Getting Started

Nicole Tresvalles

8/19/2021

Lab documents will be written in R Markdown. This is an R Markdown document. There are three components of this document that we will pay attention to in this file:

1. the YAML header at the top of the document
2. the Markdown syntax used throughout the document
3. the executable code blocks interspersed throughout this document

YAML Header

R Markdown documents begin with a YAML header. This is the text that you see between the '—' (lines 1-6 of this document). This header includes a number of key-value pairs that provide metadata about the document (such as its title, who created it, when it was last updated, and the format in which it should be published). There are certain keys that appear by default in R Markdown files (e.g. title, author, date, output). Depending on the output format you specify, there may be additional options (such as whether your document should have a table of contents and how figures should be sized by default).

> Checkpoint 1: Add your name as the author of this document by editing the YAML header where it says to ADD YOUR NAME. Be sure to keep the quotation marks around your name.
>
> After successfully completing checkpoint 1, commit your code (see instructions in Readme).

For this research, we will sometimes publish Markdown files as HTML documents (web documents) and sometimes as PDF documents. To compile this .Rmd file as one of these documents, we need to 'knit' the file. You can knit this document, but clicking the 'Knit' button in the taskbar above. This document is currently set to output as an HTML file. We know this because in the YAML header, the 'output' key is followed by the value 'html_document'.

> Checkpoint 2: Click 'Knit' in the taskbar above. A new window will open with this document formatted as an HTML document. Change the output from html_document to pdf_document. Click 'Knit' again. Notice how the output changes.
>
> After successfully completing checkpoint 2, commit your code again.

Markdown

Markdown is a mark-up language - or a computer language that we can use to communicate to a digital system how we want text and images displayed when it is published. When you type text into software like Microsoft Word, you can change the size and font, add bullets, and insert images by clicking on buttons in

the text editor, and the changes are immediately visible. Markdown is different. To format and style text in Markdown, we use certain designated symbols and syntax.

For instance, let's say we want to create a heading in our document. To do this we would add a '#' and a space in front of the text of our header like this: (note that Heading One will be the largest text, and subsequent headings will be increasingly smaller)

Heading One

Heading Two

Heading Three

Heading Four

Heading Five Heading Six

If we want to add bullets to an R Markdown file, we can add asterisks (*) and a space in front of the text we want bulletted:

- Bullet one
- Bullet two
- Bullet three

If we want to add a hyper-link to an R Markdown file, we can place the text we want hyper-linked in brackets '[]' and the link in parentheses '()' immediately following the bracketed text:

This is a link to our GitHub organization.

Section 3 of this Cheatsheet provides a more comprehensive list of syntax rules for R Markdown.

> Checkpoint 3: Format the capitalized text immediately below this set of instructions as **bold** text. Then knit the document to check how the text gets displayed in your published document.
>
> After successfully completing checkpoint 3, commit your code again.

Make **THIS TEXT** bold.

Code chunks

R Markdown documents can include chunks of code and their output. R code chunks are designated by starting a line with '`{r}' and ending a line with '`'. See below.

```
sum <- 2 + 2

sum
```

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

RStudio makes it easy for you to add a code chunk to a document by clicking the green button +C button in the taskbar and clicking R.

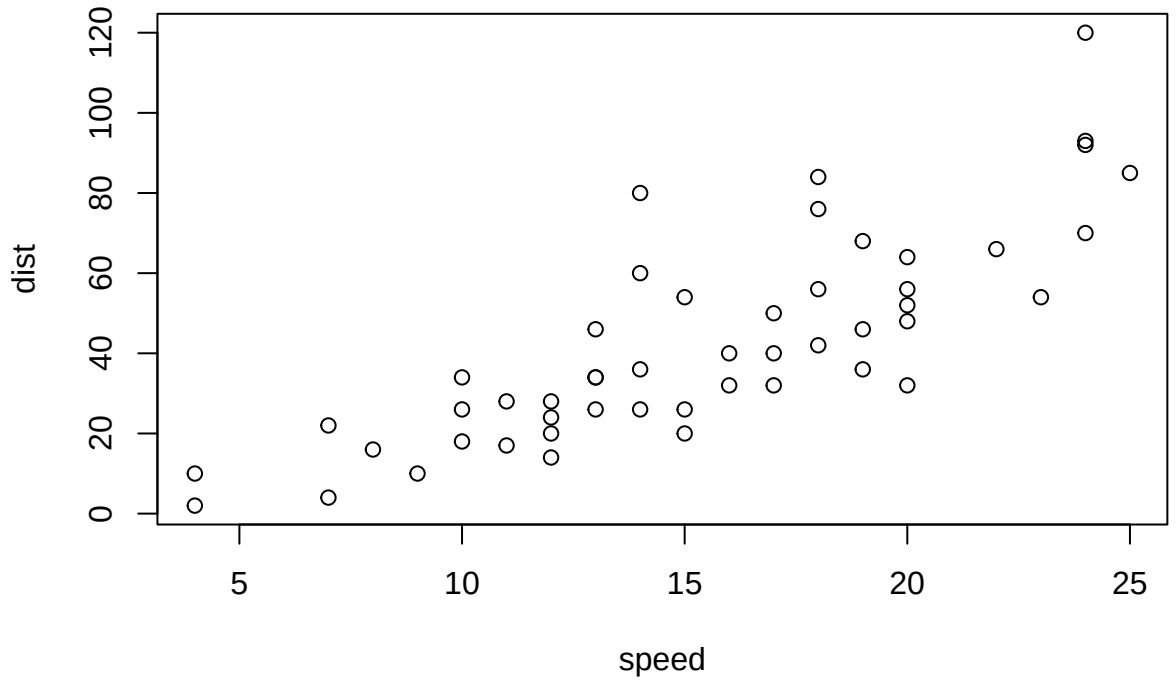
By default when you publish a document with a code chunk, both the code block and the output of the code will be displayed in the document. If you add 'echo = FALSE' between the leading curly brackets of the code chunk (following 'r'), only the output of the code will be displayed. The code that created that output will be hidden.

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

Section 5 of this Cheatsheet lists other options we can set for code chunks.

> Checkpoint 4: Add a code chunk below. Copy and paste the following code and place it in the code chunk: plot(cars) Referencing the linked cheatsheet, set the figure width to 3 and the figure height to 3. (You will need to separate these options with a comma.) 'Knit' the document to see how your plot renders in the document.

```
plot(cars)
```




```
fig.width= 3
fig.height=3
```

> Stage, commit, and push your code following instructions on GitHub.