

汤银才

Bookdown 中文书稿写作手册

目录

表格	iii
插图	v
前言	vii
第 1 章 引言	1
1.1 中国图书出版的变化	1
1.2 统计类图书的核心要素	2
1.3 统计数据分析类图书模板的选择	3
1.3.1 基于纯 $T_{\text{E}}X$ 模板	3
1.3.2 Rmarkdown 与 $T_{\text{E}}X$ 的结合	3
1.3.3 Rmarkdown 向 Bookdown 过渡	4
第 2 章 bookdown 速览	5
2.1 关于 bookdown	5
2.2 快速排版的思路	5
2.3 书的基本设置	6
2.3.1 index.Rmd 文件	6
2.3.2 _bookdown.yml 文件	7
2.3.3 _output.yml 文件	8
2.4 章节结构	10
2.5 书的编译	11
2.6 浮动对象标签与引用汇总	12
第 3 章 Bookdown 中的章节标题	15
3.1 章节标题	15
3.2 章节标题标签的设定与引用	15
第 4 章 Bookdown 中的公式与定理	17
4.1 公式标签的设定	17
4.2 定理标签的设定	18
4.3 定理与公式的引用	18

4.4	数学公式的扩展	18
第 5 章	Bookdown 中的图形	21
5.1	由 R 生成单个图形示例	21
5.2	由 R 生成两个图形并置示例	21
5.3	由 R 生成两个图形堆叠示例	22
5.4	静态图形示例	22
5.5	图形引用	22
第 6 章	Bookdown 中的表格	25
6.1	表格示例 1: 由数据生成表格	25
6.2	表格示例 2: 由数据框构造表格	25
6.3	表格引用	25
第 7 章	Bookdown 中的文本参考	29
7.1	文本标签的设定与引用	29
7.2	文本引用场景	29
第 8 章	Bookdown 中的文献排版	31
8.1	文献库建立样例	31
8.2	文献风格	32
8.3	中文文献风格的设置	32
8.4	文献库的建立工具	33
附录		35
第 A 章	Mathjax 的离线安装与使用	35
A.1	mathjax 说明	35
A.2	调用远程服务器上的 mathjax	36
A.3	mathjax 本地服务器的安装与使用	36
第 B 章	基于 biblatex 生成参考文献	39
B.1	bibtex 与 biber, natbib 与 biblatex 比较	39
B.1.1	概述	39
B.1.2	natbib 的优点	39
B.1.3	biblatex 的优缺点	39
B.2	biblatex 的在 $T_E X$ 中的使用样例	40
B.3	biblatex 在 bookdown 中的使用	40

表格

6.1	mtcars 数据的前 5 行.	26
6.2	遗传连接模型例子中观测到的频数 y_i 和频率 $p(y_i \pi)$, $i = 1, \dots, 4$, 197 个动物.	26

插图

2.1	R Bookdown 编译界面.	12
5.1	iris 数据集 Petal.Length} ~ Species 的箱线图.	21
5.2	iris 数据集 Petal.Length} ~ Species 的散点图. 右侧的图像中 散点类型通过 Species 因子的水平给出，见图例. 直线为数据集拟 合线性模型的结果.	22
5.3	iris 数据集 Petal.Length} ~ Species 的散点图. 下方的图像中 散点类型通过 Species 因子的水平给出，见图例. 直线为数据集拟 合线性模型的结果.	23
5.4	R logo	24

前言

今年接了 5 本与贝叶斯近似计算包 INLA 相关的翻译书，将由高等教育出版社出版。在准备翻译的时候，我静下来思考了一下二个问题。一是互联网时代在兼顾图书质量的同时怎么充分考虑读者阅读体验？二是什么是当下最为成熟的图书写作工具？特别是与数据科学密切相关的统计类图书的写作与出版。书稿模板的选择成为首先要考虑的事。

在书稿模板的选择与测试过程中遇到了很多的坑，幸运的是逐个踩过来了，但从 $T_E X$ 到 Rnw (Sweave+R), 再到 Rmd (Knitr + R), 最后到 Bookdown, 共经历了 4 个模板。快速、高效、高质量是写书人追求的目标。目前来看 Bookdown 是最好的选择，因为它满足我模板选择的快速编辑、高效生成、高质量输出的要求。

这本小册子可视为一个写中文书稿的 Bokdown 模板，也是中文 Bookdown 写作的一本说明书，其中汇总了书稿中几大核心要素的写作技巧。我主要参考了三个模板 Bookdown 模板和三本电子书，罗列如下，在此一并对谢益辉、李东风等表示感谢。

1. 谢益辉, [bookdown 中文范例](#)
2. 谢益辉, [A bookdown example for Chapman & Hall books](#)
3. [bookdown-chapterbib](#)
4. 谢益辉, bookdown: [Authoring Books and Technical Documents with R Markdown](#), 2021-03-15.
5. 李东风, R 语言教程, 第 23 章: 用 bookdown 制作图书, 2020-12-28.
6. Yihui Xie, J. J. Allaire, Garrett Grolmund, [R Markdown: The Definitive Guide](#), 2020-12-14.

第 1 章

引言

这是第1章的内容，回顾中国图书发展的历史及最新趋势. (R Core Team, 2021; Xie, 2015),

1.1 中国图书出版的变化

中国的图书出版经历一些折腾，方正系统一度非常流行，但现在看来是非常失败的，主要是它在解决公式排版的同时没有解决便利性，其本质上想实现在其自创的中文系统中将 $T_E X$ 命令做一个映射以实现数学公式排版，同时完成格式的定制。

中国学术界也经历了一些折腾，如中科院张林波研究员等开发的 CCT 系统和华东师范大学肖刚与陈志杰等老师开发的天元系统，它们是 $T_E X$ 系统汉化版，较好地解决了汉字生成与调用，但因没有考虑普适性或可拓展性而像方正系统一样随着 CJK 的逐渐成熟而逐渐被抛弃，均退出了历史舞台。而同期中科院吴凌云博士等在普及 $T_E X$ 的同时开发的 $T_E X$ 中文套餐 $CT_E X$ 相当成功，主要是针对汉字排版的 `ctex` 宏包，并对三个主流的文档类 `book`, `article`, `report` 进行了定制，推出了相应的中文文档类 `ctexbook`, `ctexart`, `ctexrep`, 由此避免了传统的基于 CJK 宏包需要的大幅定制，同时保证了与原有 $T_E X$ 系统的兼容性。这样我们始终可以使用跨平台的 TeXLive 进行排版或各类模板的开发，例如各个出版社的图书模板、各个期刊的模板、各高校的硕士和博士毕业论文模板等。

$T_E X$ 的出现¹，而且始终屹立不倒的原因是什么？第一个原因是它解决了一个 Word 之类文字编辑系统的痛点，即从所见所得 (WYSIWYG, What you see is what you get) 到所想所得 (WYTIWYG)，即通过一些 $T_E X$ 命令集构成一个完整的编程语言，由它完成一个封闭的体系，具有类似 C 语言一样非常强大的开发功能，由此形成了后来的 `latex`, `miktex`, `latex2.09`, `luatex`, `xetex` 等 $T_E X$ 编译引擎，它们在充分利用电脑系统资源的同时实现高质量输出需要的精度。

$T_E X$ 屹立不倒的另一个原因是浮动对象的处理，即包括公式、表格、图形、页码、章节、文献、定理等的标签化与引用，实现文档内部的自由跳转，结合 Acrobat

¹是伟大的 Knuth 创造了排版的奇迹

Reader 这样强大的 pdf 阅读器的支持,使得读者的阅读体验得到大幅改善,并为图书的电子化奠定了基础。

随着数据科学这一新兴学科的出现,开源的 R 和 Python (还有正在逐步流行的 Julia) 编程语言越来越强大。为了增加这类图书的可读性,需要将代码较完整地呈现在读者面前,并且要求代码的即时可复现能力,即数据的变化,其分析的结果(包括图形和表格)也随之发生变化。这就是现在逐步流行的文学化编程(literate programming),它实现上最早也是 $T_E X$ 的鼻祖 Knuth 提出的,后来被谢益辉得到重视并广泛推广,并通过 Rstudio 传递给 R 的用户。我们姑且把这种将写作与数据分析相结合的统计分析称为文学化统计编程(literate statistical programming),在为数据科学爱好者带来便利的同时也通过 Bookdown 为图书的写作和电子化带来了极大的好处,越来越多的网页版电子书出现在 (<https://bookdown.org/>) (<https://github.com/>)。

现在写书选择什么类型的模板,下面我们来作进一步的探索与比较。

1.2 统计类图书的核心要素

统计类图书的排版除普通图书的页面及文字风格等静态元素外,核心要素体现在浮动的对象上,使得图书的阅读体验更好地发挥出来,即在不同页面之间快速切换、跟踪、搜索,必要的 R 和 Python 代码以语法高亮方式显示。

1. **章节标题**是浮动的,最主要用于书签的生成;
2. **公式**是浮动的,这是数学、统计等理科书的特点,公式引用必不可少;
3. **图形**是浮动的,统计图形作为可视工具,在说明数据或展示分析结果时经常会引用相应的图形;
4. **表格**是浮动的,通常是原始数据或统计分析的结果以表格形式展示出来,它们可能被多次在不同的章节中引用;
5. **定理**是浮动的,这里定理是指与之相关的一大类,包括常用的定理、引理、推论、命题、例子等,它们在文中也会被反复引用;
6. **文本**可以设置浮动标签后被引用,最为常见的是图形与表格的题图(caption)通过文本方式来引用;
7. **文献**是浮动的,这在是谈及前人的已有工作、成果比较或进行综述时经常要引用大量已经发表的论文、图书、会议报告等。

$T_E X$ 有一套成熟的浮动对象的排版方式,通过给浮动对象打标签(label),然后引用(ref),Bookdown 思路一样,但比 $T_E X$ 的处理稍复杂些(可能因不习惯引起)。我们在后面章节中分别举例说明。

1.3 统计分析类图书模板的选择

统计分析类图书既有理论或原理的讲解，又会有一些案例分析，包括这些案例分析实现的代码。我们可以考虑的模板主要有三种类型。

1.3.1 基于纯 $T_{\text{E}}X$ 模板

全世界 90% 的书是由 $T_{\text{E}}X$ 排版的，包括硕士和博士毕业论文模板，这要感谢鼻祖 Knuth！开源成就了 $T_{\text{E}}X$ ！

如果仅仅是统计理论方面的书集，这显然是最好的选择，因为高质量公式的排版离不开 $T_{\text{E}}X$ 。基于 $T_{\text{E}}X$ 的排版存在三个明显的缺陷或不足：

1. 大量的 $T_{\text{E}}X$ 命令需要记忆；
2. 对于代码的排版非常不便，特别是 R 或 Python 代码执行后的输出，尤其是图形与表格；
3. 代码以 listing 等包来呈现，无法实时呈现代码运行的结果，不符合文学化编程的要求。

1.3.2 Rmarkdown 与 $T_{\text{E}}X$ 的结合

数据科学时代更注重文学化统计编程，**代码伴随**是这类图书的特点，自 Springer 出版 R 系列统计图书后，这种风格成为新趋势，大大方便了数据科学爱好者“便学习便练习”的学习方式。

针对代码伴随，早期对这类图书有二个解决方案：

1. Sweave/knitr + R

本质上它是在 $T_{\text{E}}X$ 嵌入 R 代码块，并由 R 在后台运行后将结果也嵌入到 $T_{\text{E}}X$ 中，再由 $T_{\text{E}}X$ 的编译引擎生成 pdf。这个方案的基本沿用 $T_{\text{E}}X$ 的方式，它仅解决了上面提到的第二个问题。在数据科学时代，报告快速生成成为新的要求，效率优先！随着 knitr 的出现 Sweave 退出舞台。

2. Rmarkdown + Mathjax/ $T_{\text{E}}X$

Markdown 作为一种轻量级的标记语言成为网页作为文字主要载体的互联网时代首先的写作工作，但它显然不适合数学与统计类论文或图书的撰写，但 knitr 和 pandoc 的出现使不同风格的内容整合与转换成为可能，而不同风格的内容各有善长的工具实现，作为统计类专业论文或图书类文档主要的内容有：

- a. 文字，由 markdown 完成
- b. 公式，由 $T_{\text{E}}X$ 完成

c. 代码, 由 R (或 Python) 完成

要说明的是, 在网页端, $T_E X$ 的实现可由 Mathjax 来完成或渲染 (转化或生成标准的公式), 见第4章说明。

随着 Rstudio 的越来越成熟与强大 (得益于许多优秀包的出现, 如 knitr, kableExtra), Rstudio 不仅是一个很好的代码编辑器 (Editor), 也是一个非常好的集成开发环境 (IDE), 同时正在成为一个非常优秀的论文、幻灯片及图书等撰写与出版系统 (PUB)。后者的基本流程是

1. 由 rmd 文件通过 knitr 完成初步集成
2. 由 pandoc 完成由 rmd 向 md 的转化与融合
3. 由 pandoc 完成由 md 转化为 $T_E X$, 并由 LaTeX 编译生成 pdf (形式多样!)
4. 或由 pandoc 由 md 转化为 html, 其中的数学公式由 Mathjax 完成渲染.

1.3.3 Rmarkdown 向 Bookdown 过渡

在科技高度发达的互联系时代, 读者使用的媒介基本有三类: 较为专业的电脑, 较为轻便的平板 (电脑) 和全功能的智能手机。前者以 pdf 类图书为主呈现给读者, 同时可以完成标注等工作; 后者以文字型的电子图书为主, 消磨时间为主; 而平板的使用者逐渐成为电子类图书的新势力, 包括 pdf 和 epub 之类的电子书。Bookdown 注重不同类型读者的媒体使用的差异, 并很好地实现统一编写与差异化输出。目前 Bookdown 可以生成三类图书:

1. gitbook, 可自由出版在 git pages 上
2. epub, 发表到大量的电子图书平台上
3. pdf, 正规的图书出版公司以电子或纸质形式出版

参考文献

- R Core Team, 2021. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.
- Xie, Y., 2015. Dynamic Documents with R and knitr. 2nd. Boca Raton, Florida: Chapman. <http://yihui.name/knitr/>.

第 2 章

bookdown 速览

这是第2章的内容，概要性地讲解基于 *bookdown* 拓展包进行图书排版整体思路与实现方式. (R Core Team, 2021; Xie, 2015)

2.1 关于 *bookdown*

bookdown 扩展包 (<https://github.com/rstudio/bookdown>) 是继 *knitr* 和 *rmarkdown* 扩展包之后, 另一个增强 *markdown* 格式的扩展, 使得 *Rmd* 格式可以支持公式、定理、图表、文献自动编号和引用等适用于编写书籍的功能。在 *bookdown* 的管理下一本书的内容可以按章节分解成多个 *Rmd* 文件, 其中可以包含可执行的 *R* 代码, *R* 代码生成的统计汇总结果、表格、图形可以自动插入到生成的内容中, 表格和图形可以是浮动排版的。书的输出格式包括支持 *gitbook* 格式的网页图书, 也可以经 *L^AT_EX* 编译器转换的 *PDF* 图书, 还可以生成 *ePub* 等格式的电子书。建议使用 *RStudio* 集成环境来编辑、管理和生成这样的图书, 可通过其内建的一键式编译整本书的插件 (*build*) 实现。

2.2 快速排版的思路

1. 由 *rmarkdown* 完成整个书稿的写作;
2. 由 *_output.yml* 完成不同形式呈现的书稿的设计, 其中 *bookdown::gitbook* 负责 *html* 形式的 *gitbook*, *bookdown::pdf_book* 负责 *pdf* 形式的电子书 (由 *T_EX* 支持); *bookdown::epub_book* 负责 *epub* 电子书. 部分针对书稿简单设置可放在 *index.Rmd* 文件的 *yml* 头部 (具体放在前面两组三个短线---之间);
3. 书稿按章节进行拆分, 借助 *js* 支持的 *html* 快速生成书稿的初稿, 最后再进行整合, 根据需要通过 *Build* 插件完成 *gitbook*, *pdf_book*, *epub* 的构建;
4. 借助 *mathjax* 处理数学公式的渲染;

尽快可通过联网由 cdn 上的 `mathjax.js` 进行渲染, 但速度随因公式的增加, 渲染变得很慢, 甚至出错。`mathjax` 的本地化是提速的主要解决方案. 详见第 A 章介绍;

5. 重点做好章节、数学公式、表格、图形、定理、文献等浮动对象的处理, 在编写过程中及时做好标签设定与引用, 见 2.6 节的汇总表格及后续各章的介绍与示例.

2.3 书的基本设置

一本用 bookdown 管理的书, 一般放置在某个子目录下, 并作为一个 RStudio 项目 (project) 用 RStudio 管理。该目录中的所有的文本文件都要使用 UTF-8 编码。

2.3.1 index.Rmd 文件

一本 bookdown 书, 一般都需要有一个 `index.Rmd` 文件, 这是最后生成的网站的主页的原始文件. 这个文件的开始是 YAML 元数据部分, 进行全书的有关设置, 包括标题、作者、日期及影响全书的一些选项等, 放在三个减号组成的两行之间。然后写一些这本书的说明, 如书的前言部分。`index.Rmd` 中 YAML 元数据部分的一个例子如下:

```
title: "bookdown书稿模板"
author: "汤银才"
date: "2021-07-21"
documentclass: book
bibliography: [myrefs.bib]
biblio-style: apa
link-citations: yes
site: bookdown::bookdown_site
description: "bookdown写书体验非常好."
```

注意:

1. `site` 选项很重要, 一定要有, `site: bookdown::bookdown_site` 使得 RStudio 能辨认这是一个 bookdown 图书项目, 从而为其生成一键编译的 `build` 快捷方式;
2. 在 bookdown 项目中与 `index.Rmd` 同级的所有 `.Rmd` 文件都自动作为书的一章, 其好处是作者可以任意地增删章节, 编译整本书时将按照文件

名的字典序依次进行。实际上,也可以在 `_output.yml` 文件中设置一项 `rmd_files`, 列出所有需要作为一章的文件, 并以列出次序编译;

3. 在 `index.Rmd` 的元数据中也可以指定一些 *LaTeX* 的选项, 例如

```
fontsize: 12pt
indent: true
linestretch: 2.0
link-citations: yes
colorlinks: yes
lot: true
lof: true
geometry:
- a4paper
- tmargin=2.5cm
- bmargin=2.5cm
- lmargin=3.5cm
- rmargin=2.5cm
```

2.3.2 `_bookdown.yml` 文件

一个 bookdown 图书项目除了 `index.Rmd` 文件之外, 还有一些设置文件从 `index.Rmd` 文件的元数据部分抽离出来。一个是 `_bookdown.yml` 文件, 它存放与整本书的处理有关的 YAML 元数据。例如

```
book_filename: "bookdown-template"
new_session: yes
language:
  label:
    fig: "图 "
    tab: "表 "
    thm: '定理'
    def: '定义'
    exm: '例'
    proof: '证明: '
  ui:
    chapter_name: ["第 ", " 章"]
```

其中 `new_session: true` 设置很重要, 这使得每一个 `Rmd` 文件中的 R 程序都在一个单独的 R 会话中独立地运行, 避免了不同 `Rmd` 文件之间同名变量和同名

标签的互相干扰。book_filename 是最终生成的 PDF 图书或者 ePub 电子书的主文件名。language 下可以定制一些与章节名、定理名等有关的名词。

2.3.3 _output.yml 文件

另一个设置文件是 _output.yml，用于图书输出格式的设置¹，本小册子的 _output.yml 文件内容如下

```
bookdown::gitbook:
  css: css/style.css
  split_by: chapter
  includes:
    in_header: _header.html
  config:
    toc:
      collapse: subsection
      scroll_highlight: yes
      before: |
        <li><a href=".">bookdown排版模板</a></li>
      after: |
        <li><a href="https://bookdown.org" target="blank">本书由 bookdown 强力驱动</a></li>
    download: [pdf, epub]
    edit: https://github.com/yihui/bookdown-chinese/edit/master/%s
    sharing:
      github: yes
      facebook: no
  pandoc_args: [ "--cs1", "apa-6th-edition-no-ampersand.csl" ]
bookdown::pdf_book:
  includes:
    in_header: latex/preamble.tex
    before_body: latex/before_body.tex
    after_body: latex/after_body.tex
  keep_tex: yes
  dev: "cairo_pdf"
  latex_engine: xelatex
  citation_package: biblatex
  template: latex/template.tex
```

¹这部分内容也可以包含在 index.Rmd 的元数据部分

```

toc_depth: 3
toc_unnumbered: no
toc_appendix: yes
quote_footer: ["\\begin{flushright}", "\\end{flushright}"]
pandoc_args: [ "--top-level-division=chapter" ]
bookdown::epub_book:
  stylesheet: css/style.css
  pandoc_args: [ "--csl", "apa-6th-edition-no-ampersand.csl" ]

```

它分别对 `gitbook`、`pdf_book` 和 `epub_book` 三种输出格式设置了一些输出选项。其中一些选项是通过文件形式给出设置的，我们再补充说明一下。

1. `style.css` 是自定义的 CSS 显示格式，在 `gitbook` 和 `epub_book` 中使用；
2. `_header.html` 是插入了一部分个性化的 HTML 代码，其内容将出现在每个生成的 HTML 文件的 `head` 部分。我们在此文件中给出了使用本地的 Mathjax 实现数学公式离线显示的设置，内容为

```

<script type="text/x-mathjax-config">
MathJax.Hub.Config({
  jax: ["input/TeX","output/SVG"],
  extensions: ["tex2jax.js","MathMenu.js","MathZoom.js"],
  TeX: {
    Macros: {
      bm: ["{\\boldsymbol #1}",1],
    },
    extensions: ["AMSMath.js","AMSsymbols.js","noErrors.js","noUndefined.js"]
  }
});

</script>
<script type="text/javascript"
  src="http://127.0.0.1/MathJax/MathJax.js">
</script>

```

其中 `http://127.0.0.1/MathJax/` 是本地服务器上 Mathjax 的位置。有关 Mathjax 的本地安装与启动可参考第 A 章的介绍；

3. `apa-6th-edition-no-ampersand.csl` 是 `gitbook` 和 `epub_book` 处理文献使用的风格文件；

4. `preamble.tex` 是处理 (编译)bookdown 文件经 `pandoc` 转化生成的 `tex` 文件时导言区需要额外的宏包和设置;

5. `before_body.tex` 是 `tex` 书稿类正文前面的设置, 最基本的是

```
\frontmatter
```

6. `after_body.tex` 是 `tex` 书稿类正文之后的设置, 最基本的是

```
\backmatter
```

7. ``template.tex`` 是针对 ``bookdown`` 编译经 ``pandoc`` 转化生成的 ``tex`` 文件时的模板, 由它生成供

其他选项说明:

1. `split_by: chapter`: 按章分割书稿;
2. `collapse: subsection`: 目录中隐藏子节 (仅显示二级标题);
3. `scroll_highlight: yes`: 目录滚动时高亮显示, 便于定位;
4. `keep_tex: yes`: 保留中间生成的 `tex` 源文件, 便于查错;
5. `dev: "cairo_pdf"`: 使用 `cairo_pdf()` 生成 `LATEX` 编译需要的图片文件;
6. `latex_engine: xelatex`: `TeX` 文件的排版引擎为 `XeLATEX`, 针对 UTF-8 编码;
7. `citation_package: biblatex`: 文献引用库指定为 `biblatex`, 另一个为 `natbib`;
8. `toc_depth: 3`: 目录提取至三级标题;
9. `toc_unnumbered: no`: 指定目录编号;
10. `toc_appendix: yes`: 附录添加到目录中.

2.4 章节结构

如前所述, 除了 `index.Rmd` 文件, 项目中每个 `.Rmd` 文件都作为一章, 其第一行是以一个 `#` 号和空格开头的一级标题。

每一章可以有若干节与子节, 分别用 `markdown` 的二级标题 (二个 `#` 开始) 和三级标题 (三个 `#` 开始) 编写。`bookdown` 的章、节、子节标题单独成一行, 其后可以添加标签, 章节的标签是标题后加空格, 然后是大括号内以 `#` 号开头的标签, 如

```
# 引言 {#intro}
```

```
## 关于bookdown {#bookdown}
```

bookdown 中有二个特殊的标题:

1. 部分

内容相近的章节可以作为一个“部分”。为此, 在一个部分的第一个章节文件的章标题前面增加一行, 以 # (PART) 开头, 以 {-} 结尾, 例如

```
# (PART) bookdown中的浮动对象 {-}
```

2. 附录

一本书的最后可以有附录, 附录的章节将显示为 A.1, B.1 这样的格式。为此, 在附录章节的第一个文件开头加如下的第一行标题行:

```
# (APPENDIX) 附录 {-}
```

```
# biblatex介绍 {#biber}
```

2.5 书的编译

在 `index.Rmd` 或者 `_bookdown.yml` 中设置 `site: bookdown::bookdown_site` 后, RStudio 就能识别这个项目是一个 bookdown 项目, 这时 RStudio 会有一个 Build 按钮, 其中有 Build book 快捷图标, 从下拉菜单中选择一个输出格式 (包括 `gitbook`、`pdf_book`、`epub_book`), 就可以编译整本书 (见图2.1)。

经 build 编译生成的图书默认保存在 `_book` 子目录中²。

1. 对 `gitbook` 格式 (即 HTML 网页格式), 编译完成后会弹出一个预览窗口, 点击 “Show in new window” 按钮可以将内容在操作系统默认的网络浏览器中打开。我们也可以用其他浏览器 (建议使用 Google chrome 浏览器) 打开 `_book` 子目录中的 `index.html` 文件来查看 `gitbook` 格式的图书。
2. 对于 `pdf_book` 格式, 如果成功编译³, 也会弹出一个 PDF 预览窗口。可以在 `_book` 子目录中找到这个 PDF 文件。
3. 对于 `epub_book` 格式, 如果成功编译, 会在操作系统默认的 ePub 软件 (如苹果电脑的 `book`) 中打开, 并在 `_book` 子目录中找到这个 ePub 文件。

²可以在 `_bookdown.yml` 中设置 `output_dir` 项改变图书保存的子目录。

³需要 \LaTeX 支持, 建议安排 TeXLive, 也可以仅安装谢益辉为 `rmarkdown` 开发的 `tinytex`

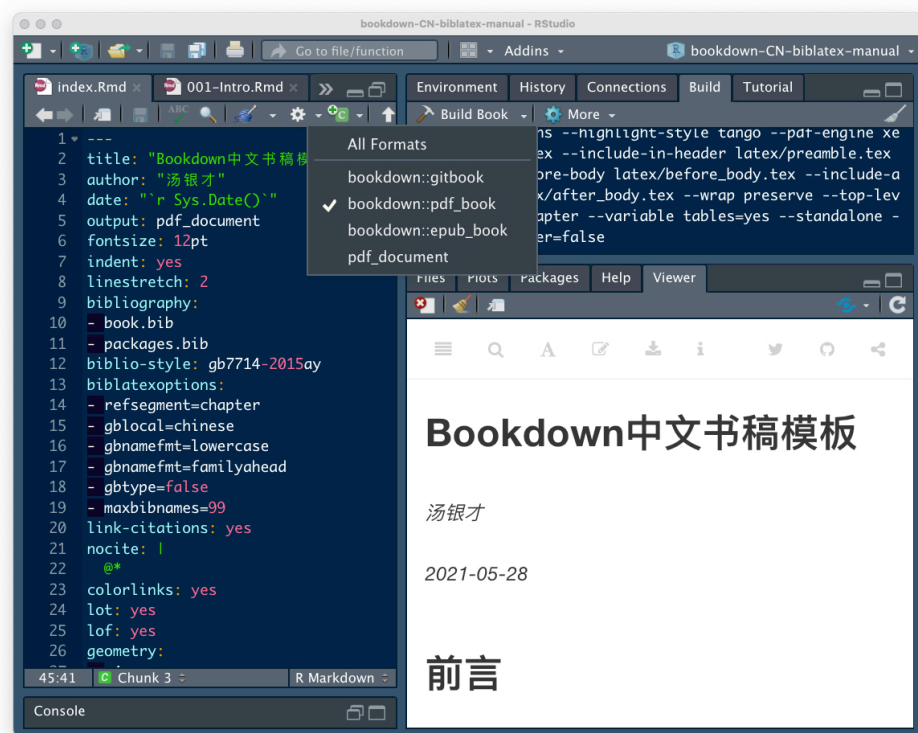


图 2.1 R Bookdown 编译界面.

2.6 浮动对象标签与引用汇总

浮动对象	标签设置	引用格式
标题	(# label)	\@ref(label)
公式	(\#eq:label)	\@ref(eq:label)
图形	label="label"	\@ref(fig:label)
表格	label="label"	\@ref(tab:label)
定理	label="label"	\@ref(prefix:label)
文本	(ref:label)	`(ref:label)`
文献	biblabeled	@biblabeled

注:

1. 定理泛指定理类, 包括定理 (thm)、引理 (lem)、推论 (cor)、命题 (prp)、设想 (cnj)、定义 (def)、例子 (exm)、习题 (exr) 等, 其中括号中是引用时的前缀 (prefix);
2. 文本标签在单独一行中设定, 可用在表格与图形的 caption 中引用, 即在 fig.caption, tab.caption 选项的设置中引用;
3. 定理类环境标签前缀的汉化可在 `_bookdown.yml` 中通过 `language` 进行⁴, 例如

Xie, Y., 2015. Dynamic Documents with R and knitr. 2nd. Boca Raton, Florida: Chapman. <http://yihui.name/knitr/>.

第 3 章

Bookdown 中的章节标题

我们在第3章讲述章节标题的设置、标签与引用. (Xie, 2015, 2016)

3.1 章节标题

章节标题用遵从 `markdown` 的规则, 用 `#` 设置,

- 一级标题用一个 `#`, 在 `bookdown` 中表示章, 相当于 $T_{\text{E}}X$ 中的 `\chapter{}`
- 二级标题用二个 `#`, 在 `bookdown` 中表示节, 相当于 $T_{\text{E}}X$ 中的 `\section{}`
- 三级标题用三个 `#`, 在 `bookdown` 中表示子节, 相当于 $T_{\text{E}}X$ 中的 `\subsection{}`

还可以有更深的标题.

3.2 章节标题标签的设定与引用

章节标题标签可在标题后用 `{#label}` 来设定, 引用方式为 `\@ref(label)`. 例如

第`\@ref(sections)`章`\@ref(sec3-2)`节讨论标题标签的设定与引用.

显示为:

第3章3.2节讨论标题标签的设定与引用.

参考文献

Xie, Y., 2015. *Dynamic Documents with R and knitr*. 2nd. Boca Raton, Florida: Chapman. <http://yihui.name/knitr/>.

Xie, Y., 2016. Bookdown: Authoring Books and Technical Documents with R Mark-down. Boca Raton, Florida: Chapman. <https://bookdown.org/yihui/bookdown>.

第 4 章

Bookdown 中的公式与定理

这是第4章的内容, 讲述浮动对象定理与公式的标签与引用. (Xie, 2015, 2016)

4.1 公式标签的设置

Rmarkdown 中公式除了无标号的公式 (用一对 `$$` 实现), 可以使用 LaTeX 中的 `equation` 环境, 尽管无法实现类似的 WYSIWYG, 但可设置标签. 标签格式为 `(\#eq:label)`, 其中 `eq` 是关键字, 例如

```
\begin{equation}
  f\left(k\right) = \binom{n}{k} p^k\left(1-p\right)^{n-k}
  \label{eq:binom}
\end{equation}
```

显示为

$$f(k) = \binom{n}{k} p^k (1-p)^{n-k} \quad (4.1)$$

对于多行公式可以采用 `align` 环境, 可对多个公式同时进行设置标签, 不需要标签则用 `\notag`, 例如

```
\begin{align}
g(X_{\{n\}}) &= g(\theta) + g'(\tilde{\theta})(X_{\{n\}} - \theta) \notag \\
\sqrt{n}[g(X_{\{n\}}) - g(\theta)] &= g'(\tilde{\theta})\sqrt{n}[X_{\{n\}} - \theta] \\
&\label{eq:align}
\end{align}
```

显示为

$$\begin{aligned} g(X_n) &= g(\theta) + g'(\tilde{\theta})(X_n - \theta) \\ \sqrt{n}[g(X_n) - g(\theta)] &= g'(\tilde{\theta})\sqrt{n}[X_n - \theta] \end{aligned} \quad (4.2)$$

4.2 定理标签的设置

这里我们先叙述几个定义和定理，并给出几个例子.

引理 4.1 A group having an infinite number of elements.

定理 4.2 (无限群) A group having an infinite number of elements.

证明 The proof comes here. □

定义 4.1 A group having an infinite number of elements.

例 4.1 The set $(\mathbb{Z}, +)$ is an infinite group.

4.3 定理与公式的引用

例4.1, 定义4.1 定理4.2为定理类引用.

公式的引用采用 `\@ref(eq:label)`, 例如上面的二个公式可引用为: 公式(4.1) 和公式 (4.2).

4.4 数学公式的扩展

有些公式无法用 $T_E X$ 中包的命令来实现, 例如粗体数学符号, 尽管在 $T_E X$ 中有个 `bm` 包在数学环境下通过 `\bm{\alpha}` 来实现 `\boldsymbol{\alpha}` 的功能, 但在 `html` 下需要给 `mathjax` 做个 $T_E X$ 宏 (macro)¹:

```
TeX: {
  extensions: ["AMSmath.js", "AMSsymbols.js", "noErrors.js", "noUndefined.js"]
  Macros: {
    bm: ["{\boldsymbol #1}", 1],
  },
}
```

此时由 `$\bm{\alpha}$` 出来的效果为 α .

有关数据公式的标签与应用可参考[mathjax 官方文档](#), `Mathjax` 的本地化安装参考第B章介绍.

¹配置在 `MathJax.Hub.Config` 下进行, 具体参见 `Mathjax` 技术文档说明

参考文献

Xie, Y., 2015. Dynamic Documents with R and knitr. 2nd. Boca Raton, Florida: Chapman. <http://yihui.name/knitr/>.

Xie, Y., 2016. Bookdown: Authoring Books and Technical Documents with R Markdown. Boca Raton, Florida: Chapman. <https://bookdown.org/yihui/bookdown>.

第 5 章

Bookdown 中的图形

5.1 由 R 生成单个图形示例

这是第5章的内容，讲述浮动对象图形的标签与引用. (Xie, 2015, 2016)

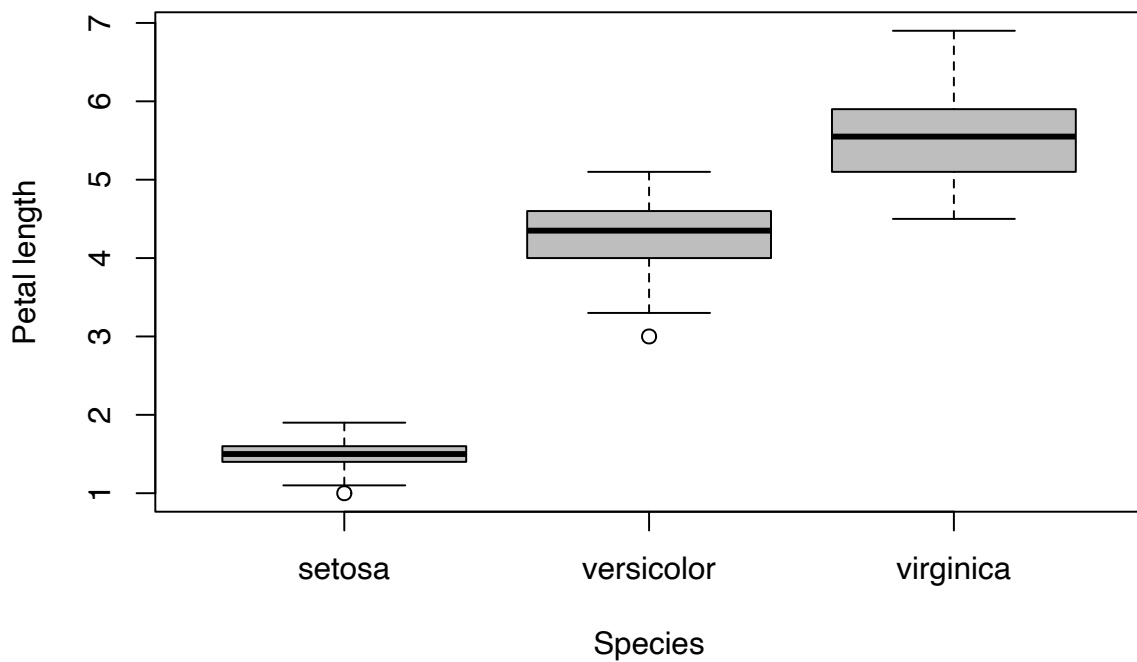


图 5.1 iris 数据集 $\text{Petal.Length} \sim \text{Species}$ 的箱线图.

5.2 由 R 生成两个图形并置示例

在 R 的代码块选项中设置 `out.width='50%', fig.show='hold'` 就可获得二个图形的并置.

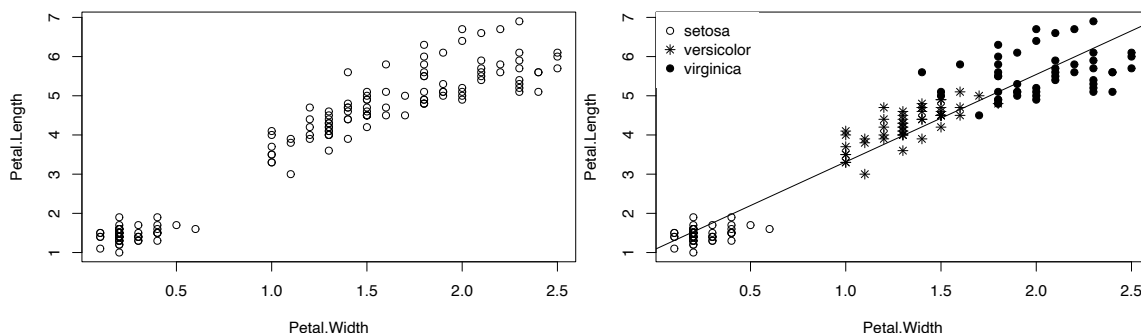


图 5.2 iris 数据集 $\text{Petal.Length} \sim \text{Species}$ 的散点图. 右侧的图像中散点类型通过 Species 因子的水平给出, 见图例. 直线为数据集拟合线性模型的结果.

5.3 由 R 生成两个图形堆叠示例

在 R 的代码块选项中设置 `out.width='90%', fig.show='hold'` 就可获得二个图形的并置.

5.4 静态图形示例

在 Bookdown 中插入本地图形可使用命令 (示例为 Rlogo)

```
knitr::include_graphics("figures/Rlogo.png")
```

5.5 图形引用

图形引用通过 R 代码块的标签引用, 并带前缀 `fig:`, 例如

图`\ref{fig:fig4-2}`和图`\ref{fig:fig4-3}`为两个图的并置与堆叠.

输出为:

图5.2和图5.3为两个图的并置与堆叠.

参考文献

Xie, Y., 2015. Dynamic Documents with R and knitr. 2nd. Boca Raton, Florida: Chapman. <http://yihui.name/knitr/>.

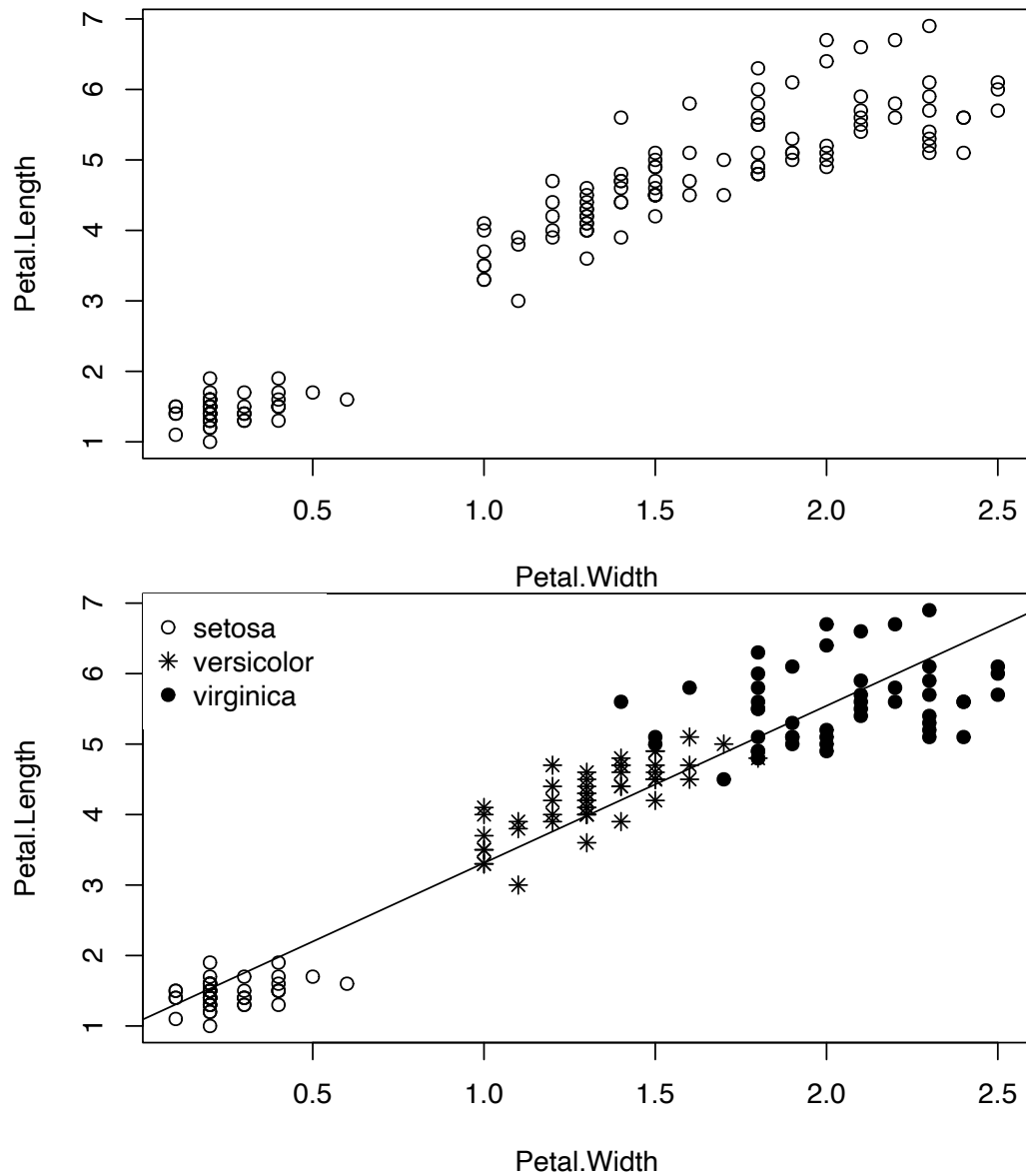


图 5.3 iris 数据集 $\text{Petal.Length} \sim \text{Species}$ 的散点图. 下方的图像中散点类型通过 Species 因子的水平给出, 见图例. 直线为数据集拟合线性模型的结果.



图 5.4 R logo

Xie, Y., 2016. Bookdown: Authoring Books and Technical Documents with R Markdown. Boca Raton, Florida: Chapman. <https://bookdown.org/yihui/bookdown>.

第 6 章

Bookdown 中的表格

这是第 6 章的内容, 讲述浮动对象图形的标签与引用. (Xie, 2015, 2016)

6.1 表格示例 1: 由数据生成表格

```
knitr::kable(  
  head(mtcars[, 1:5], 10), booktabs = TRUE,  
  caption = 'mtcars 数据的前 5 行.'  
)
```

6.2 表格示例 2: 由数据框构造表格

注意:

1. 表格中% 用\\%
2. 表格中 latex 命令用\\代替\

6.3 表格引用

表格引用由代码块的标签 (设为 label) 引用实现, 并带前缀 tab:, 由\@ref(tab:label) 实现. 例如

本章共出现二张表格, 即表\\ref{tab:tab3-1}和表\\ref{tab:tab3-2}.

输出为:

本章共出现二张表格, 即表6.1和表6.2.

注: 表格中的 caption(题图) 由文本引用生成.

表 6.1 mtcars 数据的前 5 行.

	mpg	cyl	disp	hp	drat
Mazda RX4	21.0	6	160.0	110	3.90
Mazda RX4 Wag	21.0	6	160.0	110	3.90
Datsun 710	22.8	4	108.0	93	3.85
Hornet 4 Drive	21.4	6	258.0	110	3.08
Hornet Sportabout	18.7	8	360.0	175	3.15
Valiant	18.1	6	225.0	105	2.76
Duster 360	14.3	8	360.0	245	3.21
Merc 240D	24.4	4	146.7	62	3.69
Merc 230	22.8	4	140.8	95	3.92
Merc 280	19.2	6	167.6	123	3.92

表 6.2 遗传连接模型例子中观测到的频数 y_i 和频率 $p(y_i|\pi)$, $i = 1, \dots, 4$, 197 个动物.

Category	1	2	3	4
Frequency y_i	125	18	20	34
Probability $p(y_i \mid \pi)$	$\frac{1}{2} + \frac{\pi}{4}$	$\frac{1}{4}(1 - \pi)$	$\frac{1}{4}(1 - \pi)$	$\frac{1}{4}$

参考文献

Xie, Y., 2015. Dynamic Documents with R and knitr. 2nd. Boca Raton, Florida: Chapman. <http://yihui.name/knitr/>.

Xie, Y., 2016. Bookdown: Authoring Books and Technical Documents with R Markdown. Boca Raton, Florida: Chapman. <https://bookdown.org/yihui/bookdown>.

第 7 章

Bookdown 中的文本参考

这是第7章的内容, 讲述文本的标签与引用. (R Core Team, 2021; Xie, 2015)

7.1 文本标签的设定与引用

通过 (ref:label) 设定被引文本, 其前后必须有空行与正文分开. 例如

被引用的文本: 我要被引用的**文本**.

输出为

被引用的文本: 我要被引用的**文本**.

7.2 文本引用场景

在 Bookdown 中文本引用主要用于图形与表格的题图 (caption) 上, 见第5章和第6章的例子.

参考文献

R Core Team, 2021. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.

Xie, Y., 2015. Dynamic Documents with R and knitr. 2nd. Boca Raton, Florida: Chapman. <http://yihui.name/knitr/>.

第 8 章

Bookdown 中的文献排版

这是第8章的内容，讲解文献库的建立、文献格式与排版方法. (R Core Team, 2021; Xie, 2015)

8.1 文献库建立样例

根据文献的类型，文献库的格式有 14 种类型，它们由 `title`, `author`, `journal`, `address` 等域构成. 最为常见的文献类型是论文 (`article`) 和图书 (`book`), 它们的格式如下

```
@article{CitekeyArticle,
  author   = "P. J. Cohen",
  title    = "The independence of the continuum hypothesis",
  journal  = "Proceedings of the National Academy of Sciences",
  year     = 1963,
  volume   = "50",
  number   = "6",
  pages    = "1143--1148",
}

@book{CitekeyBook,
  author    = "Leonard Susskind and George Hrabovsky",
  title     = "Classical mechanics: the theoretical minimum",
  publisher = "Penguin Random House",
  address   = "New York, NY",
  year      = 2014
}
```

其余类型参见网页:[The 14 BibTeX entry types](#)。文献库通过 BiBTeX 或 BiBer 的运行生成文献目录 (bibliography).

8.2 文献风格

不同的期刊、书集对文献目录中参考文献的呈现方式有不同的要求，BiBTeX 是通过风格 (style) 文件来控制 (配合宏包 natbib 使用)，BiBer 是通过选项来控制 (配合 biblatex 使用)。

文献呈现的风格分为作者-日期格式 (author-date style) 和数字格式 (numeric)。作者-日期格式共有 141 个式样，其中最为常用的式样有 2 个，即 alpha 和 apalike。数据格式共 88 个式样，其中有 8 类是标准式样，也是最常用的式样，即 abbrev, acm, ieeeetr, plain, siam, 和 unsrt。这些式样的具体形式与介绍见

中文的文献排版根据国标 GB/T7714-2015 的规范，对文献的类型要求提供文献的标识代码，共有 18 个，例如图书用 M 标识，期刊论文用 J 标识。

8.3 中文文献风格的设置

基于 BiBTeX，中文文献通过宏包 gbt7714 实现¹，兼容宏包 natbib，在 L^AT_EX 中使用方法如下：

1. 在导言区调用宏包 gbt7714；
2. 在正文中 \cite{} 等引用文献；
3. 使用 \bibliographystyle{} 选择参考文献表的样式；
4. 使用 \bibliography{} 命令生成参考文献表；
5. 编译生成带文献的 pdf 文件，基于 xelatex 引擎的编译方式如下 (foo.tex 为文件名)

```
xelatex foo.tex
bibtex foo
xelatex foo.tex
xelatex foo.tex
```

基于 BiBer，中文文献通过宏包 biblatex-gb7714-2015 实现²，兼容宏包 biblatex。本质上 biblatex-gb7714-2015 是个式样宏包，依附于宏包 biblatex，通过式样选项使用。在 L^AT_EX 中使用方法如下：

1. 在导言区调用宏包 biblatex，并设定文献样本和，例如；

¹详见 <https://github.com/87ouo/gbt7714-bibtex-style> 说明

²详见 <https://github.com/hushidong/biblatex-gb7714-2015>

- 使用顺序编码制：

```
\usepackage[backend=biber,style=gb7714-2015]{biblatex}
```

- 使用著者-出版年制³：

```
\usepackage[backend=biber,style=gb7714-2015ay]{biblatex}
```

2. 在`\begin{document}` 加载参考文献库，命令为`\addbibresource{}`
3. 在正文中 `\cite{}` 等引用文献；
4. 在需要出现文献目录的地方使用 `\printbibliography`，可通过 `title`, `heading`, `segment` 选项对输出进行控制；
5. 编译生成带文献的 pdf 文件，基于 `xelatex` 引擎的编译方式如下 (`foo.tex` 为文件名)

```
xelatex foo.tex
biber foo
xelatex foo.tex
xelatex foo.tex
```

8.4 文献库的建立工具

- Zotero
- JabRef

参考文献

R Core Team, 2021. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.

Xie, Y., 2015. Dynamic Documents with R and knitr. 2nd. Boca Raton, Florida: Chapman. <http://yihui.name/knitr/>.

³尽管后端 (backend) 可以使用 `bibtex`, 但不建议使用.

第 A 章

Mathjax 的离线安装与使用

附录A介绍了网页显示数学公式的插件 mathjax，本地化安装，使用方法等.
(R Core Team, 2021; Xie, 2015)

A.1 mathjax 说明

- [MathJax](#)是一款相当强悍的在网页显示数学公式的插件。基于 Mathjax, 通过 \LaTeX 的命令输出精美的数学公式. 加载 Mathjax 后¹, 就可通过一对美元符号 $(或左\右\)$ 输入行内公式, 通过一对双美元符号 $$(或左\[右\])$ 输入行间公式, 例如

$$$$

$$J\alpha(x) = \sum_{m=0}^{\infty} \frac{(-1)^m}{m! \Gamma(m + \alpha + 1)} \left(\frac{x}{2}\right)^{2m+\alpha}$$

显示出下面的数学公式

$$J\alpha(x) = \sum_{m=0}^{\infty} \frac{(-1)^m}{m! \Gamma(m + \alpha + 1)} \left(\frac{x}{2}\right)^{2m+\alpha}$$

- 可以使用 \LaTeX 自带的复杂的数学环境, 如排版多行公式的 `align`, `split` 和数学字体命令, 如`\mathscr`

```
\begin{align}
3x-1 &= \mathbb{A} \\
3x &= \mathbf{B} \\
x &= \mathscr{C}
\end{align}
```

输出为

$$3x - 1 = \mathbb{A} \tag{A.1}$$

¹需要远程或本地支持

$$3x = \mathbf{B} \quad (\text{A.2})$$

$$x = \mathcal{C} \quad (\text{A.3})$$

A.2 调用远程服务器上的 *mathjax*

一般情况下，只需要使用远程加载 *Mathjax* 的 *js* 库就行了，例如在需要渲染数学公式的网页上增加 *html* 命令

```
</script>
<script type="text/javascript" async
  src="https://cdn.mathjax.org/mathjax/latest/MathJax.js?config=TeX-MML-AM_CHTML">
</script>
```

A.3 *mathjax* 本地服务器的安装与使用

我们以 Macbook 的 *Apache* 服务器为例说明步骤²

1. 服务器的启动

在终端 (terminal) 下输入命令

```
sudo apachectl start
```

2. 检查服务是否启动成功

在浏览器中输入网址

```
http://127.0.0.1/
```

如果显示 *It Works!* 就表示服务器已经成功启动。请记住: 服务器上文件在本地的位置为

```
/Library/WebServer/Documents
```

3. 关闭服务器 (不用时)

在终端 (terminal) 下输入命令

```
sudo apachectl stop
```

²Windows 10 下的本地服务器的启动可参考 (<https://www.jianshu.com/p/d86c77942181>)

4. 将 Mathjax2.6 或 Mathjax2.7 下载并解压 (安装) 到/Library/WebServer/Documents 下³, 目录名为 Mathjax
5. 启动本地 Mathjax

在运行 Bookdown(或其他 rmarkdown 文件) 时, 须加载下面的 html 命令, 在 Bookdown 中放在文件 mathjax_header.html 中, 并由 _output.yml 加载进来.

```
</script>
<script type="text/javascript"
  src="http://127.0.0.1/MathJax/MathJax.js">
</script>
```

参考文献

- R Core Team, 2021. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.
- Xie, Y., 2015. Dynamic Documents with R and knitr. 2nd. Boca Raton, Florida: Chapman. <http://yihui.name/knitr/>.

³不要用最新的 3.1 版本

第 B 章

基于 *biblatex* 生成参考文献

附录B介绍了文献库扩展包 *biblatex* 的使用, 及其与 *natbib* 的比较. (R Core Team, 2021; Xie, 2015)

B.1 *bibtex* 与 *biber*, *natbib* 与 *biblatex* 比较

B.1.1 概述

- *bibtex* 与 *biber* 是处理参考文献信息二个外部程序 (backend), 它们起到将 \LaTeX 文档与 *bib* 文档交互的作用;
- *natbib* 与 *biblatex* 是二个处理参考文献 (bibliography) 和引用 (citation) 的 \TeX 宏包; *natbib* 仅通过 *bibtex* 起作用, 而 *biblatex* 可通过 *biber* 起作用;

B.1.2 *natbib* 的优点

- *natbib* 的优点:
 - 有大量与期刊/出版商对应的风格文件 (.sty);
- *natbib* 的缺点:
 - 修改风格文件较为困难;
 - 其设计的局限性: 主要为自然科学类期刊的 Author-Year 及数字引用方式设计.

B.1.3 *biblatex* 的优缺点

- *biblatex* 被认为是 \LaTeX 中处理参考文献 (bibliography) 的很有前途的宏包, 功能强大, 提供了很多可定制选项。
- *biblatex* 的优点
 - 提供 *natbib* 的 Author-year 和数字引用方式, 因此可视为 *natbib* 的扩展;

- 通过 *L^AT_EX* 的宏控制文献的风格, 方便修改;
- 提供许多人性化的引用风格 (例如 `author-title`);
- 提供人性化的文献数据库域名 (字段);
- 直接支持多文献库与文献分类;
- 提供大量标准与拓展的 *biblatex* 风格 (见使用手册);
- 文献可按主题分割为部分;

B.2 *biblatex* 的在 *T_EX* 中的使用样例

1. 使用

```
\usepackage[style=authoryear,backend=biber]{biblatex}
```

(代替 `\usepackage[authoryear]{natbib}`);

2. 加载一个或多个 bib 文献库文件:

```
\addbibresource{file1.bib}
```

```
\addbibresource{file2.bib}
```

3. 要出现文献的地方输入

```
\printbibliography
```

4. 引用: 使用 `\textcite` 代替 `\citet`; 使用 `\parencite` 代替 `\citep`

5. 编译方式

X_EL_AT_EX (或 L_AT_EX, pdfL_AT_EX)

biber (代替 bibtex)

X_EL_AT_EX (或 L_AT_EX, pdfL_AT_EX)

X_EL_AT_EX (或 L_AT_EX, pdfL_AT_EX)

B.3 *biblatex* 在 bookdown 中的使用

1. 在 `index.Rmd` 文件的 `yaml` 部分增加选项

```
biblatexoptions: [refsegment=chapter]
```

```
biblio-style: gb7714-2015ay
```

注意 `[]` 内可增加其他 `biblatex` 选项 (参考 `biblatex` 使用手册)

2. 在需要出现文献的地方 (如每一章后) 加

```
\printbibliography[segment=\therefsegment, heading=subbibliography, title={参考文献}]
```

3. 在 `_output.yml` 文件的 `bookdown::pdf_book:` 增加选项 (前面空二格)

```
citation_package: biblatex
```

注: 若要 `bibtex` 驱动文献的排版, 只需要在这一步改为 `citation_package: natbib`

4. 如上所述, LaTeX 要生成最终的 PDF 文档, 如果含有交叉引用 (图形、表格、公式、章节、文献等)、BibTeX/biber、术语表等等, 通常需要多次编译才行。而使用 `Latexmk` 则只需运行一次, 它会自动帮你做好其它所有事情。尽管在你已经安装的 LaTeX 发行版本已经包含了 `Latexmk`, 但我们需要运行它。使用 `XeLaTeX` 的编译格式为

```
latexmk -pvc -xelatex file.tex
```

其中选项 `pvc` 表示检查输入文件的更改并预览结果。在 `Rstudio` 中你只需要添加下面的代码块。

```
Sys.setenv(RSTUDIO_PDFLATEX = "latexmk")
```

参考文献

- R Core Team, 2021. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.
- Xie, Y., 2015. Dynamic Documents with R and knitr. 2nd. Boca Raton, Florida: Chapman. <http://yihui.name/knitr/>.

参考文献

- R Core Team, 2021. R: A Language and Environment for Statistical Computing. Vienna, Austria: R Foundation for Statistical Computing. <https://www.R-project.org/>.
- Xie, Y., 2015. Dynamic Documents with R and knitr. 2nd. Boca Raton, Florida: Chapman. <http://yihui.name/knitr/>.
- Xie, Y., 2016. Bookdown: Authoring Books and Technical Documents with R Markdown. Boca Raton, Florida: Chapman. <https://bookdown.org/yihui/bookdown>.

