Introduction to Markdown

Lecture 5

Dirk Eddelbuettel

STAT 430: Data Science Programming Methods (Fall 2019)
Department of Statistics, University of Illinois

Markdown and RMarkdown

Outline

- Basic formatting and usage intro
- Big idea: Markup is independent of rendered output
- **pandoc** as universal converter to/from
- RMarkdown convenience wrapper around pandoc calls
    - eg to JavaScript/HTML for presentations / web content
    - eg to pdf for printable / shareable writeups
- Literate Programming
- Focus *today* mostly on *Markdown*
- We will look more at *RMarkdown* later

So what is Markdown?

Markdown History

- Invented / started by John Gruber in the early 2000s
- Goal: allow "to write using an easy-to-read and easy-to-write plain text format"
- Several implementations, `commonmark` now becoming standard
- The name 'Markdown' is a play on what it is: a mark-up language
- So what is mark-up, and why should I care?

So what is Mark-Up?

Mark-up Languages

- Used by several widely-used publishing and writing systems
- The key is the *separation* of *content* and *formatting*:
    - You declare that you write a header, or paragraphs, or formula
    - Another element defines look of header, paragraph, or formula
    - That element is interchangeable which is key
    - Presentation can change without content change
- If you ever wrote (plain) HTML or LaTeX you used mark-up
- Markdown follows this evolution and some earlier attempts.
- Those did not stick – but Markdown stuck.
- And it is (literally) everywhere now.
- It is fantastically useful and popular.

So how it Markdown used?

pandoc, mostly

- One key tool is pandoc which RStudio conveniently ships with
- pandoc converts both *from* and also *to* markdown syntax
- pandoc can then be used to create
  - web pages in html
  - pdf documents (via LaTeX)
  - Word and Powerpoint documents
- All from the same source (with some caveats)

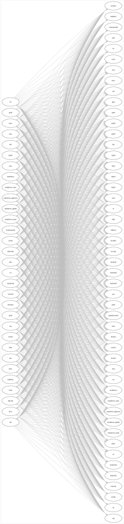
An aside about pandoc

All the different transformations **pandoc** can do …

in one largely illegible chart.

Hint: Go to source page and view in your web browser.

Source: `https://pandoc.org/diagram.jpg` via `https://pandoc.org/index.html`



How Do I use Markdown?

Basic Help: Start in RStudio

- Go to Help -> Markdown Quick Reference
- (Ignore that it already refers to R Markdown)
- The Quick Reference is useful
- Two cheatsheet are also accessible
- Lots of online resources

Getting Started

Formating Basics

Key Syntax Elements (1 of 2)

- Emphasis via `*` and `_` where single use is italic, double bold
    - so `*this*` and `_that_` become *this* and *that*
    - nesting `**_also_**` works ***like this***
- Headers of decreasing levels follow `#`, `##`, `###` and `####`
- (Unorderd) Lists follow simple marker `-`, `+` or `*`
    - Start in first column for top-level
    - Indent four space and use another for nesting like here
- (Numbered) Lists use `1.`. `pandoc` increments counter
- Line breaks can be force via two terminal spaces on line.

Formating Basics

Key Syntax Elements (2 of 2)

- Text quotes start with >
- Links have syntax of `[some text here](url_here)`
- Images are similar: `![alt text](url or path to image)`
- Special syntax for tables (we skip this)
- Code can be typeset in "fenced blocks" following three backticks
- For code, can list the programming language after the backticks

FORMATING BASICS

A simple R Example

::::: {.columns} ::: {.column width=“49%”}

This input

~~~~
```r
a <- 1.23
b <- ”some text”
aSum <- function(x) {
    x + x
}
```
~~~~

turns into this output:

```
a <- 1.23
```

Example

Playground

- Try the site at `https://www.markdowntutorial.com/`
- Immediate feedback: renders as you type
- Nice quick intro and sandbox
- So go and play with it!

Other Markdown Advantages

Some Key Points

- Markdown input is plain text
- So it works well with `git` (and other version control systems)
- Moreoever, GitHub *renders* and display Markdown very well
- The top-level `README.md` of a repository is its landing page
- So go off and create a repo now and write *something* !

Output Formats

- We can create
  - HTML Output for the web
  - Presentations via Javascript
  - Presentations via PDF
  - Write-ups in PDF or Word
  - Now also Powerpoint
- (Most of the time) this requires ‘style files’ and frameworks
- We will not get into specifics of modifying style files
- For general use, RStudio a very good starting point

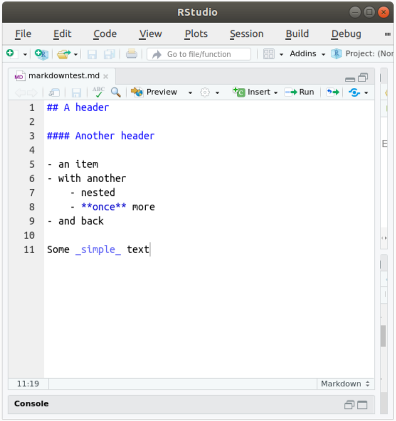
LET'S TRY THIS WITH A SIMPLE EXAMPLE

```
## A header

#### Another header

- an item
- with another
    - nested
    - **once** more
- and back

Some _simple_ text
```

In RStudio open a text file, ie an open / empty document. Add something like the text on the left combining headers, lists, markup.

Save it as e.g. `markdowntest.md`. It is important that you give it an extension `.md`.

LET'S TRY THIS WITH A SIMPLE EXAMPLE



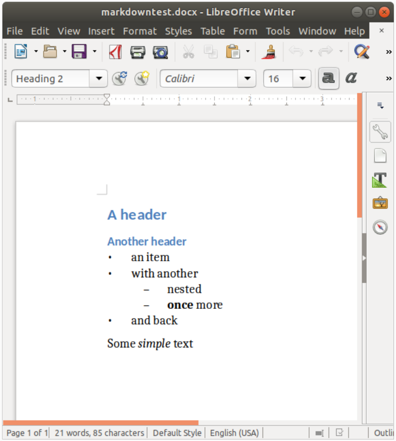
```
## A header

#### Another header

- an item
- with another
    - nested
    - **once** more
- and back

Some _simple_ text
```

As soon as you save as a `.md`, the file is recognised as markdown and highlight. You see structure.

Now click the preview button. A html version will be created and shown in the viewer.

Experiment!

Let's try this with a simple example


markdowntest.docx - LibreOffice Writer
File Edit View Insert Format Styles Table Form Tools Window Help
Heading 2
Calibri
16
Page 1 of 1 | 21 words, 85 characters | Default Style | English (USA) | Outli

A header

Another header

- an item
- with another
  – nested
  – **once** more
- and back

Some *simple* text

This shows the 'Word' version displayed by a Word-alike processor.

Note how structural elements like the headers and lists are correctly preserved.

A Second example

Markdown and RMarkdown

- The previous example showed how our selection of an output format was reflected in the header of the document.
- That header is called YAML header, YAML is the name of the `key: value` scheme it uses
- One particular option is worth exploring: `theme`
- This allows to easily select different “output themes”
- It illustrates how content and formatting are independent.

A Second example

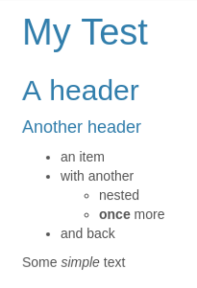
Markdown and RMarkdown

- Create an object with header such as the one below
- Vary the ‘theme’ value among the possible values, i.e. “default”, “cerulean”, “journal”, “flatly”, “readable”, “spacelab”, “united”, “cosmo”, “lumen”, “paper”, “sandstone”, “simplex”, or “yeti”

```
---
title: ”My Test”
output:
  html_document:
    theme: default
---
```

A Second example

Cerulean



My Test

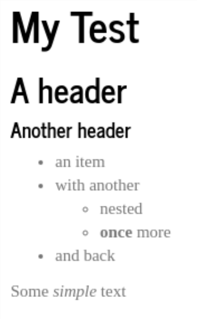
A header

Another header

- an item
- with another
  - nested
  - **once** more
- and back

Some *simple* text

Journal



My Test

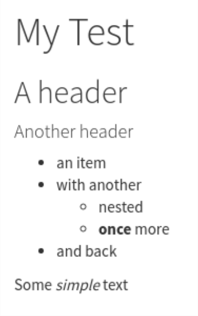
A header

Another header

- an item
- with another
  - nested
  - **once** more
- and back

Some *simple* text

Cosmo



My Test

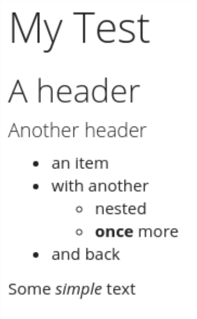
A header

Another header

- an item
- with another
  - nested
  - **once** more
- and back

Some *simple* text

Yeti



My Test

A header

Another header

- an item
- with another
  - nested
  - **once** more
- and back

Some *simple* text

(The difference in size is due to how I created the screenshots.)

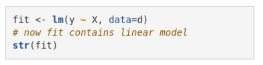
Other styles, which we may see later, vary background colors, fonts and more.

A third example: Code

Code

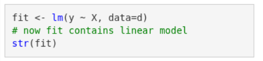
- Typesetting code is particularly easy thanks to `pandoc`
- It is also themable with the option `highlight`
- Three choices are shown below

Tango



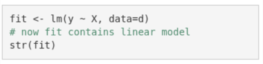
```
fit <- lm(y ~ X, data=d)
# now fit contains linear model
str(fit)
```

Haddock



```
fit <- lm(y ~ X, data=d)
# now fit contains linear model
str(fit)
```

Textmate



```
fit <- lm(y ~ X, data=d)
# now fit contains linear model
str(fit)
```

More Usage

Presentations, Websites

- Many frameworks exists for web-based (“Javascript”) presentations, some are accessible from RStudio
- There is also a rich choice for pdf presentations (such as these)
- Entire websites and blogs can now be written via markdown and additional tooling
- We may get to a bit more of this in the rmarkdown lecture
- There is more such as support for Microsoft Word and Powerpoint (which I have not used)

Summary

Markdown

- Very powerful, very readable, very easy to write
- Many website now default to it: GitHub, Stackoverflow, …
- Excellent support at GitHub and from RStudio
- Try it and feel free to experiment!