Package: textab (via r-universe)

October 17, 2024

Title Create Highly-Customized 'LaTeX' Tables

Version 1.0.1

Description Generate 'LaTeX' tables directly from R. It builds 'LaTeX' tables in blocks in the spirit of 'ggplot2' using the '+' and '/' operators for concatenation in the vertical and horizontal dimensions, respectively. It exports tables in the 'LaTeX' tabular environment using '.tex' code. It can compile '.tex' code to 'PDF' automatically.

License MIT + file LICENSE

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Suggests knitr, rmarkdown

VignetteBuilder knitr

URL https://setzler.github.io/textab/

BugReports https://github.com/setzler/textab/issues

Repository https://setzler.r-universe.dev

RemoteUrl https://github.com/setzler/textab

RemoteRef HEAD

RemoteSha 9d75b452d33f853630392fb7589076f624b93963

Contents

`+.tt_` *Concatenate textab blocks vertically.*

Description

Concatenate textab blocks vertically.

Usage

```
## S3 method for class 'tt_'
upper_block + lower_block
```

Arguments

`upper_block`	The upper block of the tabular row.
`lower_block`	The lower block of the tabular row.

Value

The output is a textab block, formed by vertically concatenating the two provided textab blocks.

Examples

```
# define some textab blocks
first_block = TexRow(c(1,2))
first_block

second_block = TexRow(3)
second_block

third_block = TexRow(4)
third_block

# concatenate two blocks vertically
first_block + second_block

# concatenate three blocks vertically
first_block + second_block + third_block

# concatenate both horizontally and vertically
# note: horizontal concatenation takes precedence over vertical concatenation
first_block + second_block / third_block
```

`/.tt_` *Concatenate textab blocks horizontally (side-by-side).*

Description

Concatenate textab blocks horizontally (side-by-side).

Usage

```
## S3 method for class 'tt_'
left_block / right_block
```

Arguments

`left_block`	The left block of the tabular row.
`right_block`	The right block of the tabular row.

Value

The output is a textab block, formed by horizontally concatenating the two provided textab blocks.

Examples

```
# define some textab blocks
first_block = TexRow(c(1,2))
first_block

second_block = TexRow(3)
second_block

third_block = TexRow(4)
third_block

# concatenate two blocks horizontally
first_block / second_block

# concatenate three blocks horizontally
first_block / second_block / third_block

# concatenate both horizontally and vertically
# note: horizontal concatenation takes precedence over vertical concatenation
first_block + second_block / third_block
```

print.tt_block *Print a textab block as a LaTeX tabular.*

Description

Print a textab block as a LaTeX tabular.

Usage

```
## S3 method for class 'tt_block'
print(x, ...)
```

Arguments

x	A textab block.
...	Placeholder for print.

Value

Print prints its argument and returns it invisibly.

TexMidrule *Create one (or many) partial midrule(s).*

Description

Create one (or many) partial midrule(s).

Usage

```
TexMidrule(rule_list = NULL)
```

Arguments

rule_list	(list). A list of integer vectors. Each integer vector must contain two integers which indicate the start and end column of the partial midrule. If rule_list = NULL, it returns the full midrule across all columns. The default is rule_list = NULL.

Value

The output is a textab block.

Examples

```
# set up two textab blocks:
block1 = TexRow(c("hello","world","block"))
block2 = TexRow(c(5.081, 2.345, 6.789), dec=1)

# add a full midrule between the two blocks
block1 + TexMidrule() + block2

# add a partial midrule to the first column and spanning the second-third columns:
block1 + TexMidrule(list(c(1,1), c(2,3))) + block2
```

TexRow *This function creates a row of a LaTeX table.*

Description

This function creates a row of a LaTeX table.

Usage

```
TexRow(
  value,
  cspan = rep(1, length(value)),
  position = "c",
  surround = "%s",
  space = 0,
  dec = 3,
  percentage = FALSE,
  dollar = FALSE,
  se = FALSE,
  pvalues = NULL
)
```

Arguments

value	The value(s) to be formatted. Must be a numeric or character vector.
cspan	(integer). If greater than 1, `multicolumn{cspan}{position}{value}` will be used. For example, cspan=c(1,2,1) means that the second entry of value should span 2 columns. Default is cspan = rep(1, length(value)).
position	(character). If cspan > 1, `multicolumn{cspan}{position}{value}` will be used. For example, position=c("l","c","r") means that the second entry of value should be centered. Default is "c".
surround	(character). This will be applied to the value as sprintf(surround, value), so surround must contain the "%s" placeholder. Default is "%s".
space	(numeric). The number of points (pt) of vertical space to append to the end of the row. Default is 0.

dec	(integer). Only relevant if value is numeric. Number of decimal places. If scalar, the same decimal will be used for each entry of value. If vector, must be the same length as value. Default is 3.
percentage	(logical). Only relevant if value is numeric. If TRUE, a percentage symbol "%" will be added to the end of each entry of value. If scalar, it will be used for all entries of value. If vector, must be the same length as value.
dollar	(logical). Only relevant if value is numeric. If TRUE, a dollar sign "$" will be added to the end of each entry of value. If scalar, it will be used for all entries of value. If vector, must be the same length as value.
se	(logical). Only relevant if value is numeric. If TRUE, value will be wrapped in parentheses. If scalar, it will be used for all entries of value. If vector, must be the same length as value.
pvalues	(numeric). Only relevant if value is numeric. If not NULL, must be numeric. If less than 0.1, a star will be added to the end of each entry of value. If less than 0.05, a second star will be appended. If less than 0.01, a third star will be appended. If scalar, the same p-value will be assumed for all entries of value. If vector, must be the same length as value.

Value

The output is a textab block.

Examples

```
# basic character row:
vec = c("hello", "world")
TexRow(vec)

# character row with LaTeX formatting:
vec = c('Hello','\\textbf{World}','$\\alpha$','$\\frac{1}{2}$')
TexRow(vec)

# basic numeric row:
vec <- c(1.0, 1.01, 1.001)
TexRow(vec)
TexRow(vec, dec = 2) # round to second decimal place

# custom formatting of numbers using surround argument:
vec = c(5.081, 2.345, 6.789)
TexRow(vec, dec = 1, surround = "{\\color{red} %s}")

# use cspan argument to merge the second and third rows:
vec = c("hello", "world")
TexRow(vec, cspan = c(1,2))
TexRow(vec, cspan = c(1,2), position = "c") # center merged columns

# concatenate blocks vertically or horizontally:
block1 = TexRow(c("hello","world","block"))
block2 = TexRow(c(5.081, 2.345, 6.789), dec=1)
block1 / block2 # horizontal
```

```
block1 + block2 # vertical

# add 3pt of vertical space between two rows using the space argument:
TexRow(c("hello", "world"), space=3) + TexRow(c('$\\alpha$','$\\frac{1}{2}$'))
```

TexSave *Compile a tabular object to a pdf file*

Description

Compile a tabular object to a pdf file

Usage

```
TexSave(
  tab,
  filename,
  positions,
  pretty_rules = TRUE,
  output_path = getwd(),
  stand_alone = FALSE,
  compile_tex = FALSE
)
```

Arguments

tab	textab block, created by TexRow().
filename	(character). The file will be saved as filename.tex.
positions	(character). Vector of positions, e.g., "c("l","c","r")" means that the first column will be left-aligned, second column will be center-aligned, and third column will be right-aligned.
pretty_rules	(logical). If TRUE, extra formatting rules will be added to the bottom and top of the tabular.
output_path	(character). This is the directory path where the file should be saved. Default is the current directory.
stand_alone	(logical). If TRUE, the tabular will be exported in a .tex file that can be compiled to PDF directly (rather than included in a separate .tex file). Default is FALSE.
compile_tex	(logical). If TRUE and stand_alone is TRUE, pdflatex is used to compile the TeX table into a PDF. This is only allowed if stand_alone=TRUE. Default is FALSE.

Value

A list containing the path to the .tex file and the name of the .tex file.

Examples

```
# consider the following example textab object:
tt = TexRow(c("hello", "world")) + TexRow(c(1,2))

# define the positions for each column:
pos = c("l","c")

# choose an output path:
op = tempdir()

# Save a simple .tex document containing this table:
TexSave(tab = tt, positions = pos, filename = "example1", output_path = op)

# Save the .tex document as stand-alone, which includes LaTeX headers and packages:
TexSave(tab = tt, positions = pos, filename = "example2", output_path = op, stand_alone = TRUE)
```

Index