

Open tools for writing open interactive textbooks (and more)

Matthew Crump

2018-03-14

Contents

Preface	5
1 Overview	7
1.1 Open online educational resources (OERs)	7
1.2 What is this tutorial?	7
1.3 Can I see an example of what you are talking about?	7
1.4 The toolbox (what do I need to download)	9
1.5 Quickstart	10
1.6 Description of Files in this Repository	11
2 R, R-studio, and RMarkdown	13
2.1 Quick start steps	13
2.2 R Markdown	13
2.3 R Markdown tips	14
3 Bookdown	15
3.1 R Packages and installing Bookdown	15
3.2 Using Bookdown	15
3.3 Compiling this tutorial using bookdown.	15
4 Github	17
4.1 Getting Started	17
4.2 Repository workflow	17
4.3 Creating a new repository on Github.com	18
4.4 Cloning a copy on your computer	18
4.5 Make changes to the repository	18
4.6 Update the online github repository	18
5 Hypothes.is	19
5.1 A case study example	19
5.2 hypothesisr	19
5.3 Hypothesis Explorer	19
5.4 Downloading annotations	20
5.5 Viewing the annotations Search and subset the database	20
6 Zotero	23
6.1 Getting Started	23
6.2 Populating Zotero with references	23
6.3 Generating a .bib file	24
6.4 Citing references in an RMarkdown file	24
7 Shiny	25
7.1 Sharing Shiny Apps	25

7.2	A minimal working example using Heroku	25
-----	--	----

Preface

This is a tutorial and set of working examples for creating web-based textbooks using a collection of open-source tools.

This web-book is itself a working example. All of the source code needed to compile this book yourself is included in the github repository for this book. So, you could download the repository, and by following the instructions laid out across the chapters, replace this text with your own, and then compile your book as a web-page, .pdf or epub.

Feel free to contribute to this tutorial by submitting pull requests to this repository.

Chapter 1

Overview

1.1 Open online educational resources (OERs)

OERs are educational resources that are public and freely available to students (or anyone) through the web. There are already numerous OER databases linking to numerous free textbooks across many domains of study (e.g., see the BC library OER site). These texts are often disseminated under a creative commons license, which allows anyone to mix, match, edit, update, and transform those texts, and then share them again with the public at large under the same license. Finding existing OERs can be a good starting point for developing your own content.

1.2 What is this tutorial?

This tutorial shows how a few open-source tools can be brought together for quickly developing and sharing content in the form of online books and webpages. The tools require some downloading of free programs (all cross-platform), but once in place, allow you to write and format text with minimal syntax, and then easily compile your source documents into multiple formats, including web-books, .pdf's, and e-pubs.

This tutorial also covers tools to enhance student interaction with the content, including the ability for students to select text in a web-book and then post comments about the text. Depending on the pedagogical goals of the instructor, posts could be to critically evaluate portions of the textbook, supplement sections with additional examples, and tag sections in need of revision or that were particularly helpful. All of these comments can be made publically accessible so that everyone viewing the textbook can see all of the comments as they are posted (inline, corresponding to the sections that were annotated). Furthermore, the database of annotations can easily be extracted and used for revising the text, or other purposes.

1.3 Can I see an example of what you are talking about?

Yes, there are many working examples already. As the preface mentions, this tutorial is an example. So the bells and whistles you see here constitute one example of an end product. Some additional examples include a Research Methods for Psychologist's textbook, and a Programming for Psychology textbook that I made using the tools in this tutorial.

If you want to see examples of the inline commenting features, simply follow the directions in the preface of the above two books. Inline commenting is made possible by the *hypothes.is* browser add-on tool. It is simple to install, requiring that you sign up with hypothesis, and then follow their directions to install

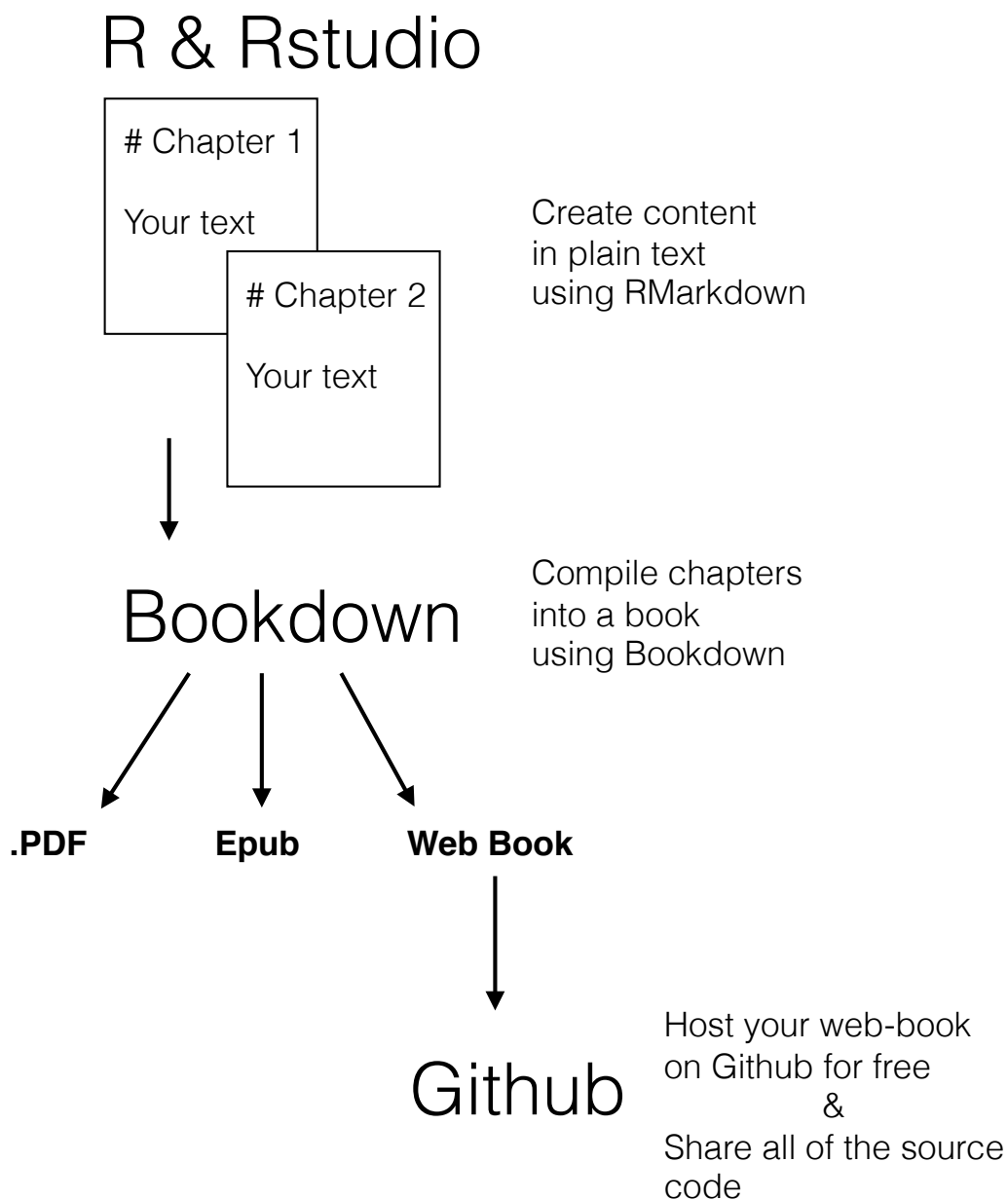


Figure 1.1: Basic steps

an add-on for chrome, or a bookmarklet for other browsers. When you turn on hypothesis viewing these web-books, you will be able to see any public comments made by other people.

1.4 The toolbox (what do I need to download)

Each of the chapters in this tutorial will discuss each of these tools in more depth, along with links to many helpful tool-specific resources already available online.

1.4.1 R

R is a free and open-source programming language primarily developed for statistical analysis. R has a large community of active open-source developers who are makin R better and more powerful everyday. R comes with existing packages for statistical analysis, but can be extended to accomplish many more tasks by downloading and installing additional packages. Download and install before you download and install R-studio.

1.4.2 R-studio

R-Studio is a free and open-source GUI for R. For those of you familiar with the Matlab environment, R-studio looks similar.

1.4.3 Markdown and R Markdown

Markdown is a simple mark-up language for creating .html files viewable in web-browsers. HTML (hyper-text markup language) is a powerful, but visually cluttered syntax for creating content to be displayed in web-pages. Markdown allows you to easily write text while avoiding writing html syntax. For example, This document is written in markdown.

RMarkdown is R's version of markdown. It's basically the same as markdown, but allows you to inject and run R code inside a markdown document. This is especially useful when you are writing documents that contain output from data-analysis you have conducted in R. This document is an .Rmd file (Rmarkdown). The paragraphs are plain text, and the header's (level I, II, III) are defined by #'s.

1.4.4 Bookdown

Bookdown is an R package for compiling Rmarkdown documents into various formats, including web-books, .pdf's, and epub's. This incredible package, along with the knitr package, does much of the heavy lifting, allowing writer's to focus on content generation, rather than the technical details of deploying that content to the various formats. Once the .Rmd files are written, deployment can is a single-click. Bookdown also provides different formats for styling the output, and styling is customizable.

1.4.5 Github

Github is a version control software development tool/website. Github provides a free option for hosting static web-pages (e.g., you could deploy your web-book to github), and allows you to store the source code for generating your book in an online repository. This allows others to easily copy your source code and make suggested edits that you can incorporate back into your content. This web-book is hosted using github pages, and the source code for this book is hosted in a github repository. Downloading the github repository

will give you all the files you need to compile this book, and by editing the text in this book, you can change it into your own.

1.4.6 Hypothes.is

Hypothes.is is a web-browser add-on for annotating the web via inline commenting. This allows anyone to select a snippet of text in a web-browser and then post a comment about the selected text. Annotations can be made publicly or privately. All public annotations can be viewed by anyone else running hypothesis on the same website. Using Hypothesis with your web-book allows you to engage students in interacting with the textbook by allowing them to contribute to content generation (by posting new content), or content revision (by tagging sections in need of revision).

1.4.7 hypothesizr

hypothesizr is an R package for scraping annotation data collected through the hypothesis app. This could be used for identifying tagged sections in need of revision, or for other analytic purposes.

1.4.8 Zotero

Zotero is a free and open source citation management tool. It is cloud-based with both browser and app versions. It is easy to create .pdf libraries along with citation information, and then compile bibliographies with citation keys for quickly citing articles. The R bookdown package is compatible with latex-style bibliography files that can be exported by Zotero.

1.5 Quickstart

If you want to skip to getting up and running, follow these steps. If they all work, then you should be able to copy the source code this book, compile it and display it as a webbook (or pdf or epub), and then start modifying the text content to create your own resources.

1.5.1 Download all the programs, and set-up needed accounts

1. Download and install R
2. Download and install R-Studio
3. Open R-studio, choose the packages tab, search for the package “bookdown”, download and install
4. Create an account with Github
5. Download and install Github Desktop

1.5.2 Create a Github repository

6. Create a new repository on the Github website, give it a name
7. Clone the repository to your local computer using Github Desktop
8. Open the local repository

1.5.3 Copy and compile this repository

9. Download this repository as a .zip file to your computer
10. Copy the contents of the folder to your new local folder for your github repository
11. In Github Desktop you should now see that there are many new changes to your repository. Commit the changes (uploads them to the online repository)
12. Open the OERbookdown.Rproj in Rstudio, this is an R project file that will automatically point R to the files in this folder.
13. In Rstudio, on the top right panel, choose the Build tab, then click build book.
14. If everything goes smoothly, you will have compiled this book into a web-book, a .pdf and an epub. The compiled books are in the “docs” folder
15. After the compilation process, a viewable web-book should be displayed on your computer.
16. Upload any changes in your repository to Github using the commit button in Github Desktop.

1.5.4 Serve your book on the web using github pages

In addition to providing free repositories, Github also allows files in the repositories to be served as web-pages. Set this up using the following steps:

16. Log-in to your github account through a web browser
17. Click on your repository
18. You should see all of your files from your local computer in the online repository
19. The html files to build the web-book are located in the docs folder. Github allows html files in a docs file to be rendered as a webpage
20. Go to the settings page for your repository. 21.Scroll down to the Github Pages options
21. Turn on the option to use github pages with the docs folder
22. Github should now display a url where you can view the web-book.

1.5.5 Make your own book

Now that you have the workflow established, you can create your own book by editing the text content of the .Rmd files in this repository. Change the text, then build the book again, then upload it to Github, and you should have a new web-book.

1.6 Description of Files in this Repository

- OERbookdown.Rproj An R-studio project file that automatically points R-studio to all of the files in this folder.
- _bookdown.yml set a few options, like folder where book is compiled, currently “docs”
- _output.yml more configuration options

Each .Rmd file is an Rmarkdown file containing the text for each chapter. The index.Rmd file contains some initialization settings, and the preface content. All other .Rmd files contain the text for each chapter. These need to be edited to create new chapters. Add as many new chapters as you want by creating a new .Rmd file with an incremented number for the new chapter.

- index.Rmd Contains initialization settings, and preface content
- 01-Overview.Rmd Chapter 1 content
- 02-R.Rmd Chapter 2 content etc.
- 03-Bookdown.Rmd
- 04-Github.Rmd

- 05-Hypothesis.Rmd
- 06-Zotero.Rmd
- 07-references.Rmd

Other files

- book.bib bibliography file
- DESCRIPTION miscellaneous
- docs : this is a folder containing the compiled book in html, .pdf and epub formats
- Figures : a folder to store figures that are imported into the book
- LICENSE a license file
- packages.bib bibliography file
- preamble.tex supporting tex file for rendering
- README.md the README file displayed on Github
- style.css style file for webbook
- toc.css style file for webbook table of contents

hypothesisR

Chapter 2

R, R-studio, and RMarkdown

2.1 Quick start steps

1. Download and install R for your platform
2. Download and install R-Studio for your platform (R must be installed first)
3. Open R-studio
4. Open a new RMarkdown document, compile it using the knit button.

2.2 R Markdown

You can open a new R Markdown (.rmd) file from Rstudio through the file menu (New File -> R markdown...), or by clicking the green plus sign icon in the top left-hand corner of R-studio, and choosing R Markdown. This will open a window allowing you to name your new document, and then will open the new R Markdown document in the R-studio text editor. The new document shows some example syntax for writing in R Markdown. You can compile the document to see what it will look like as a .pdf or website by clicking the **Knit** button.

R Markdown offers a convenient way to create content in plain-text, and then display the content as a .pdf or web-page. R Markdown separates content from style. In a typical word-processor, like Microsoft Word, an author formats the style of the content as they create it, the two are bound together. In an R Markdown document the content is written in plain text without any styling. In a second step, the R Markdown document is compiled to display the content in a particular style. This tutorial uses the styles available in the bookdown package. There are other styles available, and you can create and modify your own styles.

R Markdown is commonly used to create notebooks for the results of statistical analysis in R. R Markdown allows an author to write sections of text, together with sections of R code. When the document is compiled, it is displayed as a formatted document with sections of headers and text, interspersed with sections of the R code used to run analyses, along with the output of the analyses. For example, below is a snippet of R code that computes and then displays the mean for the set of numbers 1 to 10.

```
mean(seq(1,10,1))
```

```
## [1] 5.5
```

Although reporting statistical analyses may be the most common use of R Markdown, the framework can also be used to build books, using the bookdown package.

2.3 R Markdown tips

I recommend looking at examples of .rmd files to see how they work. For example, all of the chapters in this tutorial are written in R Markdown using .rmd files. So, you can view the source code for each chapter by opening the .rmd file corresponding to each chapter. You will find the syntax to be fairly basic. Chapter headings are lines preceded by a #. Section headers, and sub, or sub-sub headers, are lines preceded by ##, ###, or ####. Paragraphs are new lines with text. You will also see how to insert hyperlinks to websites, as well as insert graphics.

2.3.1 Inserting Graphics

In the repository for this book, you will find a Figures file. You can place .png and .pdf files in this folder. Then, to insert a graphic you create a short R code snippet, and point R toward the location of the file. See the Figure 1 example in the Overview.Rmd file.

2.3.2 Inserting References

See the chapter on Zotero.

2.3.3 Links for using R Markdown

1. The R-studio website has in-depth tutorials on using R Markdown, available [here](#).
2. Cheatsheet for R Markdown
3. In the Help menu of R-Studio, you can also find cheatsheets and formatting guides for using R Markdown.

Chapter 3

Bookdown

Bookdown is an R package written by Yihui Xie. Bookdown is a tool for stitching together multiple R Markdown files into a whole book. Most impressive, is that the final product can be rendered in multiple formats, including .pdf, epub, and as a web-page.

3.1 R Packages and installing Bookdown

R is an open-source platform. Many developers create and share packages for accomplishing different tasks in R.

Packages can be installed in R-Studio by clicking the packages tab in the lower right-hand window, and then clicking the Install button. If you know the name of the package you want to install, then type it in, and click install.

To install bookdown, type bookdown into the textbox, make sure you have “install dependencies” clicked on, and then press install.

The **packages** tab also shows all of the currently installed R packages on your system. These packages needed to be loaded, or turned on, in order to use them. You can do this by clicking on the checkmark box beside each package.

You may find yourself in a situation where your R Markdown file does not compile, and you get a message saying that a particular package is required. Usually this can be resolved by installing the missing package, using the same method used to install the bookdown package.

3.2 Using Bookdown

Yihui Xie has written a whole tutorial book, using bookdown, to describe how to use the bookdown package. Similarly, check out Yihui Xie’s minimal working example, which contains all of the source code necessary to compile a barebones book.

Similarly, this tutorial was written in bookdown, and is intended to be used as a starting point for you to create your own book.

3.3 Compiling this tutorial using bookdown.

1. Download the Github Repository for this tutorial as a .zip file [here](#).

2. Unzip the file somewhere on your computer.
3. Open R-Studio
4. Navigate to the tutorial folder using the Files tab
5. Open the R Markdown file **index.Rmd** and click the button **Build Book** on the **Build** tab of RStudio (top right-hand corner).

You can build the book as a .pdf, epub, or gitbook (renders as a web-page in the gitbook style). After you build the book, it should pop-up in a viewer screen.

Chapter 4

Github

To quote from wikipedia, “Github is a Web-based Git version control repository hosting service. It is mostly used for computer code.”— and I would add, many other things.

The purpose of this chapter is to describe how Github can be used to serve your web-book (compiled using bookdown), and to share the source code so that others can duplicate and modify your book, or so that others can contribute content to your existing book.

4.1 Getting Started

4.1.1 Make an account with Github

To get started with Github you need to create a free account. You should see a sign up option on the main page.

Your account now consists of your profile page (where you can add information about yourself), and an empty list of repositories. Repositories are Github file folders. They can be purely cloud-based; for example, you can create a repository then add files to it through the web-based interface. You can also create a copy of each repository on your local computer, make changes to the repository on your local computer, and then merge those changes with the online repository.

4.1.2 Install Github Desktop

Github has provided a free and convenient tool for using your local computer to interface with the web-based version of Github. To use the tool, download and install the desktop app for Github. Once the app is installed, make sure that you connect it to your Github account so that you can access repositories that you create.

4.2 Repository workflow

We will use the following workflow:

1. Create a new empty repository on the Github Website
2. Clone the repository to your local computer
3. Add files to the repo on your local computer (work locally)
4. Merge the files back to the Github Website (share final versions)

4.3 Creating a new repository on Github.com

1. Log in to Github.com
2. In the top right corner of the page, click the “+” icon
3. Choose new repository
4. Give it a short name
5. You have a few other options that you can revisit later
6. Click create repository

4.4 Cloning a copy on your computer

1. Open the Github desktop app
2. Login in to your Github account
3. Choose the clone repository option
4. Find the name of the repository you created on Github and select it. Github Desktop will now download the contents of the repository and save it locally (in the file folder of your choice) on your computer

4.5 Make changes to the repository

1. If you are on a mac, choose “show in finder”.
2. You can now edit any of the files locally

4.6 Update the online github repository

1. Any time you change a file, git will track all of the changes that you make
2. To submit changes, choose the commit button
3. All of your changes will now be uploaded to the online repository.

Chapter 5

Hypothes.is

Hypothes.is is a web-browser add-on for annotating the web via inline commenting. This allows anyone to select a snippet of text in a web-browser and then post a comment about the selected text. Annotations can be made publicly or privately. All public annotations can be viewed by anyone else running hypothesis on the same website. Using Hypothesis with your web-book allows you to engage students in interacting with the textbook by allowing them to contribute to content generation (by posting new content), or content revision (by tagging sections in need of revision).

If you publish your bookdown book as a webpage, then you anyone with Hypothes.is can use it to annotate the textbook.

5.1 A case study example

I recently compiled a Research Methods in Psychology textbook using the tools described in this tutorial. The landing page describes how Hypothes.is can be used in conjunction with the textbook.

In class I assigned students the task of downloading Hypothes.is, creating a Hypothes.is account, and then throughout the course gave them various assignments where they were responsible for annotating parts of the textbook.

For example, in one assignment I had students annotate sections of the textbook that were in need of improvement. This allows students to participate in content revision as they read the textbook. Other assignments could include focused online discussion about textbook content using annotations, or using annotations as a way for students to add content to the textbook.

5.2 hypothesisr

hypothesisr is an r package for scraping annotation data collected through the hypothesis app. All public annotations submitted through hypothesis to any website are publically available for download. As a result, the hypothesisr package can be used to download and process the annotations submitted to your website.

5.3 Hypothesis Explorer

As a part of this project, we have created a Shiny app that conveniently displays and manipulates hypothesis annotation databases in a website. The Shiny app is located in this github repository, <https://github.com/>

CrumpLab/hypothesis_explorer

5.3.1 Web-based version

A web-based version of the app can be accessed here: <https://hypothesis-explorer.herokuapp.com>. Note the website currently times out after 55 seconds of inactivity. Reload the page to restart the app. This web-version is currently in beta testing.

5.3.2 Running the app locally

To run this app you need to have R and R-Studio installed on your computer. You will also need to have downloaded the Shiny and hypothesisr packages. You can run the app by either of the following methods

1. You can download the hypothesis_explorer repository as a .zip file, then open the server.r, or ui.r files in R-studio and click run-app
2. You can have R-studio automatically download and the run app using by entering the following in the command line

```
shiny::runGitHub('hypothesis_explorer', 'CrumpLab')
```

5.4 Downloading annotations

The Hypothesis Explorer app currently implements two main functions: **downloading** and **displaying/searching** parts of the publicly available Hypothes.is annotation database.

Download the annotation database associated with a website: Type in the name of the website in the search bar. To see an example, enter “crumplab” into the website search and click the download button.

NOTE OF CAUTION We use the hypothesisr package to scrape parts of the entire Hypothes.is annotation database. As a result, there are some nuances in searching for particular websites that may cause you to accidentally download all of the annotations to a global domain rather than to a particular website.

For example, I have two web-books published on my github domain, these are:

1. <https://www.crumplab.github.io/ResearchMethods>
2. <https://www.crumplab.github.io/programmingforpsych>

I am interested in using the Shiny app to look at only the annotations associated with these two websites. Using “crumplab” as the website will download all of the annotations associated with all subdomains on the site, so it will download annotations for both textbooks. On the other hand, including global domain information by searching “crumplab.github.io” will cause an attempt to download all annotations associated with the entire github domain, which will cause the Shiny app to crash.

5.5 Viewing the annotations Search and subset the database

Hypothesis explorer should automatically download any annotations associated with the website that you entered. At the moment, we have only tested this for websites built with github pages.

After the annotation database has been downloaded the two viewable tabs “Summary”, and “Data” will be autopopulated. The Summary tab gives an overview of the top annotated documents, pages, tags and users. The Data tab presents the entire database.

5.5.1 Viewing all the variables in the database

The left-hand panel includes a “Variables (in Data View)” text box. Click into the box to see all of the names of each variable (column names) in the database. Clicking one of the variable names will add it to the list, and automatically display the content of that column in the database viewer. Likewise, deleting variable names will hide those columns from being displayed.

5.5.2 Subsetting the data

For quickly collapsing aspects of the data by row information, we have four subsetting operations, for displaying only annotations associated with particular documents (e.g., entire books), pages (chapters within a book), tags (associated with each annotation), and users.

5.5.3 Voting feature

When a user submits an annotation, other users can post comments underneath the annotation. For example, if a student highlights a section of the textbook and comments that this section should be improved, another student could reply to the original comment. We have implemented a rudimentary comment voting system using these features of *hypothes.is*. For example, up/down style voting can be implemented by instructing users to vote by using the reply feature. Specifically, up votes can be registered by replying with a numeric value of 1, and down votes can be registered by replying with a numeric value of -1.

Hypothesis Explorer automatically looks for any child posts that contain 1s or -1s, counts the total vote for each parent comment, and then displays the count in a column variable called “votes”.

Chapter 6

Zotero

Zotero is a free, cross-platform tool to help you collect, organize, cite, and share your research sources. Zotero is similar to Mendeley or EndNote.

The purpose of this chapter is to show how Zotero can be used as a reference manager to allow you to cite works in your book, and automatically compile bibliographies or reference sections. The bookdown package uses Latex bibliography files to generate citations and create bibliographies. These .bib files are text-based files with a specific syntax for coding the relevant information in a citation. Each citation in a .bib file has an associated key that is inserted into an Rmarkdown document to generate a citation. We will use Zotero to avoid writing our own .bib from scratch. Instead, there are convenient methods for populating Zotero with a database of references, and for compiling a .bib file from a Zotero database that can be used in bookdown.

6.1 Getting Started

1. Create an account with Zotero, click register in the top-right corner
2. Download the Zotero Desktop app
3. Download the Zotero Connector extension for your web-browser

Zotero operates on the cloud as well as on your desktop. You can connect your online Zotero account with your desktop app in the preferences.

6.2 Populating Zotero with references

There are multiple ways to import references into Zotero. In the Zotero desktop app you can create folders to organize your references.

6.2.1 Drag and Drop .pdfs

1. Create a new folder and name it
2. Drag and drop .pdfs into the folder
3. Highlight the .pdfs, then right-click, and choose “retrieve metadata for pdf”.
4. For most journal articles, Zotero will be able to automatically find the citation information for your .pdf. This will convert the .pdf into a Zotero citation that includes both the citation information, as well as the .pdf

6.2.2 Import citations and pdfs from the web

1. Ensure that your Desktop app is open, and that you have installed the Zotero plugin for your web-browser
2. Use google scholar to search for an article
3. Click the Zotero button in your web-browser
4. You should see a list of all of the articles on the google scholar page.
5. Click any or all the articles you want to import, then import them
6. Zotero will download the citation information along with any associated .pdf to the current folder that is open in the Zotero desktop app.

Zotero is fairly flexible, so the above process will generally work when you are accessing many different databases, and journal web-pages for specific articles.

6.2.3 A note of caution

The citation information that Zotero downloads is sometimes inaccurate. Be sure to check the fields for each of your citations to ensure they are accurate. For example, page ranges are often missing.

6.3 Generating a .bib file

1. Right-click a Zotero folder
2. Choose “Export Collection”
3. Choose “Bibtex”
4. Save the file
5. Copy the file into the folder for your bookdown project
6. add the file to the bibliography line in the Index.Rmd file.

6.4 Citing references in an RMarkdown file

Citations are add using the following format `@citationkey`, or `[@citationkey]` to place the author,year citation in parentheses. The citation key name is listed for each citation in the bib file. Here are a couple of links with some additional examples: examples from bookdown, and examples from RMarkdown.

The source code for this book contains two .bib files: book.bib and packages.bib. Each citation in those files has an associated citation key. Here is an example of citing the bookdown package (Xie, 2015). This is an example of citing R-core team (R Core Team, 2016)

6.4.1 Cite while you write

A minor inconvenience when using .bib for citations in Rmarkdown is that you have to know the citation key, and these are easy to forget. One option is to load up your .bib file, then search through it to find the citation key.

Another option is to download and install the `citr` package. Once this package is installed, you can use its cite while your write feature. Click the tools menu, addins, then, insert citations. This will open up a window showing all of the citations in your bib files. You can click multiple citations, and then insert the citation keys into your Rmarkdown document. This is convenient method for quickly finding needed citation keys. I recommend first openining the index.rmd file (which points to your .bib files), and then opening the insert citations tools; this will allow the tool to find your .bib files. After this point, you should be able to use the tool when you are working within .Rmd files for each chapter.

Chapter 7

Shiny

Shiny is an R package for creating a web site to interact with R. For example, a website could include various widgets for user input. User input data gets sent to a server running R, which then executes R code. Output from R can be sent back to the webpage in the form of data tables and graphs. A working example using the default shiny app available in R-studio is presented below. Note the shiny app will only display in the web-book version of this tutorial, and not the .pdf or epub versions.

```
knitr::include_app("https://shiny-crump-test.herokuapp.com",  
  height = "600px")
```

In the above example, the shiny app has loaded a dataset into R and plotted a histogram. The sidepanel contains a scrollbar that allows a user to change the bin-size of the histogram. When the scrollbar is moved the new bin-size settings are sent to the server, R recomputes the histogram, and the website dynamically shows a histogram with new bin-sizes.

This example shiny app is running as a standalone website, but it has been embedded inside the web-book see the example from bookdown for more info

7.1 Sharing Shiny Apps

Shiny apps come with a few caveats, including learning how to program in R, learning how to program reactive expressions in Shiny (lots of good tips here, and here), and then figuring out how to host the Shiny app over the web.

Shiny apps can be shared without a web server. For example, the shiny app in this repository https://github.com/CrumpLab/experimentsimulator_ttest can be downloaded and run directly in R-studio. However, in this case, the shiny app is run locally and can not be embedded in a web-book.

Shiny offers free and paid services for hosting Shiny apps on their servers. If you deploy a Shiny app using this method, then simply place the url for your shiny app into the embedding code (see the raw .Rmd file for this chapter see to how the above shiny app was embedded into this page). It is also possible to run your own Shiny server, or to use other free or paid cloud-computing services to run Shiny apps.

7.2 A minimal working example using Heroku

Heroku is a flexible and free (with paid options) cloud-computing service that can be used to serve Shiny apps.

This repository shows an example of configuring Heroku to run an R shiny app: <https://github.com/CrumpLab/testShinyHeroku>

Bibliography

- R Core Team (2016). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.
- Xie, Y. (2015). *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition. ISBN 978-1498716963.