Rmarkdown - Tutorial

Michael T. Hallworth

Wednesday, February 11, 2015

What is a Markdown file ?

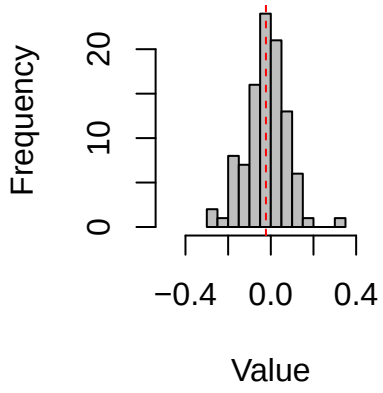
Rmarkdown is a tool to generate reports, share results, and create stand alone R code and results using html, pdf or word format.

Why use Markdown ?

Markdown files combine text, R script and R output into a single sharable file without the need for sharing data, R workspaces or R script files. Markdown files are great for generating reports, creating *'tutorials'*, or providing code to accompany manuscripts or assignments. Markdown files are great for the above tasks because they combine text, R script and R output. Combining text with R script and output allows you to explain the analysis and show script and R output all in the same document. Markdown allows text, code and output (including figures) to be interspered throughout the document so readers can see the logical flow of the analysis.

Caution No spell check is available

```
# Generate random variable
x<-rnorm(100,0,0.1) # Generate 100 random samples from a normal distribution
# Plot results
par(mar=c(4,4,0,0))
hist(x,xlim=c(-0.5,0.5),main="",ylab="Frequency",xlab="Value",breaks=15,col="gray")
abline(v=mean(x),col="red",lty=2) # Plot mean with red vertical line
```

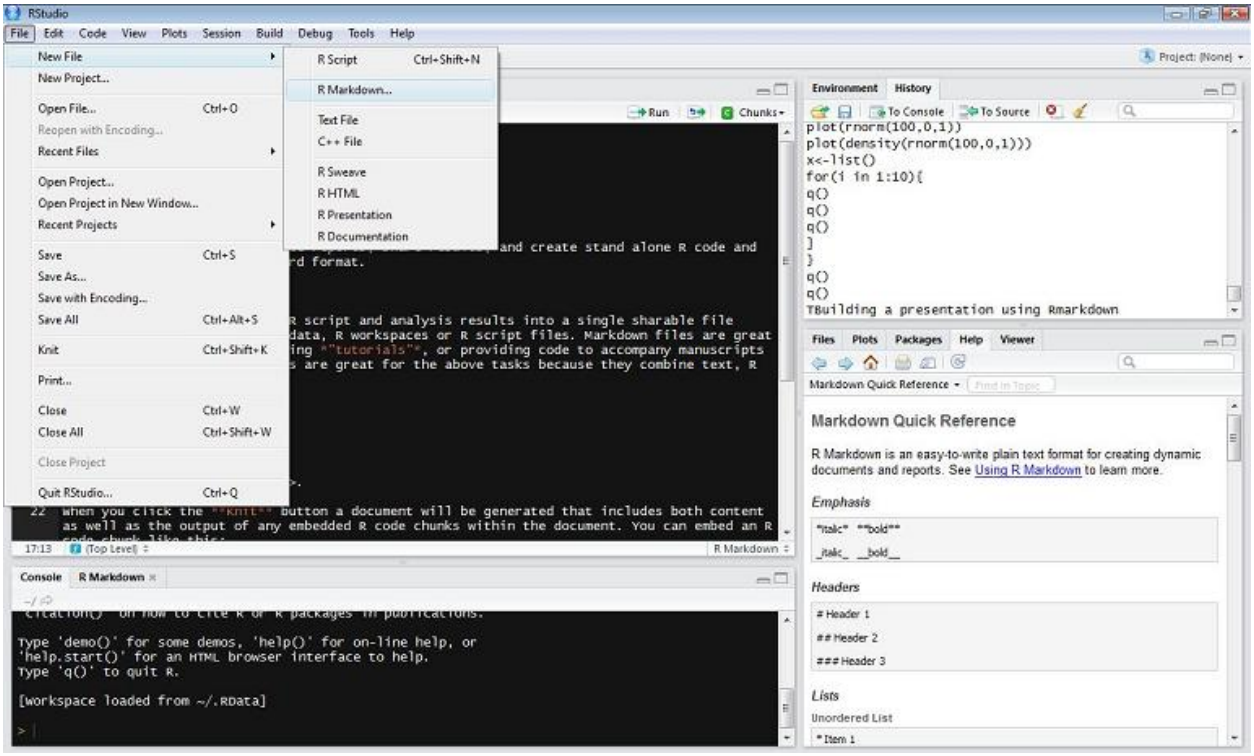


Additional software

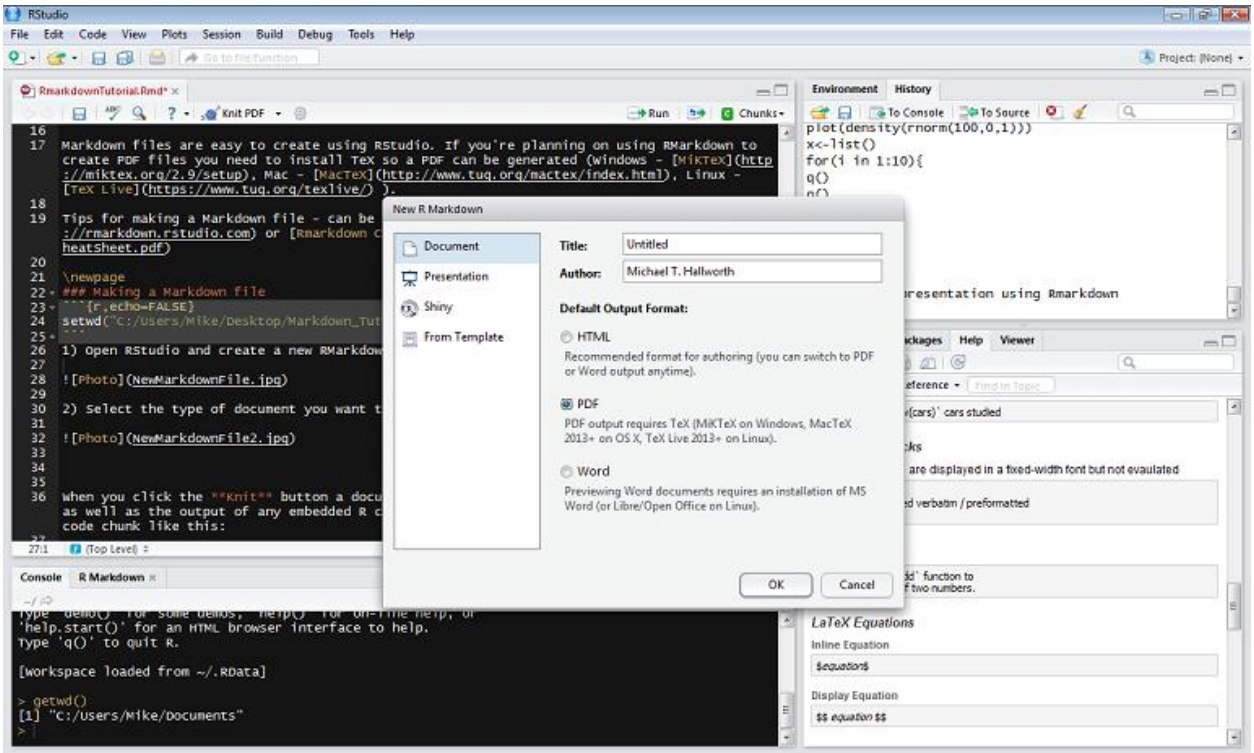
Markdown files are easy to create using RStudio. If you're planning on using RMarkdown to create PDF files you need to install TeX so a PDF can be generated (Windows - MiKTeX, Mac - MacTeX, Linux - TeX Live).

Creating a new Markdown file

1) Open RStudio and create a new RMarkdown file



2) Select the type of document you want to create and provide a name for the document

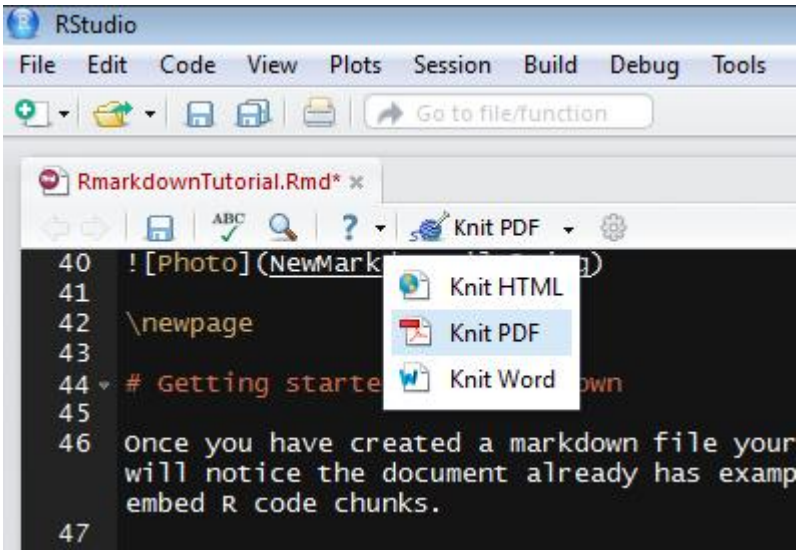


Getting started with Markdown

Once you have created a markdown file your ready to start writing your new document. You will notice the document already has example text in the document which highlights how to embed R code chunks.

Tips for making a Markdown file - can be found using the Markdown Quick Reference or Rmarkdown CheetSheet

To generate your final document or see how the document looks while making it - click the **Knit** button. This will generate a document that includes your text, runs the R code and embeds output into a single html, pdf or Word document (depending on the document type your creating).



Writing embedded R code

starting a new R chunk

Anything that is written in the R chunk will be run in R. For example, if you want to create a comment line you need to use # to comment out any line of script.

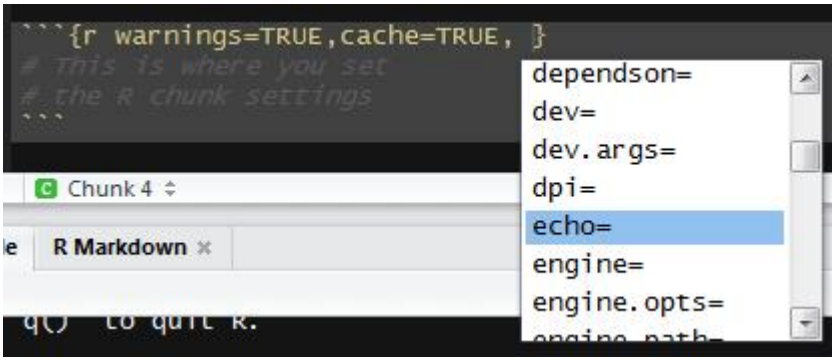
```
# Compute summary statistics of the 100 random values created earlier
summary(x,digits=5)
```

```
##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max.
## -0.293420 -0.080420 -0.023014 -0.022832  0.038829  0.303530
```

Important when you *Knit* your document the R code in the chunks is run. If there are errors in your R code or if you refer to an object or function in a specific library that is not loaded you will receive an error message and the document may not be rendered.

Working with R code chunks

When creating a new R code chunk there are multiple options which can be set for the individual R chunks. The options are set after starting the chunk between the curly brackets (see photo).



Here setting `warning=TRUE` shows any warnings associated with the R chunk for example

```
library(raster)
```

```
## Warning: package 'raster' was built under R version 3.0.3
```

```
## Loading required package: sp
```

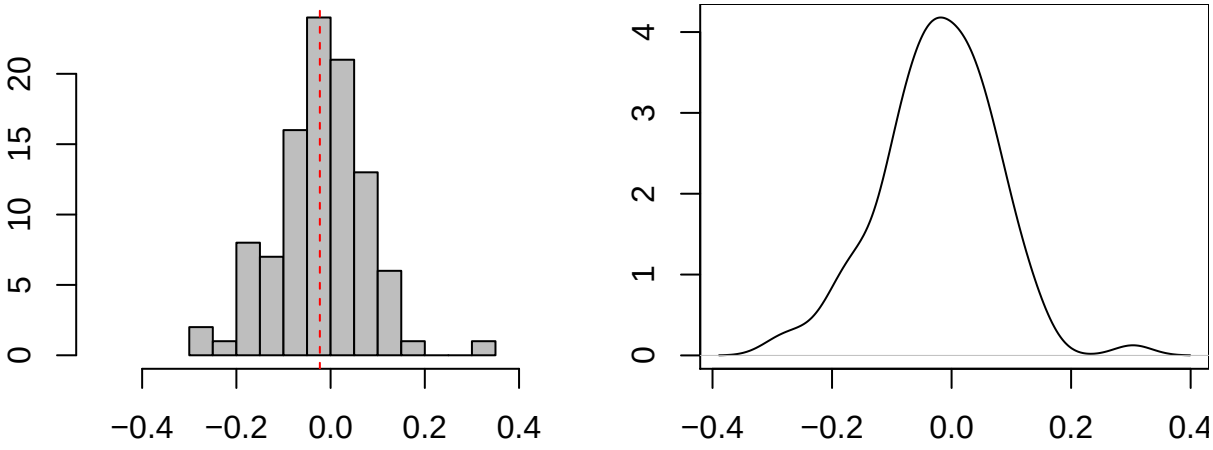
```
## Warning: package 'sp' was built under R version 3.0.3
```

now `warning=FALSE`

```
library(raster)
```

setting `cache=TRUE` is very helpful if running code that takes a while to run. Using `cache=TRUE` stores the results of that R chunk in the console and speeds up subsequent *knits* by not running the R chunk every time you *knit* your document.

setting `echo=FALSE` is another useful setting in the code chunk - especially when plotting results. When `echo=FALSE` the code used to generate a plot is not shown. For example below you see a plot of our random variable x but no code on how the plot was generated.



In line R code

In addition to creating R chunks you can also use R output directly in a sentence using in line R code.

For example We generated 100 random samples from a normal distribution using mean 0 and variance 0.1. The mean of our random sample was -0.0228322 and standard deviation was 0.0964672

```
In line text can be displayed using 'r code'.
The above code was executed using:
We generated 'r length(x)' random samples from a normal
distribution using mean 0 and variance 0.1.
The mean of our random sample was 'r mean(x)' and standard
deviation was 'r sd(x)'
```

Formatting Text

Bold/Italics:

```
**this will be bold** - Bold text
```

this will be bold

```
*this will be italics* - Italics
```

this will be italics

Headers:

```
# First order Header
## Second order
### Third order
```

First order

Second order

Third order

Strikethrough/Superscript:

```
~~strikethrough~~

superscript^2^
```

~~scratch that~~

Volumne in m^3

Page Break:

```
\newpage
```

Additional output

Equation:

```
$y=mx+b$ in line
```

the equation for a linear model is $y = mx + b$

```
$$y=mx+b$$ display equation (displayed in center)
```

We used the following equation for a linear model:

$$y = mx + b$$

Tables:

```
First Header | Second Header
-------------|-------------
cell 1 content | cell 2 content
```

X	Y
10	7
11	8
15	10

Hyperlinks:

```
[phrase](http://scholar.google.com/)
```

Google Scholar

Photos:

```
![Photo](path/to/photo)
```