

Package ‘summarytools’

April 29, 2025

Type Package

Title Tools to Quickly and Neatly Summarize Data

Version 1.1.4

Author Dominic Comtois [aut, cre]

Maintainer Dominic Comtois <dominic.comtois@gmail.com>

Language en-US

Description Data frame summaries, cross-tabulations, weight-enabled frequency tables and common descriptive (univariate) statistics in concise tables available in a variety of formats (plain ASCII, Markdown and HTML). A good point-of-entry for exploring data, both for experienced and new R users.

Imports base64enc, checkmate, dplyr, grDevices, htmltools, lubridate, magick, matrixStats, methods, pander, pryr, rapportools, stats, tcltk, tibble, tidyr, utils

Suggests forcats, formatR, haven, kableExtra, knitr, magrittr, rmarkdown, rstudioapi, backports

Depends R (>= 3.5)

VignetteBuilder knitr

LazyData true

License GPL-2

URL <https://github.com/dcomtois/summarytools>

BugReports <https://github.com/dcomtois/summarytools/issues>

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Repository CRAN

Date/Publication 2025-04-29 17:40:02 UTC

Contents

summarytools-package	2
cleartmp	3
ctable	4
define_keywords	7
descr	10
dfSummary	12
examens	17
exams	17
format_number	18
freq	19
label	22
print.list	23
print.stby	24
print.summarytools	25
stby	28
st_css	29
st_options	30
tabagisme	35
tb	36
tobacco	37
unlabel	38
use_custom_lang	38
view	39
what.is	40
zap_attr	41
Index	43

summarytools-package *Tools to Quickly and Neatly Summarize Data*

Description

summarytools is a collection of functions which neatly and quickly summarize numerical and categorical data. Data frame summaries, frequency tables and cross-tabulations, as well as common descriptive (univariate) statistics can be produced in a straightforward manner. Users with little to no prior R programming experience but who are familiar with popular commercial statistical software such as SAS, SPSS and Stata will feel right at home.

Details

These are the four core functions:

dfSummary Extensive yet legible data frame summaries.

freq Frequency tables supporting weights and displaying proportions of valid and of total data, including cumulative proportions.

descr All common univariate descriptive stats applied to a single vector or to all numerical vectors contained in a data frame.

ctable Cross-tabulations for pairs of categorical variables – accepting both numerical and character vectors, as well as factors. Choose between *Total*, *Columns* or *Rows* proportions, and optionally display chi-square statistic (with corresponding p-value), odds ratio, as well as risk ratio with flexible confidence intervals.

Choice of output formats:

plain ascii Ideal when showing results in the R console.

rmarkdown Perfect for writing short papers or presentations.

html A format very well integrated in *RStudio* – but will work with any Web browser. Use the [view](#) function to display results directly in *RStudio*'s viewer, or in your preferred Web browser.

Author(s)

Maintainer: Dominic Comtois <dominic.comtois@gmail.com>

See Also

Useful links:

- <https://github.com/dcomtois/summarytools>
- Report bugs at <https://github.com/dcomtois/summarytools/issues>

cleartmp

Delete Temporary Html Files

Description

Delete temporary files created when using generic print method with `method='browser'` or `method='viewer'`, or when calling `view()` function.

Usage

```
cleartmp(all = TRUE, silent = FALSE, verbose = FALSE)
```

Arguments

<code>all</code>	Logical. When TRUE (default), all temporary summarytools files are deleted. When FALSE, only the latest file is.
<code>silent</code>	Logical. Hide confirmation messages (FALSE by default).
<code>verbose</code>	Logical. Display a message for every file that is deleted. FALSE by default.

Note

Given that all temporary files are deleted automatically when an R session is ended, this function is an overkill in most circumstances. It could however be useful in server-type setups.

Author(s)

Dominic Comtois, <dominic.comtois@gmail.com>

 ctable

Cross-Tabulation

Description

Cross-tabulation for a pair of categorical variables with either row, column, or total proportions, as well as marginal sums. Works with numeric, character, as well as factor variables.

Usage

```
ctable(
  x,
  y,
  prop = st_options("ctable.prop"),
  useNA = "ifany",
  totals = st_options("ctable.totals"),
  style = st_options("style"),
  round.digits = st_options("ctable.round.digits"),
  justify = "right",
  plain.ascii = st_options("plain.ascii"),
  headings = st_options("headings"),
  display.labels = st_options("display.labels"),
  split.tables = Inf,
  na.val = st_options("na.val"),
  rev = "none",
  dnn = c(substitute(x), substitute(y)),
  chisq = FALSE,
  OR = FALSE,
  RR = FALSE,
  weights = NA,
  rescale.weights = FALSE,
  ...
)
```

Arguments

x	First categorical variable - values will appear as row names.
y	Second categorical variable - values will appear as column names.
prop	Character. Indicates which proportions to show: "r" (rows, default), "c" (columns), "t" (total), or "n" (none). Default value can be changed using st_options , option <code>ctable.prop</code> .
useNA	Character. One of "ifany" (default), "no", or "always". This argument is passed on 'as is' to table , or adapted for xtabs when weights are used.


```
# Show html results in browser
print(ct, method = "browser")

# Save results to html file
print(ct, file = "ct_gender_smoker.html")

# Save results to text file
print(ct, file = "ct_gender_smoker.txt")

## End(Not run)
```

define_keywords *Modify Keywords Used In Outputs*

Description

As an alternative to [use_custom_lang](#), this function allows temporarily modifying the pre-defined terms in the outputs.

Usage

```
define_keywords(..., ask = TRUE, file = NA)
```

Arguments

...	One or more pairs of keywords and their new values see <i>Details</i> for the complete list of existing keywords.
ask	Logical. When 'TRUE' (default), a dialog box comes up to ask whether to save the edited values in a csv file for later use.
file	Character. Path and name of custom language file to be saved. This comma delimited file can be reused by calling use_custom_lang . Must have .csv extension.

Details

On systems with GUI capabilities, a window will pop-up when calling `define_keywords()` without any parameters, allowing the modification of the *custom* column. The changes will be active as long as the package is loaded. When the edit window is closed, a dialog will pop up, prompting the user to save the modified set of keywords in a custom csv language file that can later be used with [use_custom_lang](#).

Here is the full list of modifiable keywords.

title.freq main heading for `freq()`

title.freq.weighted main heading for `freq()` (weighted)

title.ctable main heading for `ctable()`

title.ctable.weighted main heading `ctable()` (weighted)

title.ctable.row indicates what proportions are displayed
title.ctable.col indicates what proportions are displayed
title.ctable.tot indicates what proportions are displayed
title.descr main heading for descr()
title.descr.weighted main heading for descr() (weighted)
title.dfSummary main heading for dfSummary()
n heading item used in descr()
dimensions heading item used in dfSummary()
duplicates heading item used in dfSummary()
data.frame heading item (all functions)
label heading item (all functions) & column name in dfSummary()
variable heading item (all functions) & column name in dfSummary()
group heading item (all functions when used with stby())
by heading item for descr() when used with stby()
weights heading item - descr() & freq()
type heading item for freq()
logical heading item - type in freq()
character heading item - type in freq()
numeric heading item - type in freq()
factor heading item - type in freq()
factor.ordered heading item - type in freq()
date heading item - type in freq()
datetime heading item - type in freq()
freq column name in freq()
pct column name in freq() when report.nas=FALSE
pct.valid.f column name in freq()
pct.valid.cum column name in freq()
pct.total column name in freq()
pct.total.cum column name in freq()
pct.cum column name in freq()
valid column name in freq() and dfSummary() & column content in dfSummary()
invalid column content in dfSummary() (emails)
total column grouping in freq(), html version
mean row name in descr()
sd.long row name in descr()
sd cell content (dfSummary)
min row name in descr()

q1 row name in descr() - 1st quartile
med row name in descr()
q3 row name in descr() - 3rd quartile
max row name in descr()
mad row name in descr() - Median Absolute Deviation
iqr row name in descr() - Inter-Quartile Range
cv row name in descr() - Coefficient of Variation
skewness row name in descr()
se.skewness row name in descr() - Std. Error for Skewness
kurtosis row name in descr()
n.valid row name in descr() - Count of non-missing values
pct.valid row name in descr() - pct. of non-missing values
no column name in dfSummary() - position of column in the data frame
stats.values column name in dfSummary()
freqs.pct.valid column name in dfSummary()
graph column name in dfSummary()
missing column name in dfSummary()
distinct.value cell content in dfSummary() - singular form
distinct.values cell content in dfSummary() - plural form
all.nas cell content in dfSummary() - column has only NAs
all.empty.str cell content in dfSummary() - column has only empty strings
all.empty.str.nas cell content in dfSummary() - col. has only NAs and empty strings
no.levels.defined cell content in dfSummary() - factor has no levels defined
int.sequence cell content in dfSummary()
rounded cell content in dfSummary() - note appearing in Stats/Values
others cell content in dfSummary() - nbr of values not displayed
codes cell content in dfSummary() - When UPC codes are detected
mode cell content in dfSummary() - mode = most frequent value
med.short cell content in dfSummary() - median (shortened term)
start cell content in dfSummary() - earliest date for date-type cols
end cell content in dfSummary() - latest date for data-type cols
emails cell content in dfSummary()
generated.by footnote content
version footnote content
date.fmt footnote - date format (see [strptime](#))

Note

Setting a keyword starting with “title.” to NA or to empty string causes the main title to disappear altogether, which might be desired in some circumstances (when generating a table of contents, for instance).

Examples

```
## Not run:
define_keywords(n = "Nb. Obs.")

## End(Not run)
```

 descr

Univariate Statistics for Numerical Data

Description

Calculates mean, sd, min, Q1^{*}, median, Q3^{*}, max, MAD, IQR^{*}, CV, skewness^{*}, SE.skewness^{*}, and kurtosis^{*} on numerical vectors. (^{*}) Not available when using sampling weights.

Usage

```
descr(
  x,
  var = NULL,
  stats = st_options("descr.stats"),
  na.rm = TRUE,
  round.digits = st_options("round.digits"),
  transpose = st_options("descr.transpose"),
  order = "sort",
  style = st_options("style"),
  plain.ascii = st_options("plain.ascii"),
  justify = "r",
  headings = st_options("headings"),
  display.labels = st_options("display.labels"),
  split.tables = 100,
  weights = NULL,
  rescale.weights = FALSE,
  ...
)
```

Arguments

x A numerical vector or a data frame.

var Unquoted expression referring to a specific column in x. Provides support for piped function calls (e.g. `my_df |> descr(my_var)`).

stats	Character. Which stats to produce. Either “all” (default), “fivenum”, “common” (see <i>Details</i>), or a selection of : “mean”, “sd”, “min”, “q1”, “med”, “q3”, “max”, “mad”, “iqr”, “cv”, “skewness”, “se.skewness”, “kurtosis”, “n.valid”, “n”, and “pct.valid”. Can be set globally via <code>st_options</code> , option “descr.stats”. See <i>Details</i> .
na.rm	Logical. Argument to be passed to statistical functions. Defaults to TRUE.
round.digits	Numeric. Number of significant digits to display. Defaults to 2. Can be set globally with <code>st_options</code> .
transpose	Logical. Make variables appears as columns, and stats as rows. Defaults to FALSE. Can be set globally with <code>st_options</code> , option “descr.transpose”.
order	Character. When analyzing more than one variable, this parameter determines how to order variables. Valid values are “sort” (or simply “s”), “preserve” (or “p”), or a vector containing all variable names in the desired order. Defaults to “sort”.
style	Character. Style to be used by <code>pander</code> . One of “simple” (default), “grid”, “rmarkdown”, or “jira”. Can be set globally with <code>st_options</code> .
plain.ascii	Logical. <code>pander</code> argument; when TRUE (default), no markup characters will be used (useful when printing to console). If <code>style = 'rmarkdown'</code> is specified, value is set to FALSE automatically. Can be set globally using <code>st_options</code> .
justify	Character. Alignment of numbers in cells; “l” for left, “c” for center, or “r” for right (default). Has no effect on <i>html</i> tables.
headings	Logical. Set to FALSE to omit heading section. Can be set globally via <code>st_options</code> . TRUE by default.
display.labels	Logical. Show variable / data frame labels in heading section. Defaults to TRUE. Can be set globally with <code>st_options</code> .
split.tables	Character. <code>pander</code> argument that specifies how many characters wide a table can be. 100 by default.
weights	Numeric. Vector of weights having same length as <i>x</i> . NULL (default) indicates that no weights are used.
rescale.weights	Logical. When set to TRUE, a global constant is apply to make the total count equal <code>nrow(x)</code> . FALSE by default.
...	Additional arguments passed to <code>pander</code> or <code>format</code> .

Details

Since version 1.1, the *stats* argument can be set in a more flexible way; keywords (*all*, *common*, *fivenum*) can be combined with single statistics, or their “negation”. For instance, using `stats = c("all", "-q1", "-q3")` would show **all except q1 and q3**.

For further customization, you could redefine any preset in the following manner: `.st_env$descr.stats$common <- c("mean", "sd", "n")`. Use caution when modifying `.st_env`, and reload the package if errors ensue. Changes are temporary and will not persist across R sessions.

Value

An object having classes “*matrix*” and “*summarytools*” containing the statistics, with extra attributes useful to other functions/methods.

Author(s)

Dominic Comtois, <dominic.comtois@gmail.com>

Examples

```
data("exams")

# All stats (default behavior) for all numerical variables
descr(exams)

# Show only "common" statistics, plus "n"
descr(exams, stats = c("common", "n"))

# Selection of statistics, transposing the results
descr(exams, stats = c("mean", "sd", "min", "max"), transpose = TRUE)

# Rmarkdown-ready
descr(exams, plain.ascii = FALSE, style = "rmarkdown")

# Grouped statistics
data("tobacco")
with(tobacco, stby(BMI, gender, descr, check.nas = FALSE))

# Grouped statistics in tidy table:
tb(with(tobacco, stby(BMI, age.gr, descr, stats = "common")))

## Not run:
# Show in Viewer (or browser if not in RStudio)
view(descr(exams))

# Save to html file with title
print(descr(exams),
      file = "descr_exams.html",
      report.title = "BMI by Age Group",
      footnote = "<b>Schoolyear:</b> 2018-2019<br/><b>Semester:</b> Fall")

## End(Not run)
```

Description

Summary of a data frame consisting of: variable names and types, labels if any, factor levels, frequencies and/or numerical summary statistics, barplots/histograms, and valid/missing observation counts and proportions.

Usage

```
dfSummary(
  x,
  round.digits = 1,
  varnumbers = st_options("dfSummary.varnumbers"),
  class = st_options("dfSummary.class"),
  labels.col = st_options("dfSummary.labels.col"),
  valid.col = st_options("dfSummary.valid.col"),
  na.col = st_options("dfSummary.na.col"),
  graph.col = st_options("dfSummary.graph.col"),
  graph.magnif = st_options("dfSummary.graph.magnif"),
  style = st_options("dfSummary.style"),
  plain.ascii = st_options("plain.ascii"),
  justify = "l",
  na.val = st_options("na.val"),
  col.widths = NA,
  headings = st_options("headings"),
  display.labels = st_options("display.labels"),
  max.distinct.values = 10,
  trim.strings = FALSE,
  max.string.width = 25,
  split.cells = 40,
  split.tables = Inf,
  tmp.img.dir = st_options("tmp.img.dir"),
  keep.grp.vars = FALSE,
  silent = st_options("dfSummary.silent"),
  ...
)
```

Arguments

x	A data frame.
round.digits	Number of significant digits to display. Defaults to 1. Does not affect proportions, which always show 1 digit.
varnumbers	Logical. Show variable numbers in the first column. Defaults to TRUE. Can be set globally with <code>st_options</code> , option “dfSummary.varnumbers”.
class	Logical. Show data classes in <i>Variable</i> column. TRUE by default.
labels.col	Logical. If TRUE, variable labels (as defined with rapportools , Hmisc or summarytools ’ label functions, among others) will be displayed. TRUE by default, but the <i>labels</i> column is only shown if a label exists for at least one column. Can be set globally with <code>st_options</code> , option “dfSummary.labels.col”.

valid.col	Logical. Include column indicating count and proportion of valid (non-missing) values. TRUE by default; can be set globally with st_options , option “dfSummary.valid.col”.
na.col	Logical. Include column indicating count and proportion of missing (NA) values. TRUE by default; can be set globally with st_options , option “dfSummary.na.col”.
graph.col	Logical. Display barplots/histograms column. TRUE by default; can be set globally with st_options , option “dfSummary.graph.col”.
graph.magnif	Numeric. Magnification factor for graphs column. Useful if the graphs show up too large (then use a value such as .75) or too small (use a value such as 1.25). Must be positive. Defaults to 1. Can be set globally with st_options , option “dfSummary.graph.magnif”.
style	Character. Argument used by pander . Defaults to “multiline”. The only other valid option is “grid”. Style “rmarkdown” will fallback to “multiline”.
plain.ascii	Logical. pander argument; when TRUE, no markup characters will be used (useful when printing to console). Defaults to TRUE. Set to FALSE when in context of markdown rendering. To change the default value globally, see st_options .
justify	String indicating alignment of columns; one of “l” (left) “c” (center), or “r” (right). Defaults to “l”.
na.val	Character. For factors and character vectors, consider this value as NA. Ignored if there are actual NA values. NULL by default.
col.widths	Numeric or character. Vector of column widths. If numeric, values are assumed to be numbers of pixels. Otherwise, any CSS-supported units can be used. NA by default, meaning widths are calculated automatically.
headings	Logical. Set to FALSE to omit headings. To change this default value globally, see st_options .
display.labels	Logical. Should data frame label be displayed in the title section? Default is TRUE. To change this default value globally, see st_options .
max.distinct.values	The maximum number of values to display frequencies for. If variable has more distinct values than this number, the remaining frequencies will be reported as a whole, along with the number of additional distinct values. Defaults to 10.
trim.strings	Logical; for character variables, should leading and trailing white space be removed? Defaults to FALSE. See <i>details</i> section.
max.string.width	Limits the number of characters to display in the frequency tables. Defaults to 25.
split.cells	A numeric argument passed to pander . It is the number of characters allowed on a line before splitting the cell. Defaults to 40.
split.tables	pander argument which determines the maximum width of a table. Keeping the default value (Inf) is recommended.
tmp.img.dir	Character. Directory used to store temporary images when rendering dfSummary() with ‘method = “pander”’, ‘plain.ascii = TRUE’ and ‘style = “grid”’. See <i>Details</i> .

<code>keep.grp.vars</code>	Logical. When using <code>group_by</code> , keep rows corresponding to grouping variable(s) in output table. When <code>FALSE</code> (default), variable numbers still reflect the ordering in the full data frame (in other words, some numbers will be skipped in the variable number column).
<code>silent</code>	Logical. Hide console messages. <code>FALSE</code> by default. To change this value globally, see <code>st_options</code> .
<code>...</code>	Additional arguments passed to <code>pander</code> .

Details

The default value `plain.ascii = TRUE` is intended to facilitate interactive data exploration. When using the package for reporting with `rmarkdown`, make sure to set this option to `FALSE`.

When `trim.strings` is set to `TRUE`, trimming is done **before calculating frequencies**, be aware that those will be impacted accordingly.

Specifying `tmp.img.dir` allows producing results consistent with pandoc styling while also showing `png` graphs. Due to the fact that in Pandoc, column widths are determined by the length of cell contents **even if said content is merely a link to an image**, using standard R temporary directory to store the images would cause columns to be exceedingly wide. **A shorter path is needed.** On Mac OS and Linux, using `"/tmp"` is a sensible choice, since this directory is cleaned up automatically on a regular basis. On Windows however, there is no such convenient directory, so the user has to choose a directory and cleanup the temporary images manually after the document has been rendered. Providing a relative path such as `"img"`, omitting `"/"`, is recommended. The maximum length for this parameter is set to 5 characters. It can be set globally with `st_options` (e.g.: `st_options(tmp.img.dir = ".")`).

It is possible to **control which statistics are shown** in the *Stats / Values* column. For this, see the *Details* and *Examples* sections of `st_options`.

Value

A data frame with additional class `summarytools` containing as many rows as there are columns in `x`, with attributes to inform `print` method. Columns in the output data frame are:

No Number indicating the order in which column appears in the data frame.

Variable Name of the variable, along with its class(es).

Label Label of the variable (if applicable).

Stats / Values For factors, a list of their values, limited by the `max.distinct.values` parameter. For character variables, the most common values (in descending frequency order), also limited by `max.distinct.values`. For numerical variables, common univariate statistics (mean, std. deviation, min, med, max, IQR and CV).

Freqs (% of Valid) For factors and character variables, the frequencies and proportions of the values listed in the previous column. For numerical vectors, number of distinct values, or frequency of distinct values if their number is not greater than `max.distinct.values`.

Text Graph An ASCII histogram for numerical variables, and ASCII barplot for factors and character variables.

Graph An html encoded graph, either barplot or histogram.

Valid Number and proportion of valid values.

Missing Number and proportion of missing (NA and NAN) values.

Note

Several packages provide functions for defining *variable labels*, **summarytools** being one of them. Some packages (*Hmisc* in particular) employ special classes for labelled objects, but **summarytools** doesn't use nor look for any such classes.

Author(s)

Dominic Comtois, <dominic.comtois@gmail.com>

See Also

[label](#), [print.summarytools](#)

Examples

```
data("tobacco")
saved_x11_option <- st_options("use.x11")
st_options(use.x11 = FALSE)
dfSummary(tobacco)

# Exclude some of the columns to reduce table width
dfSummary(tobacco, varnumbers = FALSE, valid.col = FALSE)

# Limit number of categories to be displayed for categorical data
dfSummary(tobacco, max.distinct.values = 5, style = "grid")

# Using stby()
stby(tobacco, tobacco$gender, dfSummary)

st_options(use.x11 = saved_x11_option)

## Not run:

# Show in Viewer or browser - no capital V in view(); stview() is also
# available in case of conflicts with other packages)
view(dfSummary(iris))

# Rmarkdown-ready
dfSummary(tobacco, style = "grid", plain.ascii = FALSE,
          varnumbers = FALSE, valid.col = FALSE, tmp.img.dir = "./img")

# Using group_by()
tobacco %>% group_by(gender) %>% dfSummary()

## End(Not run)
```

examens

Bulletin de notes (donnees simulees)

Description

Jeu de donnees simulees contenant les notes de 30 etudiants, avec les colonnes suivantes:

- etudiant Nom de l'etudiant.
- sexe Variable categorielle (facteur). Deux niveaux: "Fille", "Garcon".
- francais Note en francais (numerique).
- math Note en maths (numerique).
- geographie Note en geographie (numerique).
- histoire Note en histoire (numerique).
- economie Note en economie (numerique).
- anglais Note en anglais (numerique).

Usage

`data(examens)`

Format

Un data frame de 30 rangees et 8 colonnes

Details

Donnees simulees. Les notes de chaque etudiant sont centrees autour d'une moyenne personnelle et ecart-type randomises.

A copy of this dataset is **available in English** under the name "exams".

exams

Report Cards - Simulated Data

Description

A simulated dataset with grades for hypothetical 30 students, with the following variables:

- student Student's name.
- gender Factor with 2 levels: "Girl", "Boy".
- french French Grade (numerical).
- math Math Grade (numerical).
- geography Geography Grade (numerical).
- history History Grade (numerical).
- economics Economics Grade (numerical).
- english English Grade (numerical).

Usage

```
data(exams)
```

Format

A data frame with 30 rows and 8 variables

Details

All names and grades are simulated. Grades for each student are centered around a personal randomized average and standard deviation.

A copy of this dataset is also **available in French** under the name “examens”.

format_number	<i>format_number</i>
---------------	----------------------

Description

Used internally (not exported) to apply all relevant formatting. It is documented here only because it can be used when setting the `dfSummary.custom.1` and `dfSummary.custom.1` options.

Usage

```
format_number(x, round.digits, ...)
```

Arguments

<code>x</code>	A numerical value to be formatted.
<code>round.digits</code>	Numerical. Number of decimals to show. Used to define both <code>digits</code> and <code>nsmall</code> when calling <code>format</code> .
<code>...</code>	Any other formatting instruction that is compatible with <code>format</code> .

Examples

```
## Not run:
format_number(IQR(column_data, na.rm = TRUE), round.digits)
format_number(IQR(column_data, na.rm = TRUE), decimal.mark = ",")

## End(Not run)
```

Description

Displays weighted or unweighted frequencies, including <NA> counts and proportions.

Usage

```
freq(
  x,
  var = NULL,
  round.digits = st_options("round.digits"),
  order = "default",
  style = st_options("style"),
  plain.ascii = st_options("plain.ascii"),
  justify = "default",
  cumul = st_options("freq.cumul"),
  totals = st_options("freq.totals"),
  report.nas = st_options("freq.report.nas"),
  rows = numeric(),
  missing = "",
  na.val = st_options("na.val"),
  display.type = TRUE,
  display.labels = st_options("display.labels"),
  headings = st_options("headings"),
  weights = NA,
  rescale.weights = FALSE,
  ...
)
```

Arguments

x	Factor, vector, or data frame.
var	Optional unquoted variable name. Provides support for piped function calls (e.g. <code>my_df %>% freq(my_var)</code>).
round.digits	Numeric. Number of significant digits to display. Defaults to 2. Can be set globally with st_options .
order	Character. Ordering of rows in frequency table; “name” (default for non-factors), “level” (default for factors), or “freq” (from most frequent to less frequent). To invert the order, place a minus sign before or after the word. “-freq” will thus display the items starting from the lowest in frequency to the highest, and so forth.
style	Character. Style to be used by pander . One of “simple” (default), “grid”, “rmarkdown”, or “jira”. Can be set globally with st_options .

<code>plain.ascii</code>	Logical. <code>pander</code> argument; when TRUE, no markup characters will be used (useful when printing to console). Defaults to TRUE unless <code>style = 'rmarkdown'</code> , in which case it will be set to FALSE automatically. Can be set globally with <code>st_options</code> .
<code>justify</code>	String indicating alignment of columns. By default (“default”), “right” is used for text tables and “center” is used for <i>html</i> tables. You can force it to one of “left”, “center”, or “right”.
<code>cumul</code>	Logical. Set to FALSE to hide cumulative proportions from results. TRUE by default. To change this value globally, see <code>st_options</code> .
<code>totals</code>	Logical. Set to FALSE to hide totals from results. TRUE by default. To change this value globally, see <code>st_options</code> .
<code>report.nas</code>	Logical. Set to FALSE to turn off reporting of missing values. To change this default value globally, see <code>st_options</code> .
<code>rows</code>	Character or numeric vector allowing subsetting of the results. The order given here will be reflected in the resulting table. If a single string is used, it will be used as a regular expression to filter row names.
<code>missing</code>	Text to display in NA cells. Defaults to “”.
<code>na.val</code>	Character. For factors and character vectors, consider this value as NA. Ignored if there are actual NA values or if it matches no value / factor level in the data. NULL by default.
<code>display.type</code>	Logical. Should variable type be displayed? Default is TRUE.
<code>display.labels</code>	Logical. Should variable / data frame labels be displayed? Default is TRUE. To change this default value globally, see <code>st_options</code> .
<code>headings</code>	Logical. Set to FALSE to omit heading section. Can be set globally via <code>st_options</code> .
<code>weights</code>	Vector of weights; must be of the same length as <code>x</code> .
<code>rescale.weights</code>	Logical parameter. When set to TRUE, the total count will be the same as the unweighted <code>x</code> . FALSE by default.
<code>...</code>	Additional arguments passed to <code>pander</code> .

Details

The default `plain.ascii = TRUE` option is there to make results appear cleaner in the console. To avoid `rmarkdown` rendering problems, this option is automatically set to FALSE whenever `style = "rmarkdown"` (unless `plain.ascii = TRUE` is made explicit in the function call).

Value

A frequency table of class `matrix` and `summarytools` with added attributes used by `print` method.

Note

The data type represents the `class` in most cases.

Author(s)

Dominic Comtois, <dominic.comtois@gmail.com>

See Also

[table](#)

Examples

```
data(tobacco)
freq(tobacco$gender)
freq(tobacco$gender, totals = FALSE)

# Ignore NA's, don't show totals, omit headings
freq(tobacco$gender, report.nas = FALSE, totals = FALSE, headings = FALSE)

# In .Rmd documents, use the two following arguments, minimally
freq(tobacco$gender, style="rmarkdown", plain.ascii = FALSE)

# Grouped Frequencies
with(tobacco, stby(diseased, smoker, freq))
(fr_smoker_by_gender <- with(tobacco, stby(smoker, gender, freq)))

# Print html Source
print(fr_smoker_by_gender, method = "render", footnote = NA)

# Order by frequency (+ to -)
freq(tobacco$age.gr, order = "freq")

# Order by frequency (- to +)
freq(tobacco$age.gr, order = "-freq")

# Use the 'rows' argument to display only the 10 most common items
freq(tobacco$age.gr, order = "freq", rows = 1:10)

## Not run:
# Display rendered html results in RStudio's Viewer
# notice 'view()' is NOT written with capital V
# If working outside RStudio, Web browser is used instead
# A temporary file is stored in temp dir
view(fr_smoker_by_gender)

# Display rendered html results in default Web browser
# A temporary file is stored in temp dir here too
print(fr_smoker_by_gender, method = "browser")

# Write results to text file (.txt, .md, .Rmd) or html file (.html)
print(fr_smoker_by_gender, method = "render", file = "fr_smoker_by_gender.md")
print(fr_smoker_by_gender, method = "render", file = "fr_smoker_by_gender.html")

## End(Not run)
```

label

Get or Set Variable or Data Frame Labels

Description

Assigns a label to a vector or data frame, or returns value stored in the object's label attribute (or NA if none exists).

Usage

```
label(x, all = FALSE, fallback = FALSE, simplify = FALSE)
label(x) <- value
llabel(x, all = TRUE, fallback = FALSE, simplify = FALSE)
```

Arguments

x	An R object to extract labels from.
all	Logical. When x is a data frame, setting this argument to TRUE will make the function return all variable labels. By default, its value is FALSE, so that if x is a data frame, it is the data frame's label itself that will be returned.
fallback	a logical value indicating if labels (returned values) should fallback to object name(s). Defaults to FALSE.
simplify	When x is a data frame and all = TRUE, coerce results to a vector and remove NA's. Default is FALSE.
value	String to be used as label. To clear existing labels, use NA or NULL.

Details

The wrapper function llabel was named that way to avoid conflicting with base function [labels](#).

Value

A single character vector if all = FALSE (default), or a named list if all = TRUE (named vector when using simplify = TRUE).

Note

Loosely based on Gergely Daróczy's [label](#) function.

Author(s)

Dominic Comtois, <dominic.comtois@gmail.com>

print.list

Print Method for Objects of Class "list"

Description

Displays a list comprised of summarytools objects created with `lapply`.

Usage

```
## S3 method for class 'list'
print(x, method = "pander", file = "",
      append = FALSE, report.title = NA, table.classes = NA,
      bootstrap.css = st_options('bootstrap.css'),
      custom.css = st_options('custom.css'), silent = FALSE,
      footnote = st_options('footnote'), collapse = 0,
      escape.pipe = st_options('escape.pipe'), ...)
```

Arguments

<code>x</code>	A <i>summarytools</i> object, created by one of the four core functions (<code>freq</code> , <code>descr</code> , <code>ctable</code> , or <code>dfSummary</code>).
<code>method</code>	Character. One of “pander”, “viewer”, “browser”, or “render”. Default value for the <code>print()</code> method is “pander”; for <code>view()/stview()</code> , default is “viewer” if session is running in <i>RStudio</i> , “browser” otherwise. The main use for “render” is in <i>R Markdown</i> documents.
<code>file</code>	Character. File name to write output to. Defaults to “”.
<code>append</code>	Logical. Append output to existing file (specified using the <i>file</i> argument). FALSE by default.
<code>report.title</code>	Character. For <i>html</i> reports, this goes into the <code><title></code> tag. When left to NA (default), the first line of the heading section is used (e.g.: “Data Frame Summary”).
<code>table.classes</code>	Character. Additional <i>html</i> classes to assign to output tables. <i>Bootstrap css</i> classes can be used. User-defined classes (see the <i>custom.css</i> argument) are also specified here. See <i>details</i> section. NA by default.
<code>bootstrap.css</code>	Logical. When generating an <i>html</i> document, include the “ <code>includes/stylesheets/bootstrap.min.css</code> ” file content inside a <code><style type="text/css"></code> tag in the document’s <code><head></code> . TRUE by default. Can be set globally with <code>st_options</code> .
<code>custom.css</code>	Character. Path to a custom <i>.css</i> file. Classes defined in this must also appear in the <code>table.classes</code> parameter in order to be applied to the table(s). Can be set globally with <code>st_options</code> . NA by default.
<code>silent</code>	Logical. Set to TRUE to hide console messages (e.g.: ignored variables or NaN to NA transformations). FALSE by default.

footnote	Character. Text to display just after <i>html</i> output tables. The default value (“ <i>default</i> ”) produces a two-line footnote indicating the package’s name and version, the R version, and the current date. Has no effect on <i>ascii</i> or <i>markdown</i> content. Can contain standard <i>html</i> tags. Set to NA to omit. Can be set globally with st_options .
collapse	Numeric. 0 by default. Set to 1 to make <code>freq()</code> sections collapsible (when clicking on the variable name). Future versions might provide alternate collapsing options.
escape.pipe	Logical. Set to TRUE when <code>style="grid"</code> and <code>file</code> argument is supplied if the intent is to generate a text file that can be converted to other formats using <i>Pandoc</i> . Can be set globally with st_options .
...	Additional arguments used to override attributes stored in the object, or to change formatting via format or pander . See <i>Details</i> .

Details

This function is there only for cases where the object to be printed was created with [lapply](#), as opposed to the recommended functions for creating grouped results ([stby](#) and [group_by](#)).

print.stby

Print Method for Objects of Class “stby”

Description

Displays a list comprised of *summarytools* objects created with `stby`.

Usage

```
## S3 method for class 'stby'
print(x, method = "pander", file = "",
      append = FALSE, report.title = NA, table.classes = NA,
      bootstrap.css = st_options('bootstrap.css'),
      custom.css = st_options('custom.css'), silent = FALSE,
      footnote = st_options('footnote'),
      escape.pipe = st_options('escape.pipe'), ...)
```

Arguments

x	A <i>summarytools</i> object, created by one of the four core functions (freq , descr , ctable , or dfSummary).
method	Character. One of “pander”, “viewer”, “browser”, or “render”. Default value for the <code>print()</code> method is “pander”; for <code>view()/stview()</code> , default is “viewer” if session is running in <i>RStudio</i> , “browser” otherwise. The main use for “render” is in <i>R Markdown</i> documents.
file	Character. File name to write output to. Defaults to “”.

append	Logical. Append output to existing file (specified using the <i>file</i> argument). FALSE by default.
report.title	Character. For <i>html</i> reports, this goes into the <title> tag. When left to NA (default), the first line of the heading section is used (e.g.: “Data Frame Summary”).
table.classes	Character. Additional <i>html</i> classes to assign to output tables. <i>Bootstrap</i> <i>css</i> classes can be used. User-defined classes (see the <i>custom.css</i> argument) are also specified here. See <i>details</i> section. NA by default.
bootstrap.css	Logical. When generating an <i>html</i> document, include the “ <i>includes/stylesheets/bootstrap.min.css</i> ” file content inside a <style type="text/css"> tag in the document’s <head>. TRUE by default. Can be set globally with st_options .
custom.css	Character. Path to a custom <i>.css</i> file. Classes defined in this must also appear in the <i>table.classes</i> parameter in order to be applied to the table(s). Can be set globally with st_options . NA by default.
silent	Logical. Set to TRUE to hide console messages (e.g.: ignored variables or NaN to NA transformations). FALSE by default.
footnote	Character. Text to display just after <i>html</i> output tables. The default value (“ <i>default</i> ”) produces a two-line footnote indicating the package’s name and version, the R version, and the current date. Has no effect on <i>ascii</i> or <i>markdown</i> content. Can contain standard <i>html</i> tags. Set to NA to omit. Can be set globally with st_options .
escape.pipe	Logical. Set to TRUE when <i>style</i> ="grid" and <i>file</i> argument is supplied if the intent is to generate a text file that can be converted to other formats using <i>Pandoc</i> . Can be set globally with st_options .
...	Additional arguments used to override attributes stored in the object, or to change formatting via format or pander . See <i>Details</i> .

```
print.summarytools    print.summarytools
```

Description

Display *summarytools* objects in the console, in Web Browser or in *RStudio*’s Viewer, or write content to file.

Usage

```
## S3 method for class 'summarytools'
print(x, method = "pander", file = "",
      append = FALSE, report.title = NA, table.classes = NA,
      bootstrap.css = st_options('bootstrap.css'),
      custom.css = st_options('custom.css'), silent = FALSE,
      footnote = st_options('footnote'), max.tbl.height = Inf,
      collapse = 0, escape.pipe = st_options("escape.pipe"), ...)
```

Arguments

x	A <i>summarytools</i> object, created by one of the four core functions (freq , descr , cTable , or dfSummary).
method	Character. One of “pander”, “viewer”, “browser”, or “render”. Default value for the <code>print()</code> method is “pander”; for <code>view()/stview()</code> , default is “viewer” if session is running in <i>RStudio</i> , “browser” otherwise. The main use for “render” is in <i>R Markdown</i> documents.
file	Character. File name to write output to. Defaults to “”.
append	Logical. Append output to existing file (specified using the <i>file</i> argument). FALSE by default.
report.title	Character. For <i>html</i> reports, this goes into the <code><title></code> tag. When left to NA (default), the first line of the heading section is used (e.g.: “Data Frame Summary”).
table.classes	Character. Additional <i>html</i> classes to assign to output tables. <i>Bootstrap css</i> classes can be used. User-defined classes (see the <i>custom.css</i> argument) are also specified here. See <i>details</i> section. NA by default.
bootstrap.css	Logical. When generating an <i>html</i> document, include the “ <i>includes/stylesheets/bootstrap.min.css</i> ” file content inside a <code><style type="text/css"></code> tag in the document’s <code><head></code> . TRUE by default. Can be set globally with st_options .
custom.css	Character. Path to a custom <i>.css</i> file. Classes defined in this must also appear in the <code>table.classes</code> parameter in order to be applied to the table(s). Can be set globally with st_options . NA by default.
silent	Logical. Set to TRUE to hide console messages (e.g.: ignored variables or NaN to NA transformations). FALSE by default.
footnote	Character. Text to display just after <i>html</i> output tables. The default value (“ <i>default</i> ”) produces a two-line footnote indicating the package’s name and version, the R version, and the current date. Has no effect on <i>ascii</i> or <i>markdown</i> content. Can contain standard <i>html</i> tags. Set to NA to omit. Can be set globally with st_options .
max.tbl.height	Numeric. Maximum table height <i>in pixels</i> allowed in rendered <code>dfSummary()</code> tables. When this argument is used, results will show up in a <code><div></code> with the specified height and a scroll bar. Intended to be used in <i>Rmd</i> documents with <code>method = "render"</code> . Inf by default.
collapse	Numeric. 0 by default. Set to 1 to make <code>freq()</code> sections collapsible (when clicking on the variable name). Future versions might provide alternate collapsing options.
escape.pipe	Logical. Set to TRUE when <code>style="grid"</code> and <code>file</code> argument is supplied if the intent is to generate a text file that can be converted to other formats using <i>Pandoc</i> . Can be set globally with st_options .
...	Additional arguments used to override attributes stored in the object, or to change formatting via format or pander . See <i>Details</i> .

Details

Ascii and *markdown* tables are generated using [pander](#).

The following arguments can be used to override formatting attributes stored in the object:

- `style`
- `round.digits` (except for *dfSummary* objects)
- `plain.ascii`
- `justify`
- `split.tables`
- `headings`
- `display.labels`
- `varnumbers` (*dfSummary* objects only)
- `labels.col` (*dfSummary* objects only)
- `graph.col` (*dfSummary* objects only)
- `valid.col` (*dfSummary* objects only)
- `na.col` (*dfSummary* objects only)
- `col.widths` (*dfSummary* objects only)
- `keep.grp.vars` (*dfSummary* objects only)
- `report.nas` (*freq* objects only)
- `display.type` (*freq* objects only)
- `missing` (*freq* objects only)
- `totals` (*freq* and *ctable* objects)
- `caption` (*freq* and *ctable* objects)

The following arguments can be used to override heading elements:

- `Data.frame`
- `Data.frame.label`
- `Variable`
- `Variable.label`
- `Group`
- `date`
- `Weights` (*freq* & *descr* objects)
- `Data.type` (*freq* objects only)
- `Row.variable` (*ctable* objects only)
- `Col.variable` (*ctable* objects only)

Value

NULL when `method="pander"`; A file path returned invisibly when `method="viewer"` or `"browser"`. In the latter case, the file path is also passed to `shell.exec` (*Windows*) or `system` (**nix*), causing the document to be opened in default Web browser.

Author(s)

Dominic Comtois, <dominic.comtois@gmail.com>

References

[Summarytools on GitHub List of pander options Bootstrap Cascading Stylesheets](#)

See Also

[pander](#)

Examples

```
## Not run:
data(tobacco)
view(dfSummary(tobacco), footnote = NA)

## End(Not run)
data(exams)
print(freq(exams$gender), style = 'markdown')
print(descr(exams), headings = FALSE)
```

stby

Obtain Grouped Statistics With summarytools

Description

An adaptation base R's [by](#) function, designed to optimize the results' display.

Usage

```
stby(data, INDICES, FUN, ..., useNA = FALSE)
```

Arguments

data	an R object, normally a data frame, possibly a matrix.
INDICES	a grouping variable or a list of grouping variables, each of length <code>nrow(data)</code> .
FUN	a function to be applied to (usually data-frame) subsets of data.
...	Further arguments to FUN.
useNA	Make NA a valid grouping value in INDICES variable(s). Set to FALSE explicitly to eliminate message.

Details

When the grouping variable(s) contain NA values, the base `:by` function (as well as `summarytools` versions prior to 1.1.0) ignores corresponding groups. Version 1.1.0 allows setting `useNA = TRUE` to make new groups using NA values on the grouping variable(s), just as `dplyr::group_by` does.

When NA values are detected and `useNA = FALSE`, a message is displayed; to disable this message, set `check.nas = FALSE`.

Value

An object of classes “list” and “summarytools”, giving results for each subset.

See Also

[by](#), [group_by](#)

Examples

```
data("tobacco")
with(tobacco, stby(data = BMI, INDICES = gender, FUN = descr,
  check.nas = FALSE))
with(tobacco, stby(data = smoker, INDICES = gender, freq, useNA = TRUE))
with(tobacco, stby(data = list(x = smoker, y = diseased),
  INDICES = gender, FUN = ctable, useNA = TRUE))
```

 st_css

*Include **summarytools**' css Into Active Document*

Description

Generate the *css* needed by **summarytools** in *html* documents.

Usage

```
st_css(main = TRUE, global = FALSE, bootstrap = FALSE, style.tag = TRUE, ...)
```

Arguments

<code>main</code>	Logical. Include <i>summarytools.css</i> file. TRUE by default. This will affects only summarytools objects, for one exception: two properties of the <code>img</code> tag are redefined to have <code>background-color: transparent</code> and <code>border: 0</code> .
<code>global</code>	Logical. Include the additional <i>summarytools-global.css</i> file, which affects all content in the document. Provides control over objects that were not <i>html-rendered</i> ; in particular, table widths and vertical alignment are modified to improve layout. FALSE by default.
<code>bootstrap</code>	Logical. Include <i>bootstrap.min.css</i> . FALSE by default.
<code>style.tag</code>	Logical. Include the opening and closing <code><style></code> tags. TRUE by default.
<code>...</code>	Character. Path to additional <i>css</i> file(s) to include.

Details

Typically the function is called right after the initial setup chunk of an *R markdown* document, in a chunk having options `echo=FALSE` and `results="asis"`.

Value

The *css* file(s) content silently as a character vector, and prints (using `cat()`) the content.

Author(s)

Dominic Comtois, <dominic.comtois@gmail.com>

st_options

Query and set summarytools global options

Description

To list all *summarytools* global options, call without arguments. To display the value of one or several options, enter the name(s) of the option(s) in a character vector as sole argument. To **reset** all options, use single unnamed argument 'reset' or \emptyset .

Usage

```
st_options(  
  option = NULL,  
  value = NULL,  
  style = "simple",  
  plain.ascii = TRUE,  
  round.digits = 2,  
  headings = TRUE,  
  footnote = "default",  
  display.labels = TRUE,  
  na.val = NULL,  
  bootstrap.css = TRUE,  
  custom.css = NA_character_,  
  escape.pipe = FALSE,  
  char.split = 12,  
  freq.cumul = TRUE,  
  freq.totals = TRUE,  
  freq.report.nas = TRUE,  
  freq.ignore.threshold = 25,  
  freq.silent = FALSE,  
  ctable.prop = "r",  
  ctable.totals = TRUE,  
  ctable.round.digits = 1,  
  ctable.silent = FALSE,  
  descr.stats = "all",
```

```

descr.transpose = FALSE,
descr.silent = FALSE,
dfSummary.style = "multiline",
dfSummary.varnumbers = TRUE,
dfSummary.class = TRUE,
dfSummary.labels.col = TRUE,
dfSummary.valid.col = TRUE,
dfSummary.na.col = TRUE,
dfSummary.graph.col = TRUE,
dfSummary.graph.magnif = 1,
dfSummary.silent = FALSE,
dfSummary.custom.1 = expression(paste(paste0(trs("iqr"), " (", trs("cv"), ") : "),
  format_number(IQR(column_data, na.rm = TRUE), round.digits), " (",
  format_number(sd(column_data, na.rm = TRUE)/mean(column_data, na.rm = TRUE),
  round.digits), ")"), collapse = "", sep = "")),
dfSummary.custom.2 = NA,
tmp.img.dir = NA_character_,
subtitle.emphasis = TRUE,
lang = "en",
use.x11 = TRUE
)

```

Arguments

option	option(s) name(s) to query (optional). Can be a single string or a vector of strings to query multiple values.
value	The value you wish to assign to the option specified in the first argument. This is for backward-compatibility, as all options can now be set via their own parameter. That is, instead of <code>st_options('plain.ascii', FALSE)</code> , use <code>st_options(plain.ascii = FALSE)</code> .
style	Character. One of “simple” (default), “rmarkdown”, or “grid”. Does not apply to <code>dfSummary</code> .
plain.ascii	Logical. <code>pander</code> argument; when TRUE, no markup characters will be used (useful when printing to console). TRUE by default, but when <code>style = 'rmarkdown'</code> , it is automatically set to FALSE. To override this behavior, <code>plain.ascii = TRUE</code> must be specified in the function call.
round.digits	Numeric. Defaults to 2.
headings	Logical. Set to FALSE to remove all headings from outputs. Only the tables will be printed out, except when <code>by</code> or <code>lapply</code> are used. In that case, the variable or the group will still appear before each table. TRUE by default.
footnote	Character. When the default value “default” is used, the package name & version, as well as the R version number are displayed below <i>html</i> outputs. Set to NA to omit the footnote, or provide a custom string. Applies only to <i>html</i> outputs.
display.labels	Logical. TRUE by default. Set to FALSE to omit data frame and variable labels in the headings section.

na.val	Character. For factors and character vectors, consider this value as NA. Ignored if there are actual NA values or if it matches no value / factor level in the data. NULL by default.
bootstrap.css	Logical. Specifies whether to include <i>Bootstrap css</i> in <i>html</i> reports' <i>head</i> section. Defaults to TRUE. Set to FALSE when using the “render” method inside a shiny app to avoid interacting with the app's layout.
custom.css	Character. Path to an additional, user-provided, CSS file. NA by default.
escape.pipe	Logical. Set to TRUE if Pandoc conversion is your goal and you have unsatisfying results with grid or multiline tables. FALSE by default.
char.split	Numeric. Maximum number of characters allowed in a column heading for <code>descr</code> and <code>ctable html</code> outputs. Any variable name having more than this number of characters will be split on two or more lines. Defaults to 12.
freq.cumul	Logical. Corresponds to the <code>cumul</code> parameter of <code>freq</code> . TRUE by default.
freq.totals	Logical. Corresponds to the <code>totals</code> parameter of <code>freq</code> . TRUE by default.
freq.report.nas	Logical. Corresponds to the <code>display.nas</code> parameter of <code>freq</code> . TRUE by default.
freq.ignore.threshold	Numeric. Number of distinct values above which numerical variables are ignored when calling <code>freq</code> with a whole data frame as main argument. Defaults to 25.
freq.silent	Logical. Hide console messages. FALSE by default.
ctable.prop	Character. Corresponds to the <code>prop</code> parameter of <code>ctable</code> . Defaults to “r” (row).
ctable.totals	Logical. Corresponds to the <code>totals</code> parameter of <code>ctable</code> . TRUE by default.
ctable.round.digits	Numeric. Defaults to 1.
ctable.silent	Logical. Hide console messages. FALSE by default.
descr.stats	Character. Corresponds to the <code>stats</code> parameter of <code>descr</code> . Defaults to “all”.
descr.transpose	Logical. Corresponds to the <code>transpose</code> parameter of <code>descr</code> . FALSE by default.
descr.silent	Logical. Hide console messages. FALSE by default.
dfSummary.style	Character. “multiline” by default. Set to “grid” for <i>R Markdown</i> documents.
dfSummary.varnumbers	Logical. In <code>dfSummary</code> , display variable numbers in the first column. Defaults to TRUE.
dfSummary.class	Logical. Show data classes in Name column. TRUE by default. variable numbers in the first column. Defaults to TRUE.
dfSummary.labels.col	Logical. In <code>dfSummary</code> , display variable labels Defaults to TRUE.
dfSummary.valid.col	Logical. In <code>dfSummary</code> , include column indicating count and proportion of valid (non-missing). TRUE by default.

<code>dfSummary.na.col</code>	Logical. In <code>dfSummary</code> , include column indicating count and proportion of missing (NA) values. TRUE by default.
<code>dfSummary.graph.col</code>	Logical. Display barplots / histograms column in <code>dfSummary</code> <i>html</i> reports. TRUE by default.
<code>dfSummary.graph.magnif</code>	Numeric. Magnification factor, useful if <code>dfSummary</code> graphs show up too large (then use a value between 0 and 1) or too small (use a value > 1). Must be positive. Default to 1.
<code>dfSummary.silent</code>	Logical. Hide console messages. FALSE by default.
<code>dfSummary.custom.1</code>	Expression. First of two optional expressions which once evaluated will populate lines 3+ of the ‘Stats / Values’ cell when column data is numerical and has more distinct values than allowed by the <code>max.distinct.values</code> parameter. By default, it contains the expression which generates the ‘IQR (CV) : ...’ line. To reset it back to this default value, use <code>st_options(dfSummary.custom.1 = "default")</code> . See <i>Details</i> and <i>Examples</i> sections for more.
<code>dfSummary.custom.2</code>	Expression. Second the two optional expressions which once evaluated will populate lines 3+ of the ‘Stats / Values’ cell when the column data is numerical and has more distinct values than allowed by the ‘ <code>max.distinct.values</code> ’ parameter. NA by default. See <i>Details</i> and <i>Examples</i> sections for more.
<code>tmp.img.dir</code>	Character. Directory used to store temporary images. See <i>Details</i> section of <code>dfSummary</code> . NA by default.
<code>subtitle.emphasis</code>	Logical. Controls the formatting of the “subtitle” (the <i>data frame</i> or <i>variable</i> name, depending on context. When TRUE (default), “h4” is used, while with FALSE, “bold” / “strong” is used. Hence the default value gives it stronger emphasis.
<code>lang</code>	Character. A 2-letter code for the language to use in the produced outputs. Currently available languages are: ‘en’, ‘es’, ‘fr’, ‘pt’, ‘ru’, and ‘tr’.
<code>use.x11</code>	Logical. TRUE by default. In console-only environments, setting this to FALSE will prevent errors occurring when <code>dfSummary</code> tries to generate <i>html</i> “Base64-encoded” graphs.

Details

The `dfSummary.custom.1` and `dfSummary.custom.2` options must be defined as expressions. In the expression, use the `column_data` variable name to refer to data. Assume the type to be numerical (real or integer). The expression must paste together both the labels (short name for the statistic(s) being displayed) and the statistics themselves. Although `round` can be used, a better alternative is to call the internal `format_number`, which uses `format` to apply all relevant formatting that is active within the call to `dfSummary`. For keywords having a translated term, the `trs()` internal function can be used (see *Examples*).

Note

To learn more about summarytools options, see `vignette("introduction", "summarytools")`.

Examples

```
# show all summarytools global options
st_options()

# show a specific option
st_options("round.digits")

# show two (or more) options
st_options(c("plain.ascii", "style", "footnote"))

## Not run:
# set one option
st_options(plain.ascii = FALSE)

# set one options, legacy way
st_options("plain.ascii", FALSE)

# set several options
st_options(plain.ascii = FALSE,
           style       = "rmarkdown",
           footnote    = NA)

# reset all
st_options('reset')
# ... or
st_options(0)

# Define custom dfSummary stats
st_options(dfSummary.custom.1 = expression(
  paste(
    "Q1 - Q3 :",
    format_number(
      quantile(column_data, probs = .25, type = 2,
              names = FALSE, na.rm = TRUE), round.digits
    ),
    "-",
    format_number(
      quantile(column_data, probs = .75, type = 2,
              names = FALSE, na.rm = TRUE), round.digits
    ),
    collapse = ""
  )
))

dfSummary(iris)

# Set back to default value
st_options(dfSummary.custom.1 = "default")
```

```
## End(Not run)
```

tabagisme

Usage du tabac et etat de sante (donnees simulees)

Description

Jeu de donnees simulees de 1000 sujets, avec les colonnes suivantes:

- sexe Variable categorielle (facteur), 2 niveaux: “F” et “M”. Environ 500 chacun.
- age Numerique.
- age.gr Groupe d’age - variable categorielle, 4 niveaux.
- IMC Indice de masse corporelle (numerique).
- fumeur Variable categorielle, 2 niveaux (“Oui” / “Non”).
- cigs.par.jour Nombre de cigarettes fumees par jour (numerique).
- malade Variable categorielle, 2 niveaux (“Oui” / “Non”).
- maladie Champs texte.
- ponderation Poids echantillonnel (numerique).

Usage

```
data(tabagisme)
```

Format

Un data frame de 1000 rangees et 9 colonnes

Details

Note sur la simulation des donnees: la probabilite pour un sujet de tomber dans la categorie “malade” est basee sur une fonction arbitraire faisant intervenir l’age, l’IMC et le nombre de cigarettes fumees par jour.

A copy of this dataset is **available in English** under the name “tobacco”.

tb	<i>Convert Summarytools Objects into Tibbles</i>
----	--

Description

Make a tidy dataset out of `freq()` or `descr()` outputs

Usage

```
tb(
  x,
  order = 1,
  drop.var.col = FALSE,
  recalculate = TRUE,
  fct.to.chr = FALSE,
  ...
)
```

Arguments

<code>x</code>	a <code>freq()</code> or <code>descr()</code> output object.
<code>order</code>	Integer. Useful for grouped results produced with <code>stby</code> or <code>dplyr::group_by</code> . When set to 1 (default), the ordering is done using the grouping variables first. When set to 2, the ordering is done according to the analytical (not grouping) variable. When set to 3, the same ordering as with 2 is used, but the analytical variable is placed in first position. Depending on what function was used for grouping, the results will be different in subtle ways. See <i>Details</i> .
<code>drop.var.col</code>	Logical. For <code>descr</code> objects, drop the variable column. This is possible only when statistics are produced for a single variable; when multiple variables are present, this parameter is ignored. FALSE by default.
<code>recalculate</code>	Logical. TRUE by default . For grouped <code>freq</code> results, recalculate percentages to have total proportions sum up to 1. Defaults to TRUE.
<code>fct.to.chr</code>	Logical. When grouped objects are created with <code>dplyr::group_by</code> , the resulting tibble will have factor columns when the grouping variable itself is a factor. To convert them to character, set this to TRUE. See <i>Details</i> .
<code>...</code>	For internal use only.

Details

`stby`, which is based on `and by`, initially make the first variable vary, keeping the other(s) constant. On the other hand, `group_by` initially keeps the first grouping variable(s) constant, making the last one vary. This will impact the ordering of the rows (and as a result, the cumulative percent columns, if present).

Also, keep in mind that while `group_by` shows NA groups by default, `useNA = TRUE` must be used to achieve the same results with `stby`.

Value

A [tibble](#) which is constructed following the *tidy* principles.

Examples

```
tb(freq(iris$Species))
tb(descr(iris, stats = "common"))

data("tobacco")
tb(stby(tobacco, tobacco$gender, descr, stats = "fivenum", check.nas = FALSE),
  order=3)
tb(stby(tobacco, tobacco$gender, descr, stats = "common", useNA = TRUE))

# Compare stby() and group_by() groups' ordering
tb(with(tobacco, stby(diseased, list(gender, smoker), freq, useNA = TRUE)))

## Not run:
tobacco |> dplyr::group_by(gender, smoker) |> freq(diseased) |> tb()

## End(Not run)
```

tobacco

Tobacco Use and Health - Simulated Dataset

Description

A simulated datasets of 1,000 subjects, with the following variables:

Usage

```
data(tobacco)
```

Format

A data frame with 1000 rows and 9 variables

Details

- gender Factor with 2 levels: “F” and “M”, having roughly 500 of each.
- age Numerical.
- age.gr Factor with 4 age categories.
- BMI Body Mass Index (numerical).
- smoker Factor (“Yes” / “No”).
- cigs.per.day Number of cigarettes smoked per day (numerical).
- diseased Factor (“Yes” / “No”).

- disease Character.
- samp.wgts Sampling weights (numerical).

A note on simulation: probability for an individual to fall into category “diseased” is based on an arbitrary function involving age, BMI and number of cigarettes per day.

A copy of this dataset is also **available in French** under the name “tabagisme”.

unlabel *Clear Variable and Data Frame Label(s)*

Description

Returns the object with all labels removed. The “label” attribute as well as the “labelled” class (used by Hmisc and labelled) are cleared.

Usage

```
unlabel(x)
```

Arguments

x An R object to remove labels from.

Author(s)

Dominic Comtois, <dominic.comtois@gmail.com>.

See Also

[label](#)

use_custom_lang *Import and use a custom language*

Description

If your language is not available or if you wish to customize the outputs’ language to suit your preference, you can set up a translations file (see details) and import it with this function.

Usage

```
use_custom_lang(file)
```

Arguments

file Character. The path to the translations file.

Details

To build the translations file, copy the *language_template.csv* file located in the installed package's *includes* directory and fill out the 'custom' column using a text editor, leaving column titles unchanged. The file must also retain its *UTF-8* encoding.

view	<i>view</i>
------	-------------

Description

Visualize results in RStudio's Viewer or in Web Browser

Usage

```
view(x, method = "viewer", file = "", append = FALSE,
     report.title = NA, table.classes = NA,
     bootstrap.css = st_options("bootstrap.css"),
     custom.css = st_options("custom.css"), silent = FALSE,
     footnote = st_options("footnote"),
     max.tbl.height = Inf,
     collapse = 0,
     escape.pipe = st_options("escape.pipe"), ...)
```

Arguments

x	A <i>summarytools</i> object, created by one of the four core functions (freq , descr , ctable , or dfSummary).
method	Character. One of "pander", "viewer", "browser", or "render". Default value for the <code>print()</code> method is "pander"; for <code>view()/stview()</code> , default is "viewer" if session is running in <i>RStudio</i> , "browser" otherwise. The main use for "render" is in <i>R Markdown</i> documents.
file	Character. File name to write output to. Defaults to "".
append	Logical. Append output to existing file (specified using the <i>file</i> argument). FALSE by default.
report.title	Character. For <i>html</i> reports, this goes into the <title> tag. When left to NA (default), the first line of the heading section is used (e.g.: "Data Frame Summary").
table.classes	Character. Additional <i>html</i> classes to assign to output tables. <i>Bootstrap css</i> classes can be used. User-defined classes (see the <i>custom.css</i> argument) are also specified here. See <i>details</i> section. NA by default.
bootstrap.css	Logical. When generating an <i>html</i> document, include the " <i>includes/stylesheet/bootstrap.min.css</i> " file content inside a <style type="text/css"> tag in the document's <head>. TRUE by default. Can be set globally with st_options .

<code>custom.css</code>	Character. Path to a custom <code>.css</code> file. Classes defined in this must also appear in the <code>table.classes</code> parameter in order to be applied to the table(s). Can be set globally with <code>st_options</code> . NA by default.
<code>silent</code>	Logical. Set to TRUE to hide console messages (<i>e.g.</i> : ignored variables or NaN to NA transformations). FALSE by default.
<code>footnote</code>	Character. Text to display just after <i>html</i> output tables. The default value (“ <i>default</i> ”) produces a two-line footnote indicating the package’s name and version, the R version, and the current date. Has no effect on <i>ascii</i> or <i>markdown</i> content. Can contain standard <i>html</i> tags. Set to NA to omit. Can be set globally with <code>st_options</code> .
<code>max.tbl.height</code>	Numeric. Maximum table height <i>in pixels</i> allowed in rendered <code>dfSummary()</code> tables. When this argument is used, results will show up in a <code><div></code> with the specified height and a scroll bar. Intended to be used in <i>Rmd</i> documents with <code>method = "render"</code> . Inf by default.
<code>collapse</code>	Numeric. 0 by default. Set to 1 to make <code>freq()</code> sections collapsible (when clicking on the variable name). Future versions might provide alternate collapsing options.
<code>escape.pipe</code>	Logical. Set to TRUE when <code>style="grid"</code> and <code>file</code> argument is supplied if the intent is to generate a text file that can be converted to other formats using <i>Pandoc</i> . Can be set globally with <code>st_options</code> .
<code>...</code>	Additional arguments used to override attributes stored in the object, or to change formatting via <code>format</code> or <code>pander</code> . See <i>Details</i> .

Details

Creates *html* outputs and displays them in *RStudio*’s viewer, in a browser, or renders the *html* code in *R markdown* documents.

For objects of class “*summarytools*”, this function is simply a wrapper around `print.summarytools` with `method = "viewer"`.

Objects of class “*by*”, “*stby*”, or “*list*” are dispatched to the present function, as it can manage multiple objects, whereas `print.summarytools` can only manage one object at a time.

what.is

Obtain Extended Properties of Objects

Description

Combination of most common “macro-level” functions that describe an object.

Usage

```
what.is(x, ...)
```


Arguments

`x` Any object.
`...` Included for backward-compatibility only. Has no real use.

Details

An alternative to calling in turn `class`, `typeof`, `dim`, and so on. A call to this function will readily give all this information at once.

Value

A list with following elements:

properties A data frame with the class(es), type, mode and storage mode of the object as well as the dim, length and object.size.

attributes.lengths A named character vector giving all attributes (*c.f.* “names”, “row.names”, “class”, “dim”, and so forth) along with their length.

extensive.is A character vector of all the *identifier functions*. (starting with “is.”) that yield TRUE when used with `x` as argument.

function.type When `x` is a function, results of `f$type` are added.

Author(s)

Dominic Comtois, <dominic.comtois@gmail.com>

See Also

`class`, `typeof`, `mode`, `storage.mode`, `dim`, `length`, `is.object`, `otype`, `object.size`, `f$type`

Examples

```
what.is(1)
what.is(NaN)
what.is(iris3)
what.is(print)
what.is(what.is)
```

 zap_attr

Remove Attributes to Get a Simplified Object

Description

Get rid of summarytools-specific attributes to get a simple data structure (matrix, array, ...), which can be easily manipulated.

Usage

```
zap_attr(x, except = c("dim", "dimnames"))
```

Arguments

x	An object with attributes
except	Character. A vector of attribute names to preserve. By default, “dim” and “dimnames” are preserved.

Details

If the object contains grouped results:

- The inner objects will lose their attributes
- The “stby” class will be replaced with “by”
- The “dim” and “dimnames” attributes will be set to available relevant values, but expect slight differences between objects created with `stby()` vs `group_by()`.

Examples

```
data(tobacco)
zap_attr(descr(tobacco))
zap_attr(freq(tobacco$gender))
```

Index

- * **IO**
 - cleartmp, 3
- * **attribute**
 - dfSummary, 12
 - what.is, 40
- * **category**
 - ctable, 4
 - dfSummary, 12
 - freq, 19
- * **classes**
 - ctable, 4
 - dfSummary, 12
 - freq, 19
 - what.is, 40
- * **datasets**
 - examens, 17
 - exams, 17
 - tabagisme, 35
 - tobacco, 37
- * **methods**
 - print.summarytools, 25
- * **print**
 - print.summarytools, 25
- * **univar**
 - descr, 10
 - dfSummary, 12
 - freq, 19
- * **utilities**
 - define_keywords, 7
 - st_css, 29
 - st_options, 30
 - stby, 28
 - use_custom_lang, 38
 - what.is, 40
- by, 28, 29, 31
- class, 20, 41
- cleartmp, 3
- ctable, 4, 23, 24, 26, 27, 32, 39
- define_keywords, 7
- descr, 10, 23, 24, 26, 27, 32, 36, 39
- dfSummary, 12, 23, 24, 26, 27, 31–33, 39
- dim, 41
- examens, 17
- exams, 17
- format, 5, 11, 18, 24–26, 33, 40
- format_number, 18, 33
- freq, 19, 23, 24, 26, 27, 32, 36, 39
- ftype, 41
- group_by, 15, 24, 29, 36
- is.object, 41
- label, 16, 22, 22, 38
- label<-(label), 22
- labels, 22
- lapply, 23, 24, 31
- length, 41
- llabel(label), 22
- mode, 41
- object.size, 41
- otype, 41
- pander, 5, 11, 14, 15, 19, 20, 24–28, 31, 40
- print.list, 23
- print.stby, 24
- print.summarytools, 16, 25, 40
- round, 33
- st_css, 29
- st_options, 4, 5, 11, 13–15, 19, 20, 23–26, 30, 39, 40
- stby, 6, 24, 28, 36
- storage.mode, 41

strptime, [9](#)
stview (view), [39](#)
summarytools (summarytools-package), [2](#)
summarytools-package, [2](#)
system, [27](#)

tabagisme, [35](#)
table, [4](#), [6](#), [21](#)
tb, [36](#)
tibble, [37](#)
tobacco, [37](#)
typeof, [41](#)

unlabel, [38](#)
use_custom_lang, [7](#), [38](#)

view, [3](#), [39](#)

what.is, [40](#)

xtabs, [4](#), [6](#)

zap_attr, [41](#)