R Markdown Lesson 02: Moving forward

Alexander Strobel & Christoph Scheffel

December, 2021

Title of the thesis

Dissertation

For the degree of
Doctor rerum naturalium
(Dr. rer. nat.)

Presented to the Faculty of Psychology
Technische Universität Dresden
by

Your Name
Born on 9 November 1989

Reviewers:
Prof. Dr. AB
Prof. Dr. CD

Submitted on 6 December 2020

Contents

Outline

In this lesson, you will learn about

- the LaTeX language
- setting up documents in a custom fashion
- formatting text in an advanced way

The LaTeX language

LaTeX, which is pronounced "Lah-tech", is a software for text processing and formulas. It follows the principle *what you mean is what you get* and thus differs fundamentally from common text processing programs. Because of its open source, the world wide availability and especially the remarkable functions for scientific texts, LaTeX becomes more and more standard for scientific work. Because text is only compiled if you click the button, the program will not have problems processing even large documents, like dissertations. However, disadvantages include poorer user-friendliness. Even for simple formatting like *italic* or **bold**, commands must be used. Further, there is no live output of the current code (an issue, however, that is addressed in several current LaTeX engines like Overleaf (https://www.overleaf.com).

LaTeX operates with numerous macros designed to increase user-friendliness. Macros can be used by means of commands and environments. For example the command `\today` pastes the current date and `\chapter{title}` creates a title of a chapter. Larger sections can be formatted with environments. The command `\begin{\*name of environment\*}` initiates a new environment and the command `\end{\*name of environment\*}` quits it. For example, `\begin{center}` initiates an environment were all text is centered. Environments can also be stacked. However, the commands in LaTeX are too extensive to be presented here comprehensively. It is therefore worth a look in the documentation (https://www.latex-project.org/help/documentation/).

The title page and the table of contents above give further examples of LaTeX formatting. Actually, when rendering R Markdown documents, the document is converted to a LaTeX document before it is finally written to a PDF file. As you may have noticed, we stated in the YAML header: "keep_tex: true". That way, the tex file generated from R Markdown will be kept in your working directory, so you can inspect it and learn more about the LaTeX language[1]. Yet, this is an R Markdown document, not a LaTeX document, so we will not elaborate in great lengths on LaTeX. There are simply a few things you want to do that you cannot do in R Markdown, and then it is great that you can resort to LaTeX.

Setting up documents in a custom fashion

One such thing is that you may want or need to format your R Markdown file according to some prespecified template. LaTeX will allow you to do so, just as in the title page example. Another way would be to use so-called cascading style sheets (*.css), and if you are familiar with HTML, you certainly will have come across this issue. The days when one of the authors of this lesson (AS) was writing webpages in HTML date back to the time when download speed was measured in kB/sec, so he forgot almost everything and therefore cannot elaborate on that matter. In the R Markdown Cookbook, you will find some remarks on this issue in section 7.

Sometimes it may also be the case that some journal offers a LaTeX template for formatting manuscripts (e.g., PLoS ONE. You can then (more or less) easily set up your R Markdown document by merging your code with the LaTeX elements specified in this template. OK, admittedly, it will be less easy to do so as a beginner (in fact, when previously writing a manuscript in R Markdown for a submission to PLoS ONE, the

[1] Please note that it is essential that you put "keep_tex: true" on a new line after "pdf_document:" and set another tab ahead of it, otherwise it will not work.

output format was set to "word_document" and then formatted according to PLoS ONE's Word template leaving a bit of frustration that this couldn't be handled more elegantly.)

Formatting text in an advanced way

When it comes to use superscripts like in m^2 or subscripts like in CO_2, the last lesson was already somewhat informative how you could do this in R Markdown language. You can achieve the same result when using LaTeX: here, it would be m^2 or CO_2, although then, all text will be italicized which may not always be preferrable (like in the CO_2 example). Special characters such as greek symbols, however, can only (easily) be inserted by resorting to LaTeX language, e.g., (remember the first exercise of the last lesson) when you want to write something like Cronbach's α, MacDonalds' ω or η_p^2.

Writing more complicated formulae also requires some basic knowledge of LaTeX. For instance, you might want to provide the formula for a Pearson corelation coefficient. You can do it this way (the double dollar symbols will center the formular, with single dollar signs, the formula is left-aligned):

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{(n-1)\sigma_x\sigma_y}}$$

or, if you want the formula being numbered, like this:

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{(n-1)\sigma_x\sigma_y}} \tag{1}$$

You may also want to use longer quotes in, say, a thesis. Here is an example of how, using LaTeX language, you can indent and center a quote and then give the source of the quote below.

> Keep the faculty of effort alive in you by a little gratuitous exercise every day. That is, be systematically ascetic or heroic in little unnecessary points, do every day or two something for no other reason than that you would rather not do it, so that when the hour of dire need draws nigh, it may find you not unnerved and untrained to stand the test.
>
> *William James (1890)*

You can also include LaTeX code in prespecified figure captions as exemplified in Fig. 1. How you achieve this is exemplified in the header of the following (here invisible) code chunk.

Note that plotting a figure automatically generates a folder in the R Markdown file's directory that contains a PDF version of your figure. Most likely this figure's format and size may not be sufficient for submission to a journal. We will come back to this issue in the fourth lesson.

Interim Summary

In this lesson, you should have learned

- basic features of the LaTeX language
- how to use LaTeX to create a custom title page
- how to use LaTeX to write more complicated formulae and to customize the appearance of your documents

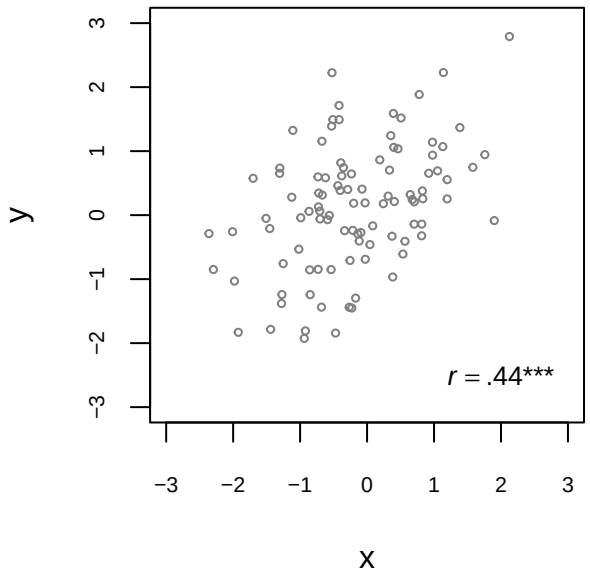

Figure 1: Scatterplot of the relationship between the variables of interest. $N = 100$. *** $p < .001$

Exercises

To exercise what you have learnt in this lesson, you might want to try the following:

1. Set up an R Markdown file with a title page conforming to APA standard.
2. Try to adapt the first paragraph of the section *Definition and features* of the *Wikipedia* entry on the t-statistic (https://en.wikipedia.org/wiki/T-statistic) using R Markdown and LaTeX.
3. Modify the function *apa.coef()* in the first code chunk of this document in a way that it returns $< .001$ in case a p-value is lower than .001.

Outlook

In the next lesson, we will elaborate on the present lesson, especially on how to

- use a BibTeX library as reference manager
- write custom code for your analysis pipeline without resorting to real data
- format tables and figures as well as their legends

References

James, W. (1890). *The Principles of Psychology.* New York: Holt.