

Data-based Statistical Decision Model

Lab 1 (Part 2) R Markdown

Sungkyu Jung

1. Introduction

R Markdown provides an unified authoring framework for data science, combining your code, its results, and your prose commentary. R Markdown documents are fully **reproducible** and support dozens of output formats, like PDFs, Word files, slideshows, and more.

R Markdown files are designed to be used in three ways:

1. For communicating to decision makers, who want to focus on the conclusions, not the code behind the analysis.
2. For collaborating with other data scientists (including future you!), who are interested in both your conclusions, and how you reached them (i.e. the code).
3. As an environment in which to do data science, as a modern day lab notebook where you can capture not only what you did, but also what you were thinking.

Reproducible analysis with R Markdown

The concept of literate programming was introduced by Knuth decades ago. His advice was:

“Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to human beings what we want a computer to do.”

Central to this prescription is the idea that the relevant documentation for the code—which is understandable not just to the programmer, but to other human beings as well—occurs alongside the code itself. In data analysis, this is manifest as the need to have three kinds of things in one document: the code, the results of that code, and the written analysis. We belong to a growing group of people who find the `rmarkdown` and `knitr` packages to be an environment that is ideally suited to support a reproducible analysis workflow.

Watch this Prezi presentation by Andrew Bray: <https://prezi.com/rtmytpb6iws/reproducible-20/>
(<https://prezi.com/rtmytpb6iws/reproducible-20/>)

Getting help

R Markdown integrates a number of R packages, including the `rmarkdown` and `knitr`, and external tools. This means that help is, by-and-large, not available through `?`. Instead, as you use R Markdown in the future, keep these resources close to hand:

- R Markdown Cheat Sheet: *Help > Cheatsheets > R Markdown Cheat Sheet*,
- R Markdown Reference Guide: *Help > Cheatsheets > R Markdown Reference Guide*.

Both cheatsheets are also available at <http://rstudio.com/cheatsheets> (<http://rstudio.com/cheatsheets>).

2. R Markdown basics

We begin by creating a new R Markdown document. Within RStudio, a new template R Markdown file can be generated by selecting R Markdown from the New File option on the File menu. This generates the dialog box. The default output format is HTML, but other options (PDF or Microsoft Word) are available. The formatted output can be generated and displayed by clicking the `Knit HTML` button in RStudio, or by using the commands in the following code block, which can also be used when running R without the benefit of RStudio.

```
library(rmarkdown)
render("filename.Rmd")      # creates filename.html
browseURL("filename.html")
```

When you *knit* the document, R Markdown sends the .Rmd file to knitr, <http://yihui.name/knitr/> (<http://yihui.name/knitr/>), which executes all of the code chunks and creates a new markdown (.md) document which includes the code and its output. The markdown file generated by knitr is then processed by *pandoc*, <http://pandoc.org/> (<http://pandoc.org/>), which is responsible for creating the finished file. The advantage of this two step workflow is that you can create a very wide range of output formats, as you'll learn about in R markdown formats.

Exercises

1. Create a new R Markdown document using File > New File > R Markdown. Read the instructions. Knit it by clicking the appropriate button.
2. Modify the code by inserting `head(cars)` and `plot(cars)`. See modified output.
3. Create a new R notebook using File > New File > R Notebook. Read the instructions. Practice *running the chunks*. Verify that you can modify the code, re-run it, and see modified output. (What are the similarity and the difference between the R notebook and the R Markdown, knit to HTML?)
4. Create one new R Markdown document for each of the three built-in formats: HTML, PDF and Word. Knit each of the three documents. How does the output differ? How does the input differ? (You may need to install LaTeX in order to build the PDF output - RStudio will prompt you if this is necessary.)

3. Text formatting with Markdown

Prose in .Rmd files is written in Markdown, a lightweight set of conventions for formatting plain text files. Markdown is designed to be easy to read and easy to write. It is also very easy to learn. The guide below shows how to use Pandoc's Markdown, a slightly extended version of Markdown that R Markdown understands.

Text formatting

```
*italic*   or _italic_
**bold**   __bold__
`code`
superscript^2^ and subscript~2~
```

Headings

```
# 1st Level Header

## 2nd Level Header

### 3rd Level Header
```

Lists

```
* Bulleted list item 1

* Item 2
  * Item 2a
  * Item 2b

1. Numbered list item 1

1. Item 2. The numbers are incremented automatically in the output.
```

Links and images

```
<http://example.com>

[linked phrase](http://example.com)

![optional caption text](path/to/img.png)
```

Tables

```
First Header | Second Header
-----|-----
Content Cell | Content Cell
Content Cell | Content Cell
```

Exercises

1. Practice what you've learned by creating a report for all exercises below. Your .Rmd file must have the filename "ID-HW1" (replace ID with your unique identifier). The title should be `Rmarkdown exercise`. The author should be your name. The first exercise is to include a bulleted list of *statistics* courses you have taken before, and the semester the course was offered. (Include at most three courses.) Highlight the semester-year in bold.

2. In the document, include any image, available through a URL, that is of interest to you. Use “``” command.
3. Render your .Rmd file to an html document.

4. Code chunks

To run code inside an R Markdown document, you need to insert a chunk. There are three ways to do so:

1. The keyboard shortcut Cmd/Ctrl + Alt + I
2. The “Insert” button icon in the editor toolbar.
3. By manually typing the chunk delimiters ````{r}` and `````.

R code chunks can be used as a means render R output into documents or to simply display code for illustration. Here is a simple R code chunk that will result in both the code and it's output being included:

```
```{r}
summary(cars)
```
```

To display the output of a code chunk but not the underlying R code, you specify the `echo=FALSE` option:

```
```{r, echo=FALSE}
summary(cars)
```
```

Note that R code chunks can also be used to render plots. To display a plot while omitting the code used to generate the plot you'd do this:

```
```{r, echo=FALSE}
plot(cars)
```
```

To display R code without evaluating it, you specify the `eval=FALSE` chunk option:

```
```{r, eval=FALSE}
summary(cars)
```
```

As seen above, chunk output can be customised with options, arguments supplied to chunk header. A full list of chunk options is available at <http://yihui.name/knitr/options/> (<http://yihui.name/knitr/options/>).

4. More Tweets, More Votes.

We will now reproduce the analysis done for the paper “More Tweets, More Votes.” Published version is available at <http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0079449> (<http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0079449>). The authors supplied the R file and the data file for reproduction of their analysis, at <https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/23103> (<https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/23103>). Download the R file and the data file. Include the following in your “ID-HW1.Rmd” file:

1. a code chunk that read the data, and perform the simple linear regression `vote_share ~ mshare` (that is, `predict vote_share from mshare`)

2. a code chunk that plot the scatter plot overlaid with the fitted regression line.

You must display both the code chunks and results. Do not include any unnecessary R code lines. Be sure to add a couple of sentence explaining your analysis.

5. Using 한글 in Rmarkdown

Try one of the tricks below:

- http://r-bong.blogspot.com/2016/03/rstudio_26.html (http://r-bong.blogspot.com/2016/03/rstudio_26.html)
- <http://www.epistemology.pe.kr/2016/08/04/1067?ckattempt=1>
(<http://www.epistemology.pe.kr/2016/08/04/1067?ckattempt=1>)
- <https://github.com/jaimyoung/data-science-in-korean/blob/master/r-markdown-korean.md>
(<https://github.com/jaimyoung/data-science-in-korean/blob/master/r-markdown-korean.md>)