Package 'tablet'

August 21, 2021

Type Package

Title Tabulate Descriptive Statistics in Multiple Formats

Version 0.4.0

Author Tim Bergsma

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), rlang, tidyr, kableExtra (>= 0.9.0), DT, spork (>= 0.2.0)

RoxygenNote 7.1.1

VignetteBuilder knitr

Suggests knitr, magrittr, rmarkdown, yamlet (>= 0.6.9), boot, testthat, shiny, shinyFiles, fs, haven, yaml, sortable, latexpdf, tools, csv, xtable

NeedsCompilation no

Repository CRAN

Date/Publication 2021-08-21 04:30:02 UTC

R topics documented:

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()
)
```

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)
escape_html	a function to pre-process column names 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, newline is replaced with
pack_rows	named list passed to pack_rows for finer control of variable names

Value

like kbl

Examples

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

---

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

`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 ~ signif(digits = 3,     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
)
```

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

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.

Value

'tablet', with columns for each combination of groups, and:

`_tablet_name`	observation identifier
`_tablet_level`	factor level (or special value 'numeric' for numerics)
`_tablet_stat`	the LHS of formulas in 'fac' and 'num'
`All (or value of 'all' argument)`	ungrouped results
`_tablet_sort`	sorting column

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