

EEB 603 – Chapter 1: Using the R Markdown language to promote Reproducible Science

Part B: Tables, Figures and References with R bookdown and allied packages

Sven Buerki

2020-09-24

Contents

1	Aim	2
2	Learning outcomes	2
3	Associated files supporting this tutorial	3
4	Set your R Markdown environment	3
4.1	Install dependencies	3
4.2	YAML metadata section	5
4.3	Load required R packages	7
4.4	Generate citation file of R packages used to produce the report	7
4.5	Generate Appendix with citations for all R packages	9
4.6	Generate Appendix with R package versions used to produce report	10
5	All set, good to go!	11
6	Insert tables and figures in R Markdown document	11
6.1	Tables	11
6.2	Figures	13
7	Cross-reference tables and figures in the text	14
7.1	General syntax	14
7.2	Step-by-step procedure	14
8	Citing references in text and adding a bibliography/references section	15
8.1	The bibliography file	15
8.2	Specifying a bibliography in the R Markdown file	15
8.3	BibTeX format	15
8.4	Citation identifier	16

8.5	Citing references	16
8.6	Adding a bibliography/references section	17
9	Formatting citations to journal styles	17
9.1	What is the citation style language (CSL)?	18
9.2	CSL repositories	18
9.3	How to use a CSL file in an R Markdown to format citations and bibliography?	18
10	References	19
A	Appendix 1	19
B	Appendix 2	20

1 Aim

The aim of this tutorial is to provide students with the expertise to generate reproducible reports using *bookdown* (Xie, 2016, 2020a) and allied R packages (see Appendix 1 for a full list). Unlike functions implemented in the R *rmarkdown* package (Xie et al., 2018, which was better suited to generating PDF reproducible reports), *bookdown* allows to use ONE unified set of functions to generate HTML and PDF documents. In addition, the same approach and functions are used to process tables and figures as well as cross-reference those in the main body of the text. In this tutorial, we will also cover procedures to cite references in the text, automatically generate a bibliography/references section and format citations to journal styles as well as generating an Appendix containing citations of all R packages used to conduct your research (and produce the report).

2 Learning outcomes

This tutorial is devoted to part B of chapter 1 and provides students with opportunities to learn procedures to:

- Set your R Markdown environment:
 - Install dependencies.
 - YAML metadata section.
 - Load required R packages.
 - Generate citation file of R packages used to produce the report.
 - Generate Appendix with citations for all R packages.
 - Generate Appendix with R package versions used to produce report.
- Insert tables and figures in R Markdown document.
- Cross-reference tables and figures in the text.
- Cite references in the text and add a bibliography/references section.
- Format citations to journal styles.

3 Associated files supporting this tutorial

Although less used in part A, a set of files are provided to support teaching of material presented in this chapter. These files are deposited in the shared Google Drive at this path:

- `Reproducible_Science/Chapters/Chapter_1/Tutorial_files`

Files are as follows:

- `EEB603_Syllabus_BUERKI.Rmd`: This is the `.Rmd` file used to compile the syllabus of this class. This file provides a good source of information for the syntax and protocols described in this tutorial.
- `Bibliography_Reproducible_Science_2.bib`: This file contains references cited in BibTex format.
- `AmJBot.csl`: This citation style language (CSL) file allows formatting citations and bibliography following citation style of *American Journal of Botany*.
- `Bioinformatic_workflow_PART2.pdf`: A pdf file containing the bioinformatic workflow taught in this class. This file will be used to learn how to incorporate a figure into R Markdown file.

4 Set your R Markdown environment

4.1 Install dependencies

4.1.1 R packages

To execute this tutorial the following R packages have to be installed on your computer using code provided below:

```
install.packages(c("knitr", "rmarkdown", "bookdown", "formattable",  
  "kableExtra", "dplyr", "magrittr", "prettydoc", "htmltools",  
  "knitcitations"))
```

4.1.2 TeX distribution

If you are planning to create PDF documents, you will need to install a TeX distribution on your computers. Please refer to this website for more details: <https://www.latex-project.org/get/>

4.1.3 Set MiKTeX for compiling PDF documents on Windows computers

Several students working on Windows computers shared difficulties in compiling PDF documents in RStudio. This issue is associated to MiKTeX preventing RStudio to install or update TeX packages required to knit your documents.

To solve this issue apply the following procedure:

1. Start MiKTeX console by searching and clicking **MiKTeX Console** in the application launcher.
2. Enable automatic package installation by:
 - a. Clicking on **Settings** tab.
 - b. Ticking the radio button **Always install missing packages on-the-fly** under the “You can choose whether missing packages are to be installed on-the-fly” header (Figure 1).
3. Restarting RStudio and you should be able to knit pdf documents.

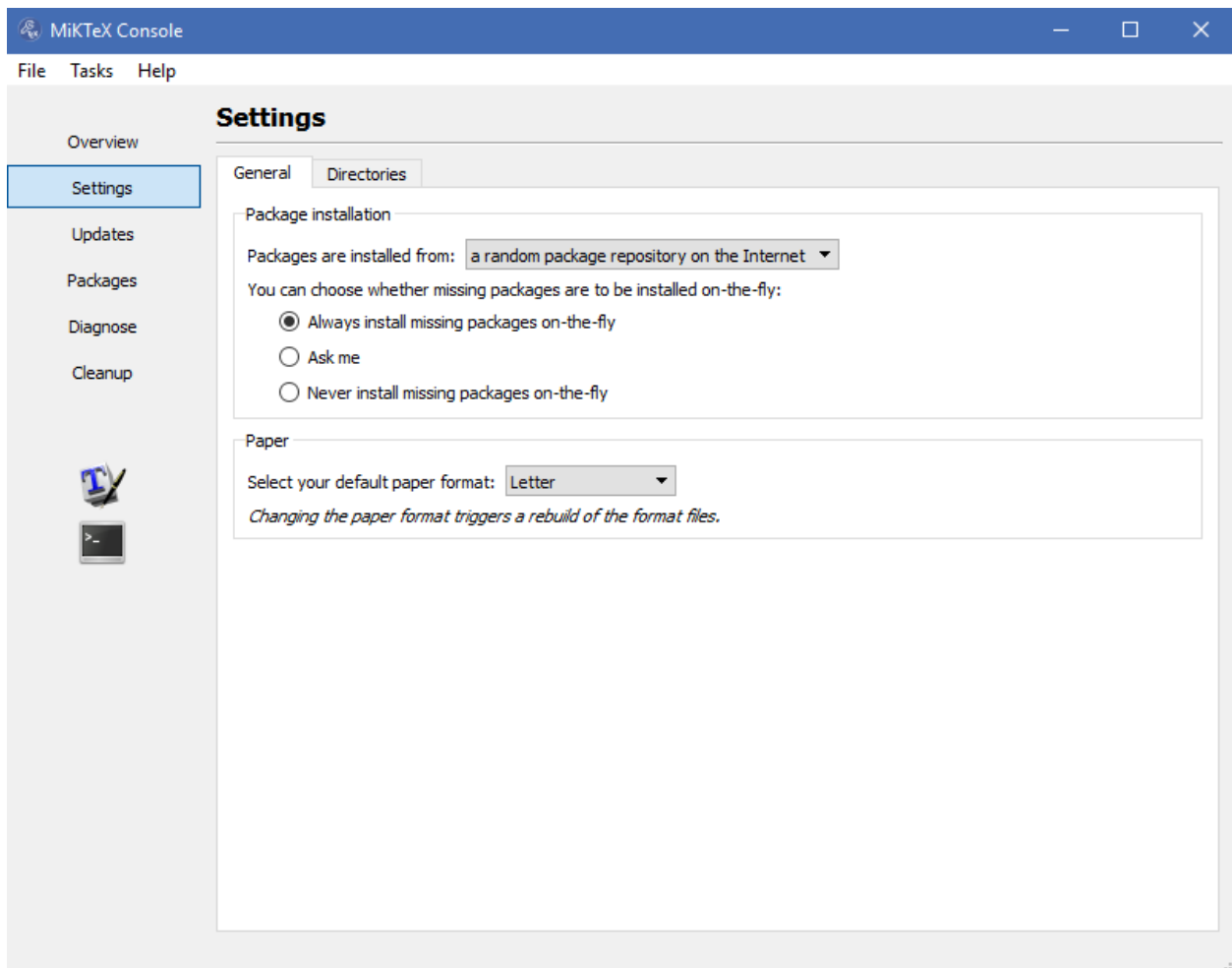


Figure 1: Snapshot of the MikTeX Console showing the procedure to always install packages on-the-fly.

4.2 YAML metadata section

The YAML metadata section allows users to provide arguments (referred to as fields) to convert their R Markdown document into its final form. Here, we will be using functions implemented in the *knitr* (Xie, 2015, 2020b) and *bookdown* packages (Xie, 2016, 2020a) to define (field names as declared in the YAML metadata section are provided between parenthesis):

1. Title (`title`).
2. Subtitle (`subtitle`).
3. Author(s) (`author`).
4. Date (`date`).
5. Output format(s) (`output`).
6. Citations link (`link-citations`).
7. Font size (`fontsize`).
8. Bibliography file(s) (`bibliography`).
9. Format for citations to follow journal styles (`cs1`).

The YAML code provided below outputs either an HTML or PDF document (see `output` field) with a table of content (see `toc` field) and generates in text citations and bibliography section as declared in the `AmJBot.cs1` file (under the `cs1` field).

```
---
title: 'Your title'
subtitle: "Your subtitle"
author: "Your name"
date: "`r Sys.Date()`"
output:
  bookdown::html_document2:
    toc: TRUE
  bookdown::pdf_document2:
    toc: TRUE
link-citations: yes
fontsize: 12pt
bibliography: [Bibliography_Reproducible_Science_2.bib, packages.bib]
cs1: AmJBot.cs1
---
```

Warning: `.bib` and `.cs1` files have to be stored in the same working directory as your `.Rmd` file.

4.2.1 Step-by-step procedure

Do the following to set your YAML metadata section:

1. Create a new R Markdown document.
2. Save the `.Rmd` document into a new project folder in `Reproducible_Science/Chapters/Chapter_1/`.

3. Copy `Bibliography_Reproducible_Science_2.bib` and `AmJBot.csl` in your project folder. These files are available on the Shared Google Drive folder:
 - `Reproducible_Science/Chapters/Chapter_1/Tutorial_files`
4. Edit the YAML metadata section as shown in the above code.

Note:

- You can use R functions in the YAML metadata section using inline R code (see part A for more details). Here, we use `Sys.Date()` to automatically date the output document.
- The `bibliography` field includes two BibTeX files, the first contains your regular citations (of publications cited in the text; see The bibliography file), whereas the second is generated by `knitr::write_bib()` and contains citations for all R packages used to generate this report (see below for more details).

4.2.2 Knitting procedure

Since you have declared two output documents in the YAML metadata section and that those are specific to `bookdown` functions, you will have to select which output format you want to use to compile your document by clicking on the drop-down list on the left side of the Knit button (see Figure 2). To use `bookdown` functions, please make sure to select one of the following options (see Figure 2): `Knit to html_document2` or `Knit to pdf_document2`.

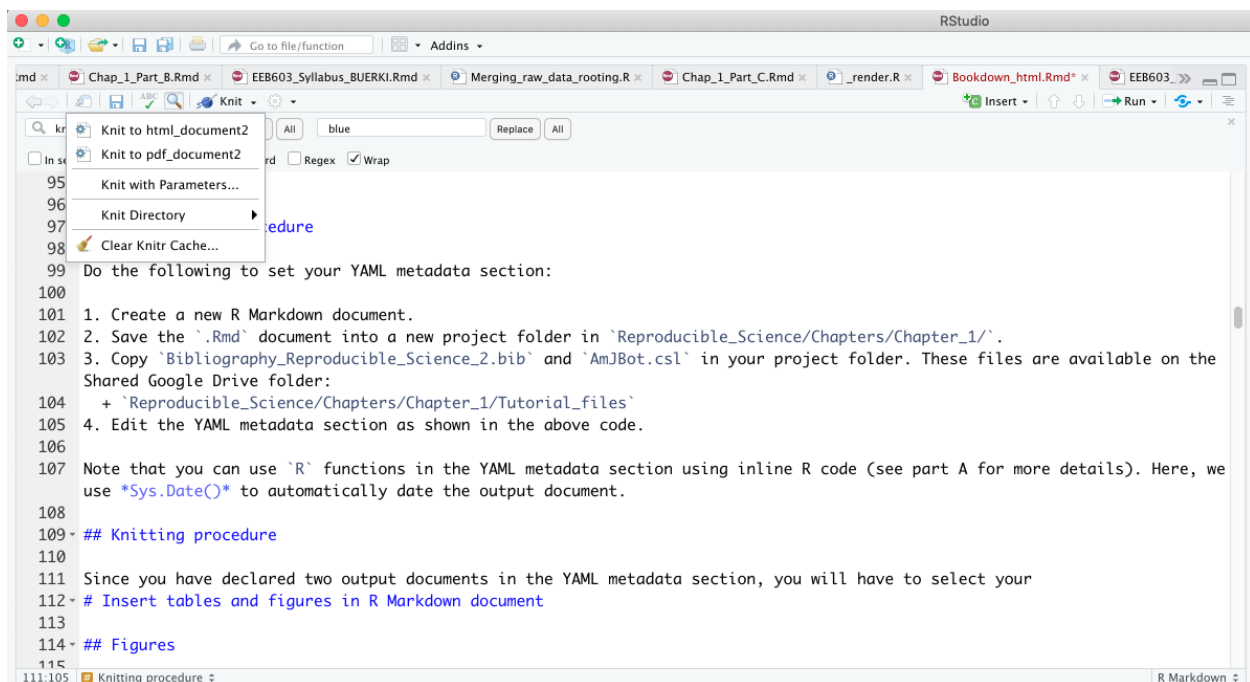


Figure 2: Snapshot of RStudio console showing the drop-down list associated to Knit button.

4.3 Load required R packages

It is best practice to add an R code chunk directly under the YAML metadata section to load all the required R packages used to produce your report. This feature will also allow to automatically generate a citation file with all R packages used to generate your report (see below). Applying this approach will contribute to improving the reproducibility of your research!

4.3.1 Step-by-step procedure

1. Include an R code chunk directly under your YAML metadata section.
2. Name code chunk `packages` and set options line as follows:
 - `echo=FALSE`,
 - `warning=FALSE`,
 - `include=FALSE`.
3. Add the following code to load required packages to produce your report:

```
#Load packages
library(knitr)
library(rmarkdown)
library(bookdown)
library(formattable)
library(kableExtra)
library(dplyr)
library(magrittr)
library(prettydoc)
library(htmltools)
library(knitcitations)
library(devtools)
```

4.4 Generate citation file of R packages used to produce the report

I don't know about you, but I am always struggling to properly cite R packages in my publications. If you want to retrieve the citation for an R package, you can use the base R function `citation()`. For instance, citations for *knitr* can be obtained as follow:

```
# Generate citation for knitr
citation("knitr")
```

```
##
## To cite the 'knitr' package in publications use:
##
## Yihui Xie (2020). knitr: A General-Purpose Package for Dynamic Report
```

```
## Generation in R. R package version 1.29.
##
## Yihui Xie (2015) Dynamic Documents with R and knitr. 2nd edition.
## Chapman and Hall/CRC. ISBN 978-1498716963
##
## Yihui Xie (2014) knitr: A Comprehensive Tool for Reproducible
## Research in R. In Victoria Stodden, Friedrich Leisch and Roger D.
## Peng, editors, Implementing Reproducible Computational Research.
## Chapman and Hall/CRC. ISBN 978-1466561595
##
## To see these entries in BibTeX format, use 'print(<citation>,
## bibtex=TRUE)', 'toBibtex(.)', or set
## 'options(citation.bibtex.max=999)'.
```

If you want to generate those latter citation entries in BibTeX format, you can pass the returned object of `citation()` to `toBibtex()` as follows:

```
# Generate citation for knitr in BibTeX format Note that
# there is no citation identifiers. Those will be
# automatically generated in our next code.
toBibtex(citation("knitr"))
```

```
## @Manual{,
##   title = {knitr: A General-Purpose Package for Dynamic Report Generation in R},
##   author = {Yihui Xie},
##   year = {2020},
##   note = {R package version 1.29},
##   url = {https://yihui.org/knitr/},
## }
##
## @Book{,
##   title = {Dynamic Documents with {R} and knitr},
##   author = {Yihui Xie},
##   publisher = {Chapman and Hall/CRC},
##   address = {Boca Raton, Florida},
##   year = {2015},
##   edition = {2nd},
##   note = {ISBN 978-1498716963},
##   url = {https://yihui.org/knitr/},
## }
##
## @InCollection{,
##   booktitle = {Implementing Reproducible Computational Research},
##   editor = {Victoria Stodden and Friedrich Leisch and Roger D. Peng},
##   title = {knitr: A Comprehensive Tool for Reproducible Research in {R}},
##   author = {Yihui Xie},
```



```
## publisher = {Chapman and Hall/CRC},
## year = {2014},
## note = {ISBN 978-1466561595},
## url = {http://www.crcpress.com/product/isbn/9781466561595},
## }
```

To use citation entries generated from `toBibtex()`, you have to copy the output to a `.bib` file and save it in your working directory. You will then be able to cite references found in this document directly in your R Markdown. This can be done by adding the following code to your `packages` R code chunk:

```
# Generate BibTex citation file for all R packages used to
# produce report
knitr::write_bib(.packages(), file = "packages.bib")
```

The `.packages()` argument returns the names of all packages loaded in the current R session. This makes sure all packages being used will have their citation entries written to the `.bib` file. Finally, to be able to cite those references (see Citation identifier) in your text, the YAML metadata section has to be edited. See Appendix 1 for a full list of references associated to the R packages used to generate this report.

4.5 Generate Appendix with citations for all R packages

Although a bibliography/references section will be provided at the end of your document to cite in text references (see References), it is customary to add citations for all R packages used to generate the research in Appendix 1. We will learn here the procedure to assemble such Appendix.

4.5.1 Step-by-step procedure

1. Include Appendix after bibliography/references section.
 - This is done by using `<div id="refs"></div>` as shown below:

```
# References

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

# (APPENDIX) Appendix {-}

# Appendix 1
```

2. Insert an R code chunk directly under `# Appendix` to read in and print citations saved in `packages.bib`. This is done as follows:

```

### Load R package
library("knitcitations")
### Process and print citations in packages.bib Clear all
### bibliography that could be in the cash
cleanbib()
# Set pandoc as the default output option for bib
options(citation_format = "pandoc")
# Read and print bib from file
read.bibtex(file = "packages.bib")

```

3. Edit your R code chunk options line as follows to correctly print out references:

```
{r generateBibliography, results="asis", echo=FALSE, warning = FALSE, message=FALSE}
```
4. Knit your code to check that it produces the right output (see Knitting procedure). See Appendix 1 to get an idea of the output.

4.6 Generate Appendix with R package versions used to produce report

In addition to providing citations to R packages, you might also would like to provide full information on R package versions and your operating systems. With R, the simplest (but a useful and important) approach to document your R environment is to report the output of `sessionInfo()` (or `devtools::session_info()`). Among other information, this will show all the packages and their versions that are loaded in the session you used to run your analysis. If someone wants to recreate your analysis, they will know which packages they will need to install, what versions and on which operating systems the code was executed. For instance, here is the output of `sessionInfo()` showing the R version and packages that I used to create this document:

```

sessionInfo()

## R version 3.6.1 (2019-07-05)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Mojave 10.14.6
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base

```

```
##
## other attached packages:
## [1] devtools_2.2.1      usethis_1.5.1      knitr_1.29
## [4] htmltools_0.4.0     prettydoc_0.3.1    magrittr_1.5
## [7] dplyr_0.8.3         kableExtra_1.1.0   formattable_0.2.0.1
## [10] bookdown_0.17       rmarkdown_2.1
##
## loaded via a namespace (and not attached):
## [1] Rcpp_1.0.3          lubridate_1.7.4    prettyunits_1.1.1 ps_1.3.0
## [5] assertthat_0.2.1    rprojroot_1.3-2    digest_0.6.23      R6_2.4.1
## [9] plyr_1.8.5          backports_1.1.5    evaluate_0.14      httr_1.4.1
## [13] pillar_1.4.3        rlang_0.4.3        rstudioapi_0.10    callr_3.4.1
## [17] desc_1.2.0          webshot_0.5.2      RefManageR_1.2.12 readr_1.3.1
## [21] stringr_1.4.0       htmlwidgets_1.5.1 munsell_0.5.0      compiler_3.6.1
## [25] xfun_0.16           pkgconfig_2.0.3    pkgbuild_1.0.6     tidyselect_0.2.5
## [29] tibble_2.1.3        fansi_0.4.1        viridisLite_0.3.0 crayon_1.3.4
## [33] withr_2.1.2         jsonlite_1.6        lifecycle_0.1.0    formatR_1.7
## [37] scales_1.1.1        bibtex_0.4.2.2     cli_2.0.1          stringi_1.4.5
## [41] fs_1.3.1            remotes_2.1.0      testthat_2.3.1     xml2_1.2.2
## [45] ellipsis_0.3.0      vctrs_0.2.2        tools_3.6.1        glue_1.3.1
## [49] purrr_0.3.3         hms_0.5.3          processx_3.4.1     pkgload_1.0.2
## [53] yaml_2.2.0          colorspace_1.4-1    sessioninfo_1.1.1 rvest_0.3.5
## [57] memoise_1.1.0
```

I have also used the approach described above to add this information in Appendix 2.

5 All set, good to go!

We have now set our R Markdown environment and can start populating it! This means that you will be inserting your text and other code chunks directly under the `packages` code chunk. The References section constitutes the end of the main body of your document. If you want to add Appendices, do so under Appendix 1, appendices will be labelled differently from the main body of the document.

6 Insert tables and figures in R Markdown document

6.1 Tables

There will be more details about figures in chapter 9; however this tutorial introduces key concepts related to table making in R Markdown, more specifically on the following topics:

- Creating a table in R.

Table 1: Grading scale applied in this class.

Percentage	Grade
100-98	A+
97.9-93	A
92.9-90	A-
89.9-88	B+
87.9-83	B
82.9-80	B-
79.9-78	C+
77.9-73	C
72.9-70	C-
69.9-68	D+
67.9-60	D
59.9-0	F

- Assigning a table caption.
- Providing a unique label to the R code chunk allowing further cross-referencing in the text.
- Displaying the table in the document.

6.1.1 Step-by-step protocol


Here, you will learn the R Markdown syntax and R code required to replicate the grading scale presented in the Syllabus (see Table 1):

1. Use the same `.Rmd` document as above to practice working with tables.
2. Add a first-level header entitled **Tables**.
3. Insert an R code chunk under your header by clicking on the **Insert** button in the editor toolbar.
4. Write the following R code in the code chunk:



```
grades <- data.frame(Percentage = c("100-98", "97.9-93", "92.9-90",
  "89.9-88", "87.9-83", "82.9-80", "79.9-78", "77.9-73", "72.9-70",
  "69.9-68", "67.9-60", "59.9-0"), Grade = c("A+", "A", "A-",
  "B+", "B", "B-", "C+", "C", "C-", "D+", "D", "F"))
knitr::kable(grades, caption = "Grading scale applied in this class.") %>%
  kable_styling(c("striped", "scale_down"))
```

5. Edit the code chunk options line by adding the following arguments (each argument should be separated by a comma):
 - `echo=FALSE`

6. Add the unique label `tabgrades` in the chunk options line (just after `{r}`) to enable further cross-referencing.
7. Test your code to check that it produces the expected table (using the Run button).
8. Knit your document using the Knit button  on the editor toolbar (remember to select your output format; Figure 2).

6.2 Figures

There will be more details about figures in chapter 10; however this tutorial introduces key concepts related to figure making in R Markdown, more specifically on the following topics:

- Creating a figure in R (based on the `cars` dataset; Figure 3).
- Assigning a figure caption.
- Providing a unique label to the R code chunk allowing further cross-referencing in the text.
- Displaying the figure in the document.

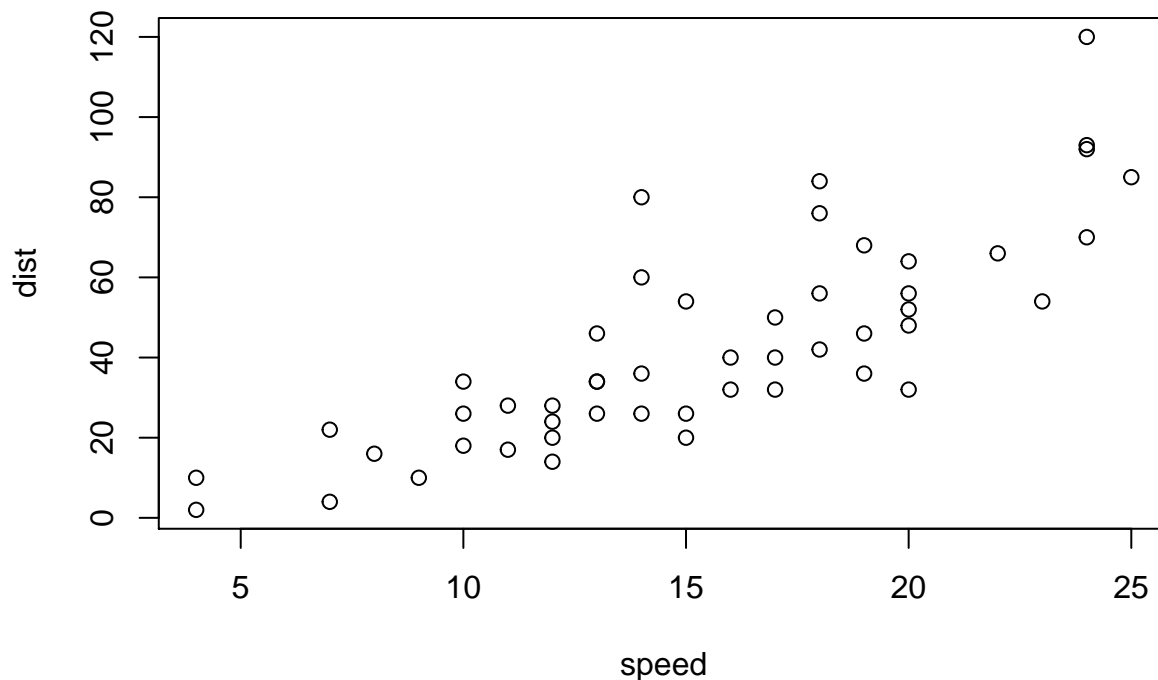


Figure 3: Plot of cars' speed in relation to distance.


6.2.1 Step-by-step protocol

Here, you will learn the R Markdown syntax and R code required to replicate Figure 3:

1. Use the same `.Rmd` document as above to practice working with figures.
2. Add a first-level header entitled **Figures**.

3. Insert an R code chunk under your header by clicking on the **Insert** button in the editor toolbar.
4. Write the following R code in the code chunk:

```
summary(cars)
plot(cars)
```

5. Edit the code chunk options line by adding the following arguments (each argument should be separated by a comma):
 - `echo=FALSE`
 - `results="hide"`
 - `fig.cap="Plot of cars' speed in relation to distance."`
 - `out.width = '100%'`
6. Add the unique label `cars` in the chunk options line (just after `{r}`) to enable further cross-referencing.
7. Test your code to check that it produces the expected plot (using the **Run** button).
8. Knit your document using the **Knit** button  on the editor toolbar (remember to select your output format; Figure 2).

7 Cross-reference tables and figures in the text

Cross-referencing tables and figures in the main body of your R Markdown document can easily be done using the `\@ref()` function implemented in the *bookdown* package.

7.1 General syntax

The general syntax is as follows:

```
# Cross-referencing tables in main body of text
\@ref(tab:code_chunk_ID)

# Cross-referencing figures in main body of text
\@ref(fig:code_chunk_ID)
```

7.2 Step-by-step procedure

To cross-reference the `tabgrades` table type:

`\@ref(tab:tabgrades)`, which translates into 1.

To cross-reference the `cars` figure type:

`\@ref(fig:cars)`, which translates into 3.

Note the this syntax doesn't automatically include the **Table** or **Figure** handles in front of the cross-reference. You will have to manually add **Table** or **Figure** in front of your `\@ref()` function.

8 Citing references in text and adding a bibliography/references section

8.1 The bibliography file

To cite references in the R Markdown document those have to be saved in a bibliography file using the **BibTeX** format. Other formats can be used, but the **BibTeX** format is open-source and easy to edit. Please see this webpage for more details on other formats: https://rmarkdown.rstudio.com/authoring_bibliographies_and_citations.html

Most journals allow saving citation of publications directly in **BibTeX** format, but when this feature is not available formats can be converted using online services (e.g. EndNote to BibTeX: <https://www.bruot.org/ris2bib/>).

8.1.1 Procedure to do prior to citing references in an R Markdown document

1. Save all your **BibTeX** references in a text file and make sure to add the `.bib` extension.
2. This file has to be deposited in the same folder as your `.Rmd` file.
3. You will be inserting the name of your references file in the YAML metadata section.
4. You can visit this webpage and click on the **Cite** icon to download a citation in `.bibtex` format. More details on the **BibTeX** format is provided below.
5. References formatted in **BibTeX** format are available in associated file:
 - `Bibliography_Reproducible_Science_2.bib`.

8.2 Specifying a bibliography in the R Markdown file

The Pandoc program can automatically generate citations in the text and a bibliography/references section following various journal styles. In order to use this feature, you need to specify a bibliography file in the YAML metadata section.

8.3 BibTeX format

Please find below an example of a reference formatted in **BibTeX** format:

```
# Example of BibTeX format for Baker (2016) published in Nature
@Article{Baker_2016,
  doi = {10.1038/533452a},
  url = {https://doi.org/10.1038/533452a},
  year = {2016},
  month = {may},
  publisher = {Springer Nature},
  volume = {533},
  number = {7604},
  pages = {452--454},
  author = {Monya Baker},
  title = {1,500 scientists lift the lid on reproducibility},
  journal = {Nature},
}
```

8.4 Citation identifier

The unique citation identifier of a reference (`Baker_2016` in the example above) is set by the user in the BibTeX citation file (see first line in the example provided above). This unique identifier is used to refer to the reference/publication in the R Markdown document and also allows citing references and generating the bibliography/references section.

8.5 Citing references

Citations go inside square brackets (`[]`) and are separated by semicolons. Each citation must have a key, composed of `@` + the citation identifier (see above) as stored into the BibTeX file.

Please find below some examples on citation protocols:

```
#Syntax
Blah blah [see @Baker_2016, pp. 33-35; also @Smith2016, ch. 1].
Blah blah [@Baker_2016; @Smith2016].
```

Once knitted (using the  button), the above code/syntax turns into:

Blah blah (see Baker, 2016 pp. 33–35; also Smith et al., 2016, ch. 1).

Blah blah (Baker, 2016; Smith et al., 2016).

A minus sign (-) before the `@` will suppress mention of the author in the citation. This can be useful when the author is already mentioned in the text:

```
#Syntax
Baker says blah blah [-@Baker_2016].
```


Once knitted, the above code/syntax turns into:

Baker says blah blah (2016).

You can also write an in-text citation, as follows:

```
#Syntax  
@Baker_2016 says blah.  
@Baker_2016 [p. 1] says blah.
```

Once knitted, the above code/syntax turns into:

Baker (2016) says blah.

Baker (2016 p. 1) says blah.

8.5.1 Practice citing references

Students have to use their .Rmd document to practice citing references in the text using procedures described above. To clearly define where you practice citing references, please do so under a `Citing references` header.

8.6 Adding a bibliography/references section

Upon knitting, a bibliography/references section will be automatically generated at the end of the document. Usually, we recommend adding a References header just after the last paragraph of the document as displayed below:

```
last paragraph...
```

```
# References
```

The bibliography will be inserted after this header (please see References section of this tutorial for more details).

9 Formatting citations to journal styles

In this section, we are studying how your bibliography can be automatically formatted following a journal style. This is achieved by providing the name of a citation style file (containing the protocol to format citations and bibliography following a journal style) in the YAML metadata section.

9.1 What is the citation style language (CSL)?

The **Citation Style Language (CSL)** was developed by an open-source project and aims at facilitating scholarly publishing by automating the formatting of citations and bibliographies. This project has developed the CSL and maintains a crowd sourced repository with over 8000 free CSL citation styles. Please see the following website for more details: <https://citationstyles.org>

9.2 CSL repositories

There are two main CSL repositories:


- **GitHub Repository:** <https://github.com/citation-style-language/styles>
- **Zotero Style Repository:** <https://www.zotero.org/styles>

9.3 How to use a CSL file in an R Markdown to format citations and bibliography?

Please follow the steps below to format your citations and bibliography following the citation style provided in a CSL file:

1. Download the CSL file using repositories provided above. Some journals provide their CSL files on their websites (one has been made available for you to use in the associated files: `AmJBot.csl`).
2. Save the CSL file in the same working directory as the `.Rmd` file.
3. Edit the YAML metadata section as follows to specify the CSL file:

```
# Add a "csl" argument and provide name of the CSL file (here AmJBot.csl)
---
title: "Sample Document"
output:
  bookdown::html_document2:
    toc: TRUE
  bookdown::pdf_document2:
    toc: TRUE
bibliography: bibliography.bib
csl: AmJBot.csl
---
```

4. Knit the R Markdown document using the Knit  Knit button. The Pandoc program will use the information stored in the YAML metadata section to format the bibliography (citations and bibliography section) following the citation style provided in the CSL file. Do not forget to add a **References** header at the end of your `.Rmd` document.

10 References

- BAKER, M. 2016. 1,500 scientists lift the lid on reproducibility. *Nature* 533: 452–454. Available at: <https://doi.org/10.1038/533452a>.
- SMITH, J.F., T.H. PARKER, S. NAKAGAWA, J. GUREVITCH, ECOLOGY, and T.(. FOR T. IN EVOLUTION) WORKING GROUP. 2016. Promoting transparency in evolutionary biology and ecology. *Systematic Botany* 41: 495–497. Available at: <http://www.bioone.org/doi/abs/10.1600/036364416X692262>.
- XIE, Y. 2020a. Bookdown: Authoring books and technical documents with r markdown. Available at: <https://CRAN.R-project.org/package=bookdown>.
- XIE, Y. 2016. Bookdown: Authoring books and technical documents with R markdown. Chapman; Hall/CRC, Boca Raton, Florida. Available at: <https://github.com/rstudio/bookdown>.
- XIE, Y. 2015. Dynamic documents with R and knitr. 2nd ed. Chapman; Hall/CRC, Boca Raton, Florida. Available at: <https://yihui.org/knitr/>.
- XIE, Y. 2020b. Knitr: A general-purpose package for dynamic report generation in r. Available at: <https://CRAN.R-project.org/package=knitr>.
- XIE, Y., J. ALLAIRE, and G. GROLEMUND. 2018. R markdown: The definitive guide. Chapman; Hall/CRC, Boca Raton, Florida. Available at: <https://bookdown.org/yihui/rmarkdown>.

A Appendix 1

Citations of all R packages used to generate this report.

- [1] J. Allaire, Y. Xie, J. McPherson, et al. *rmarkdown: Dynamic Documents for R*. R package version 2.1. 2020. <URL: <https://CRAN.R-project.org/package=rmarkdown>>.
- [2] S. M. Bache and H. Wickham. *magrittr: A Forward-Pipe Operator for R*. R package version 1.5. 2014. <URL: <https://CRAN.R-project.org/package=magrittr>>.
- [3] C. Boettiger. *knitcitations: Citations for ‘Knitr’ Markdown Files*. R package version 1.0.10. 2019. <URL: <https://CRAN.R-project.org/package=knitcitations>>.
- [4] Y. Qiu. *prettydoc: Creating Pretty Documents from R Markdown*. R package version 0.3.1. 2019. <URL: <https://CRAN.R-project.org/package=prettydoc>>.
- [5] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing. Vienna, Austria, 2019. <URL: <https://www.R-project.org/>>.
- [6] K. Ren and K. Russell. *formattable: Create ‘Formattable’ Data Structures*. R package version 0.2.0.1. 2016. <URL: <https://CRAN.R-project.org/package=formattable>>.

- [7] RStudio and Inc. *htmltools: Tools for HTML*. R package version 0.4.0. 2019. <URL: <https://CRAN.R-project.org/package=htmltools>>.
- [8] H. Wickham and J. Bryan. *usethis: Automate Package and Project Setup*. R package version 1.5.1. 2019. <URL: <https://CRAN.R-project.org/package=usethis>>.
- [9] H. Wickham, R. François, L. Henry, et al. *dplyr: A Grammar of Data Manipulation*. R package version 0.8.3. 2019. <URL: <https://CRAN.R-project.org/package=dplyr>>.
- [10] H. Wickham, J. Hester, and W. Chang. *devtools: Tools to Make Developing R Packages Easier*. R package version 2.2.1. 2019. <URL: <https://CRAN.R-project.org/package=devtools>>.
- [11] Y. Xie. *bookdown: Authoring Books and Technical Documents with R Markdown*. ISBN 978-1138700109. Boca Raton, Florida: Chapman and Hall/CRC, 2016. <URL: <https://github.com/rstudio/bookdown>>.
- [12] Y. Xie. *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.17. 2020. <URL: <https://CRAN.R-project.org/package=bookdown>>.
- [13] Y. Xie. *Dynamic Documents with R and knitr*. 2nd. ISBN 978-1498716963. Boca Raton, Florida: Chapman and Hall/CRC, 2015. <URL: <https://yihui.org/knitr/>>.
- [14] Y. Xie. “knitr: A Comprehensive Tool for Reproducible Research in R”. In: *Implementing Reproducible Computational Research*. Ed. by V. Stodden, F. Leisch and R. D. Peng. ISBN 978-1466561595. Chapman and Hall/CRC, 2014. <URL: <http://www.crcpress.com/product/isbn/9781466561595>>.
- [15] Y. Xie. *knitr: A General-Purpose Package for Dynamic Report Generation in R*. R package version 1.29. 2020. <URL: <https://CRAN.R-project.org/package=knitr>>.
- [16] Y. Xie, J. Allaire, and G. Golemund. *R Markdown: The Definitive Guide*. ISBN 9781138359338. Boca Raton, Florida: Chapman and Hall/CRC, 2018. <URL: <https://bookdown.org/yihui/rmarkdown>>.
- [17] H. Zhu. *kableExtra: Construct Complex Table with ‘kable’ and Pipe Syntax*. R package version 1.1.0. 2019. <URL: <https://CRAN.R-project.org/package=kableExtra>>.

B Appendix 2

Version information about R, the operating system (OS) and attached or R loaded packages. This appendix was generated using `sessionInfo()`.

```
## R version 3.6.1 (2019-07-05)
## Platform: x86_64-apple-darwin15.6.0 (64-bit)
## Running under: macOS Mojave 10.14.6
##
## Matrix products: default
```

```

## BLAS: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/3.6/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods   base
##
## other attached packages:
## [1] devtools_2.2.1      usethis_1.5.1      knitcitations_1.0.10
## [4] htmltools_0.4.0     prettydoc_0.3.1     magrittr_1.5
## [7] dplyr_0.8.3         kableExtra_1.1.0    formattable_0.2.0.1
## [10] bookdown_0.17       rmarkdown_2.1       knitr_1.29
##
## loaded via a namespace (and not attached):
## [1] Rcpp_1.0.3          lubridate_1.7.4     prettyunits_1.1.1  ps_1.3.0
## [5] assertthat_0.2.1    rprojroot_1.3-2     digest_0.6.23      R6_2.4.1
## [9] plyr_1.8.5          backports_1.1.5     evaluate_0.14      httr_1.4.1
## [13] pillar_1.4.3        rlang_0.4.3         rstudioapi_0.10    callr_3.4.1
## [17] desc_1.2.0          webshot_0.5.2       RefManageR_1.2.12  readr_1.3.1
## [21] stringr_1.4.0       htmlwidgets_1.5.1   munsell_0.5.0      compiler_3.6.1
## [25] xfun_0.16           pkgconfig_2.0.3     pkgbuild_1.0.6     tidyselect_0.2.5
## [29] tibble_2.1.3        fansi_0.4.1         viridisLite_0.3.0  crayon_1.3.4
## [33] withr_2.1.2         jsonlite_1.6        lifecycle_0.1.0    formatR_1.7
## [37] scales_1.1.1        bibtex_0.4.2.2      cli_2.0.1          stringi_1.4.5
## [41] fs_1.3.1            remotes_2.1.0       testthat_2.3.1     xml2_1.2.2
## [45] ellipsis_0.3.0      vctr_0.2.2          tools_3.6.1        glue_1.3.1
## [49] purrr_0.3.3         hms_0.5.3           processx_3.4.1     pkgload_1.0.2
## [53] yaml_2.2.0          colorspace_1.4-1    sessioninfo_1.1.1  rvest_0.3.5
## [57] memoise_1.1.0

```