

Statistik1

Daten einlesen

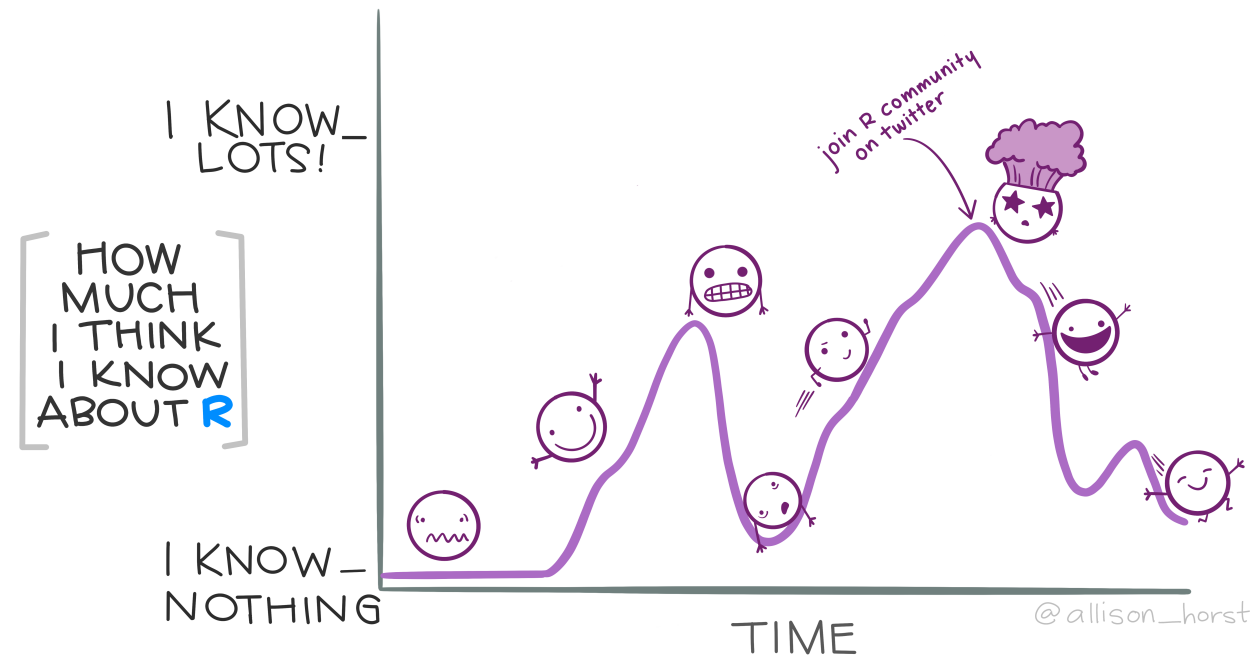
Sebastian Sauer

Einstieg

Lernziele

- Sie können R und eine IDE (RStudio oder Positron) starten.
- Sie können R-Pakete installieren und starten.
- Sie können Variablen in R zuweisen und auslesen.
- Sie können Daten in R importieren.
- Sie können den Begriff *Reproduzierbarkeit* definieren.

Lernverlauf: Höhen und Tiefen



Life is a roller-coaster. You just have to ride it (Horst, 2024)

Höhen und Tiefen beim Lernen von R sind normal — mit etwas Übung geht bald alles leichter von der Hand.

Einstieg

Errrstkontakt

Warum R?

1. R ist *kostenlos* — andere Softwarepakete für Datenanalyse sind teuer.
2. R ist *quelloffen* — jeder kann die Befehle einsehen und beitragen.
3. R hat die *neuesten* statistischen Methoden.
4. R hat eine große *Community*.
5. R ist *maßgeschneidert* für Datenanalyse.

Definition: Reproduzierbarkeit

Definition: Reproduzierbarkeit

Ein (wissenschaftlicher) Befund ist reproduzierbar, wenn andere Personen mit der Analysemethodik zum gleichen Ergebnis kommen wie in der ursprünglichen Analyse ([Plessner, 2018](#)).

Daten + Syntax + genaue Messbeschreibung = reproduzierbar

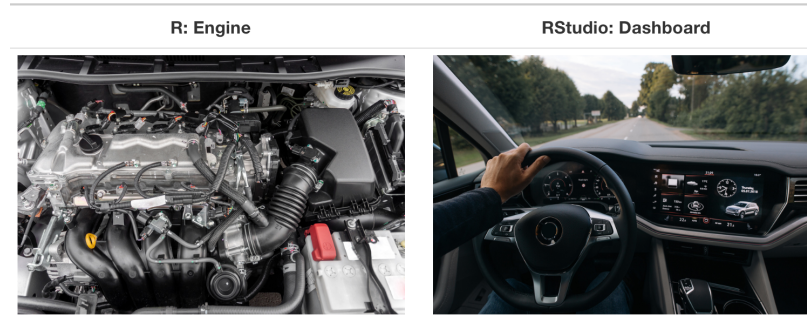
Aus der Forschung: Reproduzierbarkeit in der Psychologie

Obels et al. ([2020](#)) untersuchten die Reproduzierbarkeit von Registered Reports (2014–2018):

Von 62 Artikeln hatten 41 Daten verfügbar, 37 Analyseskripte. Für 36 Artikel waren beides verfügbar. Reproduziert werden konnten die Hauptergebnisse von 21 Artikeln.

Insgesamt war also etwa **jede dritte Studie** reproduzierbar — da gibt es noch viel Luft nach oben.

R & IDE: Arbeitsteilung



R macht die Arbeit, die IDE sorgt für Komfort und Übersicht ([Ismay & Kim, 2020](#))

- R erledigt die eigentliche Rechenarbeit.
- Eine **IDE** (integrierte Entwicklungsumgebung) sorgt für Komfort.
- Zwei gleichwertige, kostenlose IDEs stehen zur Wahl: **RStudio** und **Positron** (beide von Posit) — mehr dazu im Buch.

Installation

R und eine IDE installieren

1. R installieren: <https://cloud.r-project.org> (Windows/MacOS/Linux)
2. Eine IDE installieren — Ihre Wahl:
 - RStudio: <https://posit.co/download/rstudio-desktop/>
 - Positron: <https://positron.posit.co/download.html>

Beide IDEs finden R selbständig — gestartet wird immer nur die IDE, nicht R direkt.

Hinweis

Alle Inhalte dieses Buchs funktionieren identisch mit RStudio und Positron.

Tipp

Master-Shortcut `Strg+Shift+P` öffnet die Befehlspalette — damit finden und starten Sie *jeden* Befehl der IDE, ohne den Klickpfad im Menü suchen zu müssen (funktioniert in RStudio und Positron).

R-Pakete

Was sind R-Pakete?

- R hat einen *modularen Aufbau*: Erweiterungen (“Pakete”) bringen zusätzliche Befehle und Daten mit.
- Stellen Sie sich ein Paket wie ein Buch vor: Erst installieren (“kaufen”), dann laden (“lesen”).
- Auf CRAN, dem offiziellen “Web-Store”, gibt es ca. 20.000 Pakete ([Hornik et al., 2023](#)).

Pakete installieren & starten

- Installieren (einmalig): per Klick (RStudio) oder Code:

```
1 install.packages("tidyverse")
```

- Starten (bei jedem Neustart neu):

```
1 library(tidyverse)
```

- Für dieses Buch benötigt: `tidyverse` ([Wickham et al., 2019](#)) und `easystats` ([Lüdecke et al., 2022](#)) — beides “Sammelpakete”, die mehrere Pakete auf einmal bereitstellen.

Mit R arbeiten

Projekte

- Ein *Projekt* ist letztlich ein Ordner, der als Basis für zusammengehörige Dateien dient.
- Vorteil: Dateien lassen sich ohne vollständigen Pfad ansprechen, z. B. `"daten.csv"` statt eines langen Pfads.
- Nutzen Sie Projekte — das erspart Ärger mit Pfaden.

Skriptdateien & Quarto-Dokumente

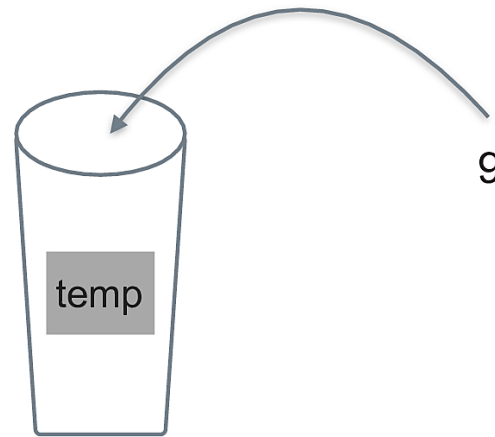
- Eine *Skriptdatei* sammelt R-Befehle (“Syntax”) in einer Textdatei.
- Speichern: **Strg+S** · Öffnen: **Strg+O**
- **Quarto** erlaubt es, R-Syntax direkt in Dokumente (HTML, PDF, ...) einzubetten — Ausgaben landen automatisch im fertigen Dokument.
- Sie sind nicht verpflichtet, Quarto zu nutzen — Skriptdateien reichen auch.

Errisch für Einsteiger

Variablen

```
1 richtige_antwort <- 42  
2 falsche_antwort <- 43  
3 typ <- "Antwort"  
4 ist_korrekt <- TRUE
```

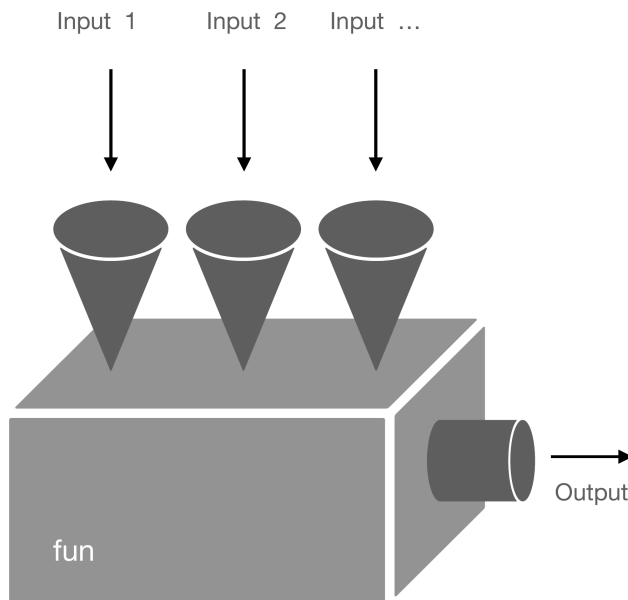
- Zuweisungspfeil `<-` (bevorzugt) oder `=` — beides funktioniert.
- Eine Variable ist wie ein Becher mit Wert und Beschriftung.



Definition: Funktion

Definition: Funktion

Eine Funktion ist eine Regel, die jedem Eingabewert (Argument) einen Ausgabewert zuordnet — eine “Maschine”, die Eingabedaten in Ausgabedaten umwandelt.



Definition: Vektor

Definition: Vektor

Ein *Vektor* ist eine geordnete Folge von Werten (“Elementen”), erstellt mit der Funktion `c()`.

```
1 x <- c(1, 2, 3)
2 namen <- c("Anni", "Bert", "Charlie") # Text-Vektor
```

Unsere erste statistische Funktion

```
1 Antworten <- c(42, 43)
2 meanAntworten)
```

R arbeitet verschachtelte Befehle *von innen nach außen* ab:

`meanAntworten) → mean(c(42, 43)) → 42.5`

Funktionsargumente

```
1 mean(x, trim = 0, na.rm = FALSE, ...)
```

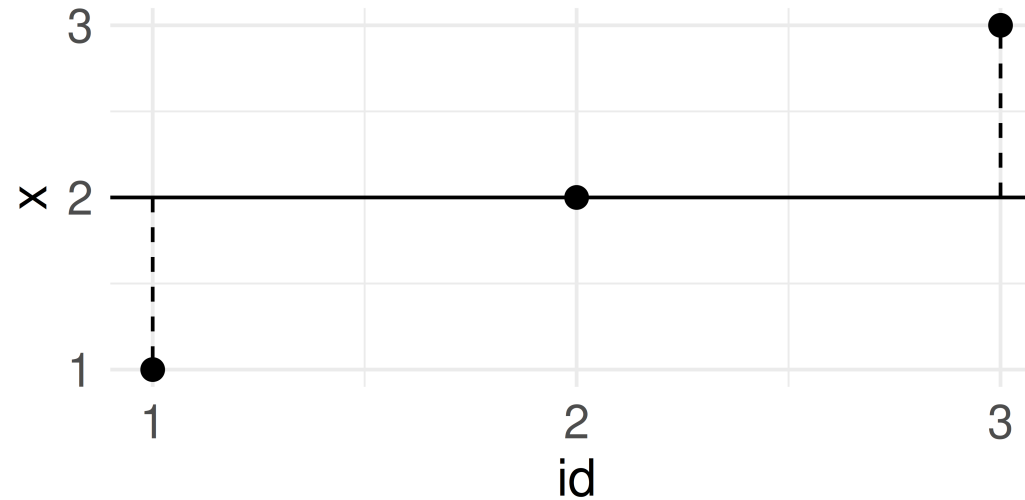
- `x`: der Vektor, für den der Mittelwert berechnet wird
- `trim`: sollen Extremwerte “abgeschnitten” werden?
- `na.rm`: sollen fehlende Werte (`NA`) entfernt werden?
- Argumente mit *Voreinstellung* (default) müssen nicht befüllt werden.
- Ohne benannte Argumente zählt die Reihenfolge: `mean(Antworten, 0, FALSE)`.

Vorsicht bei fehlenden Werten

```
1 Antworten <- c(42, 43, NA)
2 mean(Antworten)           # ergibt NA
3 mean(Antworten, na.rm = TRUE) # entfernt NA, rechnet trotzdem
```

NA steht für *not available* — R macht damit auf fehlende Werte aufmerksam, statt sie stillschweigend zu ignorieren.

Vektoriellles Rechnen



```
1 x <- c(1, 2, 3)
2 x - mean(x)
```

Eine Operation wird automatisch auf *jedes* Element angewandt
— kein Loop nötig.

Ich brauche R-Hilfe!

- Hilfe zu einer Funktion: `help(fun)` oder Help-Panel der IDE
- In welchem Paket wohnt eine Funktion? *RDocumentation* durchsuchen
- Fehler nicht klar? Meist hat schon jemand das Problem gelöst — stackoverflow.com
- R hängt oder spinnt? Konsole/Session neu starten (AEG-Prinzip: Aus-Ein-Gut)
- **WARNING** ist meist harmlos, **Error** bedeutet: Syntax läuft nicht durch

Mit Daten arbeiten

Wo sind meine Daten?

Eine Datei liegt entweder

1. irgendwo im Internet, oder
2. irgendwo auf Ihrem Computer (z. B. im Projektordner).

In beiden Fällen definieren *Pfad* und *Dateiname* den Aufenthaltsort.

Gebräuchlichste Tabellenformate: **CSV** (einfache Textdatei) und **XLSX** (Excel).

Daten importieren

```
1 # aus einem R-Paket:  
2 data("mariokart", package = "openintro")  
3  
4 # von einer Webseite:  
5 mariokart <- read.csv("https://.../mariokart.csv")  
6  
7 # vom eigenen Computer:  
8 d <- read.csv("mariokart.csv")
```

RStudio bietet zusätzlich einen Klick-Import-Assistenten ohne R-Code.

Definition: Dataframe

Definition: Dataframe

Ein Dataframe (auch “Tibble”) ist ein Datenobjekt zur Darstellung von Tabellen: eine oder mehrere gleich lange, benannte Spalten. Jede Spalte ist selbst ein Vektor.

Tabellen ansehen: Namen eingeben, oder `View(mariokart)`.

Logikprüfung

Logische Prüfungen in R

```
1 x <- 42
2 x == 42 # Gleichheit prüfen (doppeltes Gleichheitszeichen!)
```

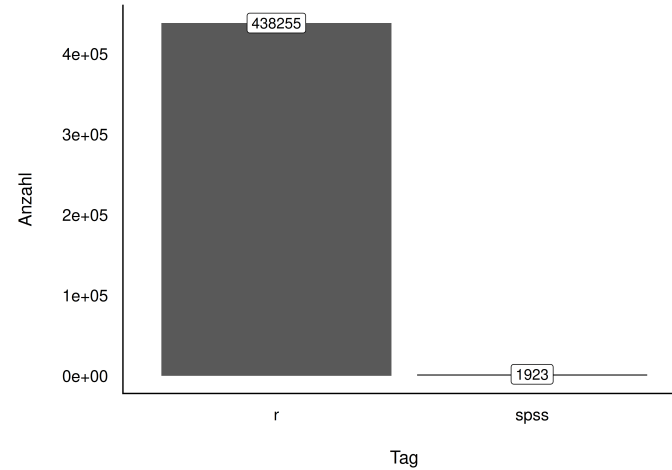
Prüfung auf	R-Syntax
Gleichheit	<code>x == Wert</code>
Ungleichheit	<code>x != Wert</code>
Größer / kleiner	<code>x > Wert / x < Wert</code>
Logisches UND / ODER	<code>& / </code>

Vorsicht

`x = 73` prüft **nicht** auf Gleichheit, sondern weist zu!

Praxisbezug

Wird R wirklich genutzt?



Faktor ~200 zwischen R und SPSS — ein Indiz, dass R in der Praxis stark gefragt ist. Berufe mit Datenbezug zählen zu den stark wachsenden Berufsfeldern ([Berger, 2019.](#)).

Zusammenfassung

- R ist kostenlos, quelloffen, aktuell und community-getragen
- R + IDE (RStudio oder Positron) arbeiten Hand in Hand
- R-Pakete: einmalig installieren, bei jeder Sitzung neu starten
- Variablen (`<-`), Funktionen (`fun(x)`) und Vektoren (`c()`) sind die Grundbausteine
- Daten importieren: aus Paketen, per URL oder vom eigenen Computer
- Logikprüfungen: `==` prüft, `=` weist zu

Literaturhinweise

- “Warum R? Warum, R?” in Sauer ([2019](#)) — Pro und Contra, Starten, Grundlagen von “Errisch”
- Ismay & Kim ([2020](#)), [Kapitel 1](#) — anwenderfreundlich, frei online
- Çetinkaya-Runde & Hardin ([2021](#)) — etwas formaler, passendes Niveau für angewandte Bachelor-Statistik

Referenzen

- Berger, G. (2019, Dezember 10). *The Jobs of Tomorrow: LinkedIn's 2020 Emerging Jobs Report*. <https://www.linkedin.com/blog/member/career/the-jobs-of-tomorrow-linkedins-2020-emerging-jobs-report>
- Çetinkaya-Runde, M., & Hardin, J. (2021). *Introduction to Modern Statistics*. <https://openintro-ims.netlify.app/>
- Hornik, K., Ligges, U., & Zeileis, A. (2023). Changes on CRAN. *The R Journal*, 15, 295–296.
- Horst, A. (2024). *Statistics Artwork* [Artwork]. <https://allisonhorst.com/>
- Ismay, C., & Kim, A. Y.-S. (2020). *Statistical Inference via Data Science: A ModernDive into R and the Tidyverse*. CRC Press / Taylor & Francis Group. <https://moderndive.com/>
- Lüdecke, D., Ben-Shachar, M. S., Patil, I., Wiernik, B. M., Bacher, E., Thériault, R., & Makowski, D. (2022). easystats: Framework for Easy Statistical Modeling, Visualization, and Reporting. *CRAN*. <https://doi.org/10.32614/CRAN.package.easystats>
- Obels, P., Lakens, D., Coles, N. A., Gottfried, J., & Green, S. A. (2020). Analysis of Open Data and Computational Reproducibility in Registered Reports in Psychology. *Advances in Methods and Practices in Psychological Science*, 3(2), 229–237. <https://doi.org/10.1177/2515245920918872>
- Plesser, H. E. (2018). Reproducibility vs. Replicability: A Brief History of a Confused Terminology. *Frontiers in Neuroinformatics*, 11, 76. <https://doi.org/10.3389/fninf.2017.00076>
- Sauer, S. (2019). *Moderne Datenanalyse mit R: Daten einlesen, aufbereiten, visualisieren und modellieren*. Springer. <https://www.springer.com/de/book/9783658215866>
- Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., ... Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43), 1686. <https://doi.org/10.21105/joss.01686>