Package 'xtable'

March 6, 2014

Version 1.7-3

Date 2014/01/08

Title Export tables to LaTeX or HTML

Author David B. Dahl <dahl@stat.tamu.edu>

Maintainer Charles Roosen <statsci@gmail.com>

Description Coerce data to LaTeX and HTML tables

URL http://xtable.r-forge.r-project.org/

Depends R (>= 2.10.0)

License GPL (>= 2)

Repository CRAN

Repository/R-Forge/Project xtable

Repository/R-Forge/Revision 44

Repository/R-Forge/DateTimeStamp 2014-01-31 13:52:59

Date/Publication 2014-03-06 21:30:47

NeedsCompilation no

R topics documented:

---

print.xtable   *Print Export Tables*

Description

Function returning and displaying or writing to disk the LaTeX or HTML code associated with the supplied object of class `xtable`.

Usage

```
## S3 method for class 'xtable'
print(x,
  type = getOption("xtable.type", "latex"),
  file = getOption("xtable.file", ""),
  append = getOption("xtable.append", FALSE),
  floating = getOption("xtable.floating", TRUE),
  floating.environment = getOption("xtable.floating.environment", "table"),
  table.placement = getOption("xtable.table.placement", "ht"),
  caption.placement = getOption("xtable.caption.placement", "bottom"),
  caption.width = getOption("xtable.caption.width", NULL),
  latex.environments = getOption("xtable.latex.environments", c("center")),
  tabular.environment = getOption("xtable.tabular.environment", "tabular"),
  size = getOption("xtable.size", NULL),
  hline.after = getOption("xtable.hline.after", c(-1,0,nrow(x))),
  NA.string = getOption("xtable.NA.string", ""),
  include.rownames = getOption("xtable.include.rownames", TRUE),
  include.colnames = getOption("xtable.include.colnames", TRUE),
  only.contents = getOption("xtable.only.contents", FALSE),
  add.to.row = getOption("xtable.add.to.row", NULL),
  sanitize.text.function = getOption("xtable.sanitize.text.function", NULL),
  sanitize.rownames.function = getOption("xtable.sanitize.rownames.function",
                                         sanitize.text.function),
  sanitize.colnames.function = getOption("xtable.sanitize.colnames.function",
                                         sanitize.text.function),
  math.style.negative = getOption("xtable.math.style.negative", FALSE),
  html.table.attributes = getOption("xtable.html.table.attributes",
                                    "border=1"),
  print.results = getOption("xtable.print.results", TRUE),
  format.args = getOption("xtable.format.args", NULL),
  rotate.rownames = getOption("xtable.rotate.rownames", FALSE),
  rotate.colnames = getOption("xtable.rotate.colnames", FALSE),
  booktabs = getOption("xtable.booktabs", FALSE),
  scalebox = getOption("xtable.scalebox", NULL),
  width = getOption("xtable.width", NULL),
  comment = getOption("xtable.comment", TRUE),
  timestamp = getOption("xtable.timestamp", date()),
  ...)
```

Arguments

x	An object of class `"xtable"`.
type	Type of table to produce. Possible values for `type` are `"latex"` or `"html"`. Default value is `"latex"`.
file	Name of file where the resulting code should be saved. If `file=""`, output is displayed on screen. Note that the function also (invisibly) returns a character vector of the results (which can be helpful for post-processing). Default value is `""`.
append	If `TRUE` and `file!=""`, code will be appended to `file` instead of overwriting `file`. Default value is `FALSE`.
floating	If `TRUE` and `type="latex"`, the resulting table will be a floating table (using, for example, \begin{table} and \end{table}). See `floating.environment` below. Default value is `TRUE`.
floating.environment	
	If `floating=TRUE` and `type="latex"`, the resulting table uses the specified floating environment. Possible values include `"table"`, `"table*"`, and other floating environments defined in LaTeX packages. Default value is `"table"`.
table.placement	
	If `floating=TRUE` and `type="latex"`, the floating table will have placement given by `table.placement` where `table.placement` must be `NULL` or contain only elements of {"h","t","b","p","!","H"}. Default value is `"ht"`.
caption.placement	
	The caption will be have placed at the bottom of the table if `caption.placement` is `"bottom"` and at the top of the table if it equals `"top"`. Default value is `"bottom"`.
caption.width	The caption will be placed in a `"parbox"` of the specified width if `caption.width` is not `NULL` and `type="latex"`. Default value is `NULL`.
latex.environments	
	If `floating=TRUE` and `type="latex"`, the specified latex environments (provided as a character vector) will enclose the tabular environment. Default value is `"center"`.
tabular.environment	
	When `type="latex"`, the tabular environment that will be used. Defaults to `"tabular"`. When working with tables that extend more than one page, using `tabular.environment="longtable"` and the LaTeX package `"longtable"` (see Fairbairns, 2005) allows one to typeset them uniformly. Note that `"floating"` should be set to `"FALSE"` when using the `"longtable"` environment.
size	An arbitrary character vector intended to be used to set the font size in a LaTeX table. The supplied value (if not `NULL`) is inserted just before the tabular environment starts. Default value is `NULL`.
hline.after	When `type="latex"`, a vector of numbers between -1 and `"nrow(x)"`, inclusive, indicating the rows after which a horizontal line should appear. If `NULL` is used no lines are produced. Default value is `c(-1,0,nrow(x))` which means draw a line before and after the columns names and at the end of the table. Repeated values are allowed.

NA.string	String to be used for missing values in table entries. Default value is `""`.
include.rownames	
	logical. If `TRUE` the rows names is printed. Default value is `TRUE`.
include.colnames	
	logical. If `TRUE` the columns names is printed. Default value is `TRUE`.
only.contents	logical. If `TRUE` only the rows of the table is printed. Default value is `FALSE`.
add.to.row	a list of two components. The first component (which should be called 'pos') is a list contains the position of rows on which extra commands should be added at the end, The second component (which should be called 'command') is a character vector of the same length of the first component which contains the command that should be added at the end of the specified rows. Default value is `NULL`, i.e. do not add commands.
sanitize.text.function	
	All non-numeric enteries (except row and column names) are sanitised in an attempt to remove characters which have special meaning for the output format. If `sanitize.text.function` is not NULL (the default), it should be a function taking a character vector and returning one, and will be used for the sanitization instead of the default internal function.
sanitize.rownames.function	
	Like the `sanitize.text.function`, but applicable to row names. The default uses the `sanitize.text.function`.
sanitize.colnames.function	
	Like the `sanitize.text.function`, but applicable to column names. The default uses the `sanitize.text.function`.
math.style.negative	
	In a LaTeX table, if TRUE, then use $-$ for the negative sign (as was the behavior prior to version 1.5-3). Default value is `FALSE`.
html.table.attributes	
	In an HTML table, attributes associated with the `<TABLE>`tag. Default value is `border=1`.
print.results	If `TRUE`, the generated table is printed to standard output. Set this to `FALSE` if you will just be using the character vector that is returned invisibly.
format.args	List of arguments for the `formatC` function. For example, standard German number separators can be specified as `format.args=list(big.mark = "'", decimal.mark = ","))`. `digits` and `format` arguments should not be included in this list. See details.
rotate.rownames	
	If `TRUE`, the row names are displayed vertically in LaTeX.
rotate.colnames	
	If `TRUE`, the column names are displayed vertically in LaTeX.
booktabs	If `TRUE`, the `toprule`, `midrule` and `bottomrule` tags from the LaTex "booktabs" package are used rather than `hline` for the horizontal line tags.
scalebox	If not `NULL`, a `scalebox` clause will be added around the tabular environment with the specified value used as the scaling factor.

width	If not NULL, the specified value is included in parenthesis between the tabular environment begin tag and the alignment specification. This allows specification of the table width when using tabular environments such as tabular* and tabularx. Note that table width specification is not supported with the tabular or longtable environments.
comment	If TRUE, the version and timestamp comment is included. Default value is TRUE.
timestamp	Timestamp to include in LaTeX comment. Set this to NULL to exclude the timestamp. Default value is date().
...	Additional arguments. (Currently ignored.)

Details

This function displays or writes to disk the code to produce a table associated with an object x of class "xtable". The resulting code is either a LaTeX or HTML table, depending on the value of type. The function also (invisibly) returns a character vector of the results (which can be helpful for post-processing).

Since version 1.4 the non default behavior of hline.after is changed. To obtain the same results as the previous versions add to the hline.after vector the vector c(-1, 0, nrow(x)) where nrow(x) is the numbers of rows of the object.

From version 1.4-3, all non-numeric columns are sanitized, and all LaTeX special characters are sanitised for LaTeX output. See Section 3 of the xtableGallery vignette for an example of customising the sanitization. From version 1.4-4, the sanitization also applies to column names. To remove any text sanitization, specify sanitize.text.function=function(x){x}.

From version 1.6-1 the default values for the arguments other than x are obtainined using getOption(). Hence the user can set the values once with options() rather than setting them in every call to print.xtable().

The argument format.args is used to supply arguments to the formatC function, but will throw an error if values for digits or format are included in the list of arguments. The recommended approach is to specify digits supply the argument digits to xtable, and to specify format supply the argument display to xtable. See the examples.

Author(s)

David Dahl <dahl@stat.tamu.edu> with contributions and suggestions from many others (see source code).

References

Fairbairns, Robin (2005) *Tables longer than a single page* The UK List of TeX Frequently Asked Questions on the Web. http://www.tex.ac.uk/cgi-bin/texfaq2html?label=longtab

See Also

xtable, caption, label, align, digits, display, formatC

Examples

```
df <- data.frame(A = c(1.00123, 33.1, 6),
                 B = c(111111, 3333333, 3123.233))
## The following code gives the error
## formal argument "digits" matched by multiple actual arguments
## print(xtable(df, display = c("s","e","e")),
##       format.args = list(digits = 3, big.mark = " ", decimal.mark = ","))
## specify digits as argument to xtable instead
print(xtable(df, display = c("s","f","f"), digits = 4),
      format.args = list(big.mark = " ", decimal.mark = ","))
## The following code gives the error
## formal argument "format" matched by multiple actual arguments
## print(xtable(df, digits = 4),
##       format.args = list(format = c("s","e","e"),
##                          big.mark = " ", decimal.mark = ","))
## specify format using display argument in xtable
print(xtable(df, display = c("s","e","e"), digits = 4),
      format.args = list(big.mark = " ", decimal.mark = ","))
```

---

string   *String handling functions*

Description

Private functions for conveniently working with strings.

Usage

```
  string(text,file="",append=FALSE)
  ## S3 method for class 'string'
print(x,...)
  ## S3 method for class 'string'
x + y
  as.string(x,file="",append=FALSE)
  is.string(x)
```

Arguments

text	A character object.
file	Name of the file that should receive the printed string.
append	Should the printed string be appended to the file?
x	A string object.
y	A string object.
...	Additional arguments. (Currently ignored.)

Details

These functions are private functions used by print.xtable. They are not intended to be used elsewhere.

Author(s)

David Dahl <dahl@stat.tamu.edu> with contributions and suggestions from many others (see source code).

See Also

print.xtable

---

table.attributes   *Retrieve and Set Options for Export Tables*

Description

Functions retrieving or setting table attributes for the supplied object of class "xtable".

Usage

```
caption(x,...)
caption(x) <- value
label(x,...)
label(x) <- value
align(x,...)
align(x) <- value
digits(x,...)
digits(x) <- value
display(x,...)
display(x) <- value
```

Arguments

x	An "xtable" object.
value	The value of the corresponding attribute.
...	Additional arguments. (Currently ignored.)

Details

These functions retrieve or set table attributes of the object x of class "xtable". See xtable for a description of the options.

Author(s)

David Dahl <dahl@stat.tamu.edu> with contributions and suggestions from many others (see source code).

See Also

xtable, print.xtable, formatC

---

tli   *Math scores from Texas Assessment of Academic Skills (TAAS)*

Description

This data set contains math scores and demographic data of 100 randomly selected students participating in the Texas Assessment of Academic Skills (TAAS).

Usage

```
data(tli)
```

Format

A data.frame containing 100 observations with the following columns:

grade Year in school of student

sex Gender of student

disadvg Is the student economically disadvantaged?

ethnicty Race of student

tlimth Math score of student

Source

Texas Education Agency, http://www.tea.state.tx.us

---

toLatex.xtable   *Convert Table to Latex*

Description

Function creating a LaTeX representation of an object of class xtable.

Usage

```
## S3 method for class 'xtable'
toLatex(object, ...)
```

Arguments

object   An object of class "xtable".

...   Other arguments to print.xtable.

Details

This function creates a LaTeX representation of an object of class `"xtable"`. This is a method for the generic `"toLatex"` in the core R package `"utils"`.

Author(s)

Charles Roosen <roosen@gmail.com> with contributions and suggestions from many others (see source code).

See Also

print.xtable

---

xtable   *Create Export Tables*

Description

Function converting an R object to an `xtable` object, which can then be printed as a LaTeX or HTML table.

Usage

```
xtable(x, caption = NULL, label = NULL, align = NULL, digits = NULL,
       display = NULL, ...)
```

Arguments

x	An R object of class found among `methods(xtable)`. See below on how to write additional method functions for `xtable`.
caption	Character vector of length 1 or 2 containing the table's caption or title. If length 2, the second item is the "short caption" used when LaTeX generates a "List of Tables". Set to NULL to suppress the caption. Default value is NULL.
label	Character vector of length 1 containing the LaTeX label or HTML anchor. Set to NULL to suppress the label. Default value is NULL.
align	Character vector of length equal to the number of columns of the resulting table indicating the alignment of the corresponding columns. Also, `"|"` may be used to produce vertical lines between columns in LaTeX tables, but these are effectively ignored when considering the required length of the supplied vector. If a character vector of length one is supplied, it is split as `strsplit(align, "")[[1]]` before processing. Since the row names are printed in the first column, the length of `align` is one greater than `ncol(x)` if x is a `data.frame`. Use `"l"`, `"r"`, and `"c"` to denote left, right, and center alignment, respectively. Use `"p\{3cm\}"` etc for a LaTeX column of the specified width. For HTML output the `"p"` alignment is interpreted as `"l"`, ignoring the width request. Default depends on the class of x.

digits   Numeric vector of length equal to one (in which case it will be replicated as
                necessary) or to the number of columns of the resulting table **or** matrix of the
                same size as the resulting table indicating the number of digits to display in the
                corresponding columns. Since the row names are printed in the first column, the
                length of the vector digits or the number of columns of the matrix digits is
                one greater than ncol(x) if x is a data.frame. Default depends of class of x.
                If values of digits are negative, the corresponding values of x are displayed in
                scientific format with abs(digits) digits.

display   Character vector of length equal to the number of columns of the resulting table
                indicating the format for the corresponding columns. Since the row names are
                printed in the first column, the length of display is one greater than ncol(x) if
                x is a data.frame. These values are passed to the formatC function. Use "d"
                (for integers), "f", "e", "E", "g", "G", "fg" (for reals), or "s" (for strings). "f"
                gives numbers in the usual xxx.xxx format; "e" and "E" give n.ddde+nn or
                n.dddE+nn (scientific format); "g" and "G" put x[i] into scientific format only
                if it saves space to do so. "fg" uses fixed format as "f", but digits as number
                of *significant* digits. Note that this can lead to quite long result strings. Default
                depends on the class of x.

...   Additional arguments. (Currently ignored.)

Details

This function extracts tabular information from x and returns an object of class "xtable". The
nature of the table generated depends on the class of x. For example, aov objects produce ANOVA
tables while data.frame objects produce a table of the entire data.frame. One can optionally pro-
vide a caption (called a title in HTML) or label (called an anchor in HTML), as well as formatting
specifications. Default values for align, digits, and display are class dependent.

The available method functions for xtable are given by methods(xtable). Users can extend
the list of available classes by writing methods for the generic function xtable. These methods
functions should have x as their first argument with additional arguments to specify caption, label,
align, digits, and display. Optionally, other arguments may be present to specify how the object
x should be manipulated. All method functions should return an object whose class if given by
c("xtable","data.frame"). The resulting object can have attributes caption and label, but
must have attributes align, digits, and display. It is strongly recommened that you set these
attributes through the provided replacement functions as they perform validity checks.

Value

An object of class "xtable" which inherits the data.frame class and contains several additional
attributes specifying the table formatting options.

Author(s)

David Dahl <dahl@stat.tamu.edu> with contributions and suggestions from many others (see
source code).

See Also

[print.xtable](), [caption](), [label](), [align](), [digits](), [display](), [formatC](), [methods]()

Examples

```
## Load example dataset
data(tli)

## Demonstrate data.frame
tli.table <- xtable(tli[1:20, ])
digits(tli.table)[c(2, 6)] <- 0
print(tli.table)
print(tli.table, type = "html")

## Demonstrate data.frame with different digits in cells
tli.table <- xtable(tli[1:20, ])
digits(tli.table) <- matrix( 0:4, nrow = 20, ncol = ncol(tli)+1 )
print(tli.table)
print(tli.table, type = "html")

## Demonstrate matrix
design.matrix <- model.matrix(~ sex*grade, data = tli[1:20, ])
design.table <- xtable(design.matrix)
print(design.table)
print(design.table, type = "html")

## Demonstrate aov
fm1 <- aov(tlimth ~ sex + ethnicty + grade + disadvg, data = tli)
fm1.table <- xtable(fm1)
print(fm1.table)
print(fm1.table, type = "html")

## Demonstrate lm
fm2 <- lm(tlimth ~ sex*ethnicty, data = tli)
fm2.table <- xtable(fm2)
print(fm2.table)
print(fm2.table, type = "html")
print(xtable(anova(fm2)))
print(xtable(anova(fm2)), type = "html")
fm2b <- lm(tlimth ~ ethnicty, data = tli)
print(xtable(anova(fm2b, fm2)))
print(xtable(anova(fm2b, fm2)), type = "html")

## Demonstrate glm
fm3 <- glm(disadvg ~ ethnicty*grade, data = tli, family = binomial())
fm3.table <- xtable(fm3)
print(fm3.table)
print(fm3.table, type = "html")
print(xtable(anova(fm3)))
print(xtable(anova(fm3)), type = "html")

## Demonstrate aov
## Taken from help(aov) in R 1.1.1
## From Venables and Ripley (1997) p.210.
N <- c(0,1,0,1,1,1,0,0,0,1,1,0,1,1,0,0,1,0,1,0,1,1,0,0)
```

```
P <- c(1,1,0,0,0,1,0,1,1,1,0,0,0,1,0,1,1,0,0,1,0,1,1,0)
K <- c(1,0,0,1,0,1,1,0,0,1,0,1,0,1,1,0,0,0,1,1,1,0,1,0)
yield <- c(49.5,62.8,46.8,57.0,59.8,58.5,55.5,56.0,62.8,55.8,69.5,55.0,
           62.0,48.8,45.5,44.2,52.0,51.5,49.8,48.8,57.2,59.0,53.2,56.0)
npk <- data.frame(block = gl(6,4), N = factor(N), P = factor(P),
                  K = factor(K), yield = yield)
npk.aov <- aov(yield ~ block + N*P*K, npk)
op <- options(contrasts = c("contr.helmert", "contr.treatment"))
npk.aovE <- aov(yield ~  N*P*K + Error(block), npk)
options(op)

summary(npk.aov)
print(xtable(npk.aov))
print(xtable(anova(npk.aov)))
print(xtable(summary(npk.aov)))

summary(npk.aovE)
print(xtable(npk.aovE), type = "html")
print(xtable(summary(npk.aovE)), type = "html")

## Demonstrate lm
## Taken from help(lm) in R 1.1.1
## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
## Page 9: Plant Weight Data.
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
group <- gl(2,10,20, labels = c("Ctl","Trt"))
weight <- c(ctl, trt)
lm.D9 <- lm(weight ~ group)
print(xtable(lm.D9))
print(xtable(anova(lm.D9)))

## Demonstrate glm
## Taken from help(glm) in R 1.1.1
## Annette Dobson (1990) "An Introduction to Generalized Linear Models".
## Page 93: Randomized Controlled Trial :
counts <- c(18,17,15,20,10,20,25,13,12)
outcome <- gl(3,1,9)
treatment <- gl(3,3)
d.AD <- data.frame(treatment, outcome, counts)
glm.D93 <- glm(counts ~ outcome + treatment, family = poisson())
print(xtable(glm.D93, align = "r|llrc"))
print(xtable(anova(glm.D93)), hline.after = c(1), size = "small")

## Demonstration of additional formatC() arguments.
print(fm1.table, format.args = list(big.mark = "'", decimal.mark = ","))

## Demonstration of "short caption" support.
fm1sc <- aov(tlimth ~ sex + ethnicty + grade, data = tli)
fm1sc.table <- xtable(fm1sc,
  caption = c("ANOVA Model with Predictors Sex, Ethnicity, and Grade",
    "ANOVA: Sex, Ethnicity, Grade"))
print(fm1sc.table)
```

```
## Demonstration of longtable support.
## Remember to insert \usepackage{longtable} on your LaTeX preamble
x <- matrix(rnorm(1000), ncol = 10)
x.big <- xtable(x, label = 'tabbig',
                caption = 'Example of longtable spanning several pages')
print(x.big, tabular.environment = 'longtable', floating = FALSE)
x <- x[1:30, ]
x.small <- xtable(x, label = 'tabsmall', caption = 'regular table env')
print(x.small)  # default, no longtable

## Demonstration of sidewaystable support.
## Remember to insert \usepackage{rotating} on your LaTeX preamble
print(x.small, floating.environment = 'sidewaystable')

if(require(stats, quietly = TRUE)) {
  ## Demonstrate prcomp
  ## Taken from help(prcomp) in mva package of R 1.1.1
  data(USArrests)
  pr1 <- prcomp(USArrests)
  print(xtable(pr1))
  print(xtable(summary(pr1)))

#  ## Demonstrate princomp
#  ## Taken from help(princomp) in mva package of R 1.1.1
#  pr2 <- princomp(USArrests)
#  print(xtable(pr2))
}

## Demonstrate include.rownames, include.colnames,
## only.contents and add.to.row arguments
set.seed(2345)
res <- matrix(sample(0:9, size = 6*9, replace = TRUE), ncol = 6, nrow = 9)
xres <- xtable(res)
digits(xres) <- rep(0, 7)
addtorow <- list()
addtorow$pos <- list()
addtorow$pos[[1]] <- c(0, 2)
addtorow$pos[[2]] <- 4
addtorow$command <- c('\vspace{2mm} \n', '\vspace{10mm} \n')
print(xres, add.to.row = addtorow, include.rownames = FALSE,
      include.colnames = TRUE, only.contents = TRUE,
      hline.after = c(0, 0, 9, 9))

## Demonstrate include.rownames, include.colnames,
## only.contents and add.to.row arguments in Rweave files

## Not run:
 \begin{small}
 \setlongtables
 \begin{longtable}{
 <<results = tex, fig = FALSE>>=
 cat(paste(c('c', rep('cc', 34/2-1), 'c'), collapse = '@{\hspace{2pt}}'))
```

```
 @
 }
 \hline
 \endhead
 \hline
 \endfoot
 <<results = tex, fig = FALSE>>=
 library(xtable)
 set.seed(2345)
 res <- matrix(sample(0:9, size = 34*90, replace = TRUE), ncol = 34, nrow = 90)
 xres <- xtable(res)
 digits(xres) <- rep(0, 35)
 addtorow <- list()
 addtorow$pos <- list()
 addtorow$pos[[1]] <- c(seq(4, 40, 5), seq(49, 85, 5))
 addtorow$pos[[2]] <- 45
 addtorow$command <- c('\vspace{2mm} \n', '\newpage \n')
 print(xres, add.to.row = addtorow, include.rownames = FALSE,
       include.colnames = FALSE, only.contents = TRUE, hline.after = NULL)
 @
 \end{longtable}
 \end{small}

## End(Not run)

## Demonstrate sanitization
mat <- round(matrix(c(0.9, 0.89, 200, 0.045, 2.0), c(1, 5)), 4)
rownames(mat) <- "$y_{t-1}$"
colnames(mat) <- c("$R^2$", "$\\bar{R}^2$", "F-stat", "S.E.E", "DW")
print(xtable(mat), type = "latex", sanitize.text.function = function(x){x})

## Demonstrate booktabs
print(tli.table)
print(tli.table, hline.after = c(-1,0))
print(tli.table, hline.after = NULL)
print(tli.table,
      add.to.row = list(pos = list(2), command = c("\vspace{2mm} \n")))

print(tli.table, booktabs = TRUE)
print(tli.table, booktabs = TRUE, hline.after = c(-1,0))
print(tli.table, booktabs = TRUE, hline.after = NULL)
print(tli.table, booktabs = TRUE,
  add.to.row = list(pos = list(2), command = c("\vspace{2mm} \n")))
print(tli.table, booktabs = TRUE, add.to.row = list(pos = list(2),
  command = c("youhou\n")), tabular.environment = "longtable")
```

Index