

Package ‘vtable’

April 11, 2019

Type Package

Title Variable Table

Version 0.4.0

Date 2019-04-10

Description Automatically generates HTML variable documentation including variable names, labels, classes, value labels (if applicable), value ranges, and summary statistics. See the vignette “vtable” for a package overview.

License MIT + file LICENSE

LazyData true

Encoding UTF-8

Imports utils, sjlabelled

Suggests rstudioapi, knitr, rmarkdown

RoxygenNote 6.1.1

VignetteBuilder knitr

NeedsCompilation no

Author Nick Huntington-Klein [aut, cre]

Maintainer Nick Huntington-Klein <nhuntington-klein@fullerton.edu>

Repository CRAN

Date/Publication 2019-04-11 07:22:41 UTC

R topics documented:

dfToHTML	2
vtable	3

Index	6
--------------	----------

`dfToHTML`*Data Frame to HTML Function*

Description

This function takes a data frame or matrix with column names and outputs an HTML table version of that data frame.

Usage

```
dfToHTML(data, out = NA, file = NA, col.width = NA,  
          row.names = FALSE)
```

Arguments

<code>data</code>	Data set; accepts any format with column names.
<code>out</code>	Determines where the completed table is sent. Set to "browser" to open HTML file in browser using <code>browseURL()</code> , "viewer" to open in RStudio viewer using <code>viewer()</code> , if available, or "htmlreturn" to return the HTML code. Defaults to "viewer" if RStudio is running and "browser" if it isn't.
<code>file</code>	Saves the completed variable table file to HTML with this filepath. May be combined with any value of <code>out</code> .
<code>col.width</code>	Vector of page-width percentages, on 0-100 scale, overriding default column widths in HTML table. Must have a number of elements equal to the number of columns in the resulting table.
<code>row.names</code>	Flag determining whether or not the row names should be included in the table. Defaults to FALSE.

Details

This function is designed to feed HTML versions of variable tables to `vtable()`.

Examples

```
df <- data.frame(var1 = 1:4, var2=5:8, var3=c('A', 'B', 'C', 'D'),  
                 var4=as.factor(c('A', 'B', 'C', 'C')), var5=c(TRUE, TRUE, FALSE, FALSE))  
dfToHTML(df, out="browser")
```

vtable	<i>Variable Table Function</i>
--------	--------------------------------

Description

This function will output a descriptive variable table either to the console or as an HTML file that can be viewed continuously while working with data.

Usage

```
vtable(data, out = NA, file = NA, labels = NA, class = TRUE,
       values = TRUE, missing = FALSE, index = FALSE, factor.limit = 5,
       char.values = FALSE, data.title = NA, desc = NA, col.width = NA,
       summ = NA)
```

Arguments

data	Data set; accepts any format with column names. If variable labels are set with the haven package, <code>set_label()</code> from <code>sjlabelled</code> , or <code>label()</code> from <code>Hmisc</code> , <code>vtable</code> will extract them automatically.
out	Determines where the completed table is sent. Set to "browser" to open HTML file in browser using <code>browseURL()</code> , "viewer" to open in RStudio viewer using <code>viewer()</code> , if available. Use "htmlreturn" to return the HTML code to R, or "return" to return the completed variable table to R in data frame form. Defaults to "viewer" if RStudio is running and "browser" if it isn't.
file	Saves the completed variable table file to HTML with this filepath. May be combined with any value of out.
labels	Variable labels. labels will accept three formats: (1) A vector of the same length as the number of variables in the data, in the same order as the variables in the data set, (2) A matrix or data frame with two columns and more than one row, where the first column contains variable names (in any order) and the second contains labels, or (3) A matrix or data frame where the column names (in any order) contain variable names and the first row contains labels. Setting the labels parameter will override any variable labels already in the data. Set to "omit" if the data set has embedded labels but you don't want any labels in the table.
class	Set to TRUE to include variable classes in the variable table. Defaults to TRUE.
values	Set to TRUE to include the range of values of each variable: min and max for numeric variables, list of factors for factor or ordered variables, and 'TRUE FALSE' for logicals. values will detect and use value labels set by the <code>sjlabelled</code> or haven packages. Defaults to TRUE.
missing	Set to TRUE to include whether the variable contains any NAs. Defaults to FALSE.
index	Set to TRUE to include the index number of the column with the variable name. Defaults to FALSE.

<code>factor.limit</code>	Sets maximum number of factors that will be included if <code>values = TRUE</code> . Set to 0 for no limit. Defaults to 5.
<code>char.values</code>	Set to <code>TRUE</code> to include values of character variables as though they were factors, if <code>values = TRUE</code> . Or, set to a character vector of variable names to list values of only those character variables. Defaults to <code>FALSE</code> . Has no effect if <code>values = FALSE</code> .
<code>data.title</code>	Character variable with the title of the dataset.
<code>desc</code>	Character variable offering a brief description of the dataset itself. This will by default include information on the number of observations and the number of columns. To remove this, set <code>desc='omit'</code> , or include any description and then include <code>'omit'</code> as the last four characters.
<code>col.width</code>	Vector of page-width percentages, on 0-100 scale, overriding default column widths in HTML table. Must have a number of elements equal to the number of columns in the resulting table.
<code>summ</code>	Character vector of summary statistics to include for numeric and logical variables, in the form <code>'function(x)'</code> . This option is flexible, and allows any summary statistic function that takes in a column and returns a single number. For example, <code>summ=c('mean(x)', 'mean(log(x))')</code> will provide the mean of each variable as well as the mean of the log of each variable. This also allows the special functions <code>'propNA(x)'</code> and <code>'countNA(x)'</code> , which provide the proportion and total number of missing values in the variable, respectively, which will always be displayed first and which are applied to factor and character variables as well as numeric and logical. NAs will be omitted from all calculations other than <code>propNA(x)</code> and <code>countNA(x)</code> .

Details

Outputting the variable table as a help file will make it easy to search through variable names or labels, or to refer to information about the variables easily.

This function is in a similar spirit to `promptData()`, but focuses on variable documentation rather than dataset documentation.

Examples

```
if(interactive()){
df <- data.frame(var1 = 1:4, var2=5:8, var3=c('A', 'B', 'C', 'D'),
  var4=as.factor(c('A', 'B', 'C', 'C')), var5=c(TRUE, TRUE, FALSE, FALSE))

#Demonstrating different options:
vtable(df, labels=c('Number 1', 'Number 2', 'Some Letters',
  'Some Labels', 'You Good?'))
vtable(subset(df, select=c(1,2,5)),
  labels=c('Number 1', 'Number 2', 'You Good?'), class=FALSE, values=FALSE)
vtable(subset(df, select=c('var1', 'var4')),
  labels=c('Number 1', 'Some Labels'),
  factor.limit=1, col.width=c(10, 10, 40, 35))
```

```
#Different methods of applying variable labels:
labelsmethod2 <- data.frame(var1='Number 1',var2='Number 2',
  var3='Some Letters',var4='Some Labels',var5='You Good?')
vtable(df,labels=labelsmethod2)
labelsmethod3 <- data.frame(a =c("var1","var2","var3","var4","var5"),
  b=c('Number 1','Number 2','Some Letters','Some Labels','You Good?'))
vtable(df,labels=labelsmethod3)

#Using value labels and pre-labeled data:
library(sjlabelled)
df <- set_label(df,c('Number 1','Number 2','Some Letters',
  'Some Labels','You Good?'))
df$var1 <- set_labels(df$var1,labels=c('A little','Some more',
  'Even more','A lot'))
vtable(df)

#efc is data with embedded variable and value labels from the sjlabelled package
library(sjlabelled)
data(efc)
vtable(efc)

#Displaying the values of a character vector
data(USJudgeRatings)
USJudgeRatings$Judge <- row.names(USJudgeRatings)
vtable(USJudgeRatings,char.values=c('Judge'))

#Adding summary statistics for variable mean and proportion of data that is missing.
vtable(efc,summ=c('mean(x)','propNA(x)'))

}
```

Index

`df`toHTML, 2

`vtable`, 3