Writing reproducible manuscripts in R Markdown and Pagedown

Camille Landesvatter
Mannheim Centre for European Research (MZES), University of Mannheim

camille.landesvatter@uni-mannheim.de
MZES Website
Camille Landesvatter

Dr. Paul C. Bauer
Mannheim Centre for European Research (MZES), University of Mannheim

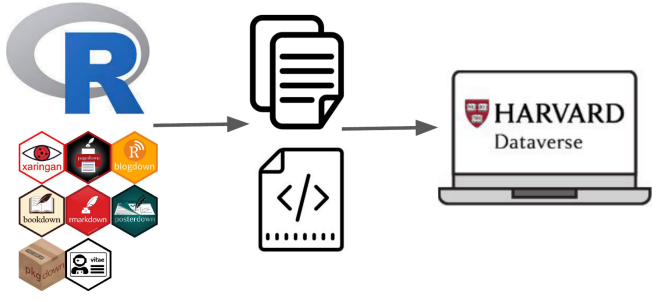
mail@paulcbauer.de
MZES Website
Paul C. Bauer

Mannheim Open Science Day Reprohack

Oktober 11, 2022

Why reproducible research?

- **Reproducibility**: Everyone (including yourself) can trace the steps that led to a particular result or statement. This increases transparency (and credibility).
- **Access**: Open access is given when research is freely available to everyone without financial, legal, or technical barriers (UNESCO 2022), for example commercial software. E.g.,



- **Errors**: Everyone (including yourself) can spot errors. Plus, automation (e.g., code that produces findings is tied to the document itself) avoids a slow and error-prone cut-and-paste approach.
- **Revisions**: Revision of manuscripts and other pieces of research gets easier when steps are reproducible and automated. Plus, easy version control with git, etc.

Tools for reproducible manuscripts in R

- **rmarkdown**
  - combines R code and markdown-formatted text to produce HTML, PDF or Word documents
- **pagedown** *(Xie et al. 2022)*
  - R Markdown based
  - creates paginated HTML reports and prints to PDF
  - this includes headers, footers, etc.
  - highly customizable due to HTML and CSS styling sheets (YAML Metadata)
  - many available templates, e.g. CV, thesis, letter, journal templates

- Other (R Markdown based): posterdown, bookdown, xaringan, etc.

Source: Indrajeet Patil on Twitter

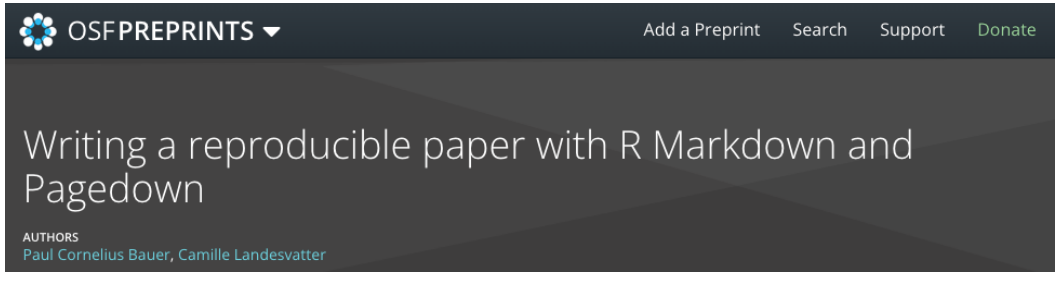
Two templates (on OSF, Gdrive and Github)

Bauer, P. 2021:

: /paulcbauer/Writing_a_reproducable_paper_in_rmarkdown

: /2021_Writing_a_reproducable_paper_in_rmarkdown

Bauer, P.; Landesvatter, C. 2022:



: /paulcbauer/Writing_a_reproducable_paper_in_pagedown

: /2021_Writing_a_reproducable_paper_in_pagedown

How can you use these templates?

- download or clone all available files:
  - 'paper.rmd'
  - 'references.bib'
  - 'data.csv'
  - 'american-sociological-association.csl'
  - styling files for pagedown: 'wp_paged.html', 'wp.css', 'wp-fonts.css', 'wp-pages.css'
- open .Rproj
- start creating your first manuscript in rmd, use our working paper (wp) style or modify the styling sheets

Pagedown: Working Paper (wp) Style

Writing a reproducible paper with R Markdown and Pagedown[2]

Paul C. Bauer
Mannheim Centre for European Social Research
mail@paulcbauer.eu

Camille Landesvatter
Mannheim Centre for European Social Research
camille.landesvatter@mzes.uni-mannheim.de

First version: 20 June, 2021
This version: 04 Oktober, 2022

Download: https://osf.io/k8jhx

Abstract

The present paper provides a template to write a reproducible scientific paper with R Markdown and Pagedown.[1] Below we outline some of the "tricks"/code (e.g., referencing tables, sections etc.) we had to figure out to produce this document. The underlying files which produce this document can be downloaded here. Importantly, we also provide different CSS and HTML files that can be used to achieve a pdf output with the look of a "working paper". We are convinced that in the future there will be many improvements and developments with regards to RStudio, R markdown and Pagedown. We intend to update this file when we discover more convenient code. You can follow any updates on the github repository.

Keywords: open science, transparency, replication, reproducible research, reproducibility, R, markdown, pagedown.

Pre-Requisites for R Markdown and Pagedown

Both require: installation of R + recommended: Rstudio IDE

- R Markdown:
  - rmarkdown package + other dependencies, e.g. knitr package
  - for working without Rstudio: Pandoc Installation
  - for PDF output: a LaTeX distribution, e.g. tinytex (a lightweight, cross-platform LaTeX distribution)

```
install.packages(c('tinytex', 'rmarkdown'))
tinytex::install_tinytex()
```

- Pagedown:
  - rmarkdown package + other dependencies, e.g. knitr
  - pagedown package
  - no LaTeX!

```
remotes::install_github('rstudio/pagedown')
```

Three Examples

1. Inline code & results
2. Figures
3. Tables

1. Inline code & results

Example 1:

Writing `` `r 3 + 7` `` (use back ticks!) will calculate and insert 10 into your text.

Example 2:

Imagine you want to automate the display of regression coefficients within your manuscript.

```
data ← read.csv("data.csv")
```

```
M1 ← lm(Fertility ~ Education + Agriculture, data = data)
```

Now we can write some plain text with inline code: The coefficient of `Education` of the model we ran above is -0.96. -0.96 was printed by writing: `` `r round(coef(M1)[2], 2)` `` (use back ticks!).

What advantages does such a workflow bring?

2. Figures

Example: You want to create, edit and insert a ggplot2 graph alongside its code and the text.

```
data$Education_cat ← cut(data$Education,
                   breaks=c(-Inf, 6, 12, Inf), labels=c("low","middle","high"))
```

````
```{r fig-1, fig.cap="Fig 1. Fertility and Agriculture by Education", fig.align="center",
fig.width=6, fig.height=3, eval=T, echo=T}
ggplot(data, aes(x=Catholic, y=Fertility, shape=Education_cat)) + geom_point() +
  labs(x="Agriculture", y = "Fertility", shape="Education") + theme_classic()
```
````

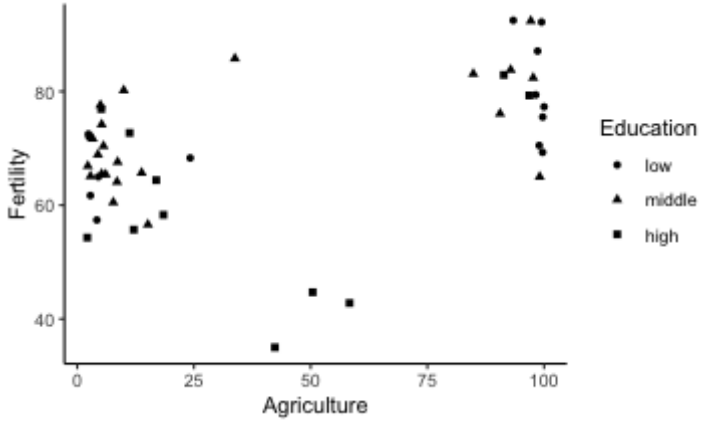


Fig 1. Fertility and Agriculture by Education

2. Figures

- For cross-referencing within your manuscript you can refer to Figure 1 by using `\@ref(fig:fig-1)`.
- But pay attention to your code chunk options. In the previous example:

> {r fig-1, fig.cap="Fig 1. Fertility and Agriculture by Education", fig.align="center", fig.width=6, fig.height=3, eval=T, echo=T}

What do the different code chunk options mean?

3. Tables

Example 1: You want to show summary statistics of your data.

```
library(stargazer)
stargazer(data,
          title = "Summary table with stargaz
          label="tab1",
          header=FALSE,
          table.placement = "H",
          type="html",
          font.size = "small")
```

Summary table with stargazer

Statistic	N	Mean	St. Dev.	Min	Max
Fertility	47	70.143	12.492	35.000	92.500
Agriculture	47	50.660	22.711	1.200	89.700
Examination	47	16.489	7.978	3	37
Education	47	10.979	9.615	1	53
Catholic	47	41.144	41.705	2.150	100.000
Infant.Mortality	47	19.943	2.913	10.800	26.600

3. Tables

Example 1b: You want to show summary statistics of a subset of your data.

```
library(stargazer)
stargazer(data %>% filter(Education >= 15),
          title = "Summary table with stargaz
          label="tab1",
          header=FALSE,
          table.placement = "H",
          type="html",
          font.size = "small")
```

Summary table with stargazer

Statistic	N	Mean	St. Dev.	Min	Max
Fertility	8	54.013	13.239	35.000	76.900
Agriculture	8	24.750	14.964	1.200	46.600
Examination	8	26.125	7.791	16	37
Education	8	28.125	11.692	15	53
Catholic	8	25.738	21.527	2.150	58.330
Infant.Mortality	8	18.875	3.634	10.800	23.000

3. Tables

```
```{r table-1, eval=T, echo=T, results='hide'}
M1 ← lm(Fertility ~ Education + Agriculture, data = data)
M2 ← lm(Fertility ~ Education + Catholic, data = data)
models ← list("M1" = M1, "M2" =  M2)

library(gt)
library(modelsummary)

modelsummary(models,
             title = 'Table 1. Linear regression',
             output = 'gt',
             notes = "Notes: some notes ... ") %>%
  tab_spanner(label = 'Dependent variable: Fertility', columns = 2:3) %>%
  tab_options(
    table.font.size = 16, data_row.padding = px(1),
    table.border.top.color = "white", heading.border.bottom.color = "black",
    row_group.border.top.color = "black",
    row_group.border.bottom.color = "white", table.border.bottom.color = "white",
    column_labels.border.top.color = "black", column_labels.border.bottom.color = "black",
    table_body.hlines.color = "white"
  )
```
```

- For cross-referencing within your manuscript you can refer to Table 1 by using `\@ref(tab:table-1)` (use table-1 as the chunk name!).

Table 1. Linear regression

	Dependent variable: Fertility	
	M1	M2
(Intercept)	84.080	74.234
	(5.782)	(2.352)
Education	-0.963	-0.788
	(0.189)	(0.129)
Agriculture	-0.066	
	(0.080)	
Catholic		0.111
		(0.030)
Num.Obs.	47	47
R2	0.449	0.575
R2 Adj.	0.424	0.555
AIC	349.7	337.6
BIC	357.1	345.0
RMSE	9.17	8.06

Notes: some notes...

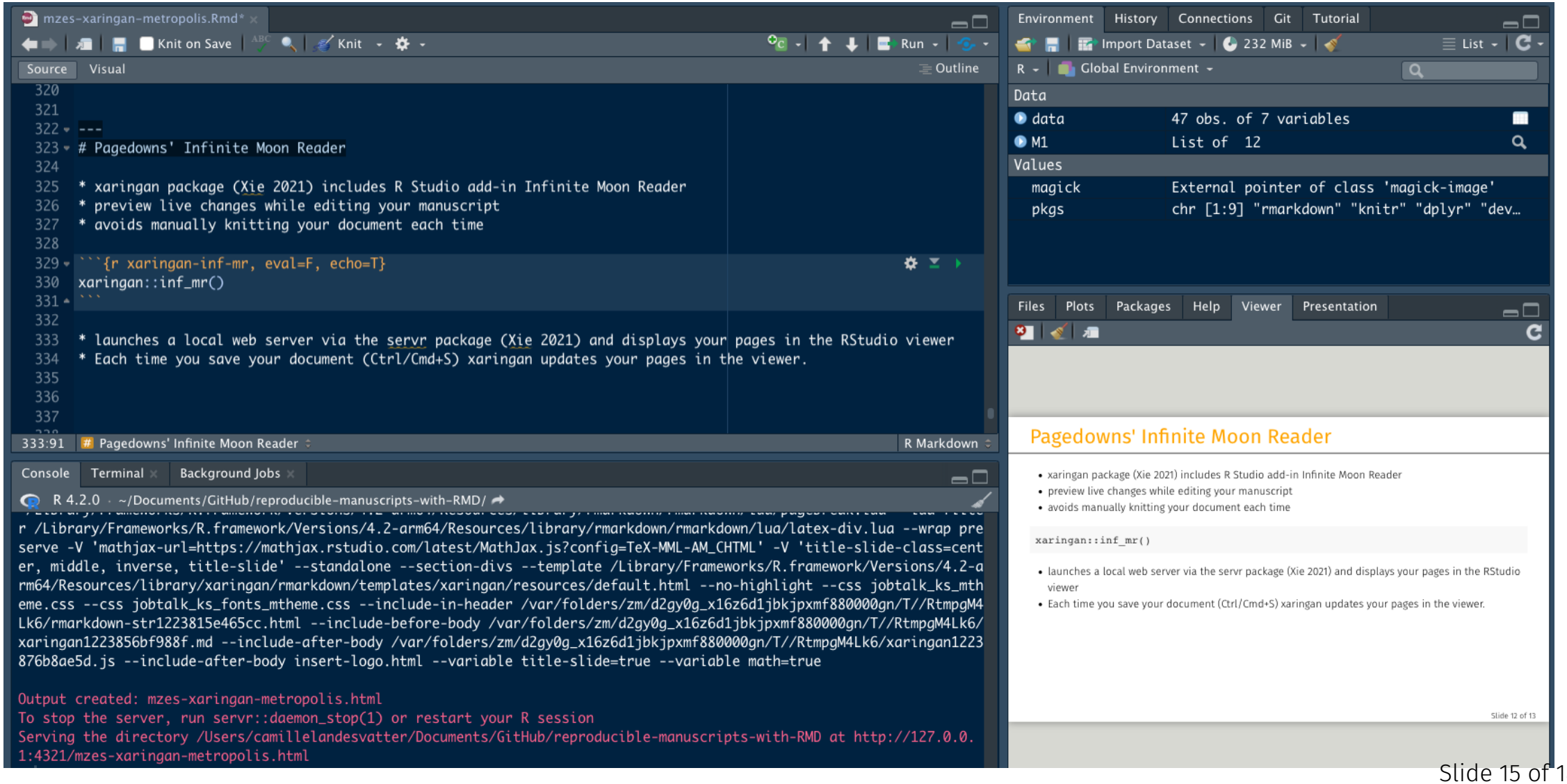
Xaringans' Infinite Moon Reader

- xaringan package (Xie 2021) includes R Studio add-in Infinite Moon Reader
- preview live changes while editing your files or even manuscript
- avoids manually knitting your document each time

```
xaringan::inf_mr()
```

- launches a local web server via the servr package (Xie 2021) and displays your pages in the RStudio viewer
- each time you save your document (Ctrl/Cmd+S) xaringan updates your pages in the viewer.

Xaringans' Infinite Moon Reader



R trackdown package for improving collaborative writing

: ClaudioZandonella/trackdown

- collaborative workflow for the writing and editing process with R Markdown
- upload local .Rmd (or .Rnw) file as a plain-text file to Google Drive
- use Markdown (or LaTeX) syntax and Google Docs options (e.g. suggest edits, add comments)
- review and integrate all contributions
- the final document can be downloaded and rendered locally

R trackdown package for improving collaborative writing

```
remotes::install_github("claudiozandonella/trackdown",
                        build_vignettes = TRUE)
```

```
upload_file(file = "trackdown_example.Rmd",
            gpath = "trackdown_example",
            hide_code = TRUE)
```

- `download_file()`
- `update_file()`
- `render_file()`

Thank you!

: clandesv/reproducible-manuscripts-with-RMD

Xaringan Presentation on GitHub Pages: https://clandesv.github.io/reproducible-manuscripts-with-RMD/slides.html

R Markdown Template on OSF: https://osf.io/q395s

R Markdown and Pagedown Template on OSF: https://osf.io/k8jhx/