Bare Demo of RMarkdown port of IEEEtran.cls for IEEE conferences

Amy Chan[§], Michael Shell[*], Homer Simpson[†], James Kirk[‡] and John Doe[† ‡]
[*] School of Electrical and Computer Engineering
Georgia Institute of Technology, Atlanta, Georgia 30332–0250
[§] Email: mathematical.coffee@gmail.com
[†] Twentieth Century Fox, Springfield, USA
[‡] Starfleet Academy, San Francisco, California 96678–2391
Telephone: (800) 555–1212, Fax: (888) 555–1212

***Abstract*—We demonstrate the use of an RMarkdown template for the IEEEtran style. For now it's just conference style, but we will eventually expand (maybe).**

I. Introduction

This demo file is intended to serve as a "starter file" for IEEE conference papers produced under LaTeX using IEEEtran.cls version 1.8b and later. I wish you the best of success.

mds

August 26, 2015

Use `rmarkdown::render()` to create this document; it essentially calls `knit()` to go from RMD to MD, and then `pandoc` (with all the configurations in the YAML) to go from MD to PDF.

One could try to compile to HTML, but of course none of the IEEE styles will be applied. And if any raw LaTeX has been put in the document (as is typically the case for a paper, as you may need the extra power of LaTeX to provide specific layout), this will of course not compile in HTML.

II. Examples

A. Knitr

You can use knitr as usual. The `echo=F` chunk option should probably be set (unless you want to show the R code in the paper). Also since this is a two-column layout it'll probably overflow, so you will need to either

- wrap the code yourself (by default knitr does not tidy code), or
- enable code tidying and specify the width: `opts_knit$set(tidy=T, tidy.opts=list(width.cutoff=40))`.
- NB: the `size` chunk option (e.g. `opts_chunk$set(size="small")` only works in Rnw, not in Rmd).

The width is pretty small. For this document, you can fit about 42 characters before it overflows off the side (see the example in section II-B).

B. Figures

You can of course generate plots using R and they will be inserted with knitr. However, since knitr goes from MD to RMD, they will be inserted with markdown format, not TeX format. I have configured knitr to put figures in the `figure/` directory (`opts_chunk$set(fig.path='figure/')`) so that is where the plot will be.

```
plot(Sepal.Length ~ Species, iris)
```

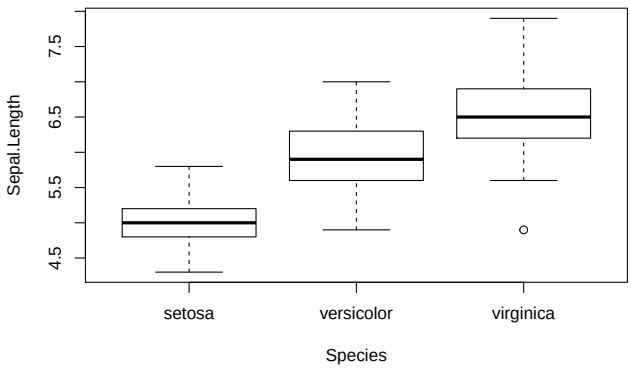


Fig. 1: Sepal lengths for various species of iris.

See figure 1. (I am unsure why this is "Fig. 1" in the caption. . . is it a knitr/rmarkdown/pandoc thing, or a IEEEtran thing?)

In practice, you will probably want to write your figure code in raw LaTeX for greater control. In the setup chunk of this Rmd is a function `latex.figure` which is an example of outputting raw LaTeX for a figure. Tweak as you wish. (Surely there's a library like `xtable` for this?)

```
latex.figure(
  'figure/iris.plot-1.pdf',
  caption='Another plot of sepal lengths
```

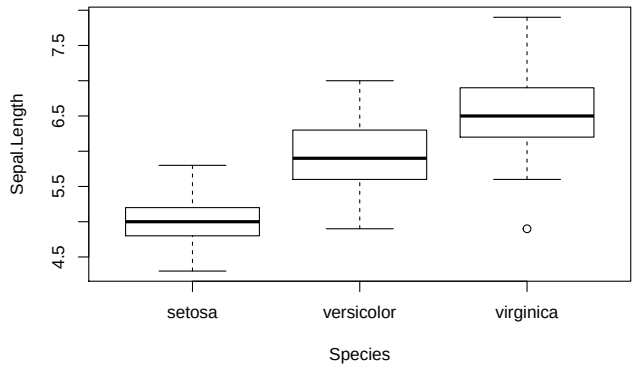


Fig. 2: Another plot of sepal lengths for the various species of iris.

```
          for the various species of iris.'
  label='fig:iris2')
```

The `latex.figure` also has basic support for subfloats: just provide multiple paths. If there are as many captions as figures, one is used for each. If there is one more than the number of figures, the first is used as the "master" caption and the rest as subfigure captions. If there is only one caption, it's used for the figure and no subcaptions are added. See figure 3 for the result.

```
# generate and save some pictures
n = 1:5
figs = sprintf('figure/x%i.png', n)
for (nn in n) {
  png(filename=figs[nn], width=480, height=3[illegible]
  plot(1:10, (1:10)^nn)
  dev.off()
}

# show as floating figure with 3 subfig
latex.figure(
  figs,
  caption=c("Polynomials",
            sprintf("$x^%i$", n)),
  label='fig:polynomials',
  linebreaks.after=3,
  width='.6\\columnwidth',
  floating=T)
```

Note that often IEEE papers with subfigures do not employ subfigure captions, but instead will reference/describe all of them (a), (b), etc., within the main caption.

Note that the IEEE typically puts floats only at the top, even when this results in a large percentage of a column being occupied by floats.

TABLE I: Example of the iris dataset

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
6.80	2.80	4.80	1.40	versicolor
5.60	2.70	4.20	1.30	versicolor
6.90	3.10	5.10	2.30	virginica
5.40	3.00	4.50	1.50	versicolor
4.90	3.10	1.50	0.20	setosa
5.10	3.30	1.70	0.50	setosa

C. Tables

You should not use the pandoc syntax, because it uses the `longtable` package (this is hard-coded in) and `longtable` doesn't play well with two column input. Use something like Hmisc or xtable to give LaTeX output and provide extra control (e.g. table I).

```
print(xtable(
  iris[sample(nrow(iris), 6), ],
  caption='Example of the iris dataset',
  label='tbl:iris.xtable',
  align=c(rep('r', 5), 'l')))
```

You may wish the table to span multiple columns. Use `table*` instead of `table` (table II). Note that the `floating.environment` is an argument to `print.xtable`, not to `xtable`.

```
print(xtable(
    head(mtcars),
    caption='Example of the motor trend
             car road tests dataset',
    label='tbl:xtable.floating'),
  floating.environment='table*')
```

Note that, for IEEE style tables, given that table captions serve much like titles, captions are usually capitalized except for words such as a, an, and, as, at, but, by, for, in, nor, of, on, or, the, to and up, which are usually not capitalized unless they are the first or last word of the caption. Table text will default to `\footnotesize` as the IEEE normally uses this smaller font for tables.

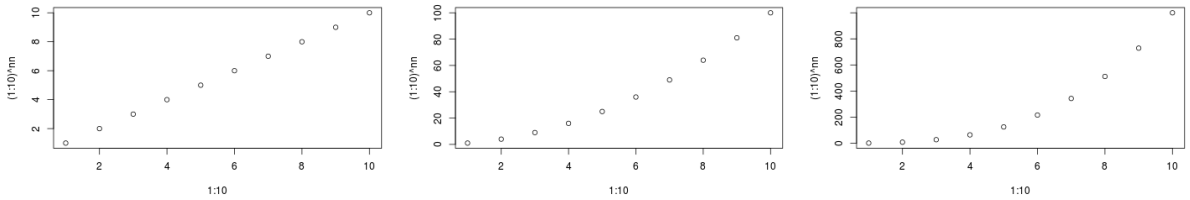
Note that the IEEE typically puts floats only at the top, even when this results in a large percentage of a column being occupied by floats.

D. Citing

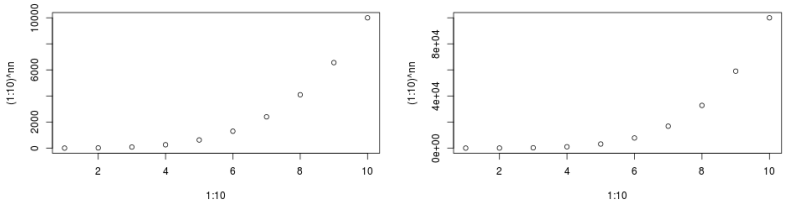
Examples of citing one author [1] and two authors [1, 2].

E. Equations

Are as you would hope. You can use pandoc-crossref syntax to do labels. i.e.



(a) x^1 (b) x^2 (c) x^3



(d) x^4 (e) x^5

Fig. 3: Polynomials

TABLE II: Example of the motor trend car road tests dataset

mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
21.00	6.00	160.00	110.00	3.90	2.62	16.46	0.00	1.00	4.00	4.00
21.00	6.00	160.00	110.00	3.90	2.88	17.02	0.00	1.00	4.00	4.00
22.80	4.00	108.00	93.00	3.85	2.32	18.61	1.00	1.00	4.00	1.00
21.40	6.00	258.00	110.00	3.08	3.21	19.44	1.00	0.00	3.00	1.00
18.70	8.00	360.00	175.00	3.15	3.44	17.02	0.00	0.00	3.00	2.00
18.10	6.00	225.00	105.00	2.76	3.46	20.22	1.00	0.00	3.00	1.00

```
$$
e = m c^2
$$ {#eq:einstein}
```

yields

$$e = mc^2. \tag{1}$$

One can use `@eq:einstein` to refer to the equation, e.g. 1. The only caveat is that the equation needs to be in its own paragraph if you wish to number it, meaning that in the resultant tex and pdf, the equation is on its own line. (If you don't wish to number the equation, it doesn't have to be on its own paragraph and will render in the paragraph as you would expect).

I haven't found a good fix for this yet. It is a requirement of `pandoc-crossref`. You have to go to the TeX and remove these extra blank lines (where appropriate) before compiling. I add a comment `% FIXME ALIGNMENT` to these equations to make them easier to find.

III. Conclusion

Hopefully you have been given a brief tour of the capabilities of this setup and will now go forth and author IEEEtran-style papers using RMarkdown with (relative) ease.

Acknowledgement

This template would not be possible without Michael Shell's IEEEtran files pandoc, pandoc-crossref, knitr, rmarkdown, and heavy googling within StackOverflow. And props to Rstudio too. It's not required for this, but it certainly makes the whole process much easier. And anyone else I forgot.

References

[1] J. Besag, "Spatial interaction and the statistical analysis of lattice systems," *Journal of the Royal Statistical Society B*, pp. 192–236, 1974.

[2] ——, "On the statistical analysis of dirty pictures," *Journal of the Royal Statistical Society B*, pp. 259–302, 1986.