

Paul Johnson, Director, CRMDA <[crmda@ku.edu](mailto:crmda@ku.edu)>  
Brent Kaplan, GRA, CRMDA <[crmda@ku.edu](mailto:crmda@ku.edu)>  
February 15, 2018

---

## 1 Introduction

### 1.1 Insert author information in the yaml header

It is necessary for the author to edit the yaml header of this document to specify the title, author data, address and the name of a logo file.

## 2 We use a template file

The look and feel of the document will be controlled by a template file.

The author will need to insert a title and names of authors, but settings in the yaml header will not generally alter the format of the output. This is an odd quirk of the Rmarkdown and pandoc framework. If the template is specified, then the document header settings are overridden. That is to say, when the template is specified, then the yaml header settings that Rmarkdown authors expect to revise, such as “toc” (table of contents) will be ignored.

Even stranger still, the declaration of a template in the header of the document will be overridden by a specification of a template in the compilation process.

If this document is edited in Rstudio, the “knit” function will probably not work as intended. The “knit” feature of Rstudio is not aware of the procedure needed to override the template settings. In order to override the template, arguments to the compiler functions must be edited. Please see the next section for details.

## 3 To compile this document

The document can be compiled either with a shell script that is provided with this document, “rmd2pdf.sh”, or by using a function of the same name that is included in the package. The ins-and-outs of this are described more fully in the main vignette for the package.

To repeat, to convert the Rmd file into pdf, either start R and use the function from this package, `rmd2pdf()` or use the shell script that `rmd2pdf.sh` file.

A person who runs the command line to compile `skeleton.Rmd` might specify many options, such as

```
$ ./rmd2pdf.sh --toc=FALSE --type='report'
  --template='theme/report-boilerplate.tex' skeleton.Rmd
```

However, if one starts an R session and loads this package, the equivalent would be

```
> rmd2pdf("skeleton.Rmd", toc=FALSE, type="report",
template="theme/report-boilerplate.tex")
```

Compiling the document will also produce an R file that is a “tangled” version of the code chunks in the document.

### 3.1 Document customization: essentials

The function `rmd2pdf` (same as the script) supplies settings, including a  $\text{\LaTeX}$  template called “report-boilerplate.tex”. That boilerplate is available in the R package and a copy is placed in the `theme` folder when the document is compiled the first time (there is an R chunk above called `texcopy` which does that work). After that, the author is allowed to revise the “report-boilerplate.tex” file and the revised version will be used by the compiler script, but only if that template file is explicitly requests in the function calls described in the previous section.

It turns out that the `yaml` header is mostly irrelevant, then, providing a most unusual Rmarkdown experience. That is the quirk of the template.

The fact that the template must be specified in our `rmd2pdf` script is caused by a design decision that we made early in the process. Because our address and author information has a peculiar style, it is necessary to use the template structure that we provide.

In summary, users can compile this once, and then edit the template file in the “theme” folder, and after that the `rmd2pdf` compiler needs to be told to use it

## 4 For math, much $\text{\LaTeX}$ Syntax is allowed

This is explained in the `crmda` package vignette `Rmarkdown`. We find that not all  $\text{\LaTeX}$  markup will work well, but most will. Users are expected to proof read their output to fine out if math markup worked well. If it does not work, the result is not an error, but rather “empty white space” where the user wanted mathematics.

Use `\[` and `\]` for display equations:

$$\Sigma_{gt} = \Lambda_{gt} \Psi_{gt} \Lambda'_{gt} + \Theta_{gt}$$

## 5 Additional LaTeX features

In addition, if you insert  $\text{\LaTeX}$  features that require packages that are not currently in the template `report-boilerplate.tex`, then those packages can be inserted into the preamble by YAML header markup like so:

```
header-includes:
- \usepackage{xcolor}
- \usepackage{amsmath}
- \usepackage{amssymb}
- \usepackage{fancybox}
```

That example has the package `amssymb` because some non-standard math symbols were needed.

## 6 R code chunks

In our report style, the author will not generally insert visible code chunks, so almost always the chunk will have the flag `include=FALSE` or, if the chunk is included, the code will not be echoed, but perhaps a  $\text{\LaTeX}$  mark-up table or a figure may be placed into the document.

The process for doing this depends on the document type. As explained in the `crmda` vignette `code_chunks`, the appearance of code chunks—whether they are revealed in the document at all—is controlled by many options are available for code chunks.

One approach might be to use one document to create graphs or tables, which are then to be saved in a folder (such as our `tmpout` folder, which is used in this document). This chunk code will create the output file “`tmpout/p-hist-1.pdf`”

```
```{r hist, include=F}
x <- rnorm(1000)
hist(x, main = "A Histogram", xlab = "Random Normal Data, N = 1000")
```
```

There are several R packages intended to create “ready to publish”  $\text{\LaTeX}$  tables. One of the oldest and most venerable of these is `xtable`, which we use here to create a simple table that displays as a cross tabulation table.

When the  $\text{\LaTeX}$  output is going directly into the document, the chunk flag is “`results=‘asis’`” and the `echo` parameter should be `FALSE`. The default configuration for `xtable` is to create tables that are floating  $\text{\LaTeX}$  objects.

See the help pages for `xtable` and `print.xtable` to find out all of the possible arguments. If one does intend to have the output go directly into the document, without any hand editing, it is almost certainly necessary to specify a large number of arguments.

It may be more workable to write the  $\text{\LaTeX}$  file on disk and then double-check its contents before manual inclusion in the document. If a  $\text{\LaTeX}$  table file has been created from another document, we do not recommend “cutting and pasting” into this document. Instead, use “`\input{filename}`”.

The following code chunk will save the same  $\text{\LaTeX}$  markup table in a file.

Table 1: Ten Lines from One Data Frame

| x     | y     |
|-------|-------|
| -1.21 | 0.41  |
| 0.28  | -0.47 |
| 1.08  | 0.07  |
| -2.35 | -0.50 |
| 0.43  | -0.83 |
| 0.51  | 0.17  |
| -0.57 | -0.90 |
| -0.55 | 0.17  |
| -0.56 | 0.35  |
| -0.89 | -0.05 |

## 7 Session Info

Reports do not include the R session replication information, generally speaking. However, compiling the document will produce a record-keeping file in which the session information is saved. This will be in the current working directory.