Feasibility and Acceptance of chatbots embedded in healthcare curricula:

–

This document details the evaluation of each resource in terms of the feasibility and acceptance from the end-users. There was evidence of identifying the feasibility of such resources into formal training and studies exist on the acceptance of such resources, with promising results. However, all these studies defined the need for further research in the area until the use of chatbots in healthcare education became common. Furthermore, the creation process of CEPEH resources was significantly different and had improvements to current methods, due to the co-creation process, and use of low cost but effective technology.

Table of Contents

# Introduction

Personalised Healthcare Education is needed to meet growing demand and quality maintenance. There is a growing evidence around chatbots, namely machine conversation systems- these programs have the potential to change the way students learn and search for information.

Chatbots can quiz existing knowledge, enable higher student engagement with a learning task, or support higher-order cognitive activities. In large-scale learning scenarios with a hight student-to-lecturer ratio, chatbots can help tackle the issue of individualized student support and facilitate personalised learning. However, limited examples of chatbots in European Healthcare Curricula have been utilised to combine both the continuum of cognitive processes presented in Bloom’s taxonomy, with the idea that some repetitive tasks can be done with a chatbot- to provide greater access or to scale faculty time.

Thus, CEPEH strategic partnership has co-created open access chatbots utilising artificial intelligence, promoting innovative practices in digital era, by supporting current curricula and fostering open education.

CEPEH Erasmus+ strategic partnership aimed to co-design and implement new pedagogical approaches and, in particular, chatbots for European medical and nursing schools. CEPEH used use participatory design to engage stakeholders (students, healthcare workforce staff, lecturers, clinicians, etc.) in order to co-design effective chatbots and release them as open access resources. Through CEPEH, effective use of digital technologies and open education were be incorporated into healthcare curricula. This enabled students to increase their health and medical related skills through flexible learning.

CEPEH expected that students adopted this new digital pedagogy and improve their skills and competences through flexible personalised learning, while the teaching staff enhanced their e-learning tool co-creation competences and make use of co-design best practices and recommendations for use. It is also expected increased cooperation between the partners. Thus, in the long term, CEPEH expects to influence the development of medical and nursing curricula with this digital innovation, foster the quality of the future healthcare workforce and further improve international competitiveness of the partners’ healthcare curricula. This document details the evaluation of the resources created by the CEPEH team.

The evaluation specifically explored the feasibility and acceptance from the end-users. These end-users are learners in European healthcare higher education institutions.

There was firstly evidence for the need to identify the feasibility of chatbots and similar resources into formal education and training, with a further need to improve access to these types of learning resources. Of course, studies exist on the acceptance of chatbots, virtual patients, and many other healthcare applications, with promising results. However, through various limitations, we believed there was further research to be completed to accelerate the design, development, implementation, and evaluation processes. These have financial, stakeholder, time, and efficacy benefits. The creation process of CEPEH resources was significantly different to most in the literature, and this report highlights the approach of the CEPEH team towards enhancing personalised healthcare education can be achieved.

## Background

The working practices of CEPEH are aimed at maximizing efficacy of these chatbots as learning resources, and provided a sense of shared development and ownership from all stakeholders. The process normally begins with workshops in which the project is scoped and team building occurs. The CEPEH workshops involve the widest possible team of stakeholders including tutors, students, healthcare workers, learning technologists, health service users and carers- depending on the materials being created.

For readers who are interested in using these high quality digital resources please access them for free at CEPEH.EU

The next section will now present the evaluation of all CEPEH chatbot resources.

# 1 Method

## 1.1 Participants

This data set had 109148 participants

## 1.2 Procedure

## 1.3 Design

## 1.4 Materials

### 1.4.1 System Usability Scale

The System Usability Scale (SUS) was used [10] and is a widely used and adopted usability questionnaire. It is popular due to its unbiased and agnostic properties, a non proprietary, and quick scale of 10 questions.

1. I think that I would like to use this system frequently.
2. I found the system unnecessarily complex.
3. I thought the system was easy to use.
4. I think that I would need the support of a technical person to be able to use this system.
5. I found the various functions in this system were well integrated.
6. I thought there was too much inconsistency in this system.
7. I would imagine that most people would learn to use this system very quickly.
8. I found the system very cumbersome to use.
9. I felt very confident using the system.
10. I needed to learn a lot of things before I could get going with this system.

The SUS was developed with a scoring system, in which the following should be performed: For each of the odd numbered questions, subtract 1 from the score. For each of the even numbered questions, subtract their value from 5. Add up these numbers to find the total score, then multiply this by 2.5. The result is a score out of 100 and can be compared against a determined average score of 68. Further, 80.3 or higher is excellent, and 51 or under suggests significant usability problems.

### 1.4.2 Computer Self-Efficacy Scale Tool

The 10 question CSEST was based on the 32-item questionnaire by Murphy, Coover, and Owen (1989). Participants were provided with the facilitator stating ’Imagine you have found a new technology product that you have previously not used. You believe this product will make your life better. It doesn’t matter specifically what this technology product does, only that it is intended to make your life easier and that you have never used it before. I could use the new technology…

1. If there was no one around to tell me what to do as I go
2. If I had never used a product like it before
3. If I had only the product manuals for reference
4. If I had seen someone else using it before trying it myself
5. If I could call someone for help if I got stuck
6. If someone else had helped me get started
7. If I had a lot of time to complete the job for which the product was provided
8. If I had just the built-in help facility for assistance
9. If someone showed me how to do it first
10. If I had used similar products before this one to do the same job

### 1.4.3 Unified Theory of Acceptance and Use of Technology

### 1.4.4 Technology Acceptance Model (TAM)

The Technology Acceptance Model (TAM) [1] was specifically developed with the primary aim of identifying the determinants involved in computer acceptance in general; secondly, to examine a variety of information technology usage behaviours; and thirdly, to provide a parsimonious theoretical explanatory model. TAM suggests that attitude would be a direct predictor of the intention to use technology, which in turn would predict the actual usage of the technology. The only modification to the nine sub-scales of the questionnaire consists of applying the items to the context of chatbots. All the items, except those measuring attitudes, utilize a seven-point Likert scale ranging from “strongly agree” to “strongly disagree” with a middle neutral point [2].

The nine sub-scales of the questionnaire:

• Ease of use of chatbots • Perceived usefulness of chatbots • Intention of use. • Attitude toward usage of chatbots. • Perception of personal efficacy to use a chatbot resource. • Perception of external control toward chatbots. • Anxiety toward chatbot use. • Intrinsic motivation to use chatbot resources. • Perceived costs of chatbots.

### 1.4.5 Qualitative Measure- Focus Group Discussions

Focus groups are a pervasive means of market research and provides credible acceptance evaluators regarding the penetration that a product or service will have on a target demographic. Focus groups are a form of qualitative research consisting of interviews or structured discussions, in which a group of people are asked about their perceptions, opinions, beliefs, and attitudes towards a product, service, concept, advertisement, idea, or packaging. Questions are asked in an interactive group setting where participants are free to talk with other group members. During this process, the researcher either takes notes or records the vital points he or she is getting from the group. Researchers select members of the focus group carefully for effective and authoritative responses. Relevant stakeholders, then, can use the information collected through focus groups to receive insights on a specific product, issue, or topic focus [7].

A series of short focus group sessions identified the feasibility of CEPEH resources for formal curricular integration. These sessions, spanning no more than 1-1.5 hours and consisting of no more than 5-7 persons each explored all axes of curricular integration such as accessibility in the classroom, use case scenarios, technology requirements for curricular integration etc. These axes were formalized by the research team, in each evaluation site, to consider the curricular details of each institution.

## 1.5 Analysis

# 2 Results

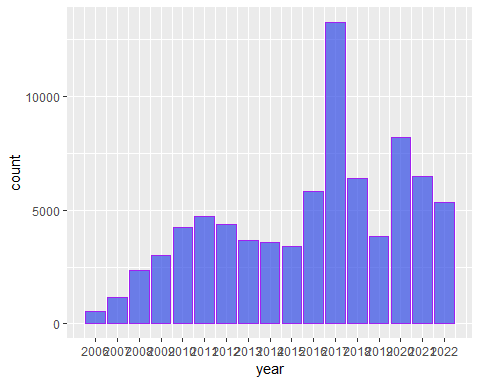
Have users in previous years shared the HELM Open RLO catalogue?

Well, we have so much data we haven’t looked through it all yet.

33,571 learners told us how they found out- each answer is different.

We are through about 10% of this data and will update weekly.

## 2.1 System Usability Scale (SUS) Scores

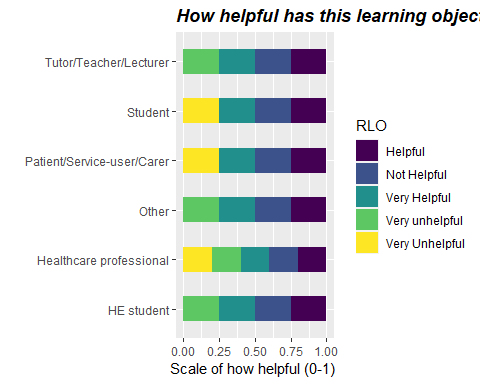


To keep it short, here is a summary table showing if Stakeholders found the RLO(s) they used to be *worth recommending to others*. We hope there are few NO responses, and many YES responses.

| Identity | Recommend | n |
| --- | --- | --- |
| HE student | No | 30 |
| HE student | Yes | 786 |
| Healthcare professional | No | 55 |
| Healthcare professional | Yes | 1311 |
| Other | No | 34 |
| Other | Yes | 767 |
| Patient/Service-user/Carer | No | 2 |
| Patient/Service-user/Carer | Yes | 49 |
| Student | No | 152 |
| Student | Yes | 3164 |
| Tutor/Teacher/Lecturer | No | 5 |
| Tutor/Teacher/Lecturer | Yes | 329 |

| Identity | RLO | n |
| --- | --- | --- |
| HE student | Helpful | 303 |
| HE student | Not Helpful | 8 |
| HE student | Very Helpful | 474 |
| HE student | Very unhelpful | 10 |
| Healthcare professional | Helpful | 501 |
| Healthcare professional | Not Helpful | 28 |
| Healthcare professional | Very Helpful | 817 |
| Healthcare professional | Very unhelpful | 13 |
| Healthcare professional | Very Unhelpful | 13 |
| Other | Helpful | 322 |
| Other | Not Helpful | 24 |
| Other | Very Helpful | 451 |
| Other | Very unhelpful | 5 |
| Patient/Service-user/Carer | Helpful | 16 |
| Patient/Service-user/Carer | Not Helpful | 1 |
| Patient/Service-user/Carer | Very Helpful | 42 |
| Patient/Service-user/Carer | Very Unhelpful | 2 |
| Student | Helpful | 1392 |
| Student | Not Helpful | 77 |
| Student | Very Helpful | 1863 |
| Student | Very Unhelpful | 9 |
| Tutor/Teacher/Lecturer | Helpful | 97 |
| Tutor/Teacher/Lecturer | Not Helpful | 3 |
| Tutor/Teacher/Lecturer | Very Helpful | 229 |
| Tutor/Teacher/Lecturer | Very unhelpful | 1 |

if you don’t like boring tables, here is the same data in a graph!

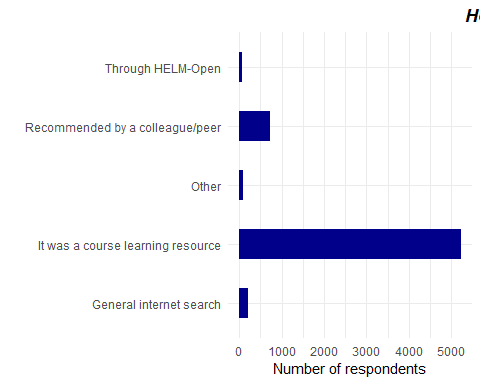


## 2.2 Discovering the HELM Open RLOs

### 2.2.1 Course Learning, Recommendations, and more

The data showed that learners *strongly recommend* the RLO(s) they used to others, but how does this translate?

For the 10% of data we have, the figure below ’*How did you find out about the RLO you used’* shows 700 respondents were recommended from a friend, peer, tutor, or other.



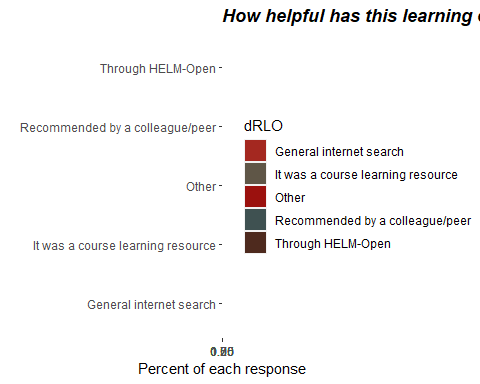
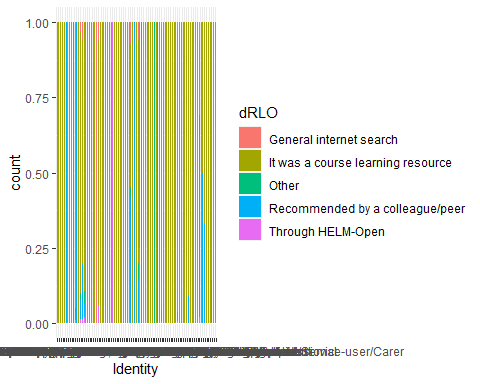
This figure also shows how more than 6000 respondents first used the RLOs as instructed by their tutors on their course.

and if you’re browsing the internet for information on a healthcare topic and come across <https://www.nottingham.ac.uk/helmopen/> :- You’re 1 of about 350 people finding about our resources from internet search. Hopefully that grows, but it seems social networking is the key to sharing these tools.

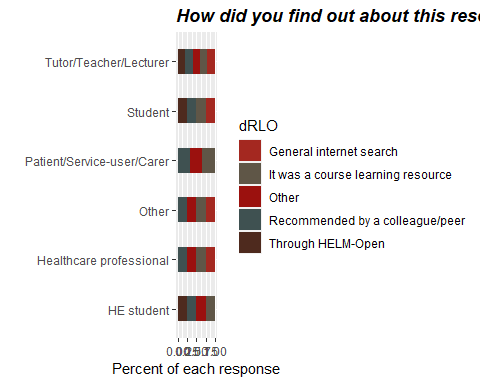
A random sample of other sources are: Twitter, Aim higher days, Barnardos ignite learning, and, well, ‘*a random Google photo*;’- our online presence seems to be in many places!

UP TO HERE1

# A tibble: 24 × 3  
# Groups: Identity, dRLO [24]  
 Identity dRLO n  
 <chr> <chr> <int>  
 1 HE student It was a course learning resource 80  
 2 HE student Other 2  
 3 HE student Recommended by a colleague/peer 4  
 4 HE student Through HELM-Open 1  
 5 Healthcare professional General internet search 7  
 6 Healthcare professional It was a course learning resource 131  
 7 Healthcare professional Other 3  
 8 Healthcare professional Recommended by a colleague/peer 10  
 9 Other General internet search 8  
10 Other It was a course learning resource 50  
# … with 14 more rows



# A tibble: 24 × 3  
# Groups: Identity, dRLO [24]  
 Identity dRLO n  
 <chr> <chr> <int>  
 1 HE student It was a course learning resource 80  
 2 HE student Other 2  
 3 HE student Recommended by a colleague/peer 4  
 4 HE student Through HELM-Open 1  
 5 Healthcare professional General internet search 7  
 6 Healthcare professional It was a course learning resource 131  
 7 Healthcare professional Other 3  
 8 Healthcare professional Recommended by a colleague/peer 10  
 9 Other General internet search 8  
10 Other It was a course learning resource 50  
# … with 14 more rows



### 2.2.2 Italics and bold

* *Italics* are done like \*this\* or \_this\_
* **Bold** is done like \*\*this\*\* or \_\_this\_\_
* ***Bold and italics*** is done like \*\*\*this\*\*\*, \_\_\_this\_\_\_, or (the most transparent solution, in my opinion) \*\*\_this\_\*\*

### 2.2.3 Inline code

* Inline code is created with backticks like `this`

### 2.2.4 Sub and superscript

Sub2 and super2 script is created like this~2~ and this^2^

### 2.2.5 Strikethrough

* ~~Strikethrough~~ is done ~~like this~~

### 2.2.6 ‘Escaping’ (aka “What if I need an actual asterisk?”)

* To include an actual \*, \_ or \, add another \ in front of them: \\*, \\_, \\

### 2.2.7 Endash (–), emdash (—)

* – and — with -- and ---

### 2.2.8 Blockquotes

Do like this:

Put a > in front of the line.

### 2.2.9 Headings

Section headers are created with #’s of increasing number, i.e.

* # First-level heading
* ## Second-level heading
* ### Etc.

In PDF output, a level-five heading will turn into a paragraph heading, i.e. \paragraph{My level-five heading}, which appears as bold text on the same line as the subsequent paragraph.

### 2.2.10 Lists

Unordered list by starting a line with an \* or a -:

* Item 1
* Item 2

Ordered lists by starting a line with a number. Notice that you can mislabel the numbers and *Markdown* will still make the order right in the output:

1. Item 1
2. Item 2

To create a sublist, indent the values a bit (at least four spaces or a tab):

1. Item 1
2. Item 2
3. Item 3
   * Item 3a
   * Item 3b

### 2.2.11 Line breaks

The official *Markdown* way to create line breaks is by ending a line with more than two spaces.

Roses are red. Violets are blue.

This appears on the same line in the output, because we didn’t add spaces after red.

Roses are red.

Violets are blue.

This appears with a line break because I added spaces after red.

I find this is confusing, so I recommend the alternative way: Ending a line with a backslash will also create a linebreak:

Roses are red.

Violets are blue.

To create a new paragraph, you put a blank line.

Therefore, this line starts its own paragraph.

### 2.2.12 Hyperlinks

* [This is a hyperlink](https://www.google.com) created by writing the text you want turned into a clickable link in [square brackets followed by a](https://hyperlink-in-parentheses)

### 2.2.13 Footnotes

* Are created[[1]](#footnote-66) by writing either ^[my footnote text] for supplying the footnote content inline, or something like [^a-random-footnote-label] and supplying the text elsewhere in the format shown below [[2]](#footnote-67):

[^a-random-footnote-label]: This is a random test.

### 2.2.14 Comments

To write comments within your text that won’t actually be included in the output, use the same syntax as for writing comments in HTML. That is, .

### 2.2.15 Math

The syntax for writing math is stolen from LaTeX. To write a math expression that will be shown **inline**, enclose it in dollar signs. - This: $A = \pi\*r^{2}$ Becomes:

To write a math expression that will be shown in a block, enclose it in two dollar signs.

This: $$A = \pi\*r^{2}$$

Becomes:

To create numbered equations, put them in an ‘equation’ environment and give them a label with the syntax (\#eq:label), like this:

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

Becomes: $$

For more (e.g. how to theorems), see e.g. the documentation on [bookdown.org](https://bookdown.org/yihui/bookdown/markdown-extensions-by-bookdown.html#equations)

## 2.3 Executable code chunks

The magic of R Markdown is that we can add executable code within our document to make it dynamic.

We do this either as *code chunks* (generally used for loading libraries and data, performing calculations, and adding images, plots, and tables), or *inline code* (generally used for dynamically reporting results within our text).

The syntax of a code chunk is shown in Figure 2.1.

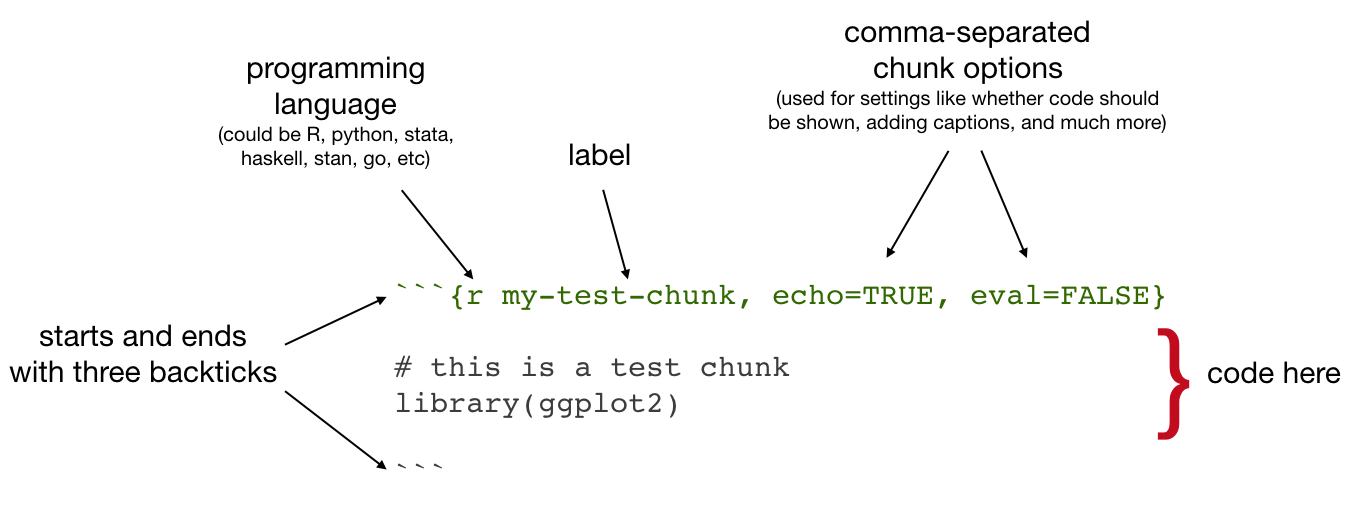


Figure 2.1: Code chunk syntax

Common chunk options include (see e.g. [bookdown.org](https://bookdown.org/yihui/rmarkdown/r-code.html)):

* echo: whether or not to display code in knitted output
* eval: whether or to to run the code in the chunk when knitting
* include: whether to include anything from the from a code chunk in the output document
* fig.cap: figure caption
* fig.scap: short figure caption, which will be used in the ‘List of Figures’ in the PDF front matter

**IMPORTANT**: Do *not* use underscoores in your chunk labels - if you do, you are likely to get an error in PDF output saying something like “! Package caption Error: \caption outside float”.

### 2.3.1 Setup chunks - setup, images, plots

An R Markdown document usually begins with a chunk that is used to **load libraries**, and to **set default chunk options** with knitr::opts\_chunk$set.

In your thesis, this will probably happen in **index.Rmd** and/or as opening chunks in each of your chapters.

```{r setup, include=FALSE}  
# don't show code unless we explicitly set echo = TRUE  
knitr::opts\_chunk$set(echo = FALSE)  
  
library(tidyverse)  
```

### 2.3.2 Including images

Code chunks are also used for including images, with include\_graphics from the knitr package, as in Figure 2.2



Figure 2.2: Oxford logo

Useful chunk options for figures include:

* out.width (use with a percentage) for setting the image size
* if you’ve got an image that gets waaay to big in your output, it will be constrained to the page width by setting out.width = "100%"

#### Figure rotation

You can use the chunk option out.extra to rotate images.

The syntax is different for LaTeX and HTML, so for ease we might start by assigning the right string to a variable that depends on the format you’re outputting to:

Then you can reference that variable as the value of out.extra to rotate images, as in Figure 2.3.



Figure 2.3: Oxford logo, rotated

### 2.3.3 Including plots

Similarly, code chunks are used for including dynamically generated plots. You use ordinary code in R or other languages - Figure 2.4 shows a plot of the cars dataset of stopping distances for cars at various speeds (this dataset is built in to **R**).

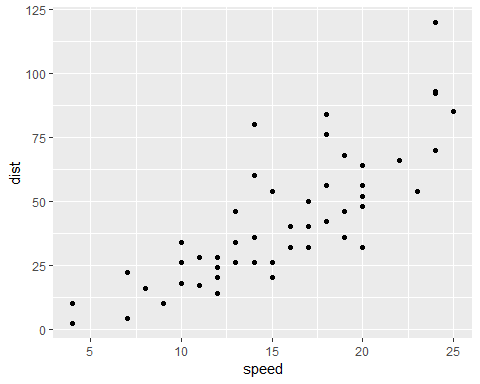


Figure 2.4: A ggplot of car stuff

Under the hood, plots are included in your document in the same way as images - when you build the book or knit a chapter, the plot is automatically generated from your code, saved as an image, then included into the output document.

### 2.3.4 Including tables

Tables are usually included with the kable function from the knitr package.

Table ?? shows the first rows of that cars data - read in your own data, then use this approach to automatically generate tables.

* Gotcha: when using [kable](https://www.rdocumentation.org/packages/knitr/versions/1.21/topics/kable), captions are set inside the kable function
* The kable package is often used with the [kableExtra](https://cran.r-project.org/web/packages/kableExtra/vignettes/awesome_table_in_html.html) package

### 2.3.5 Control positioning

One thing that may be annoying is the way *R Markdown* handles “floats” like tables and figures. In your PDF output, LaTeX will try to find the best place to put your object based on the text around it and until you’re really, truly done writing you should just leave it where it lies.

In general, you should allow LaTeX to do this, but if you really *really* need a figure to be positioned where you put in the document, then you can make LaTeX attempt to do this with the chunk option fig.pos="H", as in Figure 2.5:



Figure 2.5: An Oxford logo that LaTeX will try to place at this position in the text

As anyone who has tried to manually play around with the placement of figures in a Word document knows, this can have lots of side effects with extra spacing on other pages, etc. Therefore, it is not generally a good idea to do this - only do it when you really need to ensure that an image follows directly under text where you refer to it (in this document, I needed to do this for Figure 4.1 in section 4.1.4). For more details, read the relevant section of the [R Markdown Cookbook](https://bookdown.org/yihui/rmarkdown-cookbook/figure-placement.html).

## 2.4 Executable inline code

‘Inline code’ simply means inclusion of code inside text. The syntax for doing this is `r R\_CODE` For example, `r 4 + 4` will output 8 in your text.

You will usually use this in parts of your thesis where you report results - read in data or results in a code chunk, store things you want to report in a variable, then insert the value of that variable in your text. For example, we might assign the number of rows in the cars dataset to a variable:

We might then write:  
“In the cars dataset, we have `r num\_car\_observations` observations.”

Which would output:  
“In the cars dataset, we have 50 observations.”

## 2.5 Executable code in other languages than R

If you want to use other languages than R, such as Python, Julia C++, or SQL, see [the relevant section of the *R Markdown Cookbook*](https://bookdown.org/yihui/rmarkdown-cookbook/other-languages.html)

# 3 Discussion

The usual way to include citations in an *R Markdown* document is to put references in a plain text file with the extension **.bib**, in **BibTex** format.[[3]](#footnote-101) Then reference the path to this file in **index.Rmd**’s YAML header with bibliography: example.bib.

Most reference managers can create a .bib file with you references automatically. However, the **by far** best reference manager to use with *R Markdown* is [Zotero](https://www.zotero.org) with the [Better BibTex plug-in](https://retorque.re/zotero-better-bibtex/), because the citr plugin for RStudio (see below) can read references directly from your Zotero library!

Here is an example of an entry in a **.bib** file:

@article{Shea2014,  
 author = {Shea, Nicholas and Boldt, Annika},  
 journal = {Trends in Cognitive Sciences},  
 pages = {186--193},  
 title = {{Supra-personal cognitive control}},  
 volume = {18},  
 year = {2014},  
 doi = {10.1016/j.tics.2014.01.006},  
}

In this entry highlighted section, ‘Shea2014’ is the **citation identifier**. To default way to cite an entry in your text is with this syntax: [@citation-identifier].

So I might cite some things ([Lottridge et al., 2012](#ref-Lottridge2012); [Mill, 1965 [1843]](#ref-Mill1965); [Shea et al., 2014](#ref-Shea2014)).

### 3.0.1 Appearance of citations and references section (pandoc)

By default, oxforddown lets [Pandoc](https://pandoc.org) handle how citations are inserted in your text and the references section. You can change the appearance of citations and references by specifying a CSL (Citation Style Language) file in the csl metadata field of **index.Rmd**. By default, oxforddown by the Americal Psychological Association (7th Edition), which is an author-year format.

With this style, a number of variations on the citation syntax are useful to know:

* Put author names outside the parenthesis
  + This: @Shea2014 says blah.
  + Becomes: Shea et al. ([2014](#ref-Shea2014)) says blah.
* Include only the citation-year (in parenthesis)
  + This: Shea et al. says blah [-@Shea2014]
  + Becomes: Shea et al. says blah ([2014](#ref-Shea2014))
* Add text and page or chapter references to the citation
  + This: [see @Shea2014, pp. 33-35; also @Wu2016, ch. 1]
  + Becomes: Blah blah (see [Shea et al., 2014, pp. 33–35](#ref-Shea2014); also [Wu, 2016](#ref-Wu2016), ch. 1).

If you want a numerical citation style instead, try csl: bibliography/transactions-on-computer-human-interaction.csl or just have a browse through the [Zotero Style Repository](https://www.zotero.org/styles) and look for one you like. For convenience, you can set the line spacing and the space between the bibliographic entries in the reference section directly from the YAML header in **index.Rmd**.

If you prefer to use biblatex or natbib to handle references, see [this chapter](#customising-citations).

### 3.0.2 Insert references easily with RStudio’s Visual Editor

For an easy way to insert citations, use RStudio’s [Visual Editor](https://rstudio.github.io/visual-markdown-editing/citations.html). Make sure you have the latest version of RStudio – the visual editor was originally really buggy, especially in relation to references, but as per v2022.02.0, it’s great!

## 3.1 Cross-referencing

We can make cross-references to **sections** within our document, as well as to **figures** (images and plots) and **tables**.

The general cross-referencing syntax is **\@ref(label)**

### 3.1.1 Section references

Headers are automatically assigned a reference label, which is the text in lower caps separated by dashes. For example, # My header is automatically given the label my-header. So # My header can be referenced with \@ref(my-section)

Remember what we wrote in section ???

We can also use **hyperlink syntax** and add # before the label, though this is only guaranteed to work properly in HTML output:

* So if we write Remember what we wrote up in [the previous section](#citations)?
* It becomes Remember what we wrote up in [the previous section](#citations)?

#### 3.1.1.1 Creating custom labels

It is a very good idea to create **custom labels** for our sections. This is because the automatically assigned labels will change when we change the titles of the sections - to avoid this, we can create the labels ourselves and leave them untouched if we change the section titles.

We create custom labels by adding {#label} after a header, e.g. # My section {#my-label}. See [our chapter title](#cites-and-refs) for an example. That was section 3.

### 3.1.2 Figure (image and plot) references

* To refer to figures (i.e. images and plots) use the syntax \@ref(fig:label)
* **GOTCHA**: Figures and tables must have captions if you wish to cross-reference them.

Let’s add an image: 

We refer to this image with \@ref(fig:captain). So Figure 3.1 is [this image](#fig:captain).

And in Figure 2.4 we saw a [cars plot](#fig:cars-plot).

### 3.1.3 Table references

* To refer to tables use the syntax \@ref(tab:label)

Let’s include a table:

We refer to this table with \@ref(tab:cars-table2). So Table ?? is [this table](#tab:cars-table2).

And in Table ?? we saw more or less [the same cars table](#tab:cars-table).

### 3.1.4 Including page numbers

Finally, in the PDF output we might also want to include the page number of a reference, so that it’s easy to find in physical printed output. LaTeX has a command for this, which looks like this: \pageref{fig/tab:label} (note: curly braces, not parentheses)

When we output to PDF, we can use raw LaTeX directly in our .Rmd files. So if we wanted to include the page of the cars plot we could write:

* This: Figure <a href="#fig:cars-plot">2.4</a> on page \pageref(fig:cars-plot)
* Becomes: Figure 2.4 on page

#### 3.1.4.1 Include page numbers only in PDF output

A problem here is that LaTeX commands don’t display in HTML output, so in the gitbook output we’d see simply “Figure 2.4 on page”.

One way to get around this is to use inline R code to insert the text, and use an ifelse statement to check the output format and then insert the appropriate text.

* So this: `r ifelse(knitr::is\_latex\_output(), "Figure \<a href="#fig:cars-plot">2.4</a> on page \\pageref{fig:cars-plot}", "")`
* Inserts this (check this on both PDF and gitbook):

Note that we need to escape the backslash with another backslash here to get the correct output.

## 3.2 Collaborative writing

Best practices for collaboration and change tracking when using R Markdown are still an open question. In the blog post [**One year to dissertate**](https://livefreeordichotomize.com/2018/09/14/one-year-to-dissertate/) by Lucy D’Agostino, which I highly recommend, the author notes that she knits .Rmd files to a word document, then uses the googledrive R package to send this to Google Drive for comments / revisions from co-authors, then incorporates Google Drive suggestions *by hand* into the .Rmd source files. This is a bit clunky, and there are ongoing discussions among the *R Markdown* developers about what the best way is to handle collaborative writing (see [issue #1463](https://github.com/rstudio/rmarkdown/issues/1463) on GitHub, where [CriticMarkup](http://criticmarkup.com) is among the suggestions).

For now, this is an open question in the community of R Markdown users. I often knit to a format that can easily be imported to Google Docs for comments, then go over suggested revisions and manually incorporate them back in to the .Rmd source files. For articles, I sometimes upload a near-final draft to [Overleaf](https://www.overleaf.com/), then collaboratively make final edits to the LaTeX file there. I suspect some great solution will be developed in the not-to-distant future, probably by the RStudio team.

## 3.3 Additional resources

* *R Markdown: The Definitive Guide* - <https://bookdown.org/yihui/rmarkdown/>
* *R for Data Science* - <https://r4ds.had.co.nz>

# 4 Reusable Learning Objects- Exploring Uptake

Here is a (very large) table with all of the currently active RLOS.

## 4.1 Reach, Impact, and Qualatative analysis

Dealing with tables in LaTeX can be painful. This section explains the main tricks you need to make the pain go away.

(Note: if you are looking at the eBook version, you will not see much difference in this section, as it is only relevant for PDF output!)

### 4.1.1 Making your table pretty

When you use kable to create tables, you will almost certainly want to set the option booktabs = TRUE. This makes your table look a million times better:

Compare this to the default style, which looks terrible:

### 4.1.2 If your table is too wide

You might find that your table expands into the margins of the page, like the tables above. Fix this with the kable\_styling function from the [kableExtra](https://haozhu233.github.io/kableExtra/) package:

This scales down the table to fit the page width.

### 4.1.3 If your table is too long

If your table is too long to fit on a single page, set longtable = TRUE in the kable function to split the table across multiple pages.

When you do this, you’ll probably want to make the header repeat on new pages. Do this with the kable\_styling function from kableExtra:

Unfortunately, we cannot use the scale\_down option with a longtable. So if a longtable is too wide, you can either manually adjust the font size, or show the table in landscape layout. To adjust the font size, use kableExtra’s font\_size option:

To put the table in landscape mode, use kableExtra’s landscape function:

### 4.1.4 Max power: manually adjust the raw LaTeX output

For total flexibility, you can adjust the raw LaTeX output from kable/kableExtra that generates the table. Let us consider how we would do this for the example of adjusting the font size if our table is too wide: Latex has a bunch of standard commands that set an approximate font size, as shown below in Figure 4.1.



Figure 4.1: Font sizes in LaTeX

You could use these to manually adjust the font size in your longtable in two steps:

1. Wrap the longtable environment in, e.g., a scriptsize environment, by doing a string replacement in the output from kable/kableExtra
2. Add the attributes that make R Markdown understand that the table is a table (it seems R drops these when we do the string replacement)

# 5 Text Mining, Natural Language Processing, and Sentiment Analysis

This chapter describes a number of additional tips and tricks as well as possible customizations to the oxforddown thesis.

## 5.1 Chunk caching and the **\_bookdown\_files** folder

If you set cache=TRUE in a code chunk, in order to cache its results if it’s time-consuming to run see [the R Markdown documentation](https://bookdown.org/yihui/rmarkdown-cookbook/cache.html), then the files for the caching are stored in the \*\*\_bookdown\_files\*\* folder.

If you don’t use caching and you would like to just have the \*\*\_bookdown\_files\*\* folder deleted after the build process is complete, then set allow\_cache = FALSE in **index.Rmd**’s call to knit\_thesis.

That is, your YAML should then look like this:

knit: (function(input, ...) {  
 thesis\_formats <- "pdf";  
   
 source("scripts\_and\_filters/knit-functions.R");  
 knit\_thesis(input, thesis\_formats, allow\_cache = FALSE, ...)  
 })

## 5.2 Front matter

### 5.2.1 Shorten captions shown in the list of figures (PDF)

You might want your list of figures (which follows the table of contents) to have shorter (or just different) figure descriptions than the actual figure captions.

Do this using the chunk option fig.scap (‘short caption’), for example {r captain-image, fig.cap="A very long and descriptive (and potentially boring) caption that doesn't fit in the list of figures, but helps the reader understand what the figure communicates.", fig.scap="A concise description for the list of figures"

### 5.2.2 Shorten captions shown in the list of tables (PDF)

You might want your list of tables (which follows the list of figures in your thesis front matter) to have shorter (or just different) table descriptions than the actual table captions.

If you are using knitr::kable to generate a table, you can do this with the argument caption.short, e.g.:

knitr::kable(mtcars,  
 caption = "A very long and descriptive (and potentially  
 boring) caption that doesn't fit in the list of figures,  
 but helps the reader understand what the figure   
 communicates.",  
 caption.short = "A concise description for the list of tables")

## 5.3 Shorten running header (PDF)

You might want a chapter’s running header (i.e. the header showing the title of the current chapter at the top of page) to be shorter (or just different) to the actual chapter title.

Do this by adding the latex command \chaptermark{My shorter version} after your chapter title.

For example, chapter 3‘s running header is simply ’Cites and cross-refs’, because it begins like this:

# Citations, cross-references, and collaboration {#cites-and-refs}   
\chaptermark{Cites and cross-refs}

## 5.4 Unnumbered chapters

To make chapters unnumbered (normally only relevant to the Introduction and/or the Conclusion), follow the chapter header with {-}, e.g. # Introduction {-}.

When you do this, you must also follow the heading with these two latex commands:

\adjustmtc  
\markboth{The Name of Your Unnumbered Chapter}{}

Otherwise the chapter’s mini table of contents and the running header will show the previous chapter.

## 5.5 Beginning chapters with quotes (PDF)

The OxThesis LaTeX template lets you inject some wittiness into your thesis by including a block of type savequote at the beginning of chapters. To do this, use the syntax ```{block type='savequote'}.[[4]](#footnote-146)

Add the reference for the quote with the chunk option quote\_author="my author name". You will also want to add the chunk option include=knitr::is\_latex\_output() so that quotes are only included in PDF output.

It’s not possible to use markdown syntax inside chunk options, so if you want to e.g. italicise a book name in the reference use a [‘text reference’](https://bookdown.org/yihui/bookdown/markdown-extensions-by-bookdown.html#text-references): Create a named piece of text with ‘(ref:label-name) My text’, then point to this in the chunk option with quote\_author='(ref:label-name)'.

## 5.6 Highlighting corrections (HTML & PDF)

For when it comes time to do corrections, you may want to highlight changes made when you submit a post-viva, corrected copy to your examiners so they can quickly verify you’ve completed the task. You can do so like this:

### 5.6.1 Short, inline corrections

Highlight **short, inline corrections** by doing [like this]{.correction} — the text between the square brackets will then be highlighted in blue in the output.

Note that pandoc might get confused by citations and cross-references inside inline corrections. In particular, it might get confused by "[what @Shea2014 said]{.correction}" which becomes what Shea et al. ([2014](#ref-Shea2014)) said In such cases, you can use LaTeX syntax directly. The correction highlighting uses the [soul](https://ctan.org/pkg/soul) package, so you can do like this:

* If using biblatex for references, use "\hl{what \textcite{Shea2014} said}
* If using natbib for references, use "\hl{what \cite{Shea2014} said}

Using raw LaTeX has the drawback of corrections then not showing up in HTML output at all, but you might only care about correction highlighting in the PDF for your examiners anyway!

### 5.6.2 Blocks of added or changed material

Highlight entire **blocks of added or changed material** by putting them in a block of type correction, using the syntax ```{block type='correction'}.[[5]](#footnote-152) Like so:

*Note that correction blocks cannot be included in word output.*

### 5.6.3 Stopping corrections from being highlighted

To turn off correction highlighting, go to the YAML header of **index.Rmd**, then:

* PDF output: set corrections: false
* HTML output: remove or comment out - templates/corrections.css

## 5.7 Apply custom font color and highlighting to text (HTML & PDF)

The lua filter that adds the functionality to highlight corrections adds two more tricks: you can apply your own choice of colour to highlight text, or change the font color. The syntax is as follows:

Here’s [some text in pink highlighting]{highlight="pink"}  
Becomes: Here’s some text in pink highlighting.

[Here's some text with blue font]{color="blue"}  
Becomes: Here’s some text with blue font

Finally — never, ever actually do this – [here's some text with black highlighting and yellow font]{highlight="black" color="yellow"}  
Becomes: here’s some text with black highlighting and yellow font

The file **scripts\_and\_filters/colour\_and\_highlight.lua** implements this, if you want to fiddle around with it. It works with both PDF and HTML output.

## 5.8 Adding a second abstract (PDF)

You may need two abstracts in your thesis, if you e.g. need both an abstract in English and some other language.

You can add a second abstract in **index.Rmd** like so:

abstract-second-heading: "Resumé"  
abstract-second: "This is the second abstract, for example in beautiful French."

## 5.9 Including another paper in your thesis - embed a PDF document

You may want to embed existing PDF documents into the thesis, for example if your department allows a ‘portfolio’ style thesis and you need to include an existing typeset publication as a chapter.

In gitbook output, you can simply use knitr::include\_graphics and it should include a scrollable (and downloadable) PDF. You will probably want to set the chunk options out.width='100%' and out.height='1000px':

In LaTeX output, however, this approach can cause odd behaviour. Therefore, when you build your thesis to PDF, split the PDF into an alphanumerically sorted sequence of **single-page** PDF files (you can do this automatically with the package pdftools). You can then use the appropriate LaTeX command to insert them, as shown below (for brevity, in the oxforddown PDF sample content we’re only including two pages). *Note that the chunk option results='asis' must be set.* You may also want to remove margins from the PDF files, which you can do with Adobe Acrobat (paid version) and likely other software.

# install.packages(pdftools)  
# split PDF into pages stored in figures/sample-content/pdf\_embed\_example/split/  
# pdftools::pdf\_split("figures/sample-content/pdf\_embed\_example/Lyngs2020\_FB.pdf",  
# output = "figures/sample-content/pdf\_embed\_example/split/")  
  
# grab the pages  
pages <- list.files("figures/sample-content/pdf\_embed\_example/split", full.names = TRUE)  
  
# set how wide you want the inserted PDFs to be:   
# 1.0 is 100 per cent of the oxforddown PDF page width;  
# you may want to make it a bit bigger  
pdf\_width <- 1.2  
  
# for each PDF page, insert it nicely and  
# end with a page break  
cat(stringr::str\_c("\\newpage \\begin{center} \\makebox[\\linewidth][c]{\\includegraphics[width=", pdf\_width, "\\linewidth]{", pages, "}} \\end{center}"))

## 5.10 Including another paper in your thesis - R Markdown child document

Sometimes you want to include another paper you are currently writing as a chapter in your thesis. Above 5.9, we described the simplest way to do this: include the other paper as a pdf. However, in some cases you instead want to include the R Markdown source from this paper, and have it compiled within your thesis. This is a little bit more tricky, because you need to keep careful track of your file paths, but it is possible by [including the paper as a child document](https://bookdown.org/yihui/rmarkdown-cookbook/child-document.html). There are four main steps:

1. Include the paper as a child document
2. Make file paths compatible with knitting the article on its own, as well as when it’s include in your thesis
3. Make header levels correct
4. Make figure widths correct

### 5.10.1 An example paper in another folder

Take this simple example (files for this are in [this GitHub repository](https://github.com/ulyngs/oxforddown-external-article)):

|--paper\_to\_include  
| |--my\_paper.Rmd  
| |--data  
| | |--cat\_salt.csv  
| |--figures  
| | |--cat.jpg  
|  
|--thesis

As the chart suggests, you have another folder, **paper\_to\_include/** living in the same containing folder as your thesis folder. In the **paper\_to\_include** folder, the file **my\_paper.Rmd** is where you write the paper. In **my\_paper.Rmd**, you read in a CSV file found in the subfolder **data/cats.csv**, and also an image from the subfolder **figures/cat.jpg**.

### 5.10.2 Step 1: Include paper as a child document

In your thesis folder, create an Rmd file for the chapter where you want to include another paper. Add one or more code chunks that include R Markdown files from that paper as child documents:

# Including an external chapter   
  
```{r child = "../paper\_to\_include/my\_paper.Rmd"}  
```

### 5.10.3 Step 2: Make file paths compatible

Use [parameters](https://rmarkdown.rstudio.com/lesson-6.html) to adjust the file path of images based on values you set in the YAML header of an R Markdown file. In **my\_paper.Rmd**, create a parameter called other\_path and set it to an empty string:

---  
title: "A fabulous article in a different folder"  
params:  
 other\_path: ""  
---

In **my\_paper.Rmd**, put this at the start of the filepath when you read in data or include images:

library(tidyverse)  
library(knitr)  
  
cat\_data <- read\_csv(str\_c(params$other\_path, "data/cats.csv"))  
include\_graphics(str\_c(params$other\_path, "figures/cat.jpg"))

Finally, in your thesis folder’s **index.Rmd** file, also create the parameter other\_path. But here, set it to where the **paper\_to\_include/** folder is relative to your thesis folder:

params:  
 other\_path: "../paper\_to\_include/"

#### 5.10.3.1 Note on HTML output

Note that if you want to host an HTML version on your thesis online, you will need to include graphics in the content that you host online - the internet obviously won’t be able to see filepaths that are just referring to stuff in another folder on your computer!

### 5.10.4 Step 3: Make sure header levels are correct

Unless the paper you want to include is also written as a book, your header levels are probably going to be off. That is, the level 1 headers (# Some header) you use for main sections in the other paper turns into chaper titles when included in your thesis.

To avoid this, first *increment all heading levels by one in* ***paper\_to\_include/my\_paper.Rmd*** (# Some header -> ## Some header). Then in **paper\_to\_include/** create a [lua filter](https://bookdown.org/yihui/rmarkdown-cookbook/lua-filters.html#lua-filters) that decrements header levels by one: Create a text file, save it as **reduce\_header\_level.lua**, and give it the content below.

function Header(el)  
 if (el.level <= 1) then  
 error("I don't know how to decrease the level of h1")  
 end  
 el.level = el.level - 1  
 return el  
end

In the YAML header of **paper\_to\_include/my\_paper.Rmd**, use this filter:

---  
title: "A fabulous article in a different folder"  
params:  
 other\_path: ""  
output:  
 pdf\_document:   
 pandoc\_args: ["--lua-filter=reduce\_header\_level.lua"]  
---

Now, your header levels will be correct both when you knit the paper on its own and when its included in your thesis.

NOTE: There might be no need to use a lua filter to shift heading - it seems you could simply use pandoc\_args: ["--shift-heading-level-by=-1"] (see <https://pandoc.org/MANUAL.html#reader-options>)

### 5.10.5 Step 4. Make sure figure widths are correct

It might be that your figure widths when knitting your paper on its own, and when including it in your thesis, need to be different. You can again use parameters to set figure widths.

Imagine you want figure width to be 80% of the page width when knitting your paper on its own, but 100% in your thesis. In **paper\_to\_include/my\_paper.Rmd**, first add a parameter we could call out\_width and set it to the string “80%”:

---  
title: "A fabulous article in a different folder"  
params:  
 other\_path: ""  
 out\_width: "80%"  
output:  
 pdf\_document:   
 pandoc\_args: ["--lua-filter=reduce\_header\_level.lua"]  
---

Then, make sure use that parameter to set the output width when you include figures in **paper\_to\_include/my\_paper.Rmd**:

```{r, out.width=params$out\_width, fig.cap="A very funny cat"}  
include\_graphics(str\_c(params$other\_path, "figures/cat.jpg"))  
```

Finally, create the parameter out\_width in your thesis’ **index.Rmd** file:

params:  
 other\_path: "../paper\_to\_include/"  
 out\_width: "80%"

Now, the output width of your figure will be 80% when knitting your paper on its own, and 100% when knitting it as child document of your thesis.

## 5.11 Customizing citations and referencing

### 5.11.1 Using a .csl file with pandoc

See section 3.0.1.

The only drawbacks to letting pandoc handle citations is that (i) it does not support chapter bibliographies, (ii) if you’re a LaTeX veteran, you might be more comfortable with biblatex or natbib.

### 5.11.2 Using biblatex

To use [biblatex](https://www.overleaf.com/learn/latex/Bibliography_management_with_biblatex) to handle citations, first uncomment this in **index.Rmd**, YAML header:

use-biblatex: true  
bib-latex-options: "style=authoryear, sorting=nyt, backend=biber, maxcitenames=2, useprefix, doi=true, isbn=false, uniquename=false"

Then tell R Markdown to use biblatex when inserting citations, by setting citation\_package: biblatex:

output:  
 bookdown::pdf\_book:  
 citation\_package: biblatex

To customise the appearance of citations, change bib-latex-options. For example, to get **numerical citations**, with references in order of their appearance in the text, set it to

bib-latex-options: "style=numeric-comp, sorting=none, backend=biber, maxcitenames=2, useprefix, doi=true, isbn=false, uniquename=false"

#### 5.11.2.1 Adding chapter bibliographies

If you would like chapter bibliographies, first add “refsection=chapter” to the biblatex options, for example like this:

bib-latex-options: "refsection=chapter, style=authoryear, sorting=nyt, backend=biber, maxcitenames=2, useprefix, doi=true, isbn=false, uniquename=false"

Second, set the parameter insertHeadingInPDF: false in **index.Rmd**, to suppress the inclusion of a ‘References’ heading at the end of the thesis.

params:  
 insertHeadingInPDF: false

Finally insert this line at the end of each chapter, to print the bibliographies there:

\printbibliography[segment=\therefsection,heading=subbibliography]

### 5.11.3 Using natbib

To use [natbib](https://www.overleaf.com/learn/latex/Bibliography_management_with_natbib) to handle citations, first uncomment this in **index.Rmd**, YAML header:

use-natbib: true  
natbib-citation-style: authoryear #for science, you might want numbers,square  
natbib-bibliography-style: templates/ACM-Reference-Format.bst #e.g. "plainnat", unsrtnat, or path to a .bst file

Then tell R Markdown to use natbib when inserting citations, by setting citation\_package: natbib:

output:  
 bookdown::pdf\_book:  
 citation\_package: natbib

To customise the appearance of citations, change what **.bst** file you point to in natbib-bibliography-style.

## 5.12 Customizing the page headers and footers (PDF)

This can now be done directly in **index.Rmd**’s YAML header. If you are a LaTeX expert and need further customisation that what’s currently provided, you can tweak the relevant sections of **templates/template.tex** - the relevant code is beneath the line that begins \usepackage{fancyhdr}.

## 5.13 Diving in to the OxThesis LaTeX template (PDF)

For LaTeX minded people, you can read through **templates/template.tex** to see which additional customisation options are available as well as **templates/ociamthesis.cls** which supplies the base class. For example, **template.tex** provides an option for master’s degree submissions, which changes identifying information to candidate number and includes a word count. At the time of writing, you must set this directly in **template.tex** rather than from the YAML header in **index.Rmd**.

## 5.14 Customising to a different university

### 5.14.1 The minimal route

If the front matter in the OxThesis LaTeX template is suitable to your university, customising oxforddown to your needs could be as simple as putting the name of your institution and the path to your university’s logo in **index.Rmd**:

university: University of You  
university-logo: figures/your-logo-here.pdf

### 5.14.2 Replacing the entire title page with your required content

If you have a **.tex** file with some required front matter from your university that you want to replace the OxThesis template’s title page altogether, you can provide a filepath to this file in **index.Rmd**. oxforddown’s sample content includes and example of this — if you use the YAML below, your front matter will look like this:

alternative-title-page: front-and-back-matter/alt-title-page-example.tex

# Conclusion

If we don’t want Conclusion to have a chapter number next to it, we can add the {-} attribute.

## More info

And here’s some other random info: the first paragraph after a chapter title or section head *shouldn’t be* indented, because indents are to tell the reader that you’re starting a new paragraph. Since that’s obvious after a chapter or section title, proper typesetting doesn’t add an indent there.

This paragraph, by contrast, *will* be indented as it should because it is not the first one after the ‘More info’ heading. All hail LaTeX. (If you’re reading the HTML version, you won’t see any indentation - have a look at the PDF version to understand what in the earth this section is babbling on about).

# (APPENDIX) Appendix

# 6 The First Appendix

This first appendix includes an R chunk that was hidden in the document (using echo = FALSE) to help with readibility:

**In 02-rmd-basics-code.Rmd**

library(tidyverse)  
knitr::include\_graphics("figures/sample-content/chunk-parts.png")

**And here’s another one from the same chapter, i.e. Chapter 2.3:**

knitr::include\_graphics("figures/sample-content/beltcrest.png")

# 7 The Second Appendix, for Fun

# References

Lottridge, D., Marschner, E., Wang, E., Romanovsky, M., & Nass, C. (2012). Browser design impacts multitasking. *Proceedings of the Human Factors and Ergonomics Society 56th Annual Meeting*. <https://doi.org/10.1177/1071181312561289>

Mill, J. S. (1965 [1843]). *A system of logic, ratiocinative and inductive: Being a connected view of the principles of evidence and the methods of scientific investigation*. Longmans.

Shea, N., Boldt, A., Bang, D., Yeung, N., Heyes, C., & Frith, C. D. (2014). Supra-personal cognitive control and metacognition. *Trends in Cognitive Sciences*, *18*(4), 186–193. <https://doi.org/10.1016/j.tics.2014.01.006>

Wu, T. (2016). *The Attention Merchants: The Epic Scramble to Get Inside Our Heads*. Knopf Publishing Group.

1. my footnote text [↑](#footnote-ref-66)
2. This is a random test. [↑](#footnote-ref-67)
3. The bibliography can be in other formats as well, including EndNote (**.enl**) and RIS (**.ris**), see [rmarkdown.rstudio.com/authoring\_bibliographies\_and\_citations](https://rmarkdown.rstudio.com/authoring_bibliographies_and_citations.html). [↑](#footnote-ref-101)
4. For more on custom block types, see the relevant section in [*Authoring Books with R Markdown*](https://bookdown.org/yihui/bookdown/custom-blocks.html). [↑](#footnote-ref-146)
5. In the **.tex** file for PDF output, this will put the content between \begin{correction} and \end{correction}; in gitbook output it will be put between <div class="correction"> and </div>. [↑](#footnote-ref-152)