Design Principles, Comparisons and Limitations

David Hugh-Jones

2018-10-03

This document briefly describes the design of huxtable, and compares it with other R packages for creating tables. A current version is on the web in HTML or PDF formats.

Design principles

I wrote this package because I wanted a simple way to create tables in my LaTeX documents. At the same time, I wanted to be able to output HTML or Markdown for use in RStudio. And, I wanted to be able to edit tables intuitively using standard R features. My typical use case is creating tables of regression outputs, but I also wanted to be able to represent arbitrary data, like a table of descriptive statistics or of plain text.

The idea behind huxtable is to store data in a normal data frame, along with properties that describe how to display the data, at cell, column, row or table level. Operations on the data frame work as normal, and they also affect the display properties. Then, the data can be output in an appropriate format. At the moment, those formats are LaTeX, HTML, Markdown, Word/Excel/Powerpoint, RTF and on-screen pretty printing. More could be added.

Another design choice was to have separate functions per feature. Many existing packages use a single function with a large number of options. For instance, `print.xtable` in the `xtable` package has 34 options, and `texreg` in the `texreg` package has 41. Having one function per feature should make life easier for the end user. It should also lead to clearer code: each function starts with a valid huxtable, changes one thing, and returns a valid huxtable.

The output formats are very different, and decisions have to be made as to what any package will support. My background is more in HTML. This is reflected in some of the huxtable properties, like per-cell borders and padding. The package tries to keep output reasonably similar between LaTeX and HTML, but there are inevitably some differences and limitations (see below). For Markdown and on-screen output, obviously, only a few basic properties are supported.

The package makes no attempt to output beautiful HTML or LaTeX source code. In fact, in the case of LaTeX, it's pretty ugly. The approach is "do what it takes to get the job done".

Comparing Huxtable With Other Packages

R has many different packages to create LaTeX and HTML tables. The table(s) below list those I know and the features they have. The table is produced with huxtable, of course ;-)

This comparison doesn't necessarily tell you the important stuff: how easy is the interface? Is the code currently maintained? I have not used all these packages, but my personal (and subjective) recommendations are:

- `texreg` is very good for producing regression tables. It can cope with a huge variety of inputs. Hopefully, with the advent of broom, it will get easier for many packages to do this.
- `ztable` seems to support a lot of functionality, though I haven't used it.
- `xtable` is old, but reliable and widely available.
- `tables` has an interesting interface for producing summary statistics. It looks complex but powerful.
- `pixiedust` is quite close to `huxtable`. It has many features and is well-written. The interface is slightly different: you use `sprinkle()` to add features to a tidy data frame.

- Lastly, `formattable` is a new kid on the block. It has well-written code and some interesting ideas. It's HTML-only at present.

Limitations

Some people love LaTeX. Other people think they have to use it to be "scientific". (Sadly, this is all too common in my field.) Personally, I can tolerate it at a distance. It's certainly not easy to produce LaTeX code combining a wide variety of table features. Current limitations of huxtable include:

- You can't change horizontal border widths within a single line.
- Table width may be unpredictable and your text may spill over. Adjust on a trial and error basis.
- If table position is problematic, check that your table is wide enough, and set `width()` if necessary.
- Vertical alignment is unlikely to work as you might expect.

Some of these may be fixed... at some point!

There are also limitations in HTML:

- Rotation is likely to mess up your cells. This won't be fixed until there is a *simple* way to do cell rotation in CSS. (There are plenty of complicated ways.)

Lastly, as mentioned above, HTML and LaTeX output is likely to differ. For example, in LaTeX, height is set by putting the table inside a \resizebox object. That can obviously mess up other sizes like table width. Your Mileage May Vary.

Feel free to report bugs at github, or to email me (davidhughjones@gmail.com) with brickbats and bouquets.

Table 1: Comparison table, part 1

	huxtable	xtable	formattable	ascii	kableExtra	condformat	htmlTable
HTML output	Y	Y	Y	Y	Y	Y	Y
Background color	Y		Y	Y		Y	Y
Width and height	Y		Y	Y	(Y)		
Text formatting	Y		Y	Y	(Y)		
Borders	Y						Y
Rotation	Y						
Multicolumn/multirow	Y			Y			Y
Alignment	Y			Y			Y
Numeric formatting	Y		Y	Y			Y
Captions	Y	Y		Y			
Format NA	Y			Y			
LaTeX output	Y	Y		(Y)	Y	Y	?
Background color	Y			Y		Y	
Width and height	Y	Y		Y	(Y)		
Text formatting	Y			Y	(Y)		
Borders	Y	(Y)					
Rotation	Y	Y					
Multicolumn/multirow	Y			Y			
Alignment	Y	Y		Y			
Numeric formatting	Y	Y		Y			
Captions	Y	Y		Y			
Format NA	Y			Y			
Long tables		Y					
Labels	Y	Y					
Other features							
Conditional formats	Y		Y			Y	
Autocreate summary statistics							
Dynamic HTML widgets			Y				
Create tables from model objects	Y	Y		Y			
dplyr compatible	Y						
Add footnotes	(Y)				Y		
Other formats							
Markdown	Y						
Screen	Y						
Excel	Y					Y	
Powerpoint	Y						
Word	Y						
DocBook					Y		
RTF	Y						
Notes					Via markdown		

A (Y) means that there is limited support for the feature. For example, multirow cells may only be supported in headers, or only horizontal border lines may work.

Table 2: Comparison table, part 2

	Hmisc::latex	ztable	kable	texreg	stargazer	tables	DT	pander	flextable	pixiedust
HTML output	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Background color		Y							Y	Y
Width and height									Y	Y
Text formatting		(Y)		(Y)	Y			Y	Y	Y
Borders		Y							Y	Y
Rotation									Y	Y
Multicolumn/multirow	(Y)	Y			(Y)	(Y)			Y	Y
Alignment	Y	Y	Y		Y	Y		Y	Y	Y
Numeric formatting	Y	Y	Y		Y	Y		Y	Y	Y
Captions	Y	Y	Y				Y	Y	Y	Y
Format NA	Y					Y		Y	Y	Y
LaTeX output	Y	Y	Y	Y	Y	Y		Y		Y
Background color										Y
Width and height										Y
Text formatting		(Y)		(Y)	Y			Y		Y
Borders	(Y)	Y				(Y)				Y
Rotation										Y
Multicolumn/multirow	(Y)	Y			(Y)					Y
Alignment	Y	Y	Y		Y	Y		Y		Y
Numeric formatting	Y	Y	Y		Y	Y		Y		Y
Captions	Y	Y	Y					Y		Y
Format NA	Y					Y		Y		Y
Long tables	Y	Y				Y				Y
Labels	Y	Y			Y	Y				Y
Other features										
Conditional formats									Y	
Autocreate summary statistics						Y				
Dynamic HTML widgets							Y			
Create tables from model objects		Y		Y	Y				Y	
dplyr compatible										
Add footnotes									Y	
Other formats										
Markdown			Y					Y		Y
Screen					Y					Y
Excel										
Powerpoint									Y	
Word									Y	(Y)
DocBook										
RTF										
Notes	HTML via external translator						Via Javascript	Via markdown		

A (Y) means that there is limited support for the feature. For example, multirow cells may only be supported in headers, or only horizontal border lines may work.