

Praxislabor Digitale Geisteswissenschaften

# Einführung in R

Michelle Kamolz  
m.kamolz@ub.uni-frankfurt.de



## Kursinhalt

- Kurzer Einstieg
- Was ist R?
- Typen und Syntax
- Die Entwicklungsumgebung
- First Steps, Packages und Datenimporte
- Anwendungsbeispiele



## Was ist R?

- 1992 von Statistikern entwickelt
- Anwendungsorientiert und kostenlos, auch für Nicht-Programmierer
- Wird von einer weltweiten Community gepflegt und weiterentwickelt
- Alternative zu kostenpflichtigen Programmen

## Was ist R? - Stärken

- Statistische und graphische Aufbereitung von Daten
- Aber auch Textmining und Verarbeitung für Präsentation von Hypothesen (Rmarkdown)
- Andere Programmiersprachen und Anwendungen lassen sich verknüpfen (z.B. Python, Latex)
- Starke weltweite Community
- Viele Probleme können durch Googlesuche oder Einsicht auf bekannten Seiten gelöst werden, wie <https://stackoverflow.com/questions/tagged/r> oder z.B. <https://github.com/rstudio/>

## Was ist R? - Schwächen

- Teilweise kryptische Fehlermeldungen
- Updates vs. Funktionalität von Packages
- Der Einstieg wird oft als schwer empfunden
- <https://github.com/awesomedata/awesome-public-datasets>



## Warum R als Geisteswissenschaftler?

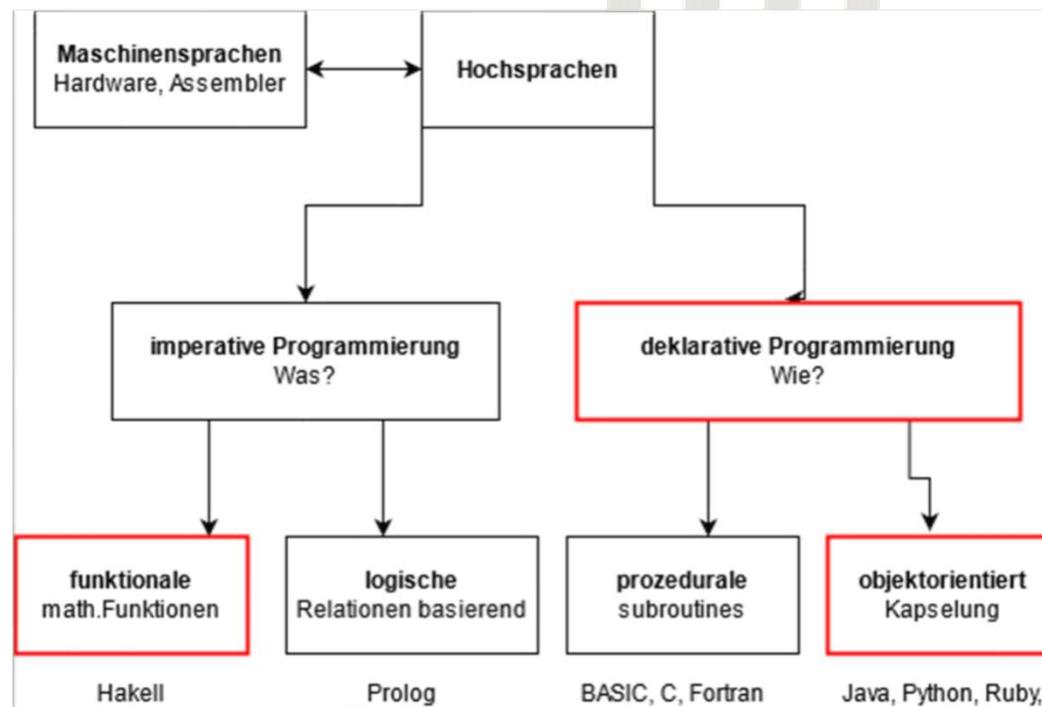
- Erfahrung mit Daten
- Niederschwellige Zugänge zur Sprache; kostenlose Software, große Community, viele kostenfreie Beginner-Kurse

*“Data scientists realize that they face technical limitations, but they don’t allow that to bog down their search for novel solutions. As they make discoveries, they **communicate** what they’ve learned and suggest its implications for new business directions. Often they are **creative in displaying information visually and making the patterns they find clear and compelling.**”*

- Davenport, Thomas H/ Patil, D.J. (2012): *Data Scientist: The Sexiest Job of the 21st Century*

## R: funktional, objektorientiert und dynamisch

- R hat funktionale Fähigkeiten, lässt sich aber als deklarative Sprache verstehen
- Dynamisch: Bearbeitung in Teilabschnitten



## Typen und Syntax

- Wie in jeder Sprache gibt es verschiedene Typen und Klassen, die man nur in den geeigneten Kontexten verwenden kann
- Auch R unterscheidet in verschiedene Datentypen
- Warum muss man das wissen? – weil die Daten dementsprechend verarbeitet werden (können), Funktionen können nicht Texte und Zahlen miteinander vermischen
- Importieren und reinigt man also seine Daten vor der Verwendung, muss man ein Grundverständnis der Datentypen haben

## Typen und Syntax

- **Strings:** Zeichenketten; **character** sind eigentlich einzelne Zeichen, in R werden **strings** jedoch so genannt
- **Integer:** ganze Zahlen
- Achtung: R weist in der Regel allen Zahlen den Datentypen **numeric** zu, eine Gleitkommazahl (3 wird als 3.0 gezählt).  
Man umgeht das, indem man ein L dahinter setzt (3L) oder – im Fall von größeren Datensätzen – `as.numeric` verwendet
- **Logical:** boolesche Variablen (TRUE/ FALSE)

## Typen und Syntax

- **Objekte**; Speicher von zugewiesenen Daten [values],  
z.B.  
`a ← 3`  
`print(a)`  
`[1] 3`
- **Funktionen**; Komplexere Befehle, damit werden Objekte und Werte bearbeitet  
Sie bestehen dabei aus dem Namen und den jeweiligen Argumenten (den Dingen, die durchgeführt, zugewiesen, ... werden sollen) in einer Klammer, z.B.  
`colnames(tabelle1) ← c("Alter", "Wohnort", "Familienstand")`  
oder  
`seq(1,13,3)`  
`[1] 1 4 7 10 13`

## Typen und Syntax

- Groß- und Kleinschreibung: R ist case-sensitiv  
`view(tabelle1)` funktioniert nicht  
`View(tabelle1)` jedoch schon
- Besetzte Namen: Funktionen können nicht für Objekte (Vektoren) verwendet werden, beachten sie auch; keine Zahlen am Anfang, am besten sprechend, aber nicht zu lang
- Klammern – achten sie darauf, dass alle Klammern geschlossen sind
- Nesting: R liest von innen nach außen (die inneren Funktionen werden zuerst abgearbeitet)
- Innerhalb der Klammern liest R erst eine Zeile und dann eine Spalte  
`[3,5]` ist also die Zeile 3 die Spalte 5

## R installieren

R: <https://cran.rstudio.com/>

RStudio: <https://posit.co/download/rstudio-desktop/>



## Weitere Literatur

### Zur Motivation:

Somers, James (2018): *The scientific paper is obsolete. Here's what's next*, The Atlantic, <https://www.theatlantic.com/science/archive/2018/04/the-scientific-paper-is-obsolete/556676/> , zuletzt abgerufen am 07.04.21, 11:18 Uhr.

Davenport, Thomas H/ Patil, D.J. (2012): *Data Scientist: The Sexiest Job of the 21st Century Meet the people who can coax treasure out of messy, unstructured data.*, Harvard Business Review, <https://hbr.org/2012/10/data-scientist-the-sexiest-job-of-the-21st-century>, zuletzt abgerufen am 18.05.2021, 17:20 Uhr.

### Zum vertiefen der Kenntnisse:

Grogan, Michael(2018): Python vs. R for data science, O'Reilly Media: Sebastopol, CA. [über die Bib als Ebook verfügbar]

Grolemund, Garrett/ Wickham, Hadley (2016): R for Data Science. Visualize, transform, model, tidy and import data, O'Reilly: Sebastopol, CA, <https://r4ds.had.co.nz/index.html>, zuletzt abgerufen am 07.04.21, 12:05 Uhr.

Wickham, Hadley (2020): Advanced R. Second Edition.