Introduction to R Markdown

TMA4268 Statistical Learning V2020

Stefanie Muff, Department of Mathematical Sciences, NTNU

February 3, 2019

Thanks to Mette Langaas for the original version of this introduction.

Aim

We will use R Markdown for our compulsory exercises in this course - you therefore need to know

- what is R Markdown?
- what is an YAML header?
- how to render the document using `knit`.
- how to write text and maths.
- how to include R commands.
- how to handle error message when you `knit`.
- how to hand in compulsory exercise 1.

What is R Markdown?

1 minute introduction video

R Markdown structure

YAML Header

- Where: at start of file, between lines of - - -
- Warning: indentation important here.

Hands-on: We look at the template for Compulsory exercise 1, available from the course website.

- Download and open in RStudio, save with a different file name. Important: **The file name should start with CompEx1**.
- Change the name of the group and the name of the group members
- Press the `knit` button and observe that happens.

In the future: If you want to make a new document in RStudio: "File New R Markdown"

Remember: for Compulsory exercise 1 we have list all packages you need to install on the top of the assignment. Smart of you to start by installing all of these!

YAML output options

output value	creates
html_document	html
pdf_document	pdf requires Tex
word_document	Microsoft Word (.docx)
odt_document	OpenDocument Text
rtf_document	Rich Text Format
md_document	Markdown
github_document	Github compatible markdown
ioslides_presentation	ioslides HTML slides
slidy_presentation	slidy HTML slides
beamer_presentation	Beamer pdf slides (requires Tex)

See also list of sub-options in cheat sheet. https://github.com/rstudio/cheatsheets/raw/master/rmarkdown-2.0.pdf

How to render a document?

You do this by pressing `knit`.
Knitting is also done by: Ctrl+Shift+K (Windows) or Cmd+Shift+K (MacOS).

1. Creating documents with R Markdown starts with an .Rmd file that contains a combination of markdown (content with simple text formatting) and R code chunks.
2. The .Rmd file is fed to `knitr` which executes all of the R code chunks and creates a new markdown (.md) document which includes the R code and it's output.
3. The markdown file generated by `knitr` is then processed by `pandoc` which is responsible for creating a finished web page, PDF, MS Word document, slide show, handout, book, dashboard, package vignette or other format.

- More: About pandoc - the swiss army knife for file conversion
- NB: even if you write tex this is first translated to md and then via pandoc to pdf, so subtle tex stuff may be missed on the way.
- Do you get a separate window popping up, or is your output shown in the Viewer tab of one of the window panes? Go to RStudio-Tools-Global Options-RMarkdown and check what is your value of "show output preview in".

What output type do you want to produce?

- Just keep track of your own work: html_document
- For TMA4268 Compulsory exercise 1: we ask for a pdf-file (because that is easy to read and grade when you upload that to Blackboard)
- To produce a `pdf_document` RStudio (using pandoc) will call a latex-installation, so you need to have latex installed on your laptop to be able to produce a pdf-file.
- Toggle comment/uncomment with hashtag in YAML header `output` to make different options active, then press `knit`. Alterntively this can be done by calling the function `rmarkdown::render()` from your Console window.
- Optional: check that uncommenting `pdf_document` and commenting out `html_document` and pressing `knit` will give you a pdf-file.

- During rendering we use the location of the .Rmd file as the working directory, and the rendering is done in a *new session*.
- Optional: can instead run `rmarkdown::render("file.Rmd","pdf_document")` from command line,
  or`rmarkdown::render("file.Rmd","html_document")`. Then you can have several options in the YAML together!

Formatting your R Markdown file

Text, mathematics

- formatted with markdown
- mathematics (in latex) with formulas staring and ending with one `$` and equation with `$$`
- boldface with two stars and italic with one, new line with two spaces,
- sections, bulleted or numbered lists, tables, footnotes, rulers,

Hands-on: go the the Compulsory exercise 1 template, and just write and press `knitr` to see!

Check how a nice formula using latex is generated for
$Y_i = \beta_0 + \beta_1 x_{i1} + \varepsilon_i$.

Links

- **First option**: The link displayed with the address like here: https://github.com/rstudio/cheatsheets/raw/master/rmarkdown-2.0.pdf
- **Second option**: if you do not want to display the address but some other text name of link as with R Markdown cheat sheet

Code Chunks

- Chunks of embedded code. Each chunk:
- Begins with “‘{r} and ends with “‘
- Set of code chunk options - but I have mainly used these two:
  - `echo`: display the code in the chunk, TRUE or FALSE or selected lines, or maybe with an R-object (later)
  - `eval`: run code in the chunk, TRUE or FALSE
- Remember to include packages to be used within the chunk.
- Chunks can have (unique) names, may help when debugging.

Set-up chunk

- The set-up chunk is a code chunk that you add before you actually start to do the work.
- Smart things to add to the setup-chunk:

```
library(knitr)
knitr::opts_chunk$set(echo = TRUE, tidy = TRUE, message = FALSE,
    strip.white = TRUE, prompt = FALSE, cache = TRUE, size = "sc
    fig.height = 3)
```

Calling R outside of the code chunks

Use the ‘ r before and ‘ after an R command to integrate into the text. For example,

```
2 + 2
```

```
## [1] 4
```

is equal to 4.
This is what we have done in the YAML-header to include todays date on your submission:

```
r format(Sys.time(), '%d %B, %Y')
```

Problems

When I `knit` with output: pdf_document no pdf-file is produced. Why?

- html_document is more forgiving than pdf_document wrt tex-errors
- a tex-error is not easy to spot - log is terrible
- many students have problems here, and some just end up handing in html or Rmd for the projects

My solution

- first I render html_document and look for tex-errors and fix them
- then I render pdf_document, and include `keep_tex: yes` yaml option
- then I compile the tex in my favorite `texshop` and look for sensible log for errors,
- and then go back to the Rmd and fix the error.

Handing in Compulsory exercise 1

- Join a group from Bb (front page - groups)
- then under Compulsory exercises you will see “Hand in” is possible (this will come)
- Upload **both your Rmd and pdf-version of you R Markdown file** with your solutions to the exercise (based on the template)
- Scores and comments will be given on Bb.

References

- Cheat sheet: https://github.com/rstudio/cheatsheets/raw/master/rmarkdown-2.0.pdf
- https://rmarkdown.rstudio.com/lesson-1.html
- https://bookdown.org/yihui/rmarkdown/
- It is also possible to include parameters in the YAML header, and use them in the text/chunks. The parameters can be set in the rendering. More about this in Chapter 15 of the R Markdown book: https://bookdown.org/yihui/rmarkdown/