

Writing Reproducible Research Papers with R Markdown

Resul Umit

April 2020

Who am I?

Resul Umit

- post-doctoral researcher and lecturer at the Department of Political Science, University of Lucerne
- interested in representation, elections, and parliaments
 - **most recent publication**: When members of parliament disagree with their principals (country, constituency, party), how do they justify this to them?
 - more at resulmit.com

How did I use to write?

First, with **Stata + Word**, I was ...

- frustrated with Word
 - formatting tables, figures, citations, and equations
 - managing references
- tired of switching between programmes/screens
 - and, worried about making mistakes in between
- paying for programme licences

How did I use to write?

Then, with Stata + R + LaTeX, I was ...

- ~~frustrated with Word~~
 - ~~formatting tables, figures, citations, and equations~~
 - ~~managing references~~
- tired of switching between programmes/screens
 - and, worried about making mistakes in between
- paying for the Stata licence
- converting PDF documents to Word manually
 - coordinating work with co-authors who don't use LaTeX/PDF
 - submitting to journals which don't accept LaTeX/PDF

How do I write now?

Now, with **R Markdown**, I am ... happy!

- ~~frustrated with Word~~
 - ~~formatting tables, figures, citations, and equations~~
 - ~~managing references~~
- ~~tired of switching between programmes/screens~~
 - ~~and, worried about making mistakes in between~~
- ~~paying for the stata licence~~
- ~~converting PDF documents to Word, manually~~
 - ~~coordinating work with co-authors who don't use LaTeX/PDF~~
 - ~~submitting to journals which don't accept LaTeX/PDF~~

R Markdown

- Efficient
 - write text, cite sources, tidy data, analyse, table, and plot it in one programme/screen
 - re-do one, more, or all of these with ease
 - decrease the possibility of making mistakes in the process
- Flexible
 - output to various formats
 - e.g., HTML, LaTeX, PDF, Word
- Open access/source
 - use for free
 - create documents accessible to anyone with a computer and internet connection
 - benefit from the work of a great community of users/developers

Reproducibility — Before Publication

- Having written a complete draft
 - with data including re-coded variables, tables, figures, and text with references to specific results (e.g., numbers from summary and/or regression statistics)
- If you and/or your co-authors decide
 - to reverse a re-coded variable to its previous/original measure
 - and/or, to exclude a subgroup of observations from analysis
- How resource intensive would this revision be?
 - how long would this revision take?
 - how many programmes would be needed for this revision, and how much would they cost?
 - there is an inverse relationship between this resource intensity and reproducibility

Reproducibility — After Publication

- After your paper is published, if others, including your future self, would like to test how robust the results are
 - to reversing a re-coded variable to its previous/original measure
 - and/or, to excluding a subgroup of observations from analysis
- How resource intensive would this test be?
 - how accessible is the data, documentation (how was the variable re-coded in the first place?), and the code?
 - how long would the test take?
 - how many programmes would be needed for this revision, and how much would they cost?
 - there is an inverse relationship between this resource intensity and reproducibility

The Workshop — Overview

- Two half days, on how to write reproducible research papers with R Markdown
 - ideally, about ten hours
- Based on converting a mock manuscript written in Word to R Markdown
 - plus, improving its reproducibility and version-controlling it
 - with a PDF output in mind
- Designed for researchers with basic knowledge of R programming language
 - does not cover programming with R
 - e.g., writing functions
 - ability to regress, plot, and table in R will be very helpful
 - but not absolutely necessary — these skills can be developed after learning R Markdown as well

The Workshop — Contents

Part 1. Getting the Tools Ready

- e.g., downloading course material

Part 2. Introducing R Markdown

- e.g., creating a new document

Part 3. Setting Metadata

- e.g., defining output format

Part 4. Writing Text

- e.g., adding emphasis to text

Part 5. Managing References

- e.g., citing sources

Part 6. Adding Code, Figures, and Tables

- e.g., plotting data

Part 7. Addressing Functionality Gaps

- e.g., adjusting line spacing

Part 8. Using Version Control

- e.g., integrating Git and GitHub

Part 9. Collaborating with Others

- e.g., working simultaneously with co-authors

Part 10. Working on a Real Project

- e.g., converting a work-in-progress of yours

The Workshop — Organisation

- Sit in groups of two
 - participants learn as much from their partner as from instructors
 - switch partners after every second part
- Type, rather than copy/paste, the code that you will find on these slides
 - typing is a part of the learning process
- When you have a R-related question:
 - ask your partner
 - Google together
 - ask me

The Workshop — Aims

- To make you aware what is possible with R Markdown
 - we will cover a large breath of issues, not all of it is for long-term memory
 - awareness of what is possible, Google, and perseverance are all we need
- To encourage you to convert into R Markdown
 - practice with the mock manuscript (Parts 3–9)
 - start converting a real one (Part 10)

Part 1. Getting the Tools Ready

Course Materials — Download from the Internet

Materials, available at https://github.com/resulunit/rmd_workshop, have the following structure:

```
rmd_workshop
|
|- manuscript
|   |- journals.Rmd
|   |- references.bib
|   |- apa_7th.csl
|   |- reproduce_this.pdf
|
|-data
|   |- journals.csv
|
|-image
|   |- google_scholar.png
```

Course Materials — Contents

- `reproduce_this.pdf`
 - the document, formatted in Word but saved as PDF, that we will re-create with R Markdown
 - randomly generated sentences,^{*} with figures and tables from randomly generated data, with key sections in-need of attention highlighted
- `journals.Rmd`
 - the R Markdown document that we will work on
 - includes unformatted text from `reproduce_this.pdf` to save time
- `references.bib`
 - the BibTeX document with three fabricated references
- `apa_7th.csl`
 - a *Citation Style Language* document, with APA (7th Edition) referencing style (Wiernik, 2020)

[*] *Lorem ipsum* text, generated with the `stringi` package (Gagolewski, 2020).

Course Materials — Contents

journals.csv

- a dataset created with the `fabricatr` package (Blair et al., 2019), imagined to explore the Google Scholar rankings of fictitious journals
- including the following variables:
 - **name**: journals (1090 random titles)
 - **origin**: geographic origins (five continents)
 - **branch**: major discipline of journals (four branches)
 - **since**: time of first publication (years)
 - **h5_index**: H5 Index (integers)
 - **h5_median**: H5 Median (integers)
 - **english**: English (1) vs. other-language (0) journals
 - **subfield**: subfield (1) vs. generalist (0) journals
 - **issues**: number of issues published per year (integers)

Git — Download from the Internet and Install

- For Windows, install *Git for Windows*, downloading from <https://gitforwindows.org>
 - select 'Git from the command line and also from 3rd-party software'
- For Mac, install *Git*, downloading from <https://git-scm.com/downloads>

GitHub — Open an Account

Sign up for GitHub at <https://github.com>

- registering an account is free
- usernames are public
 - either choose an anonymous username (e.g., asdf029348)
 - or choose one carefully — it becomes a part of users online presence
- usernames can be changed later

R and RStudio — Download from the Internet and Install

- R at <https://cloud.r-project.org>
 - choose the version for your operating system
- RStudio at <https://rstudio.com/products/rstudio/download>
 - choose the free version

R Packages — Install from within RStudio

```
install.packages(c("rmarkdown", "tinytex",  
                  "dplyr", "ggplot2", "stargazer"))
```

- `rmarkdown` (Allaire et al., 2020), for automating the process of converting R Markdown documents into other formats
- `tinytex` (Xie, 2020), for PDF outputs
 - alternative: a TeX/LaTeX system installed on your computer
- `dplyr` (Wickham, 2020a), for data manipulation
 - alternatives: e.g., `base`, `data.table` (Dowle & Srinivasan, 2019)
- `ggplot2` (Wickham, 2020b), for figures
 - alternatives: e.g., `base`, `plotly` (Sievert et al., 2020)
- `stargazer` (Hlavac, 2018), for tables
 - alternatives: e.g., `knitr` (Xie, 2020b), `kableExtra` (Zhu, 2019)

R Project — Create from within RStudio

- RStudio allows for dividing your work with R into separate projects, each with own history etc.
 - [this page](#) has more information on why R projects are recommended
- Create a new R project for the existing directory `.../rmd_workshop/manuscript` from the RStudio menu:

```
File -> New Project -> Existing Directory -> Browse ->  
.../rmd_workshop/manuscript -> Open
```

- Rename the `.Rproj` file into something more meaningful
 - the default RStudio behaviour is to name the projects after the existing directory
 - but, we don't want all our projects to be named `manuscript.Rproj`

Other Resources*

- R Markdown Cheat Sheet
 - follow from the RStudio menu
- Help -> Cheatsheets -> R Markdown Cheat Sheet
- Pandoc User's Guide
 - available at <https://pandoc.org/MANUAL.html>
- R Markdown: The Definitive Guide (Xie et al., 2019)
 - open access at <https://bookdown.org/yihui/rmarkdown>
- R for Data Science (Wickham and Grolemund, 2019)
 - open access at <https://r4ds.had.co.nz>

[*] *During* the workshop, R Markdown Cheat Sheet is likely to be the most helpful of these resources. I recommend the remaining to be consulted after the workshop.

Part 3. Introducing R Markdown

R Markdown Document — Create from within RStudio

- Create a new R Markdown document from the RStudio menu:*

```
File -> New File -> R Markdown -> OK
```

- Save your new document:**

```
File -> Save
```

- Observe that
 - the document has been saved to your working directory, and
 - it has the .Rmd extension

[*] This is for demonstration purposes only. Otherwise, we will work with `journals.Rmd`, which you have already downloaded, to save time.

[**] Alternatively, use the Save button or the keyboard shortcut (e.g., `Ctrl + S` on Windows). For shortcuts, follow `Tools -> Keyboard Shortcuts Help` or `Tools -> Modify Keyboard Shortcuts....`

R Markdown Document — Components

Observe also that the document has three components

- **YAML**

```
1 ---  
2 title: "Untitled"  
3 output: html_document  
4 ---
```

R Markdown Document — Components

Observe also that the document has three components

- YAML
- text

```
1 ---
2 title: "Untitled"
3 output: html_document
4 ---
5
6
7
8
9
10
11
12 ## R Markdown
13
14 This is an R Markdown document. Markdown is
15 and MS Word documents. For more details on u
16 When you click the Knit button a document
17 well as the output of any embedded R code ch
18 chunk like this:
```

R Markdown Document — Components

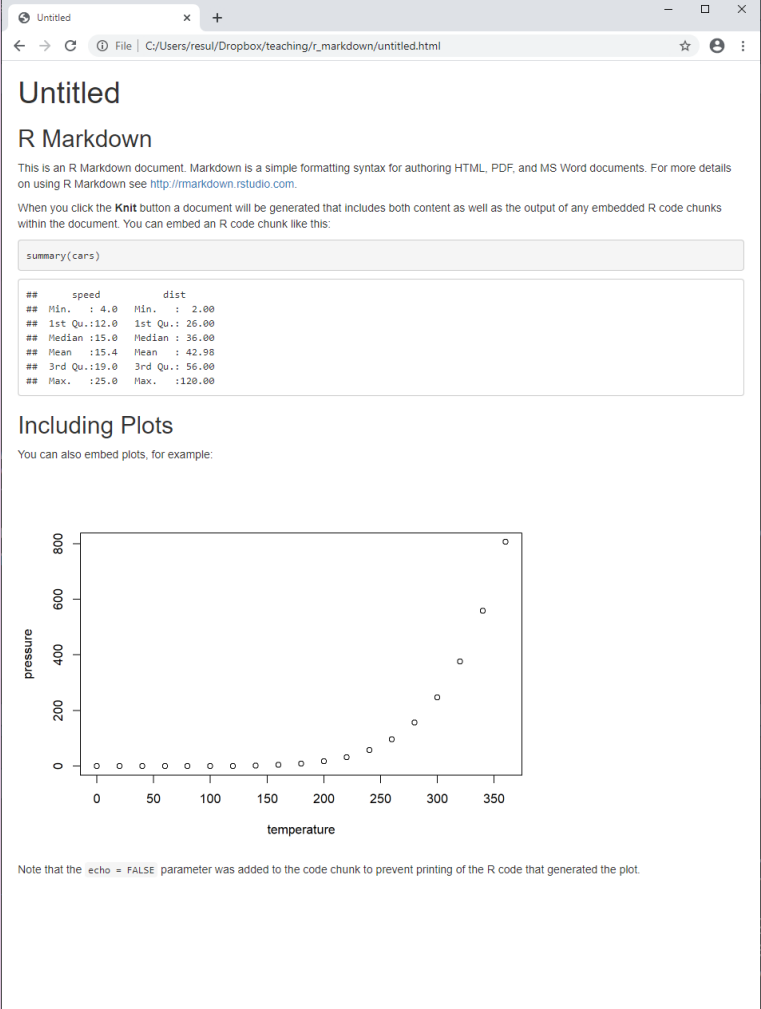
Observe also that the document has three components

- YAML
- text
- code chunks

```
1 ---
2 title: "Untitled"
3 output: html_document
4 ---
5
6
7
8
9
10
11
12 ## R Markdown
13
14 This is an R Markdown document. Markdown is a lightweight
15 and easy to write markup language that can be used to create
16 and MS Word documents. For more details on using R Markdown see
17
18 When you click the Knit button a document will be generated
19 that includes both the text and the output of any embedded R code
20 chunks like this:
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

R Markdown Document — Compile

- Click the **Knit** button to compile it, and observe that
 - the output document has the same name as your `.Rmd` document
- You may want to delete these newly created files, as we will work with `journals.Rmd` instead to save time.



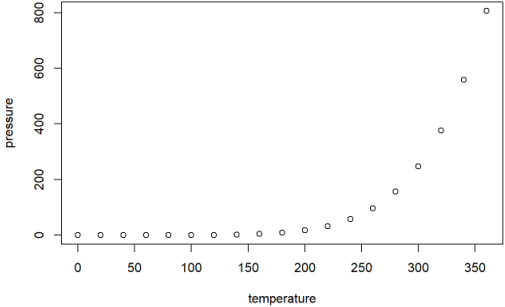
The screenshot shows a web browser window titled "Untitled" with the address bar displaying "C:/Users/resul/Dropbox/teaching/r_markdown/untitled.html". The page content is an R Markdown document titled "Untitled" and "R Markdown". It includes a text block explaining R Markdown, a code chunk titled "summary(cars)" that displays a summary of the 'cars' dataset, a section titled "Including Plots" with an example plot of pressure vs. temperature, and a note about the 'echo' parameter.

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

Including Plots

You can also embed plots, for example:



Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

R Markdown Document — Compilation Process

- When you Knit, the following happens:

```
.Rmd --knitr--> .md --pandoc--> output
```

- `knitr`^{*} executes the code if there is any, converts the resulting document from .Rmd (R Markdown) into .md (Markdown)
- `pandoc`^{**} transforms the .md document into your preferred output format(s)
 - e.g., HTML, LaTeX, PDF, Word
- This process is automated by the `rmarkdown` package

[*] If you had not already have the `knitr` package, it would have been installed together with the `rmarkdown` package.

[**] RStudio comes with a copy of `pandoc` (<http://pandoc.org>), which is not an R package, so that you do not have to install it separately.

R Markdown Document — Notes

- Behind the scenes, each .Rmd file is compiled in its own session, and therefore
 - the code needs to stand alone, for reproducibility reasons
 - you might have already loaded a package and/or imported data elsewhere (e.g., in a separate .R and/or .Rmd file), but even in the same session, these won't be available to a given .Rmd file
- R Markdown can produce more than documents,^{*} including
 - presentations, again with rmarkdown
 - books, with bookdown (Xie, 2020c)
 - websites, with blogdown (Xie, 2020d)

[*] Here we will focus on academic papers only. In a separate workshop, I teach how to create professional websites with R Blogdown.

Part 3. Setting Metadata

YAML — Overview

.Rmd documents start^{*} with YAML

- it includes the metadata variables
 - e.g., title, output format
- it is written between a pair of three hyphens -

```
---  
title:  
output:  
---
```

[*] Tecnically, we can place YAML anywhere in a .Rmd document. However, it is a good practice to start with YAML.

YAML — Overview

.Rmd documents start with YAML

- it includes the metadata variables
 - e.g., title, output format
- it is written between a pair of three hyphens -

```
---  
title:  
output:  
---
```

Variables can take strings

```
---  
title: "Journals: Random Words With Random Data"  
output:  
---
```

YAML — Overview

.Rmd documents start with YAML

- it includes the metadata variables
 - e.g., title, output format
- it is written between a pair of three hyphens -

```
---  
title:  
output:  
---
```

Variables can take strings, options

```
---  
title: "Journals: Random Words With Random Data"  
output: pdf_document  
---
```

YAML — Overview

.Rmd documents start with YAML

- it includes the metadata variables
 - e.g., title, output format
- it is written between a pair of three hyphens -

```
---  
title:  
output:  
---
```

Variables can take strings, options, sub-options

```
---  
title: "Journals: Random Words With Random Data"  
output:  
  pdf_document:  
    keep_tex: true  
---
```

YAML — Overview

.Rmd documents start with YAML

- it includes the metadata variables
 - e.g., title, output format
- it is written between a pair of three hyphens -

```
---  
title:  
output:  
---
```

Variables can take strings, options, sub-options, and code

```
---  
title: "Journals: Random Words With Random Data"  
date: "r format(Sys.Date(), '%d %B %Y')"  
output:  
  pdf_document:  
    keep_tex: true  
---
```

YAML — Variables

- `title` and `output` are the basic variables of YAML
 - variable names are typed in lower case, followed by a colon `:`
 - the list of available variables, as well as options and sub-options for these variables, depends on the output format
 - [Pandoc User's Guide](#) provides a comprehensive documentation
 - R Markdown Cheat Sheet provides a helpful list
- Typical YAML variables for an academic paper:

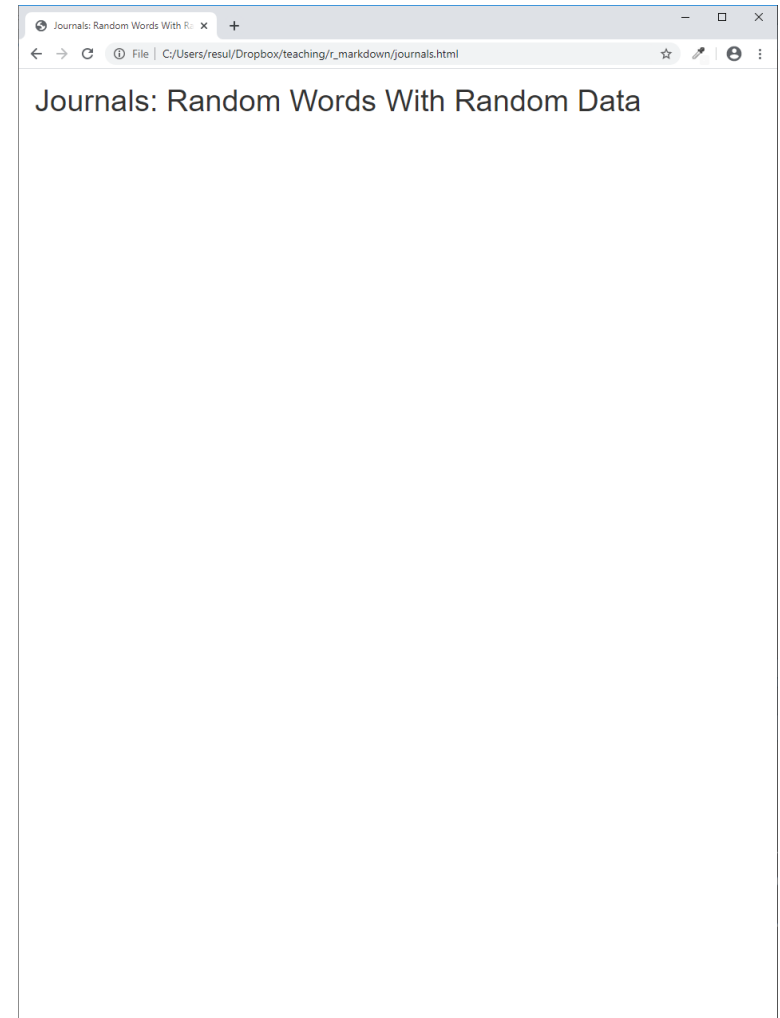
```
---  
title:  
author:  
date:  
bibliography:  
csl:  
output:  
---
```

YAML — Variables — Output Formats

Documents as output formats include

- **HTML**

```
---  
title: "Journals: Random Words With Random Data"  
output: html_document  
---
```

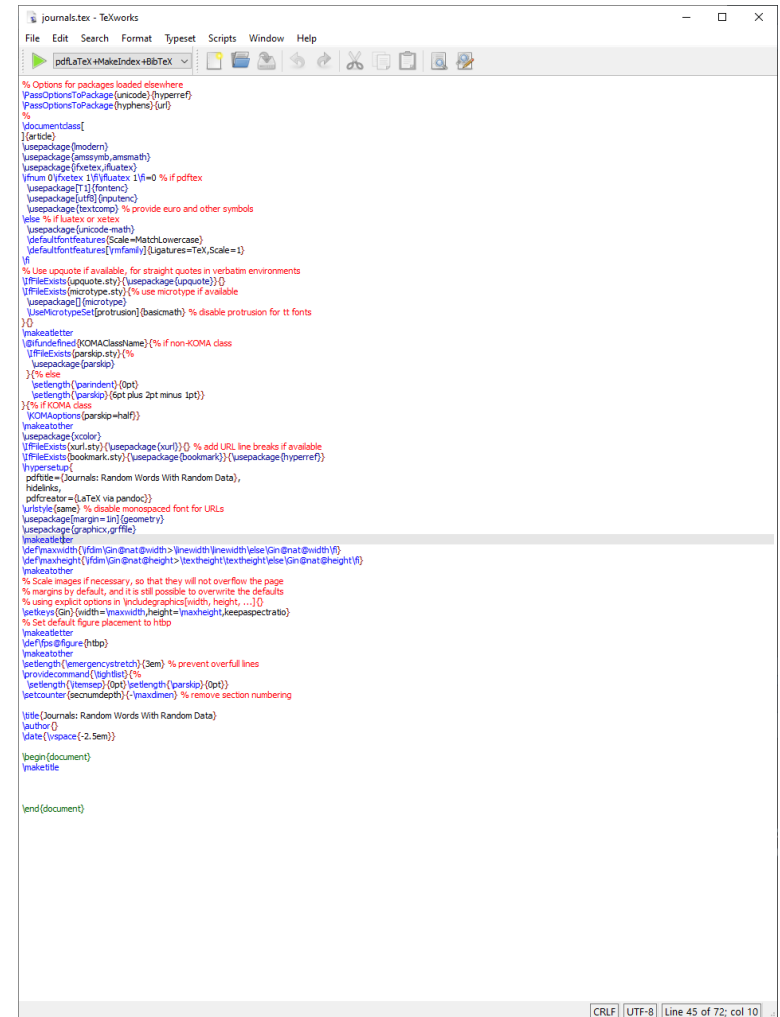


YAML — Variables — Output Formats

Documents as output formats include

- HTML
- **LaTeX**

```
---  
title: "Journals: Random Words With Random Data"  
output: latex_document  
---
```

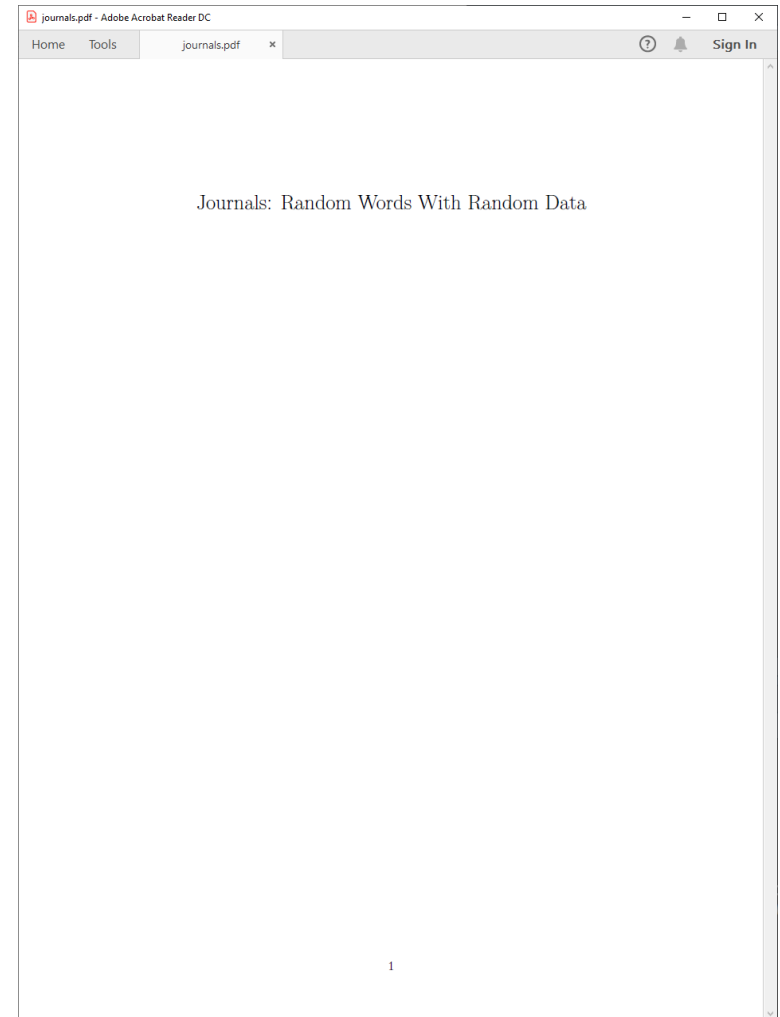


YAML — Variables — Output Formats

Documents as output formats include

- HTML
- LaTeX
- PDF

```
---  
title: "Journals: Random Words With Random Da  
output: pdf_document  
---
```

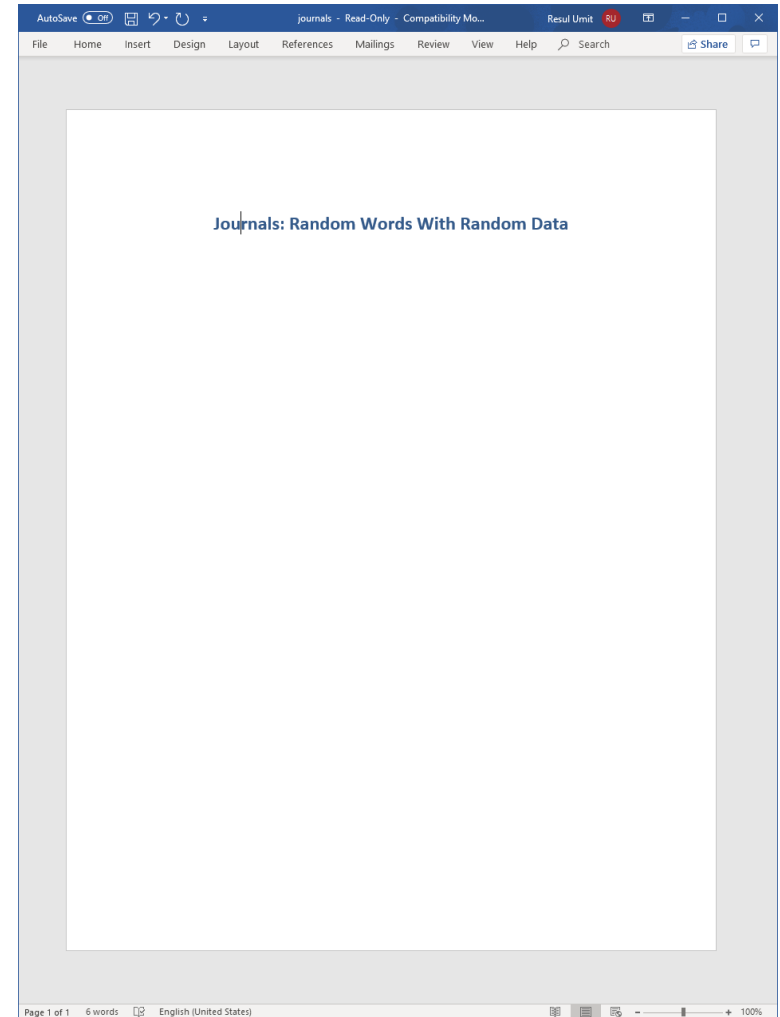


YAML — Variables — Output Formats

Documents as output formats include

- HTML
- LaTeX
- PDF
- Word

```
---  
title: "Journals: Random Words With Random Data"  
output: word_document  
---
```



YAML — Variables — Output Formats

- Documents as output formats

- `html_document`
- `latex_document`
- `pdf_document`*
- `word_document`
- `github_document`
- `md_document`
- `odt_document`
- `rtf_document`

- Presentations as output formats

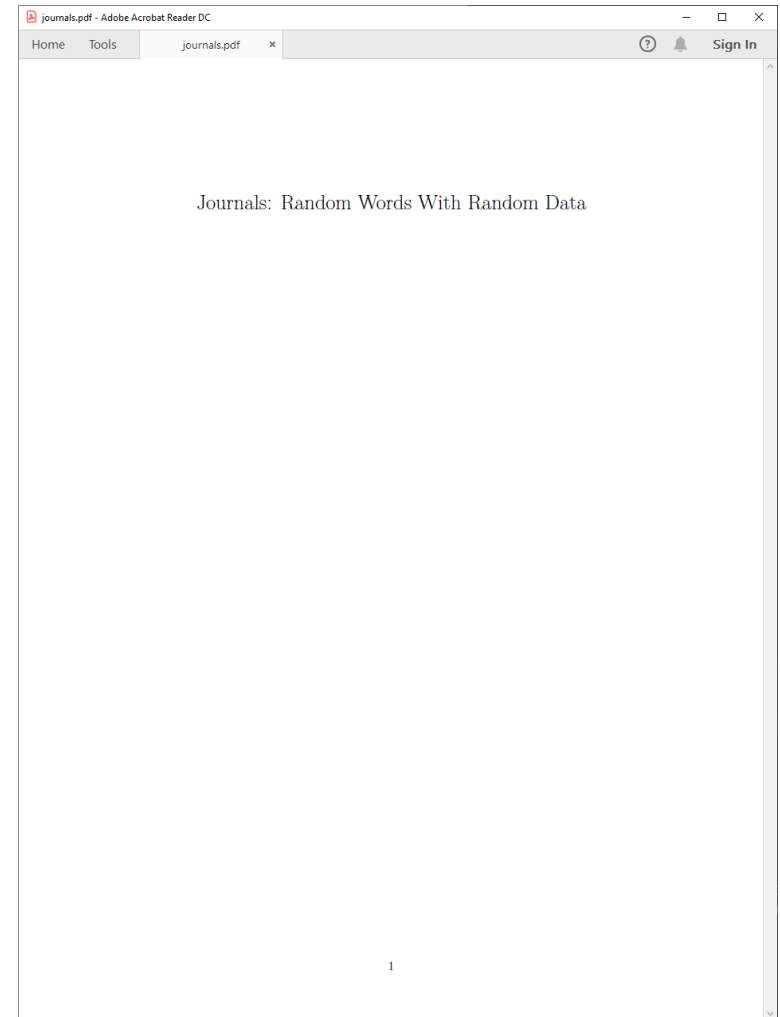
- `beamer_presentation`
- `ioslides_presentation`
- `powerpoint_presentation`
- `slidy_presentation`

[*] For reasons of simplicity, this workshop focuses on LaTeX and/or PDF outputs. Different output formats have slightly different customisations. see [Pandoc User's Guide](#) and/or R Markdown Cheat Sheet.

YAML — Strings

Strings with special characters, such as colon, require quotation marks

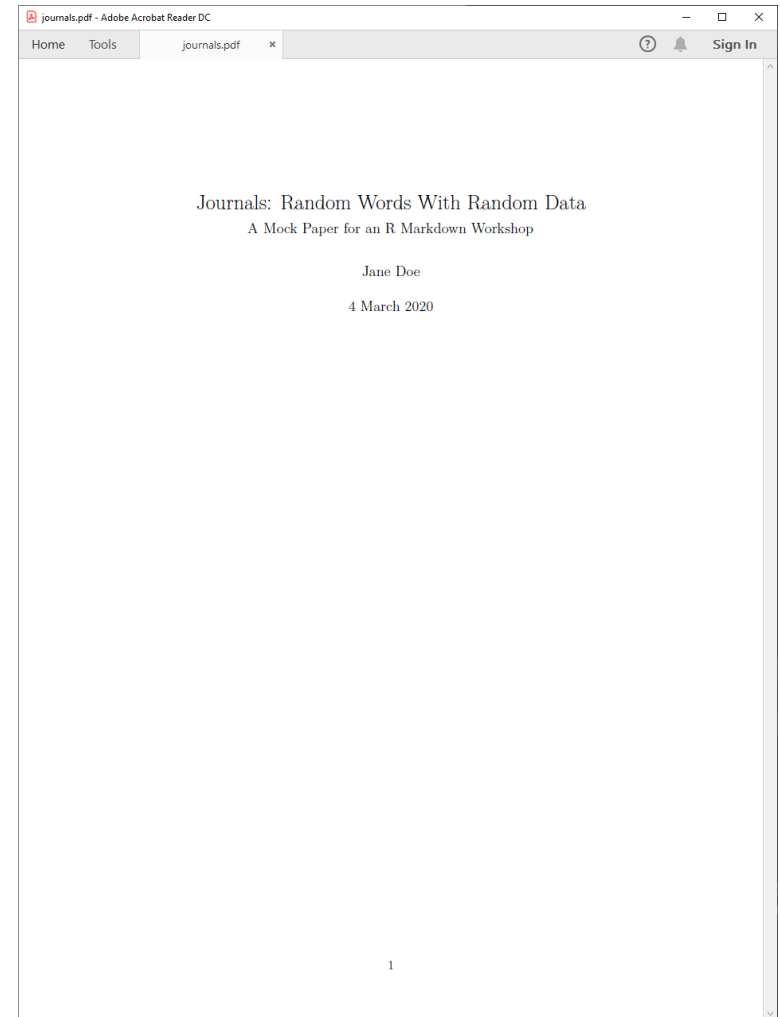
```
---  
title: "Journals: Random Words With Random Data"  
output: pdf_document  
---
```



YAML — Strings

Quotation marks are optional for strings without special characters

```
---  
title: "Journals: Random Words With Random Data"  
subtitle: A Mock Paper for an R Markdown Workshop  
author: Jane Doe  
date: 4 March 2020  
output: pdf_document  
---
```



YAML — Strings

The syntax `[^footnotes_go_here]` adds footnotes to strings

```
---
title: "Journals: Random Words With Random Data^[Prelimi
subtitle: A Mock Paper for an R Markdown Workshop
author: "Jane Doe^[Department of Science, University of
date: 4 March 2020
output: pdf_document
---
```

YAML — Strings

The bibliography and csl variables take strings as well

```
---
title: "Journals: Random Words With Random Data^[The paragraphs in this article are randomly generated]"
subtitle: A Mock Paper for an R Markdown Workshop
author: "Jane Doe^[Department of Science, University of Random. Email: jane.doe@random.edu. Website: random.edu]"
date: 4 May 2020
bibliography: references.bib
csl: apa_7th.csl
output: pdf_document
---
```

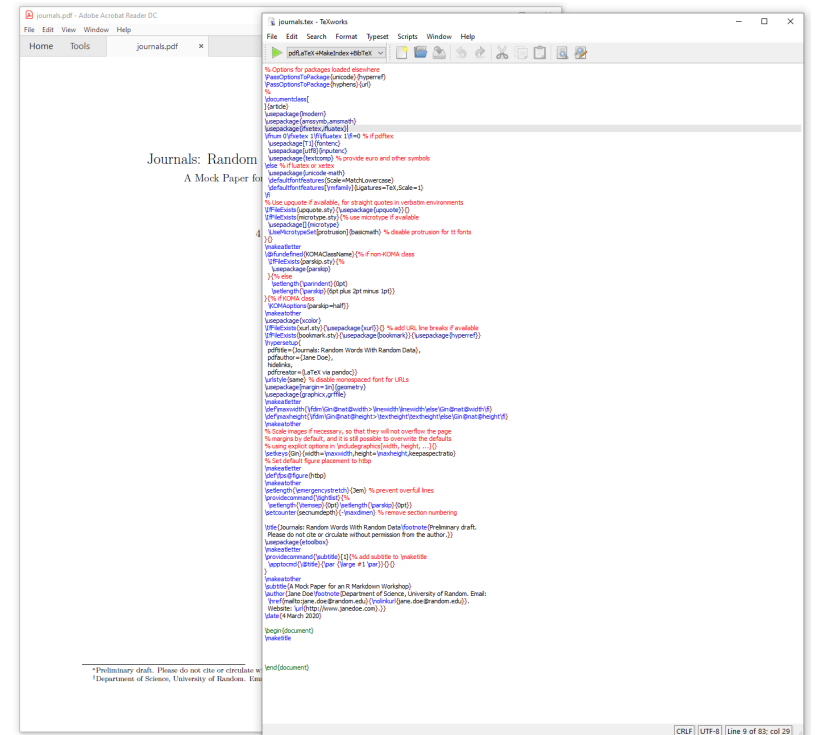
These strings depend on (a) where the files are located and (b) how they are named

```
---
...
bibliography: references/ref_library.bib
csl: "C:/Users/resul/Dropbox/teaching/r_markdown/references/chicago_manual_17.csl"
...
---
```

YAML — Options and Sub-Options

Options can have sub-options

```
---
title: "Journals: Random Words With Random Data^[Preliminary draft]"
subtitle: A Mock Paper for an R Markdown Workshop
author: "Jane Doe^[Department of Science, University of Random]"
date: 4 March 2020
bibliography: references.bib
csl: apa_7th.csl
output:
  pdf_document:
    keep_tex: true
---
```



YAML — Options and Sub-Options

Options can have sub-options

```
---
title: "Journals: Random Words With Random Da
subtitle: A Mock Paper for an R Markdown Work
author: "Jane Doe^[Department of Science, Uni
date: 4 March 2020
bibliography: references.bib
csl: apa_7th.csl
output:
  pdf_document:
    keep_tex: true
---
```

Notice that

- this specific setting, highlighted, will create multiple outputs
 - a LaTeX and a PDF document
- all but the last option (i.e., `true`) takes a colon
- options and sub-options (except the last option, again) are stepwise indented
 - exactly with four spaces
 - the alignment between the colons for `pdf_document` and `keep_tex` is coincidental

YAML — R Code

Variables can take code as well

```
---  
title: "Journals: Random Words With Random Data^[Prelimi  
subtitle: A Mock Paper for an R Markdown Workshop  
author: "Jane Doe^[Department of Science, University of  
date: "r format(Sys.Date(), '%d %B %Y') "  
bibliography: references.bib  
csl: apa_7th.csl  
output: pdf_document  
---
```

YAML — R Code

Variables can take code as well

```
---  
title: "Journals: Random Words With Random Da  
subtitle: A Mock Paper for an R Markdown Work  
author: "Jane Doe^[Department of Science, Uni  
date: "r format(Sys.Date(), '%d %B %Y')"  
bibliography: references.bib  
csl: apa_7th.csl  
output: pdf_document  
---
```

Notice that

- such codes can be particularly useful for variables
 - that need frequent updates
 - and that can be automatically updated
 - e.g., date
- there are quotation marks " around the code
- we'll cover codes in .Rmd documents later on in the workshop

YAML — R Code

Code and text can be combined in a string

```
---  
title: "Journals: Random Words With Random Data^[Prelimi  
subtitle: A Mock Paper for an R Markdown Workshop  
author: "Jane Doe^[Department of Science, University of  
date: "First version: 4 March 2020. This version: r for  
bibliography: references.bib  
csl: apa_7th.csl  
output: pdf_document  
---
```

YAML — Some Further Settings for PDF Outputs

- `fontsize`
 - the default is 10pt
 - the other options are 11pt and 12pt
- `linkcolor, urlcolor, citecolor`
 - the default is the colour of the text
 - the other options are white, red, green, blue, cyan, magenta, yellow
- `link-citations`
 - the default is no
 - the other option is yes — a click on an (part of) citation will take the screen to the relevant entry in the list of references

Part 3 — Exercises

10:00

- 1) Open `journals.Rmd` and fill in the YAML variables for the mock paper
 - take cues from `reproduce_this.pdf` and/or the slides
- 2) Add and set one of the variables mentioned among *Some Further Settings for PDF Outputs*
 - i.e., `fontsize`, `linkcolor`, `urlcolor`, `citecolor`, `link-citations`
- 3) Add and set a completely new variable not covered so far
 - see R Markdown Cheat Sheet
- 4) Knit your `journals.Rmd`
- 5) Consider tidying your working directory
 - you may wish to delete any trial files you have created so far
 - except, perhaps, `journals.pdf`, which we will keep re-creating
 - the original documents are `journals.Rmd`, `references.bib`, and `apa_tth.csl`

Part 4. Writing Text

Syntax — Overview

- R Markdown follows the syntax in Pandoc's Markdown
 - for the complete rules of the syntax, see [Pandoc User's Guide](#)
 - for a useful summary of the syntax, see the R Markdown Cheat Sheet

Syntax — Lines

Multiple spaces on a given line are reduced to one

```
This is a sentence followed by four spaces.    This is another sentence on the same line.
```

This is a sentence followed by four spaces. This is another sentence on the same line.

Line endings with fewer than two spaces are ignored

```
This is a sentence followed by one space.  
This is another sentence on a new line.
```

This is a sentence followed by one space. This is another sentence on a new line.

Syntax — Hard Brakes

Two or more spaces at the end of lines introduce hard brakes, forcing a new line

```
This is a sentence followed by two spaces.  
This is another sentence on a new line.
```

This is a sentence followed by two spaces.
This is another sentence on a new line.

Syntax — Line Blocks

Spaces in lines starting with a vertical line `|` are kept

```
| a one-space indent  
|   a five-space indent  
|     a ten-space indent
```

```
a one-space indent  
  a five-space indent  
    a ten-space indent
```

Syntax — Block Quotes

Lines starting with the grater-than sign `>` introduce block quotes

```
> In God, we trust. All others must bring data.  
>  
> --- Anonymous
```

In God, we trust. All others must bring data.

— Anonymous

Syntax — Paragraphs

One or more* blank lines introduce a new paragraph

This is the first sentence of a paragraph as it is preceded by a blank line. This is the second sentence of that paragraph, which is followed by a blank line.

This is the first sentence of a **new paragraph** as it is preceded by a blank line. This is the second sentence of that paragraph, which is followed by a blank line.

This is the first sentence of a paragraph as it is preceded by a blank line. This is the second sentence of that paragraph, which is followed by a blank line.

This is the first sentence of a *new paragraph* as it is preceded by a blank line. This is the second sentence of that paragraph, which is followed by a blank line.

[*] Multiple blank lines between paragraphs reduced to one

Syntax — Comments

Text with the syntax `<!-- comments -->` is omitted from output

```
<!-- This paragraph needs re-writing -->
```

This is the first sentence of a paragraph as it is preceded by a blank line. This is the second sentence of that paragraph, which is followed by a blank line.

This is the first sentence of a new paragraph *<!-- I've removed italics -->* as it is preceded by a blank line. This is the second sentence of that paragraph, which is followed by a blank line.

This is the first sentence of a paragraph as it is preceded by a blank line. This is the second sentence of that paragraph, which is followed by a blank line.

This is the first sentence of a new paragraph as it is preceded by a blank line. This is the second sentence of that paragraph, which is followed by a blank line.

Part 4 — Exercises

03:00

6) Hard Brakes

- see `reproduce_this.pdf`: page 1
- apply in `journals.Rmd`: paragraph 1

7) Line Blocks / Block Quotes

- see `reproduce_this.pdf`: page 1
- apply in `journals.Rmd`: block quote, between paragraphs 1 and 2

Syntax — Headers

The number sign `#` introduces headers; lower levels are created with additional signs — up to total five levels

```
# Introduction becomes
```

Introduction

```
## 1. Introduction becomes
```

1. Introduction

```
### 3.1 Introduction becomes
```

3.1 Introduction

```
#### Introduction becomes
```

Introduction

```
##### Introduction becomes
```

Introduction

Syntax — Emphases

A pair of single asterics `*` or underscores `_` introduces italics

`*italics*` becomes *italics*

`_italics_` becomes *italics* as well

A pair of double asterics or underscores introduces bold

`**bold**` becomes **bold**

`__bold__` becomes **bold** as well

These two syntaxes can be combined

`**_bolditalics_**` becomes ***boldsitalics***

`_**bolditalics**_` becomes ***bolditalics*** as well

Syntax — Strikethrough

A pair of double tildes ~ introduces strikethrough

~~strikethrough~~ becomes ~~strikethrough~~

Strikethrough can be combined with italics or bold

***~strikebold~~** or __~~strikebold~~__, they both become ~~**strikebold**~~

~~**strikebold*** or ~~__strikebold__~~, they both become ~~**strikebold**~~ as well

~~~strikeitalitcs~~~ or \_~~strikeitalitcs~~\_, they both become ~~*strikeitalitcs*~~

~~\*strikeitalitcs\*~~ or ~~\_strikeitalitcs\_~~, they both become ~~*strikeitalitcs*~~ as well

# Part 4 — Exercises

03:00

## 8) Headers

- see `reproduce_this.pdf`: pages 1 to 11
  - 11 headers, Abstract to References
- apply in `journals.Rmd`

## 9) Emphases

- see `reproduce_this.pdf`: pages 1 and 2
  - bold and italics
- apply in `journals.Rmd`: paragraph 2

# Syntax — Links — Internal\*

You can link text to section headers in the same document

[Conclusion] (#conclusion) becomes **Conclusion**, and a click takes the screen to that section

Multi-word headers need hyphenation

[Literature Review] (#literature-review) becomes **Literature Review**, and it works only if the second part is hyphenated

[\*] We will cover links to references, figures, and tables later on.

# Syntax — Links — External

You can link text to URLs

`[click here](https://resulunit.com/)` becomes [click here](https://resulunit.com/)

`<https://resulunit.com>` becomes <https://resulunit.com>

`[https://resulunit.com](https://resulunit.com/)` becomes <https://resulunit.com> as well

You can also link text to email addresses

`[email me](mailto:resul.umat@unilu.ch)` becomes [email me](mailto:resul.umat@unilu.ch)\*

`<resul.umat@unilu.ch>` becomes [resul.umat@unilu.ch](mailto:resul.umat@unilu.ch)

[\*] Notice the prefix `mailto:` in the second part

# Part 4 — Exercises

03:00

## 10) Links — Internal

- see `reproduce_this.pdf`: page 2
  - the link to the Literature Review section
- apply in `journals.Rmd`: paragraph 4

## 11) Links — External

- see `reproduce_this.pdf`: page 1
  - email and website links in one of the footnotes
- apply in `journals.Rmd`: title page items

# Syntax — Equations

Inline equations go between a pair of single dollar signs `$` — with no space between the signs and the equation itself

`$E = mc^{2}$` becomes  $E = mc^2$

Block equations go in between a pair of double dollar signs — with or without spaces, it works

`$$ E = mc^{2} $$` becomes

$$E = mc^2$$

`$$E = mc_{2} $$` becomes

$$E = mc_2$$

# Syntax — Footnotes — Inline Notes

For inline footnotes, use the `^[footnote]` syntax

Jane Doe<sup>^[Corresponding author.]</sup> becomes Jane Doe<sup>1</sup>

Notice that

- the caret sign `^` comes *before* the left square bracket `[`
- this syntax works in YAML as well as in text
  - footnotes in YAML get symbols, in text they get numbers

[1] Corresponding author.

# Syntax — Footnotes — Notes with Identifiers

An alternative is to use the `[^identifier]` syntax, with identifiers defined elsewhere in the same document

```
Dr Doe holds a PhD in rock science.[^defence_date]  
  
[^defence_date]: She defended her thesis in 2017.
```

Dr Doe holds a PhD in rock science.<sup>1</sup>

Notice that

- the caret sign comes *after* the left square bracket
- this syntax works in text, but not in YAML

[1] She defended her thesis in 2017.



# Part 4 — Exercises

03:00

## 12) Equations

- see `reproduce_this.pdf`: page 7
- apply in `journals.Rmd`: paragraph 22; block equation, between paragraphs 22 and 23

## 13) Footnotes

- see `reproduce_this.pdf`: page 2
- apply in `journals.Rmd`: paragraph 3

# Syntax — Lists

Lines starting with asterics `*` as well as plus `+` or minus `-` signs introduce lists

```
- books  
- articles  
- reports
```

- books
- articles
- reports

# Syntax — Lists — Nesting

Lists can be nested within each other, with indentation

```
+ books
+ articles
  - published
  - under review
    + revised and resubmitted
  - work in progress
```

- books
- articles
  - published
  - under review
    - revised and resubmitted
  - work in progress

# Syntax — Lists — Numbering

List items can be numbered

- ```
1. books
2. articles
  - published
  - under review
    + revised and resubmitted
  - work in progress
```

- ```
1. books
2. articles
  ◦ published
  ◦ under review
    ▪ revised and resubmitted
  ◦ work in progress
```

# Syntax — Dashes

Two hyphens grouped together introduce an en-dash

-- becomes —

Three hyphens grouped together introduce an em-dash

--- becomes —

# Syntax — Sub- and Super-scripts

A pair of tildes introduces subscript

`CO~2~` becomes **CO<sub>2</sub>**

A pair of carets introduces subscript

`R^2^` becomes **R<sup>2</sup>**

# Syntax — Sub- and Super-scripts

A pair of tildes introduces subscript

`CO~2~` becomes **CO<sub>2</sub>**

A pair of carets introduces subscript

`R^2^` becomes **R<sup>2</sup>**

Notice that

- the syntax here (Markdown-based) is different than the one for equations (LaTeX-based)
  - e.g., `R^2^` versus `mc^{2}`

# Part 4 — Exercises

03:00

## 14) Lists

- see `reproduce_this.pdf`: page 3
- apply in `journals.Rmd`: list, between paragraphs 10 and 11

## 15) Dashes

- see `reproduce_this.pdf`: page 2
- apply in `journals.Rmd`: paragraph 6

## 16) Sub- and Super-scripts

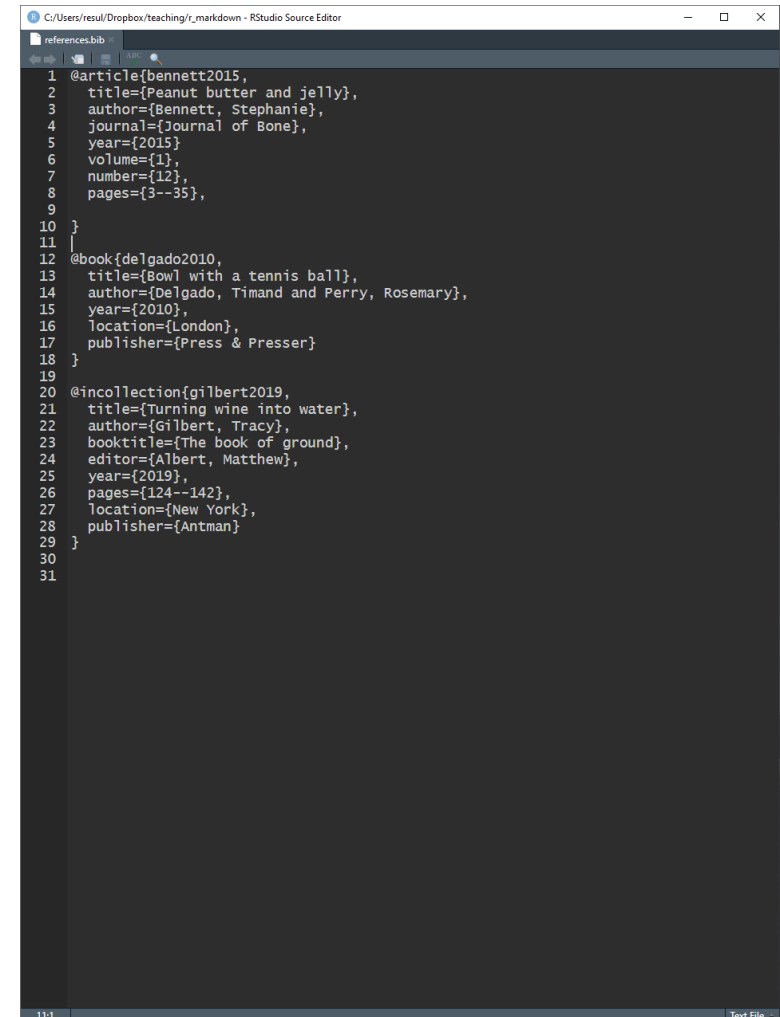
- see `reproduce_this.pdf`: page 2
- apply in `journals.Rmd`: paragraph 5



# Part 5. Managing References

# References — Bibliography Database

- References are defined in .bib files
  - they follow the BibTeX format
- pandoc looks for a .bib file, and for the definitions therein, to process citations
  - .bib files are specified with the bibliography variable in YAML
- pandoc can process a citation only if there is a linked entry in the .bib file
  - but not all entries have to be cited



```
1 @article{bennett2015,  
2   title={Peanut butter and jelly},  
3   author={Bennett, Stephanie},  
4   journal={Journal of Bone},  
5   year={2015}  
6   volume={1},  
7   number={12},  
8   pages={3--35},  
9  
10  }  
11 |  
12 @book{delgado2010,  
13   title={Bowl with a tennis ball},  
14   author={Delgado, Timand and Perry, Rosemary},  
15   year={2010},  
16   location={London},  
17   publisher={Press & Presser}  
18  }  
19  
20 @incollection{gilbert2019,  
21   title={Turning wine into water},  
22   author={Gilbert, Tracy},  
23   booktitle={The book of ground},  
24   editor={Albert, Matthew},  
25   year={2019},  
26   pages={124--142},  
27   location={New York},  
28   publisher={Antman}  
29  }  
30  
31
```

# References — Bibliography Database — Entries

- A BibTeX entry consists of three elements
  - a type
    - e.g., @article
  - a citation-key
    - e.g., bennett2015
  - a number of tags
    - e.g., title, author
- Different tags are available for different reference types
  - some tags are required, others are optional

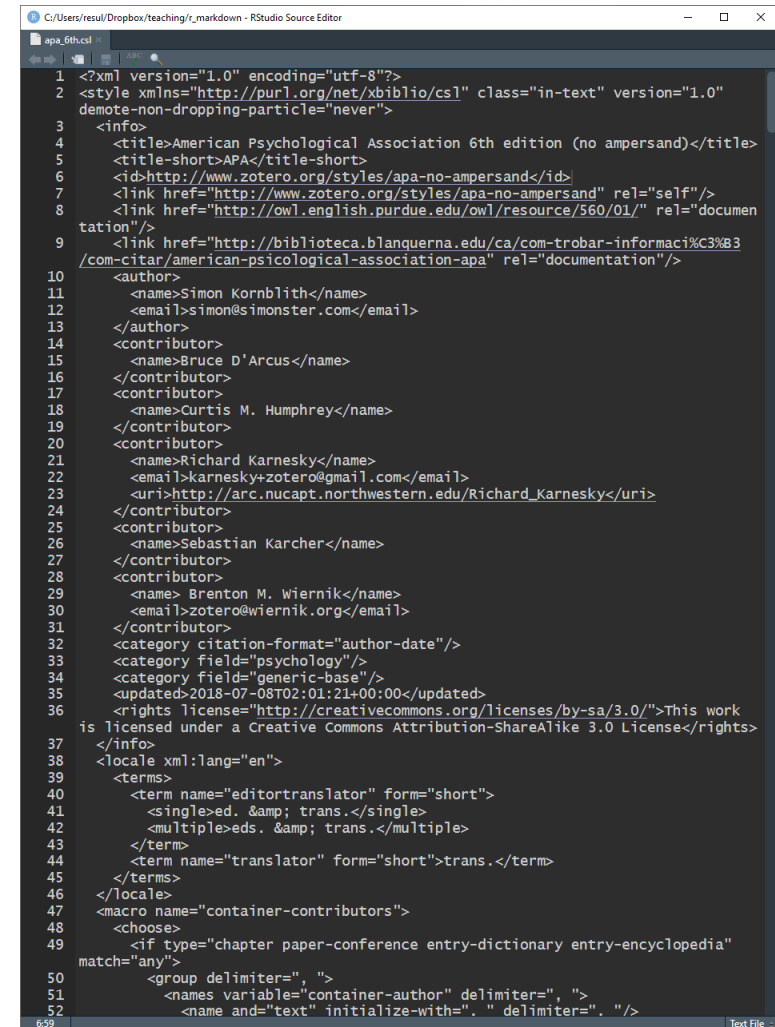
```
1 @article{bennett2015,  
2   title={Peanut butter and jelly},  
3   author={Bennett, Stephanie},  
4   journal={Journal of Bone},  
5   year={2015}  
6   volume={1},  
7   number={12},  
8   pages={3--35},  
9  
10  }  
11
```

# References — Bibliography Database — Entries

- One could create entries by hand
  - this requires knowing the BibTeX format, entry types, tags, and related information about references to be cited
    - this is not efficient
- A good alternative Google Scholar, which provides BibTeX entries
  - follow `cite` -> BibTeX and copy
  - paste into `.bib`, edit if necessary, and save
- Some publishers and journals provide BibTeX entries on their website as well

# References — Style

- Reference styles are defined in .csl files
  - files for different styles are available at <https://www.zotero.org/styles>
- pandoc looks for a .csl file, and for the styles therein, to style citations and references
  - .csl files are specified with the `csl` variable in YAML
  - if unspecified, it uses a Chicago author-date format
- .csl files affect the style only in outputs
  - no matter which the style is used, the citation syntax in .Rmd documents is the same



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <style xmlns="http://purl.org/net/xbiblio/csl" class="in-text" version="1.0"
  demote-non-dropping-particle="never">
3   <info>
4     <title>American Psychological Association 6th edition (no ampersand)</title>
5     <title-short>APA</title-short>
6     <id>http://www.zotero.org/styles/apa-no-ampersand</id>
7     <link href="http://www.zotero.org/styles/apa-no-ampersand" rel="self"/>
8     <link href="http://owl.english.purdue.edu/owl/resource/560/01/" rel="documenta
  tion"/>
9     <link href="http://biblioteca.blanquerna.edu/ca/com-trobar-informaci%C3%B3
    /com-citar/american-psicological-association-apa" rel="documentation"/>
10    <author>
11      <name>Simon Kornblith</name>
12      <email>simon@simonster.com</email>
13    </author>
14    <contributor>
15      <name>Bruce D'Arcus</name>
16    </contributor>
17    <contributor>
18      <name>Curtis M. Humphrey</name>
19    </contributor>
20    <contributor>
21      <name>Richard Karnesky</name>
22      <email>karnesky+zotero@gmail.com</email>
23      <uri>http://arc.nucapt.northwestern.edu/Richard_Karnesky</uri>
24    </contributor>
25    <contributor>
26      <name>Sebastian Karcher</name>
27    </contributor>
28    <contributor>
29      <name>Brenton M. Wiernik</name>
30      <email>zotero@wiernik.org</email>
31    </contributor>
32    <category citation-format="author-date"/>
33    <category field="psychology"/>
34    <category field="generic-base"/>
35    <updated>2018-07-08T02:01:21+00:00</updated>
36    <rights license="http://creativecommons.org/licenses/by-sa/3.0/">This work
  is licensed under a Creative Commons Attribution-ShareAlike 3.0 License</rights>
37  </info>
38  <locale xml:lang="en">
39    <terms>
40      <term name="editortranslator" form="short">
41        <single>ed. & trans.</single>
42        <multiple>eds. & trans.</multiple>
43      </term>
44      <term name="translator" form="short">trans.</term>
45    </terms>
46  </locale>
47  <macro name="container-contributors">
48    <choose>
49      <if type="chapter paper-conference entry-dictionary entry-encyclopedia"
  match="any">
50        <group delimiter=" ">
51          <names variable="container-author" delimiter=" ">
52            <name and="text" initialize-with="." delimiter="." />
```

# References — In-text Citation Syntax

All citations keys take the 'at' sign @, square brackets and/or minus signs introduce variation

[@bennett2015] becomes (Bennett, 2015)

@bennett2015 becomes Bennett (2015)

[-@bennett2015] becomes (2015)

-@bennett2015 becomes 2015

[@bennett2015 35] becomes (Bennett, 2015, p. 35)

[@bennett2015 33–35] becomes (Bennett, 2015, pp. 33–35)

[@bennett2015, ch. 1] becomes (Bennett, 2015, ch. 1)

[@bennett2015; @gilbert2019] becomes (Bennett, 2015; Gilbert, 2019)

[see @bennett2015, for details] becomes (see Bennett, 2015, for details)

@bennett2015 [33–35] becomes Bennett (2015, pp. 33–35)

# Citations — Reference List

The list of references is inserted after the last line of the document, with no section header.

- ending your .Rmd documents with a header for the list of references

```
This is the last sentence of the manuscript.
```

```
## References
```

This is the last sentence of the manuscript.

## References

Bennett, S. (2015). Peanut butter and jelly. *Journal of Bone*, 1(12), 3–35.

Gilbert, T. (2019). Turning wine into water. In M. Albert (Ed.), *The book of ground* (pp. 124–142). New York: Antman.

# References — Internal Links

For internal links from in-text citations to the reference list, set the YAML variable `link-citations` to `yes`

- a click on these links take the screen to the relevant entry in the list
- the `linkcolor` variable make these links explicit
  - this is not necessary for the links to work

```
---  
...  
bibliography: references.bib  
csl: apa_7th.csl  
link-citations: yes  
linkcolor: blue  
...  
---
```



## Part 5 — Exercises

05 : 00

17) Add an entry to `references.bib` for the following book

- *R Markdown: The Definitive Guide* by Xie and co-authors

18) Reproduce the citations and reference list in the mock paper

- see `reproduce_this.pdf`: pages 3 and 11
- apply in `journals.Rmd`: paragraph 7 to 9

19) Change reference style

- download your `.csl` file for your favourite style from <https://www.zotero.org/styles>
- update the YAML variable

20) Link the citations to the reference list

## Part 6. Adding Code, Figures, and Tables

# Code, in and outside Chunks

# Code — Overview

Most codes go inside code chunks

- e.g., code that imports and cleans data, and/or produces tables and/or figures

```
```${r}

df <- read.csv("rmd_workshop_files/images_data/journals.csv") %>%
  mutate(age = 2020 - since,
         english = factor(english),
         subfield = factor(subfield))

```
```

Codes can also go in line with text

- .e.g., code that result in a single statistic

```
The average H5 Index for the journals in the dataset is `r mean(df$h5_index)`.
```

# Code Chunks — Overview

- Code chunks are delimited spaces between a pair of three backticks
  - below is an empty chunk for R code

```
```${r}  
```
```

- `r` is an option of the chunk, indicating that the code in the chunk above should be run by R
  - it could have been `python`, which we will not cover in this workshop
  - options go in a pair of curly brackets, on the same line with the first delimiter
- Chunks can be placed anywhere in `.Rmd` documents
  - their output, if there is any, might float around text to avoid breaking across pages

# Code Chunks — Labels

It is recommended to label the code chunks, which are otherwise automatically numbered

- informative labels can be helpful for navigating through error messages as well as filenames of plots and cache
  - in the example below, the chunk is labelled as `data_import`
- but note that duplicate labels lead to an error during compilation

```
```${r, data_import}  
df <- read_csv("data/journals.csv")  
```
```

# Code Chunks — Options

- Code chunks take options, listed on the same line with the first delimiter, in curly brackets
  - avoid spaces around the equal sign `=` between option tags and values
    - such spaces might lead to errors
  - in the example below, the chunk is labelled as `setup`, and the `include` option is set to `FALSE`
    - with this option and value, nothing from this chunk will be included in the output document

```
```${r, setup, include=FALSE}  
```
```

- The list of options is available at <https://yihui.org/knitr/options>
  - R Markdown Cheat Sheet provides a helpful list as well

# Code Chunks — Options — Defaults

Options have default values

- e.g., for `echo`, the default is `TRUE`
  - `echo`: should the source code printed in the output? — yes
- therefore the following two chunks have the same function

```
```${r}  
```
```

```
```${r, echo=TRUE}  
```
```



# Code Chunks — Options — Defaults

This chunk prints two things in the output document — (a) the code and (b) the head of the data frame

```
```{r}  
head(df)  
```
```

```
head(df)
```

```
##           name  origin  branch h5_index h5_median  
## 1 Journal of Bears Americas Physical      73        97  
## 2 Journal of Moon      Asia  Social      72       106  
## 3 Journal of Lumber Americas Physical      72       100  
## 4 Journal of Houses  Europe  Social      72       102  
## 5 Journal of Water  Europe  Social      70       100  
## 6 Journal of Jeans Americas Physical      69       101  
## english subfield issues age  
## 1      1      1      7  61
```

# Code Chunks — Options

Setting `echo=FALSE` prevents the code from being displayed in the output document

```
```{r, echo=FALSE}  
head(df)  
```
```

This chunk therefore prints one thing in the output document — the head of the data frame

```
##           name    origin  branch h5_index h5_median  
## 1 Journal of Bears Americas Physical      73        97  
## 2 Journal of Moon      Asia   Social      72       106  
## 3 Journal of Lumber Americas Physical      72       100  
## 4 Journal of Houses  Europe   Social      72       102  
## 5 Journal of Water   Europe   Social      70       100  
## 6 Journal of Jeans  Americas Physical      69       101  
##  english subfield issues age  
## 1         1         1      7  61  
## 2         1         0      6  64  
## 3         1         1      8  30  
## 4         1         0      8  38
```

# Code Chunks — Other Useful Options

Prevent the result(s) of the source code from being displayed in the output document

```
```{r, results="hide"}  
head(df)  
```
```

This chunk therefore prints one thing in the output document — the source code

```
head(df)
```

Setting `results="asis"` passes the results as they are produced by the code. In creating tables for PDF output with the `stargazer` package,\* it is a must.

# Code Chunks — Other Useful Options

Cache results for future compilations

```
```${r ... cache=TRUE}  
```
```

Prevent R from running the code in the chunk altogether

```
```${r ... eval=FALSE}  
```
```

Prevent messages and/or warnings from being displayed in the output

```
```${r ... error=FALSE, message=FALSE, warning=FALSE}  
```
```

# Code Chunks — Other Useful Options

Define the *actual* dimensions of figures, in inches

```
```{r ... fig.height=6, fig.width=9}  
```
```

Define the size of figures *in the output document*, with `out.width` and/or `out.height`

```
```{r ... out.width="50%"}  
```
```

Define the alignment of figures — `left`, `right`, or `center`

```
```{r ... fig.align="center"}  
```
```

# Code Chunks — Other Useful Options

Define captions for figures

```
```{r ... fig.caption="A Scatter Plot"}  
```
```

Set the resolution for figures

```
```{r ... dpi=300}  
```
```

Set extra options, such as angle, that output format would accept for figures

```
```{r ... out.extra="angle=45"}  
```
```

# Code Chunks — The Setup Chunk

It is recommended to use the first code chunk for general setup, where you can

- define *your own defaults* for chunk options, with `knitr::opts_chunk$set()`
  - avoids repeating chunk options
- load the necessary packages
- import raw data

```
`` `{r, setup, include=FALSE}  
# chunk option defaults  
knitr::opts_chunk$set(echo=FALSE, message=FALSE)  
  
# packages  
library(dplyr)  
library(ggplot2)  
library(stargazer)  
  
# data  
df_raw <- read.csv("journals.csv")  
`` `
```

# Code Chunks — The Data Chunk

I recommend using the second chunk for the main operations\* on raw data

- e.g., for data cleaning and other transformations
- some minor transformations could be left to lower chunks
  - e.g., capitalizing variable names for figures

```
`r, data, echo=FALSE ...}  
  
df <- df_raw %>%  
  ...  
  
`r
```

[\*] I will be using the pipe operator %>% and other functions from the dplyr package for such operations in the following slides.



# Code Chunks — The Data Chunk

Transform `subfield` and `english` into factor variables

- despite being numeric — i.e., 0s and 1s — they are meant to be categorical variables

```
`r, data, echo=FALSE ...}  
  
df <- df_raw %>%  
  mutate(subfield = as.factor(subfield),  
         english = as.factor(english))  
  
`r, ...`
```

# Code Chunks — The Data Chunk

Create a new variable `age`, based on the existing variable `since`, to be used in regression models later on

```
`r, data, echo=FALSE ...}  
  
df <- df_raw %>%  
  mutate(subfield = as.factor(subfield),  
         english = as.factor(english),  
         age = 2020 - since)  
  
...
```

Drop the `since` variable, so that it won't appear in our summary statistics table later on

```
`r, data, echo=FALSE ...}  
  
df <- df_raw %>%  
  mutate(subfield = as.factor(subfield),  
         english = as.factor(english),  
         age = 2020 - since) %>%  
  select(-since)  
  
...
```

# Inline Code — Overview

Code can also be incorporated in text, with the ``r`` syntax

- unline chunks, these do not take options
- the output document will display the result of the code
  - in the exact place of the source code
- the result of the code will have the same formatting with the text

# Inline Code — Examples

If we multiply `_pi_` by `5`, we get ``r pi * 5``.

If we multiply *pi* by 5, we get 15.7079633.

The average H5 Index **for** the journals **in** the dataset is ``r mean(df$h5_index)``, which would round to ``r round(mean(df$h5_index), digits = 1)``.

The average H5 Index for the journals in the dataset is 26.3611366, which would round to 26.4.

`__Only `r nrow(subset(df, english == 0))` journals__` **in** the dataset are published **in** a language other than English.

**Only 113 journals** in the dataset are published in a language other than English.

# Part 6 — Exercises

07:30

## 21) Introduce a setup chunk

- define one or more defaults chunk options, with `knitr::opts_chunk$set()`
- load the packages that we will need — `dplyr`, `ggplot2`, and `stargazer`
- import raw data

## 22) Introduce a data chunk

- transform `subfield` and `english` into factors
- create a new variable `age`, based on `since`
- drop `since` from the data frame

## 23) Inline Code

- see `reproduce_this.pdf`: page 6
  - i.e., 1091 observations
- apply in `journals.Rmd`: paragraph 21
  - hint: use the `nrow` function

# Figures

# Figures — Images — Markdown Syntax

The syntax `![Figure Caption](figure.extension)` embeds images, and/or figures produced elsewhere,<sup>\*</sup> into .Rmd documents

- similar to the link syntax, only this time it is preceded by an exclamation mark **!**
- goes outside code chunks, on a new line
- simple, but not very customisable

[\*] Ideally, reproducible papers should produce their own images with data and code. However, there might be situations where this is not possible.

# Figures — Images — Markdown Syntax

```
![A screenshot of the Google Scholar homepage](../image/google_scholar.png)
```

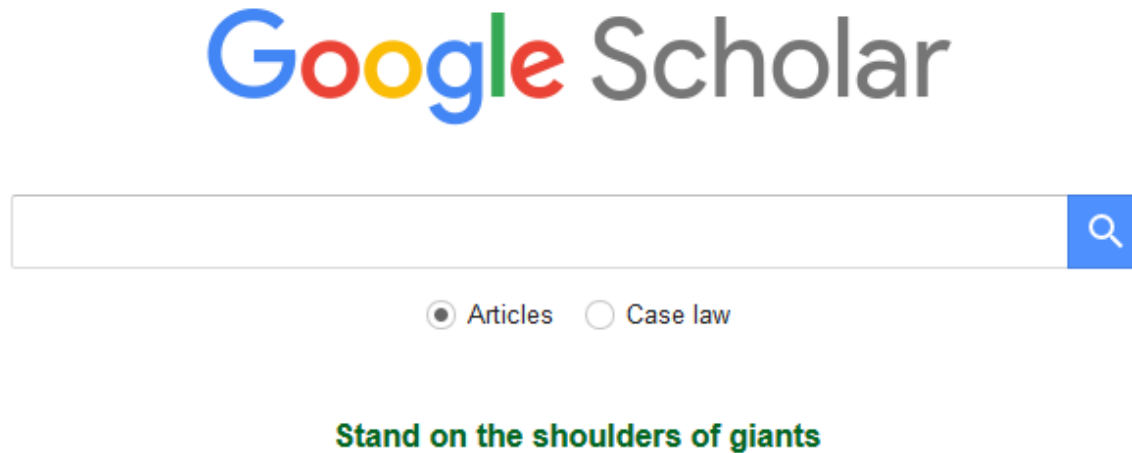


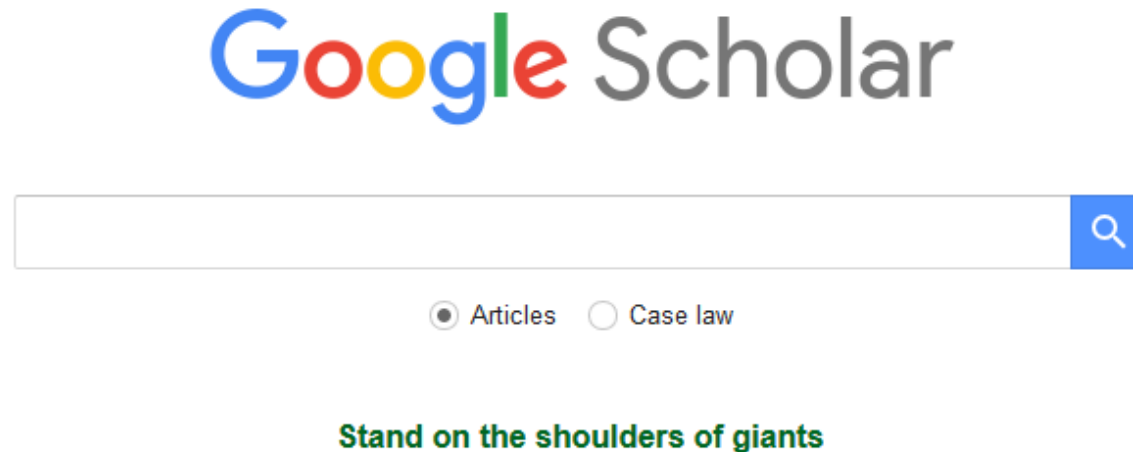
Figure 1: A screenshot of the Google Scholar homepage.



# Figures — Images — Markdown Syntax

Figures are numbered automatically

```
![A screenshot of the Google Scholar homepage](../image/google_scholar.png)
```



**Figure 1:** A screenshot of the Google Scholar homepage.

# Figures — Images — Markdown Syntax

The syntax can accept width or height attributes as follows

```
[A screenshot of the Google Scholar homepage](../image/google_scholar.png) { width=40% }
```

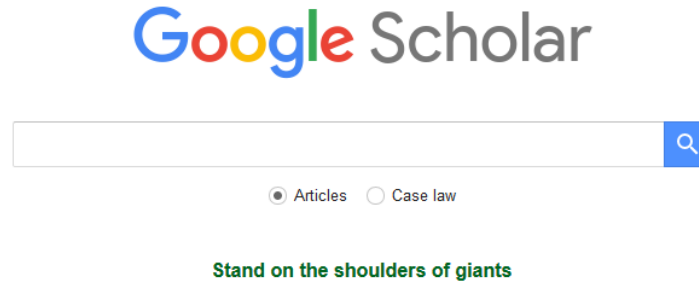


Figure 1: A screenshot of the Google Scholar homepage.

# Figures — Images — knitr

The knitr package offers a capable alternative with the `include_graphics()` function

- this goes inside code chunks
  - use the function with the double-colon operator `::`
    - e.g., `knitr::include_graphics("figure.extension")`
- this is more customisable, through the use of code chunks
  - size is defined with the `out.width` or `out.hight` options
    - rather than `fig.height` and/or `fig.width`

# Figures — Images — knitr

The knitr package offers a capable alternative with the `include_graphics()` function

```
```${r, screenshot, echo=FALSE, fig.cap="A screenshot of the Google Scholar homepage."}  
knitr::include_graphics("../figures/google_scholar.png")  
```
```

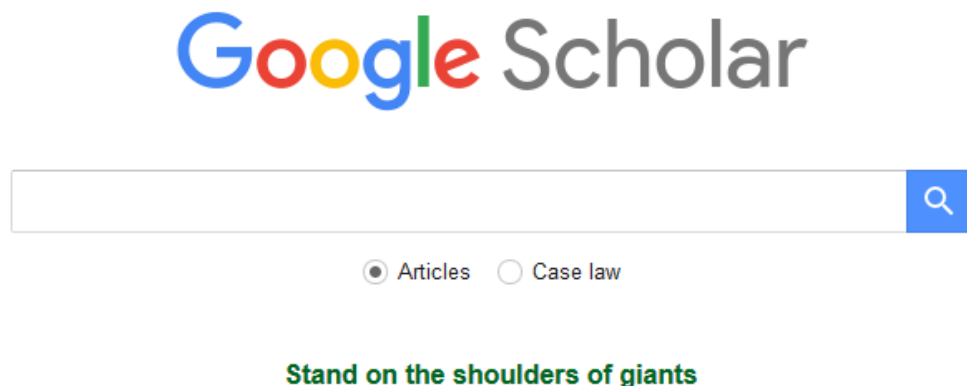


Figure 1: A screenshot of the Google Scholar homepage.

# Figures — Images — knitr

Size is defined with the chunk options `out.width` or `out.height`

```
```\r ... out.width="40%"}  
knitr::include_graphics("../figures/google_scholar.png")  
```
```

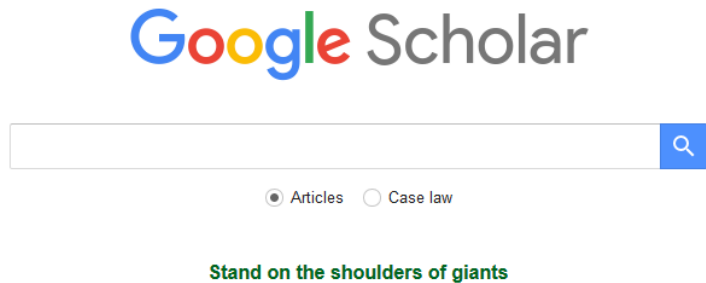


Figure 1: A screenshot of the Google Scholar homepage.

# Figures — Images — knitr

Most other chunk options are common with figures plotted within R Markdown, such as `fig.align`

```
```\r ... fig.align="center"\nknitr::include_graphics("../figures/google_scholar.png")\n```
```

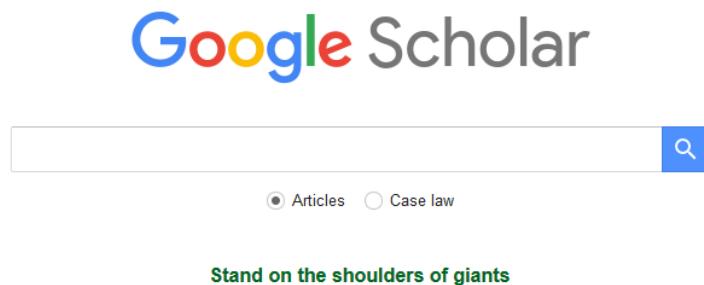


Figure 1: A screenshot of the Google Scholar homepage.

# Figures — ggplot2 — Overview

- A powerful package for visualising data
- Used widely, not only by academics, but also by large corporations such as the New York Times
- A huge amount is written on this package. See, for example,
  - the [package documentation](#)
  - this [book](#) by its creator Hadley Wickham
  - this [reference page](#)
  - this [webinar](#) by one of its authors, Thomas Lin Pedersen
  - these [further extensions](#), maintained by the ggplot2 community
- Among its alternatives are the base and plotly packages

# Figures — ggplot2 — Basics

## 1) The `ggplot` function and the data argument

- specify a data frame in the main `ggplot` function

```
ggplot(data = df)
```

## 2) The mapping aesthetics, or *aes*; most importantly, the variable(s) that we want to plot

- specify as an additional argument in the same `ggplot` function

```
ggplot(data = df, mapping = aes(x = h5_median, y = h5_index, color = subfield))
```

## 3) The geometric objects, or *geom*; the visual representations

- specify, after a plus sign `+`, as an additional function

```
ggplot(data = df, mapping = aes(x = h5_median, y = h5_index, color = subfield)) +  
  geom_point()
```



# Figures — ggplot2

Put the code in a chunk, and give it a caption

```
```{r, scatterplot, fig.cap = "A scatterplot"}  
ggplot(data = df, mapping = aes(x = h5_median,  
  geom_point()  
```\n
```

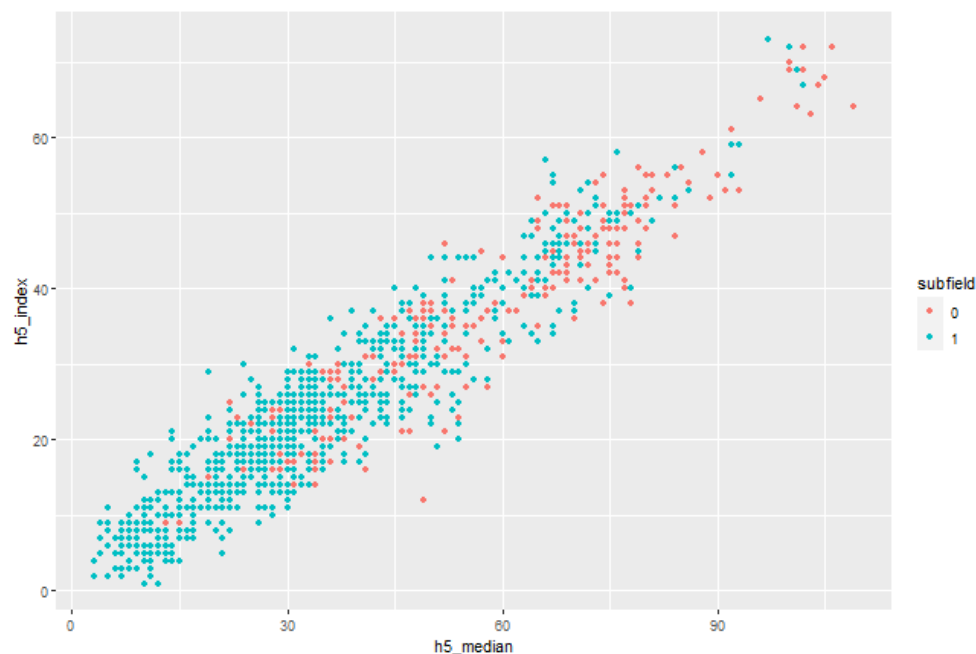


Figure 1. A scatterplot of journal metrics.

# Figures — ggplot2

Add facets for subgroups, e.g., branch

```
```{r, scatterplot, fig.cap = "A scatterplot"}
ggplot(data = df, mapping = aes(x = h5_median,
                                y = h5_index)) +
  facet_wrap(. ~ branch)
```
```

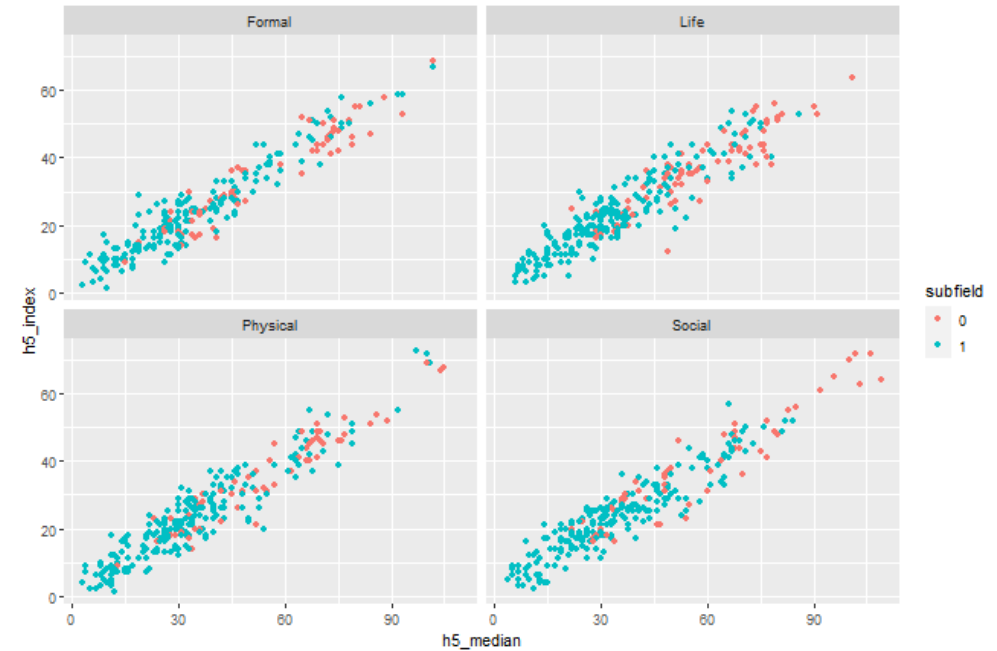


Figure 1. A scatterplot of journal metrics.

# Figures — ggplot2

Scale the colour to improve the legend

```
```{r, scatterplot, fig.cap = "A scatterplot  
ggplot(data = df, mapping = aes(x = h5_median  
  geom_point() +  
  facet_wrap(. ~ branch) +  
  scale_colour_discrete(name = "Journal  
```
```



Figure 1. A scatterplot of journal metrics.

# Figures — ggplot2

Change the theme

```
```{r, scatterplot, fig.cap = "A scatterplot"}
ggplot(data = df, mapping = aes(x = h5_median,
  geom_point() +
  facet_wrap(. ~ branch) +
  scale_colour_discrete(name = "Journal
  theme_bw()

```\n
```

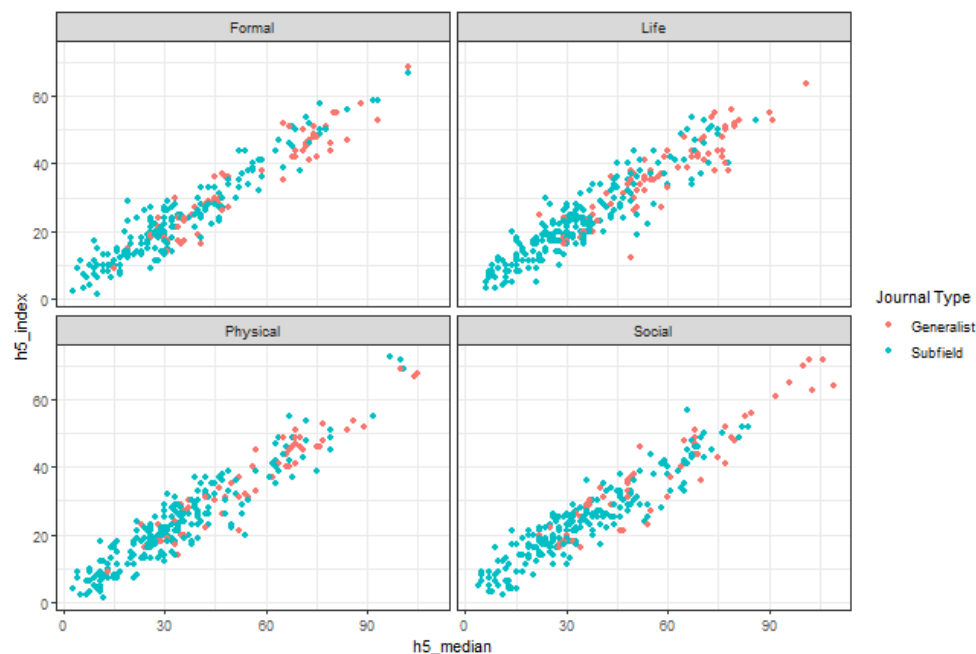


Figure 1. A scatterplot of journal metrics.

# Figures — ggplot2

Improve the axis labels, e.g., with capital first letters

```
```{r, scatterplot, fig.cap = "A scatterplot"}
ggplot(data = df, mapping = aes(x = h5_median,
  geom_point() +
  facet_wrap(. ~ branch) +
  scale_colour_discrete(name = "Journal
  theme_bw() +
  labs(x = "H5 Median", y = "H5 Index")
```
```

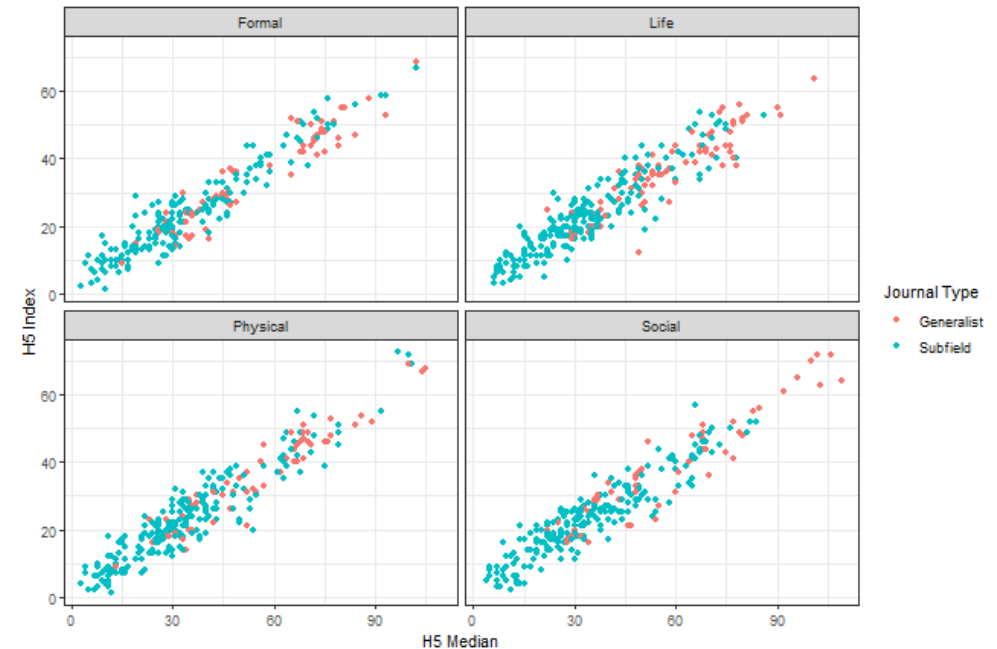


Figure 1. A scatterplot of journal metrics.

# Figures — ggplot2 — Notes

`geom_point` is one of many geoms available

- see this <https://ggplot2.tidyverse.org/reference> for other options, including
  - `geom_bar` for bar charts
  - `geom_boxplot` for box and whiskers plots

# Tables

# Tables — Markdown Syntax

The following syntax, outside code chunks, introduces tables that pandoc can recognise

| First Column | Second Column |
|--------------|---------------|
| First cell   | First cell    |
| Second cell  | Second cell   |
| Third cell   | Third cell    |

| First Column | Second Column |
|--------------|---------------|
| First cell   | First cell    |
| Second cell  | Second cell   |
| Third cell   | Third cell    |



# Tables — Markdown Syntax

The position of headers, relative to their line underneath, defines column alignments

| Left-Aligned | Centered    |
|--------------|-------------|
| -----        | -----       |
| First cell   | First cell  |
| Second cell  | Second cell |
| Third cell   | Third cell  |

| Left-Aligned | Centered    |
|--------------|-------------|
| First cell   | First cell  |
| Second cell  | Second cell |
| Third cell   | Third cell  |

# Tables — Markdown Syntax

A line starting with a colon, placed before or after tables, introduces captions

| Centered                            | Right-Aligned |
|-------------------------------------|---------------|
| -----                               | -----         |
| First cell                          | First cell    |
| Second cell                         | Second cell   |
| Third cell                          | Third cell    |
| : A hand-made table with R Markdown |               |

Table 1: A hand-made table with R Markdown

| Centered    | Right-Aligned |
|-------------|---------------|
| First cell  | First cell    |
| Second cell | Second cell   |
| Third cell  | Third cell    |

# Tables — Markdown Syntax

The caption line itself needs to be surrounded by empty lines

| Centered                            | Right-Aligned |
|-------------------------------------|---------------|
| -----                               | -----         |
| First cell                          | First cell    |
| Second cell                         | Second cell   |
| Third cell                          | Third cell    |
|                                     |               |
| : A hand-made table with R Markdown |               |
|                                     |               |

Table 1: A hand-made table with R Markdown

| Centered    | Right-Aligned |
|-------------|---------------|
| First cell  | First cell    |
| Second cell | Second cell   |
| Third cell  | Third cell    |

# Tables — Markdown Syntax

Tables are numbered automatically

```
: A hand-made table with R Markdown
```

| Centered    | Right-Aligned |
|-------------|---------------|
| -----       | -----         |
| First cell  | First cell    |
| Second cell | Second cell   |
| Third cell  | Third cell    |

**Table 1:** A hand-made table with R Markdown

| Centered    | Right-Aligned |
|-------------|---------------|
| First cell  | First cell    |
| Second cell | Second cell   |
| Third cell  | Third cell    |

# Tables — Markdown Syntax

Grid tables, with the following syntax, can handle complex cells with multiple lines and/or lists

```
+-----+-----+
| First Column | Second Column |
+=====+=====+
| - First item | First cell    |
| - Second item|               |
| - Third item |               |
+-----+-----+
| Second cell  | Second cell with a |
|              | long text          |
+-----+-----+
| Third cell   | Third cell         |
|              |                     |
+-----+-----+
```

: A grid table with multi-line cells

Table 1: A grid table with multi-line cells

| First Column                                  | Second Column                   |
|-----------------------------------------------|---------------------------------|
| - First item<br>- Second item<br>- Third item | First cell                      |
| Second cell                                   | Second cell with a long<br>text |
| Third cell                                    | Third cell                      |

# Tables — Markdown Syntax

Grid tables can be aligned as well, with colons at the boundaries of the header separator\*

```
+-----+-----+
| Left-Aligned  | Centered  |
+:=====+:=====+:
| - First item  | First cell|
| - Second item|           |
| - Third item  |           |
+-----+-----+
| Second cell   | Second cell with a |
|               | long text          |
+-----+-----+
| Third cell    | Third cell         |
|               |                     |
+-----+-----+
```

: A grid table with multi-line cells

Table 1: A grid table with multi-line cells

| Left-Aligned  | Centered                     |
|---------------|------------------------------|
| - First item  | First cell                   |
| - Second item |                              |
| - Third item  |                              |
| Second cell   | Second cell with a long text |
| Third cell    | Third cell                   |

[\*] Use `:=` for left-aligned, `::=` for centered, `:=` for right-aligned columns.

# Part 6 — Exercises

05:00

## 24) Markdown Tables

- see `reproduce_this.pdf`: table 1 on page 4
- apply in `journals.Rmd`: table 1, between paragraphs 11 and 12

# Tables — stargazer — Overview

- A capable package for creating tables to present
  - data in columns and rows
  - descriptive/summary statistics
  - regression models
- Used widely by academics
- Creates LaTeX code, HTML/CSS code, and ASCII text to be knitted
- A lot is written on this package. See, for example,
  - the [package documentation](#)
  - this [vignette](#) by its author Marek Hlavac
  - this [tutorial](#) by Jake Russ



# Tables — stargazer — Notes

- We must
  - set the chunk option `results="asis"` for chunks with stargazer tables
  - change the argument `type` in the `stargazer()` function for different output formats accordingly
    - e.g., the default `type = "latex"` is for LaTeX and PDF, `type = "html"` for HTML
- It is currently not quite possible to knit stargazer code into tables in Word documents
  - stargazer tables will not appear in Word documents automatically
  - workarounds available
    - knit to HTML as well as Word, copy the tables from HTML to Word
    - knit to PDF as well as Word, open PDF as Word, and copy the tables accross
- stargazer tables might look slightly different in different output formats
  - on the following slides, they will have the HTML look

# Tables — stargazer — Basics

- The `stargazer()` function
  - this is probably the only function you will ever use from this package
    - but it accepts many, many arguments to customise tables
- The data argument of that function, with two main options
  1. a data frame for data or summary statistics tables
    - e.g., `df`, here coming from `df <- read_csv(journals.csv)`
  2. one or more regression models for regression tables
    - e.g., `lm1`, here coming from `lm1 <- lm(h5_index ~ issues, data = df)`

# Tables — stargazer — Data Tables

Table the first four rows of the dataset

```
```{r data_table, echo=FALSE, results="asis"}  
stargazer(data = head(df, n = 4), type = "latex", summary = FALSE)  
```
```

Notice the options of the chunk and the arguments of the function

- with `echo=FALSE`, the code will not be displayed in the output document
- with `results="asis"`, knitr will pass through results without reformatting them
  - these results are produced in LaTeX, due to `type = "latex"`
  - they should remain LaTeX because our outcome document is PDF, converted from LaTeX
- with `summary = FALSE`, the table will present the data, not its descriptive statistics

# Tables — stargazer — Data Tables

Table the first four rows of the dataset

```
```{r data_table, echo=FALSE, results="asis"}  
  
stargazer(data = head(df, n = 4), type = "latex", summary = FALSE)  
  
```
```

% Table created by stargazer v.5.2.2 by Marek Hlavac, Harvard University. E-mail: hlavac at fas.harvard.edu  
% Date and time: Fri, Apr 10, 2020 - 12:31:21

Table 1:

|   | name              | origin   | branch   | h5_index | h5_median | english | subfield | issues | age |
|---|-------------------|----------|----------|----------|-----------|---------|----------|--------|-----|
| 1 | Journal of Bears  | Americas | Physical | 73       | 97        | 1       | 1        | 7      | 61  |
| 2 | Journal of Moon   | Asia     | Social   | 72       | 106       | 1       | 0        | 6      | 64  |
| 3 | Journal of Lumber | Americas | Physical | 72       | 100       | 1       | 1        | 8      | 30  |
| 4 | Journal of Houses | Europe   | Social   | 72       | 102       | 1       | 0        | 8      | 38  |

# Tables — stargazer — Data Tables

Set `header = FALSE` to remove the note preceding tables

```
```{r data_table, echo=FALSE, results="asis"}  
stargazer(data = head(df, n = 4), type = "latex", summary = FALSE, header = FALSE)  
```
```

Table 1:

|   | name              | origin   | branch   | h5_index | h5_median | english | subfield | issues | age |
|---|-------------------|----------|----------|----------|-----------|---------|----------|--------|-----|
| 1 | Journal of Bears  | Americas | Physical | 73       | 97        | 1       | 1        | 7      | 61  |
| 2 | Journal of Moon   | Asia     | Social   | 72       | 106       | 1       | 0        | 6      | 64  |
| 3 | Journal of Lumber | Americas | Physical | 72       | 100       | 1       | 1        | 8      | 30  |
| 4 | Journal of Houses | Europe   | Social   | 72       | 102       | 1       | 0        | 8      | 38  |

# Tables — stargazer — Data Tables

Define a caption with the title argument

```
```{r data_table, echo=FALSE, results="asis"}  
stargazer(data = head(df, n = 4), type = "latex", summary = FALSE, header = FALSE,  
          title = "First four rows of the dataset")  
```
```

Table 1: First four rows of the dataset

|   | name              | origin   | branch   | h5_index | h5_median | english | subfield | issues | age |
|---|-------------------|----------|----------|----------|-----------|---------|----------|--------|-----|
| 1 | Journal of Bears  | Americas | Physical | 73       | 97        | 1       | 1        | 7      | 61  |
| 2 | Journal of Moon   | Asia     | Social   | 72       | 106       | 1       | 0        | 6      | 64  |
| 3 | Journal of Lumber | Americas | Physical | 72       | 100       | 1       | 1        | 8      | 30  |
| 4 | Journal of Houses | Europe   | Social   | 72       | 102       | 1       | 0        | 8      | 38  |

# Tables — stargazer — Summary Statistics Tables

Create a table of summary statistics instead, for the complete dataset

```
```{r summary_table, echo=FALSE, results="asis"}  
stargazer(data = df, type = "latex", summary = TRUE, header = FALSE,  
          title = "Descriptive statistics")  
```
```

| Table 1: Descriptive statistics |       |        |          |     |          |          |     |
|---------------------------------|-------|--------|----------|-----|----------|----------|-----|
| Statistic                       | N     | Mean   | St. Dev. | Min | Pctl(25) | Pctl(75) | Max |
| h5_index                        | 1,091 | 26.361 | 13.814   | 1   | 17       | 35       | 73  |
| h5_median                       | 1,091 | 39.400 | 21.272   | 3   | 25       | 52       | 109 |
| issues                          | 1,091 | 4.676  | 1.786    | 1   | 3        | 6        | 12  |
| age                             | 1,091 | 42.902 | 26.370   | 1   | 23       | 56       | 158 |

# Tables — stargazer — Summary Statistics Tables

Keep only a selection of statistics

```
```{r summary_table, echo=FALSE, results="asis"}  
stargazer(data = df, type = "latex", summary = TRUE, header = FALSE,  
          title = "Descriptive statistics", summary.stat = c("n", "mean", "sd", "min", "max"))  
```
```

| Table 1: Descriptive statistics |       |        |          |     |     |
|---------------------------------|-------|--------|----------|-----|-----|
| Statistic                       | N     | Mean   | St. Dev. | Min | Max |
| h5_index                        | 1,091 | 26.361 | 13.814   | 1   | 73  |
| h5_median                       | 1,091 | 39.400 | 21.272   | 3   | 109 |
| issues                          | 1,091 | 4.676  | 1.786    | 1   | 12  |
| age                             | 1,091 | 42.902 | 26.370   | 1   | 158 |



# Tables — stargazer — Summary Statistics Tables

Omit a selection of statistics for the same effect

```
```{r summary_table, echo=FALSE, results="asis"}  
stargazer(data = df, type = "latex", summary = TRUE, header = FALSE,  
          title = "Descriptive statistics", omit.summary.stat = c("p25", "p75"))  
```
```

| Table 1: Descriptive statistics |       |        |          |     |     |
|---------------------------------|-------|--------|----------|-----|-----|
| Statistic                       | N     | Mean   | St. Dev. | Min | Max |
| h5_index                        | 1,091 | 26.361 | 13.814   | 1   | 73  |
| h5_median                       | 1,091 | 39.400 | 21.272   | 3   | 109 |
| issues                          | 1,091 | 4.676  | 1.786    | 1   | 12  |
| age                             | 1,091 | 42.902 | 26.370   | 1   | 158 |

# Tables — stargazer — Summary Statistics Tables

Flip the table

```
```{r summary_table, echo=FALSE, results="asis"}  
stargazer(data = df, type = "latex", summary = TRUE, header = FALSE, flip = TRUE,  
          title = "Descriptive statistics", omit.summary.stat = c("p25", "p75"))  
```
```

| Table 1: Descriptive statistics |          |           |        |        |
|---------------------------------|----------|-----------|--------|--------|
| Statistic                       | h5_index | h5_median | issues | age    |
| N                               | 1,091    | 1,091     | 1,091  | 1,091  |
| Mean                            | 26.361   | 39.400    | 4.676  | 42.902 |
| St. Dev.                        | 13.814   | 21.272    | 1.786  | 26.370 |
| Min                             | 1        | 3         | 1      | 1      |
| Max                             | 73       | 109       | 12     | 158    |

# Part 6 — Exercises

05:00

## 25) Summary Statistics Tables

- see `reproduce_this.pdf`: table 2 on page 8
- apply in `journals.Rmd`: table 2, between paragraphs 23 and 24

# Tables — stargazer — Regression Tables

Create a table of regression models instead

```
```{r regression_table, echo=FALSE, results="asis"}
stargazer(data = lm(h5_index ~ subfield, data = df),
  type = "latex", header = FALSE,
  title = "Regression Results")
```
```

Table 1: Regression Results

|                         | <i>Dependent variable:</i>     |
|-------------------------|--------------------------------|
|                         | h5_index                       |
| subfield1               | -12.926***<br>(0.896)          |
| Constant                | 36.171***<br>(0.781)           |
| Observations            | 1,091                          |
| R <sup>2</sup>          | 0.160                          |
| Adjusted R <sup>2</sup> | 0.160                          |
| Residual Std. Error     | 12.665 (df = 1089)             |
| F Statistic             | 207.926*** (df = 1; 1089)      |
| Note:                   | * p<0.1; ** p<0.05; *** p<0.01 |

# Tables — stargazer — Regression Tables

Models can also be estimated outside the function first

```
```{r regression_table, echo=FALSE, results="asis"}  
lm1 <- lm(h5_index ~ subfield, data = df)  
stargazer(data = lm1, type = "latex", header = FALSE,  
          title = "Regression Results")  
```
```

Table 1: Regression Results

|                         | <i>Dependent variable:</i>     |
|-------------------------|--------------------------------|
|                         | h5_index                       |
| issues                  | 1.913***<br>(0.227)            |
| Constant                | 17.415***<br>(1.137)           |
| Observations            | 1,091                          |
| R <sup>2</sup>          | 0.061                          |
| Adjusted R <sup>2</sup> | 0.060                          |
| Residual Std. Error     | 13.391 (df = 1089)             |
| F Statistic             | 70.959*** (df = 1; 1089)       |
| Note:                   | * p<0.1; ** p<0.05; *** p<0.01 |

# Tables — stargazer — Regression Tables

Keep only a selection of statistics

```
```{r regression_table, echo=FALSE, results="asis"}  
stargazer(data = lm1, type = "latex", header = FALSE,  
          title = "Regression Results",  
          keep.stat = c("n", "rsq"))  
```
```

Table 1: Regression Results

|                | <i>Dependent variable:</i>     |
|----------------|--------------------------------|
|                | h5_index                       |
| issues         | 1.913***<br>(0.227)            |
| Constant       | 17.415***<br>(1.137)           |
| Observations   | 1,091                          |
| R <sup>2</sup> | 0.061                          |
| Note:          | * p<0.1; ** p<0.05; *** p<0.01 |

# Tables — stargazer — Regression Tables

Display multiple models in the same table

```
```{r regression_table, echo=FALSE, results="asis"}
stargazer(data = list(lm1, lm2), type = "latex",
          header = FALSE, title = "Regression Results",
          keep.stat = c("n", "rsq"))
```
```

Table 1: Regression Results

|                | <i>Dependent variable:</i>  |           |
|----------------|-----------------------------|-----------|
|                | h5_index                    |           |
|                | (1)                         | (2)       |
| issues         | 1.913***                    | 1.424***  |
|                | (0.227)                     | (0.212)   |
| english1       |                             | 17.262*** |
|                |                             | (1.244)   |
| Constant       | 17.415***                   | 4.226***  |
|                | (1.137)                     | (1.415)   |
| Observations   | 1,091                       | 1,091     |
| R <sup>2</sup> | 0.061                       | 0.202     |
| Note:          | *p<0.1; **p<0.05; ***p<0.01 |           |

# Tables — stargazer — Regression Tables

Change variable labels

```
```{r regression_table, echo=FALSE, results="asis"}
stargazer(data = list(lm1, lm2), type = "latex",
  header = FALSE, title = "Regression Results",
  keep.stat = c("n", "rsq"),
  dep.var.labels = "H5 Index",
  covariate.labels = c("Issues", "English"))
```
```

Table 1: Regression Results

|                | <i>Dependent variable:</i>     |           |
|----------------|--------------------------------|-----------|
|                | H5 Index                       |           |
|                | (1)                            | (2)       |
| Issues         | 1.913***                       | 1.424***  |
|                | (0.227)                        | (0.212)   |
| English        |                                | 17.262*** |
|                |                                | (1.244)   |
| Constant       | 17.415***                      | 4.226***  |
|                | (1.137)                        | (1.415)   |
| Observations   | 1,091                          | 1,091     |
| R <sup>2</sup> | 0.061                          | 0.202     |
| Note:          | * p<0.1; ** p<0.05; *** p<0.01 |           |



# Tables — stargazer — Regression Tables

Change significance levels

```
```{r regression_table, echo=FALSE, results="asis"}  
stargazer(data = list(lm1, lm2), type = "latex",  
  header = FALSE, title = "Regression Results",  
  keep.stat = c("n", "rsq"),  
  dep.var.labels = "H5 Index",  
  covariate.labels = c("Issues", "English"),  
  star.cutoffs = c(0.05, 0.01, 0.001))  
```
```

Table 1: Regression Results

|                | <i>Dependent variable:</i>    |                      |
|----------------|-------------------------------|----------------------|
|                | H5 Index                      |                      |
|                | (1)                           | (2)                  |
| Issues         | 1.913***<br>(0.227)           | 1.424***<br>(0.212)  |
| English        |                               | 17.262***<br>(1.244) |
| Constant       | 17.415***<br>(1.137)          | 4.226**<br>(1.415)   |
| Observations   | 1,091                         | 1,091                |
| R <sup>2</sup> | 0.061                         | 0.202                |
| Note:          | *p<0.05; **p<0.01; ***p<0.001 |                      |

# Tables — stargazer — Regression Tables

Label the table for in-text reference

```
```{r regression_table, echo=FALSE, results="asis"}  
stargazer(data = list(lm1, lm2), type = "latex",  
  header = FALSE, title = "Regression Results",  
  keep.stat = c("n", "rsq"),  
  dep.var.labels = "H5 Index",  
  covariate.labels = c("Issues", "English"),  
  star.cutoffs = c(0.05, 0.01, 0.001),  
  label = "regression_results")  
```\n`
```

Table 1: Regression Results

|                | <i>Dependent variable:</i>    |           |
|----------------|-------------------------------|-----------|
|                | H5 Index                      |           |
|                | (1)                           | (2)       |
| Issues         | 1.913***                      | 1.424***  |
|                | (0.227)                       | (0.212)   |
| English        |                               | 17.262*** |
|                |                               | (1.244)   |
| Constant       | 17.415***                     | 4.226**   |
|                | (1.137)                       | (1.415)   |
| Observations   | 1,091                         | 1,091     |
| R <sup>2</sup> | 0.061                         | 0.202     |
| Note:          | *p<0.05; **p<0.01; ***p<0.001 |           |

# Part 6 — Exercises

07:30

## 26) Regression Tables

- see `reproduce_this.pdf`: table 3 on page 10
- apply in `journals.Rmd`: table 3, between paragraphs 30 and 31

# Tables — stargazer — Referring to Tables

Refer to tables themselves with the `\autoref{table_label}` syntax

```
\autoref{regression_results} provides results from two OLS models.
```

**Table 1** provides results from two OLS models.

Refer to specific results within tables with inline code

```
In Model 1, the coefficient for _Issues_ is  
'r round(coef(summary(lm1))["issues", "Estimate"], digits = 2)`.
```

In Model 1, the coefficient for *Issues* is 1.91.

# Part 6 — Exercises

05:00

## 27) Referring to Tables

- see `reproduce_this.pdf`: pages 7 and 9
- apply in `journals.Rmd`: paragraph 23 and 29

## Part 7. Addressing Functionality Gaps

# Functionality Gaps

- Not everything is possible to achieve with R Markdown syntax, code chunks, and/or code
  - e.g., centering text
- Workarounds available through inclusion of other languages and/or syntaxes in .Rmd documents
  - e.g., incorporating HTML or LaTeX code into R Markdown
  - workarounds might be output specific
    - e.g., LaTeX-based workarounds may work only for LaTeX and PDF outputs
- There are no exclusive list of gaps or workarounds
  - these are gap- or problem-specific
  - after writing a few manuscripts with R Markdown, you will have addressed most typical gaps in your workflow

# Functionality Gaps — Examples

## Problem:

There is no chunk option for figures labels. Without these, we cannot refer to figures with a link within text.

## Solution:

Insert a LaTeX label into the `fig.caption` option, and use the `\autoref{figure_caption}` syntax in text

```
\autoref{scatter_plot} visualises the relationship between the two journal metrics.
```

```
```{r ... fig.caption = "A Scatter Plot \\label{scatter_plot}"}  
ggplot(data = df) +  
  geom_point(...  
  ...
```

**Figure 1** visualises the relationship between the two journal metrics.



# Functionality Gaps — Examples

## Problem:

Markdown tables do not have labels, which would otherwise allow us to refer to them in text

## Solution:

Insert a LaTeX label after the table caption, and use the `\autoref{table_caption}` syntax in text

See `\autoref{handmade_table}` for further details.

: A hand-made table with R Markdown `\label{handmade_table}`

|               |          |  |
|---------------|----------|--|
| +-----+-----+ |          |  |
| Left-Aligned  | Centered |  |
| ...           |          |  |

See **Table 1** for further details.

# Part 7 — Exercises

05:00

## 28) Referring to Figures

- see `reproduce_this.pdf`: pages 6 and 9
- apply in `journals.Rmd`: paragraphs 19, 21, and 27

## 29) Referring to Markdown Tables

- see `reproduce_this.pdf`: page 4
- apply in `journals.Rmd`: paragraph 11

# Functionality Gaps — Examples

## Problem:

R Markdown adds the list of references to the end of documents. This might be undesirable for some manuscripts, for example those with an appendix. Similarly, some journals require tables and figures to be added after references.

## Solution:

Define where exactly the list of references should appear with the HTML code `<div id = "refs">`

```
# References
<div id = "refs"></div>

# Appendix
```

# Functionality Gaps — Examples

## Problem:

R Markdown produces outputs with single-line-spaced text while we might prefer or be required (e.g., by journal submission rules) to double-space our manuscripts.

## Solution:

Use the `doublespacing` command from the LaTeX package `setspace` (Carlisle et al., 2011)

- because the command comes from a package, we need to add it to YAML with `header-includes`
- including commands in YAML ensures they are applied through the output\*

```
---  
...  
header-includes:  
  - \usepackage{setspace}\doublespacing  
---
```

[\*] This can be reversed for specific sections in text, with `singlespacing` placed before and `doublespacing` after a section.

# Part 7 — Exercises

02:00

## 30) Line Spacing

- introduce 1.5 spacing to the manuscript
  - hint: the command is called `onehalfspacing`
- except for the Abstract, which should be single spaced

# Functionality Gaps — Examples

## Problem:

Pages, tables, figures etc. are numbered continuously across an output. We might prefer or be required (e.g., by journal submission rules) to change this behaviour, for example for appendices.

## Solution:

Use the `setcounter` in combination with the `renewcommand` command, outside code chunks

```
\setcounter{page}{1}  
\renewcommand*{\thepage}{A\arabic{page}}  
  
\setcounter{table}{0}  
\renewcommand*{\thetable}{A\arabic{table}}  
  
\setcounter{figure}{0}  
\renewcommand*{\thefigure}{A\arabic{figure}}
```

# Part 8. Using Version Control

# Version Control

- Academic papers have many versions before publication
  - typically written over a long period of time, in numerous sittings, and often by multiple authors
  - at the end of every sitting by each co-author, essentially a different version of the same manuscript is created
- With many versions created over time on different computers by different co-authors, there emerges at least three challenges
  1. keeping track of changes and versions
  2. communicating these changes and versions between/among co-authors
  3. merging the changes and versions
- We all version control, in different ways, such as
  1. edit, rename, save
    - for collaboration, save in a shared folder or attach to an email message
  2. use applications or websites such as Dropbox, Google Docs, Overleaf
  3. use distributed version control systems such as Git and GitHub



# Version Control — Manual Attempts

Typically, hand-made attempts to version control lead to uncontrollable folders

## Single-authored projects

```
manuscript
|
|- journals_FINAL_19MAY.Rmd
|- journals_FINAL.Rmd
|- journals_26APRIL_newlit.Rmd
...
|- journals.Rproj
|- references.bib
|- apa_7th.csl
```

## Co-authored projects

```
manuscript
|
|- journals_FINAL_19MAY_RU.Rmd
|- journals_FINAL_EK.Rmd
|- journals_FINAL_RU.Rmd
|- journals_FINAL.Rmd
|- journals_29APRIL_newlit_EK.Rmd
|- journals_26APRIL_newlit.Rmd
...
|- journals.Rproj
|- references.bib
|- apa_7th.csl
```

# Version Control — Git and GitHub — Definitions

- Git
  - a software that keeps track of versions of a set of files
  - it is local, the records are kept on your computer
- GitHub
  - a hosting service, or a website, that can keep the records
  - it is like Dropbox, but specifically structured to keep records with Git
- Repository, or repo
  - a set of files whose records are kept together, by Git and/or on GitHub
  - it is like a folder, which can keep files and other folders containing files

# Version Control — Git and GitHub — Definitions

- To commit
  - to take a snapshot of, to version a repository
  - it is like saving a new version of all changed files in your folder with a new name
  - it is local, the records are kept on your computer unless you push
- To push
  - to move the records from Git to GitHub, from your computer to online
  - it is like uploading (the new versions of) your files to a website,
  - it also involves mergeing, if this not the first push
- To pull
  - to move the updated records (e.g., by your co-authors) from GitHub to your computer
  - it is like downloading a zipped folder of files, and merging with the ones on your computer
- To clone
  - to move a repository from GitHub to your computer for first time
  - it is like downloading a zipped folder, which you do not have on your computer yet

# Version Control — Git and GitHub

Version control with Git and GitHub requires

1. *initial setup*, done once<sup>\*</sup>
  - unless for a new computer or, if ever, a new GitHub account
  - a bit technical, but worth the hassle
2. *project setup*, repeated for every paper
  - shorter, less complicated

[\*] We have *started* this process already, in Part 1 of the workshop, by downloading and installing Git and signing up for GitHub.

# Version Control — Git — Initial Setup

## 1) Enable version control with RStudio

- from the RStudio menu, follow:

```
Tools -> Global Options -> Git/SVN -> Enable version control interface for  
RStudio projects
```

- RStudio will likely find Git automatically. In case it cannot, Git is likely to be at

```
c:/Program Files/Git/bin/git.exe  
  
/usr/local/git/bin/git
```

## 2) Set Git Bash as your shell (Windows-only step)

- from the RStudio menu, follow:

```
Tools -> Global Options -> Terminal -> New terminals open with: Git Bash
```

# Version Control — Git — Initial Setup

## 3) Introduce yourself to Git

- from the RStudio menu, follow:

Tools -> Terminal -> New Terminal

- Enter the following lines in the Terminal, with the email address that you have used to sign up for GitHub

```
git config --global user.name "YOUR NAME"  
git config --global user.email "YOUR EMAIL ADDRESS"  
git config --global --list
```

# Version Control — Git and Github — Project Setup

1) Initiate local version control with Git

- from the RStudio menu, follow:

```
Tools -> Version Control -> Project Setup... -> Version Control System (Git)
```

# Version Control — Git and Github — Project Setup

## 2) Create a new GitHub repository

- on GitHub, follow:

```
Repositories -> New -> Repository name (e.g., "rwd_workshop") -> Public ->  
Create repository
```

- observe that:
  - repository URLs have the following structure: [https://github.com/USER\\_NAME/REPOSITORY\\_NAME](https://github.com/USER_NAME/REPOSITORY_NAME)
    - this is the address to view the repository online
    - the address gets `.git` added to the end, ([https://github.com/USER\\_NAME/REPOSITORY\\_NAME.git](https://github.com/USER_NAME/REPOSITORY_NAME.git)), for use in the Terminal



# Version Control — Git and Github — Project Setup

## 3) Push an existing repository

- from the RStudio menu, follow:

```
Tools -> Terminal -> New Terminal
```

- enter the following lines in the Terminal, with your username and repository name

```
git remote add origin https://github.com/USER_NAME/REPOSITORY_NAME.git  
git push -u origin master
```

- observe that:
  - now there is now a Git tab in RStudio, documenting the differences between you local repository and the one on GitHub. When you change a file, it will appear here.
  - your files now include a `.gitignore` file, where you can list files to be excluded from being tracked

# Version Control — Git and Github — Workflow

## 1) Edit and save one or more files under version control

- e.g., delete the first sentence of the abstract, and save

## 2) Commit and Push

- under the Git tab on RStudio, find the list of files edited (`Status = M`; modified) since the last push
- tick `Staged*` for one or more files that you would like to commit
  - enter a `Commit` message that summarises the edits
  - click `Commit` to create a record of the new version locally to your computer
  - click `Close -> Push` to push the version to GitHub

[\*] To stage is to add files to be committed. It allows us to commit one or more files together or separately.

## Part 7 — Exercises

05:00

### 30) Reproducibility and Version Control

- imagine that, after producing all these tables and figures, and writing up your results, you have decided to exclude *Oceania* from analysis
  - hint: use the `filter` function in the data chunk
  - create a new version, and push it to GitHub

## Part 9. Collaborating with Others

# Colloboration — Git and Github — Workflow

## 1. Create a new RStudio project

- From the Rstudio menu, follow

File -> New Project -> Version Control -> Git

- Enter Repository URL<sup>\*</sup>, give your local project folder a name, and save it to a desired location.

Open in a new session -> Create Project

## 2. Edit, save, commit, and push as before

- In a collaborative project, it is recommended to push frequently to avoid merge conflicts

[\*] Notice that this is the URL and not the .git version. If the collaborator's repository is private, they will have to invite you with administrative privileges first.

# Colloboration — Git and Github — Workflow

## 3. Pull frequently

- Colloborative projects require pulling as well as pushing because your colloborator(s) might have pushed their commits to GitHub
- Pulling and pushing frequently minimises the risk of merge conflicts

# Colloboration — Git and Github — Merge Conflicts

- Changes are tracked line by line.
- If colloborators work on the same document, but edits different lines, Git merges the versions from different colloborators without problems.
- Conflicts occur when colloborators edit the same line in the same document.
  - This requires human intervention, to decide which edit to keep and which to discharge.
- Pulling and pushing frequently minimises the risk that colloborators will be editing the same line.

## Part 10. Working on a Real Project



**Thank You!**

# References

- Allaire, J. J., Xie, Y., McPherson, J., Luraschi, J., Ushey, K., Atkins, A., Wickham, H., Cheng, J., Chang, W. and Iannone, R. (2020). **rmarkdown: Dynamic documents for R**. R package, version 2.1.
- Blair, G., Cooper, J., Coppock, A., Humphreys, M., Rudkin, A. and Fultz, N. (2019). **fabricatr: Imagine your data before you collect it**. R package, version 0.10.0.
- Carlisle, D., Fairbairns, R., Harris, E. and Tobin, G. (2011). **setspace – Set space between lines**. LaTeX package, version 6.7a.
- Dowle, M. and Srinivasan, A. (2019). **data.table: Extension of 'data.frame'**. R package, version 1.12.8.
- Gagolewski, M. (2020). **stringi: Character String Processing Facilities**. R package, version 1.4.6.
- Hlavac, M. (2018). **stargazer: Well-formatted regression and summary statistics tables**. R package, version 5.2.2.
- Wickham, H. and Grolemund, G. (2019). R for data science. O'Reilly. Open access at <https://r4ds.had.co.nz>.
- Wickham, H., Chang, W., Henry, L., Pedersen, T. L., Takahashi, K., Wilke, C., Woo, K., Yutani, H. and Dunnington, D. (2020a). **dplyr: A grammar of data manipulation**. R package, version 0.8.5.

# References

- Wickham, H., François, W., Henry L. and Müller, K. (2020b). **ggplot2: Create elegant data visualisations using the grammar of graphics**. R package, version 0.8.5.
- Wiernik, B. M. (2020). **American Psychological Association 7th edition (no ampersand)**. Citation style language file, version 1.0.
- Xie, Y. (2020a). **tinytex: Helper functions to install and maintain TeX Live and compile LaTeX documents**. R package, version 0.21.
- Xie, Y. (2020b). **knitr: A general-purpose package for dynamic report generation in R**. R package, version 1.28.
- Xie, Y. (2020c). **bookdown: Authoring books and technical documents with R Markdown**. R package, version 0.18.
- Xie, Y. (2020d). **blogdown: Create blogs and websites with R Markdown**. R package, version 0.18.
- Xie, Y., Allaire, J. J., and Golemund, G. (2019). R markdown: The definitive guide. CRC Press. Open access at <https://bookdown.org/yihui/rmarkdown>.
- Zhu, H. (2019). **kableExtra: Construct Complex Table with 'kable' and Pipe Syntax**. R package, version 1.1.0.