

Abschlussarbeit in RStudio mit Bookdown - A Minimal Tech Example

Felix Kretschmer

2017-08-05

Contents

Vorbereitung und Vorwort	9
Get started	10
Links und Tipps	12
1 Buchbestandteile	15
1.1 Überschriften, Labels und Absätze	15
1.2 Abbildungen	15
1.3 Tabellen	16
1.4 Inline R-Code	20
2 Verfassen	23
2.1 Zitationen	23
2.2 Text wiederholen und Blockquote	24
2.3 Spracheinstellungen	24
2.4 Spell Check	24
2.5 Collaboration	25
3 To Be Done	27

List of Tables

1.1	This knitr::kable() table looks great in any output format.	17
1.2	Broken Table.	17
1.3	Pure LaTeX-Table	18
1.4	An xtable table	19
1.5	This table would be interactive in HTML.	21

List of Figures

1.1	Here is a nice figure!	16
1.2	dynamicTableWebshot	20

Vorbereitung und Vorwort

Become a more efficient scientist and collaborate more easily with colleagues!

I really much appreciate what the RStudio Team and people like Yihui Xie are doing to push reproducible research and our all efficiency working with R. And I would love if this book makes their accomplishments more accessible for students and every scientist, who is about to set up her or his next R project.

Why should I read this book?

1. *Fully dynamic reports*: You have added some new observations? Don't worry! All graphics, tables and values will have changed automatically after compiling the thesis again. You won't have to copy results from your R console and paste it to your word or presentation processing program any longer, since
 - a. R-Chunks allow you to insert tables and graphics using the R code you are familiar with.
 - b. You can use inline R code directly in your thesis text, allowing you to have R output in your sentences. Just store a result in a variable and let R and Markdown print it's value within a sentence.
2. *Be a good scientist!* You are doing real reproducible research as we all should do¹.
 - a. Having a dynamic report means every single result you report is traceable, so your reader could see, where every specific value comes from.
 - b. Everything, even the used R packages, are stored within one folder. So you can easily share the project and lastly it should compile correctly even some years in the future.
 - c. *Collaborate*: Since your are manipulating plain text files, you can use your favorite version control system (e.g. your University's GitLab) and collaborate easily with colleagues.

Contribute!

I put a lot of completely unpaid effort and time (since I'm a student hardcore procrastinating writing his thesis) into making this book and I give it to you for free, hoping it will save you a lot of time and make you a better or at least a more efficient researcher.

I though have one request for you: Everyone whom this book helped I want to invite to contribute to this project as ideas and resolutions, that would make this book continously better, come during actual thesis writing or while coding in R.

Maybe you simply want to say "Thank you!". Whether my project saved you time worth 1,000EUR (because you are of the rare species of a well paid scientist) or you just grant me an after work beer, I would really appreciate your donation!²

Visit

¹I'd really wish publisher would reward this!

²Depending on the way your are reading this, you might visit <https://kreflix.github.io/thesiswritingUsingRStudioAndBookdown/index.html#contribute> to see how to donate.

Objective

This book and it's corresponding R project shall become a great starting point for other student's theses and scientist's statistical reports. It shall contain everything you usually need while writing a thesis (especially in Psychology) out of the box, so you can use/copy this R project folder whenever you start your next R project. Though it shall still be a minimal project and it shall *not* become another bookdown or R manual. Ferner und von einer nicht zu unterschätzenden Schwierigkeit (in Details wie Dezimaltrennzeichen etc.) soll dieses R-Projekt Abschlussarbeiten auf Deutsch ermöglichen.

See To be done if you want to help, but don't know where to start.

As this book is written in order to prepare writing a thesis in German, you will be confronted with a mixture of languages.³ Feel free to translate and optimize this book, so it could become a great starting point for other student's theses.



tbd: Dieses Beispielbuch ist noch nicht voll funktionsfähig. Manches, was in HTML funktioniert, klappt noch nicht im PDF-Output. Manches davon ist eine Frage der korrekten LaTeX-Einstellungen, manches wird von bookdown und/oder pandoc nicht unterstützt. Eine Übersicht findet ihr unter To Be Done.

Get started

tl;dr:

- To open and use this book's project just download it's source files from GitHub http://statistify.de/thesiswritingUsingRStudioAndBookdown_repo or the **zip** from the toolbar on top of the book's html-version and open the **.Rproj**-file in RStudio (of course you need R and RStudio).
 - After successfully installing bookdown (see Rstudio und Packrat) you should find the button "Build Book" in the "Build"-tab in RStudio. "Build Book" creates the book in all output formats. You can find the **index.html**, **pdf** etc. in the **docs**-folder.
 - For PDF-Output you also need to install LaTeX. Pandoc assumes some LaTeX packages to be available. See <http://pandoc.org/MANUAL.html#creating-a-pdf>.
-

beta:

In theory the only thing you need to do is to download the **.zip**-folder of this RProject from my Github Repository (or clone or fork it if your familiar with GitHub) and open the **.RProject**-file. As the R packages contained in the folder need to be installed for your machine (see RStudio und Packrat), opening the RProject may take a few seconds or actually even some minutes. If you move or share your own project, it might be a good idea to delete the folder **/packrat/lib/** before starting the RProject on a new machine in order to have a clean installation of the packages.

There is also a zip-file containing the RProject downloadable via the download button on top of this book, if your reading it online. As I haven't been implementing a routine creating the latter zip automatically or figured out how to link to my GitHub repo for now, I cannot promise that it will be the current version of this project.

³Auch mein Fließtext ist noch work in progress und ich springe zwischen direkter und indirekter Anrede, vllt auch zwischen den Anreden Du und Sie.



Direct link to the RProject:

<https://github.com/kreflix/thesiswritingUsingRStudioAndBookdown/archive/master.zip>

Be patient while opening the project the first time. Also you might need an internet connection. Check R Console for what is happening.

stable:

You have to install the **bookdown**-package in R or could install the development versions of **bookdown** from Github:

```
install.packages("bookdown")
library(bookdown)
#devtools::install_github("rstudio/bookdown")
```

Remember each Rmd file contains one and only one chapter, and a chapter is defined by the first-level heading #.

To compile this example to PDF, you need to install XeLaTeX. Pandoc assumes some LaTeX packages to be available. See <http://pandoc.org/MANUAL.html#creating-a-pdf>.

I assume you have already installed R and RStudio and you know how to install any other package this RProject may need (see `_Auswertungsskript.r`).

RStudio und Packrat

Wenn du den Projektordner dieses Buches heruntergeladen hast (siehe Links und Tipps), verwendest du automatisch Packrat, da die entsprechende Projekt-Option bereits aktiviert ist. Das bedeutet, dass sich die für dieses Buch verwendeten R-Pakete in einer private library direkt im Projektordner installieren (somit hat jedes Projekt seine eigenen Pakete).

Da die Pakete auf jedem Computer neu installiert werden müssen, beinhaltet der Unterordner `/packrat` anfangs nur die Infos, welche Pakete du benötigst (einen sogenannten Snapshot). Der Ordner `/src` beinhaltet die package tarballs und sollte der Versionskontrolle unterliegen, wenn du z.B. Git verwendest, sodass das Projekt auch langfristig reproduzierbar bleibt, selbst dann, wenn die Pakete nicht mehr online zur Verfügung stehen.⁴ Die Pakete werden in den Ordner `/packrat/lib` installiert, sodass du bei jedem Umzug des Projektes auf einen neuen PC diesen Ordner zunächst löschen solltest.

Mit den folgenden Packrat-Befehlen installierst du alle benötigten Pakete auf deinem Computer (aufgrund der dependencies der Pakete ist die Liste recht lang).



Beim erstmaligen Start des Projektes wird R versuchen, Packrat und alle notwendigen Pakete auf deinem Computer zu installieren. Dies kann einige Minuten dauern und eine Internetverbindung benötigen. Beobachte währenddessen die Meldungen in der Console. Sollte es nicht automatisch klappen, dann installiere Packrat manuell. Nutze anschließend folgende Befehle, um die Pakete zu installieren oder schaue in `_Auswertungsskript.r`, welche Pakete für dieses Buch geladen werden müssen.

```
#packrat::init() #Initializes the current working directory as a Packrat project.
packrat::status()
packrat::restore()
```

⁴*td*: Packrat lädt in meinem Falle trotz hinterlegter tarballs die Pakete aus dem Internet.

```
packrat::clean() #löscht scheinbar nicht benötigte Pakete (erkennt Pakete an library() und require() )
packrat::get_opts() #aktuelle Packrat-Einstellungen
```

After successfully installing R, RStudio, all R packages and LaTeX⁵ (if you want to create PDF files), you should find the button “Build Book” in the “Build”-tab in RStudio. “Build Book” creates the book in all output formats. You can find the `index.html`, `pdf` etc. in the `docs`-folder.

Links und Tipps

- Yihui Xie wrote a great book, explaining all you need to know about bookdown. (<https://bookdown.org/yihui/bookdown/>)
 - Actually I edited his Minimal Book Example to create this book. See his original book here: <https://bookdown.org/yihui/bookdown-demo/> (or download it from GitHub).
 - See how the Bookdown-book works under the hood or to follow Yihui’s example: <https://github.com/rstudio/bookdown/tree/master/inst/examples>
- I agree with Sean Kross who says “getting started with bookdown is not yet a totally straightforward process”. Read his How to Start a Bookdown Book here!
 - Actually he provides a minimal bookdown example as well, but for some reasons his didn’t work for me. You can download it from GitHub.
- Rosanna van Hespen wrote a great and very straight forward blog on “Writing your thesis with R Markdown” (read at Rosanna’s Research) with a focus on solely creating a PDF file. Special thanks to her for eventually showing me how to use inline R code in a Markdown report. I was haunted by this idea for years and thank to her blog I came in contact with Pandoc and Co.
- Into Microsoft Word? http://rmarkdown.rstudio.com/articles_docx.html
- If you move or share your own project, it might be a good idea to delete the folder `/packrat/lib/` before starting the RProject on a new machine in order to have a clean installation of the packages.
- There are some cool addins for RStudio, e. g.
 - `citr` Just install it like any other R-package and then you can easily insert citations browsing your bib-file within Rstudio.
 - `ggThemeAssist` or `ggedit`
 - Find more <https://github.com/daattali/addinslist>
- Annotations: `hypothes.is` lets you easily and in collaboration with others (or several supervisors) annotate text on any website. Of course PDF annotations, (Online-)Word’s markups, an HTML comment area (e.g. Disqus) or Git issues could fit your needs as well.

Auswertungsskript ausführen

Damit alle Ergebnisse, Grafiken und Variablen vorhanden sind, müssen erst die Berechnungen in R vorgenommen werden. Theoretisch könnten alle R-Berechnungen auch in einem R-Chunk im Buch ausgeführt werden. Da das aber bei langen Skripten sehr unübersichtlich werden kann, habe ich ein klassisches R-Skript zur Auswertung erstellt, dass vor dem Kompilieren des Buches (ggf. mit Dummy-Daten) ausgeführt werden muss. Entweder per Hand oder per Source-Befehl.

⁵How to correctly install and set up LaTeX is another chapter, that I can’t cover by now, because it is still some sort of a mystery for me, why some things work while others don’t.

```
source('_Auswertungsskript.r')
```


Chapter 1

Buchbestandteile

In vermutlich jeder Thesis gibt es neben Text mindestens auch Abbildungen und Tabellen. Wie diese eingefügt werden, sodass am Ende immernoch jedes Output-Format korrekt dargestellt wird, schauen wir uns im Folgenden an.

This is a *sample* book written in **Markdown**. You can use anything that Pandoc's Markdown supports, e.g., a math equation $a^2 + b^2 = c^2$.

Ich versuche, Fußnoten, Links, Listen, Aufzählungen etc. nebenher beispielhaft mit einfließen zu lassen, ohne konkret darauf einzugehen.¹

1.1 Überschriften, Labels und Absätze

You can label chapter and section titles using `{#label}` after them, e.g., we can reference Chapter 1. If you do not manually label them, there will be automatic labels anyway (always in lowercase), e.g., Chapter 1.2.

Ein neuer Absatz beginnt, wenn eine Leerzeile in Markdown gelassen wurde.

Soll ein Satz in einer neuen Zeile, aber im selben Absatz beginnen, so müssen nach dem vorhergehenden Satz mehrere Leerzeichen stehen.

1.2 Abbildungen

Figures and tables with captions will be placed in `figure` and `table` environments, respectively.

```
par(mar = c(4, 4, .1, .1))
plot(pressure, type = 'b', pch = 19)
```

Reference a figure by its code chunk label with the `fig:` prefix, e.g., see Figure 1.1.

Externe Grafiken müssen über `knitr::include_graphics()` eingefügt werden, um in PDF, HTML etc. automatisch korrekt referenziert werden zu können. Fußnoten können überall per `~[Fußnotentext]` eingefügt werden – so wie hier in der Bildunterschrift.

```
knitr::include_graphics('img/important.png')
```

¹Für mehr Details sei noch einmal auf Yihui Xie's Bookdown-Buch hingewiesen!

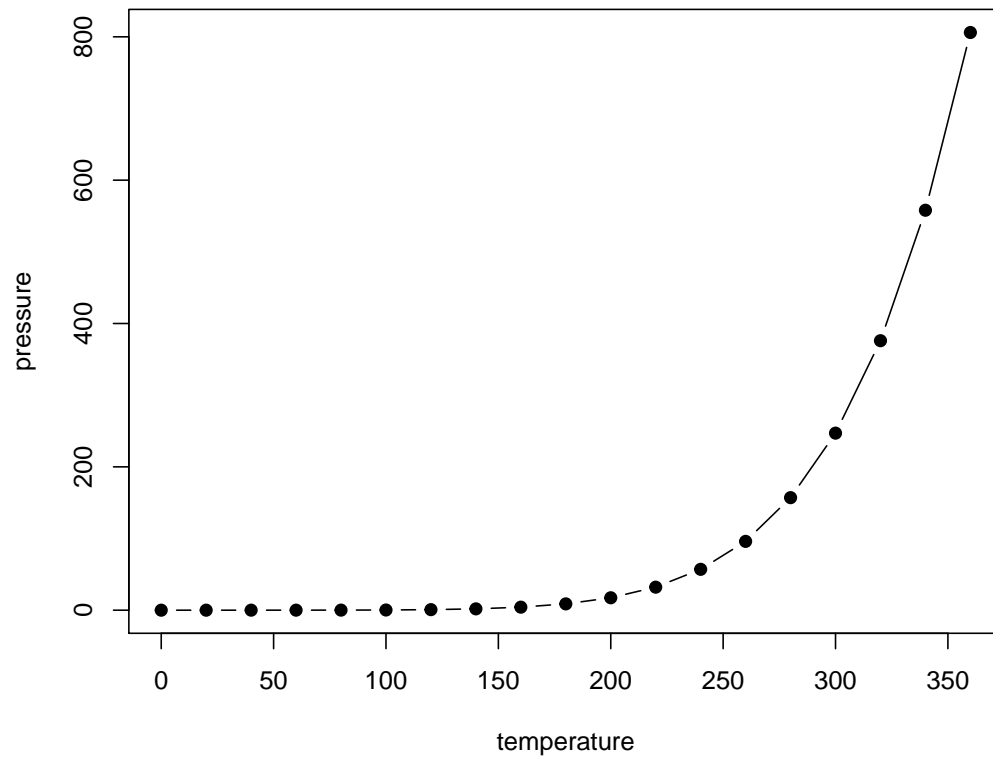


Figure 1.1: Here is a nice figure!

```
\begin{figure}
```



```
}
```

```
\caption{Zentriertes Bild aus externer PNG-Datei.2} \end{figure}
```

1.3 Tabellen

1.3.1 knitr-Tabelle

Similarly to figures, you can reference tables generated from `knitr::kable()`, e.g., see Table 1.1.

```
knitr::kable(
  head(iris, 5), caption = 'This knitr::kable() table looks great in any output format.',
  booktabs = TRUE
)
```

²<http://statistify.de>

Table 1.1: This knitr::kable() table looks great in any output format.

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
5.1	3.5	1.4	0.2	setosa
4.9	3.0	1.4	0.2	setosa
4.7	3.2	1.3	0.2	setosa
4.6	3.1	1.5	0.2	setosa
5.0	3.6	1.4	0.2	setosa

1.3.2 komplexere (LaTeX-)Tabellen

knitr::kable() erlaubt leider kein row- oder colspan, also keine Tabellenzellen, die über mehrere Reihen oder Spalten gehen. Ebenso gehen Markdown-Tabellen mit Span nicht, da dies in Pandoc flavored markdown nicht unterstützt wird.

Table: \label{tab:rmdTable} Broken Table.

```
| One      | Two | Three | Four      | Five  | Six
| -
| Span <td colspan=3>triple <td colspan=2>double
```

Ergibt eine kaputte Tabelle 1.2:

Table 1.2: Broken Table.

```

=====
One
=====

| Span
triple
double
```

Möchte man eine Tabelle ohne knitr::kable() und ohne Pandoc erstellen, kann das Buch nicht mehr automatisch in jedes Output-Format kompiliert werden. Man kann aber selbst die Bedingung schreiben, bei welchem Output-Format welches Tabellenformat gewählt werden soll.

```
if( knitr::is_latex_output() ) { ... }
```

Dies kann man nutzen, um zumindest in PDF komplexe LaTeX-Tabellen zu erzeugen, während in HTML und Co. eine nicht ganz so schöne Tabelle dargestellt wird. Beachtet die R-Chunk-Optionen results='asis' und comment='' sowie kleine Syntaxanpassungen im R-Befehl cat() wie \\ und \n, um in LaTeX kompilierbaren Output aus R heraus zu generieren.

```
{r latexTable, fig.cap='latexTable', results='asis', comment=''}
```

```
if( knitr::is_latex_output() ) {
  #erstelle LaTeX-Tabelle über R-Output
  #(beachte die Syntaxanpassung "cat()", "\\" und "\n", um LaTeX-Output aus R zu erzeugen):
  cat("\\begin{table}[]\n")
  cat("\\centering\n")
  cat("\\caption{(\n#tab:latexTable) Pure LaTeX-Table}\n")
  cat("\\label{latexTable}\n")
  cat("\\begin{tabular}{lllll}\n")
  cat("      & \\textbf{c1} & \\textbf{c2} & \\textbf{c3} & \\textbf{c4} \\\\ \n")
  cat("\\textit{r1} & \\multicolumn{2}{c}{r1c1c2} & r1c3 & r1c4 \\\\ \n")
  cat("\\textit{r2} & r2c1 & r2c2 & \\multicolumn{2}{c}{r2c3c4}\n")
  cat("\\end{tabular}\n")
  cat("\\end{table}\n")
}
```

Table 1.3: Pure LaTeX-Table

	c1	c2	c3	c4
r1	r1c1	r1c2	r1c3	r1c4
r2	r2c1	r2c2	r2c3c4	

```

# \caption{caption="(\\#tab:xTable) An xtable table")Pure LaTeX-Table}
# \label{latexTable}
# \begin{tabular}{lllll}
#           & & \textbf{c1} & \textbf{c2} & \textbf{c3} & \textbf{c4} & \\
# \textit{r1} & & \multicolumn{2}{c}{r1c1c2} & r1c3 & & r1c4 & \\
# \textit{r2} & r2c1 & & r2c2 & & \multicolumn{2}{c}{r2c3c4} & \\
# \end{tabular}
# \end{table}
} else {
  #erstelle Tabelle für alle anderen Outputformate:
  rcMatrix <- t(data.frame(c("r1c1c2", "", "r1c3", "r1c4"),
                           c("r2c1", "r2c2", "", "r2c3c4") ))
  colnames(rcMatrix) <- c("c1", "c2", "c3", "c4")
  rownames(rcMatrix) <- c("r1", "r2")

  knitr::kable(
    rcMatrix, caption = 'This table would be a pure LaTeX table with proper colspan in PDF',
    booktabs = TRUE
  )
}

```

As Yihui mentions “[if] you decide to use other R packages to generate tables, you have to make sure the label for the table environment appears in the beginning of the table caption in the form (`\#label`) (again, label must have the prefix `tab`):”

Since `\` (backslash) is an escape sequence, we have to write (`\\#tab:Beschriftung`) here.

Da der R-Chunk die Option `fig.cap='latexTable'` beinhaltet und der Tabellenüberschrifts-Latex-Befehl (`\\#tab:latexTable`) enthält, kann man in beiden Output-Bedingungen per `@ref(tab:latexTable)` auf die Tabelle referenzieren: Siehe Tabelle 1.3.

1.3.3 xTable()

Ein beliebtes R-package zur Erstellung von LaTeX-Tabellen ist `xtable`. Auch wenn Tabelle 1.4 aussieht wie jede anderen, wurde sie mit `xtable` erstellt.

`xtable` unterstützt nur die Outputformate PDF (default) und HTML (`type='html'`), die aber nicht automatisch je nach gewähltem Outputformat ausgegeben werden. So muss, wie schon in Abschnitt 1.3.2 gezeigt, für jede Output-Bedingung eine eigene Tabelle erstellt werden.

Die R-Chunk-Option `results='asis'` sowie `print.xtable(..., comment=FALSE)` sorgen dafür, dass der R-Output in LaTeX kompilierbar ist.

```
{r xTable, fig.cap='xTable', echo=TRUE, results='asis'}
```

```

if( knitr:::is_latex_output() ) {
  #PDF
  library(xtable)
  print.xtable(

```

```

    xtable(mtcars[1:3,1:4], label="xTableInternLabel", caption="(\\#tab:xTable) An xtable table"), comm
} else if ( knitr::is_html_output() ) {
  #HTML
  library(xtable)
  print.xtable(
    xtable(mtcars[1:3,1:4], label="xTableInternLabel", caption="(\\#tab:xTable) An xtable table"), comm
    type='html') #only 'latex' (default) or 'html'
} else {
  #Word und andere Outputformate
  knitr::kable(
    mtcars[1:3,1:4], caption = 'In other output formats than PDF and HTML we cannot use xtable',
    booktabs = TRUE
  )
}

```

	mpg	cyl	disp	hp
Mazda RX4	21.00	6.00	160.00	110.00
Mazda RX4 Wag	21.00	6.00	160.00	110.00
Datsun 710	22.80	4.00	108.00	93.00

Table 1.4: An xtable table



Only HTML and PDF are supported output formats in xtable().

1.3.4 interaktive Tabellen

Es können auch interaktive Tabellen eingefügt werden. Diese ergeben natürlich in statischen Dokumenten wie PDF keinen Sinn. In PDF kann aber ein **Screenshot** der dynamischen Tabelle/Abbildung automatisch eingefügt werden.

Problematisch beim Erstellen von Tabellen mit anderen Paketen ist die korrekte *Tabellennummerierung*! HTML-Widgets sind meistens Plots, weswegen Yihui vorerst auch die DT-Widgets nur als “Abbildung” bezeichnen und nummerieren lassen wird. Siehe issue 313 auf GitHub.

Um immerhin die Nummerierung als Abbild hinzubekommen, benötigt der R-Code-Chunk, der die Tabelle(ngrafik) erstellt, die Option `fig.cap`. Für einen schönen Screenshot gibt es etliche Chunk-Options via `screenshot.opts`.

```
{r dynamicTableWebshot, fig.cap='dynamicTableWebshot', dev='png', cache=TRUE, cache.extra=packageVersion}
```

```

#library(webshot)
#webshot::install_phantomjs() #muss für Screenshot installiert werden
DT::datatable(
  head(iris, 20), caption = 'This table is a screenshot in PDF, but interactive in HTML.')

```

1. See Table `\\ref{tab:dynamicTableWebshot}` (Tabellenreferenz-Syntax)
1. See Table `\\ref{fig:dynamicTableWebshot}` (Abbildungsreferenz-Syntax)
1. See Table `\\ref{dynamicTableWebshot}` (Überschriftsreferenz-Syntax)
1. See Table ?? (Tabellenreferenz-Syntax)
2. See Table 1.2 (Abbildungsreferenz-Syntax)
3. See Table ?? (Überschriftsreferenz-Syntax)

Show **10** entries

This table is a screenshot in PDF, but interactive in HTML.

Search:

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5	3.4	1.5	0.2	setosa
9	4.4	2.9	1.4	0.2	setosa
10	4.9	3.1	1.5	0.1	setosa

Showing 1 to 10 of 20 entries

Previous **1** 2 Next

Figure 1.2: dynamicTableWebshot



HTML-Widgets, müssen als **Abbildung** (`\@ref(fig:BeschriftungstextFigCap)`) referenziert werden, auch wenn es sich um eine DT-Tabelle handelt.

1.4 Inline R-Code

1.4.1 Referenzieren mit Inline-Bedingung

Kann ich nicht je nach Outputformat eine statische oder interaktive Tabelle erzeugen?

Leider kann das Vorgehen wie unter Abschnitt 1.3.2 beschrieben für DT-Tabellen (HTML-Widget) nicht empfohlen werden. Da in HTML die interaktive `htmlwidget`-Tabelle als Abbildung aufgefasst wird, aber als

Tabelle in PDF, ergeben sich unterschiedliche Nummerierungen in HTML und PDF. Zwar sind die Nummerierungen innerhalb eines Dokuments konsistent, aber man muss stets darauf achten, ob man die Tabelle mit `\@ref(tab:Tabellenbezeichnung)` (HTML) oder mit `\@ref(fig:Tabellenbezeichnung)` (PDF) referenziert.



HTML-Widgets werden auch dann als Abbildung geführt, wenn es sich um eine DT-Tabelle handelt.

Wie referenziere ich nun auf die Tabelle, wenn sie einmal eine Tabelle und einmal eine Abbildung ist? Probieren wir es anhand der folgenden Tabelle aus.

```
if( knitr::is_html_output() ) {
  #interactives HTML-Widget
```

Table 1.5: This table would be interactive in HTML.

	mpg	cyl	disp	hp
Mazda RX4	21.0	6	160	110
Mazda RX4 Wag	21.0	6	160	110
Datsun 710	22.8	4	108	93

```
#library(webshot)
#webshot::install_phantomjs() #muss für Screenshot installiert werden
DT::datatable(
  mtcars[1:3,1:4], caption = 'This table is a screenshot in PDF, but interactive in HTML.')
} else {
  #statische Tabelle in allen anderen Outputformaten
  knitr::kable(
    mtcars[1:3,1:4], caption = 'This table would be interactive in HTML.',
    booktabs = TRUE
  )
}
```

Folgende Markdown-Textschnipsel ergeben zum Teil unterschiedlichen Text in PDF und HTML.

1. See Table `\\ref{tab:htmlWidgetTableCondition}` will work in PDF, but not in HTML.
1. See Table `\\ref{fig:htmlWidgetTableCondition}` will work in HTML (and Word btw.), but not in PDF.
1. See Table `\\ref{htmlWidgetTableCondition}` is just not correct here.
1. See Table ``r ifelse (knitr:::is_html_output(), '\\\\ref{fig:htmlWidgetTableCondition}', '\\\\ref{tab:htmlWidgetTableCondition})`` will work in PDF, but not in HTML.
2. See Table ?? will work in HTML (and Word btw.), but not in PDF.
3. See Table ?? is just not correct here.
4. See Table 1.5 will work in any Output, but is a bit monstrous.

1.4.2 Inline R-Output und bedingte Textanzeige

Letztlich können wir mithilfe von Inline-R-Code auf jede R-Variable zurückgreifen und überall bedingte Textbausteine in die Thesis einfügen, was vor allem beim Report statistische Analysen sehr nützlich ist.

```
Fahrzeuge mit Automatikgetriebe haben
`r if (t.test(gear~am, data=mtcars)$p.value > 0.05) 'nicht'`
signifikant mehr oder weniger Gänge als Autos mit manuellem Schaltgetriebe.
```

Fahrzeuge mit Automatikgetriebe haben signifikant mehr oder weniger Gänge als Autos mit manuellem Schaltgetriebe.

Im Schnitt haben alle Autos des `mtcars`-Datensatzes `mean(mtcars$hp)= 146.6875PS`.



Mithilfe von Inline-R-Code ``r R-Code`` können wir auf jede R-Variable und gewohnte R-Funktionalitäten wie bedingte Textanzeigen zurückgreifen.

Chapter 2

Verfassen

Neben Überschriften, Abbildungen und Co. besteht eine Thesis nun mal hauptsächlich aus Text. Leider unterstützen in einigen Punkten andere Editoren den Schreiber etwas besser. Microsoft Word hat eine gute as-you-type Rechtschreibung- und Grammatik-Prüfung inkl. Synonymfunktion. U.a. beim Atom-Editor scheint man eine Autocompletion nicht nur für Variablen, sondern auch für Referenzschlüssel (bibtexkeys) aus der .bib-Literaturliste zu haben. Im Folgenden wollen wir uns Kleinigkeiten anschauen, die uns das Schreiben und Argumentieren in RStudio erleichtern.

2.1 Zitationen

You can write citations, too. For example, we are using the **bookdown** package (Xie, 2017) in this sample book, which was built on top of R Markdown and **knitr** (Xie, 2015).

Um die genutzten R-Pakete zu zitieren, kann wie folgt eine bib-Datei erstellt werden, die als bibliography ausgegeben werden kann:

```
# automatically create a bib database for R packages
knitr::write_bib(c(
  .packages(), 'bookdown', 'knitr', 'rmarkdown'
), 'packages.bib')
```

Mit dem RStudio Addin **citr** lässt sich, möchte man einen Autor zitieren, der passende bibtexkey per Klick herausuchen. Dabei durchsucht **citr** die im YAML-Bereich der **index.Rmd** angegebenen bibliography-files (siehe .bib-Dateien im Projektordner).

```
#install.packages("citr")
library(citr)
```

- `@xie2015` ergibt: ... (Xie, 2015) ...
- `Xie [-xie2015]` ergibt: ... Xie (2015) ...
- `@xie2015` ergibt: ... Xie (2015) ...

Der Zitationsstil übernimmt auch, wenn Autoren das erste Mal anders zitiert werden sollen (Cole et al., 2012) als beim zweiten Mal (Cole et al., 2012), was bei mehreren Autoren üblich ist.

Und man kann mehrere Autoren gleichzeitig zitieren.

[Vergleiche `@xie2015`, Kapitel 1; und auch `@Cole2012`, S. 33-35 und 41] ergibt: ... (Vergleiche Xie, 2015, Kapitel 1; und auch Cole et al., 2012, S. 33-35 und 41) ...

2.2 Text wiederholen und Blockquote

Man kann ganze Textabschnitte wiederholen. Für dieses Beispiel habe ich mich selbst in einem block quote zitiert, ohne den Text noch einmal tippen zu müssen.

Wie ich vorher schon schrieb:

```
> Zu wiederholender Text.
>
> --- me
```

Zu wiederholender Text. Zu wiederholender Text.

Wie ich vorher schon schrieb:

Zu wiederholender Text.

— me

2.3 Spracheinstellungen

In HTML werden die Tabellen korrekt nummeriert und auch die Übersetzung (“Abbildung” statt “Figure”) kann über die `_bookdown.yml` eingestellt werden. *tbd*: Der Output in LaTeX stimmt leider noch nicht.

2.4 Spell Check

RStudio nutzt die Hunspell Rechtschreibprüfung. Diese kann man pro Wort, bei dem man sich unsicher ist, oder über ein ganzes Dokument anwenden. Ich habe die Sprachdateien für Englisch und Deutsch bereits unter `/dictionaries` abgespeichert¹.

Spell-Checking whole Text-Document in English and German:

```
#install.packages('hunspell')
library(hunspell)
#dic - und aff - files aus C:\Users\Username\AppData\Local\RStudio-Desktop\dictionaries\languages-system
#hunspell_info()

deutsch <- dictionary("./dictionaries/de_DE_neu.dic")
english <- dictionary("./dictionaries/en_US.dic")

text <- readLines("03-tbd.Rmd", warn = FALSE)
bad_words_english <- hunspell(text, format = "text", dict = english)

## Warning in R_hunspell_find(dictionary, text, format, ignore):
## '.Random.seed' ist kein Integer-Vektor sondern vom Typ 'NULL', wird also
## ignoriert
```

¹für mehr Infos siehe <https://support.rstudio.com/hc/en-us/articles/200551916-Spelling-Dictionaries>


```
bad_words_deutsch <- hunspell(text, format = "text", dict = deutsch)

bad_words_list <- sort(c(unique(unlist(bad_words_deutsch)), unique(unlist(bad_words_english))))
bad_words_list <- bad_words_list[duplicated(bad_words_list)]
bad_words_list
```

```
## [1] "Ã"           "Ãobersetzung" "autocomplete"  "autocompletion"
## [5] "availabe"    "ber"          "cachen"        "citeproc"
## [9] "citr"        "deutshes"     "DT"            "eglish"
## [13] "eval"        "fÃ"           "favicon"       "github"
## [17] "htmlwidget" "https"        "hunspell"      "io"
## [21] "knitr"       "linksbÃ"      "mÃ"            "makeZip"
## [25] "nder"        "ndig"         "nftig"         "nocite"
## [29] "Packrat"     "pandoc"       "Pandoc"        "rmarkdown"
## [33] "RProject"    "rstudio"      "RStudio"       "rter"
## [37] "SeitenrÃ"    "Spellchecking" "statistify"    "Sys"
## [41] "tbd"         "vllt"         "WÃ"            "Worng"
## [45] "wrongWords" "xtable"       "yihui"         "Yihui"
## [49] "ZukÃ"
```

```
#hunspell_suggest(bad_words_list, dict = deutsch) #gibt Verbesserungsvorschläge
```

Ich persönlich finde es aber einfacher, die Vorteile von Pandoc/bookdown/knitr zu nutzen und würde die Word-Version meines Dokuments gegenlesen, wo direkt im Text unterstrichen wird, wenn ein Wort falsch oder ein Satz grammatikalisch fehlerhaft ist.

2.5 Collaboration

Since you are manipulating plain text files, you can use your favorite version control system (e.g. your University's GitLab) and collaborate easily with colleagues.

An einer Abschlussarbeit schreibt man aber in der Regel allein, möchte aber regelmäßig Verbesserungsvorschläge von BetreuerInnen oder FreundInnen bekommen. Neben den Word- oder PDF-Versionen, die herumgeschickt und von den meisten Nutzern für Annotationen verwendet werden können, sind vielleicht folgende Dienste noch interessant:

- [hypothes.is](#) lets you easily and in collaboration with others (or several supervisors) annotate text on any website.
- [Disqus](#)
- [Git issues](#)

Chapter 3

To Be Done

Notwendig:

- best practice: Wann cachieren? Und weitere sinnvolle R-Chunk-Options. <https://yihui.name/knitr/options/>
 - ich cache erstmal alles, damit knitting schnell geht
- Pandoc assumes some LaTeX packages to be available. See <http://pandoc.org/MANUAL.html#creating-a-pdf>.
- “References” statt “Literatur” unter HTML-Seite
- LaTeX Übersetzung
- Dezimaltrennzeichen vor allem von inline r-Code: Komma statt Punkt
- LaTeX Float
- LaTeX nocite
- LaTeX Code schreibt über Seitenränder -> manuell Zeilen ab Spalte 90 umbrechen
- Word zentriertes Bild ist linksbündig
- `'r format(Sys.time(), '%d. %B %Y')'` verursacht LaTeX Probleme
- custom block icons und favicon werden online nicht angezeigt
- Zitieren von Yihui. Bib-file entschlacken.

Zukünftig:

- Grafik in custom block wird auf git pages nicht angezeigt
- LaTeX Text ist zentriert nach kaputter Tabelle
- auto create .zip RProject (vllt via makeZip.bat oder zip())
- autocompletion für `\@ref`
 - Atom-Editor kann das vielleicht <https://discuss.atom.io/t/autocompletion-of-citeproc-references-in-markdown/28177> ?
- xtable bzw. generell LaTeX-Tabellen in Word und co möglich? Per Screenshot?
- citr geht nicht mehr
- mehrere Literaturverzeichnisse
- Packrat offline restore
- hunspell und Umlaute
- R-Code in Custom Block und umgekehrt
- github-Links zu statistify.de-Links wandeln/umleiten

Im Moment noch nicht möglich:

- continuous spell checking in RStudio
- autocomplete `@ref` <https://github.com/rstudio/rmarkdown/issues/958>
- htmlwidet / DT-table als “Tabelle” statt als “Abbildung”

`sessionInfo()`

```
## R version 3.4.0 (2017-04-21)
## Platform: x86_64-w64-mingw32/x64 (64-bit)
## Running under: Windows 7 x64 (build 7600)
##
## Matrix products: default
##
## locale:
## [1] LC_COLLATE=German_Germany.1252 LC_CTYPE=German_Germany.1252
## [3] LC_MONETARY=German_Germany.1252 LC_NUMERIC=C
## [5] LC_TIME=German_Germany.1252
##
## attached base packages:
## [1] stats      graphics  grDevices  utils      datasets  methods    base
##
## other attached packages:
## [1] hunspell_2.6 citr_0.2.0  xtable_1.8-2
##
## loaded via a namespace (and not attached):
## [1] Rcpp_0.12.10    bookdown_0.4    assertthat_0.2.0 digest_0.6.12
## [5] rprojroot_1.2   mime_0.5        R6_2.2.1        backports_1.0.5
## [9] magrittr_1.5    evaluate_0.10    highr_0.6        stringi_1.1.5
## [13] miniUI_0.1.1    rstudioapi_0.6   rmarkdown_1.6    tools_3.4.0
## [17] stringr_1.2.0   shiny_1.0.3      httpuv_1.3.3     yaml_2.1.14
## [21] compiler_3.4.0  htmltools_0.3.6 knitr_1.16
```

Bibliography

- Cole, G. G., Skarratt, P. A., and Billing, R.-C. (2012). Do action goals mediate social inhibition of return? *Psychological Research*, 76(6):736–746.
- Xie, Y. (2015). *Dynamic Documents with R and knitr*. Chapman and Hall/CRC, Boca Raton, Florida, 2nd edition. ISBN 978-1498716963.
- Xie, Y. (2017). *bookdown: Authoring Books and Technical Documents with R Markdown*. R package version 0.4.