Oxford University Thesis Template for R Markdown

Author Name

Your College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy

Michaelmas 2018

Abstract

This *R Markdown* template is for writing an Oxford University thesis. The template is built using the Yihui Xie's `bookdown` package, with heavy inspiration and reuse of content from Chester Ismay's `thesisdown` package, and the `OxThesis` LaTeX template (most recently adapted by John McManigle).

This template's sample content include illustrations (mostly taken from `thesisdown`) of how to do the various things you need to write a thesis in R Markdown.

Congratulations for taking a step further into the lands of open, reproducible science, by writing your thesis using a tool that allows you to transparently include tables and dynamically generated plots directly from the underlying data. Hip hooray!

Oxford University Thesis Template for R Markdown

Author Name

Your College

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Michaelmas 2018

For Yihui Xie

Acknowledgements

This is where you will normally thank your advisor, colleagues, family and friends, as well as funding and institutional support. In our case, we will give our praises to the people who developed the ideas and tools that allow us to push open science a little step forward by writing plain-text, transparent, and reproducible theses in R Markdown.

We must be grateful to John Gruber for inventing the original version of Markdown, to John MacFarlane for creating Pandoc (`http://pandoc.org`) which converts Markdown to a large number of output formats, and to Yihui Xie for creating `knitr` which introduced R Markdown as a way of embedding code in Markdown documents, and `bookdown` which added tools for technical and longer-form writing.

Special thanks to Chester Ismay, who created the `thesisdown` package that helped many a PhD student write their theses in R Markdown. And a very special tahnks to John McManigle, whose adaption of Sam Evans' adaptation of Keith Gillow's original maths template for writing an Oxford University DPhil thesis in LaTeX provided the template that I adapted for R Markdown.

Finally, profuse thanks to JJ Allaire, the founder and CEO of RStudio, and Hadley Wickham, the mastermind of the tidyverse without whom we'd all just given up and done data science in Python instead. Thanks for making data science easier, more accessible, and more fun for us all.

Ulrik Lyngs
Linacre College, Oxford
2 December 2018

Abstract

This *R Markdown* template is for writing an Oxford University thesis. The template is built using the Yihui Xie's `bookdown` package, with heavy inspiration and reuse of content from Chester Ismay's `thesisdown` package, and the `OxThesis` LaTeX template (most recently adapted by John McManigle).

This template's sample content include illustrations (mostly taken from `thesisdown`) of how to do the various things you need to write a thesis in R Markdown.

Congratulations for taking a step further into the lands of open, reproducible science, by writing your thesis using a tool that allows you to transparently include tables and dynamically generated plots directly from the underlying data. Hip hooray!

Contents

Appendices

List of Figures

List of Tables

List of Abbreviations

1-D, 2-D	One- or two-dimensional, referring in this thesis to spatial dimensions in an image.
Otter	One of the finest of water mammals.
Hedgehog	Quite a nice prickly friend.

Introduction

Welcome to the *R Markdown* Oxford University thesis template. The sample content is adapted from `thesisdown` and the formatting of PDF output is adapted from the OxThesis LaTeX template. Hopefully writing your thesis in R Markdown will provide a nicer interface for those that have never used TeX or LaTeX before. More importantly, using *R Markdown* allows you to embed chunks of code directly into your thesis, and generate plots and tables directly from their underlying data, avoiding copy-paste steps. Hopefully, this *R Markdown* template gets you in the habit of doing reproducible research, which benefits you long-term as a researcher, but also will greatly help anyone that is trying to reproduce or build onto your results down the road.

Hopefully, you won't have much of a learning period to go through and you will reap the benefits of a nicely formatted thesis. The use of LaTeX in combination with *Markdown* is more consistent than the output of a word processor, much less prone to corruption or crashing, and the resulting file is smaller than a Word file. While you may never have had problems using Word in the past, your thesis is likely going to be about twice as large and complex as anything you've written before, taxing Word's capabilities. After working with *Markdown* and **R** together for a few weeks, we are confident this will be your reporting style of choice going forward.

0.1 Why use it?

R Markdown creates a simple and straightforward way to interface with the beauty of LaTeX. Packages have been written in **R** to work directly with LaTeX to produce nicely formatting tables and paragraphs. In addition to creating a user friendly interface to LaTeX, *R Markdown* also allows you to read in your data, to analyze it and to visualize it using **R**, **Python** or other languages, and also to provide the documentation and commentary on the results of your project. Further, it allows for results of code output to be passed inline to the commentary of your results. You'll see more on this later, focusing on **R**. If you are more into **Python** or something else, you can still use *R Markdown* - see 'Other language engines' in Yihui Xie's *R Markdown: The Definitive Guide.*

0.2 Who should use it?

Anyone who needs to use data analysis, math, tables, a lot of figures, complex cross-references, or who just cares about reproducibility in research should use *R Markdown.* If you are working in 'softer' fields, the user-friendly nature of *Markdown* and its ability to keep track of and easily include figures, automatically generate a table of contents, index, references, table of figures, etc. should still make it of great benefit to nearly anyone writing a thesis project.

Neque porro quisquam est qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit…

There is no one who loves pain itself, who seeks after it and wants to have it, simply because it is pain…

— Cicero's *de Finibus Bonorum et Malorum*

1

R Markdown Basics

Contents

Here is a brief introduction to using *R Markdown*. *Markdown* is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. *R Markdown* provides the flexibility of *Markdown* with the implementation of **R** input and output. For more details on using *R Markdown* see `http://rmarkdown.rstudio.com`.

Be careful with your spacing in *Markdown* documents. While whitespace largely is ignored, it does at times give *Markdown* signals as to how to proceed. As a habit, try to keep everything left aligned whenever possible, especially as you type a new paragraph. In other words, there is no need to indent basic text in the Rmd document (in fact, it might cause your text to do funny things if you do).

1.1 Lists

It's easy to create a list. It can be unordered like

- Item 1
- Item 2

or it can be ordered like

1. Item 1
2. Item 2

Notice that I intentionally mislabeled Item 2 as number 4. *Markdown* automatically figures this out! You can put any numbers in the list and it will create the list. Check it out below.

To create a sublist, just indent the values a bit (at least four spaces or a tab). (Here's one case where indentation is key!)

1. Item 1
2. Item 2
3. Item 3
    - Item 3a
    - Item 3b

1.2 Line breaks

Make sure to add white space between lines if you'd like to start a new paragraph. Look at what happens below in the outputted document if you don't:

Here is the first sentence. Here is another sentence. Here is the last sentence to end the paragraph. This should be a new paragraph.

Now for the correct way: Here is the first sentence. Here is another sentence. Here is the last sentence to end the paragraph.

This should be a new paragraph.

1.3 R chunks

When you click the **Knit** button above a document will be generated that includes both content as well as the output of any embedded **R** code chunks within the document. You can embed an **R** code chunk like this (`cars` is a built-in **R** dataset):

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00  
```

1.4 Inline code

If you'd like to put the results of your analysis directly into your discussion, add inline code like this:

> The `cos` of $2\pi$ is 1.

Another example would be the direct calculation of the standard deviation:

> The standard deviation of `speed` in `cars` is 5.2876444.

One last neat feature is the use of the `ifelse` conditional statement which can be used to output text depending on the result of an **R** calculation:

> The standard deviation is less than 6.

Note the use of `>` here, which signifies a quotation environment that will be indented.

As you see with `$2 \pi$` above, mathematics can be added by surrounding the mathematical text with dollar signs. More examples of this are in Mathematics and Science.

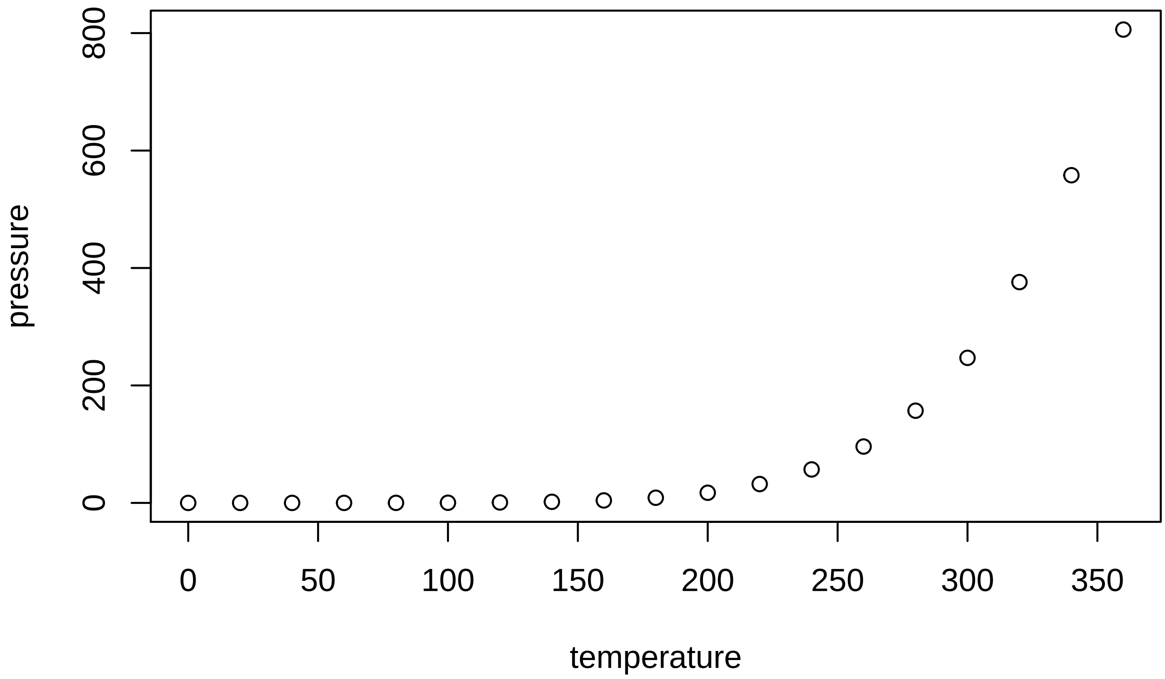

Figure 1.1: A dynamically generated plot

1.5 Including plots

You can also embed plots. For example, Figure 1.1 shows a way to use the base **R** graphics package to produce a plot using the built-in `pressure` dataset.

Note that the `echo=FALSE` parameter was added to the code chunk to prevent printing of the **R** code that generated the plot. There are plenty of other ways to add chunk options. More information is available at `http://yihui.name/knitr/options/`.

Another useful chunk option is the setting of `cache=TRUE` as you see here. If document rendering becomes time consuming due to long computations or plots that are expensive to generate you can use knitr caching to improve performance. Later in this file, you'll see a way to reference plots created in **R** or external figures.

1.6 Loading and exploring data

Included in this template is a file called `flights.csv`. This file includes a subset of the larger dataset of information about all flights that departed from Seattle and Portland in 2014. More information about this dataset and its **R** package is available at `http://github.com/ismayc/pnwflights14`. This subset includes only Portland flights and only rows that were complete with no missing values. Merges were also done with the `airports` and `airlines` data sets in the `pnwflights14` package to get more descriptive airport and airline names.

We can load in this data set using the following command:

```
flights <- read.csv("data/flights.csv")
```

The data is now stored in the data frame called `flights` in **R**. To get a better feel for the variables included in this dataset we can use a variety of functions. Here we can see the dimensions (rows by columns) and also the names of the columns. (Note also how we use the chunk option `vspaceout` to add a bit of white space after each R command's output - this chunk option does not exist by default, but is created by the chunk you will find if you inspect **index.Rmd**)

```
dim(flights)
```

```
## [1] 52808    16
```

```
names(flights)
```

```
##  [1] "month"        "day"          "dep_time"     "dep_delay"
##  [5] "arr_time"     "arr_delay"    "carrier"      "tailnum"
##  [9] "flight"       "dest"         "air_time"     "distance"
## [13] "hour"         "minute"       "carrier_name" "dest_name"
```

Another good idea is to take a look at the dataset in table form. With this dataset having more than 50,000 rows, we won't explicitly show the results of the command here. I recommend you enter the command into the Console ***after*** you have run the **R** chunks above to load the data into **R**.

```
View(flights)
```

While not required, it is highly recommended you use the `dplyr` package to manipulate and summarize your data set as needed. It uses a syntax that is easy to understand using chaining operations. Below I've created a few examples of using dplyr to get information about the Portland flights in 2014. You will also see the use of the ggplot2 package, which produces beautiful, high-quality academic visuals.

We begin by checking to ensure that needed packages are installed and then we load them into our current working environment:

```r
# List of packages required for this analysis
pkg <- c("dplyr", "ggplot2", "knitr", "bookdown", "devtools", "thesisdown")
# Check if packages are not installed and assign the
# names of the packages not installed to the variable new.pkg
new.pkg <- pkg[!(pkg %in% installed.packages())]
# If there are any packages in the list that aren't installed,
# install them
if (length(new.pkg))
  install.packages(new.pkg, repos = "http://cran.rstudio.com")
# Load packages (thesisdown will load all of the packages as well)
library(thesisdown)
```

The example we show here does the following:

- Selects only the `carrier_name` and `arr_delay` from the `flights` dataset and then assigns this subset to a new variable called `flights2`.
- Using `flights2`, we determine the largest arrival delay for each of the carriers.

```
flights2 <- flights %>%
  select(carrier_name, arr_delay)
max_delays <- flights2 %>%
  group_by(carrier_name) %>%
  summarise(max_arr_delay = max(arr_delay, na.rm = TRUE))
```

A useful function in the `knitr` package for making nice tables in *R Markdown* is called `kable`. It is much easier to use than manually entering values into a table by copying and pasting values into Excel or LaTeX. This again goes to show how nice reproducible documents can be! (Note the use of `results="asis"`, which will produce the table instead of the code to create the table.) The `caption.short` argument is used to include a shorter title to appear in the List of Tables.

```
kable(max_delays,
      col.names = c("Airline", "Max Arrival Delay"),
      caption = "Maximum Delays by Airline",
      caption.short = "Max Delays by Airline",
      longtable = TRUE,
      booktabs = TRUE)
```

Table 1.1: Maximum Delays by Airline

Airline	Max Arrival Delay
Alaska Airlines Inc.	338
American Airlines Inc.	1539
Delta Air Lines Inc.	651
Frontier Airlines Inc.	575
Hawaiian Airlines Inc.	407
JetBlue Airways	273
SkyWest Airlines Inc.	421

Southwest Airlines Co.	694
United Air Lines Inc.	472
US Airways Inc.	347
Virgin America	366

The last two options make the table a little easier-to-read.

We can further look into the properties of the largest value here for American Airlines Inc. To do so, we can isolate the row corresponding to the arrival delay of 1539 minutes for American in our original `flights` dataset.

```
flights %>% filter(arr_delay == 1539,
                   carrier_name == "American Airlines Inc.") %>%
  select(-c(month, day, carrier, dest_name, hour,
            minute, carrier_name, arr_delay))
```

```
##   dep_time dep_delay arr_time tailnum flight dest air_time distance
## 1     1403      1553     1934  N595AA   1568  DFW      182     1616
```

We see that the flight occurred on March 3rd and departed a little after 2 PM on its way to Dallas/Fort Worth. Lastly, Figure 1.2 shows how we can visualize the arrival delay of all departing flights from Portland on March 3rd against time of departure.

```
flights %>% filter(month == 3, day == 3) %>%
  ggplot(aes(x = dep_time, y = arr_delay)) + geom_point()
```

1.7 Additional resources

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

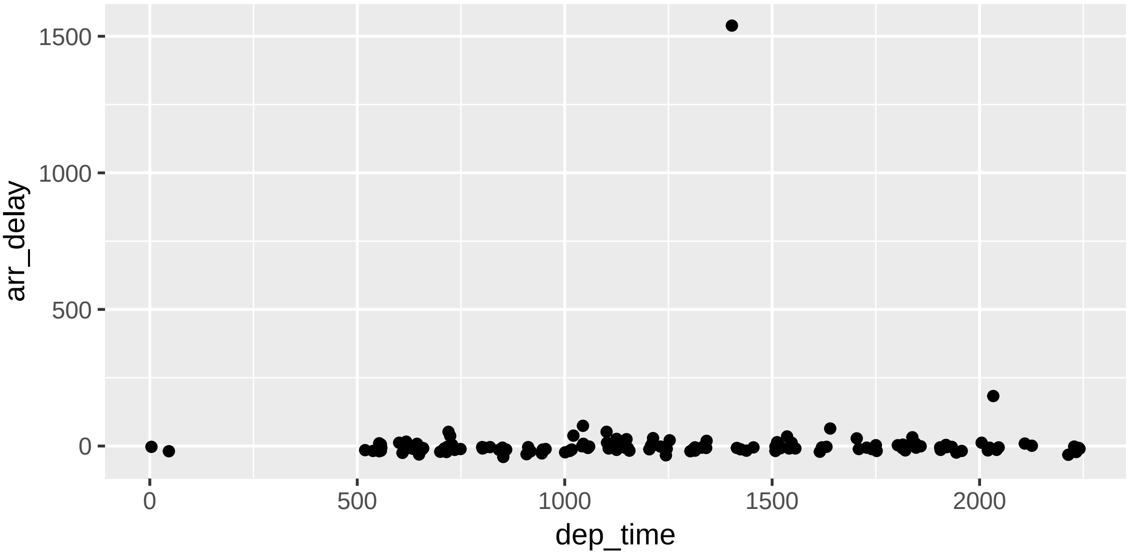

Figure 1.2: Arrival delay of departing flights from Portland on March 3rd.

2
Mathematics and Science

Contents

2.1 Math

We now extend the basics introduced in Chapter 1. TEX is the best way to typeset mathematics. Donald Knuth designed TEX when he got frustrated at how long it was taking the typesetters to finish his book, which contained a lot of mathematics. One nice feature of *R Markdown* is its ability to read LaTeX code directly.

If you are doing a thesis that will involve lots of math, you will want to read this chapter.

$$\sum_{j=1}^{n}(\delta\theta_j)^2 \leq \frac{\beta_i^2}{\delta_i^2+\rho_i^2}\left[2\rho_i^2+\frac{\delta_i^2\beta_i^2}{\delta_i^2+\rho_i^2}\right] \equiv \omega_i^2$$

From Informational Dynamics, we have the following (Dave Braden):

After n such encounters the posterior density for θ is

$$\pi(\theta|X_1 < y_1, \ldots, X_n < y_n) \propto \pi(\theta) \prod_{i=1}^{n} \int_{-\infty}^{y_i} \exp\left(-\frac{(x-\theta)^2}{2\sigma^2}\right)\, dx$$

Another equation:

$$\det \begin{vmatrix} c_0 & c_1 & c_2 & \cdots & c_n \\ c_1 & c_2 & c_3 & \cdots & c_{n+1} \\ c_2 & c_3 & c_4 & \cdots & c_{n+2} \\ \vdots & \vdots & \vdots & & \vdots \\ c_n & c_{n+1} & c_{n+2} & \cdots & c_{2n} \end{vmatrix} > 0$$

Lapidus and Pindar, Numerical Solution of Partial Differential Equations in Science and Engineering. Page 54

$$\int_t \left\{ \sum_{j=1}^{3} T_j \left(\frac{d\phi_j}{dt} + k\phi_j \right) - kT_e \right\} w_i(t)\ dt = 0, \qquad i = 1, 2, 3.$$

L&P Galerkin method weighting functions. Page 55

$$\sum_{j=1}^{3} T_j \int_0^1 \left\{ \frac{d\phi_j}{dt} + k\phi_j \right\} \phi_i\ dt = \int_0^1 k\, T_e \phi_i dt, \qquad i = 1, 2, 3$$

Another L&P (p145)

$$\int_{-1}^{1}\int_{-1}^{1}\int_{-1}^{1} f\big(\xi, \eta, \zeta\big) = \sum_{k=1}^{n}\sum_{j=1}^{n}\sum_{i=1}^{n} w_i w_j w_k f\big(\xi, \eta, \zeta\big).$$

Another L&P (p126)

$$\int_{A_e} (\,\cdot\,) dx dy = \int_{-1}^{1}\int_{-1}^{1} (\,\cdot\,) \det[J] d\xi d\eta.$$

2.2 Chemistry 101: Symbols

Chemical formulas will look best if they are not italicized. Get around math mode's automatic italicizing in LaTeX by using the argument `$\mathrm{formula here}$`, with your formula inside the curly brackets. (Notice the use of the backticks here which enclose text that acts as code.)

So, $\mathrm{Fe_2^{2+}Cr_2O_4}$ is written `$\mathrm{Fe_2^{2+}Cr_2O_4}$`.

Exponent or Superscript: $\mathrm{O^-}$

Subscript: $\mathrm{CH_4}$

To stack numbers or letters as in $\mathrm{Fe_2^{2+}}$, the subscript is defined first, and then the superscript is defined.

Bullet: $\mathrm{CuCl} \bullet \mathrm{7H_2O}$

Delta: Δ

Reaction Arrows: $\longrightarrow$ or $\xrightarrow{solution}$

Resonance Arrows: $\leftrightarrow$

Reversible Reaction Arrows: $\rightleftharpoons$

2.2.1 Typesetting reactions

You may wish to put your reaction in an equation environment, which means that LaTeX will place the reaction where it fits and will number the equations for you.

$$\mathrm{C_6H_{12}O_6 + 6O_2} \longrightarrow \mathrm{6CO_2 + 6H_2O} \tag{2.1}$$

We can reference this combustion of glucose reaction via Equation (2.1).

2.2.2 Other examples of reactions

$\mathrm{NH_4Cl_{(s)}} \rightleftharpoons \mathrm{NH_{3(g)} + HCl_{(g)}}$

$\mathrm{MeCH_2Br + Mg} \xrightarrow[below]{above} \mathrm{MeCH_2 \bullet Mg \bullet Br}$

2.3 Physics

Many of the symbols you will need can be found on the math page `http://web.reed.edu/cis/help/latex/math.html` and the Comprehensive LaTeX Symbol Guide (`http://mirror.utexas.edu/ctan/info/symbols/comprehensive/symbols-letter.pdf`).

2.4 Biology

You will probably find the resources at `http://www.lecb.ncifcrf.gov/~toms/latex.html` helpful, particularly the links to bsts for various journals. You may also be interested in TeXShade for nucleotide typesetting (`http://homepages.uni-tuebingen.de/beitz/txe.html`). Be sure to read the proceeding chapter on graphics and tables.

3
Tables, Graphics, References, and Labels

Contents

3.1 Tables

In addition to the tables that can be automatically generated from a data frame in **R** that you saw in R Markdown Basics using the `kable` function, you can also create tables using *pandoc*. (More information is available at `http://pandoc.org/README.html#tables`.) This might be useful if you don't have values specifically stored in **R**, but you'd like to display them in table form. Below is an example. Pay careful attention to the alignment in the table and hyphens to create the rows and columns.

Table 3.1: Correlation of Inheritance Factors for Parents and Child

Factors	Correlation between Parents & Child	Inherited
Education	-0.49	Yes
Socio-Economic Status	0.28	Slight
Income	0.08	No

Factors	Correlation between Parents & Child	Inherited
Family Size	0.18	Slight
Occupational Prestige	0.21	Slight

We can also create a link to the table by doing the following: Table 3.1. If you go back to Loading and exploring data and look at the `kable` table, we can create a reference to this max delays table too: Table 1.1. The addition of the `(\#tab:inher)` option to the end of the table caption allows us to then make a reference to Table `\@ref(tab:label)`. Note that this reference could appear anywhere throughout the document after the table has appeared.

We will next explore ways to create this label-ref link using figures.

3.2 Figures

If your thesis has a lot of figures, *R Markdown* might behave better for you than that other word processor. One perk is that it will automatically number the figures accordingly in each chapter. You'll also be able to create a label for each figure, add a caption, and then reference the figure in a way similar to what we saw with tables earlier. If you label your figures, you can move the figures around and *R Markdown* will automatically adjust the numbering for you. No need for you to remember! So that you don't have to get too far into LaTeX to do this, a couple **R** functions have been created for you to assist. You'll see their use below.

One thing that may be annoying is the way *R Markdown* handles "floats" like tables and figures (it's really LaTeX's fault). 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. There are some optional arguments specified in the options parameter of the `label` function. If you need to shift your figure around, it might be good to look here on tweaking the options argument: `https://en.wikibooks.org/wiki/LaTeX/Floats,_Figures_and_Captions`

If you need a graphic or tabular material to be part of the text, you can just put it inline. If you need it to appear in the list of figures or tables, it should be placed in a code chunk.

In the **R** chunk below, we will load in the Oxford logo stored as `beltcrest.png` in the figures directory. We then give it the caption of "Oxford logo", the label of "oxfordlogo", and specify that this is a figure. Make note of the different **R** chunk options that are given in the R Markdown file (not shown in the knitted document).

```
include_graphics(path = "figures/beltcrest.png")
```

Here is a reference to the Oxford logo: Figure 3.1. Note the use of the `fig:` code here. By naming the **R** chunk that contains the figure, we can then reference that figure later as done in the first sentence here. We can also specify the caption for the figure via the R chunk option `fig.cap`.

Figure 3.1: Oxford logo

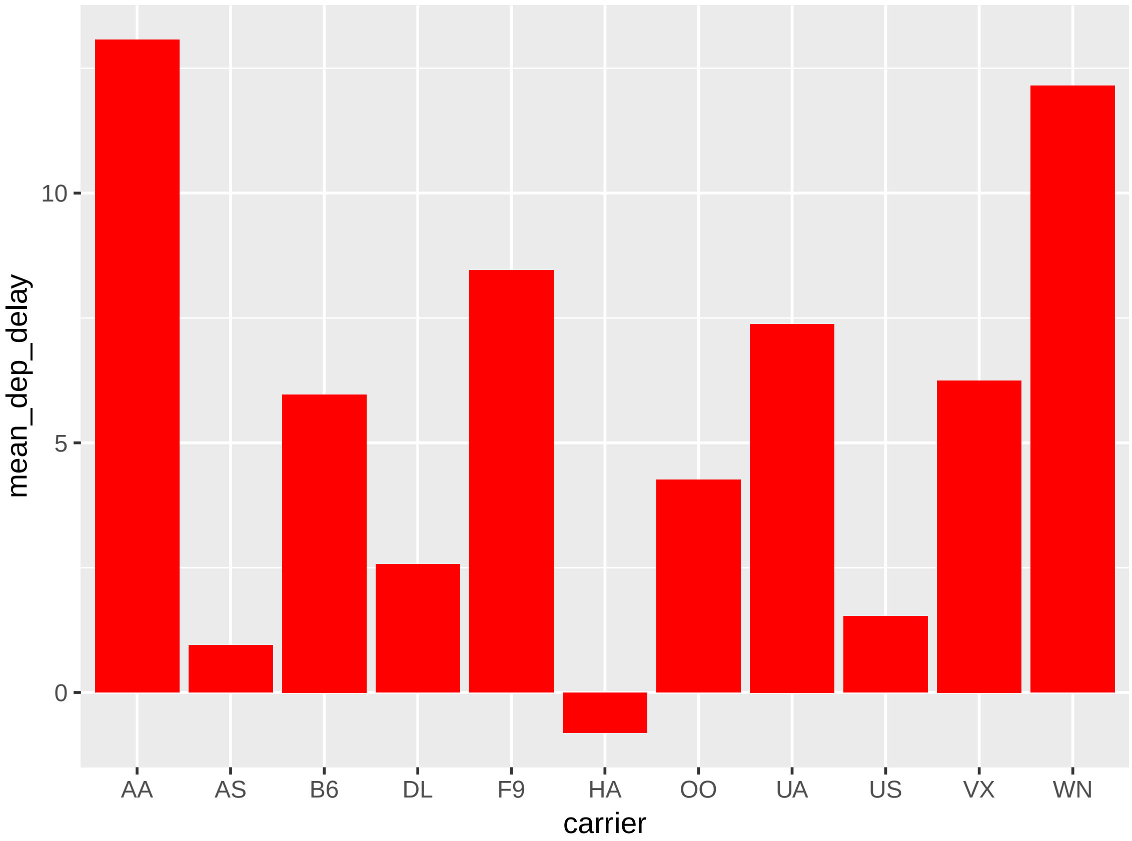

Figure 3.2: Mean Delays by Airline

Below we will investigate how to save the output of an **R** plot and label it in a way similar to that done above. Recall the `flights` dataset from Chapter 1. (Note that we've shown a different way to reference a section or chapter here.) We will next explore a bar graph with the mean flight departure delays by airline from Portland for 2014. Note also the use of the `scale` parameter which is discussed on the next page.

```r
flights %>% group_by(carrier) %>%
  summarize(mean_dep_delay = mean(dep_delay)) %>%
  ggplot(aes(x = carrier, y = mean_dep_delay)) +
  geom_bar(position = "identity", stat = "identity", fill = "red")
```

Here is a reference to this image: Figure 3.2.

A table linking these carrier codes to airline names is available at `https://github.com/ismayc/pnwflights14/blob/master/data/airlines.csv`.

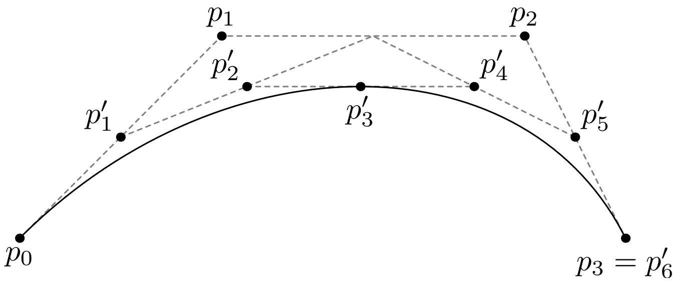

Figure 3.3: Subdiv. graph

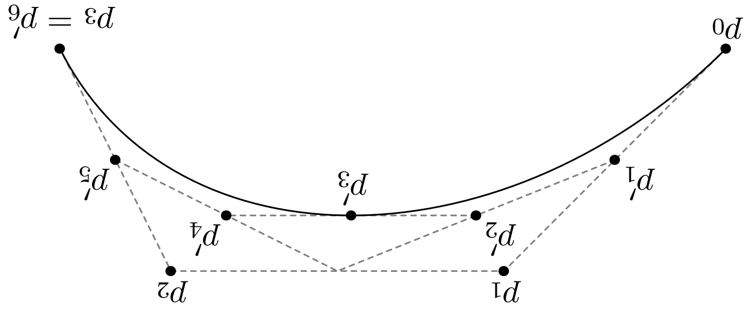

Figure 3.4: A Larger Figure, Flipped Upside Down

Next, we will explore the use of the `out.extra` chunk option, which can be used to shrink or expand an image loaded from a file by specifying `"scale= "`. Here we use the mathematical graph stored in the "subdivision.png" file.

Here is a reference to this image: Figure 3.3. Note that `echo=FALSE` is specified so that the **R** code is hidden in the document.

More Figure Stuff Lastly, we will explore how to rotate and enlarge figures using the `out.extra` chunk option. (Currently this only works in the PDF version of the book.)

As another example, here is a reference: Figure 3.4.

3.3 Footnotes

You might want to footnote something. [1] The footnote will be in a smaller font and placed appropriately.

[1] footnote text

3.4 Bibliographies

Of course you will need to cite things, and you will probably accumulate an armful of sources. There are a variety of tools available for creating a bibliography database (stored with the .bib extension). In addition to BibTeX suggested below, you may want to consider using the free and easy-to-use tool called Zotero.

R Markdown uses *pandoc* (`http://pandoc.org/`) to build its bibliographies. One nice caveat of this is that you won't have to do a second compile to load in references as standard LaTeX requires. To cite references in your thesis (after creating your bibliography database), place the reference name inside square brackets and precede it by the "at" symbol. For example, here's a reference to a book about worrying: [1]. This `Molina1994` entry appears in a file called `references.bib`. This bibliography database file was created by a program called BibTeX. You can call this file something else if you like (look at the YAML header in the main .Rmd file).

Neque porro quisquam est qui dolorem ipsum quia dolor sit amet, consectetur, adipisci velit…

There is no one who loves pain itself, who seeks after it and wants to have it, simply because it is pain…

— Cicero's *de Finibus Bonorum et Malorum*

4 Some Final Notes on The OxThesis template

Contents

4.1 Quotes by beginning of chapters

The OxThesis LaTeX template lets you add some wittiness to you thesis by including a block of type `savequote` at the beginning of chapters, using the syntax `` ```{block type='savequote'} ``. You will probably also want to add the chunk option `include=knitr::opts_knit$get('rmarkdown.pandoc.to') == 'latex'` so these quotes are only included in PDF output.

For more on custom block types, see the relevant section in *Authoring Books with R Markdown*.

4.2 Highlighting of corrections

The OxThesis template supplies two LaTeX commands you can use directly in your .Rmd files if you only care about PDF output. First, the `mccorrect` command allows you to highlight a short correction like this one. When the thesis is typeset normally, the correction will just appear as part of the text. However, when you declare `corrections: yes` in **index.Rmd**, that correction will be highlighted in blue in PDF output. That might be useful for submitting a post-viva, corrected copy to your examiners so they can quickly verify you've completed the task. You must use LaTeX rather than *R Markdown* syntax inside of a `mccorrect` short correction.

The obvious problem is that raw LaTeX doesn't show up in HTML output, and we might want to showcase corrections in our gitbook output as well. To do that, the `oxforddown` template adds a little **R** function also named `mccorrect` that lets us do the same thing (this is created in the setup chunk that begins **index.Rmd**).[1]

Now we can apply corrections inline using this function, which will apply the blue corrections `to background color in both PDF`{mccorrect} and HTML output.[2]

For larger chunks, you can put them in a block of type `mccorrection` (which in the **`.tex`** file for PDF output will put the content between `\begin{mccorrection}` and `\end{mccorrection}`, and in the HTML file for gitbook output between `<div class="mccorrection">` and `</div>`. Rather than doing it directly, we use params$corrections to set it from the YAML header in **index.Rmd**.

For larger chunks, like this paragraph or indeed entire figures, you can use the `mccorrection` environment. This environment **highlights paragraph-sized and larger blocks** with the same blue colour.

[1] This alternative, perhaps more hack-ish, route is taken from this stack overflow tip and uses the function `text_spec` from the package `kableExtra` to apply a background color to corrections.

[2] This in turn causes a problem with markdown formatting syntax - asterisks will correctly be turned into italics in HTML output, but rendered literally with LaTeX... I will eventually correct this by instead using attributes assigned to inline code, so that the stuff to add coloring is only added *after* the markdown has been converted to its output format, see lee-phillips.org/panflute-gnuplot/.

4.3 Diving in to the OxThesis LaTeX template

For LaTeX minded people, you can read through **templates/template.tex** to understand which additional customisation options are available in there as well as in **templates/ociamthesis.cls** which supplies the base class. For example, **template.tex** provides a separate 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 these directly in **template.tex** rather than from the YAML header in **index.Rmd**.

4.4 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** by Lucy D'Agostino, which I highly recommend, the author notes how she would knit .Rmd files to a `word_document`, then use the `googledrive` R package to send this to Google Drive for comments / revisions from co-authors, then incorporate 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 on GitHub, with CriticMarkup being among the suggestions.

For now, this is still an open question in the community of R Markdown users. Knitting to a format that can easily be imported to Google Docs for comments, then going over suggested revisions and manually incorporating them back in to the .Rmd source files, has worked ok for me personally. For article writing, I sometimes upload the near-final draft to Overleaf, then collaboratively make final edits to the LaTeX file in there. I suspect some great solution will be developed in the not-to-distant future, probably by the RStudio team.

Alles Gescheite ist schon gedacht worden. Man muss nur versuchen, es noch einmal zu denken.

All intelligent thoughts have already been thought; what is necessary is only to try to think them again.

— Johann Wolfgang von Goethe [2]

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.

Appendices

A
The First Appendix

This first appendix includes the R chunks of code that were hidden throughout the document (using the `include = FALSE` chunk tag) to help with readibility and/or setup.

In the index.Rmd file

```
library(knitr)
library(kableExtra)

##### add chunk options for PDF output ####
# add option vspace_output which adds white space after each R command's output
hook_output_def = knitr::knit_hooks$get('output')
knitr::knit_hooks$set(output = function(x, options) {
  if (!is.null(options$vspace_output)) {
    end <- paste0("\\vspace{", options$vspace_output, "}")
    stringr::str_c(hook_output_def(x, options), end)
  } else {
    hook_output_def(x, options)
  }
})
```

```
# add chunk option vspace_chunk which positions entire chunk vertically
# hook_chunk = knit_hooks$get('chunk')
# knitr::knit_hooks$set(chunk = function(x, options) {
#   txt = hook_chunk(x, options)
#   # add chunk option 'vspaceout' to position chunks vertically with \vspace
#   if (!is.null(options$vspace_chunk)) {
#     latex_vspace <- paste0("\\1\\\\vspace\\{", options$vspace_chunk, "\\}")
#     txt <- sub('(\\\\begin[^}]+})', latex_vspace, txt)
#   }
#   return(txt)
# })

### function to grab all inline code
# knitr::knit_hooks$set(inline = function(x){
#       if (strsplit(as.character(x), " ")[[1]][[1]] == "mccorrect") {
#         paste0("\\mccorrect{", x, "}")
#       } else {
#         paste0("FALSE! ", x)
#       }
# })

#this creates a function to supply inline background shading of corrections
options(knitr.table.format = as.character(knitr::opts_knit$get('rmarkdown.pandoc.t
mccorrect <- function(text) {
  text_spec(text, background = ifelse(params$corrections, "#CCCCFF", "#FFFFFF"))
}
```

In Chapter 3:

```
if(!require(devtools))
  install.packages("devtools", repos = "http://cran.rstudio.com")
```

```r
if(!require(dplyr))
    install.packages("dplyr", repos = "http://cran.rstudio.com")
if(!require(ggplot2))
    install.packages("ggplot2", repos = "http://cran.rstudio.com")
if(!require(ggplot2))
    install.packages("bookdown", repos = "http://cran.rstudio.com")
if(!require(thesisdown)){
  library(devtools)
  devtools::install_github("ismayc/thesisdown")
  }
library(thesisdown)
flights <- read.csv("data/flights.csv")
```

B
The Second Appendix, for Fun

References

[1] S. T. Molina and T. D. Borkovec. “The Penn State Worry Questionnaire: Psychometric properties and associated characteristics”. In: *Worrying: Perspectives on theory, assessment and treatment.* Ed. by G. C. L. Davey and F. Tallis. New York: Wiley, 1994, pp. 265–283.

[2] Johann Wolfgang von Goethe. *Wilhelm Meisters Wanderjahre oder die Entsagenden.* de. Cotta, 1829.