Package 'tibbleOne'

January 28, 2020

Type Package

Title Table One for 'Latex', 'Word', and 'Html' 'R Markdown' Documents

Version 0.1.1

Maintainer Byron Jaeger `<bcjaeger@uab.edu>`

Description Table one is a tabular description of characteristics, e.g., demographics of patients in a clinical trial, presented overall and also stratified by a categorical variable, e.g. treatment group. There are many excellent packages available to create table one. This package focuses on providing table one objects that seamlessly fit into 'R Markdown' analyses.

License GPL-3

Depends R (>= 3.5.0)

Imports tibble (>= 2.1.3),
knitr (>= 1.23),
officer,
flextable,
kableExtra,
forcats,
labelled,
tidyr (>= 1.0.0),
tidyselect,
stringr,
rlang,
vctrs,
purrr,
glue,
stats,
magrittr,
dplyr,
lifecycle

Suggests rmarkdown,
survival,
testthat (>= 2.1.0),
covr

VignetteBuilder knitr

RdMacros lifecycle

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE, roclets = c(``rd'', ``namespace'',
    ``collate''))

RoxygenNote 7.0.2

R topics documented:

---

tibbleOne-package   *tibbleOne: tidy characteristics tables*

Description

In many academic papers, table 1 shows participant characteristics, often stratified by a categorical variable such as treatment group. There are many excellent packages available to compute the numbers in table 1. This package focuses on getting those numbers into a nice format that works with R Markdown. Additionally, there is a fair amount of diversity in researchers' preferred writing mediums. Some may prefer LaTeX, while others want to work in Microsoft Word. Recently, html documents have grown more common for research papers. tibbleOne is meant to be applicable for each of these settings, and should meet the needs of most studies.

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

Author(s)

Maintainer: Byron Jaeger <bcjaeger@uab.edu> (ORCID)

Other contributors:

• Boyi Guo <boyiguo1@uab.edu> [contributor]

---

adapt_round *Adaptive rounding for tables*

Description

Adaptive rounding for tables

Usage

```
adapt_round(x)
```

Arguments

x   a numeric vector

Value

a character vector comprising rounded values.

Examples

```
adapt_round(c(0.12, 10.12, 100.12))
```

---

build_meta *Meta data builder*

Description

creates a dataset that describes the characteristics of another dataset

Usage

```
build_meta(
  data,
  expand_binary_catgs = FALSE,
  add_perc_to_cats = TRUE,
  max_catgs = 10
)
```

Arguments

data   a data frame with any combination of the following attributes: `label`, `unit`,
                `group`, `abbrs`, and `notes`. Columns in the meta data are based on these at-
                tributes.

expand_binary_catgs

                T/F, should all categories be included for binary categorical variables? (This
                only applies to binary variables.)

add_perc_to_cats

        T/F, should categorical variables in Table 1 have a % sign following their label? Note that if the user specifies include_freq = TRUE in tibble_one, then the % symbol may be confusing to readers. However, when include_freq = FALSE, setting this to TRUE should clarify the table.

max_catgs   largest number of categories accepted in a factor variable. A warning message is printed if a factor variable has more categories than max_catg.

Value

A list containing components of data, group_levels, and var_levels. The data component comprises 8 columns:

- variable: variable name - this is the column name of the variable.
- label: variable labels - this is presented in tables
- type: type of variable (numeric or factor)
- unit: units for continuous variables
- group: a group identifier for each variable
- abbr: abbreviations associated with the label of a variable
- note: strings that will be place in tables as a footnote
- labels: labels of variables, including categories of factors.

data is also a tibble.

The group_levels component shows the order that groups will appear in the table, and var_levels shows the order that variables will appear in the table and within groups.

---

pbc_tbl1   *Mayo Clinic Primary Biliary Cirrhosis*

Description

These data contain a subset of the columns found in the pbc data in the survival package. Also, the description below is taken from the description provided by the survival package.

cirrhosis (PBC) of the liver conducted between 1974 and 1984. A total of 424 PBC patients, referred to Mayo Clinic during that ten-year interval, met eligibility criteria for the randomized placebo controlled trial of the drug D-penicillamine. The first 312 cases in the data set participated in the randomized trial and contain largely complete data. The additional 112 cases did not participate in the clinical trial, but consented to have basic measurements recorded and to be followed for survival. Six of those cases were lost to follow-up shortly after diagnosis, so the data here are on an additional 106 cases as well as the 312 randomized participants.

Usage

pbc_tbl1

Format

a data frame with 418 rows and 9 variables.

- age: in years

- sex: male or female

- status death, censor, or transplant.

- trt D-penicillmain, placebo, and not randomized (NA).

- stage: histologic stage of disease (needs biopsy)

- ascites: presence of ascites

- bili: serum bilirunbin (mg/dl)

- edema: no edema, untreated or successfully treated edema, or edema despite diuretic therapy

- albumin: serum albumin (g/dl)

Source

T Therneau and P Grambsch (2000), Modeling Survival Data: Extending the Cox Model, Springer-Verlag, New York. ISBN: 0-387-98784-3.

---

select_labelled   *Select and label variables*

Description

Select and label variables

Usage

```
select_labelled(data, ...)
```

Arguments

data   a data frame

...   name-value pairs of variable labels

Value

a data frame containing the columns indicated by `...`, adorned with attributes based on a user's specified labels.

---

set_variable_labels   *Set variable attributes*

Description

These functions allow you to embed attributes in data so that you only need to think about them once. Some functions may overwrite or delete attributes in data, so it is recommended that you create a meta data object with the [build_meta](#) function after you have set variable labels, groups, notes, abbreviations, and units.

Usage

```
set_variable_labels(data, ...)

set_variable_groups(data, ...)

set_variable_abbrs(data, ...)

set_variable_notes(data, ...)

set_variable_units(data, ...)
```

Arguments

data	a data frame.
...	name-value pairs of variable groups and names (see examples)

Details

- [tibble_one](#) will handle attributes of data automatically, e.g., replacing variable names with variable labels, and placing variable acronyms (i.e., abbreviations) at the bottom of the table in a footnote.

- Use set_variable_labels() to set the values that will represent variables in table one.

- Use set_variable_groups() to change the variables that are listed in the variable categories of table one.

- Use set_variable_notes to add descriptions of variables that will be placed at the bottom of table one as a footnote.

- Use set_variable_abbrs to indicate what acronyms in variable labels mean (see examples).

- Use set_variable_units to indicate the unit of measurement for continuous variables.

For set_variable_labels, names are variables and values are labels. For example, writing gfr_variable = "estimated GFR" as an input to set_variable_labels will set the label for gfr_variable as the indicated string. Since GFR is an acronym, we would also want to use set_variable_abbrs() and say gfr_variable = c("GFR" = "glomerular filtration rate") (see examples).

Value

a [tibble](#), adorned with additional attributes based on user input.

Note

The `set_variable_labels` function in `tibbleOne` is a wrapper of the labelled function, developed by Joseph Larmarange.

Examples

```
df <- data.frame(
  gfr = c(1,2,3),
  sbp = c(3,2,1)
)

df <- set_variable_labels(df,
  gfr = 'Estimated GFR',
  sbp = 'Systolic BP'
)

df <- set_variable_units(df,
  gfr = 'mL/min/1.73 m2',
  sbp = 'mm Hg'
)

df <- set_variable_abbrs(df,
  gfr = c("GFR" = "glomerular filtration rate", "min" = 'minute'),
  sbp = c("BP" = "blood pressure")
)

df <- set_variable_notes(df,
  sbp = "blood pressure was measured by trained personnel"
)

build_meta(df)
```

---

tibble_one   *Tidy characteristics data*

Description

Stable

Table one is a tabular description of characteristics, e.g., demographics of patients in a clinical trial, presented overall and also stratified by a categorical variable, e.g. treatment group.

Usage

```
tibble_one(
  data,
  formula = NULL,
  meta_data = NULL,
  row_vars = NULL,
  strat = NULL,
  by = NULL,
```

```
    specs_table_vals = NULL,
    specs_table_tests = NULL,
    include_pval = FALSE,
    expand_binary_catgs = FALSE,
    include_freq = FALSE,
    add_perc_to_cats = TRUE
)
```

Arguments

`data`	a data frame
`formula`	an optional formula object. The left hand side of the formula should be blank. The right hand side of the formula should contain row variables for the table. The 'l' symbol can be used to include stratifying variables. If this option is used, no more than two stratifying variables should be used, and they must be separated by a * symbol. If formula is used, the strat, by, and row_vars inputs are ignored.
`meta_data`	a meta data frame. If unspecified, a meta data frame will be created using `data`.
`row_vars`	a character vector indicating column names of row variables in the table. If unspecified, all columns are used.
`strat`	a character value indicating the column name in data that will be used to stratify the table
`by`	a character value indicating the column name in data that will be used to split the table into groups, prior to stratification.
`specs_table_vals`	
	named vector of character values. Names should be variables, while values should be specs. Valid specs are 'mean' and 'median' (see examples).
`specs_table_tests`	
	named vector of character values. Names should be variables, while values should be specs. Valid specs are 'params' or 'noparm' (see examples).
`include_pval`	T/F, should the table include a column for p-values? If p-values are included, factor variables are handled using chi-square tests, continuous variables are handled using t-tests or ANOVA, depending on the number of categories in the table stratification.
`expand_binary_catgs`	
	T/F, should all categories be included for binary categorical variables? (This only applies to binary variables.)
`include_freq`	T/F, should frequency values be included for categorical variables?
`add_perc_to_cats`	
	T/F, should categorical variable labels be appended with a percent sign?

Value

a [tibble](#) containing summary values that describe characteristics of observations in data , which can subsequently be sent to different modes of output (see [to_word](#) and [to_kable](#)).

Examples

```
data("pbc_tbl1")
# report median albumin instead of mean
# use kruskal wallis test for albumin
```

```
tibble_one(
  pbc_tbl1,
  formula = ~ . | trt,
  include_freq = FALSE,
  include_pval = TRUE,
  specs_table_vals = c(albumin = 'median'),
  specs_table_tests = c(albumin = 'nopars')
)
```

---

to_kable   *Pass tibble_one to kable*

Description

Stable

Tabular summaries of characteristics (i.e., table one) are generally presented in a table with columns that describe the overall sample and subsets of the sample designated by a grouping variable. In addition, it is expected that variable labels and units will be indicated, and footnotes will be placed at the bottom of the table with full descriptions of variables and abbreviations that appear in the table. This function automates these tasks.

Usage

```
to_kable(
  object,
  use_groups = TRUE,
  indent_groups = FALSE,
  footnote_notation = "symbol",
  include_1st_header = TRUE,
  include_2nd_header = TRUE,
  include_3rd_header = TRUE,
  bold_headers = TRUE,
  ...
)
```

Arguments

object	a tibble_one object
use_groups	T/F, should rows be grouped?
indent_groups	T/F, should entries within groups be indented? (this has no effect if use_groups is FALSE)
footnote_notation	
	character value indicating footnote symbols to use in tables. Eligible values are symbol, number, and alphabet.
include_1st_header	
	T/F, should bottom header be included?
include_2nd_header	
	T/F, should middle header be included?
include_3rd_header	
	T/F, should top header be included?

bold_headers	T/F, should header labels be printed in bold?
...	Arguments passed on to [knitr::kable](#)

> format   A character string. Possible values are latex, html, markdown, pandoc, and rst; this will be automatically determined if the function is called within **knitr**; it can also be set in the global option knitr.table.format. If format is a function, it must return a character string.
>
> caption   The table caption.
>
> label   The table reference label. By default, the label is obtained from knitr::[opts_current](#)$get(

Value

A character vector of the table source code, i.e., code that can be presented in 'R Markdown' documents.

---

to_word	*pass tibble_one to flextable*

Description

Stable

Tabular summaries of characteristics (i.e., table one) are generally presented in a table with columns that describe the overall sample and subsets of the sample designated by a grouping variable. In addition, it is expected that variable labels and units will be indicated, and footnotes will be placed at the bottom of the table with full descriptions of variables and abbreviations that appear in the table. This function automates these tasks.

Usage

```
to_word(
  object,
  font_size = 11,
  use_groups = TRUE,
  indent_groups = TRUE,
  footnote_notation = "symbol",
  include_1st_header = TRUE,
  include_2nd_header = TRUE,
  include_3rd_header = TRUE
)
```

Arguments

object	a tibble_one object
font_size	the size of font in the table.
use_groups	T/F, should rows be grouped?
indent_groups	T/F, should entries within groups be indented? (this has no effect if use_groups is FALSE)
footnote_notation	
	character value indicating footnote symbols to use in tables. Eligible values are symbol, number, and alphabet.

```
include_1st_header
```
     T/F, should bottom header be included?

```
include_2nd_header
```
     T/F, should middle header be included?

```
include_3rd_header
```
     T/F, should top header be included?

Value

a flextable.

Index