Package: tablet (via r-universe)

October 26, 2024

Type Package

Title Tabulate Descriptive Statistics in Multiple Formats

Version 0.6.11

Maintainer Tim Bergsma <bergsmat@gmail.com>

BugReports https://github.com/bergsmat/tablet/issues

Description Creates a table of descriptive statistics for factor and numeric columns in a data frame. Displays these by groups, if any. Highly customizable, with support for 'html' and 'pdf' provided by 'kableExtra'. Respects original column order, column labels, and factor level order. See ?tablet.data.frame and vignettes.

License GPL-3

Encoding UTF-8

Imports dplyr (>= 1.0.2), yamlet (>= 0.10.21), rlang, tidyr, kableExtra (>= 0.9.0), spork (>= 0.2.7), magrittr, fs, reactable

RoxygenNote 7.3.2

VignetteBuilder knitr

Suggests knitr, rmarkdown, boot, testthat, shiny, shinyFiles, haven, yaml, sortable, latexpdf, tinytex, tools, csv, xtable, shinyAce, R.utils

NeedsCompilation no

Author Tim Bergsma [aut, cre]

Repository CRAN

Date/Publication 2024-10-25 21:40:02 UTC

Contents

`as_kable.tablet` *Coerce Tablet to Kable*

Description

Renders a tablet. Calls `kbl` and implements special features like grouped columns.

Usage

```
## S3 method for class 'tablet'
as_kable(
  x,
  ...,
  booktabs = TRUE,
  escape = FALSE,
  escape_latex = tablet::escape_latex,
  escape_html = function(x, ...) x,
  variable = " ",
  col.names = NA,
  linebreak = TRUE,
  align = "c",
  double_escape = FALSE,
  linebreaker = "\n",
  pack_rows = list(escape = escape),
  secondary = FALSE
)
```

Arguments

`x`	`tablet`
`...`	passed to `kbl`
`booktabs`	passed to `kbl`
`escape`	passed to `kbl`; defaults FALSE to allow header linebreaks
`escape_latex`	a function to pre-process column names and content if 'escape' is FALSE (e.g., manual escaping, latex only); default `escape_latex`
`escape_html`	a function to pre-process column names and content if 'escape' is FALSE (e.g., manual escaping, html only)
`variable`	a column name for the variables
`col.names`	passed to `kbl` after any linebreaking

linebreak	whether to invoke linebreak for column names
align	passed to linebreak for column names
double_escape	passed to linebreak for column names
linebreaker	passed to linebreak for column names in latex; for html, linebreaker is replaced with
pack_rows	named list passed to pack_rows for finer control of variable names
secondary	passed to escape_latex

Details

See also tablet.data.frame. Column _tablet_name must inherit 'character' and by default (in a latex render context) its values will eventually be processed by escape_latex. Thus, if _tablet_name is of class 'latex' it will be handled by method escape_latex.latex (which tries not to re-escape metacharacters).

Value

like kbl

Examples

```
library(boot)
library(dplyr)
library(magrittr)
library(haven)
library(yamlet)
library(spork)
melanoma %>%
  select(-time, -year) %>%
  mutate(sex = factor(sex), ulcer = factor(ulcer)) %>%
  group_by(status) %>%
  tablet %>%
  as_kable

x <- system.file(package = 'tablet', 'shiny-examples/mesa/data/adsl.sas7bdat')
x %<>% read_sas %>% data.frame
decorations(x)

# calculate BMI by assuming all males are 1.75 m, all females 1.63 cm
x %<>% mutate(height = ifelse(sex == 'F', 1.63, 1.75))
x %<>% mutate(bmi = signif(digits = 3, weight / (height^2)))
x %<>% filter(saffl == 'Y')
x %<>% select(trt01a, age, sex, weight, bmi)
x %<>% redecorate('
trt01a: [ Treatment, [ Placebo, TRT 10 mg, TRT 20 mg ]]
age:    [ Age, year ]
sex:    [ Sex, [ Female: F, Male: M ]]
weight: [ Body Weight, kg ]
bmi:    [ Index_body mass, kg/m^2 ]
')
```

```
x %<>% resolve
x %<>% group_by(trt01a)

x %>% tablet %>% as_kable

# supply default and unit-conditional latex titles
x %<>% modify(title = concatenate(as_latex(as_spork(c(.data$label)))))
x %<>% modify(
age, weight, bmi,
  title = concatenate(
    sep = '',  # default ok in pdf
    as_latex(
      as_spork(
        c(.data$label, ' (', .data$units, ')')
      )
    )
  )
)
x %>% tablet %>% as_kable
```

`as_tablet.data.frame` *Coerce data.frame to tablet*

Description

Coerces data.frame to tablet. Checks format and assigns the class. See `tablet.data.frame`.

Usage

```
## S3 method for class 'data.frame'
as_tablet(x, ...)
```

Arguments

`x`	data.frame
`...`	passed arguments

Value

tablet

See Also

Other tablet: `as_tablet()`, `header_rows()`, `header_rows.tablet()`, `tablet.tablette()`, `tablette.tablet()`

header_rows.tablet *Identify Header Rows for tablet*

Description

Identifies header rows for tablet.

Usage

```
## S3 method for class 'tablet'
header_rows(x, ...)
```

Arguments

x	tablet
...	ignored

Value

integer: indices for those rows representing headers

See Also

Other tablet: as_tablet(), as_tablet.data.frame(), header_rows(), tablet.tablette(), tablette.tablet()

mesa *Drag-and-drop Descriptive Statistics*

Description

Generate a table of descriptive statistics by selecting columns from a file. Currently supported formats include *.xpt, *.sas7bdat, and *.csv. Launch the application using mesa() and use the interface to select a data file, such as 'mtcars.xpt' under 'examples/') (or select configuration file 'mtcars.conf' under 'examples/'). Then classify the columns of interest to generate the corresponding displays.

Usage

```
mesa(launch.browser = TRUE, display.mode = "normal", ...)
```

Arguments

launch.browser	passed to runApp
display.mode	passed to runApp
...	passed to runApp

Details

Currently,

* xpt files are read using the defaults for `read.xport`,

* sas7bdat files are read using the defaults for `read_sas`, and

* csv files are read using the defaults for `as.csv`.

If a file in the same directory has a corresponding base name but a `.yaml` extension, it is treated as metadata and an attempt is made to apply it to the internal version of the data. This file will not be over-written, but it WILL be constructed if missing. You can hand-edit it to supply metadata. See `?yamlet` for format; see the Variables tab for an easy interface.

This is a metadata-driven application. Columns in the data that are *not* in the metadata will be ignored, and columns in the metadata that are *not* in the data will be constructed (maybe *all* of them).

The `mtcars` datasets in the 'examples' volume is from **datasets**.

Value

used for side effects: launches shiny application shinyWidgets',

---

recap.knitr_kable *Recap knitr_kable.*

Description

Recaps knitr_kable. Specifically, it replaces the first non-tabled caption with multicolumn text. The intent is to prevent repeat bookmarks when generating pdf.

Usage

```
## S3 method for class 'knitr_kable'
recap(x, cols = NULL, pos = "c", ...)
```

Arguments

`x`	object of dispatch
`cols`	number of columns to span; guesses `ncol(x)` by default
`pos`	position of text: 'l','c' (default), or 'r'
`...`	ignored

See Also

Other recap: `recap()`

`tablet.data.frame` *Generate a Tablet for Data Frame*

Description

Generates a 'tablet': a summary table of formatted statistics for factors (is.factor()) and numerics (is.numeric()) in x, with and without grouping variables (if present, see `group_by`). Column names represent finest level of grouping, distinguished by attribute 'nest' (the values of higher other groups, if any) along with the 'all' column for ungrouped statistics. Column attribute 'n' indicates relevant corresponding observation count. Input should not have column names beginning with '_tablet'.

Usage

```
## S3 method for class 'data.frame'
tablet(
 x,
 ...,
 na.rm = FALSE,
 all = 'All',
 fun = list(
  sum ~ sum(x,  na.rm = TRUE),
  pct ~ signif(digits = 3,     sum / n * 100        ),
  ave ~ signif(digits = 3,    mean(x,  na.rm = TRUE)),
  std ~ signif(digits = 3,      sd(x,  na.rm = TRUE)),
  med ~ signif(digits = 3,  median(x,  na.rm = TRUE)),
  min ~ signif(digits = 3,     min(x,  na.rm = TRUE)),
  max ~ signif(digits = 3,     max(x,  na.rm = TRUE))
 ),
 fac = list(
  ` ` ~ sum + ' (' + pct + '%' + ')'
 ),
 num = list(
  `Mean (SD)` ~ ave + ' (' + std + ')',
  `Median (range)` ~ med + ' (' + min + ', ' + max + ')'
  ),
 lab = list(
  lab ~ name + '\n(N = ' + n + ')'
 ),
 na.rm_fac = na.rm,
 na.rm_num = na.rm,
 exclude_fac = NULL,
 exclude_name = NULL,
 all_levels = FALSE
)
```

Arguments

x	data.frame (possibly grouped)
...	substitute formulas for elements of fun, fac, num, lab
na.rm	whether to remove NA in general
all	a column name for ungrouped statistics; can have length zero to suppress ungrouped column
fun	default aggregate functions expressed as formulas
fac	a list of formulas to generate widgets for factors
num	a list of formulas to generate widgets for numerics
lab	a list of formulas to generate label attributes for columns (see details)
na.rm_fac	whether to drop NA 'factor' observations; passed to gather as na.rm, interacts with exclude_fac
na.rm_num	whether to drop NA numeric observations; passed to gather as na.rm
exclude_fac	which factor levels to exclude; see factor (exclude)
exclude_name	whether to drop NA values of column name (for completeness); passed to gather
all_levels	whether to supply records for unobserved levels

Details

Arguments 'fun', 'fac', 'num', and 'lab' are lists of two-sided formulas that are evaluated in an environment where '+' expresses concatenation (for character elements). The values of LHS should be unique across all four lists. 'fun' is a list of aggregate statistics that have access to N (number of original records), n (number of group members), and x (the numeric observations, or 1 for each factor value). Aggregate statistics generated by 'fun' are available for use in 'fac' and 'num' which create visualizations thereof ('widgets'). Column-specific attributes are available to elements of 'lab', including the special attribute name (the current column name). For 'lab' only, if the RHS succeeds, it becomes the label attribute of the corresponding output column. 'lab' is used here principally to support annotation of *output* columns; if *input* columns have attributes 'label' or 'title' (highest priority) those will have been already substituted for default column names at the appropriate positions in the output.

Missingness of observations (and to a lesser extent, levels of grouping variables) merits special consideration. Be aware that na.rm_fac and na.rm_num take their defaults from na.rm. Furthermore, na.rm_fac may interact with exclude_fac, which is passed to factor as exclude. The goal is to support all possible ways of expressing or ignoring missingness. That said, if aggregate functions are removing NA, the values of arguments beginning with 'na.rm' or 'exclude' may not matter.

Column 1 of output is character. Its values are typically the names of the original columns that were factor or numeric but not in groups(x). If the first of these had a label attribute or (priority) a title attribute with class 'latex', then column 1 is assigned the class 'latex' as well. It makes sense therefore to be consistent across input columns regarding the presence or not of a 'latex' label or title. By default, as_kable.tablet dispatches class-specific methods for escape_latex.

Similarly, row 1 of output is typically character. As of version 0.6.6, if any of the grouping variables inherits 'latex', then the return value of tablet.data.frame() has an attribute 'name_class' with value 'latex'.

Value

'tablet' A tablet is a special case of data.frame with grouped rows and columns.

* There is always one level of row groups.
* There can be any number of column groups, including zero.
* All columns are character (as tested by `is.character()`).
* The first column has empty strings that represent the last non-empty value. It can be class 'latex' or 'character'.
* Leading element(s) of first column are deliberately blank (one space character) and correspond to header rows. See `header_rows`.
* The second column represents group-specific property names. It is populated always and only where column 1 is not.
* All other columns represent group-specific property values; elements before the first non-empty group value represent nested headers.
* Header values may be repeated.
* Header values may be empty strings, representing the last non-empty value to the left, or single spaces, which are deliberately blank.
* Internally, character NA is equivalent to an empty string.

See Also

`as_kable.tablet`

Examples

```
library(boot)
library(dplyr)
library(magrittr)
melanoma %>%
  select(-time, -year) %>%
  mutate(sex = factor(sex), ulcer = factor(ulcer)) %>%
  group_by(status) %>%
  tablet
```

Index