

aswp: An opinionated LaTeX template for R Markdown working papers*

A Short Introduction

András Svraka[†]

December 13, 2019

Abstract

This short vignette gives an introduction to the aswp R Markdown template. This template provides nice typography for a research paper and offers a simple way to add paragraph-long notes to figures and tables.

JEL codes: C88, Y20

Keywords: R Markdown, Reproducible research, Typography

*This is from the optional thanks field. Use this to add further notes that do not belong to any specific author. I'm standing on the shoulders of giants.

[†]This is from the optional affiliation field. It's just an unnumbered footnote for each author. Besides affiliation, you can put e.g. contact information here.

1 First steps

This package provides an `aswp()` function which is an R Markdown document format creator. To use it, you need to add `aswp::aswp` to the output field in the YAML metadata block of your R Markdown file. The recommended way to use this template is however by

```
rmarkdown::draft("paper.Rmd", template = "aswp", package = "aswp")
```

Or use your editor's interface to this function:

- In RStudio, open a new R Markdown document (**File** → **New File** and choose **R Markdown**) and select this template ("Working Paper") under **From Template**.
- In Emacs, use `M-x poly-r-rmarkdown-create-from-template` with **Poly-mode**, which is also available from the iESS menu.

This provides a basic skeleton for the most important, and non-obvious features and helps setting up a more complex document with references and appendices which are often used in research papers.

Then knit and compile like any other R Markdown document and you'll get a nicely formatted PDF. **TinyTeX** should automatically take care of the required \LaTeX packages.

2 Metadata fields

The basic YAML metadata fields for title, subtitle, abstract, data and thanks are available and behave similarly to the standard R Markdown template used in `rmarkdown::pdf_document()`. The keywords fields is also available and keywords are displayed after the abstract, not just added to the PDF's metadata fields. Additionally, a list of **JEL Codes** can be added in the same way using the `jel` field. These are also displayed after the abstract.

The author field works differently than in other templates, you need to add a name label, and an optional affiliation can be used to add affiliations and contact information.

```
author:  
- name: First Author  
  affiliation: Institute of Whatever, <name@whatever.gov>
```

```
- name: Second Author
  affiliation: Company, Inc., <name@company.com>
```

3 Citations

Add a bibliography field to the YAML frontmatter with the name of your bibliography file and cite works in the text using Pandoc’s standard [citation forms](#), e.g. @Xie2019 for Xie (2019). Both Bib \LaTeX and Pandoc’s default pandoc-citeproc can be used for citation rendering but pandoc-citeproc is recommended. It is the only way currently to put a “References” section before the appendices. Unfortunately Pandoc is not aware of \LaTeX ’s concept of appendices so you need to do the following incantations to add all sections to the paper in the right order, and with the right numbering.¹

This is the last sentence of the paper.

```
# References {-}

<div id="refs"></div>

\appendix
\clearpage

# Appendix

Start appendix here.
```

If you start your document with a template from this package (`rmarkdown::draft()`), references and an appendix section are already set up.

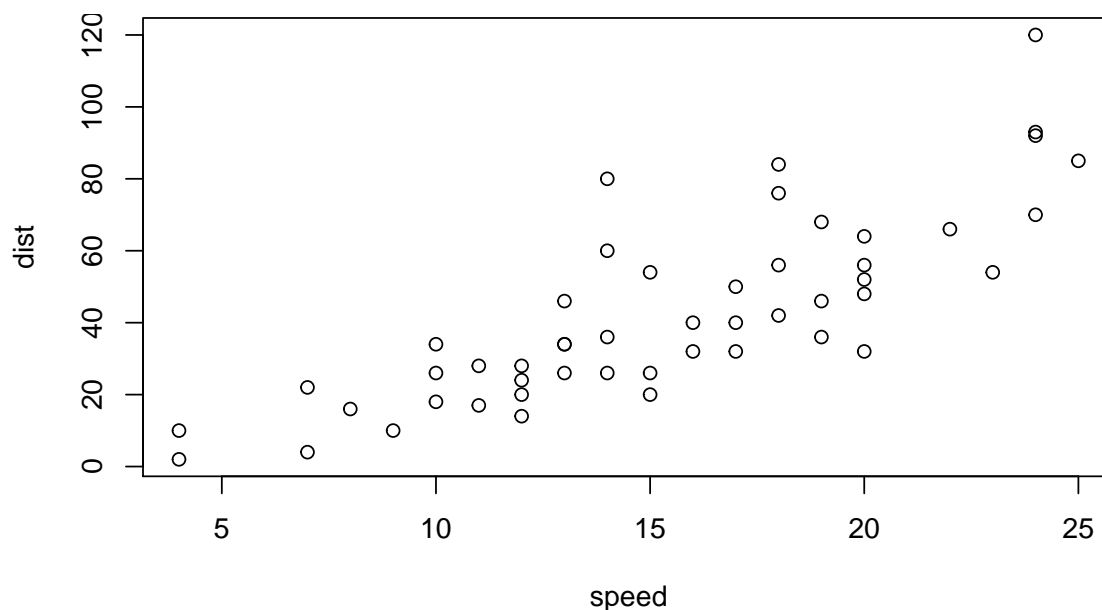
4 R Markdown

All the usual R Markdown features work but there are some extra functionalities.

This package builds on [bookdown](#) to produce a single long-form document. This means chapters are not available but you can use cross-references with the `\@ref(...)` syntax for [equations](#) and [text references](#): See Section [4](#).

¹I’m not aware of similar hacks for Bib \LaTeX .

Figure 1: A figure caption.



A figure note.

4.1 Figure and Table notes

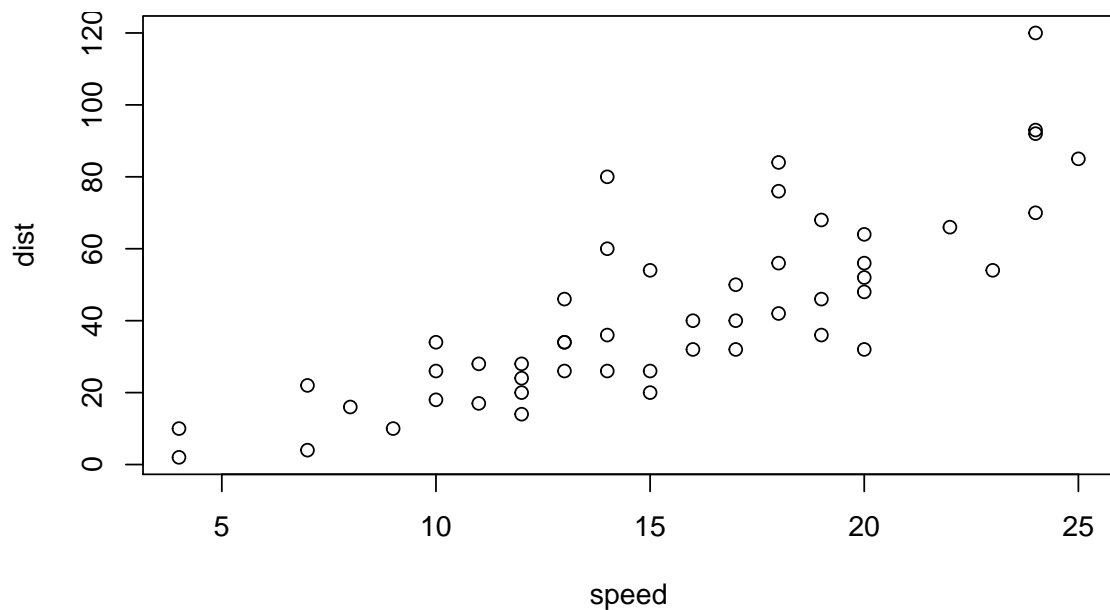
It is often useful to add a paragraph below a figure, or table with notes, like explanations and references. This allows self-contained figures and tables with short, concise captions yet you can still add more context. Standard R Markdown and **knitr** doesn't support this for floating environments, like \LaTeX figures and tables.

For this feature you need to load this package at the top your R Markdown document: `library(aswp)`. This will set up **knitr** hooks to insert `floatfoot` macros from the `floatrow` \LaTeX package into floating environments using `floatfoot` chunk options. See Figure 1 for a simple example.

```
```{r chunk-label, fig.cap = "A figure caption.",  
floatfoot = "A figure note."}
plot(cars)
```
```

It is however quite inconvenient to write longer sentences, or even full paragraphs as **knitr** chunk options. And you'd have to write raw \LaTeX code and escape all the backslashes. Fortunately **bookdown** has a nice feature where you can use text references anywhere. First, define them:

Figure 2: A scatterplot of the data cars using **base** R graphics.



Note: This my note from a *text reference*. It is a *very* long note that should overflow into multiple rows. This is all possible thanks to **bookdown** (Xie 2019). Figure 1 is much simpler. Let's mix some maths and R code, this dataset has $N = 50$ observations. Unfortunately this note *cannot be split into separate paragraphs* but that's probably OK.

```
(ref:foo) Define a text reference **here**.
```

Then you can use (ref:foo) as caption, or note. Both the caption and the note of Figure 2 were written in Markdown.

The same option can be used with R code that generates other floating environments, like tables, see Tables 1. The types of floating environments that make use of floatfoot options is defined in the **knitr** chunk option float-foot.envs, where it can be extended if needed:

```
knitr::opts_knit$get("floatfoot.envs")
```

```
## [1] "figure" "table"
```

```
knitr::opts_knit$set(floatfoot.envs = c("figure", "table", "myfloat"))
```

This can be useful for a number of packages and functions, like `knitr::kable()`, `stargazer::stargazer()`, `texreg::texreg()`, and

Table 1: This is a table

| speed | dist |
|-------|------|
| 4 | 2 |
| 4 | 10 |
| 7 | 4 |
| 7 | 22 |
| 8 | 16 |
| 9 | 10 |

Note: This my note from a *text reference*. It is a *very* long note that should overflow into multiple rows. This is all possible thanks to **bookdown** (Xie 2019). Figure 1 is much simpler. Let's mix some maths and R code, this dataset has $N = 50$ observations. Unfortunately this note **cannot** be split into separate paragraphs but that's probably OK.

`xtable::xtable()` that don't support any from of notes. Make sure these functions generate table environments and not `longtables`, as they are not yet supported. Other packages, like **huxtable** come with their own implementation for notes, while **kableExtra** has support for footnotes in tables. This package is mostly orthogonal to these features and the different notes they can be mixed. Here, we add notes with text at footnote size and the same width as the text column. This tends to look good with paragraph-long notes.

In these packages captions are defined using function arguments inside the code chunk. Support for markup in these captions varies but simple captions usually don't require additional markup, while notes can be written using text references.

As an added convenience, **knitr** hooks are set up to evaluate after the code chunk.

References

- Allaire, JJ, Yihui Xie, Jonathan McPherson, Javier Luraschi, Kevin Ushey, Aron Atkins, Hadley Wickham, Joe Cheng, Winston Chang, and Richard Iannone. 2019. *Rmarkdown: Dynamic Documents for R*. <https://github.com/rstudio/rmarkdown>.
- R Core Team. 2019. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.

- Xie, Yihui. 2016. *Bookdown: Authoring Books and Technical Documents with R Markdown*. Boca Raton, Florida: Chapman; Hall/CRC. <https://github.com/rstudio/bookdown>.
- . 2019. *Bookdown: Authoring Books and Technical Documents with R Markdown*. <https://github.com/rstudio/bookdown>.
- Xie, Yihui, J. J. Allaire, and Garrett Grolemund. 2018. *R Markdown: The Definitive Guide*. Boca Raton, Florida: Chapman; Hall/CRC. <https://bookdown.org/yihui/rmarkdown>.

A Tests

This is the appendix, which is numbered differently. This only serves for testing.

A.1 Typography

Body: The quick brown fox jumped over the lazy dog. Now in Hungarian: Árvíztűrő tükörfúrógép, and a visual test: aáeéííoóöőuúüő. "Simple quotes" "fancy quotes", and in „continental European” – just an aside – and again – but with unicode dashes – to see if works. Now in an American fashion—with an m-dash. Ligatures and other OpenType features: prefix, quantitative, a maffia mafla fia.

Emphasis: *The quick brown fox jumped over the lazy dog. Now in Hungarian: Árvíztűrő tükörfúrógép, and a visual test: aáeéííoóöőuúüő. "Simple quotes" "fancy quotes", and in „continental European” – just an aside – and again – but with unicode dashes – to see if works. Now in an American fashion—with an m-dash. Ligatures and other OpenType features: prefix, quantitative, a maffia mafla fia.*

Bold: The quick brown fox jumped over the lazy dog. Now in Hungarian: Árvíztűrő tükörfúrógép, and a visual test: aáeéííoóöőuúüő. "Simple quotes" "fancy quotes", and in „continental European” – just an aside – and again – but with unicode dashes – to see if works. Now in an American fashion—with an m-dash. Ligatures and other OpenType features: prefix, quantitative, a maffia mafla fia.

Monospaced: The quick brown fox jumped over the lazy dog. Now in Hungarian: Árvíztűrő tükörfúrógép, and a visual test: aáeéííoóöőuúüő. "Simple quotes" "fancy quotes", and in „continental European” – just an aside – and again – but with unicode dashes – to see if works. Now in an American fashion—with an m-dash. Ligatures and other OpenType features: prefix, quantitative, a maffia mafla fia.

A.1.1 Subsubsection

Lorem ipsum dolor sit amet, placerat, sed habitant odio egestas sed aliquet eget sed, fermentum. Arcu lacinia justo interdum integer mauris. At ut nibh parturient turpis. Vel nullam at urna, ut ad himenaeos nisi augue convallis convallis eu.

Dictumst ac, augue vel egestas inceptos. Feugiat sed lacus sed venenatis habitant luctus, ipsum vestibulum mattis leo eget iaculis nullam. Odio fames et eu urna. Rhoncus purus malesuada convallis in elementum, euismod at fusce id, tempus vitae, nec. Odio justo eros cursus justo, eros lacinia lorem lectus nec et. Elit nunc suscipit venenatis nibh nec nec amet ultrices suscipit duis egestas volutpat eget semper dolor ligula. Rutrum in, praesent gravida mauris malesuada ut cursus, ut, leo.

Porta sodales donec, purus sed tempus dictum. Id egestas vulputate sed finibus nisl eget malesuada. Vulputate sagittis justo purus nec class fringilla eget. Curabitur vivamus euismod pretium malesuada nam scelerisque volutpat neque dapibus imperdiet. Id nunc sed in vulputate quis mauris nec aptent, facilisis sapien. Vivamus sed sed donec est. Eros tempus pretium est ipsum posuere suspendisse et.

Paragraph This is a paragraph that shouldn't be numbered.

Another paragraph Default Pandoc templates redefine this but we're using standard L^AT_EX titled paragraphs, where text starts on the same line as the title.

A.2 Link colours

This is a [link to a web address](#), these are citations Xie (2019) and (2016; R Core Team 2019; Xie, Allaire, and Golemund 2018; Allaire et al. 2019), this is an internal reference to Section 4, and this sentence ends with a footnote.²

A.3 Maths

Check spacing of indices and parentheses:

$$f_a(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!} (x-a)^n \quad (x+a)^n = \sum_{k=0}^n \binom{n}{k} x^k a^{n-k}$$

Unicode and traditional symbols:

xxx αβγδεΓ αβγ

²This is the footnote.

A.4 Numbers

- In math mode: 123,456,789.0
- The same in a math environment as text: 123,456,789.0
- In text mode 123456789.