Bookdown Template

subtitle here

Jack Dougherty   Ilya Ilyankou

2020-07-29

Contents

List of Tables

List of Figures

Preface

This test book was last updated on 29 Jul 2020

Insert additional preface items below (acknowledgements, etc.), with non-numbering symbols `{-}` to ensure that the preface is not numbered, and the first chapter is properly numbered as chapter one.

Reminder: Bookdown does not auto-number figures in the `index.Rmd` file, which serves as a "preface" file.

Acknowledgements

Cupcake ipsum dolor sit amet jelly beans wafer pudding. Bear claw lemon drops carrot cake pie wafer chocolate jelly cheesecake. Chocolate cheesecake chocolate bar sugar plum sweet dessert tart. Tootsie roll bear claw chocolate bar wafer powder sugar plum tiramisu bear claw gummies. Tart macaroon pastry lemon drops candy tootsie roll chocolate candy canes lollipop. Pudding fruitcake bear claw sweet cake cupcake. Chupa chups pudding candy canes chupa chups powder jujubes chocolate cake cotton candy jelly.

Chapter 1

Publishing with Bookdown

This open-access book is built with free-to-use, open-source tools—primarily Bookdown, GitHub, and Zotero—and this chapter explains how, so that readers may do it themselves and share their knowledge to improve the process. In addition to our notes below, see also Yihui Xie's more comprehensive Bookdown guide.[1]

Our broad goal is an efficient workflow to compose one document in the easy-to-write Markdown format that Bookdown generates into multiple book products: an HTML web edition to read online, a PDF print edition for traditional book publishing, a Microsoft Word edition for editors who request it for copyediting, and option for other formats as desired.

Since Bookdown is an R code package, we composed the book manuscript in R-flavored Markdown, with one file (.Rmd) for each chapter. We use Bookdown to build these files in its GitBook style as a set of static HTML pages, which we upload to our GitHub repository. Readers can view the open-access web edition of the book at our custom domain: https://HandsOnDataViz. Also, we use Bookdown to build additional book outputs (PDF, MS Word, Markdown) and upload these to the `docs` folder of our GitHub repository, so that our O'Reilly Media editor may download and comment on the manuscript as we revise. Finally, we also have the option to use Pandoc alone to convert the full-book Markdown file (.md) into an AsciiDoc file (.asciidoc) for easier importing into the O'Reilly Atlas platform. See some caveats and workarounds below.

File Organization and Headers

We organized the GitHub repository for this book as a set of .Rmd files, one for each chapter. As co-authors, we are careful to work on different chapters of the

---

[1]Yihui Xie, *Bookdown: Authoring Books and Technical Documents with R Markdown*, 2018, https://bookdown.org/yihui/bookdown/

book, and to regularly push our commits to the repo. Only one of us regularly builds the book with Bookdown to avoid code merge conflicts.

Bookdown assigns a default ID to each header, which can be used for cross-references. The default ID for `# Introduction` is `{#introduction}`, and the default ID for `## Part One` is `{#part-one}`, where spaces are replaced by dashes. But we do *not* rely on default IDs because they might change due to editing or contain duplicates across the book.

Instead, we *manually assign a unique ID* to each first- and second-level header in the following way. Note that the `{-}` symbol, used alone or in combination *with a space* and a unique ID, prevents auto-numbering in the second- thru fourth-level headers:

```
# Top-level chapter title {#unique-name}
## Second-level section title {- #unique-name}
### Third-level subhead {-}
#### Fourth-level subhead {-}
```

Also, we match the unique ID keyword to the file name for top-level chapters this way: `01-introduction.Rmd` to keep our work organized. Unique names should contain only *alphanumeric* characters (a-z, A-Z, 0-9) or dashes (-).

In the Bookdown `index.Rmd` for the HTML book output and the PDF output, the `toc_depth: 2` setting displays chapter and section headers down to the second level in the Table of Contents.

The `split_by: section` setting divides the HTML pages at the second-level header, which creates shorter web pages with reduced scrolling for readers. For each web page, the unique ID becomes the file name, and is stored in the `docs` subfolder.

The `number_sections` setting is true for the HTML and PDF editions, and given the `toc_depth: 2`, this means that they will display two-level chapter-section numbering (1.1, 1.2, etc.) in the Table of Contents. Note that `number_sections` must be true to display Figure and Table numbers in x.x format, which is desired for this book. See relevant settings in this excerpt from `index.Rmd`:

```
output:
  bookdown::gitbook:
    ...
    toc_depth: 2
    split_by: section
    number_sections: true
    split_bib: true
    ...
```

```
bookdown::pdf_book:
  toc_depth: 2
  number_sections: true
```

Note that chapter and section numbering do *not* appear automatically in the MS Word output unless you supply a reference.docx file, as described below:

- https://bookdown.org/yihui/rmarkdown/word-document.html
- https://stackoverflow.com/questions/52924766/numbering-and-referring-sections-in-bookdown
- https://stackoverflow.com/questions/50609212/caption-styles-for-word-document2-in-bookdown

In the `_bookdown.yml` settings, all book outputs are built into the `docs` subfolder of our GitHub repo, as shown in this excerpt:

```
output_dir: "docs"
book_filename: "bookdown-template"
language:
  label:
    fig: "Figure "
chapter_name: "Chapter "
```

In our GitHub repo, we set GitHub Pages to publish to the web using `master/docs`, which means that visitors can browse the source files at the root level, and view the HTML web pages hosted in the `docs` subfolder. We use the GitHub Pages custom domain setting so that the HTML edition is available at https://HandsOnDataViz.org.

The `docs` subfolder also may contain the following items, which are *not* generated by Bookdown and need to be manually created:

- `CNAME` file for the custom domain, generated by GitHub Pages.
- `.nojekyll` invisible empty file to ensure speedy processing of HTML files by GitHub Pages.
- `404.html` custom file to redirects any mistaken web addresses under the domain back to the `index.html` page.

One more option is to copy the Google Analytics code for the web book, paste it into an HTML file in the book repo, and include this reference in the `index.Rmd` code:

```
output:
  bookdown::gitbook:
  ...
  includes:
    in_header: google-analytics.html
```

Style Guide

View the source code to better understand how this page was composed at: https://github.com/HandsOnDataViz/bookdown-template/blob/master/01-bookdown.Rmd

This book is composed in R-flavored Markdown (.Rmd), and each paragraph begins on a separate line. O'Reilly style guide prefers *italics* rather than bold. Use single back tics to display a monospaced `code` word.

Insert an embedded link to O'Reilly. This appears as a colored clickable link in HTML and Word editions, and a non-colored but clickable link in the PDF edition. According to O'Reilly Atlas documentation, the AsciiDoc version should automatically unfurl for the printed edition.

For lists, always insert a blank line *before* the items, unless they appear directly after hashtag header.

- unordered
- list

1. ordered
2. list

Dashes:

- Use a hyphen (1 dash) for hyphenated words, such as two-thirds or dog-friendly hotel.
- Use an en-dash (2 dashes) for ranges, such as the May–September magazine issue.
- Use an em-dash (3 dashes) to insert an additional thought—like this—in a sentence.

Insert `TODO` to note items to finish or review with co-author or editor.

Insert three back tics to insert a code block. Check character line length limits in O'Reilly style guide:

```
<link rel="stylesheet" href="https://unpkg.com/leaflet@1.6.0/dist/leaflet.css" />
<script src="https://unpkg.com/leaflet@1.6.0/dist/leaflet.js"></script>
```

Conditional Formatting

Conditional formatting offers the option to display text or images in some editions, but not other editions. Options:

1. Insert a HTML code comment `<!-- Comment -->` in the .Rmd file to hide a few lines of text. This appears as commented-out text in the HTML and .md formats, is not displayed in the HTML browser, and does not appear in any way in the PDF, MS Word or AsciiDoc formats.

Demo:

2. R package function `is_[html/latex]_output` allows conditional output for different book products, such as text that should appear in the HTML edition but not the PDF edition, or vice versa.

Demos:

This line appears in the PDF and Word versions, and is commented-out in the HTML and Markdown and AsciiDoc versions.

TODO: Create conditional formatting that displays *only* in the HTML edition, and allows the inclusion of R code-chunks to conditionally display images. See links for more complex conditional formatting:

- https://stackoverflow.com/questions/56808355/how-to-conditionally-process-sections-in-rmarkdown
- https://bookdown.org/yihui/rmarkdown-cookbook/latex-html.html
- https://blog.earo.me/2019/10/26/reduce-frictions-rmd/
- https://stackoverflow.com/questions/53861244/html-specific-section-in-bookdown
- https://stackoverflow.com/questions/41084020/add-a-html-block-above-each-chapter-header
- https://stackoverflow.com/questions/45360998/code-folding-in-bookdown

3. Option to customize the `style.css` code for the HTML book.

4. Option to add headers, footers, preambles to the HTML or LaTeX versions.

5. Build different versions of the HTML and LaTeX (PDF) books using different chapters by listing them in order in the `_bookdown.yml` file:

```
rmd_files:
  html: ["index.Rmd", "abstract.Rmd", "intro.Rmd"]
  latex: ["abstract.Rmd", "intro.Rmd"]
```

Cross-references

In order to cross-reference in Bookdown, assign a unique name or R code-chunk label to each chapter, section, figure, and table. Unique names and labels should contain only *alphanumeric* characters (a-z, A-Z, 0-9) or dashes (-).

To cross-reference any *chapter or section*, and allow readers to jump there, use a HTML link with the unique name, such as `index.html` or `style-guide.html`. Demos:

- See Preface
- See "Style Guide" in Chapter x.

Contrary to the Bookdown manual, *avoid* using Bookdown unique ID links to cross-reference chapters or sections, because these create extraneous and imprecise URLs, such as this example: Chapter 1 "Style Guide"

To cross-reference figures and tables, and display their auto-number and allow readers to jump there, write a call-out with a Bookdown reference to a code-chunk label, such as `See Figure \@ref(fig:sample-map)` or `See Table \@ref(tab:left-table)`. Demos:

- See Figure 1.1.
- See Table 1.2.

Cross-reference interactivity varies by output:

- In HTML, all cross-refs are clickable.
- In PDF, all cross-refs are clickable (except chapter-level HTML links).
- In Word, no cross-refs are clickable (unless this varies with reference.docx).
- TBA with Markdown (.md) and AsciiDoc.

When writing cross-references in the text, the O'Reilly Style Guide prefers live cross references (e.g., "see Figure 2-1"), but if not feasible, use "preceding" or "following" because physical placement of elements may vary across print and digital formats. *Avoid* using "above" or "below."

Images

Create high-resolution color screenshots and other static images in .png or .jpg format, with tight cropping on a high-resolution Retina monitor, typically at 144 ppi. Save items into the `images` subfolder by chapter. Make sure that color images include high contrast and/or shading, because they will be converted to

grayscale by the publisher for the print book. Write file names in lowercase with dashes (not spaces) and begin with keyword of relevant section to keep related images grouped together. Despite being in separate folders, avoid duplicate image file names across the book. Avoid numbering images since they may not match the final sequence.

If a screenshot requires additional artwork or text for the HTML edition, make a copy of the original and add `-ann` to note that this version is annotated, save into the same folder with the same root file name, and use in the code-chunk image pathnames. The publisher will use the original image and add their own artwork for their editions.

If an image is larger than approximately 300px on either side, one more option is to reduce the image size in the PDF version. Use Photoshop (*not* Preview) to reduce the image size, and save a copy with the same file name with the .pdf extension into the folder. In some cases, the folder will contain only one version of each image, but in other cases it may contain up to three versions of an image:

```
images/05-chart/design-no-junk.png
images/05-chart/design-no-junk-ann.png
images/05-chart/design-no-junk-ann.pdf
```

If creating images to appear as the same size in sequence, add a code-comment with the image width, height, and resolution as a reminder to make others match up, like this:

```
<!-- Images below are 200x200 at 300 resolution -->
```

In this book, use *Markdown formatting only for images that appear inside tables* or *do not require captions or figure numbering*, and leave the caption field blank, like this example:

Co-Authors	About Us
	About Jack Dougherty
	About Ilya Ilyankou

Although Markdown formatting offers a simple syntax that easily converts into other formats with Bookdown/Pandoc, there is no auto-numbering in the HTML

edition, while auto-numbering appears in the PDF edition, and numbered figures are required by the publisher. Furthermore, Markdown formatting does not allow conditional output.

Images using R code-chunks

For these reasons, this book *primarily uses R code-chunk formatting for images*. The syntax is more complex but supports auto-numbering in HTML and PDF, and conditional output for interactive and static images. Note that R code-chunk images do *not* easily convert with Pandoc from Markdown to AsciiDoc, but "Figure x Caption" appears as a placeholder.

Auto-numbering appears in `Figure x.x` format in HTML and PDF, but `Figure x` format in MS Word. Note that Word formatting can be changed with reference.docx.

Write R code-chunk labels that follow the image file name, and avoid duplicate labels across the book:

```
ref:design-no-junk
```

```
images/05-chart/design-no-junk.png
```

Do not insert spaces inside the `ref:chunk-label` for the caption, but add a blank line to separate it from the code-chunk. After the code-chunk, add another blank line.

For each figure, manually add a cross-reference call-out and insert `fig.pos='h'` to place the image "here" on the page in the PDF output, to *attempt* to avoid PDF floating. Ignore the Bookdown LaTeX warning message "h float specifier changed to ht." See more at https://bookdown.org/yihui/bookdown/figures. html and https://bookdown.org/yihui/rmarkdown-cookbook/figure-placement. html

This Bookdown `index.Rmd` file includes two global R code-chunk settings immediately after the first header. One setting displays each code-chunk image without a code echo. The other setting automatically inserts the PDF version of an PNG/JPG image, whenever it exists, in the PDF output, which allows us to manually reduce the image sizes for the PDF book. Read more about these options in this Bookdown chapter: https://bookdown.org/yihui/ bookdown/figures.html.

```
{r setup, include=FALSE}
knitr::opts_chunk$set(echo = FALSE)
options(knitr.graphics.auto_pdf = TRUE)
```

Demo: R code-chunk for small static image

Small is defined as each side less than 300px, as shown in Figure 1.1.

Figure 1.1: Caption here. Markdown embedded links are acceptable.

R code-chunk for larger static image using out.width and PDF img

For larger images, where one side is greater than 300px, set the out.width to a pixel number for ideal display in the HTML edition. Also, if needed, copy the image, use Photoshop to create a smaller image size, and save with same file name and a .pdf extension for auto-substitution in the PDF output . . . as shown in Figure 1.2.

Figure 1.2: Using out.width=200 and smaller PDF image size.

R code-chunks allow more complex conditional formatting, where an interactive map or animated GIF or YouTube video clip appears in the web version, and a static image with an embedded link appears in the PDF and MS Word outputs. To change the height of the default 400px iframe, add the new

height to `include_url` as shown in the examples. However, to change the width of the default 675px iframe to less than 100 percent, add a line in a `custom-scripts.html` file.

Demo: R code-chunk for HTML iframe and static image
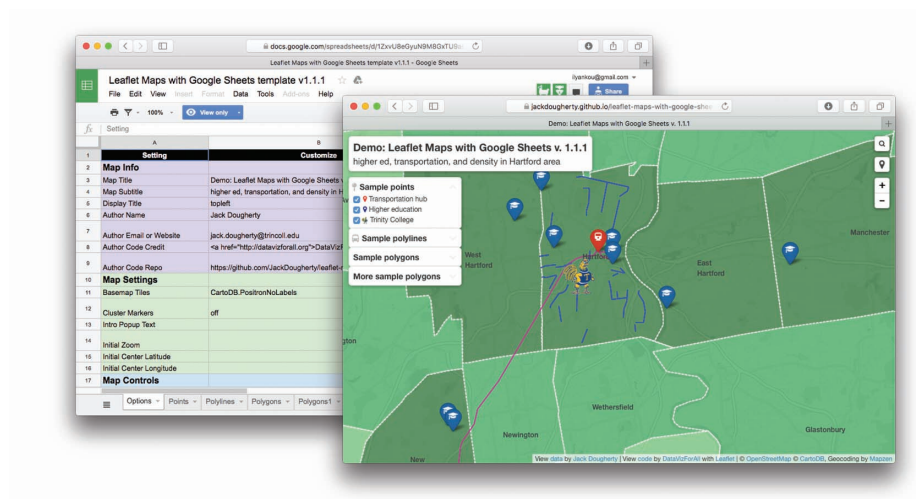
. . . as shown in Figure 1.3.

Figure 1.3: Caption here, and add embedded link to explore the full-screen interactive map

Demo: R code-chunk for GIF animation and static image

. . . as shown in Figure 1.4.

Demo: R code-chunk for Youtube video and static image

Be sure to use the *embed* link from the YouTube *share* button.

. . . as shown in the video 1.5.

Figure 1.4: Caption here, with embedded link to animated GIF.

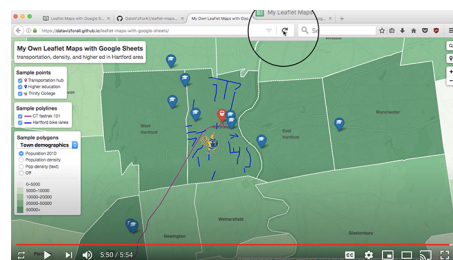
Figure 1.5: Caption here, with embedded link to the YouTube video.

Demo: R code-chunk for YouTube video in HTML, with NO static image in PDF

Tables

Create tables in Markdown format, since it produces good output for HTML, PDF, Word, and Markdown. Use a tool such as Tables Generator to import significant table data in CSV format, format the column alignment as desired, and press Generate button to create table in Markdown format. For significant table data, save the CSV version in a GitHub repo for potential later use.

Add the Markdown table code shown below to auto-number (Table x) in HTML, PDF, Word.

Table 1.2: Left-justify content, remember blank Line

Much Much Longer Header	Short Header	Short Header
Left-justify text content with left-colons	Less	Here
Use more hyphens to grant more space to some columns	Less	Here

Table 1.3: Right-justify content, remember blank line

Header1	Header2	Header3
123	456	789
Right-justify	numerical content	with right-colons
Use equal hyphens	to make equal space	for all columns

Workaround for Markdown-to-AsciiDoc: Currently, our attempt to use Pandoc to directly convert a Bookdown-generated Markdown file to AsciiDoc fails because Bookdown creates the .md file with tables in .html format, not Markdown. Our workaround is to paste the individual Markdown-formatted tables directly from the .Rmd into the large .md file prior to converting with Pandoc to AsciiDoc.

Notes and Bibliography

This book displays endnotes for each chapter in the HTML book, and footnotes at the bottom of pages for the PDF and MS Word books, followed by an alphabetized bibliography of all references cited on the last page. The notes and bibliography also appear in the full-book Markdown file.

To create notes, insert citation keys in the text, such as `@huffHowLieStatistics1954`, which are generated by Zotero bibliographic database with the Better Bib-Tex extension, and export these in the *Better BibLaTeX* format into the `dataviz.bib` in the book repo. The repo also contains `.csl` file to generate the notes and bibliography in a specific Chicago-style format, downloaded from the Zotero Styles Repository. These instructions are referenced in the `index.Rmd` file for both the HTML and PDF formats, as shown in these excerpts:

```
bibliography: dataviz.bib
citation-style: chicago-fullnote-bibliography.csl
...
output:
  bookdown::gitbook:
    ...
    pandoc_args: [ "--csl", "chicago-fullnote-bibliography.csl" ]

  bookdown::pdf_book:
    ...
    citation_package: none
    pandoc_args: [ "--csl", "chicago-fullnote-bibliography.csl" ]
```

Here's a text-only note, with no Zotero citation.[2]

To create a note with citations only, separate Zotero/BibTeX citation keys with semi-colons:[3]

Since notes also may include text and punctuation in Markdown syntax, always insert a caret symbol prior to the brackets to demarcate a note:[4]

Note that the `chicago-fullnote-bibliography.csl` format automatically shortens the note after it its first reference.

Custom Blocks

Note: I moved code to `TODO` folder because it's not working as expected for LaTeX PDF output.

Pandoc Conversion

- Download Pandoc

---

[2] This is a note, with no bibliographic reference.

[3] Darrell Huff, *How to Lie with Statistics* (W. W. Norton & Company, 1954–2010), http://books.google.com/books?isbn=0393070875; Mark S. Monmonier, *How to Lie with Maps*, 2nd ed. (University of Chicago Press, 1996), http://books.google.com/books?isbn=0226534219

[4] Compare how "lying" is justified by Huff, *How to Lie with Statistics*, pp. 10-11 and Monmonier, *How to Lie with Maps*, pp. 11-12.

- TODO: Ask Ilya about my Pandoc PATH and/or overwriting older version
- Set Bookdown to build the book as one large Markdown file (docs folder, suffix .md)
- Use command line to navigate to subfolder with `pwd` and `cd`.
- Convert with: `pandoc bookdown-template.md --from markdown --to asciidoc --standalone --output bookdown-template-modified.asciidoc`
- Confirm if AsciiDoc file matches O'Reilly Atlas import style.

Chapter 2

Supplements

Cupcake ipsum dolor sit amet jelly beans wafer pudding. Bear claw lemon drops carrot cake pie wafer chocolate jelly cheesecake. Chocolate cheesecake chocolate bar sugar plum sweet dessert tart. Tootsie roll bear claw chocolate bar wafer powder sugar plum tiramisu bear claw gummies. Tart macaroon pastry lemon drops candy tootsie roll chocolate candy canes lollipop. Pudding fruitcake bear claw sweet cake cupcake. Chupa chups pudding candy canes chupa chups powder jujubes chocolate cake cotton candy jelly.

Section Header

Cupcake ipsum dolor sit amet jelly beans wafer pudding. Bear claw lemon drops carrot cake pie wafer chocolate jelly cheesecake. Chocolate cheesecake chocolate bar sugar plum sweet dessert tart. Tootsie roll bear claw chocolate bar wafer powder sugar plum tiramisu bear claw gummies. Tart macaroon pastry lemon drops candy tootsie roll chocolate candy canes lollipop. Pudding fruitcake bear claw sweet cake cupcake. Chupa chups pudding candy canes chupa chups powder jujubes chocolate cake cotton candy jelly.

A Third-Level Section

Cupcake ipsum dolor sit amet jelly beans wafer pudding. Bear claw lemon drops carrot cake pie wafer chocolate jelly cheesecake. Chocolate cheesecake chocolate bar sugar plum sweet dessert tart. Tootsie roll bear claw chocolate bar wafer powder sugar plum tiramisu bear claw gummies. Tart macaroon pastry lemon drops candy tootsie roll chocolate candy canes lollipop. Pudding fruitcake bear claw sweet cake cupcake. Chupa chups pudding candy canes chupa chups powder jujubes chocolate cake cotton candy jelly.

Figure 2.1: Caption for sample static image using R code-chunk method.

Chapter 3

References

Huff, Darrell. *How to Lie with Statistics.* W. W. Norton & Company, 1954–2010. http://books.google.com/books?isbn=0393070875.

Monmonier, Mark S. *How to Lie with Maps.* 2nd ed. University of Chicago Press, 1996. http://books.google.com/books?isbn=0226534219.

Xie, Yihui. *Bookdown: Authoring Books and Technical Documents with R Markdown*, 2018. https://bookdown.org/yihui/bookdown/.