Template **solutions** using the automated approach via Make

Clement Lee

2021-03-17

Instructions

1. Edit this template as necessary.
2. Either go to the terminal tab within RStudio, or open a terminal if you are on Unix-like platforms.
3. Set the current directory to where this file is located, if this has not been done yet.
4. Type `make automated` in the terminal to generate all four documents in one go:
    i. Questions in pdf;
    ii. Questions in html;
    iii. Solutions in pdf;
    iv. Solutions in html.

Question 0

What are the main differences between this template and the "manual" template?

Solution:

1. If you compare the first `R` code chunk i.e. the code between the pair of lines with triple backticks, the following line is not in the source code of this automated template:

```
knitr::opts_chunk$set(include = TRUE) # TRY TOGGLING BETWEEN TRUE & FALSE
```

1. (cont'd) This is because the option `include` is passed from the makefile, instead of being set manually.
2. While you can still generate a pdf or html manually by clicking the ***Knit*** button in RStudio, you will only get the solutions as `include` is default to `TRUE` if it's not supplied.
3. This way of parametrising the report also allows the title to be changed according to the `include` option. See the line beginning with `title:` at the top of the source code.

Question 1

TYPE YOUR QUESTION HERE.

Solution:

TYPE YOUR SOLUTION HERE. SEE THE NEXT Q&A FOR EXPLANATION.

Question 2

(This is not really a question but an explanation - see above.) You type the "normal" text, usually the question, in a line without any "environment" i.e. there are no backticks, quotes, or hashtags etc. before or after the line. If you want `R` to evaluate something inline, type

1. a single backtick, then
2. the character "r" followed by a space, then
3. the expression in `R` you want to evaulate, and finally
4. another single backtick.

See how it is done in the source code: $1 + 2 = 3$. You will also see that maths being typeset between a pair of single dollar signs are allowed.

Solution:

You should only see this line in the generated solutions, pdf or html, but not in the generated questions. This is because we harness the power of the inline evaluation using a pair of backticks, and the object `is_answer` we have defined in the source code. This is why the text typed after `r if (is_answer) {"` and before `"}` in the source code will only be visible in the solution.

Question 3

This is a "pure" maths question, with no statistical or data analysis in `R` required. Prove that

$$(a + b)^2 = a^2 + 2ab + b^2.$$

(In the source code you will also see that maths typeset between a pair of double dollar signs are allowed. Other LaTeX environments allowed are `align`, `equation`, `eqnarray`, their no-numbering versions i.e. with asterisks, and \[\].)

Solution:

The solution is

$$
\begin{aligned}
(a + b)^2 &= (a + b)(a + b) \\
&= a^2 + ab + ba + b^2 \\
&= a^2 + 2ab + b^2
\end{aligned}
$$

You can see that maths environments also work within the solution. However, any backslash in LaTeX has to be doubled because there is an additional layer of evaluated. The solution content is evaluated by `R` first before being rendered by Markdown. This also applies to other symbols and commands e.g. α and **1** (see source code).
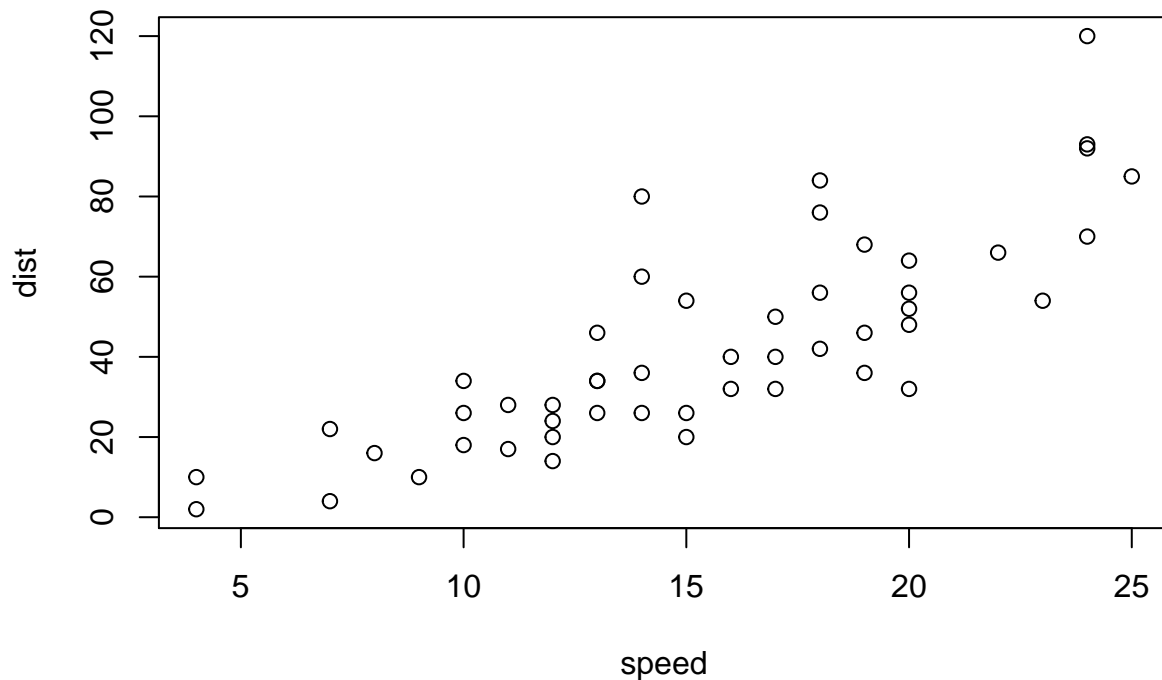
Question 4

This is a data analysis question. Plot the `cars` data set that is available in `R`.

Solution:

As previous questions, we can type here any solution text unrelated to data analysis in `R`. For the data analysis, both the `R` code and the output (which is a plot here) can done within the code chunk i.e. anything sandwiched by two lines of triple backticks, the first one which also has `{r}`. The cars data set is

```r
plot(cars)
```

Remember the plot (or in fact other kinds of output) doesn't need to be evalulated and saved manually prior to the generation of the pdf or html, as long as they are proper `R` commands.

Question 5

What if I want to evaluate something in `R` but not print the output? Or what if I want to show the code but not evaluate it? More generally, what if I want to customise the printing of some/all of the `R` code, numerical output, and graphical output?

Solution:

All evaluations and output printing of `R` code chunks can be customised. A whole list of options can be found at https://yihui.org/knitr/options/.

Question 6

What do I need to be aware of when modifying this template?

Solution:

1. The source code contains at least four types of syntax: Markdown, LaTeX, `R` inline, and `R` code chunk. Make sure you are typing the correct syntax in the correct environment / language.
   - For example, a line starting with a hashtag # and a space represents a section heading in Markdown, but becomes a comment line in an `R` code chunk. Also, see the solution to Question 4 on typing LaTeX commands within a solution.
2. Also, there is a trade-off between the flexibility of generating pdf & html using one source code file and the full capabilities within LaTeX.
   - For example, self-defined LaTeX commands may work for the pdf but not for the html.
   - Another aspect that may not function fully is the use of environments - see Q3.
   - Also mentioned in Q3 is the use of double backslashes, instead of single ones, within solution text.

Question 7

What if I want to generate another set of {questions, answers} × {pdf, html} using this automated approach?

1. You will need to create another Rmd of source code, then add a few lines in the makefile, similar to the ones for this file.
2. As an example, we have made a template block in the makefile, beginning with the line `another:    another.Rmd`. This means you need to first create the file 'another.Rmd', then type `make another` in a terminal.