Using table1 with LaTeX

Benjamin Rich

2021-04-21

Introduction

A long reqested feature has been the ability to use `table1` within LaTeX documents. Since version 1.4 of `table1`, this is now possible (with some limitations) by converting the output of `table1()` to a `data.frame`, `kableExtra` or `flextable`, using the functions `as.data.frame()`, `t1kable()` and `t1flex()` respectively, as these objects can be rendered to LaTaX (note: `data.frame` (via pandoc) and `flextable` objects can also be rendered to .docx format, while, `kableExtra` cannot).

Examples

We demonstrate this using a familiar example from the main vignette. First, we can try the `data.frame` approach:

```
x <- table1(~ age + sex + wt | treat, data=dat)
as.data.frame(x)
```

```
##                                  Placebo           Treated           Overall
## 1                                 (N=52)            (N=94)           (N=146)
## 2          Age (years)
## 3            Mean (SD)       39.2 (14.2)       40.1 (13.3)       39.8 (13.6)
## 4     Median [Min, Max] 37.5 [18.0, 65.0] 39.5 [18.0, 65.0] 39.0 [18.0, 65.0]
## 5                   Sex
## 6                Female      34.0 (65.4%)      53.0 (56.4%)      87.0 (59.6%)
## 7                  Male      18.0 (34.6%)      41.0 (43.6%)      59.0 (40.4%)
## 8          Weight (kg)
## 9            Mean (SD)       68.1 (16.3)       68.3 (16.7)       68.2 (16.5)
## 10    Median [Min, Max]  66.7 [37.5, 116]  64.9 [40.0, 119]  66.2 [37.5, 119]
## 11              Missing       2.00 (3.8%)       3.00 (3.2%)       5.00 (3.4%)
```

By default, this does not produce a LaTeX table, but the same text output you would see in the R console. If the `printr` package is loaded, however, we do get a LaTeX table by default:

```
library(printr, quietly=TRUE)
```

```
## Registered S3 method overwritten by 'printr':
##   method                from
##   knit_print.data.frame rmarkdown
```

```
as.data.frame(x)
```

	Placebo	Treated	Overall
	(N=52)	(N=94)	(N=146)
Age (years)			
Mean (SD)	39.2 (14.2)	40.1 (13.3)	39.8 (13.6)
Median [Min, Max]	37.5 [18.0, 65.0]	39.5 [18.0, 65.0]	39.0 [18.0, 65.0]
Sex			
Female	34.0 (65.4%)	53.0 (56.4%)	87.0 (59.6%)
Male	18.0 (34.6%)	41.0 (43.6%)	59.0 (40.4%)
Weight (kg)			
Mean (SD)	68.1 (16.3)	68.3 (16.7)	68.2 (16.5)
Median [Min, Max]	66.7 [37.5, 116]	64.9 [40.0, 119]	66.2 [37.5, 119]
Missing	2.00 (3.8%)	3.00 (3.2%)	5.00 (3.4%)

Alternatively, we can use the `knitr::kable()` function:

```
library(printr, quietly=TRUE)
kable(as.data.frame(x), booktabs=TRUE)
```

	Placebo	Treated	Overall
	(N=52)	(N=94)	(N=146)
Age (years)			
Mean (SD)	39.2 (14.2)	40.1 (13.3)	39.8 (13.6)
Median [Min, Max]	37.5 [18.0, 65.0]	39.5 [18.0, 65.0]	39.0 [18.0, 65.0]
Sex			
Female	34.0 (65.4%)	53.0 (56.4%)	87.0 (59.6%)
Male	18.0 (34.6%)	41.0 (43.6%)	59.0 (40.4%)
Weight (kg)			
Mean (SD)	68.1 (16.3)	68.3 (16.7)	68.2 (16.5)
Median [Min, Max]	66.7 [37.5, 116]	64.9 [40.0, 119]	66.2 [37.5, 119]
Missing	2.00 (3.8%)	3.00 (3.2%)	5.00 (3.4%)

The output here is a bit nicer because we have specified the `booktabs` option, but because we are talking about a simple `data.frame`, there is not option to specify formatting (like bold text for variable labels, for instance).

Next, we can try the `t1kable()` function, to produce a `kableExtra` object:

```
t1kable(x)
```

	Placebo	Treated	Overall
	(N=52)	(N=94)	(N=146)
Age (years)			
Mean (SD)	39.2 (14.2)	40.1 (13.3)	39.8 (13.6)
Median [Min, Max]	37.5 [18.0, 65.0]	39.5 [18.0, 65.0]	39.0 [18.0, 65.0]
Sex			
Female	34.0 (65.4%)	53.0 (56.4%)	87.0 (59.6%)
Male	18.0 (34.6%)	41.0 (43.6%)	59.0 (40.4%)
Weight (kg)			
Mean (SD)	68.1 (16.3)	68.3 (16.7)	68.2 (16.5)
Median [Min, Max]	66.7 [37.5, 116]	64.9 [40.0, 119]	66.2 [37.5, 119]
Missing	2.00 (3.8%)	3.00 (3.2%)	5.00 (3.4%)

	Placebo (N=52)	Treated (N=94)	Overall (N=146)
Age (years)			
Mean (SD)	39.2 (14.2)	40.1 (13.3)	39.8 (13.6)
Median [Min, Max]	37.5 [18.0, 65.0]	39.5 [18.0, 65.0]	39.0 [18.0, 65.0]
Sex			
Female	34.0 (65.4%)	53.0 (56.4%)	87.0 (59.6%)
Male	18.0 (34.6%)	41.0 (43.6%)	59.0 (40.4%)
Weight (kg)			
Mean (SD)	68.1 (16.3)	68.3 (16.7)	68.2 (16.5)
Median [Min, Max]	66.7 [37.5, 116]	64.9 [40.0, 119]	66.2 [37.5, 119]
Missing	2.00 (3.8%)	3.00 (3.2%)	5.00 (3.4%)

This looks a bit better: it usese the `booktabs` option by default, and also has bold variable labels. But because of a limitation with multiline headers (i.e. headers that contain line breaks), the `N=XX` for each column is placed in the first row of the table, which is less than ideal.

Finally, we can try using the `t1flex()` function to produce a `flextable` object:

```
t1flex(x)
```

The output is less attractive in my opinion: it uses a different font which is less LaTeX-like (and less consistent with the rest of the document), doesn't have bold labels or line break in the column headers (also `flextable` places the table in a float, whereas `kableExtra` leaves it inline). Hence, I personally would not use `flextable` in a LaTeX document, I would use `kableExtra`, but a big advantage of `flextable` is that in can render to .docx format (i.e. Microsoft Word), which `kableExtra` cannot.

Nested groups

Nested groups are supported with `kableExtra` and `flextable`, but not simple `data.frames`. Here, one example with `kableExtra`:

```
x2 <- table1(~ age + wt | treat*sex, data=dat, overall=FALSE)
t1kable(x2)
```

	Placebo		Treated	
	Female	Male	Female	Male
	(N=34)	(N=18)	(N=53)	(N=41)
Age (years)				
Mean (SD)	40.6 (14.5)	36.6 (13.6)	40.1 (13.4)	40.1 (13.3)
Median [Min, Max]	39.5 [18.0, 65.0]	33.5 [18.0, 64.0]	39.0 [18.0, 65.0]	41.0 [18.0, 65.0]
Weight (kg)				
Mean (SD)	68.8 (14.8)	66.8 (19.3)	65.6 (15.1)	71.5 (18.0)
Median [Min, Max]	67.2 [45.8, 116]	66.6 [37.5, 105]	61.4 [41.9, 103]	68.3 [40.0, 119]
Missing	1.00 (2.9%)	1.00 (5.6%)	3.00 (5.7%)	0 (0%)

Closing remarks

As of version 1.4, it is now possible to use `table1` within LaTeX documents. There are some limitations on the formatting, however (it looks fine, but not quite the same as the HTML output and there is less flexibilty to control it).

The actual LaTeX code generation is handled by an external package (i.e., `knitr`, `kableExtra` or `flextable`). The LaTeX generated by `kableExtra` and `flextable` is very different, and in my opinion `kableExtra` produces the better result. Both `kableExtra` and `flextable` have an extensive number of functions that can be applied to the converted object in order to alter the appearance of the table but these will not be described here; refer to each package's documentation for the complete details.