

APA-style manuscripts with RMarkdown and papaja

Dr James Bartlett

What problems are we trying to solve?

Computational reproducibility

- 31% reproducibility success rate in one journal ([Hardwicke et al., 2018](#))
- 58% reproducibility success rate in registered reports ([Obels et al., 2020](#))

Reporting errors

- 49.6% of articles contained an inconsistency between test statistic, degrees of freedom, and p -values ([Nuijten et al., 2016](#))

What problems are we trying to solve?

Connection between code and output

- Even when authors share data and reproducible code, there might not be a clear connection between the code and relevant output in the manuscript (anecdote)

Reproducible manuscripts

- Potential to avoid these errors as you combine code, results, and prose in one document
- When there are errors, there are *reproducible errors*

My journey

It took me several years and projects to adopt this workflow:

1. Copying results from SPSS output
2. Copying results from R output
3. Using R Markdown to create reproducible results
4. Using papaja to write reproducible manuscripts

papaja

papaja ([Aust & Barth, 2022](#)) adds templates when you create a new R Markdown document

```
1 # Install latest CRAN release
2 install.packages("papaja")
```

In this tutorial, I will walk through:

- YAML options
- Citations/references via a .bib file
- Inline code
- Tables / figures

Mock example: Error-free vs error-full code

- I've created a mock example using papaja and simulated data to show its capabilities
- Building on [Hoffman and Elmi \(2021\)](#): What is the effect of teaching debugging skills on students' data wrangling ability?
- Randomly allocate students to an error-free or error-full lecture (IV) and measure performance on a data skills assignment (DV)

YAML options

Author information

- Name and affiliation for each author, but only one corresponding author
- Option to include contributorship roles, such as [CRediT](#).

```
---
author:
  - name      : "James Bartlett"
    affiliation : "1"
    corresponding : yes      # Define only one corresponding author
    address     : "62 Hillhead Street, Glasgow"
    email       : "james.bartlett@glasgow.ac.uk"
    role:
      - "Conceptualization"
      - "Writing - Original Draft Preparation"
      - "Writing - Review & Editing"
---
```

YAML options

Adding a .bib file

I recommend Zotero as a reference manager:

<https://www.zotero.org/>

- Create a collection
- Export collection
- Format BibTeX and OK
- Save as in your document working directory

```
---  
bibliography      : ["references.bib", "r-references.bib"]  
---
```


YAML options

Changing the reference style

- Once you have a .bib file, you can easily change the style by selecting a different citation style language (CSL)
- Over 10,000 in the Zotero style repository, just save as and add .csl to the file: <https://www.zotero.org/styles>

E.g., APA 7th edition

```
---  
csl          : apa7.csl  
---
```

YAML options

Changing the reference style

- Once you have a .bib file, you can easily change the style by selecting a different citation style language (CSL)
- Over 10,000 in the Zotero style repository, just save as and add .csl to the file: <https://www.zotero.org/styles>

E.g., Vancouver

```
---  
csl          : vancouver.csl  
---
```

YAML options

Changing the reference style

APA 7th edition

Data skills are increasingly recognised as a key component of psychological literacy. To promote reproducible data preparation and analysis workflows, educators have highlighted the role of teaching students how to use statistical programming languages instead of point-and-click software (McAleer et al., 2022). However, programming is rare in UK psychology curricula (TARG Meta-Research Group, 2022) and offers unique challenges such as how to prepare students to debug their code. Debugging code is a separate problem solving skill to learn alongside statistics, so it is important to understand how best to teach students debugging skills.

Vancouver

Data skills are increasingly recognised as a key component of psychological literacy. To promote reproducible data preparation and analysis workflows, educators have highlighted the role of teaching students how to use statistical programming languages instead of point-and-click software (1). However, programming is rare in UK psychology curricula (2) and offers unique challenges such as how to prepare students to debug their code. Debugging code is a separate problem solving skill to learn alongside statistics, so it is important to understand how best to teach students debugging skills.

YAML options

Manuscript options

Depending on the journal submission guidelines, you can change different features like:

- Floating figures/tables in-text or at the end
- Being kind to your reviewer and adding line numbers
- Masking the manuscript and omitting author information

```
---
floatsintext      : yes # Figures and tables floating or at the end?
linenumbers       : yes # Add line numbers?
draft             : no  # Add draft watermark on every page?
mask              : no  # Hide author details for blind submission?
---
```

YAML options

Output options

- The default output with the best functionality for the knitted document is a PDF:

```
---  
output              : papaja::apa6_pdf  
---
```

- However, you can also knit a Word document if you need it:

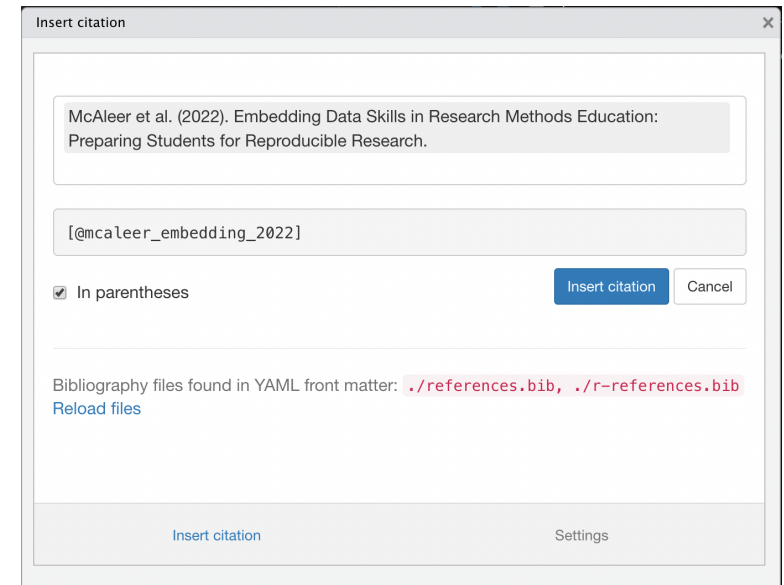
```
---  
output              : papaja::apa6_word  
---
```

Citations and references

citr

Alongside papaja, Aust created a great helper package called citr which makes it easy to browse a .bib file and insert citations.

```
1 # Not currently on CRAN
2 devtools::install_github("crsh/citr")
```



Inline code

Power analysis

```
1 # Inputs hidden on slides to save space
2
3 sample_size <- ceiling(
4   pwr.t.test(d = small_telescopes,
5             sig.level = alpha,
6             power = power,
7             type = "two.sample",
8             alternative = "two.sided")$n
9 )
```

Using an effect size of $d = 0.38$, we aimed to recruit 149 participants per group for an independent samples t-test ($\alpha = 0.05$, power = 0.9).

Inline code

Power analysis

Behind the scenes...

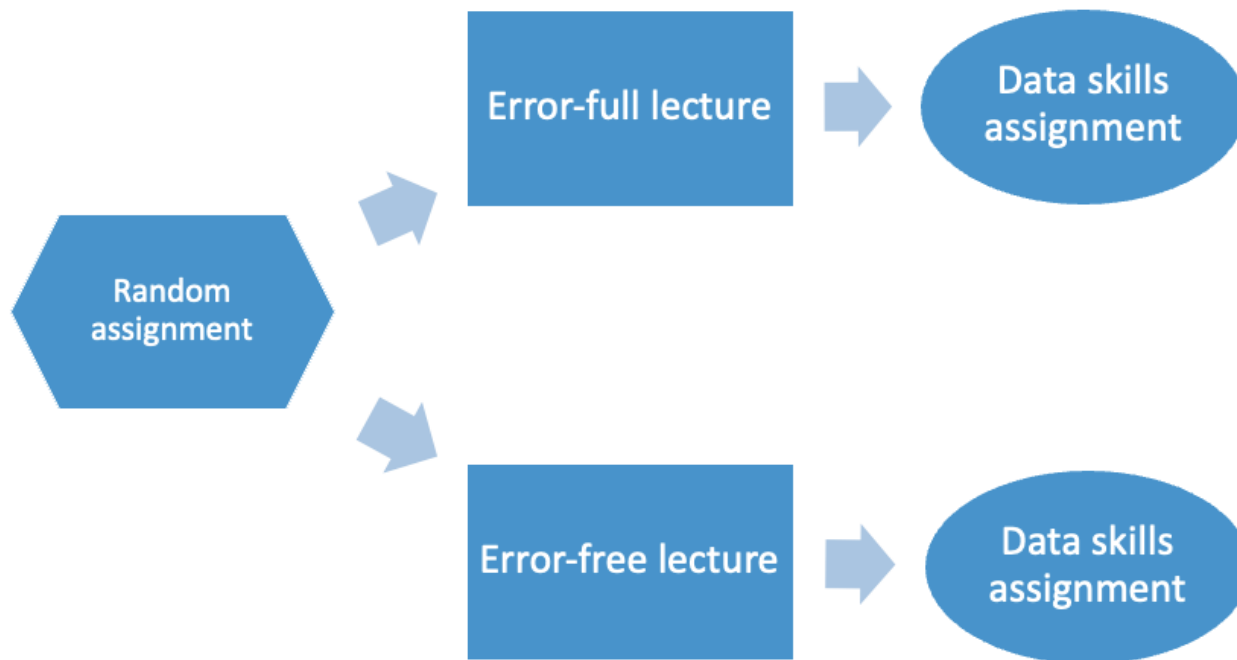
Using an effect size of $d = 'r_small_telescopes'$, we aimed to recruit $'r_sample_size'$ participants per group for an independent samples t-test ($'\alpha' = 'r_alpha'$, $power = 'r_power'$).

Adding external images

papaja supports adding external images via knitr:

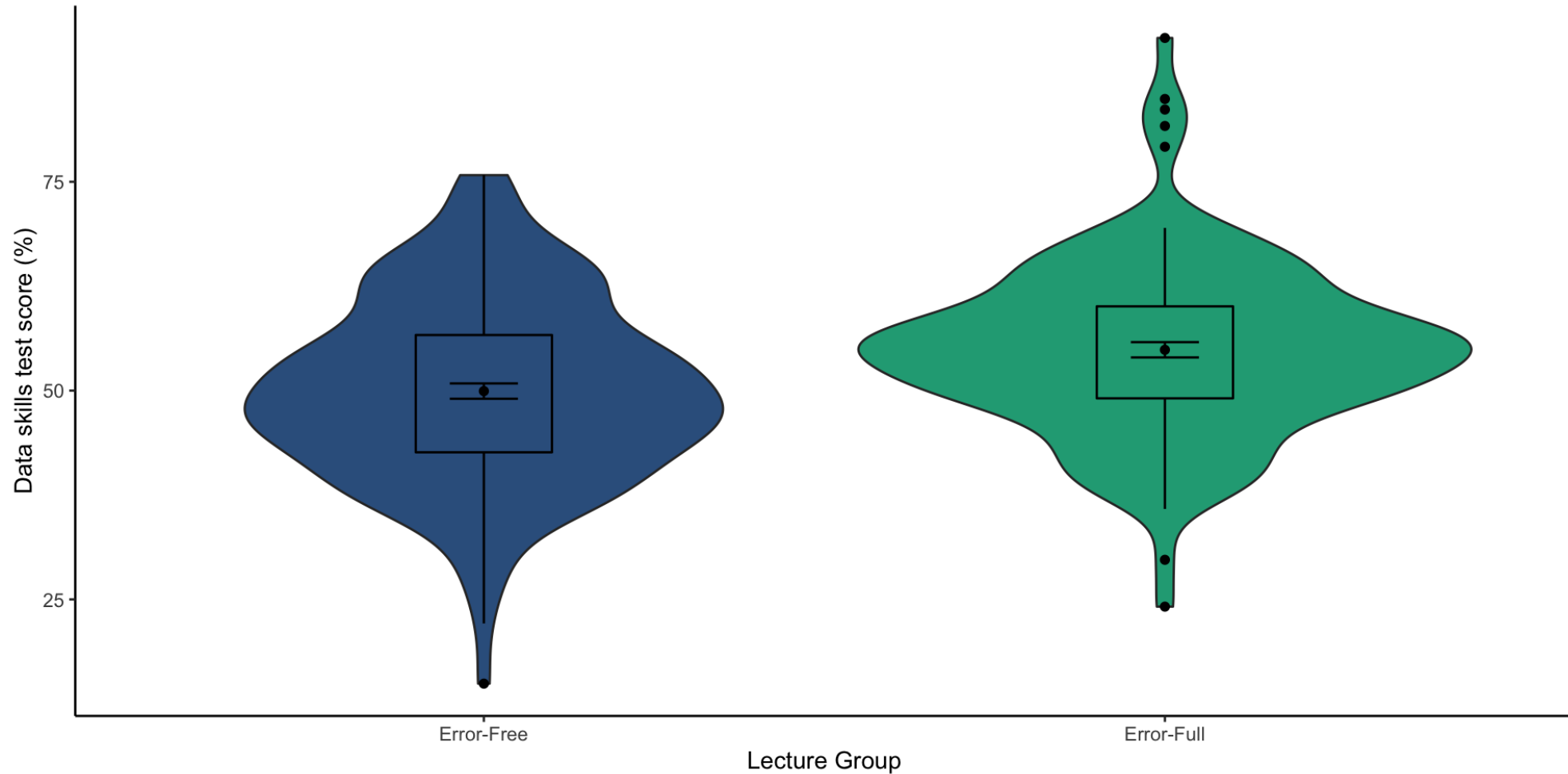
http://frederikaust.com/papaja_man/reporting.html#figures

```
1 knitr::include_graphics("Screenshots/procedure_diagram.png")
```



Figures

You display reproducible graphs from your code chunks



Figures

Behind the scenes...

```
1 mock_data %>%
2   ggplot(aes(x = Group, y = DV, fill = Group)) +
3   geom_violin() +
4   # remove the median line with fatten = NULL
5   geom_boxplot(width = .2,
6               fatten = NULL, colour = "black") +
7   stat_summary(fun = "mean", geom = "point") +
8   stat_summary(fun.data = "mean_se",
9               geom = "errorbar",
10              width = .1) +
11   scale_fill_viridis_d(option = "D", begin = 0.3, end = 0.6) +
12   theme_classic() +
13   theme(legend.position = "None") +
14   labs(x = "Lecture Group",
15        y = "Data skills test score (%)")
```

Figures

In the code chunk settings, you can do things like reference a caption and control the size of figures

```
Figure \@ref(fig:violin-plot) shows...

(ref:violin-plot-caption) Violin and boxplot of...

```{r violin-plot, fig.cap="(ref:violin-plot-caption)", out.width="100%"}

mock_data %>%
 ggplot(aes(x = Group, y = DV, fill = Group)) +
 geom_violin()...

```
```

Tables

papaja has some helper functions for creating APA style tables (which don't play nicely with html...):

http://frederikaust.com/papaja_man/reporting.html#tables

(#tab:unnamed-chunk-8)

Descriptive statistics of...

| Group | Mean | SD | Min | Max |
|------------|-------|-------|-------|-------|
| Error-Free | 49.94 | 11.03 | 14.91 | 75.80 |
| Error-Full | 54.88 | 10.42 | 24.13 | 92.22 |

Note. Test scores could range from 0-100%

Tables

Behind the scenes...

```
1 # Calculate descriptives
2 mock_descriptives <- mock_data %>%
3   group_by(Group) %>%
4   summarise(Mean = mean(DV),
5             SD = sd(DV),
6             Min = min(DV),
7             Max = max(DV)) %>%
8   ungroup()
9
10 # papaja function to round and save as character
11 descriptives <- printnum(mock_descriptives)
12
13 # papaja function to create APA table
14 apa_table(descriptives,
15           caption = "Descriptive statistics of...",
16           note = "Test scores could range from 0-100%")
```

Inline code

Statistical tests

papaja has helper functions for creating APA style result formatting:

http://frederikaust.com/papaja_man/reporting.html#statistical-models-and-tests

“Consistent with our hypothesis, a Welch t-test shows that participants in the error-full group produced significantly higher data skills assignment scores than those in the error-free group, $\Delta M = -4.94$, 95% CI $[-7.49, -2.40]$, $t(272.23) = -3.82, p < .001$.”

Inline code

Statistical tests

Behind the scenes...

```
1 # Save ttest as object
2 mock_ttest <- t.test(DV ~ Group,
3                       data = mock_data,
4                       paired = FALSE)
5
6 # papaja helper function of printing results in APA
7 apa_ttest <- apa_print(mock_ttest)$full_result
```

“Consistent with our hypothesis, a Welch t-test shows that participants in the error-full group produced significantly higher data skills assignment scores than those in the error-free group, 'r apa_ttest'.”

Where to learn more?

- Slides and folder containing mock example available on Github: https://github.com/BartlettJE/papaja_demo
- Full example from my recent publication: <https://osf.io/gm4jr/>
- papaja manual (section 7 includes published manuscripts using papaja): http://frederikaust.com/papaja_man/