

Empirisch-Wissenschaftliches Arbeiten

Übung zur computergestützten Datenanalyse

© Prof. Dr. Stephan Huber

5. Juni 2024

Inhaltsverzeichnis

Vorwort	1
1 Die Programmiersprache R	3
2 Wissenschaftliche Texte schreiben	4
2.1 WYSIWYG Anwendungen	4
2.2 Vorteile von codebasierten Anwendungen	5
2.3 Einführung in Quarto	6
2.4 Erste Schritte mit Quarto	7
2.5 APA konformes Manuscript erstellen mit Quarto	7
Literatur	8

Abbildungsverzeichnis

Tabellenverzeichnis

Vorwort

Wikipedia sagt:

“Die Psychologie [...] ist eine empirische Wissenschaft”

Diese Unterlagen helfen...

- die Abfolge und die Inhalte der *Übung zur computergestützten Datenanalyse* zu überblicken,
- die Übungsaufgaben zu verstehen und zu bearbeiten und
- die Projektarbeit der Veranstaltung *Empirisch-Wissenschaftliches Arbeiten* erfolgreich zu gestalten.

Die Übung vermittelt...

- Kenntnisse der Programmiersprache R welche eine wissenschaftliche Datenbearbeitung und Datenanalyse ermöglichen.
- Kenntnisse zum programm-basierten Verfassen von wissenschaftlichen Arbeiten (Aufsätze, Bücher, Arbeitspapiere, Hausarbeiten).

Studierende lernen...

- Daten mit der Programmiersprache R und mit Hilfe der integrierten Entwicklungsumgebung RStudio einzulesen, zu bearbeiten und empirisch auszuwerten.
- Empirische Ergebnisse in ein publikationswürdiges Format zu übertragen.
- Einen APA konformes Manuskript mit Quarto, bzw. (R)Markdown, zu erstellen und dies entsprechend zu publizieren.
- Literatur entsprechend wählbaren Zitationsregeln unter Verwendung von Quarto und BibTeX in einen Aufsatz einzuarbeiten.

Studierende sollen...

- Die angeführte Literatur studieren: Ohne eigenständige Vor- und Nachbereitung lassen sich die Programmierkenntnisse nicht erlernen.
- Aktiv um Hilfe bitten: Wenn etwas unklar ist, kann ich individuell während des Kurses versuchen zu helfen. Für eine intensivere Betreuung, bitte ich mich zu kontaktieren, in die [Sprechstunde](#) zu kommen, oder eine außerordentliche Sprechstunde zu vereinbaren. Dies ist möglich und erwünscht.
- Inhaltliche Fragen und Wünsche jederzeit kommunizieren. Es besteht die Möglichkeit diese in das Curriculum aufzunehmen.

Liebe Studierende,

das Erlernen einer Programmiersprache in Verbindung mit empirischen Arbeiten ist eine Herausforderung die Vielen keinen Spaß macht. So ist es nur Verständlich, dass die Sinnhaftigkeit dieses Kurses teilweise von Studierenden angezweifelt wird. Tätigkeiten die keinen Spaß machen, sollten sinnstiftend sein oder zumindest ein monetäres Einkommen sichern. Da das Vorhandensein von empirischen Kenntnissen und einer Programmiersprache in einem Lebenslauf zweifelsfrei in der heutigen Zeit die Vermittlungsfähigkeit und die Verhandlungsposition am Arbeitsmarkt wesentlich verbessern, will ich mich hier kurz bemühen, die Sinnhaftigkeit zu thematisieren.

Ich verstehe die Abneigung gegenüber diesen Kurs: Viele haben sich nicht für ein Studium der Psychologie entschieden, um empirische Methoden und deren computergestützte Umsetzung zu erlernen. In der modernen Welt aber, insbesondere in der psychologischen Forschung, ist ein Verständnis von empirischen Methoden sowie deren computergestützten Umsetzung die praktische Voraussetzung zum Erkenntnisgewinn. Ohne dieses Verständnis verharret man bei rein theoretische und philosophische Überlegungen ohne jede Evidenz. Eine professionell agierende Psychologin und Psychologe, sollte die Fähigkeit besitzen die Literatur in seinem Fach zu begreifen sowie in der Lage ein die Ergebnisse kritisch zu hinterfragen und/oder zu überprüfen.

Ich bemühe mich, die Veranstaltung so attraktiv wie möglich zu gestalten. Ich biete...

- ein ausführliches [Skript zur Programmiersprache R](#) an, welches
 - eine Batterie an [Übungsaufgaben mit Lösungskripten](#) und
 - eine Vielzahl an [interaktive Übungen](#) zum eigenständigen bearbeiten enthält.
- dieses Skript, welches
 - Psychologie-spezifische empirische Inhalte aufgreift und
 - Software vorstellt, welche die Erstellung der Projektarbeit erleichtert.
- mündliche Erklärungen in der Veranstaltung.
- die Möglichkeit spezifische Fragen zu stellen und Unklarheiten anzusprechen.
- individuelle Betreuung während und außerhalb der [Sprechstunde](#).

Wenn sie Vorschläge und Wünsche bezüglich der Inhalte oder der didaktischen Aufbereitung haben, bitte ich diese auszusprechen. Konstruktive Kritik ist sehr willkommen. Ich nehme diese an und ernst. Ob Sie diesen Kurs letztendlich als gelungen betrachten, ist ihrer Wahrnehmung überlassen. Bevor Sie den Kurs aber schlecht evaluieren, bitte ich sie um Folgendes: Fragen Sie sich, ob ihr Wille und ihr Wunsch ausgeprägt genug waren, um sich ernsthaft mit den Inhalten und den Angeboten auseinanderzusetzen und ob sie mir evtl. die Gelegenheit gegeben haben auf Ihre Wünsche einzugehen.

Abschließend wünsche ich Ihnen viel Freude mit dem Kurs und den angebotenen Unterlagen und Inhalten. Ich freue mich, diesen Kurs halten zu dürfen und zu können. Es ist mir stets eine Freude, den anwesenden Studierenden R, Quarto, BibTeX und Co. erklären zu können. Ich wünsche mir, möglichst Viele mit den dargebotenen Inhalten, das Studium zu bereichern und die Bearbeitung der Projektarbeit sowie der Abschlusarbeit zu erleichtern.

Ihr
Stephan Huber

1 Die Programmiersprache R

Ich bitte Sie, studieren sie das Skript *How to use R for data science* [Huber, 2024].

In den ersten Wochen werden wir uns ausschließlich damit beschäftigen, die Programmiersprache R zu erlernen. Das ist ähnlich mühsam wie das Erlernen einer wirklichen Sprache. Wer keine Lust darauf hat, wird es schwer haben. Ich beispielsweise hatte in der Schule überhaupt keine Lust auf Englisch und Latein. Dementsprechend schlecht waren meine Noten. Ich musste die siebte Klasse wiederholen und bis zum Abitur waren Sprachen für mich ein nötiges Übel. Erst als ich im Studium sah, dass praktisch alle relevanten und für mich interessanten Artikel und Bücher in englischer Sprache verfasst sind, machte das Erlernen der Sprache einen Sinn für mich. Jetzt lehre ich abseits dieses Kurses ausschließlich auf Englisch und publiziere in englischer Sprache. Interesse und Freude sind mächtige Katalysatoren für Erfolg.

Das Schreiben von Code ist für die meisten Studierenden Neuland. Studierende im Jahr 2024 sind zumeist mit dem Smartphone aufgewachsen und demnach sind Sie es gewohnt, ihre Geräte (Smartphone, Tablet, Desktop-PC) ohne zur Hilfenahme einer Programmiersprache zu steuern. Das ist wunderbar: Die grafische Benutzeroberfläche heutzutage erlaubt eine effiziente und intuitive Art der Steuerung mit der Computermaus, durch Wischen, Tippen oder durch Spracheingabe. Leider hat diese Art der Steuerung massive Nachteile beim wissenschaftlich orientierten Arbeiten mit Daten. Insbesondere was die Reproduzierbarkeit der Ergebnisse und die Flexibilität des Arbeitsprozesses anbelangt, stößt man bei Applikationen ohne Code an Grenzen. Die Vor- und Nachteile von Script-basierten Arbeiten werden im Kapitel *The limitations of no-code applications* [Huber, 2024] ausführlich erläutert.

Zusammenfassend sollten Studierende nach den ersten 5-6 Unterrichtseinheiten folgendes getan haben beziehungsweise erlernt haben:

- Installation von
 - R,
 - RStudio und der
 - gängigsten Pakete.
- Wissen über...
 - den Aufbau von R Skripten.
 - die Verwendung von Funktionen, Objekten und Pakete in R.
 - die grundsätzlichen Eigenheiten der Programmiersprache R.
 - das Ausführen von Code (**Ctrl+Enter**, Klicken von **Run**, oder durch die Funktion `source()`).
 - die Verwendung von Pipes mit dem Pipe Operator (`|>`).
 - die Verwendung von logischen und relativen Operatoren.
 - die Funktionen des Pakets `dplyr` (`filter()`, `select()`, `mutate()`, `summarise()`, etc.)

2 Wissenschaftliche Texte schreiben

Studierende verwenden zum Verfassen wissenschaftlicher Texte gerne Microsoft Word, Apples Pages oder LibreOffice. Diese Textverarbeitungsprogramme zeigen das Dokumentenlayout bereits während des Schreibens an. Dies wird auch als “What you see is what you get” (WYSIWYG) Prinzip bezeichnet. Dieses Prinzip und die entsprechenden Anwendungen sind weit verbreitet und erscheinen vielen als alternativlos. Dies trifft jedoch keineswegs zu: Es gibt eine Vielzahl alternativer Textsatz-Systeme wie LaTeX, Markdown, R Markdown und Quarto, die beachtenswerte Vorteile bieten. Nicht ohne Grund nutzen viele professionell arbeitende Wissenschaftler und Publizisten diese Alternativen. Eine große Anzahl von Doktorarbeiten und wissenschaftlichen Aufsätzen wird mit LaTeX verfasst, und nahezu alle Herausgeber und Verlage arbeiten mit codebasierten Alternativen, die nicht dem WYSIWYG-Prinzip folgen.

Bei codebasierten Alternativen werden die Angaben zum Layout entweder an den Anfang des Textes oder direkt in den Fließtext eingefügt. Das endgültige Dokument wird erst nach der Umwandlung (auch Kompilieren oder Rendern genannt) in ein Dokumentformat wie PDF sichtbar. Dies mag zunächst gewöhnungsbedürftig sein und sicherlich weniger intuitiv als eine WYSIWYG-Benutzeroberfläche. Doch die intuitivste Lösung ist nicht zwangsläufig die beste oder einfachste. Die vielen Studienarbeiten, die ich betreuen durfte, zeigen deutlich, dass die intuitiven Features von MS Word und Pages sich mittel- und langfristig oft als zeitaufwändig erweisen und Nutzer nur unzureichend dabei unterstützen, Fehler beim Verfassen von wissenschaftlichen Arbeiten zu vermeiden. Studierende, die sich gegen eine WYSIWYG-Office-Anwendung entscheiden, sind in der Regel weniger frustriert und erfolgreicher – zumindest trifft dies auf die von mir betreuten Arbeiten zu.

Codebasierte Anwendungen ermöglichen es den Schreibenden, sich auf die eigentliche Textarbeit zu konzentrieren. Die Feinheiten des Formats und die Einhaltung von Zitationsregeln werden größtenteils automatisch von der Software übernommen. Die notwendige Anfangsinvestition, sich etwa mit Quarto vertraut zu machen, macht sich schnell bezahlt, und die Qualität der wissenschaftlichen Texte verbessert sich spürbar.

In den folgenden Unterabschnitten werde ich zunächst die typische Nutzung von WYSIWYG-Anwendungen umreißen. Anschließend werde ich die Vorteile des codebasierten Verfassens von Texten am Beispiel von Quarto beleuchten, um dann zu erläutern, wie das Verfassen von Texten mit Quarto gelingen kann.

2.1 WYSIWYG Anwendungen

Die Nutzung klassischer Textverarbeitungsprogramme wie Microsoft Word oder Apple Pages zum Verfassen wissenschaftlicher Texte ist in der studentischen Welt weit verbreitet. Obwohl diese Programme für alltägliche Schreibprojekte benutzerfreundlich sind, erzeugen sie erheblichen Mehraufwand, um den Ansprüchen wissenschaftlicher Arbeit gerecht zu werden.

Ein erstes Problem ist die Einbindung von Literatur. Die korrekte Formatierung nach verschiedenen Zitierrichtlinien ist oft alles andere als intuitiv und Fehler treten leicht auf. Dies gilt insbesondere, wenn die von der Software bereitgestellten Zitat- und Bibliografiefunktionen

nicht oder nicht richtig genutzt werden. Anstelle externer Zitationsmanager zu nutzen und sich in deren Gebrauch einzuarbeiten, verfassen viele Studierende Zitate und Literaturlisten manuell. Dies führt erfahrungsgemäß zu zahlreichen kleinen und manchmal größeren Fehlern, die vermeidbar wären.

Ein weiterer Schwachpunkt von studentischen Arbeiten ist die Einhaltung spezifischer Formatierungsvorgaben. Akademische Institutionen und Journale fordern oft eine strenge Beachtung von Formatierungsrichtlinien, inklusive der Gestaltung von Titelseiten, Kopf- und Fußzeilen, Seitenrändern und Überschriftenhierarchien. Zwar bieten Word und Pages Vorlagen und Stile an, diese müssen jedoch für jedes Dokument individuell angepasst und oft aufgrund geringfügiger Änderungen im Text modifiziert werden. Wenn eine Formatanpassung erforderlich wird, ist dies meistens nur mit großem Aufwand möglich.

Das Einfügen empirischer Ergebnisse wie statistischer Daten und Grafiken stellt eine zusätzliche Hürde dar. In Word und Pages gestaltet sich der Vorgang häufig manuell: Forschungsdaten müssen aus Statistiksoftware exportiert, als Bilder abgespeichert und anschließend in das Dokument eingebunden werden. Ändert sich etwas an den Daten, so muss dieser mühsame Prozess wiederholt werden, was den Arbeitsaufwand signifikant erhöht und die Fehleranfälligkeit steigert.

2.2 Vorteile von codebasierten Anwendungen

Der traditionelle Ansatz zum Verfassen wissenschaftlicher Texte mittels MS Word oder Pages kann für Studierende zeitaufwendig und fehleranfällig sein. Im folgenden Abschnitt möchte ich Quarto (bzw. R Markdown) vorstellen, eine moderne Alternative, die folgende Vorteile bietet:

- Mit Quarto lassen sich mühelos verschiedene Ausgabeformate generieren. So kann derselbe Text beispielsweise als Website (HTML), Manuskript (PDF, DOCX), Buch (EPUB, PDF) oder in Form von Präsentationsfolien (PDF). Diese Flexibilität erlaubt es, sich eher auf den Inhalt als auf das Format zu konzentrieren.
- Die Formatierung kann in Quarto einfach geändert werden indem bestimmte Vorlagen verwendet werden.
- Literaturreferenzen lassen sich unkompliziert einbinden, während die Einhaltung von Zitierregeln von der Software übernommen wird. Quartos Integration mit Zitationsverwaltungssystemen ermöglicht es, Literaturverweise und Bibliografien effizienter und konsistenter zu handhaben als beispielsweise in Word.
- Querverweise auf Abschnitte, Tabellen und Abbildungen lassen sich leicht erstellen.
- Die Datenanalyse und das Erstellen von Datenoutputs erfolgen direkt in Quarto. Dadurch sind dargestellte Grafiken und Tabellen stets aktuell und manuelles Nachbearbeiten entfällt, wodurch die Reproduzierbarkeit der Ergebnisse gewährleistet ist.
- Forschende können ihre Datenvisualisierungen ohne manuelle Zwischenschritte direkt in den Text einbetten.
- Versionskontrollsysteme wie Git erleichtern die Zusammenarbeit an wissenschaftlichen Dokumenten, da Änderungen nachverfolgbar sind und integriert werden können, ohne auf komplexe und konfliktträchtige Vergleichstools angewiesen zu sein.

Lesetipp

Für Interessierte empfehle ich den Onlinekurs *Introduction to Reproducible Publications with RStudio*, der explizit erläutert, wie man empirisch nachvollziehbar arbeitet. Eine etwas kompaktere Einführung bietet [Bauer and Landesvatter \[2023\]](#) und das Standardwerk zum

Thema stammt von Gandrud [2020].

2.3 Einführung in Quarto

Quarto kann in RStudio genutzt werden, um APA-konforme Texte zu erstellen. Gehen Sie dazu bitte wie folgt vor:

- Installieren Sie R und R Studio.
- Installieren Sie Quarto folgendermaßen:

```
install.packages("quarto")
```

- Installieren Sie das Paket `tinytex`, um PDF-Dateien zu generieren:

```
install.packages("tinytex")
tinytex::install_tinytex()
```

- Es ist zudem ratsam, weitere Pakete zu installieren, die später benötigt werden könnten:

```
if (!require(pacman)) install.packages("pacman")
pacman::p_load(knitr, rmarkdown, papaja)
```

- Eignen Sie sich Kenntnisse in Markdown an. Markdown ist eine leichtgewichtige Markup-Language zur Formatierung von Klartext. Sie ist eine essenzielle Fähigkeit für die effektive Nutzung von Quarto. Beginnen Sie damit, ausreichend Markdown für die Strukturierung Ihrer Arbeit zu erlernen, einschließlich Überschriften, Listen, Links und Codeblöcken. Markdown ist schnell zu erlernen; ich empfehle dazu den Besuch von www.markdowntutorial.com und das Durcharbeiten der interaktiven Lektionen sowie des Abschnitts *Markdown Basics* auf quarto.org.
- Machen Sie sich vertraut mit Quarto. Als Lektüre dient Telford [2023]: *Enough Markdown to Write a Thesis*, welcher fast alles abdeckt, was für das akademische Schreiben hilfreich ist. Alternativ finden Sie umfassende Informationen zur Arbeit mit Quarto direkt auf der Webseite quarto.org/docs/guide.

i Quarto und R markdown

Quarto ist ein relativ neues Werkzeug und kann als Nachfolger von R Markdown betrachtet werden. Die meisten R Markdown-Dokumente sind mit Quarto kompatibel. Allerdings bietet Quarto einige verbesserte Funktionen gegenüber R Markdown, die die Benutzerfreundlichkeit steigern. Einen detaillierten Überblick über die Unterschiede und Gemeinsamkeiten zwischen den beiden Plattformen finden Sie in [diesem Artikel] (<https://quarto.org/docs/faq/rmarkdown.html>).

2.4 Erste Schritte mit Quarto

- Öffnen Sie RStudio.
- Wählen Sie “File” -> “New File” -> “Quarto Document” und dann “Create”.
- Speichern Sie die neue Datei in einem leeren Ordner und definieren Sie diesen Ordner als Ihr Arbeitsverzeichnis.
- Klicken Sie auf “Render”.
- Besuchen Sie die Webseite [Markdown Basics](#), fügen Sie etwas Markdown in Ihr Dokument ein und klicken Sie erneut auf “Render”.
- Klicken Sie auf den Pfeil neben dem “Render”-Knopf. Hier können Sie andere Dateiformate auswählen und diese generieren. Probieren Sie es aus.
- Konsultieren Sie die Webseite PDF Basics und ergänzen Sie Ihren Header mit den dort gefundenen Informationen.
- Versuchen Sie das Paper von [Huber and Rust \[2016\]](#), das Sie hier finden, in Ihrem Dokument zu zitieren.
 - Klicken Sie dazu auf “Visual”,
 - gehen Sie an die Stelle im Text, an der Sie das Paper zitieren möchten, und wählen Sie “Insert” -> “Citation”.
 - Suchen Sie im Kontextmenü mithilfe der entsprechenden DOI nach dem Papier und fügen Sie es ein.
- Um mit dem APA Version 7-Stil zu zitieren, schreiben Sie folgendes in den YAML-Header:

```
csl: "https://www.zotero.org/styles/apa"
```

- Wählen Sie einen anderen Zitierstil von www.zotero.org/styles. Rendern Sie dann das Dokument erneut und beobachten Sie die Unterschiede.

2.5 APA konformes Manuscript erstellen mit Quarto

Literatur

Paul C. Bauer and Camille Landesvatter. Writing a reproducible paper with rstudio and quarto. Technical report, 2023. URL <https://doi.org/10.31219/osf.io/ur4xn>.

Christopher Gandrud. *Reproducible research with R and R studio*. Chapman and Hall/CRC, 3 edition, 2020.

Stephan Huber. How to use R for data science, 2024. URL <https://hubchev.github.io/ds/>.

Stephan Huber and Christoph Rust. Calculate travel time and distance with openstreetmap data using the open source routing machine (osrm). *The Stata Journal*, 16(2):416–423, 2016.

Richard J Telford. Enough markdown to write a thesis, 9 2023. URL <https://biostats-r.github.io/biostats/quarto/>.