

MGK OS WS 3 - Rmd practice

Alexander Strobel and Christoph Scheffel

July, 2022

R Markdown practice

In the following part, you will work on a few short tasks to gain a first insight into formatting text with R Markdown. Your first task is to format the text in this document so that all statistical coefficients are displayed correctly. That means, that (according to APA standard) statistical parameters are displayed in italics, are presented without leading zeroes if the respective coefficients cannot be greater than one, are given with two decimal digits except for p -values that are given with three decimal digits. As a help, here are some LaTeX examples on how to report coefficients in italics, use greek symbols or superscripts:

We standardized our variables to have a mean of $\mu = 0$ and a variance $\sigma^2 = 1$.

If you take a look in the .Rmd file, you can see three R chunks. In the first chunk called “variables”, three random variables were generated. Each variable has 100 observations. The variables were created in a way that the variables y and z to some extent covary with the variable x , but not with each other. Change the echo option of the R chunk “variables” so that it is visible in the output file.

Correlation analysis

Now calculate the correlations between x and y and x and z in the following chunk using the `cor.test` function. Report the results below the chunk as $r = .xx$, 95% CI $[.xx, .xx]$, $t(df) = x.xx$, and $p = .xxx$ by referring to the respective slots of the R object generated by `cor.test`. As an example, the correlation coefficient is stored in the slot `$estimate`. To do so properly, you will need to get rid of the leading zero of both the correlation coefficient and the p -value. We wrote a helper function `apa.coef` for this purpose.

To access and print the correlation coefficient, you need to refer to it in the text via $r = 0.2525676$. This gives you an unformatted correlation. A formatted version is printed via $r = .25$. The solution for the correlation of x and y would look like this: The correlation of the variables x and y was $r = .25$, 95% CI $[.06, .43]$, $t(98) = 2.58$, $p = .011$. That requires a lot of typing and you need to know which slot of the `cor.test` object contains what. Here, the `papaja` package comes in handy. Using its `apa_print` function, reporting the results of the correlation analysis boils down to: The correlation between x and y was $r = .25$, 95% CI $[.06, .43]$, $t(98) = 2.58$, $p = .011$.

Regression analysis

Now, we want to calculate a simple linear regression model using the `lm` function. The formula describing the model is $x \sim y + z$. Use the chunk “regression” to run the linear regression. Report the results in the text body of the document including beta-weights, t - and p -values, and multiple R-squared, all properly formatted according to APA standard.

Dynamic output

Finally, save the PDF output generated previously under a new name. Then change the factors by which y and z are multiplied by x in the chunk “variables”. Knit the document and compare the results in the new output file with the previous ones. If the resulting p -values are smaller than .001, the `apa.coef` function as provided will no longer produce a proper output. If time allows, you may want to edit it so that it retrieves $p < .001$ in such a case.

Additional task

If time allows, you may want to provide scatterplots of x and y as well as of x and z . You may try one or more of the following difficulty levels:

- simply provide two separate figures with the scatterplot of x and y and x and z , each with a figure caption
- provide a two panel plot denoted as A and B in the upper left corner of the plot together with one figure caption
- provide correlation of the variables in the lower right corner of the plot as (italic) $r = .xx$
- provide regression lines in the plots via `abline(lm(...))`
- provide confidence intervals of the regression line via `predict(lm())`