Example manuscript demonstrating the use of the papaja template

Frederik Aust[1]

[1] University of Cologne

Author note

Currently, a development version of `papaja` is available at https://github.com/crsh/papaja.

Correspondence concerning this article should be addressed to Frederik Aust, Department Psychology, University of Cologne, Herbert-Lewin-Str. 2, 50931 Köln, Germany. E-mail: frederik.aust@uni-koeln.de

Abstract

This example manuscript demonstrates how to use RStudio and RMarkdown to produce an APA conform manuscript. Using pandoc your RMarkdown can be converted to PDF or Word documents.

Keywords: APA, knitr, R, RMarkdown, papaja

Word count: Too lazy to count

Example manuscript demonstrating the use of the papaja template

What is papaja?

As you may have heard, recently, interest in reproducible research has been growing. Reproducible data analysis is an easy to implement and important aspect of the strive towards reproducibility. For *R* users, RMarkdown has been suggested as one possible framework for reproducible analyses. `papaja` is a R-package in the making including a RMarkdown template that can be used with RStudio (or without) to produce complete manscripts (PDF and Word documents), which conform to the American Psychological Association (APA) manuscript guidelines (6th Edition). To do so, papaja uses the LaTeXdocument class `apa6` and a `.docx`-reference file. The supplied R-functions are ment to facilitate the reporting of statistics in accordance with APA guidelines.

Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents (among others). In the following I will assume you have hoped onto the band wagon and know how to use RMarkdown to conduct and comment your analyses. If this is not the case, I recommend you get to grips with RMarkdown first. I use RStudio (which makes use of pandoc) to create my documents, but the general process should work using any other R-editor.

How do I use papaja?

Once you have installed `papaja` and all other required software you can select the APA template when creating a new Markdown file through the menus in RStudio. When you click RStudio's *Knit* button, `papaja`, `rmarkdown`, and `knitr` together create an APA conform manuscript that includes both your text and the output of any embedded R code chunks within the manuscript. Don't worry about the first chunk of R code at the beginning of the document. It sets the stage for the creation of your document.

```
require("papaja")

apa_prepare_doc() # Prepare document for rendering
```

Printing R output

Any output from R is included as you usually would using RMarkdown. By default the R code will not be displayed in the final documents. If you wish to show off your code you need to set `echo = TRUE` in the chunk options.

```
summary(cars)
```

```
 speed           dist
```

Min. : 4.0 Min. : 2.00

1st Qu.:12.0 1st Qu.: 26.00

Median :15.0 Median : 36.00

Mean :15.4 Mean : 42.98

3rd Qu.:19.0 3rd Qu.: 56.00

Max. :25.0 Max. :120.00

But, surely, this is not what you want your submission to look like. I think we can do better.

Print tables. For prettier tables, I suggest you have a look at `apa_table()`. Of course, e.g, the popular `xtable` or `tables` packages can also be used to create tables when knitting PDF documents. Unfortunately, `xtable()` captions are set to the left page margin. More importantly, these packages cannot be used when you want to create Microsoft Word documents because they rely on LaTeXfor typesetting. `apa_table()` creates tables that conform to APA guidelines and are correctly rendered in PDF and Word documents. But don't fool yourself. Table formatting is somewhat limited in Word documents due to missing functionality in pandoc (e.g., it is not possible to have cells span across multiple columns).

As required by the APA guidelines, in manuscripts tables are pushed to the final pages of the document when creating a PDF. Again, this is not the case in Word documents due to limited pandoc functionality. To place figures and tables in your text instead, set the `figsintext` parameter in the document header to `yes` or `true` as I have done in this document.

```
my_table <- apply(cars, 2, function(x) round(
    c(Mean = mean(x), SD = sd(x), Min = min(x), Max = max(x)), 2)
  )


apa_table(
  my_table
  , align = c("l", "r", "r")
  , caption = "A summary table of the cars dataset."
  , note = "This table was created using apa\\_table()"
  , added_colnames = "Descriptives"
)
```

The bottom line is, Word documents will be less polished than PDF. The resulting documents should suffice to enable collaboration with Wordy colleagues and prepare a journal submission.

Plots. You can also embed plots, for example:

```
plot(cars)
```

Again, as required by the APA guidelines, figures are pushed to the final pages of the document unless you set `figsintext` to `yes`.

Report statistical analyses. `apa_print()` will help you report the results of your statistical analyses. The function will format your input to produce readily reportable text.

Table 1

A summary table of the cars dataset.

Descriptives	speed	dist
Mean	15.4	42.98
SD	5.29	25.77
Min	4	2
Max	25	120

Note. This table was created using apa_table()

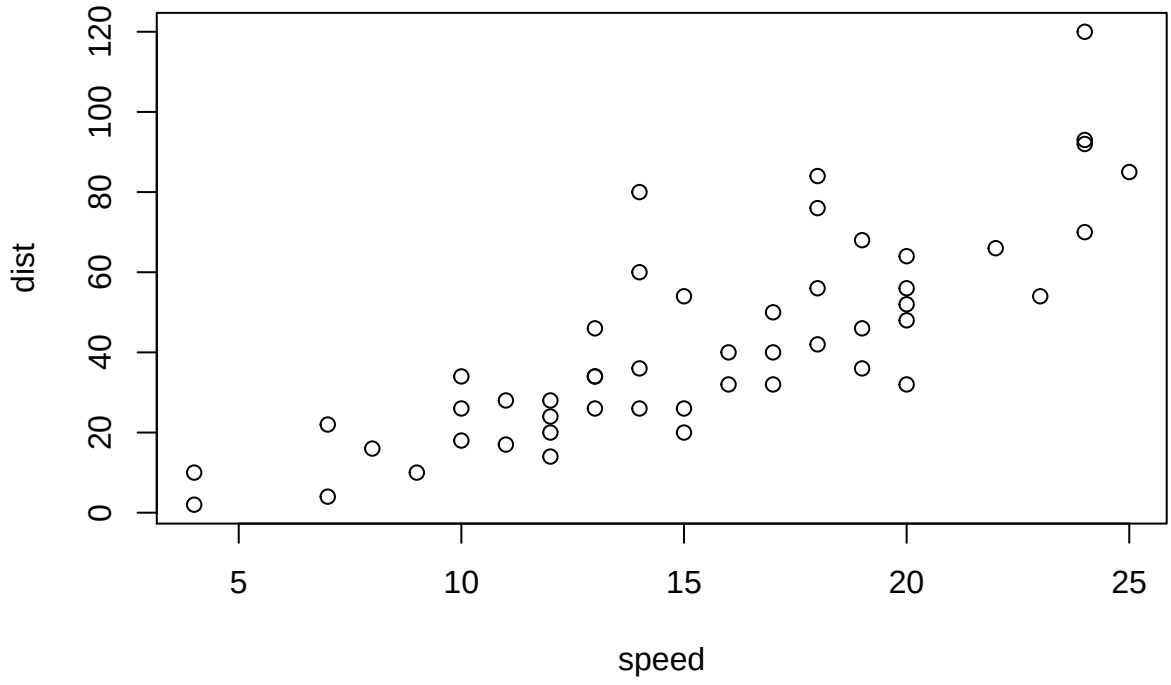

Figure 1. Exmple figure created by in-document R code.

Table 2

Regression table

Term	*b*	95% CI	*t*	*df*	*p*
Intercept	-17.58	[−31.17, −3.99]	-2.60	48	.012
Speed	3.93	[3.10, 4.77]	9.46	48	< .001

Note. This is a table generated by apa_print().

```
my_regression <- lm(dist ~ speed, cars)
my_results <- apa_print(my_regression)
```

In this case speed is a significant predictor of the distance taken to stop, $b = 3.93$, 95% CI [3.10, 4.77], $t(48) = 9.46$, $p < .001$. The regression explains $R^2 = .65$, 90% CI [0.51, 0.73] variance, which is of course statistically significant, $F(1, 48) = 89.57$, $p < .001$.

The previous paragraph was produced by the following text:

```
In this case speed is a significant precitor of the distance taken
to stop, `r my_results$full$speed`. The regression explains
`r my_results$est$modelfit$r2` variance, which is of course
statistically significant, `r my_results$stat$modelfit$r2`.
```

What's even more fun: You can easily create a complete regression table using by passing `my_results$table` to `apa_table()`:

Citations

You can insert citations like this:

`[e.g., @bauer_2014; @bem_2011]` → (e.g., Baumer, Cetinkaya-Rundel, Bray, Loi, & Horton, 2014; Bem, 2011).

Citing without parentheses is, of course, also possible:

`@bauer_2014` → Baumer et al. (2014).

The citation style is set in the header of this document with the `csl` parameter. The relevant references will, of course, be added to the documents references automatically. In order for citations to work, you need to supply a .bib-file to the `bibliography` parameter in the document header. See the RMarkdown documentation and Citation Style Language for further details.

Document options

This text is set as manuscript. If you want a thesis-like document you can change the `class` in the document header from `man` to `doc`. You can also preview a polished journal typesetting by changing the `class` to `jou`. Refer to the `apa6` document class documentation for further class options, such as paper size or draft watermarks.

When creating PDF documents, line numbering can be activated by setting the `lineno` argument in the header of this document to true. This option has no effect on Word documents.

Last words

That's all I have. Enjoy writing your manuscript. If you have any trouble or ideas for improvements, open an issue on GitHub or make a pull request with the fix. ;)

References

Baumer, B., Cetinkaya-Rundel, M., Bray, A., Loi, L., & Horton, N. J. (2014). R Markdown: Integrating A Reproducible Analysis Tool into Introductory Statistics. *ArXiv E-Prints*. Retrieved from http://adsabs.harvard.edu/abs/2014arXiv1402.1894B

Bem, D. J. (2011). Feeling the future: experimental evidence for anomalous retroactive influences on cognition and affect. *Journal of Personality and Social Psychology*, *100*(3), 407—425. doi:10.1037/a0021524