

Statistik in der Archäologie:
eine anwendungsorientierte Einführung
auf Basis freier Software

Frank Siegmund

Update: Korrekturen und Ergänzungen zu Vers. 1.0 (Febr. 2020)

Stand: August 2020

**Statistik in der Archäologie:
eine anwendungsorientierte Einführung auf Basis freier Software**

von Frank Siegmund (Münster)

ORCID <https://orcid.org/0000-0002-0555-3451>

Veröffentlicht mit der Lizenzierung CC BY

Eine PrePrint-Fassung sowie die im Text angesprochenen Skripte, Beispiel- und Übungsdateien finden sich auf der Website des Verfassers:

www.frank-siegmund.de

Umschlaggestaltung: Michaela Fischer.

Text, Lektorat, Abbildungen, Layout und Satz: Frank Siegmund.

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über dnb.dnb.de abrufbar.

© Frank Siegmund 2020, veröffentlicht unter der Lizenz CC BY

Herstellung und Verlag: BoD - Books on Demand GmbH, Norderstedt

2020

ISBN 978-3-752 825-70-1

Inhaltsverzeichnis

1	Corrigenda	4
2	Aktualisierungen	4
2.1	zu Kap. 8.5.....	4
2.2	zu Kap. 14.3: R++ / Rplusplus	4
2.3	zu Kap. 15.1 GUIs zu R	5
2.4	zu Kap. 15.3.: Fehlermeldung „RTools“ bei Installation von Paketen	6
2.5	zu Kap. 15.8: BlueSky Statistics	6
2.6	zu Kap. 23.2 Korrelation	7
3	Ergänzungen	7
3.1	Ergänzung zu Kap. 15.7.5: Den R-Commander „aufbohren“	7
4	Markdown und RMarkdown.....	8
4.1	Was ist und wozu dient „Markdown“	8
4.2	Hinweise zu Bookdown, knitr und pandoc.....	9
4.3	Zielsetzung und Anwendung von RMarkdown im Kontext Archäo-Statistik	10
4.4	RMarkdown im RCommander	10
4.5	Kurzanleitung RMarkdown mit RStudio.....	11
4.6	Konventionen in den diesem Werk beigegebenen Markdown-Dateien.....	12
5	Literatur	13

1 CORRIGENDA

2 AKTUALISIERUNGEN

2.1 ZU KAP. 8.5

[*am Ende des Abschnitts*] Wer schon jetzt überzeugt ist, mit dem Profi-Werkzeug **R** arbeiten zu wollen, kann im Folgenden die Kap. 9, 12 und 14 schnell überspringen und mit Kap. 15 ernsthaft einsteigen.

2.2 ZU KAP. 14.3: R++ / RPLUSPLUS

[*vorläufig*] Im Juli 2018 tauchten bei YouTube Ankündigungen von R++ auf, einer grafischen Benutzeroberfläche für R, die vor allem auf Geschwindigkeit und das Arbeiten mit vielen Automatismen zielt: kurze, gut gemachte Erklärvideos wahlweise in französischer und englischer Sprache, die neugierig machten. Mitte Juni 2020 war das Vorhaben dann verfügbar, wenn auch etwas anders als erwartet: Die bisherige Doppelspurigkeit der Website in französischer und englischer Sprache wurde zu Gunsten einer rein französischen Fassung aufgegeben, und auch die Benutzerführung von R++ ist rein französisch. Aus Benutzersicht wesentlicher ist jedoch m.E. die Unklarheit hinsichtlich der Lizenzierung: Die ersten Ankündigungen von R++ hinsichtlich der Lizenzierung konnte man so verstehen, dass es wie z. B. bei RStudio oder BlueSky gedacht sei: Open Access für Alle, doch wer Beratung oder Hosting benötigt, zahlt. Beim Start von R++ im Juni 2020 gab es indes eine Gratisversion nur für Studierende und Universitätsangehörige, zeitlich begrenzt auf 12 Monate, alle übrigen zahlen eine erkleckliche jährliche Lizenzgebühr. Inwieweit der auf Studierende und Universitätsangehörige begrenzte Open Access nach einem Jahr weitergeführt wird, ist unklar.

Trotz dieser Einschränkung scheint ein Test von R++ attraktiv: Nach Anmeldung über eine Uni-Emailadresse erhält man nach ca. 1-2 Arbeitstagen eine Lizenznummer. Das Herunterladen und Installieren verläuft problemlos. Ein Eintippen dieser Lizenznummer ist nicht möglich, sondern „nur“ ein Übertragen mit Copy & Paste – wenn man's weiß, sehr bequem. Der erste Programmstart verläuft seeehr langsam. Danach erlebt man – wie in den Videos angekündigt – eine frisch und bunt daherkommende, fröhlich machende Oberfläche und ein problemloses Einlesen externer Datensätze

(viele Formate, auch ODBC und MySQL möglich). Was folgt, darf man statt „einarbeiten“ auch „einspielen“ nennen, so leicht ist die Bedienung der grundlegenden Optionen. R++ macht Spaß, meist ist die Bedienung noch simpler, als man es erwartet. Nach ca. 45 Minuten ist erlernt: Daten einlesen, Daten sortieren und filtern, die primären Statistiken, Standardgrafiken sowie grundlegende statistische Tests. Dazu gibt es lineare Regression und ANOVA. Hinter dem „Éditeur graphique“ verbirgt sich eine schnell und wirklich sehr gute funktionierende Oberfläche zum Anpassen der Standardgraphiken: Schrift, Beschriftung, Farben, Skalierung etc. lassen sich gut steuern, zudem ist direktes Hineinschreiben und -zeichnen in die Graphik möglich: sehr angenehm! Vieles ist auch didaktisch sehr anschaulich, wie z. B. die Auswahl der verschiedenen Kerne und der Kernweite bei der Schätzung der empirischen Dichtefunktion. Der grundlegende Graphik-Stil von R++ ist ein schick gemachtes R Base, nicht *ggplot2*.

Aber: Das Einspielen der dann später angebotenen Updates hakelte beträchtlich, zwischenzeitlich war das Programm nicht mehr lauffähig. Vor allem aber ist der derzeit (August 2020) mitgegebene Funktionsumfang noch sehr schlank, multivariate Verfahren fehlen gänzlich, d. h. bei ernsthaftem Arbeiten wird man schnell mehr Funktionen / Verfahren brauchen als derzeit verfügbar. Wer also aktuell **R** mit graphischer Benutzeroberfläche braucht, kommt inhaltlich z. B. mit einem gezielt durch Plugins erweiterten R-Commander oder dem im Vergleich zu R++ etwas altbacken aussehenden, aber leistungsfähigen BlueSky (viel) weiter. Daher: Projekt R++ unbedingt im Auge behalten, doch für ernsthaftes Arbeiten derzeit lieber andere Wege verfolgen.

R++ bei YouTube: <https://www.youtube.com/channel/UC3D79-iSkciduLFKmZAugXw/videos> [1.7.2020].

Website: <https://rplusplus.com/> [1.7.2020].

2.3 ZU KAP. 15.1 GUIs ZU R

[*am Ende des Kapitels*] Aus meinen Einführungskursen für Anfänger ergibt sich folgende Rückmeldung: Das Arbeiten sogleich mit R & RStudio ohne GUI wird als mühsam empfunden und wird nicht sonderlich geschätzt. Die überwiegende Mehrheit der Kursteilnehmer konnte sich schnell mit dem R-Commander anfreunden. Im Vergleich dazu fiel bei JAMOVI die Einfachheit und Eleganz auf wie auch die angebotenen Erweiterungsmöglichkeiten, doch fehlt es (Stand Mitte 2020) sehr bald an Möglichkeiten der händischen Nachsteuerung, insbes. bei den Grafiken. Bei BlueSky-Statistics erwies

sich das erste Einarbeiten als aufwändiger, doch das eigentliche Problem war die überwältigende Fülle des Angebots an Optionen – eine Fülle, die Anfänger ob all der fremden Begriffe eher abschreckt als inspiriert. Nicht zuletzt wurde es als schwierig empfunden, gezielt Gesuchtes in den vielfach gestaffelten Menüs zu finden. BlueSky „mag“ deutsche Umlaute in den Variablenamen nicht, die resultierenden Fehlermeldungen waren aber für Anfänger nicht hinreichend verständlich, so dass sie dies als das ausschlaggebende Problem der Fehlermeldungen nicht erkannt haben. Andererseits schätzten Anfänger die Ausgabe von R-Code durch BlueSky und die Option, dann den per Point & Click erzeugten Code gezielt ergänzen zu können. Daher ist BlueSky wohl eher für Nutzer geeignet, die bereits mit den Grundlagen der Statistik vertraut sind und Bedürfnisse haben, die über die ersten Grundlagen hinausgehen.

2.4 ZU KAP. 15.3.: FEHLERMELDUNG „RTOOLS“ BEI INSTALLATION VON PAKETEN

Bei der Installation von Paketen liest man gelegentlich die Meldung:

```
WARNING: Rtools is required to build R packages, but is not currently installed.
```

Eine Warnung ist keine Fehlermeldung, kann man darüber hinweggehen, und was hat es damit auf sich? Die meisten R-Pakete werden gebrauchsfertig ausgeliefert, als sog. Binaries. Bei ihrer Installation begegnet die o.g. Warnmeldung nicht. Manchmal aber müssen die Pakete bei der Installation noch kompiliert werden, und in diesem Fall wird das Vorhandensein der RTools erwartet. Fehlt RTools, werden diese Pakete nicht kompiliert, d.h. sie sind nicht ausführbar. Lösung? RTools installieren! Am besten als zweiter Schritt, gleich nachdem man R installiert hat. Zu R 4.0.0 (ff.) gehört Rtools40, man findet es auf CRAN: <https://cran.r-project.org/bin/windows/Rtools/> [1.7.2020]. Das Paket wie gewohnt herunterladen und mit dem Windows-Installer wie übliche Programme installieren und dabei den Default-Ordnerpfad am besten nicht verändern. Problem gelöst. Für ein gutes Zusammenspiel von R, RTools und RStudio darauf achten, dass man jeweils die neuesten, zueinander passenden Versionen installiert. Wer begründet mit älteren Versionen von R oder RStudio arbeitet, sollte auch die passende ältere Version von RTools installieren – die ebenfalls über CRAN erreichbar ist.

2.5 ZU KAP. 15.8: BLUESKY STATISTICS

Zu dieser Benutzeroberfläche ist Anfang August 2020 ein gutes Handbuch in englischer Sprache erschienen, das (z.Zt.?) als kostenloses PDF angeboten wird:

Muenchen, Robert A. (2020). *BlueSky Statistics User's Guide*. 2020 Edition, First Draft. <https://www.blueskystatistics.com/v/vspfiles/downloadables/BlueSky-Users-Guide.pdf> [8.8.2020].

Eine Information, die nur für fortgeschrittene Nutzer relevant ist: BlueSky bietet zahlreiche “Extensions“ an, d.h. zusätzliche Module zu speziellen statistischen Verfahren. Nutzer der Open-Source-Version (wie z.B. der Autor dieser Zeilen) können bis zu fünf dieser Extensions kostenlos erhalten und nutzen: <https://www.blueskystatistics.com/category-s/122.htm>

2.6 ZU KAP. 23.2 KORRELATION

Frage: Muss ich wie beim Test auf Mittelwertunterschiede zwischen abhängigen und unabhängigen Stichproben unterscheiden? Antwort: nein.

3 ERGÄNZUNGEN

3.1 ERGÄNZUNG ZU KAP. 15.7.5: DEN R-COMMANDER „AUFBOHREN“

Für die hier bis einschließlich Kap. 24 behandelten Themen reichen die Fähigkeiten der Basis-Installation der R-Commanders im Grunde aus. Mit vier Plugins lässt er sich jedoch zu einem viel weiterführenden, mächtigen Werkzeug ausbauen: „KMggplot2“ erweitert den R-Commander um die Fähigkeit, Grafiken mit *ggplot2* anzufertigen. Das Plugin „EZR“ (vgl. Kap. 14.3.2) fügt vor allem Verfahren zur Versuchsplanung und zum Arbeiten mit nominalskalierten Variablen hinzu. Das Plugin „FactoMineR“ erweitert den R-Commander im Bereich der multivariablen Verfahren – die in dieser Einführung nicht behandelt werden – sehr weitreichend, insbes. zum Themenfeld PCA / Hauptkomponentenanalyse, Faktorenanalyse und Clusteranalyse. Die dort ebenfalls angebotene Korrespondenzanalyse ist nach Einschätzung des Autors „speziell“. Das Plugin „biostat“ bündelt die drei genannten Plugins wiederum in eine gut funktionierende gemeinsame Oberfläche und fügt dem R-Commander weitere und vor allem bequeme Optionen der Datenverwaltung hinzu.

Die drei erstgenannten Plugins erreicht man über CRAN. Entweder man geht den Weg „Paket herunterladen“ und fährt dann vom R-Commander aus mit „Plugin installieren“ fort, oder man geht den m.E. einfacheren Weg über RStudio, um diese Installation vorzunehmen. Man kann die Installation auch gleich mit dem Plugin „biostat“ beginnen, das dann die gen. drei Plugins nach sich zieht, aber der Weg, zuerst die drei gen. Plugins einzeln zu installieren, und dann *biostat* zu installieren, scheint mir stabiler. Das Paket *biostat* und eine Beschreibung finden sich auf GitHub: <https://geg-znav.github.io/RcmdrPlugin.biostat/> [4.7.2020]. Auch wenn die dort gegebene Beschreibung kompliziert klingen mag, die Installation mit Hilfe von RStudio verläuft nach meinen Erfahrungen problemlos; sie ist aber nicht schnell, da mehrere Komponenten nicht als Binaries vorliegen, sondern kompiliert werden müssen (siehe oben, „RTools“).

4 MARKDOWN UND RMARKDOWN

4.1 WAS IST UND WOZU DIENT „MARKDOWN“

„Markdown“ ist eine einfache Auszeichnungssprache, die für ein anderes Schreiben von Texten steht, als wir es seit dem Aufkommen der PCs in den 1980er Jahren samt der seitdem üblichen Textverarbeitungen wie z.B. MS-Word, LO-Writer oder WordPerfect gewohnt sind. Bei den modernen Textverarbeitungen gilt WYSIWYG – *what you see is what you get*: Die Texte werden während des Schreibens so angezeigt, wie sie später auch im Ausdruck oder als PDF sichtbar sind. Auszeichnungssprachen hingegen trennen das Schreiben vom Formatieren und Setzen. Während des Schreibens werden lediglich Auszeichnungen festgelegt wie z.B. „dies bitte in Fettschrift“, „dies ist eine Überschrift zweiten Grades“ usw. Praktisch umgesetzt werden diese Auszeichnungen erst nach dem Schreiben, wenn aus dem mit Auszeichnungen erfassten Text ein gesetzter Text erzeugt wird. Eine sehr bekannte und weitverbreitete Auszeichnungssprache ist das in den 1980er Jahren entwickelte LaTeX (gesprochen: latech). Aber auch HTML ist eine Auszeichnungssprache, der man z.B. begegnet, wenn man einen Wikipedia-Artikel editiert oder zumindest einmal im Editiermodus liest. Die Auszeichnungssprache „Markdown“ wurde entwickelt, um im Gegensatz zu LaTeX oder HTML mit sehr wenigen, d.h. nur den wichtigsten Auszeichnungen auskommen zu können bei gleichzeitig hoher Lesbarkeit der rohen Texte, d.h. möglichst wenig Störungen des Lesens durch eben diese Auszeichnungen. Beim Schreiben von Texten in Markdown benutzt man einfache Editoren wie z.B. den Windows Editor oder –

weitaus leistungsfähiger – „NotePad++“, oder speziell auf Markdown ausgerichtete Editoren wie z.B. „Typora“ oder „PanWriter“. Der Autor schätzt Typora sehr.

Im Kontext des Themas Archäo-Statistik wird Markdown relevant durch RMarkdown, die bei RStudio eingesetzte R-Variante von Markdown. RMarkdown ermöglicht es, wie mit Markdown einfache Texte zu schreiben, zusätzlich aber ausführbaren R Code in diese einzubetten, d.h. eine enge Verbindung von Text und Code. RMarkdown wird daher zum idealen Schreibwerkzeug für Texte, die viel Code enthalten, wie z.B. Anleitungen zu R, Datenverarbeitung und Statistik mit R usw. Nach dem Schreiben kann aus RStudio heraus aus diesen Texten u.a. ein HTML-Dokument erzeugt werden (also eine Website), aber auch ein sorgsam gesetztes PDF oder ein MS-Word-File. Für Weiteres verweise ich auf die nachfolgend angeführte Literatur.

Bates, C. (9.7.2020). Getting started with R Markdown - Guide and cheatsheet. *R-Bloggers*, 9.7.2020: <https://www.r-bloggers.com/getting-started-with-r-markdown-guide-and-cheat-sheet/> [12.7.2020].

Tierney, Nicholas (2020). RMarkdown for Scientists. <https://rmd4sci.njtierney.com> [13.8.2020].

Xie, Yihui, Allaire, J. & Grolemond, Garrett (2019). R Markdown: The definitive Guide. CRC-Press. www.bookdown.org [16.2.2020].

Wickham, H. & Grolemond, G. (2018). R für Data Science: Daten importieren, bereinigen, umformen, modellieren und visualisieren. (übers. v. Frank Langenau). (p. 399-414). Heidelberg: dpunkt.verlag.

4.2 HINWEISE ZU BOOKDOWN, KNITR UND PANDOC

Mit Hilfe der R Pakete *knitr* und *bookdown* sowie dem Tool *pandoc* lassen sich ganze Bücher in Markdown oder RMarkdown schreiben – doch dies näher zu erläutern, ist hier nicht das Ziel. Dennoch einige Literaturhinweise auch dazu.

Lappan, Rachel (9.10.2017). Using Bookdown for tidy documentation: <https://rachaellappan.github.io/bookdown/> [12.7.2020].

Kross, Jean (17.11.2016). How to Start a Bookdown Book. <http://seankross.com/2016/11/17/How-to-Start-a-Bookdown-Book.html> [12.7.2020].

Marwick, B. (2018). rrttools: Tools for writing reproducible research in R. GitHub. <https://github.com/benmarwick/rrtools> [12.7.2020]. – Dazu ergänzend: Schmid, Cl. (2018). rrttools.addin. GitHub. <https://github.com/nevrome/rrtools.addin>

Xie, Yihui (2020). bookdown: Authoring Books and Technical Documents with R Markdown. <https://bookdown.org/yihui/bookdown/> [12.7.2020].

Xie, Yihui (2020). knitr: elegant, flexible, and fast dynamic report generation with R. <https://yihui.org/knitr/options/> [12.7.2020].

4.3 ZIELSETZUNG UND ANWENDUNG VON RMARKDOWN IM KONTEXT ARCHÄO-STATISTIK

Die Einführung von RMarkdown an dieser Stelle ist keine Aufforderung, sich beim Schreiben seiner Werke von den gewohnten Textverarbeitungen zu lösen und zu einer Auszeichnungssprache wie z.B. RMarkdown zu wechseln – diese Abklärung und Entscheidung sollte Jeder für sich treffen. Aber als Notizzettel für den eigenen, kommentierten Code während des Arbeitens mit R & RStudio sowie vor allem für die Dokumentation des eigenen Tuns für Dritte ist RMarkdown sehr gut geeignet – weshalb seine Verwendung zu diesem Zweck ausdrücklich empfohlen sei. Daher im Folgenden eine rudimentäre Anleitung und ein paar kurze praktische Hinweise zum Arbeiten mit RMarkdown.

Eine knappe Übersicht und Kurzfassung bietet das R Markdown::cheat sheet: <https://raw.githubusercontent.com/rstudio/cheatsheets/master/rmarkdown-2.0.pdf> [12.7.2020]. Die ausführliche Dokumentation (5 Seiten) findet sich hier: R Markdown Reference Guide: <https://rstudio.com/wp-content/uploads/2015/03/rmarkdown-reference.pdf> [12.7.2020].

R Markdown Tutorial from RStudio: <https://rmarkdown.rstudio.com/lesson-1.html> [12.7.2020].

4.4 RMARKDOWN IM RCOMMANDER

RStudio und RMarkdown sind eng verbunden. Da RStudio auf die Codierung UTF-8 gesetzt werden kann, macht es in RStudio auch keine Probleme, deutschsprachige Texte mit deutschen Umlauten und „ß“ zu schreiben. Auch der R-Commander beinhaltet die Option, Texte in RMarkdown zu schreiben resp. sein Arbeiten per RMarkdown zu protokollieren und kommentieren. Nur eben mit etwas weniger Komfort: Der R-Commander verweigert bei deutschen Umlauten und „ß“ seine Mitwirkung, und auch das Schreiben ist wg. der Zeilenumbrüche, die der R-Commander nicht macht, etwas gewöhnungsbedürftig. Kurz: das Schreiben in RMarkdown ist im R-Commander möglich, aber mit praktischen Einschätzungen. Die Dateien mit Beispiel-Code, die diesem Buch beigegeben sind, wurden – weil das Buch auf den R-Commander setzt – konsequenterweise mit dem R-Commander geschrieben und sind mit ihm gut lesbare und editierbar.

4.5 KURZANLEITUNG RMARKDOWN MIT RSTUDIO

Einen (neuen) Text in RMarkdown legt man wie folgt an: in RStudio => File; => New File; => RMarkdown; in das aufklappende Fenster den Titel des neuen Dokuments eintragen und den Autor. Als „Default Output Format“ ist gesetzt „HTML“ – was man zunächst so belassen sollte, doch zeigen die Optionen „PDF“ und „Word“ an, das ohne weiteren Aufwand auch PDF- und MS-Word-Dokumente ausgegeben werden können. Nach OK legt RStudio das bestellte Dokument an und zeigt es im linken oberen Fenster von RStudio an. Man sieht ganz oben, eingekapselt zwischen zwei Zeilen mit „- -“, den Kopf des Textes (sog. YAML-Header), darunter einen kurzen Beispieltext, den man löschen kann.

Wenn man nach dem Schreiben der ersten eigenen Textzeilen in RStudio (oben) auf das Symbol „Knit“ klickt, wird aus dem Geschriebenen ein HTML-Dokument erzeugt, das sowohl den geschriebenen Text als auch den R-Code enthält. In den (im fertigen HTML- oder docx-Dokument nicht mehr sichtbaren) Anweisungen ganz oben findet sich die Zeile "output: html_document", womit als Ausgabe nach dem „Knitten“ ein HTML-Dokument bestellt wird. Wählt man statt dessen „output: word_document“, erzeugt Knit eine .docx-Datei.

R-Code wird als „chunk“ wie folgt mit drei Schrägstrichen in den Text eingebunden, wobei die von R ausgegebenen Ergebnisse beim „Knitten“ gleich in das Dokument eingebunden werden (können):

```
```\n\nhier steht ausführbarer R Code\n\n```
```

Nun kurz zu den Textauszeichnungen in RMarkdown wie auch Markdown. Überschriften werden mit „#“ markiert und zwar auch in Hierarchie-Ebenen, mit #, ##, ###, usw.

Im Text wird Schrägschrift durch ein vor- und nachgestelltes Sternchen erzeugt, d.h. \*so\*. Fettschrift in ähnlicher Weise durch zwei Sternchen, d.h. \*\*so\*\*. Für das Tiefstellen z.B. in H<sub>2</sub>O wird die Tilde (~) genutzt, also so: H~2~O. Für das Hochstellen z.B. in <sup>14</sup>C-Daten das Hütchen (^), also so: ^14^C.

Sehr einfach ist das Einbetten von aktiven Links (beim R-Commander so nicht möglich), entweder so: < <https://www.frank-siegmund.de/>> oder so: ["www.DGUF.de"](<https://www.dguf.de/>) – wobei die schließende eckige Klammer

mit dem Linktext und die öffnende runde Klammer mit dem Link dicht auf dicht folgen müssen.

Als Listen sind automatische Bullet-Point-Listen möglich oder nummerierte Listen:

- \* Bullet-Liste Punkt 1
- \* Bullet-Liste Punkt 2
  - \* Liste Unterpunkt 1
  - \* Liste Unterpunkt 2
- \* Bullet-Liste Punkt 3

oder:

1. Nummerierte Liste Zeile 1
1. Nummerierte Liste Zeile 2
1. Nummerierte Liste Zeile 3
  - \* mit Unterpunkt 3a
  - \* mit Unterpunkt 3b

R-Code wird (wie oben beschrieben) in drei führende und drei abschließende Schrägstriche eingebettet. Man kann während des Arbeitens an seinem Projekt und seiner Dokumentation in RStudio die jeweils relevanten Code-Zeilen markieren und – in RStudio – mit „Run“ (oben) ausführen. Dies ermöglicht es, in enger Verbindung mit ausführbaren R-Code sein Tun zugleich textlich zu dokumentieren, während man ggf. sukzessive seinen R-Code aufbaut und schrittweise verbessert. Später, nach dem „Knitten“, wird der R-Code in einem markierten Fenster in die Text-Ausgabe eingebettet.

Genug! Für Weiteres sei auf die o.g. Einführungen und Dokumentationen verweisen. Vor allem: Markdown ist – wenn man nicht gerade ganze Bücher damit schreiben will – nicht kompliziert und erschließt sich im „learning by doing“.

#### **4.6 KONVENTIONEN IN DEN DIESEM WERK BEIGEgebenEN MARKDOWN-DATEIEN**

Zum weiterführenden Lernen und Üben sind die in das Buch eingebetteten Übungen und Code-Beispiele auch als Markdown-Dateien verfügbar gemacht. Dabei werden wenige Konventionen verfolgt, die ich auch für das praktische Arbeiten empfehle: Man trenne zwischen dem oder den Archiv-Ordner für seine Daten etc. und dem aktiven R-Arbeitsordner. Um stets wechselnde und ggf. sehr lange Ordnernamen für

das Arbeiten mit R zu vermeiden, lege man sich auf der obersten Ebene eines Laufwerks einen Arbeitsordner mit kurzem Namen an, z.B. RData, und lege diesen in RStudio mit `setwd()` auch als Arbeitsordner fest. Alle aktiv bearbeiteten Daten werden dorthin kopiert. In den hier mitgegebenen Markdown-Dateien ist das Arbeitsverzeichnis „d:\RData“.

## 5 LITERATUR

---

- Efron, Bradley & Hastie, Trevor (2016, 2019). *Computer age statistical inference. Algorithms, evidence, and data science*. Cambridge: Cambridge University Press. – Vertiefung zu Kap. 7: zeigt aus Sicht renommierter Statistiker die jüngere Fachgeschichte der Statistik auf, inkl. des Vorhandenseins von Moden und unterschiedlichen Stilen. Anregend, anspruchsvoll.
- Kabacoff, Robert I. (2021). *R in action: Data analysis and graphics with R. 3<sup>rd</sup> ed.* Shelter Island: Manning. – Die dritte, stark überarbeitete und erweiterte Auflage dieses Klassikers ist auf der Verlagswebsite in einer noch in Arbeit befindlichen Preview-Fassung offen einsehbar: <https://livebook.manning.com/book/r-in-action-third-edition/welcome/v-4/> [8.8.2020].
- Marwick, Ben (2020). *CRAN Task View: Archaeological Science*. GitHub. <https://github.com/benmarwick/ctv-archaeology> [8.8.2020]. – Kommentierte Sammlung von R-Werkzeugen, die beim Arbeiten mit R in der Archäologie nützlich sind. Sehr wertvolles und anregendes Nachschlagewerk.
- Starmer, Josh (2017-2019). *StatQuest*. YouTube, 183 Videos. <https://www.youtube.com/user/joshstarmer> [8.8.2020]. – Kurze YouTube-Videos zu Themen der Statistik und maschinellem Lernen. Sehr anschaulich animierte Erklär-Videos zu Themen der Statistik, m.E. sehr gut verständlich und hilfreich, eine Empfehlung.
- Wollschläger, Daniel (2017). *Grundlagen der Datenanalyse mit R. Eine anwendungsorientierte Einführung*. 4. Aufl. Berlin: Springer Spektrum. – Das Lehrbuch bietet, was sein Titel verspricht. Es wird hier angeführt, weil es die m.E. beste deutschsprachige Einführung in die Grafik mit R gibt, und zwar zunächst in die R-Standardgrafiken („base“) und nachfolgend in *ggplot2*.
- Marin, Mike (2013-2019). Marin StatsLectures: <https://www.statslectures.com/>

Wer auf YouTube Hilfe für ein Statistik- & R-Problem sucht, erhält in der Regel viele Treffer, von ausnehmend durchwachsender Güte. Ein verlässlicher Quell konkreter, stets trefflicher und vor allem nicht-geschwätziger Hilfe: "Marin Stats Lectures". Mike Marin, University of British Columbia, hat die Lehrvideos für seine Statistik-Lehrveranstaltungen frei online gestellt: kurze Videos von meist 5 - 10 Minuten Dauer, klar auf ein exakt benanntes Thema fokussiert, stets hilfreich. Mike Marin arbeitet mit R-Studio und den Funktionen von R Base, also außerhalb des Tidyverse und der *ggplot2*-Welt. Auf seiner Website erhält man (besser als via YouTube) einen Überblick über alle verfügbaren Videos, kann dort den roten Faden erkennen und sich ggf. sehr gezielt das Gesuchte herauspicken.

• „Meta-Bücher“: (sehr) umfangreiche Sammlungen von freien Büchern und Lern-Materialien über **R** und Themen rund um **R**:

- Baruffa, Oscar (2020). Big Book of R. <https://www.bigbookofr.com/> [29.8.2020].
- Javed, Ali (2020). Free Resources for Learning R Programming for Data Science. <https://www.javedali.net/post/r-resources/> [29.8.2020].