors (a character vector).

Usage

```
get_ttu_dv_predrs(
  ttu_dv_dss_tb = NULL,
  ttu_dv_nms_chr = "TTU",
  server_1L_chr = "dataverse.harvard.edu",
  key_1L_chr = NULL
)
```

Arguments

ttu_dv_dss_tb Ttu dataverse datasets (a tibble), Default: NULL
 ttu_dv_nms_chr Ttu dataverse names (a character vector), Default: 'TTU'
 server_1L_chr Server (a character vector of length one), Default: 'dataverse.harvard.edu'
 key_1L_chr Key (a character vector of length one), Default: NULL

Value

Predictors (a character vector)

make_balanced_fake_ds *Make balanced fake dataset*

Description

make_balanced_fake_ds() is a Make function that creates a new R object. Specifically, this function implements an algorithm to make balanced fake dataset. The function returns Dataset (a tibble).

Usage

```
make_balanced_fake_ds(
  ds_tb,
  match_on_vars_chr,
  id_var_nm_1L_chr = "UID_chr",
  round_var_nm_1L_chr = "Timepoint_chr",
  timepoint_bl_val_1L_chr = "Baseline",
  cmprsn_var_nm_1L_chr = "study_arm_chr",
  cmprsn_groups_chr = c("Intervention", "Control")
)
```

Arguments

ds_tb Dataset (a tibble)
 match_on_vars_chr Match on variables (a character vector)
 id_var_nm_1L_chr Identity variable name (a character vector of length one), Default: 'UID_chr'
 round_var_nm_1L_chr Round variable name (a character vector of length one), Default: 'Timepoint_chr'
 timepoint_bl_val_1L_chr Timepoint baseline value (a character vector of length one), Default: 'Baseline'
 cmprsn_var_nm_1L_chr Comparison variable name (a character vector of length one), Default: 'study_arm_chr'
 cmprsn_groups_chr Comparison groups (a character vector), Default: c("Intervention", "Control")

Value

Dataset (a tibble)

```
make_costs_vec_from_gamma_dstr
```

Make costs vector from gamma distribution

Description

`make_costs_vec_from_gamma_dstr()` is a Make function that creates a new R object. Specifically, this function implements an algorithm to make costs vector from gamma distribution. The function returns Costs (a double vector).

Usage

```
make_costs_vec_from_gamma_dstr(n_int, costs_mean_dbl, costs_sd_dbl)
```

Arguments

`n_int` N (an integer vector)
`costs_mean_dbl` Costs mean (a double vector)
`costs_sd_dbl` Costs standard deviation (a double vector)

Value

Costs (a double vector)

```
make_cst_efns_smry
```

Make cost efns summary

Description

`make_cst_efns_smry()` is a Make function that creates a new R object. Specifically, this function implements an algorithm to make cost efns summary. The function returns Summary (a double vector).

Usage

```
make_cst_efns_smry(
  ds_tb,
  idxs_int,
  change_types_chr = "dbl",
  benefits_pfx_1L_chr = "qalys_dbl",
  benefits_var_nm_1L_chr = "qalys",
  costs_pfx_1L_chr = "costs_dbl",
  costs_var_nm_1L_chr = "costs",
  change_sfx_1L_chr = "change",
  change_vars_chr = NA_character_,
  cmprsn_groups_chr = c("Intervention", "Control"),
  cmprsn_var_nm_1L_chr = "study_arm_chr",
  round_fup_val_1L_chr = "Follow-up"
)
```


Arguments

ds_tb	Dataset (a tibble)
idxs_int	Indices (an integer vector)
change_types_chr	Change types (a character vector), Default: 'dbl'
benefits_pfx_1L_chr	Benefits prefix (a character vector of length one), Default: 'qalys_dbl'
benefits_var_nm_1L_chr	Benefits variable name (a character vector of length one), Default: 'qalys'
costs_pfx_1L_chr	Costs prefix (a character vector of length one), Default: 'costs_dbl'
costs_var_nm_1L_chr	Costs variable name (a character vector of length one), Default: 'costs'
change_sfx_1L_chr	Change suffix (a character vector of length one), Default: 'change'
change_vars_chr	Change variables (a character vector), Default: 'NA'
cmpsrn_groups_chr	Comparison groups (a character vector), Default: c("Intervention", "Control")
cmpsrn_var_nm_1L_chr	Comparison variable name (a character vector of length one), Default: 'study_arm_chr'
round_fup_val_1L_chr	Round follow-up value (a character vector of length one), Default: 'Follow-up'

Value

Summary (a double vector)

make_fake_ds_one	<i>Make fake dataset one</i>
------------------	------------------------------

Description

make_fake_ds_one() is a Make function that creates a new R object. Specifically, this function implements an algorithm to make fake dataset one. The function returns Fake data (a tibble).

Usage

```
make_fake_ds_one()
```

Value

Fake data (a tibble)

make_fake_ds_two *Make fake dataset two*

Description

make_fake_ds_two() is a Make function that creates a new R object. Specifically, this function implements an algorithm to make fake dataset two. The function returns Matched dataset (a tibble).

Usage

```
make_fake_ds_two()
```

Value

Matched dataset (a tibble)

make_fake_trial_ds *Make fake trial dataset*

Description

make_fake_trial_ds() is a Make function that creates a new R object. Specifically, this function implements an algorithm to make fake trial dataset. The function returns Updated dataset (a tibble).

Usage

```
make_fake_trial_ds(  
  ds_tb,  
  id_var_nm_1L_chr = "fkClientID",  
  round_var_nm_1L_chr = "round",  
  round_lvls_chr = c("Baseline", "Follow-up"),  
  match_on_vars_chr,  
  cmprsn_var_nm_1L_chr = "study_arm_chr",  
  cmprsn_groups_chr = c("Intervention", "Control"),  
  fns_ls,  
  var_nms_chr,  
  abs_mean_diff_dbl,  
  diff_sd_dbl,  
  multiplier_dbl,  
  min_dbl,  
  max_dbl,  
  integer_lgl,  
  match_idx_var_nm_1L_chr = "match_idx_int"  
)
```

Arguments

ds_tb	Dataset (a tibble)
id_var_nm_1L_chr	Identity variable name (a character vector of length one), Default: 'fkClientID'
round_var_nm_1L_chr	Round variable name (a character vector of length one), Default: 'round'
round_lvls_chr	Round levels (a character vector), Default: c("Baseline", "Follow-up")
match_on_vars_chr	Match on variables (a character vector)
cmpsrn_var_nm_1L_chr	Comparison variable name (a character vector of length one), Default: 'study_arm_chr'
cmpsrn_groups_chr	Comparison groups (a character vector), Default: c("Intervention", "Control")
fns_ls	Functions (a list)
var_nms_chr	Variable names (a character vector)
abs_mean_diff_dbl	Absolute mean difference (a double vector)
diff_sd_dbl	Difference standard deviation (a double vector)
multiplier_dbl	Multiplier (a double vector)
min_dbl	Minimum (a double vector)
max_dbl	Maximum (a double vector)
integer_lgl	Integer (a logical vector)
match_idx_var_nm_1L_chr	Match index variable name (a character vector of length one), Default: 'match_idx_int'

Value

Updated dataset (a tibble)

make_hlth_ec_smry	<i>Make health economic summary</i>
-------------------	-------------------------------------

Description

make_hlth_ec_smry() is a Make function that creates a new R object. Specifically, this function implements an algorithm to make health economic summary. The function returns He summary (a list).

Usage

```
make_hlth_ec_smry(
  ds_tb,
  predn_ds_ls,
  wtp_dbl = 50000,
  bootstrap_iters_1L_int = 1000,
  benefits_pfx_1L_chr = "qalys_dbl",
  benefits_var_nm_1L_chr = "qalys",
  costs_var_nm_1L_chr = "costs",
  change_sfx_1L_chr = "change"
)
```

Arguments

ds_tb	Dataset (a tibble)
predn_ds_ls	Prediction dataset (a list)
wtp_dbl	Willingness to pay (a double vector), Default: 50000
bootstrap_iters_1L_int	Bootstrap iterations (an integer vector of length one), Default: 1000
benefits_pfx_1L_chr	Benefits prefix (a character vector of length one), Default: 'qalys_dbl'
benefits_var_nm_1L_chr	Benefits variable name (a character vector of length one), Default: 'qalys'
costs_var_nm_1L_chr	Costs variable name (a character vector of length one), Default: 'costs'
change_sfx_1L_chr	Change suffix (a character vector of length one), Default: 'change'

Value

He summary (a list)

make_matched_ds	<i>Make matched dataset</i>
-----------------	-----------------------------

Description

make_matched_ds() is a Make function that creates a new R object. Specifically, this function implements an algorithm to make matched dataset. The function returns Matched dataset (a tibble).

Usage

```
make_matched_ds(sngl_grp_ds_tb, cmprsn_smry_tb, ds_smry_ls)
```

Arguments

sngl_grp_ds_tb	Single group dataset (a tibble)
cmprsn_smry_tb	Comparison summary (a tibble)
ds_smry_ls	Dataset summary (a list)

Value

Matched dataset (a tibble)

make_matched_ds_spine *Make matched dataset spine*

Description

make_matched_ds_spine() is a Make function that creates a new R object. Specifically, this function implements an algorithm to make matched dataset spine. The function returns Matched dataset (a tibble).

Usage

```
make_matched_ds_spine(  
  ds_tb,  
  round_var_nm_1L_chr = "Timepoint_chr",  
  timepoint_bl_val_1L_chr = "Baseline",  
  cmprsn_var_nm_1L_chr = "study_arm_chr",  
  active_arm_val_1L_chr = "Intervention",  
  id_var_nm_1L_chr = "fkClientID",  
  match_on_vars_chr  
)
```

Arguments

ds_tb	Dataset (a tibble)
round_var_nm_1L_chr	Round variable name (a character vector of length one), Default: 'Timepoint_chr'
timepoint_bl_val_1L_chr	Timepoint baseline value (a character vector of length one), Default: 'Baseline'
cmprsn_var_nm_1L_chr	Comparison variable name (a character vector of length one), Default: 'study_arm_chr'
active_arm_val_1L_chr	Active arm value (a character vector of length one), Default: 'Intervention'
id_var_nm_1L_chr	Identity variable name (a character vector of length one), Default: 'fkClientID'
match_on_vars_chr	Match on variables (a character vector)

Value

Matched dataset (a tibble)

 make_predn_metadata_ls

Make prediction metadata

Description

make_predn_metadata_ls() is a Make function that creates a new R object. Specifically, this function implements an algorithm to make prediction metadata list. The function returns Prediction metadata (a list).

Usage

```
make_predn_metadata_ls(
  data_tb,
  cmprsn_groups_chr = NULL,
  cmprsn_var_nm_1L_chr = NULL,
  costs_var_nm_1L_chr = NULL,
  id_var_nm_1L_chr = "UID",
  mdl_meta_data_ls = NULL,
  mdl_ls_lup = NULL,
  mdl_nm_1L_chr = NULL,
  msrmt_date_var_nm_1L_chr = NULL,
  predr_vars_nms_chr = NULL,
  round_var_nm_1L_chr,
  round_bl_val_1L_chr,
  utl_var_nm_1L_chr = "AQoL6D_HU",
  server_1L_chr = "dataverse.harvard.edu",
  key_1L_chr = NULL
)
```

Arguments

data_tb	Data (a tibble)
cmprsn_groups_chr	Comparison groups (a character vector), Default: NULL
cmprsn_var_nm_1L_chr	Comparison variable name (a character vector of length one), Default: NULL
costs_var_nm_1L_chr	Costs variable name (a character vector of length one), Default: NULL
id_var_nm_1L_chr	Identity variable name (a character vector of length one), Default: 'UID'
mdl_meta_data_ls	Model meta data (a list), Default: NULL
mdl_ls_lup	Models (a lookup table), Default: NULL
mdl_nm_1L_chr	Model name (a character vector of length one), Default: NULL
msrmt_date_var_nm_1L_chr	Measurement date variable name (a character vector of length one), Default: NULL

predr_vars_nms_chr Predictor variables names (a character vector), Default: NULL
 round_var_nm_1L_chr Round variable name (a character vector of length one)
 round_bl_val_1L_chr Round baseline value (a character vector of length one)
 utl_var_nm_1L_chr Utility variable name (a character vector of length one), Default: 'AQoL6D_HU'
 server_1L_chr Server (a character vector of length one), Default: 'dataverse.harvard.edu'
 key_1L_chr Key (a character vector of length one), Default: NULL

Value

Prediction metadata (a list)

make_sngl_grp_ds	<i>Make single group dataset</i>
------------------	----------------------------------

Description

make_sngl_grp_ds() is a Make function that creates a new R object. Specifically, this function implements an algorithm to make single group dataset. The function returns Single group dataset (a tibble).

Usage

```
make_sngl_grp_ds(seed_ds_tb = NULL, ds_smry_ls)
```

Arguments

seed_ds_tb Seed dataset (a tibble), Default: NULL
 ds_smry_ls Dataset summary (a list)

Value

Single group dataset (a tibble)

predict_from_md1_coefs

Predict from model coefficients

Description

predict_from_md1_coefs() is a Predict function that makes predictions from data using a specified statistical model. Specifically, this function implements an algorithm to predict from model coefficients. The function returns Pred (a double vector).

Usage

```
predict_from_md1_coefs(smry_of_md1_tb, new_data_tb)
```

Arguments

smry_of_md1_tb Summary of model (a tibble)
 new_data_tb New data (a tibble)

Value

Pred (a double vector)

transform_ds_for_cmprsn

Transform dataset for comparison

Description

transform_ds_for_cmprsn() is a Transform function that edits an object in such a way that core object attributes - e.g. shape, dimensions, elements, type - are altered. Specifically, this function implements an algorithm to transform dataset for comparison. Function argument ds_tb specifies the object to be updated. Argument cmprsn_var_nm_1L_chr provides the object to be updated. The function returns Dataset (a tibble).

Usage

```
transform_ds_for_cmprsn(
  ds_tb,
  cmprsn_var_nm_1L_chr,
  id_var_nm_1L_chr = "UID_chr",
  round_var_nm_1L_chr = "Timepoint_chr",
  cmprsn_groups_chr = c("Intervention", "Control")
)
```


Arguments

ds_tb	Dataset (a tibble)
cmprsn_var_nm_1L_chr	Comparison variable name (a character vector of length one)
id_var_nm_1L_chr	Identity variable name (a character vector of length one), Default: 'UID_chr'
round_var_nm_1L_chr	Round variable name (a character vector of length one), Default: 'Timepoint_chr'
cmprsn_groups_chr	Comparison groups (a character vector), Default: c("Intervention", "Control")

Value

Dataset (a tibble)

update_col_with_diff *Update column with difference*

Description

update_col_with_diff() is an Update function that edits an object, while preserving core object attributes. Specifically, this function implements an algorithm to update column with difference. Function argument ds_tb specifies the object to be updated. Argument var_nm_1L_chr provides the object to be updated. The function is called for its side effects and does not return a value.

Usage

```
update_col_with_diff(
  ds_tb,
  var_nm_1L_chr,
  fn,
  abs_mean_diff_1L_dbl,
  diff_sd_1L_dbl,
  multiplier_1L_dbl,
  min_1L_dbl,
  max_1L_dbl,
  integer_1L_lgl
)
```

Arguments

ds_tb	Dataset (a tibble)
var_nm_1L_chr	Variable name (a character vector of length one)
fn	Function (a function)
abs_mean_diff_1L_dbl	Absolute mean difference (a double vector of length one)
diff_sd_1L_dbl	Difference standard deviation (a double vector of length one)
multiplier_1L_dbl	Multiplier (a double vector of length one)

min_1L_dbl Minimum (a double vector of length one)
 max_1L_dbl Maximum (a double vector of length one)
 integer_1L_lgl Integer (a logical vector of length one)

Value

New (a dataset)

update_multpl_cols_with_diffs
Update multiplier columns with differences

Description

update_multpl_cols_with_diffs() is an Update function that edits an object, while preserving core object attributes. Specifically, this function implements an algorithm to update multiplier columns with differences. Function argument ds_tb specifies the object to be updated. Argument var_nms_chr provides the object to be updated. The function returns Updated dataset (a tibble).

Usage

```
update_multpl_cols_with_diffs(  
  ds_tb,  
  var_nms_chr,  
  fns_ls,  
  abs_mean_diff_dbl,  
  diff_sd_dbl,  
  multiplier_dbl,  
  min_dbl,  
  max_dbl,  
  integer_lgl  
)
```

Arguments

ds_tb Dataset (a tibble)
 var_nms_chr Variable names (a character vector)
 fns_ls Functions (a list)
 abs_mean_diff_dbl
 Absolute mean difference (a double vector)
 diff_sd_dbl Difference standard deviation (a double vector)
 multiplier_dbl Multiplier (a double vector)
 min_dbl Minimum (a double vector)
 max_dbl Maximum (a double vector)
 integer_lgl Integer (a logical vector)

Value

Updated dataset (a tibble)

Index

* datasets

- abbreviations_lup, 4
- fn_type_lup_tb, 12
- fns_dmt_tb, 12

- abbreviations_lup, 4
- add_change_in_ds_var, 4
- add_costs_by_tmpt, 5
- add_costs_from_gamma_dstr, 6
- add_dates_from_dstr, 6
- add_diffs_by_group_and_tmpt, 7
- add_qalys, 9
- add_qalys_to_ds, 10
- add_utl_predn, 11

- fn_type_lup_tb, 12
- fns_dmt_tb, 12

- get_dv_ds_publication, 14
- get_dv_dss_mdls_smrys, 13
- get_dv_mdls_smrys, 14
- get_filtered_ttu_dss, 15
- get_mdls_ctlg_url, 16
- get_mdls_ds_url, 17
- get_mdls_from_dv, 17
- get_mdls_metadata, 18
- get_mdls_smrys, 19
- get_mdls_lup, 15
- get_model, 19
- get_predictors_lup, 20
- get_tfmn_from_lup, 21
- get_ttu_ds_smrys, 21
- get_ttu_dv_dss, 22
- get_ttu_dv_predrs, 22

- make_balanced_fake_ds, 23
- make_costs_vec_from_gamma_dstr, 24
- make_cst_efns_smry, 24
- make_fake_ds_one, 25
- make_fake_ds_two, 26
- make_fake_trial_ds, 26
- make_hlth_ec_smry, 27
- make_matched_ds, 28
- make_matched_ds_spine, 29

- make_predn_metadata_ls, 30
- make_sngl_grp_ds, 31

- predict_from_mdls_coefs, 32

- transform_ds_for_cmprsn, 32

- update_col_with_diff, 33
- update_multpl_cols_with_diffs, 34

- youthu (youthu-package), 3
- youthu-package, 3