Package: BiostatsUHNplus (via r-universe)

August 23, 2024

Title Nested Data Summary, Adverse Events and REDCap

Version 0.0.10

Description Tools and code snippets for summarizing nested data, adverse events and REDCap study information.

License MIT + file LICENSE

Depends R (>= 4.2)

Imports afex, coda, cowplot, dplyr, forcats, ggh4x, ggplot2, lifecycle, MCMCglmm, modeest, openxlsx, plyr, purrr, reportRmd, rlang, rstatix, stats, stringi, stringr, tibble, utils

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Encoding UTF-8

LazyData true

RoxygenNote 7.2.3

Collate 'data.R' 'globals.R' 'as_numeric_parse.R' 'covsum_nested.R' 'dsmb_ccru.R' 'nice_mcmcglmm.R' 'nice_mcmcglmm_icc.R' 'ae_timeline_plot.R' 'caterpillar_plot.R' 'redcap_data_out.R'

NeedsCompilation no

Author Lisa Avery [aut] (<https://orcid.org/0000-0002-8431-5143>), Zeynep Baskurt [aut], Osvaldo Espin-Garcia [aut], Katrina Hueniken [aut], Katherine Lajkosz [aut] (<https://orcid.org/0000-0003-3760-5401>), Xuan Li [aut], Tyler Pittman [cre, aut] (<https://orcid.org/0000-0002-5013-6980>), Anna Santiago [aut] (<https://orcid.org/0000-0002-0932-2386>), Jessica Weiss [aut], Wei Xu [aut]

Maintainer Tyler Pittman <tyler.pittman@uhn.ca>

Date/Publication 2024-05-24 21:10:30 UTC

Repository https://tylerpittman.r-universe.dev

RemoteUrl https://github.com/cran/BiostatsUHNplus

RemoteRef HEAD

RemoteSha d89656ef514b1d1521b7bf3845aded31f79dc396

Contents

---

ae *Simulated adverse events for patients receiving two study agents.*

Description

Simulated adverse events for patients receiving two study agents.

Usage

```
ae
```

Format

A data frame with 394 rows and 9 variables:

Subject Patient ID

ae_detail Adverse event detail, also known as lowest level term

ae_category Adverse event category, also known as system organ class

CTCAE5_LLT_NM Common Terminology Criteria for Adverse Events (CTCAE) version 5

AE_VERBATIM_TRM_TXT Adverse event verbatim text entered by clinical registered nurse, for "Other, specify"

AE_SEV_GD Adverse event severity grade, scale from 1 to 5

AE_ONSET_DT_INT Adverse event onset date

CTC_AE_ATTR_SCALE Attribution scale of adverse event to first study agent

CTC_AE_ATTR_SCALE_1 Attribution scale of adverse event to second study agent

ae_timeline_plot	*Outputs related adverse event timeline plots including just system organ class (AE category), or system organ class and lowest level term (AE detail). This function can fit up to 5 different attributions. Modify width, height and scale parameters in ggsave() to customize fit for large plot.*

Description

Outputs related adverse event timeline plots including just system organ class (AE category), or system organ class and lowest level term (AE detail). This function can fit up to 5 different attributions. Modify width, height and scale parameters in ggsave() to customize fit for large plot.

Usage

```
ae_timeline_plot(
  subjID,
  subjID_ineligText = NULL,
  baseline_datasets,
  ae_dataset,
  ae_attribVars,
  ae_attribVarsName = NULL,
  ae_attribVarText = NULL,
  startDtVars,
  ae_detailVar,
  ae_categoryVar,
  ae_severityVar,
  ae_onsetDtVar,
  time_unit = c("day", "week", "month", "year"),
  include_ae_detail = TRUE,
  legendPerSpace = NULL,
  fonts = NULL,
  fontColours = NULL,
  panelColours = NULL,
  attribColours = NULL,
  attribSymbols = NULL,
  columnWidths = NULL
)
```

Arguments

subjID	key identifier field for participant ID in data sets
subjID_ineligText	character text that denotes participant IDs to exclude, for example, c("New Subject") (if provided)

`baseline_datasets`
: list of data frames that contain baseline participant characteristics, for example, list(enrollment_DF,demography_DF,ineligibility_DF)

`ae_dataset`
: data frame that contains subject AEs

`ae_attribVars`
: field(s) that denotes attribution to intervention under study, for example, c("CTC_AE_ATTR_SCALE","C (if provided)

`ae_attribVarsName`
: character text that denotes name of interventions under study, for example, c("Drug 1", "Drug 2") (if provided)

`ae_attribVarText`
: character text that denotes related attribution, for example c("Definite", "Probable", "Possible") (if provided)

`startDtVars`
: field(s) that denotes participant start date (i.e. 10MAY2021). For example, it could be enrollment date or screening date. If more than one field given (unique names are required), each field is assumed to be specific start date for attribution in corresponding field order

`ae_detailVar`
: field that denotes participant AE detail (lowest level term)

`ae_categoryVar`
: field that denotes participant AE category (system organ class)

`ae_severityVar`
: field that denotes participant AE severity grade (numeric)

`ae_onsetDtVar`
: field that denotes participant AE onset date

`time_unit`
: character text that denotes time unit for desired timeline, for example, could be one of c("day","week","month","year") (if provided)

`include_ae_detail`
: boolean that denotes if AE detail should be included in timeline plot. Default is True

`legendPerSpace`
: parameter at denotes proportion of vertical image space dedicated to legend at bottom. Default is 0.05 for AE detail and 0.1 for AE Category

`fonts`
: character text that denotes font for AE category, AE detail, axis, legend and plot labels (if provided)

`fontColours`
: character text that denotes system font colours for AE category and AE detail (if provided)

`panelColours`
: character text that denotes panel background colours for AE category, AE detail and plot area (if provided)

`attribColours`
: character text that denotes colours for attributions, supports up to 10 distinct colours (if provided)

`attribSymbols`
: text that denotes median plot symbols for attributions, supports up to 10 distinct symbols (if provided)

`columnWidths`
: text that denotes character columns widths for AE category and AE detail columns (if provided)

Value

ggplot object of AE timeline plot

Examples

```
data("drug1_admin", "drug2_admin", "ae");
p <- ae_timeline_plot(subjID="Subject",subjID_ineligText=c("01","11"),
                      baseline_datasets=list(drug1_admin, drug2_admin),
                      ae_dataset=ae,
                      ae_attribVars=c("CTC_AE_ATTR_SCALE","CTC_AE_ATTR_SCALE_1"),
                      ae_attribVarsName=c("Drug 1","Drug 2"),
                      ae_attribVarText=c("Definite", "Probable", "Possible"),
                      startDtVars=c("TX1_DATE_INT","TX2_DATE_INT"),
                      ae_detailVar="ae_detail",
                      ae_categoryVar="ae_category",ae_severityVar="AE_SEV_GD",
                      ae_onsetDtVar="AE_ONSET_DT_INT",time_unit="month",
                      include_ae_detail=FALSE,
                     fonts=c("Calibri","Albany AMT","Gadugi","French Script MT","Forte"),
                      fontColours=c("#FFE135"),
                      panelColours=c("#E52B50",NA,"#FFE4C4"),
                      attribColours=c("#9AB973","#01796F","#FFA343","#CC7722"),
                      attribSymbols=c(7,8,5,6),
                      columnWidths=c(23))
```

---

as_numeric_parse *Modification of the as.numeric function that prints entries that fail to parse as a message*

Description

Modification of the as.numeric function that prints entries that fail to parse as a message

Usage

```
as_numeric_parse(x)
```

Arguments

x	string or vector to coerce to numeric

Value

No return value, called for side effects

Examples

```
z <- as_numeric_parse(c(1:5, "String1",6:10,"String2"))
z
```

`caterpillar_plot` *Caterpillar plot. Useful for plotting random effects from hierarchical models, such as MCMCglmm::MCMCglmm() object, that have binary outcome.*

Description

Caterpillar plot. Useful for plotting random effects from hierarchical models, such as MCMCglmm::MCMCglmm() object, that have binary outcome.

Usage

```
caterpillar_plot(
  subjID,
  subjLabel = NULL,
  remove.text.subjID = FALSE,
  mcmcglmm_object,
  orig_dataset,
  binaryOutcomeVar,
  prob = NULL,
  title = NULL,
  no.title = FALSE,
  subtitle = NULL,
  ncol = NULL,
  fonts = NULL,
  columnTextWidth = NULL,
  break.label.summary = FALSE
)
```

Arguments

`subjID`	key identifier field for participant ID in data sets
`subjLabel`	text label field in dataset to replace key identifier field for participant ID with in plot (if provided)
`remove.text.subjID`	boolean indicating if non-numeric text should be removed from subjID in plot label. Note that this can only be used if there are non-duplicate participant IDs when non-numeric text is removed. Default is FALSE (if provided)
`mcmcglmm_object`	MCMCglmm model output
`orig_dataset`	data frame supplied to MCMCglmm function
`binaryOutcomeVar`	name of binary variable (0,1) that denotes outcome in MCMCglmm model
`prob`	probability for highest posterior density interval, similar to a confidence interval. Default is 0.95 (if provided)

title	title of the plot. Overrides default title (if provided)
no.title	boolean that denotes if title should be outputted in plot. Default is TRUE (if provided)
subtitle	subtitle of the plot. Overrides default subtitle (if provided)
ncol	number of columns in plot. Default is 2 (if provided)
fonts	character text that denotes font for title, subtitle, category labels, x-axis plot labels (if provided)
columnTextWidth	numeric that denotes character width for label text before breaking to start new line. Default is 20 characters (if provided)
break.label.summary	boolean to indicate if new line should start in label before (n, event) summary. Default is FALSE

Value

ggplot object of caterpillar plot

Examples

```
data("ae");

ae$G3Plus <- 0;
ae$G3Plus[ae$AE_SEV_GD %in% c("3", "4", "5")] <- 1;
ae$Drug_1_Attribution <- 0;
ae$Drug_1_Attribution[ae$CTC_AE_ATTR_SCALE %in% c("Definite", "Probable", "Possible")] <- 1;
ae$Drug_2_Attribution <- 0;
ae$Drug_2_Attribution[ae$CTC_AE_ATTR_SCALE_1 %in% c("Definite", "Probable", "Possible")] <- 1;

prior2RE <- list(R = list(V = diag(1), fix = 1),
  G=list(G1=list(V=1, nu=0.02), G2=list(V=1, nu=0.02)));

model1 <- MCMCglmm::MCMCglmm(G3Plus ~ Drug_1_Attribution + Drug_2_Attribution,
  random=~Subject + ae_category, family="categorical", data=ae, saveX=TRUE,
  verbose=FALSE, burnin=2000, nitt=10000, thin=10, pr=TRUE, prior=prior2RE);

p <- caterpillar_plot(subjID = "Subject",
  mcmcglmm_object = model1,
  prob = 0.99,
  orig_dataset = ae,
  binaryOutcomeVar = "G3Plus")

p <- caterpillar_plot(subjID = "ae_category",
  mcmcglmm_object = model1,
  prob = 0.95,
  orig_dataset = ae,
  remove.text.subjID = FALSE,
  ncol = 4,
  binaryOutcomeVar = "G3Plus",
  subtitle = "System organ class (n, event)",
```

```
  title = "Odds Ratio for G3+ Severity with 95% Highest Posterior Density Interval",
  fonts = c("Arial", "Arial", "Arial", "Arial"),
  break.label.summary = TRUE)
```

`covsum_nested` *Nested version of reportRmd covsum()*

Description

Nested version of reportRmd covsum()

Usage

```
covsum_nested(
  data,
  covs,
  maincov = NULL,
  id = NULL,
  digits = 1,
  numobs = NULL,
  markup = TRUE,
  sanitize = TRUE,
  nicenames = TRUE,
  IQR = FALSE,
  all.stats = FALSE,
  pvalue = TRUE,
  effSize = TRUE,
  show.tests = TRUE,
  nCores = NULL,
  nested.test = NULL,
  nsim = NULL,
  excludeLevels = NULL,
  dropLevels = TRUE,
  full = TRUE,
  digits.cat = 0,
  testcont = c("rank-sum test", "ANOVA"),
  testcat = c("Chi-squared", "Fisher"),
  include_missing = FALSE,
  percentage = c("column", "row")
)
```

Arguments

`data`	dataframe containing data
`covs`	character vector with the names of columns to include in table
`maincov`	covariate to stratify table by
`id`	covariates to nest summary by

digits	number of digits for summarizing mean data, does not affect p-values
numobs	named list overriding the number of people you expect to have the covariate
markup	boolean indicating if you want latex markup
sanitize	boolean indicating if you want to sanitize all strings to not break LaTeX
nicenames	boolean indicating if you want to replace . and _ in strings with a space
IQR	boolean indicating if you want to display the inter quantile range (Q1,Q3) as opposed to (min,max) in the summary for continuous variables
all.stats	boolean indicating if all summary statistics (Q1,Q3 + min,max on a separate line) should be displayed. Overrides IQR.
pvalue	boolean indicating if you want p-values included in the table
effSize	boolean indicating if you want effect sizes included in the table. Can only be obtained if pvalue is also requested.
show.tests	boolean indicating if the type of statistical used should be shown in a column beside the p-values. Ignored if pvalue=FALSE.
nCores	if > 1, specifies number of cores to use for parallel processing for calculating the nested p-value (default: 1).
nested.test	specifies test used for calculating nested p-value from afex::mixed function. Either *parametric bootstrap* method or *likelihood ratio test* method (default: "LRT"). Parametric bootstrap takes longer.
nsim	specifies number of simulations to use for calculating nested p-value with *parametric bootstrap* method used for nested.test (default: 1000).
excludeLevels	a named list of covariate levels to exclude from statistical tests in the form list(varname =c('level1','level2')). These levels will be excluded from association tests, but not the table. This can be useful for levels where there is a logical skip (i.e. not missing, but not presented). Ignored if pvalue=FALSE.
dropLevels	logical, indicating if empty factor levels be dropped from the output, default is TRUE.
full	boolean indicating if you want the full sample included in the table, ignored if maincov is NULL
digits.cat	number of digits for the proportions when summarizing categorical data (default: 0)
testcont	test of choice for continuous variables,one of *rank-sum* (default) or *ANOVA*
testcat	test of choice for categorical variables,one of *Chi-squared* (default) or *Fisher*
include_missing	Option to include NA values of maincov. NAs will not be included in statistical tests
percentage	choice of how percentages are presented, one of *column* (default) or *row*

See Also

fisher.test,chisq.test, wilcox.test,kruskal.test, anova and mixed

demography *Simulated demography for patients.*

Description

Simulated demography for patients.

Usage

```
demography
```

Format

A data frame with 12 rows and 2 variables:

Subject Patient ID

GENDER_CODE Patient gender

drug1_admin *Simulated study agent 1 for patients.*

Description

Simulated study agent 1 for patients.

Usage

```
drug1_admin
```

Format

A data frame with 12 rows and 2 variables:

Subject Patient ID

TX1_DATE_INT Study agent 1 start date of patient on study

drug2_admin — *Simulated study agent 2 for patients.*

Description

Simulated study agent 2 for patients.

Usage

```
drug2_admin
```

Format

A data frame with 12 rows and 2 variables:

Subject Patient ID

TX2_DATE_INT Study agent 2 start date of patient on study

dsmb_ccru — *Outputs the three DSMB-CCRU AE summary tables in Excel format per UHN template*

Description

Outputs the three DSMB-CCRU AE summary tables in Excel format per UHN template

Usage

```
dsmb_ccru(
  protocol,
  setwd,
  title,
  comp = NULL,
  pi,
  presDate,
  cutDate,
  boundDate = NULL,
  subjID,
  subjID_ineligText = NULL,
  baseline_datasets,
  ae_dataset,
  ineligVar = NULL,
  ineligVarText = NULL,
  genderVar,
  enrolDtVar,
```

```
  ae_detailVar,
  ae_categoryVar,
  ae_severityVar,
  ae_onsetDtVar,
  ae_detailOtherText = NULL,
  ae_detailOtherVar = NULL,
  ae_verbatimVar = NULL,
  numSubj = NULL,
  fileNameUnderscore = TRUE
)
```

Arguments

protocol	study protocol name (uppercase, no spaces permitted)
setwd	directory to write Excel summary files to
title	full character vector with name of study
comp	baseline comparison group, for example, cohort (if provided)
pi	character vector name of study principal investigator
presDate	presentation date (i.e. 17NOV2023) for DSMB
cutDate	recent cutoff date for AEs (i.e. 31AUG2023)
boundDate	lower bound cutoff date for AEs (if provided)
subjID	key identifier field for participant ID in data sets
subjID_ineligText	character text that denotes participant IDs to exclude, for example, c("New Subject") (if provided)
baseline_datasets	list of data frames that contain baseline participant characteristics, for example, list(enrollment_DF,demography_DF,ineligibility_DF)
ae_dataset	data frame that contains subject AEs
ineligVar	field that denotes participant ineligibility (if provided)
ineligVarText	character text that denotes participant ineligibility, for example, c("Yes", "Y") (if provided)
genderVar	field that denotes participant gender
enrolDtVar	field that denotes participant enrollment date (i.e. 10MAY2021)
ae_detailVar	field that denotes participant AE detail (lowest level term)
ae_categoryVar	field that denotes participant AE category (system organ class)
ae_severityVar	field that denotes participant AE severity grade (numeric)
ae_onsetDtVar	field that denotes participant AE onset date
ae_detailOtherText	character text that denotes referencing verbatim AE field, for example, c("Other, specify", "OTHER") (if provided)
ae_detailOtherVar	field that denotes participant AE detail other (if provided)

`ae_verbatimVar` field that denotes participant AE detail verbatim (if provided)

`numSubj` vector to override value for number of participants in summary (if provided)

`fileNameUnderscore`
boolean that denotes if spaces should be underscore in filename

Value

three Excel files containing DSMB-CCRU AE summary tables

Examples

```
data("enrollment", "demography", "ineligibility", "ae");
dsmb_ccru(protocol="EXAMPLE_STUDY",setwd="./man/tables/",
  title="Phase X Study to Evaluate Treatments A-D",
  comp="COHORT",pi="Dr. Principal Investigator",
  presDate="30OCT2020",cutDate="31AUG2020",
  boundDate=NULL,subjID="Subject",subjID_ineligText=c("New Subject","Test"),
  baseline_datasets=list(enrollment,demography,ineligibility),
  ae_dataset=ae,ineligVar="INELIGIBILITY_STATUS",ineligVarText=c("Yes","Y"),
  genderVar="GENDER_CODE",enrolDtVar="ENROL_DATE_INT",ae_detailVar="ae_detail",
  ae_categoryVar="ae_category",ae_severityVar="AE_SEV_GD",
  ae_onsetDtVar="AE_ONSET_DT_INT",ae_detailOtherText="Other, specify",
  ae_detailOtherVar="CTCAE5_LLT_NM",ae_verbatimVar="AE_VERBATIM_TRM_TXT",
  numSubj=c(2,4,5,6))
```

`enrollment` *Enrollment data Simulated enrollment for patients.*

Description

Enrollment data

Simulated enrollment for patients.

Usage

```
enrollment
```

Format

A data frame with 12 rows and 3 variables:

Subject Patient ID

COHORT Study cohort for patient

ENROL_DATE_INT Enrollment date of patient to study

ineligibility *Simulated ineligibility for patients.*

Description

Simulated ineligibility for patients.

Usage

```
ineligibility
```

Format

A data frame with 11 rows and 2 variables:

Subject Patient ID

INELIGIBILITY_STATUS Recorded ineligibility status of patient to study

nice_mcmcglmm *Nice table of model output from MCMCglmm::MCMCglmm()*

Description

Nice table of model output from MCMCglmm::MCMCglmm()

Usage

```
nice_mcmcglmm(mcmcglmm_object, dataset)
```

Arguments

mcmcglmm_object	returned output from MCMCglmm()
dataset	dataframe containing data

Value

grouped_df

Examples

```
## Not run:
data(ae)

ae$AE_SEV_GD <- as.numeric(ae$AE_SEV_GD);
ae$Drug_1_Attribution <- "No";
ae$Drug_1_Attribution[ae$CTC_AE_ATTR_SCALE %in% c("Definite", "Probable", "Possible")] <- "Yes";
ae$Drug_1_Attribution <- as.factor(ae$Drug_1_Attribution);
ae$Drug_2_Attribution <- "No";
ae$Drug_2_Attribution[ae$CTC_AE_ATTR_SCALE_1 %in% c("Definite", "Probable", "Possible")] <- "Yes";
ae$Drug_2_Attribution <- as.factor(ae$Drug_2_Attribution);

prior2RE <- list(R = list(V = diag(1), fix = 1), G=list(G1=list(V=1, nu=0.02),
          G2=list(V=1, nu=0.02)));

model1 <- MCMCglmm::MCMCglmm(Drug_1_Attribution ~ AE_SEV_GD + Drug_2_Attribution,
          random=~ae_detail + Subject, family="categorical", data=ae, saveX=TRUE,
          verbose=FALSE, burnin=2000, nitt=10000, thin=10, pr=TRUE, prior=prior2RE);

mcmcglmm_mva <- nice_mcmcglmm(model1, ae);

## End(Not run)
```

`nice_mcmcglmm_icc` *Nice table of intraclass correlation coefficients from MCMCglmm::MCMCglmm() model output*

Description

Nice table of intraclass correlation coefficients from MCMCglmm::MCMCglmm() model output

Usage

```
nice_mcmcglmm_icc(mcmcglmm_object, prob = NULL, decimals = NULL)
```

Arguments

`mcmcglmm_object`	returned output from MCMCglmm()
`prob`	probability for highest posterior density interval, similar to a confidence interval. Default is 0.95 (if provided)
`decimals`	number of decimal places to use in estimates

Value

grouped_df

Examples

```
## Not run:
data(ae)
ae$AE_SEV_GD <- as.numeric(ae$AE_SEV_GD);
ae$Drug_1_Attribution <- 0;
ae$Drug_1_Attribution[ae$CTC_AE_ATTR_SCALE %in% c("Definite", "Probable", "Possible")] <- 1;
ae$Drug_2_Attribution <- 0;
ae$Drug_2_Attribution[ae$CTC_AE_ATTR_SCALE_1 %in% c("Definite", "Probable", "Possible")] <- 1;
prior2RE <- list(R = list(V = diag(1), fix = 1), G=list(G1=list(V=1, nu=0.02),
   G2=list(V=1, nu=0.02)));
model1 <- MCMCglmm::MCMCglmm(Drug_1_Attribution ~ AE_SEV_GD + Drug_2_Attribution,
   random=~ae_detail + Subject, family="categorical", data=ae, saveX=TRUE,
   verbose=FALSE, burnin=2000, nitt=10000, thin=10, pr=TRUE, prior=prior2RE);
mcmcglmm_icc <- nice_mcmcglmm_icc(model1);

## End(Not run)
```

`redcap_data_out` *Combines exported REDCap raw and label .csv files together with data dictionary. Tranforms the exported data into Excel sheets by survey instrument with one row per participant*

Description

Combines exported REDCap raw and label .csv files together with data dictionary. Tranforms the exported data into Excel sheets by survey instrument with one row per participant

Usage

```
redcap_data_out(
  protocol,
  pullDate = NULL,
  subjID,
  subjID_ineligText = NULL,
  subjID_eligPattern = NULL,
  varFilter = NULL,
  varFilter_eligPattern = NULL,
  setWD_files,
  setWD_dataDict = NULL,
  outDir
)
```

Arguments

`protocol`	study protocol name (i.e. Example_Study)
`pullDate`	date of data pull, for example, 2024_01_02 (if provided)
`subjID`	key identifier field(s) for participant ID in data sets

`subjID_ineligText`
: character text that denotes participant IDs to exclude using first key identifier field. For example, c("New Subject") (if provided)

`subjID_eligPattern`
: character text that denotes pattern for participant IDs to include using first key identifier field. For example, c("^Site_A") (if provided)

`varFilter`
: field to use for filtering data (if provided)

`varFilter_eligPattern`
: character text that denotes pattern for filter variable to include, for example, c("^Arm_A") (if provided)

`setWD_files`
: directory where the both raw and label REDCap export .csv files are stored, following the convention for file names of 1_DATA.csv, 1_DATA_LABELs.csv, 2_DATA.csv, 2_DATA_LABELs.csv, etc

`setWD_dataDict`
: directory where the REDCap .csv data dictionary is stored. Make sure that file is saved as basic .csv file in Excel, and not UTF-8. Must contain "DataDictionary" in file name (if provided)

`outDir`
: output directory where the Excel files are saved

Value

two Excel files, one containing variable names and labels and the other containing REDCap survey instrument data by sheet

Examples

```
## Not run:
redcap_data_out(protocol="Example_Study",pullDate="2024_01_03",

## End(Not run)
```

`rm_covsum_nested` *Outputs a nested version of reportRmd::rm_covsum()*

Description

Outputs a nested version of reportRmd::rm_covsum()

Usage

```
rm_covsum_nested(
  data,
  covs,
  maincov = NULL,
  id = NULL,
  caption = NULL,
```

```
  tableOnly = FALSE,
  covTitle = "",
  digits = 1,
  digits.cat = 0,
  nicenames = TRUE,
  IQR = FALSE,
  all.stats = FALSE,
  pvalue = TRUE,
  effSize = TRUE,
  p.adjust = "none",
  unformattedp = FALSE,
  show.tests = TRUE,
  just.nested.pvalue = FALSE,
  nCores = NULL,
  nested.test = NULL,
  nsim = NULL,
  testcont = c("rank-sum test", "ANOVA"),
  testcat = c("Chi-squared", "Fisher"),
  full = TRUE,
  include_missing = FALSE,
  percentage = c("column", "row"),
  dropLevels = TRUE,
  excludeLevels = NULL,
  numobs = NULL,
  markup = TRUE,
  sanitize = TRUE,
  chunk_label
)
```

Arguments

data	dataframe containing data
covs	character vector with the names of columns to include in table
maincov	covariate to stratify table by
id	covariates to nest summary by
caption	character containing table caption (default is no caption)
tableOnly	Logical, if TRUE then a dataframe is returned, otherwise a formatted printed object is returned (default).
covTitle	character with the names of the covariate (predictor) column. The default is to leave this empty for output or, for table only output to use the column name 'Covariate'.
digits	number of digits for summarizing mean data
digits.cat	number of digits for the proportions when summarizing categorical data (default: 0)
nicenames	boolean indicating if you want to replace . and _ in strings with a space

IQR	boolean indicating if you want to display the inter quantile range (Q1,Q3) as opposed to (min,max) in the summary for continuous variables
all.stats	boolean indicating if all summary statistics (Q1,Q3 + min,max on a separate line) should be displayed. Overrides IQR.
pvalue	boolean indicating if you want p-values included in the table
effSize	boolean indicating if you want effect sizes included in the table. Can only be obtained if pvalue is also requested.
p.adjust	p-adjustments to be performed
unformattedp	boolean indicating if you would like the p-value to be returned unformatted (ie not rounded or prefixed with '<'). Best used with tableOnly = T and outTable function.
show.tests	boolean indicating if the type of statistical used should be shown in a column beside the p-values. Ignored if pvalue=FALSE.
just.nested.pvalue	boolean indicating if the just the nested p-value should be shown in a column, and not unnested p-value, unnested statistical tests and effect size. Overrides effSize and show.tests arguments.
nCores	number of cores to use for parallel processing if calculating the nested p-value (if provided).
nested.test	specifies test used for calculating nested p-value from afex::mixed function. Either *parametric bootstrap* method or *likelihood ratio test* method (default: "LRT"). Parametric bootstrap takes longer.
nsim	specifies number of simulations to use for calculating nested p-value with *parametric bootstrap* method used for nested.test (default: 1000).
testcont	test of choice for continuous variables,one of *rank-sum* (default) or *ANOVA*
testcat	test of choice for categorical variables,one of *Chi-squared* (default) or *Fisher*
full	boolean indicating if you want the full sample included in the table, ignored if maincov is NULL
include_missing	Option to include NA values of maincov. NAs will not be included in statistical tests
percentage	choice of how percentages are presented, one of *column* (default) or *row*
dropLevels	logical, indicating if empty factor levels be dropped from the output, default is TRUE.
excludeLevels	a named list of covariate levels to exclude from statistical tests in the form list(varname =c('level1','level2')). These levels will be excluded from association tests, but not the table. This can be useful for levels where there is a logical skip (ie not missing, but not presented). Ignored if pvalue=FALSE.
numobs	named list overriding the number of people you expect to have the covariate
markup	boolean indicating if you want latex markup
sanitize	boolean indicating if you want to sanitize all strings to not break LaTeX
chunk_label	only used if output is to Word to allow cross-referencing

Value

A character vector of the table source code, unless tableOnly=TRUE in which case a data frame is returned

See Also

covsum,fisher.test, chisq.test, wilcox.test, kruskal.test, anova, mixed and outTable

Examples

```
## Not run:
# Example 1
data(ae)
rm_covsum_nested(data = ae, id = c("ae_detail", "Subject"), covs = c("AE_SEV_GD",
"AE_ONSET_DT_INT"), maincov = "CTC_AE_ATTR_SCALE")

# Example 2: set variable labels and other options, save output with markup
data("ae")
lbls <- data.frame(c1=c('AE_SEV_GD','AE_ONSET_DT_INT'),
    c2=c('Adverse event severity grade','Adverse event onset date'))
ae$AE_SEV_GD <- as.numeric(ae$AE_SEV_GD)
ae <- reportRmd::set_labels(ae, lbls)
output_tab <- rm_covsum_nested(data = ae, id = c("ae_detail", "Subject"),
    covs = c("AE_SEV_GD", "AE_ONSET_DT_INT"), maincov = "CTC_AE_ATTR_SCALE",
    testcat = "Fisher", percentage = c("col"), show.tests = FALSE, pvalue = TRUE,
    effSize = FALSE, full = TRUE, IQR = FALSE, nicenames = TRUE, sanitize = TRUE,
    markup = TRUE, include_missing = TRUE, just.nested.pvalue = TRUE,
    tableOnly = TRUE)
cat(reportRmd::outTable(tab=output_tab))
cat(reportRmd::outTable(output_tab, format="html"), file = paste("./man/tables/",
    "output_tab.html", sep=""))
cat(reportRmd::outTable(output_tab, format="latex"), file = paste("./man/tables/",
    "output_tab.tex", sep=""))

## End(Not run)
```

Index