

# STAT 801- R Markdown

Ved Piyush

8/31/2020

## R Markdown

R Markdown is a tool that is integrated within R studio. It can be used to create efficient reports by making it seamless to integrate R code, visualizations, and user commentary. For example, you can use R Markdown for your projects and homework in this class. The benefit from it is that you wouldn't have to copy and paste code and output into Microsoft Word, because everything can be integrated in one simple code file which will produce a nice PDF document or Word file. Having your entire work done within R Markdown also ensures reproducibility of your work, which is a very important concept if you are conducting a quantitative analyses. R Markdown files have a .rmd extension and these are simply plain ascii text files that can be viewed in notepad or similar editors.

## Installation

Please follow the link <https://bookdown.org/yihui/rmarkdown/installation.html> for installation of R Markdown.

## Elements of R Markdown

There are three main elements of any R markdown file:

1. R code that is run by the knitr package
2. Text for the report
3. Metadata

We will go over each of these elements.

## Code

R Markdown code contains the R code that you will use for your analysis. We will consider a simple example of incorporating R code in your report. For this please refer to **page number 4 of NumericaSummary1.docx** from the STAT 801 website. This example has R code for finding the mean and median of a vector of four numbers. R code should be included within an 'R chunk'. You can get a template structure for the chunk by pressing **Ctrl + Alt + I**. You can also get it by going to Insert and selecting R or by simply copying/pasting previous chunk code. This structure is as follows:

```

``{r}
# vector for which we have to find the mean and median
y <- c(2.9, 3.4, 3.6, 2.8)
# find the mean
mean(y)
# find the median
median(y)
``

```

The following is an example of an R code chunk.

```

> # vector for which we have to find the mean and median
> y <- c(2.9, 3.4, 3.6, 2.8)
> # find the mean
> mean(y)

```

```
[1] 3.175
```

```

> # find the median
> median(y)

```

```
[1] 3.15
```

The report can be automated by using inline R code. Suppose you want to state the mean and the median calculated above, but you do not want the normal output. You can use inline code for this specific purpose. For example, the mean is 3.175 and the median is 3.15 (see code in the .rmd file).

**Code Chunk Options** There are options that you can use within chunks depending upon different scenarios. For example, consider the code chunk in the example above. Suppose we want to keep the code chunk and the objects inside it but don't want to show either the code or any results from it. For this you can use the option `include = FALSE`. If you want to hide the code but want to display the output, you can use the option `echo = FALSE`.

```
[1] 3.175
```

```
[1] 3.15
```

A list of options is available at <https://yihui.org/knitr/options>. Chris has found the following options with the `knitr` package function `opts_chunk()` tend to make the output look good. You can insert them in the first chunk.

```

> # Removes a leading '+' from being displayed when an R
> # command extends on to the next line
> options(continue = " ")
> # Options to control knitr settings
> opts_chunk$set(prompt = TRUE, comment = NA, background = "white",
  tidy = TRUE, tidy.opts = list(width.cutoff = 60, blank = FALSE))

```

## Text for the report

The text portion of R Markdown is whatever you want to explain in your report. There is no specific requirements for the text portion as there were for the chunks or the inline R code. You can use headers which use the `#` character (this is different from a “comment” within an R program). Header with only one `#` will be the top level header. If you want to nest other headers within the top level headers you have to use `##` character. For example consider page number 6 of `NumericaSummary1.docx`. `Measures of variation` is the main level header, `Range`, and `Example` are the second level headers within the main level header. We can structure these headers as follows:

**Measures of variation** Notice it has 4 # characters because we want to put it within the header *Text for the report* which has 3 # characters. Similarly *Range*, and *Example* have 5 # characters because we want to put those headers within the *Measures of variation* header which has 4 # characters.

**Range** maximum data value - minimum data value

**Example** Suppose the data consists of 100, 10, 11, and 9. If you want to write a mathematical formula, you have to enclose it within \$ symbols. For example,  $Range = 100 - 9 = 91$

**Mathematical symbols** Mathematical symbols can be rendered using the same syntax as within LaTeX (this is a free typesetting package). If you do not have direct experience with LaTeX already, you may have used this syntax when creating equations in Word's equation editor. Consider when we calculated the sample mean in the previous example. The sample mean has a designated symbol  $\bar{x}$ . Therefore, if you want to nicely format your result, you can use `\bar{x}` combined with \$ symbols and write  $\bar{x} = 3.175$ . Similarly you can create many other symbols such as  $\sigma$ ,  $\sigma^2$ , and fractions such as the formula for page number 7 of `NumericaSummary1.docx` can be written as  $\sigma^2 = \frac{\sum_{i=1}^N (y_i - \mu)^2}{N}$ . If you write an equation within single \$ symbols, then that equation will appear inline with other text. If you want to display an equation in the center of the page, then you have to enclose it in double \$\$ symbols. For example, the previous equation when enclosed within double \$\$ symbols will look as below.

$$\sigma^2 = \frac{\sum_{i=1}^N (y_i - \mu)^2}{N}$$

**Text Formatting options** You can also add many more formatting options, such as adding links, images etc. For example, suppose we want to add links in our report. There is a very detailed online companion on R markdown by R Studio which can be accessed using this link <https://rmarkdown.rstudio.com/lesson-1.html>. There are many other formatting options as well, please see the R Markdown cheatsheet at [https://rstudio.com/wp-content/uploads/2016/03/rmarkdown-cheatsheet-2.0.pdf?\\_ga=2.75624853.1071750233.1597632777-986877313.1597632777](https://rstudio.com/wp-content/uploads/2016/03/rmarkdown-cheatsheet-2.0.pdf?_ga=2.75624853.1071750233.1597632777-986877313.1597632777). For a complete list of formatting options, see page number 2 under the heading "Pandoc's Markdown". A reference for writing code for equations can be found at <http://www.stat.cmu.edu/~cshalizi/rmarkdown/> under the heading "Math in R Markdown". An intro to LaTeX can be found on Chris Bilder's website at <http://www.chrisbilder.com/stat850/sections.html> (see Section 2 and course videos on the schedule web page).

**Tables** Tables can be put in a markdown document using the `kable` function in the `knitr` package. The file `Lincoln_Feb_wind.csv` is a file in section 2 of the section materials web page. If we want to read in the csv file and display the first few lines of this dataset as a table in the markdown document, we can do it as follows.

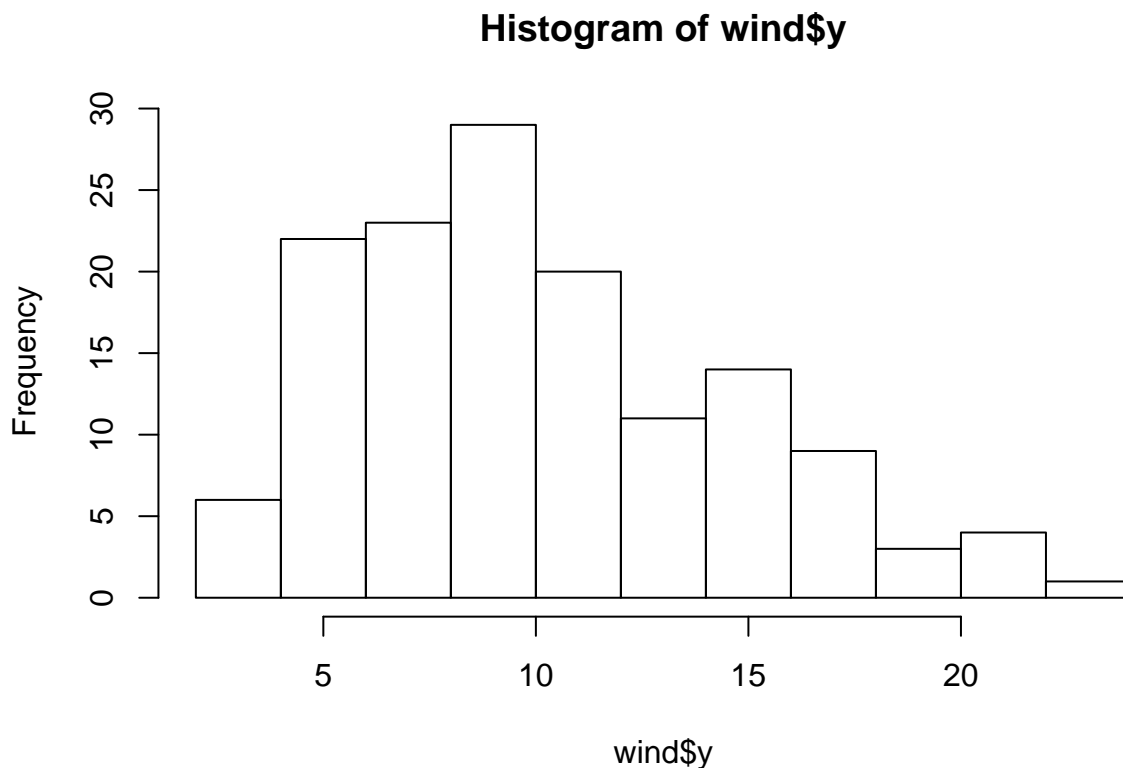
```
> # read in the csv file
> wind <- read.csv("Lincoln_Feb_wind.csv")
> # use the kable function
> kable(head(wind), caption = "A knitr kable.")
```

Table 1: A knitr kable.

Year	Day	y
1	1	9.4
1	2	12.7
1	3	3.9
1	4	9.8

Year	Day	y
1	5	9.5
1	6	15.0

**Embedding visualizations** Embedding visualizations in a markdown file is similar to embedding the R code. You need to embed the code generating the visualization in a chunk. For example, here is the histogram on page number 3 of `GraphicalSummary-2.docx`.



## Metadata

Metadata is what we put at the very beginning of the R markdown file. It is used to mention things such as title of the document, date, and author. Please refer to the very beginning of the R Markdown code for the usage of these metadata fields.

## Alternatives to R Markdown

There are a few alternatives to using R Markdown that provide similar functionality. For example, LyX is a free software package that allows one to use a Word-like interface but still embed R code within the document. Introductions to LyX are available on Chris Bilder's STAT 850 website at <http://www.chrisbilder.com/stat850/sections.html> (see Section 2 and course videos on the schedule web page). In fact, all of Chris's R notes for that course were created using the combination of LyX and R and are available for download. An introduction to including R within a LyX document is available in the `knitr` portion of the LyX section.

A second alternative is StatTag that was recently created by researchers at Northwestern University. This is a free plug-in for Word that allows for similar functionality as R Markdown. Please see <https://sites.northwestern.edu/stattag> if you are interested in using it.

## Creating documents with R Markdown

You can create HTML, Word, and PDF documents with R Markdown. From the dropdown of the knit option, you can select “Knit to HTML”, “Knit to Word” or “Knit to PDF”. When you use R Markdown to prepare documents in this class, you have to either submit the Word or the PDF file, not the HTML file. You might find it easier to focus on the Word documents because `TinyTeX` package is needed for PDF.

## Use of R Markdown is optional

The use of R Markdown or similar alternatives is optional for students in STAT 801. We wanted to present these options to you because they can make your work easier to complete in this class. Also, trends in the creation of reports/papers that use R are going in the direction of using “dynamic document creation” tools like these. We understand though there is a start-up time cost associated with learning these tools. We recommend that you at least try these tools, especially if you find the material at the beginning of the course to be review.