Creating an article with rmarkdown

John Doe[a], Jane Doe[b]

[a] *Institute of RMarkdown, Townville*
[b] *RMarkdown Academy, Townville*

Abstract

This is an example document showing some of the features required to make a document that will output to both pdf and word

Keywords: rmarkdown

1. Compiling this document

This document assumes you have pandoc 2.8 and R 3.6 with sufficiently recent markdown (Allaire et al. 2019, xie2018), rticles (R. authors 2019), huxtable (hughjonesd 2019), and flextable packages installed.

To compile from R we then run the following commands

```
> library(rmarkdown)
> rmarkdown::render('example.Rmd', output_format='all')
```

This should produce both a pdf and word document.
Some things that don't quite work are:

- The table code shown in the pdf output is not the same as the actual code. This is due to an interplay between the mechanism this document uses to allow cross-referencing of tables in Word documents and removal the relevant tags (and escaping of backslashes) for the pdf document.
- The table column width meanings seem to be different between the word output and the pdf output

2. PDF templates

We will be using the rticles LaTeX templates to produce a pdf of this document. Since we are using pandoc 2.8 we also require a recent version of rticles. This is because pandoc 2.8 requires LaTeX templates to contain the cslreferences environment. For rticles this was introduced in commit `1c5dfcd`.

To enable cross-referencing of figures and tables in markdown in a uniform manner which also works for producing word documents we will also use the bookdown (B. authors 2019).

The section of the YAML header of this document required to output to pdf looks like

```
bookdown::pdf_book:
    base_format: rticles::elsevier_article
    keep_tex: true
    toc: false
    pandoc_args:
      - --lua-filter=captionref.lua
```

We specify that we are outputting a bookdown `pdf_book` and specify a `base_format` or `rticles::elsevier_article` to make use of the rticles template for elsevier articles. The inclusion of `keep_tex` is optional and was used for debugging purposes. We set `toc` to false to explicitly specify we don't want a table of contents. We mention the use of lua-filters a bit further.

3. Multiple authors

The current solution to this in this document is not entirely statisfactory since it requires data being repeated in the YAML header. One entry for word documents and one for pdf documents. However, they are at least able to co-exist together simultaneously in the same document.

3.1. Word documents

To produce a title page with multiple authors we make use of some lua filters. These are `scholarly-metadata` and `author-info-blocks` which are available in the `lua-filters` repository (luafilter authors 2019).

Firstly we add these to the YAML under the `word_document2` section like so

```
bookdown::word_document2:
    pandoc_args:
      - --lua-filter=scholarly-metadata.lua
      - --lua-filter=author-info-blocks.lua
```

This will tell pandoc to produce a word document and to run the two filters *in order*.

These filters look for meta-data in the header about the authors in a particular format and then use this to produce the text in the word document.

The relevant lines in our document are

```
author:
  - name: John Doe
    institute: [inst1]
  - name: Jane Doe
    institute: [inst2]
institute:
  - inst1: Institute of RMarkdown
  - inst2: RMarkdown Academy
```

3.2. PDF document

The `elsevier_article` template we are using does not require these filters to run. It uses the same author names format but requires affiliations and addresses (among other optional meta-data). We can add this information into the YAML we had in the previous section like so

```
author:
  - name: John Doe
    institute: [inst1]
    affiliation: add1
  - name: Jane Doe
    institute: [inst2]
    affiliation: add2
institute:
  - inst1: Institute of RMarkdown
  - inst2: RMarkdown Academy
address:
  - code: add1
    address: Institute of RMarkdown, Townville
  - code: add2
    address: RMarkdown Academy, Townville
```

4. Figure example

Here we make Figure 1. In this example bookdown is enabling us to reference the figure. The figure is defined as

The R cell is given the name `flow`. Bookdown then produces a label for this figure by appending `fig:` to the name. We can then reference this in the document using `\@ref`.

5. Table example

Here we make Table 1. We use huxtable to make the table as it outputs to both word and LaTeX. For huxtable tables we currently need to add a marker in the `set_caption` function to give the table a label. Unfortunately without further intervention this marker remains in the document. However we can make use of a custom lua filter to remove these markers after they serve their purpose. In this document these are found in `captionref.lua`.

```
hdr1 <- hux("Sub-header for FooBar", "", "")
hdr2 <- hux("Sub-header for FizzBuzz", "", "")

example.huxtable <- as_hux(example.df)

example.huxtable %>%
```

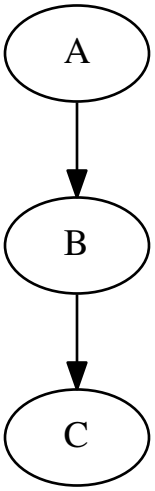

Figure 1: An example of a figure.

```
add_colnames %>%
set_bold(1, everywhere, TRUE) %>%
set_bottom_border(1, everywhere, 2) %>%
set_top_border(1, everywhere, 2) %>%
add_rows(hdr1, 1) %>%  # the row is added **after** the specified row (new row is row 2)
merge_cells(2, c(1, 3)) %>%
add_rows(hdr2, 4) %>%  # at this point "Bar" is in row 4
merge_cells(5, c(1, 3)) %>%
add_footnote("This is the footnote text for the table") %>%
set_caption("( #tab:table)Table \@ref(tab:table): Caption text for table") %>%
set_caption_pos("bottom") %>%
set_col_width(c(2.0, 3.0, 1.5)) %>%
set_col_width(c(.3, .3, .3)) %>%
set_contents(1, 1, "New Col 1") %>%
set_width(0.9)
```

New Col 1	**Col 2**	**Col 3**
Sub-header for FooBar		
Foo	1	1.12
Bar	2	1.54
Sub-header for FizzBuzz		
Fizz	3	4.12
Buzz	4	4.54

This is the footnote text for the table

Table 1: : Caption text for table

Allaire, JJ, Yihui Xie, Jonathan McPherson, Javier Luraschi, Kevin Ushey, Aron Atkins, Hadley Wickham, Joe Cheng, Winston Chang, and Richard Iannone. 2019. *Rmarkdown: Dynamic Documents for R*. https://github.com/rstudio/rmarkdown.

authors, Bookdown. 2019. "Bookdown: Authoring Books and Technical Documents with R Markdown." 2019. https://github.com/rstudio/bookdown.

authors, Rticles. 2019. "Rticles: LaTeX Journal Article Template for Rmarkdown." 2019. https://github.com/rstudio/rticles.

authors. 2019. "Lua-Filters: A Collection of Lua Filters for Pandoc." 2019. https://github.com/pandoc/lua-filters.

hughjonesd. 2019. "Huxtable." 2019. https://github.com/hughjonesd/huxtable.