

Some examples of conditional typesetting using the latex() function.

David Whiting
david.whiting@ncl.ac.uk

December 19, 2005

Contents

1	Introduction	2
2	Additions to help page	3
3	Vector	4
3.1	Everything set to NULL	4
3.2	Everything set to something	5
3.3	Named vector—everything set to something	6
3.4	Named vector—row groups and everything set to null	6
3.5	Named vector—row groups and everything set to something	6
4	Matrix	7
4.1	Matrix without names, everything set to NULL	7
4.2	Matrix—named, everything set to something	9
5	Data frame	9
6	Several column groups	10
7	Various LaTeX font attributes	11
8	Numeric dollar	11
9	Column names, column groups and title	14
10	Shading individual cells	14

List of Tables

1	Vector. All attributes set to NULL	5
2	Vector. All attributes set to something (but most ignored because they don't apply).	5
3	Named vector. All attributes set to something.	6
4	Named vector. Row groups and all attributes set to NULL. . .	7
5	Named vector. Row groups and all attributes set to something.	8
6	A matrix, no names, formats set to NULL	8
7	A named matrix with formats.	9
8	Conditional typesetting of a data frame.	10
9	More than two column groups.	11
10	A variety of font attributes, and some other LaTeX commands. This is ugly, but shows how easy it is to format table font attributes.	12
11	A variety of font attributes, this time without numeric.dollar = FALSE.	13
12	Colouring individual cells.	14

1 Introduction

The main reason for this was to make it easier to typeset different parts of tables produced by the `latex()` function, for example selecting **bold** or *italic*. Colours stand out better so I have used them in this document to highlight where I have specified typesetting commands. I have tested this on simple and named vectors, matrices and data frames. I have tested setting all options to `NULL` and also specifying all options. I have also varied a few of the many other options that `latex()` provides to try to catch combinations that might be problematic.

In the preamble of the LaTeX file I created the following commands:

```
\providecommand{\redscshape}{\color{red}\scshape}
```

```
\providecommand{\avantgarde}{%  
\fontfamily{pag}\fontsize{14}{16}\selectfont}
```

```
\providecommand{\bookman}{%  
\fontfamily{pbk}\fontsize{14}{16}\selectfont}
```

```
\providecommand{\palatino}{%
```

```

\fontfamily{ppl}\fontsize{14}{16}\selectfont\color{blue}}

\providecommand{\newcentury}{%
\fontfamily{pnc}\fontsize{14}{16}\selectfont\color{red}}

%% The LaTeX package colortbl provides a way to color rows and cells
%% of tables. I'll define a colour to use for shading row.
\providecommand{\shadeRow}{\rowcolor[rgb]{0, 0.99, 0}}

```

2 Additions to help page

`cgroupTexCmd`: A character string specifying a LaTeX command to be used to format column group labels. The default, “bfseries”, sets the current font to ‘bold’. It is possible to supply a vector of strings so that each column group label is formatted differently. Please note that the first item of the vector is used to format the title (even if a title is not used), and that there is an empty column between each column group. Currently the user needs to handle these issues. Multiple effects can be achieved by creating custom LaTeX commands; for example, `\providecommand{\redscshape}{\color{red}\scshape}` creates a LaTeX command called `redscshape` that formats the text in red small-caps.

`rgroupTexCmd`: A character string specifying a LaTeX command to be used to format row group labels. The default, “bfseries”, sets the current font to ‘bold’. A vector of strings can be supplied to format each row group label differently. Normal recycling applies if the vector is shorter than `n.rgroups`. See also `cgroupTexCmd` above regarding multiple effects.

`rownamesTexCmd`: A vector of character strings specifying LaTeX commands to be used to format rownames. The default, NULL, applies no command. See also `cgroupTexCmd` above regarding multiple effects.

`colnamesTexCmd`: A character strings specifying a LaTeX command to be used to format column labels. The default, NULL, applies no command. It is possible to supply a vector of strings to format each column label differently. If column groups are not used, the first item in the vector will be used to format the title. Please note that if column groups are used the first item of `cgroupTexCmd` and not `colnamesTexCmd` is used to format the title, and that there are empty columns between each group. The user needs to allow for these issues when supplying a vector. See also `cgroupTexCmd` above regarding multiple effects.

`cellTexCmds`: A matrix of character strings of LaTeX commands to be

used to format each element, or cell, of the object. The matrix must have the same `NROW()`¹ and `NCOL()` as the object. The default, `NULL`, applies no formats. Empty strings also apply no formats, and one way to start might be to create a matrix of empty strings with

```
matrix(rep("", NROW(x) * NCOL(x)), nrow=NROW(x))
```

and then selectively change appropriate elements of the matrix. Note that you might need to set `numeric.dollar=FALSE` (to disable math mode) for some effects to work. See also `cgroupTexCmd` above regarding multiple effects.

Hmisc library by Frank E Harrell Jr

```
Type library(help='Hmisc'), ?Overview, or ?Hmisc.Overview')
to see overall documentation.
```

NOTE:Hmisc no longer redefines `[.factor` to drop unused levels when subsetting. To get the old behavior of Hmisc type `dropUnusedLevels()`.

```
Attaching package: 'Hmisc'
```

The following object(s) are masked from package:stats :

```
ecdf
```

3 Vector

```
> x <- letters[1:10]
```

3.1 Everything set to NULL

See Table 1.

```
> latex(x, file = "", numeric.dollar = FALSE, label = "tab:example-v-null",
+       caption = "Vector. All attributes set to NULL")
```

¹Note that this is `NROW()` and not `nrow()`.

Table 1: Vector. All attributes set to NULL

a
b
c
d
e
f
g
h
i
j

Table 2: Vector. All attributes set to something (but most ignored because they don't apply).

a
b
c
d
e
f
g
h
i
j

3.2 Everything set to something

See Table 2.

```
> latex(x, file = "", colnamesTexCmd = "scshape", cgroupTexCmd = "color{blue}",  
+   rgroupTexCmd = "itshape", cellTexCmds = matrix(rep("bfseries",  
+   10), ncol = 1), rownamesTexCmd = "bfseries", numeric.dollar = FALSE,  
+   label = "tab:example-v", caption = "Vector. All attributes set to something (but most ignored because they don't")
```

Table 3: Named vector. All attributes set to something.

X	
a	1
b	2
c	3
d	4
e	5
f	6
g	7
h	8
i	9
j	10

3.3 Named vector—everything set to something

See Table 3.

```
> x <- 1:10
> names(x) <- letters[x]
> latex(x, file = "", colnamesTexCmd = "scshape", cgroupTexCmd = "color{red}",
+   rgroupTexCmd = "itshape", cellTexCmds = matrix(rep("bfseries",
+   10), ncol = 1), rownamesTexCmd = "color{blue}", numeric.dollar = FALSE,
+   label = "tab:example-v-named", caption = "Named vector. All attributes set to something.")
```

3.4 Named vector—row groups and everything set to null

See Table 4.

```
> x <- 1:10
> names(x) <- letters[x]
> latex(x, file = "", colnamesTexCmd = NULL, cgroupTexCmd = NULL,
+   rgroupTexCmd = NULL, cellTexCmds = NULL, rownamesTexCmd = NULL,
+   rgroup = c("First row group", "Second row group"), n.ingroup = c(5,
+   length(x) - 5), numeric.dollar = FALSE, label = "tab:example-v-named-rgroups-null",
+   caption = "Named vector. Row groups and all attributes set to NULL.")
```

3.5 Named vector—row groups and everything set to something

See Table 5.

Table 4: Named vector. Row groups and all attributes set to NULL.

x	
First row group	
a	1
b	2
c	3
d	4
e	5
Second row group	
f	6
g	7
h	8
i	9
j	10

```

> x <- 1:10
> names(x) <- letters[x]
> row.format <- rep("", 10)
> row.format[4] <- "color{red}"
> cell.format <- matrix(rep("", 10), ncol = 1)
> cell.format[3] <- "color{red}"
> latex(x, file = "", colnamesTexCmd = "color{green}", cgroupTexCmd = "scshape",
+   rgroupTexCmd = "color{blue}", cellTexCmds = cell.format,
+   rownamesTexCmd = row.format, rgroup = c("First row group",
+     "Second row group"), n.rgroup = c(5, length(x) - 5),
+   numeric.dollar = FALSE, label = "tab:example-v-rg-everthing",
+   caption = "Named vector. Row groups and all attributes set to something.")

```

4 Matrix

4.1 Matrix without names, everything set to NULL

See Table 6.

```

> x <- matrix(1:30, ncol = 5)
> latex(x, title = "", file = "", colnamesTexCmd = NULL, rownamesTexCmd = NULL,
+   cgroupTexCmd = NULL, rgroupTexCmd = NULL, cellTexCmds = NULL,
+   numeric.dollar = FALSE, ctable = TRUE, label = "tab:matrix-1",
+   caption = "A matrix, no names, formats set to NULL")

```

Table 5: Named vector. Row groups and all attributes set to something.

x	
First row group	
a	1
b	2
c	3
d	4
e	5
Second row group	
f	6
g	7
h	8
i	9
j	10

Table 6: A matrix, no names, formats set to NULL

1	7	13	19	25
2	8	14	20	26
3	9	15	21	27
4	10	16	22	28
5	11	17	23	29
6	12	18	24	30

Table 7: A named matrix with formats.

	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>
First group					
j	1	7	13	19	25
k	2	8	14	20	26
l	3	9	15	21	27
m	4	10	16	22	28
Second group					
n	5	11	17	23	29
o	6	12	18	24	30

4.2 Matrix—named, everything set to something

See Table 7.

```
> x <- matrix(1:30, ncol = 5)
> colnames(x) <- c("A", "B", "C", "D", "E")
> rownames(x) <- c(letters[10:15])
> cell.format <- matrix(rep("", nrow(x) * ncol(x)), ncol = 5)
> cell.format[3, ] <- "color{red}"
> latex(x, title = "", file = "", colnamesTexCmd = "itshape", rownamesTexCmd = "bfseries",
+   cgroupTexCmd = "color{green}", rgroupTexCmd = "palatino",
+   cellTexCmds = cell.format, numeric.dollar = FALSE, rgroup = c("First group",
+   "Second group"), n.rgroup = c(4, 2), ctable = TRUE, label = "tab:matrix-2",
+   caption = "A named matrix with formats.")
```

5 Data frame

```
> library(Hmisc)
> data(USArrests)
> x <- head(USArrests)
```

Create a matrix with one element for each cell of the table, and a vector of character strings for the row labels.

```
> cell.format <- matrix(rep("", nrow(x) * ncol(x)), ncol = ncol(x))
> my.rownamesTexCmd <- rep("", nrow(x))
```

I created a new LaTeX command called `\redscshape` in the preamble that changes the font to red and small caps using the following LaTeX code:

```
\newcommand{\redscshape}{\color{red}\scshape}
```

Table 8: Conditional typesetting of a data frame.

	Murder	Assault	UrbanPop	Rape
Beginning with A				
Alabama	13.2	236	58	21.2
ALASKA	10.0	263	48	44.5
ARIZONA	8.1	294	80	31.0
Arkansas	8.8	190	50	19.5
Beginning with C				
CALIFORNIA	9.0	276	91	40.6
COLORADO	7.9	204	78	38.7

This new LaTeX command is used to format the row labels in Table 8. This format is applied to rows where a certain value is exceeded. The cells of the table where this condition is met are formatted separately.

```
> cell.format[which(x$Rape > 25), ] <- "color{red}"
> my.rownamesTexCmd[which(x$Rape > 25)] <- "redscshape"

> latex(x, file = "", colnamesTexCmd = "color{green}", cgroupTexCmd = "bfseries",
+   rgroupTexCmd = "bfseries", cellTexCmds = cell.format, rownamesTexCmd = my.rownamesTexCmd,
+   n.rgroup = c(4, 2), numeric.dollar = FALSE, rgroup = c("Beginning with A",
+     "Beginning with C"), title = "", ctable = TRUE, label = "tab:example",
+   caption = "Conditional typesetting of a data frame.")
```

6 Several column groups

I want to test to make sure that everything works okay with several column groups—the examples above only test two groups and no groups (or should that be one group?). It looks okay (Table 9).

```
> x <- matrix(1:30, ncol = 5)
> y <- letters[1:6]
> x <- data.frame(x, y)
> latex(x, file = "", cgroup = c("Group A", "Group B", "Group C"),
+   n.cgroup = c(2, 2, 2), rgroup = c("First", "Second"), n.rgroup = c(4,
+   2), colnamesTexCmd = "color{green}", cgroupTexCmd = "scshape",
+   rgroupTexCmd = "bfseries", numeric.dollar = FALSE, title = "",
+   ctable = TRUE, label = "tab:severalCgroups", caption = "More than two column groups.")
```

Table 9: More than two column groups.

	GROUP A		GROUP B		GROUP C	
	X1	X2	X3	X4	X5	y
First						
1	1	7	13	19	25	a
2	2	8	14	20	26	b
3	3	9	15	21	27	c
4	4	10	16	22	28	d
Second						
5	5	11	17	23	29	e
6	6	12	18	24	30	f

7 Various LaTeX font attributes

This example uses a variety of LaTeX font attributes. See Table 10. It uses a number of custom commands created in the preamble of the LaTeX file. These commands are shown in Section 1.

```
> LaTeX <- c("tiny", "footnotesize", "scriptsize", "large", "LARGE",
+ "tffamily", "sffamily", "scshape", "itshape", "slshape",
+ "bfseries", "avantgarde", "bookman", "palatino", "newcentury",
+ "textsterling")
> numbers <- c(1:length(LaTeX)) * 100
> x <- data.frame(LaTeX, numbers)
> names(x) <- c("LaTeX Commands", "Numbers")
> cell.format <- matrix(rep("", NCOL(x) * NROW(x)), ncol = NCOL(x))
> cell.format[, 1] <- LaTeX
> cell.format[, 2] <- LaTeX
> my.rownamesTexCmd <- rep("", nrow(x))
> index <- (1:nrow(x)/2) == (1:nrow(x)%/%2)
> my.rownamesTexCmd[index] <- "shadeRow"
> latex(x, title = "", file = "", ctable = TRUE, colnamesTexCmd = c("bfseries",
+ "scshape", "itshape"), cgroupTexCmd = c("scshape", "tffamily",
+ "scshape"), rgroupTexCmd = c("itshape", "scshape"), cellTexCmds = cell.format,
+ rownamesTexCmd = my.rownamesTexCmd, rgroup = c("First row group",
+ "Second row group"), n.rgroup = c(5, nrow(x) - 5), numeric.dollar = FALSE,
+ cgroup = c("Group A", "Group B"), label = "tab:example3",
+ caption = "A variety of font attributes, and some other LaTeX commands. This is ugly, but shows how easy it is t
```

8 Numeric dollar

I think users will need to be careful about the use of the `numeric.dollar` option. The default is `TRUE` if `dcolumn=FALSE` (which is the default). This

Table 10: A variety of font attributes, and some other LaTeX commands. This is ugly, but shows how easy it is to format table font attributes.

	Group A	GROUP B
	L ^A T _E X COMMANDS	Numbers
<i>First row group</i>		
1	tiny	100
2	footnotesize	200
3	scriptsize	300
4	large	400
5	LARGE	500
SECOND ROW GROUP		
6	ttfamily	600
7	sffamily	700
8	SCSHAPE	800
9	<i>itshape</i>	<i>900</i>
10	<i>slshape</i>	<i>1000</i>
11	bfseries	1100
12	avantgarde	1200
13	bookman	1300
14	palatino	1400
15	newcentury	1500
16	£textsterling	£1600

Table 11: A variety of font attributes, this time without `numeric.dollar = FALSE`.

TITLE	Group A L ^A T _E X COMMANDS	GROUP B Numbers
<i>First row group</i>		
1	<code>tiny</code>	100
2	<code>footnotesize</code>	200
3	<code>scriptsize</code>	300
4	<code>large</code>	400
5	<code>LARGE</code>	500
SECOND ROW GROUP		
6	<code>ttfamily</code>	600
7	<code>sffamily</code>	700
8	<code>scshape</code>	800
9	<code>itshape</code>	900
10	<code>slshape</code>	1000
11	<code>bfseries</code>	1100
12	<code>avantgarde</code>	1200
13	<code>bookman</code>	1300
14	<code>palatino</code>	1400
15	<code>newcentury</code>	1500
16	<code>£textsterling</code>	£1600

puts the output in math mode which is generally a good thing. However, math mode works differently to normal text mode and will ignore (?) some additional typesetting commands. Table 11 is the same as Table 10 except that `numeric.dollar=TRUE`. Colours and font sizes work, font shapes and series do not.

```
> latex(x, title = "title", file = "", ctable = TRUE, colnamesTexCmd = c("bfseries",
+ "scshape", "itshape"), cgroupTexCmd = c("scshape", "ttfamily",
+ "scshape"), rgroupTexCmd = c("itshape", "scshape"), cellTexCmds = cell.format,
+ rownamesTexCmd = my.rownamesTexCmd, rgroup = c("First row group",
+ "Second row group"), n.rgroup = c(5, nrow(x) - 5), cgroup = c("Group A",
+ "Group B"), label = "tab:example4", caption = "A variety of font attributes, this time without numeric.dolla
```

Table 12: Colouring individual cells.

x	V1	V2	V3
1	-0.1925971	0.0000000	0.0000000
2	0.0000000	0.8005406	0.0000000
3	0.0000000	0.0000000	0.3707994

9 Column names, column groups and title

If `cgroups` are specified `title` gets its format from `cgroupTexCmd`, but if there are *no* `cgroups` then `title` gets its format from `colnamesTexCmd`.

If column groups are specified then the format of `title` is the first item in `cgroupTexCmd`, even if the `title` is set to the empty string. This means that if, for some reason that I currently cannot imagine, the user wants to use different formats for each of the column headings the vector of formats must include one for the title at the beginning.

Note also that there is a blank column separating each column group. At the moment `cgroupTexCmd` is not set up to add this so if the user wants to use a vector of different formats the user will need to create a vector that includes the blank columns. This also applies to `colnamesTexCmd`.

But I really can't see why people are going to want to format different column headers or column group labels differently. Perhaps this is a non-issue.

10 Shading individual cells

This one is from Dieter Menne. You need to have a reasonably up-to-date version of the latex package `colortbl`. Ubuntu Breezy ships with an (astonishingly?) old version that does not contain the `\cellcolor` command. I had to get the latest version from CTAN.

See table 12.

```
> x <- as.data.frame(diag(rnorm(3), nrow = 3))
> cellTex <- matrix(rep("", NROW(x) * NCOL(x)), nrow = NROW(x))
> cellTex[2, 2] <- "cellcolor{red}"
> latex(x, file = "", cellTexCmds = cellTex, numeric.dollar = FALSE,
+   label = "tab:colourcells", caption = "Colouring individual cells.")
```