

# Markdown-Demo für das IJK

Julia Niemann-Lenz

4. Januar 2021

## R Markdown

Hallo! In dieser kleinen Demo möchte ich euch die Möglichkeiten von R Markdown aufzeigen. Mit R Markdown kann man Dokumente schreiben, die sowohl den Text als auch den Code sowie statistische Analysen enthalten und die sich in viele verschiedene Output-Formate exportieren lassen (z. B. HTML-Webseiten, PDFs oder Word). So kann man z. B. aus RStudio heraus Berichte, Präsentationen oder Bücher erstellen, die jeweils aktuelle Analysen enthalten. R Markdown ist eine sehr einfache Auszeichnungssprache, die gut lesbar und auch leicht zu erlernen ist. In diesem File demonstriere ich Euch die Basics.

## Markdown-Datei anlegen

Eine neue Markdowndatei (Dateiendung .Rmd) könnt ihr in RStudio ganz einfach über das Menü anlegen:

“File” > “new File” > “R Markdown”

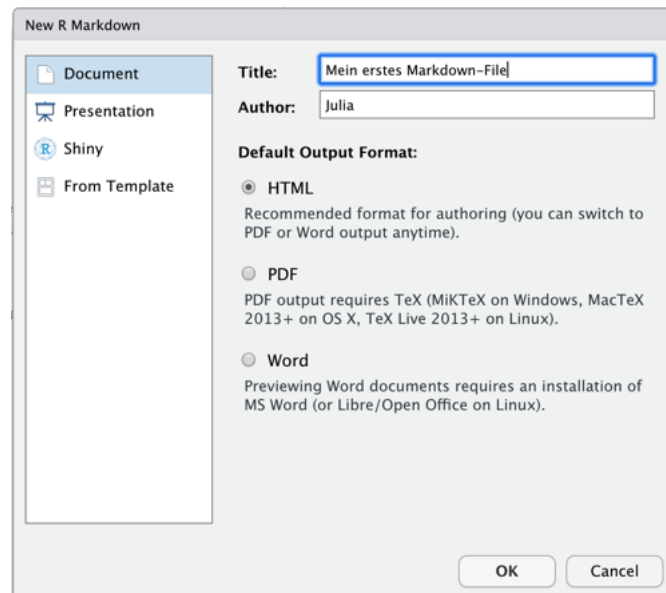


Figure 1: Screenshot “Neue R Markdown-Datei”

Im Pop-Up links könnt ihr auswählen, welches Template benutzt werden soll (z.B. “Document” für einen Bericht). Neben Berichten/Textseiten kann man auch Präsentationen mit R Markdown bauen oder interaktive Outputs mit Shiny. Dazu einfach das gewünschte anklicken, wenn ihr die Datei anlegt. Man kann selbstverständlich auch eigene Templates erstellen oder R-Pakete laden, die Templates enthalten. Aber bleiben wir erstmal bei einem einfachen, textlastigen Dokument mit R-Output.

Oben könnt ihr schon mal den Titel des Dokuments eintagen und euren Namen. Das könnt ihr auch später immer noch nachholen oder wieder ändern. Weiter unten könnt ihr auswählen, in welches Dateiformat eure

Markdowndatei standardmäßig gerendert werden soll. Zur Auswahl stehen HTML (Website), PDF und Word. Das ist nur der Standard, ihr könnt aus derselben Datei auch immer Outputs in den anderen Formaten erzeugen.

Dieses “Erzeugen” heißt bei R Markdown übrigens “Knit” oder “knitten” (vom Englischen to knit, man strickt sich als ein Dokument aus dem Markdown-File).

Klickt dazu einfach oben auf den Knit-Button in RStudio und wählt aus, welche Art Output erstellt werden soll (siehe Screenshot). Zum erstellen eines *PDFs* sind leider ein paar (einmalig durchzuführende) Vorarbeiten nötig. Euer Rechner braucht dazu “Tex”. – Das ist ein System zum Textsatz, mit dem man eben auch PDFs schreiben kann. \* Wenn ihr bereits Tex auf eurem System habt, weil ihr z.B. schon mal mit L<sup>A</sup>T<sub>E</sub>X gearbeitet habt, dann müsst ihr RStudio den Pfad zu eurer Tex-Installation mitteilen. Wenn es der Standardpfad ist, geht es möglicherweise auch automatisch. \* Wenn ihr noch kein Tex habt, kann es sein, dass RStudio die entsprechenden Pakete automatisch installiert sobald ihr auf den Button klickt (das kann dann möglicherweise ein Weilchen dauern). Sollte das nicht klappen, installiert bitte das R-Paket tinytex und versucht es nochmal. Achtung, kontrolliert bitte, ob die Installation von tinytex geklappt hat. Eventuell funktioniert es nicht beim ersten Mal.

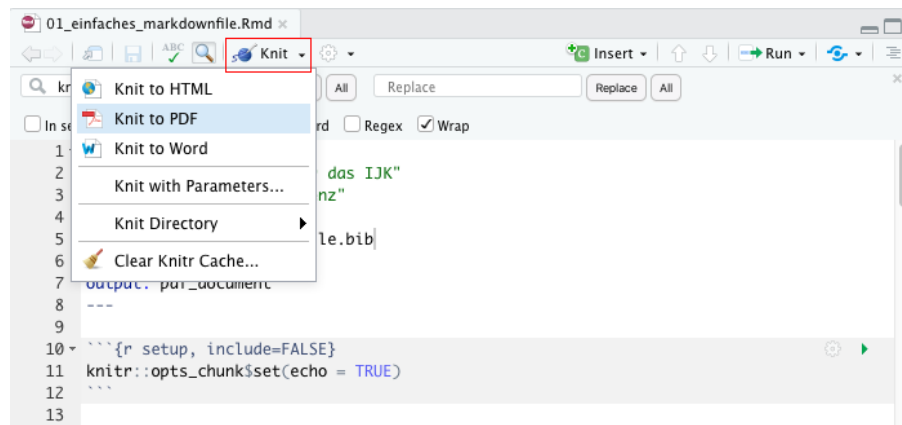


Figure 2: Screenshot “Knit-Button”

Das “Knitten” geht natürlich auch mit diesem Demo-File. Es ist super einfach, probiert es aus! Wenn alles gut geht, dauert es ein bisschen und es laufen viele Meldungen durch die Console (im Tab “R Markdown”) und der Output wird in eurem Arbeitsverzeichnis gespeichert. Der öffnet sich entweder ganz automatisch oder ihr müsst im Ordner drauf klicken.

## Basic Markdown-Formatierungen

Mit Markdown kann man natürlich alle möglichen Formatierungen im Text machen. Die Sprache ist sehr leicht zu lernen, denn einfache Schreib- und Lesbarkeit ist ihre Philosophie (mehr dazu hier).

Man kann **fetten** und *kursiv* stellen, Text <sup>hoch</sup> stellen oder Formeln einfügen  $A = \pi * r^2$  usw.

Man kann Text hervorheben, indem man ihn in einem Blockquote einrückt.

### Hier eine Überschrift kleinerer Ebene

#### Hier ein paar Listen

- Man kann
  - ungeordnete
  - Listen erstellen.

1. Oder auch

1. geordnete
2. Listen,
2. wie diese hier.

Für einen neuen Absatz muss man übrigens am Ende der Zeile zwei Leerzeichen machen. Oder man macht einfach eine Leerzeile im Markdownfile. Im Output wird dies dann als neuer Absatz formatiert.

Selbstverständlich kann man auch Links einfügen, entweder so <http://rmarkdown.rstudio.com> oder so [Link zum Reference Guide](#). Im ersten Link, bei RStudio findet ihr übrigens eine gute Einführung in R Markdown, die natürlich viel umfangreicher ist als diese Demo hier. Im zweiten Link ist eine Referenz, in der ihr noch viel mehr Formatierungsmöglichkeiten findet.

## R Code in R

Man kann im Markdownfile R-Code schreiben, der ausgeführt wird, wenn das Markdownfile geknitted wird. Das macht man so:

```
summary(cars)
```

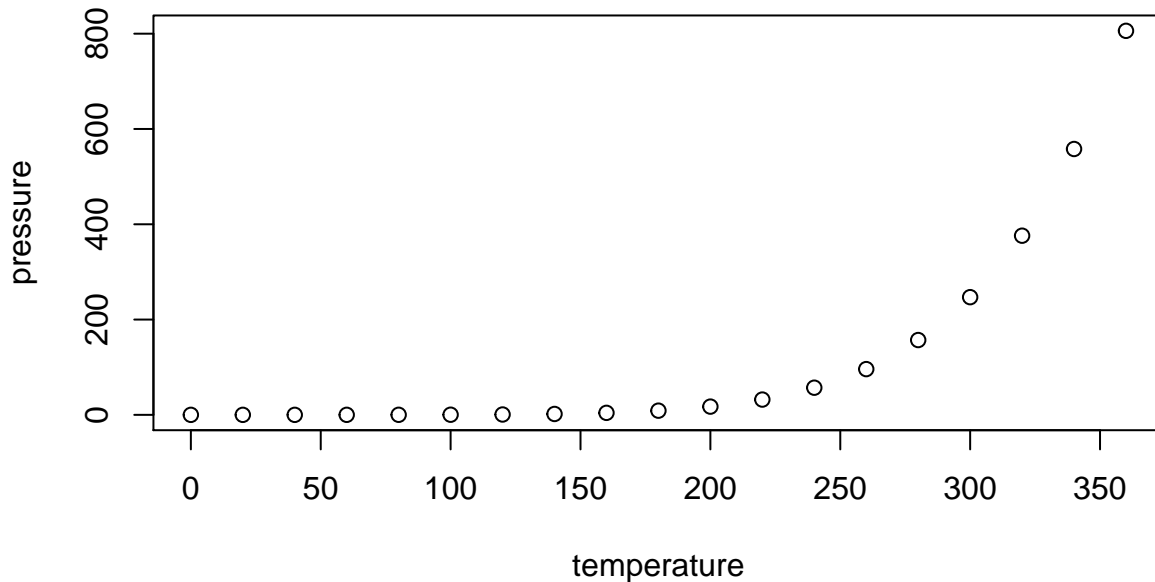
```
##      speed      dist
## Min.   : 4.0    Min.   :  2.00
## 1st Qu.:12.0    1st Qu.: 26.00
## Median :15.0    Median : 36.00
## Mean   :15.4    Mean   : 42.98
## 3rd Qu.:19.0    3rd Qu.: 56.00
## Max.   :25.0    Max.   :120.00
```

Ok, dieser Output ist jetzt nicht sooo hübsch, aber natürlich, man kann auch viel längeren, komplexeren und tolleren R-Code einfügen, so dass es richtig schön wird. Ist ja nur eine Demo hier.

Und auch in den Zeilen kann man auf Inhalte aus R zugreifen bzw. diese auch dynamisch erzeugen. Hier lasse ich bspw. einen Mittelwert berechnen:  $M = 15.4$ .

## Grafiken einfügen

Selbstredend kann man mittels R-Code auch Grafiken direkt in den Report einbauen:



Der Befehl `echo = FALSE` in dieser Zelle sorgt übrigens dafür, dass der R-Code nicht mit geknittet wird, sondern nur die Grafik. Das geht selbstverständlich auch bei allen anderen Code-Schnipseln.

## Tabellen

Per Default-Einstellung gibt R MArkdown Tabellen so aus, wie sie in der Console dargestellt würden, also so:

```
head(mtcars, 5)
```

```
##           mpg  cyl  disp  hp  drat    wt   qsec  vs  am  gear  carb
## Mazda RX4      21.0   6  160  110 3.90 2.620 16.46  0   1    4    4
## Mazda RX4 Wag  21.0   6  160  110 3.90 2.875 17.02  0   1    4    4
## Datsun 710     22.8   4  108   93 3.85 2.320 18.61  1   1    4    1
## Hornet 4 Drive  21.4   6  258  110 3.08 3.215 19.44  1   0    3    1
## Hornet Sportabout 18.7   8  360  175 3.15 3.440 17.02  0   0    3    2
```

Für eine schönere Formatierung der Tabellen gibt es viele Lösungen, ein guter Einstieg ist das paket `knitr`, das man wie folgt einsetzt:

```
knitr::kable(
  head(mtcars, 5),
  caption = "Tabelle mit knitr"
)
```

Table 1: Tabelle mit knitr

	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2

## Zitieren

In der Gallery bei RStudio <http://rmarkdown.rstudio.com> findet ihr ganz viele Beispiele für tolle Markdown-Outputs. Darunter sind auch interaktive Dashboards und sogar ganze Bücher.

Appropos Bücher. Natürlich gibt es gaaaanz viel Literatur zum Thema. Neben dem bereits bekannten R4DS von Wickham und Golemund (2017, Kapitel 27) möchte ich hier insbesondere auf “R Markdown: The definitive guide” und – ganz neu – das “R Markdown Cookbook” hinweisen (Xie et al. 2018; Xie und Dervieux 2020). Aber die bereits mehrfach erwähnte Website von RStudio liefert auch einen umfangreichen Einstieg sowie ein praktisches Cheat Sheet.

Damit die Datei auch ein Literaturverzeichnis hat, muss man auf eine BibTeX-Datei verweisen<sup>1</sup>. BibTeX ist ein standardisiertes Format für den Austausch von Literaturquellen. So eine Datei könnt ihr zum Beispiel mit Citavi erzeugen (über Datei > Exportieren). Legt die Datei einfach im Arbeitsverzeichnis von R ab. Ganz oben im Markdown-Dokument (im “YAML Header”) macht ihr dann den Verweis auf die Datei. Habe ich in diesem Dokument auch gemacht (`bibliography: my_bibtexfile.bib`). Den Namen müsst ihr natürlich anpassen. In der Zeile danach habe ich den Zitierstil umgestellt auf den der Zeitschrift Publizistik (`cs1: publizistik.csl`). Solche csl-Dateien, die den Zitationsstil formatieren, könnt ihr bspw. ganz einfach über <https://www.zotero.org/styles> recherchieren. Das Literaturverzeichnis wird automatisch beim knitten am Ende des Outputs erzeugt.

## Ausblick

Natürlich kann man R Markdown auch so anpassen, dass Text und Literaturverzeichnis im APA-Stil ausgegeben werden (Demo folgt). Es gibt sogar ein eigenes R-Paket, dass sich damit beschäftigt. Man kann den *Stil* eines Dokumentes sehr einfach wechseln und viele Journals bieten sogar Vorlagen an, mit deren Hilfe das Umformatieren auf die spezifischen Satz-Anforderungen und Zitierweisen eines Journals einfach wird.

R Markdown ist mit L<sup>A</sup>T<sub>E</sub>X verzahnt und deshalb kann man die L<sup>A</sup>T<sub>E</sub>X-Befehle nutzen z.B. so: *kursiv*. Damit sind dann noch viel mehr Formatierungen, z. B. bei mathematischen Formeln und sogar professioneller Satz möglich (z. B. automatische Ligaturen, mein Typografie-Herz schlägt höher!). - Damit sehen eure Berichte dann sogar toller aus als mit MS Word... Man kann auch Markdown und L<sup>A</sup>T<sub>E</sub>Xim gleichen Dokument benutzen, wie hier geschehen.

Und selbstverständlich kann man mit R Markdown Präsentationen, Webseiten oder ganze (interaktive) Bücher schreiben (bookdown). Verschränkt mit weiteren Technologien werden auch interaktive Inhalte in den Dateien möglich (z.B. mit jupyter notebooks, shiny apps).

## Literaturverzeichnis

Wickham, H., & Golemund, G. (2017). *R for Data Science: Import, tidy, transform, visualize, and model data*. Beijing; Boston; Farnham; Sebastopol; Tokyo: O'Reilly. <https://r4ds.had.co.nz/r-markdown.html>

Xie, Y., Allaire, J. J., & Golemund, G. (2018). *R Markdown: The definitive guide*. Boca Raton; London; New York: CRC Press Taylor & Francis Group a Chapman & Hall book. <https://bookdown.org/yihui/rmarkdown/>

Xie, Y., & Dervieux, C. (2020). *R Markdown Cookbook*. <https://bookdown.org/yihui/rmarkdown-cookbook/>

---

<sup>1</sup>Ok, es geht auch anders. Aber das hier ist ein prima Weg.