

Statistik 2 – Tutorate

Sitzung 1: R-Basics

Marco Giesselmann, Rémy Blum, Federica Bruno, Simon Honegger, Nora Zumbühl

Lernziele dieser Sitzung



Organisatorisches

Semesterplan
Vorbereitung
Literatur
Website

R-Basics

Starten von RStudio
Arbeiten mit Skripten
R-Syntax Basics
Installation und Aktivierung von Packages
Einlesen der Daten
Dateninspektion

Teil 1: Organisatorisches



1.1 Semesterplan

Termine (Grau: Zentrale Referenz, vorläufiger Plan, Stand 16.1.2025)

	Diaz-Bone	Gehring/Weins	Urban/Mayerl	Ludwig-Mayerhofer / Liebeskind/ Geissler	Tutorat (Folgwoche)
Block 1: Wiederholung und Einführung					
2.0 Übersicht, Grundlagen und Organisatorisches					R Basics
2.1 Mittelwertvergleich mit gepaarten und ungepaarten Daten		12.3			Datenmanagement
2.2 Regressionsanalyse: Einführung	4.3.1; 4.3.3a	8.1-8.3	2.2.1; 2.2.4	6-6.2	Datenmanagement
2.3 Der Regressionskoeffizient: Illustration und Interpretation	4.3.1; 4.3.3a	8.1-8.3	2.2.1; 2.2.4	6-6.2	Regressionsanalyse Basics
2.4 Regression als Vorhersagemodell	4.3.3b	8.4	2.2.3		Regressionsanalyse Basics
Block 2: Bivariate Regression: Vertiefung und Erweiterung					
2.5 Nicht-Linearität in der Regressionsanalyse 1			4.3-4.3.1		Regressionsanalyse Basics
2.6 Nicht Linearität in der Regressionsanalyse-2			4.3-4.3.1		Linearität und Ausreißer
2.7 «Ausreißer»: Probleme & Lösungen			4.1.1		Linearität und Ausreißer
2.8 Regression und Hypothesentest			3.2-3.3	6.3	Inferenzstatistik
2.9 Konfidenz- und Vorhersageintervalle der Regressionsgerade			3.3.1	6.3	Inferenzstatistik
Block 3: Multiple Regression					
2.10 Kausalität, Störmerkmal und multiple Regression	8.-8.1.3		2.3-2.3.1		Drittvariablen
2.11 Multiple Regression: Praktische Anwendung	8.-8.1.3		2.3-2.3.1		Drittvariablen
2.13 Multiple Regression: Kategoriale unabhängige Variablen	4.3.3		5.1-5.1.3		Visualisierung & Darstellung
2.14 Multiple Regression: Vorhersagemodellierung und Konfidenz					ANOVA
					Probenvorträge ^b
					^b Termine in den Semesterferien, buchbar ab Mitte Mai

5

- Keine Tutorate in der Woche 26.-30.5 (Auffahrt).
- Tutoratsausfall 1.5. (Maifeiertag). Betroffene bitte auf beliebiges anderes Tutorat ausweichen.

- Haben alle R und R-Studio installiert?
 - <https://posit.co/download/rstudio-desktop/>
- Alle automatisch im OLAT-Kurs registriert / Mail bekommen?
- URL zur Website erreichbar?
- Zentrale Packages (*tidyverse*) installiert? (->siehe nächste Folie)

- R-Studio bietet einen **Basis-Pool** an Kommandos und Funktionen an
- Dieser Pool lässt sich erweitern mit verschiedenen Add-ons bzw. Apps, den **Packages**
- Besonders wichtig: Das Package-Set **tidyverse**, bestehend aus mehreren Packages (z.B. **dplyr** oder **ggplot2**)
- Für die Regressionsanalyse ausserdem sinnvoll:
 - **stargazer**
 - **visreg**

1.2

Vorbereitung: Reminder Packages

Ähnlich wie bei einer App muss ein Package...

- **nur einmal installiert,**
- aber jedes Mal neu gestartet bzw. **aktiviert** werden, bevor es nach Start von RStudio genutzt werden kann.

```
#Packages installieren  
install.packages("tidyverse")
```

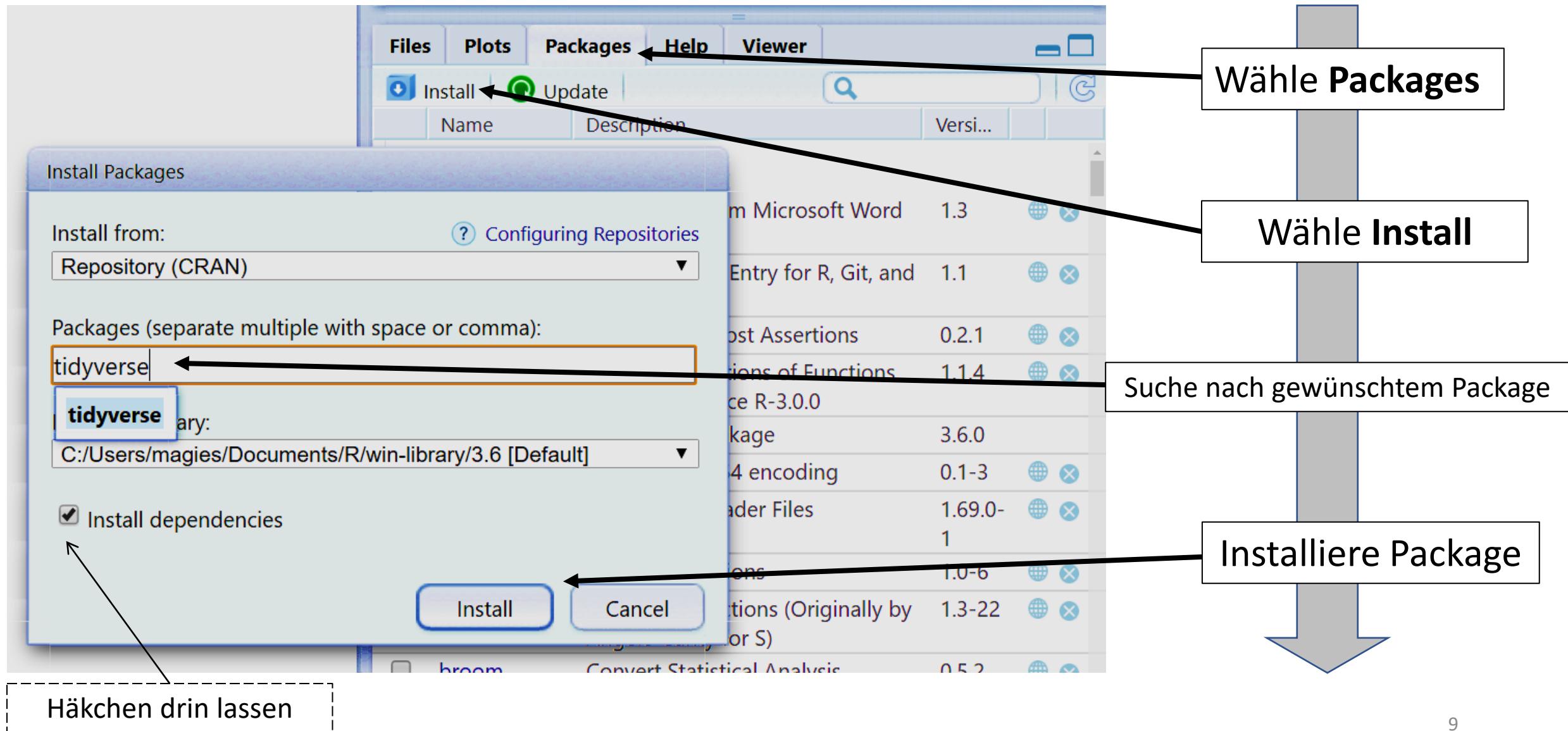
Mit Befehl
install.packages ("...")
direkt herunterladen / installieren

```
#Packages aktivieren  
library(tidyverse)
```

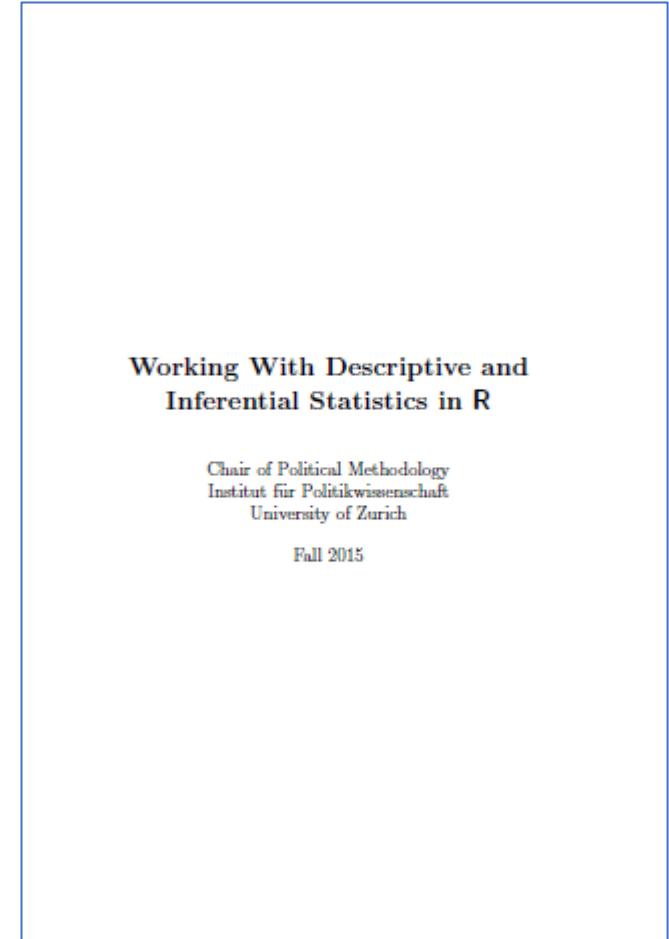
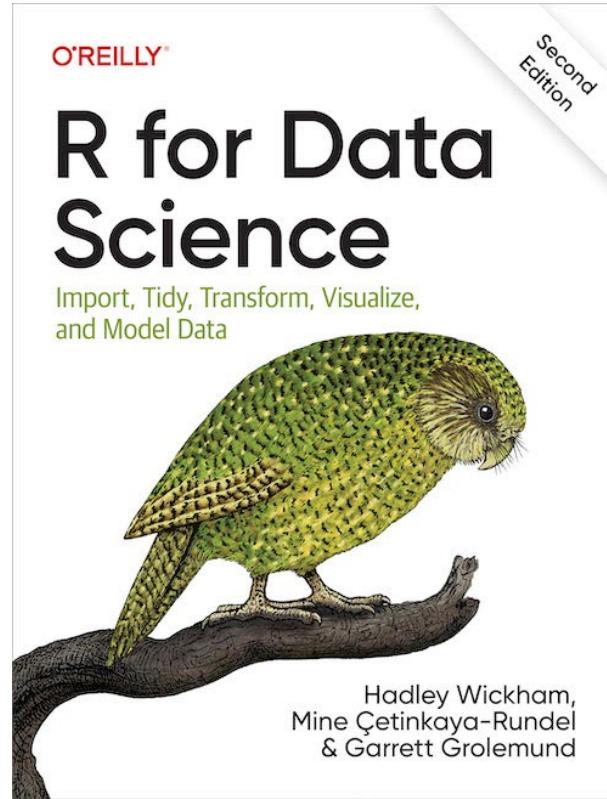
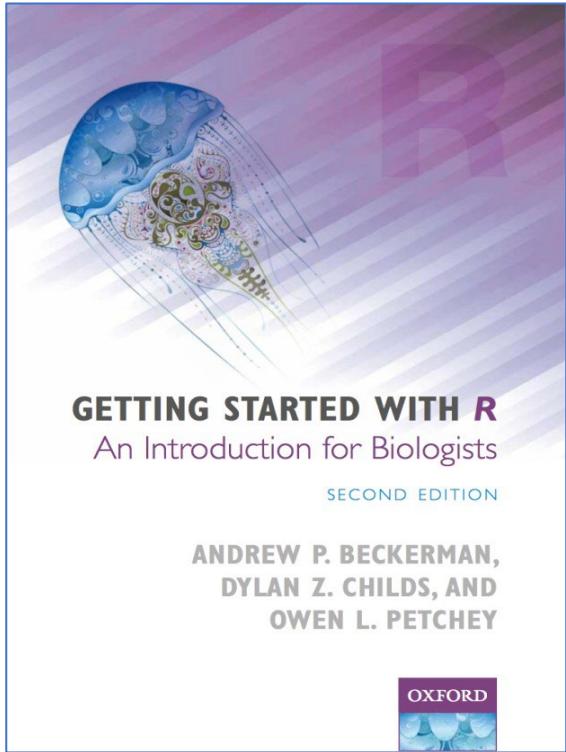
Mit Befehl
library()
müssen Packages dann nur noch
starten bzw. aktiviert werden.

1.2

....Alternativ über die Konsole



1.3 Literatur



Die Links zur Literatur findet ihr auf der [Webseite](#) und/oder auf OLAT

1.4

<http://www.suz.uzh.ch/dataforstat/statistik2/>

Einheitliche und tutoratsübergreifende Arbeitsressourcen

Work in Progress!

Darauf findet ihr:

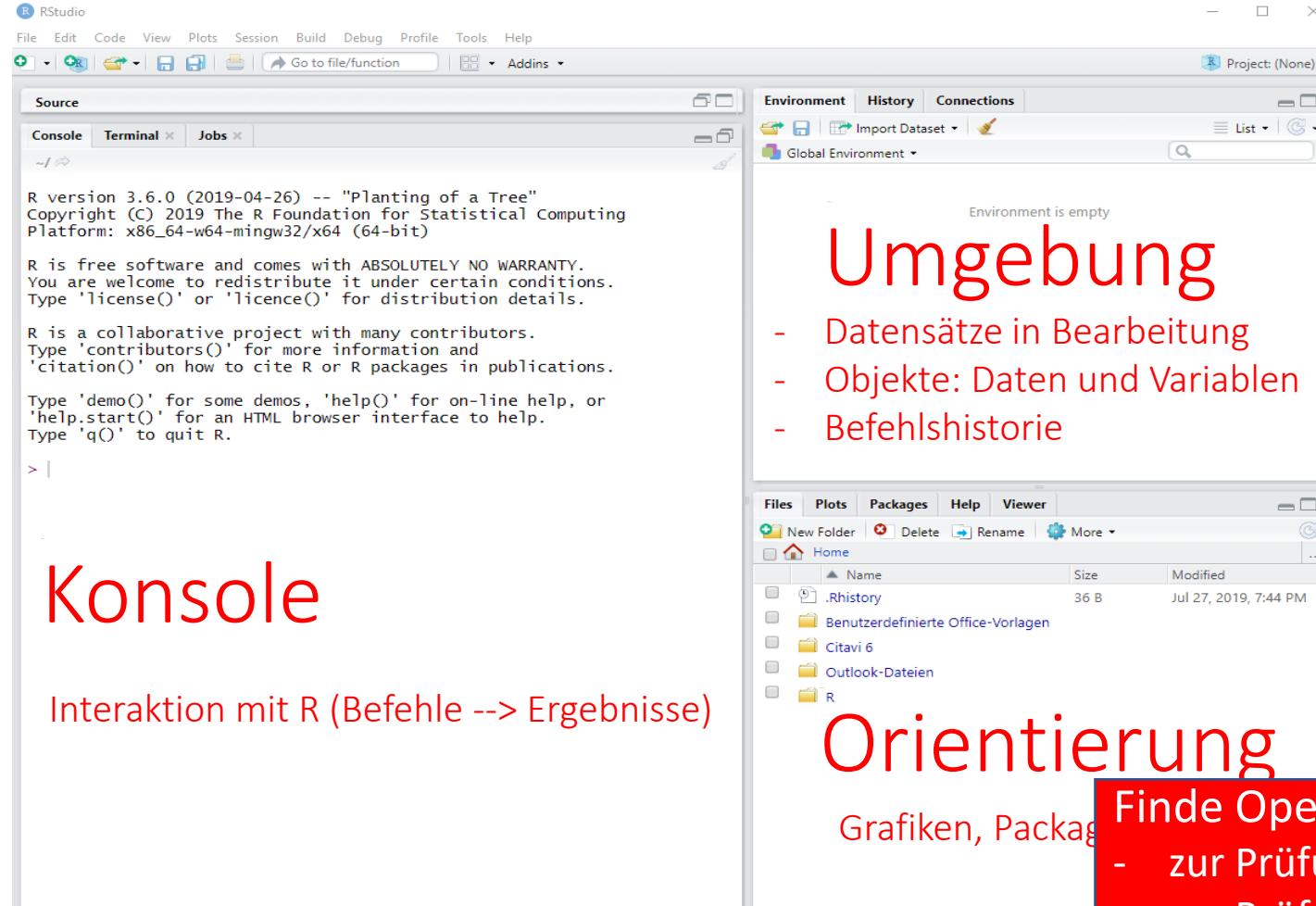
- Rekapitulation der Tutoratsinhalte (Code, Erklärungen, Interpretationen)
- Links zu der tutoratsspezifischen Literatur
- Folien
- Zusätzliche Übungsaufgaben plus Musterlösung



Teil 2: R-Basics



2.1 Starten von R-Studio



Konsole

Interaktion mit R (Befehle --> Ergebnisse)

In die Konsole schreiben:

2<8

2>8

2=8

2==8

2!=2

2!=8

2>8 | 2<8

2>8 & 2<8

Was sind dies für Operationen?

Was bedeuten «&» und «|»

Finde Operationen...

- zur Prüfung, ob 3 mal 3 gleich 9 ist
- zur Prüfung, ob sowohl 3 mal 3 gleich 9 als auch 4 mal 4 gleich 16 ist
- zur Berechnung: Quadrat von 10
- zum Ziehen der Quadratwurzel aus 81

3*3 == 9
3*3 == 9 & 4*4 == 16
10^2
sqrt(81)

2.2

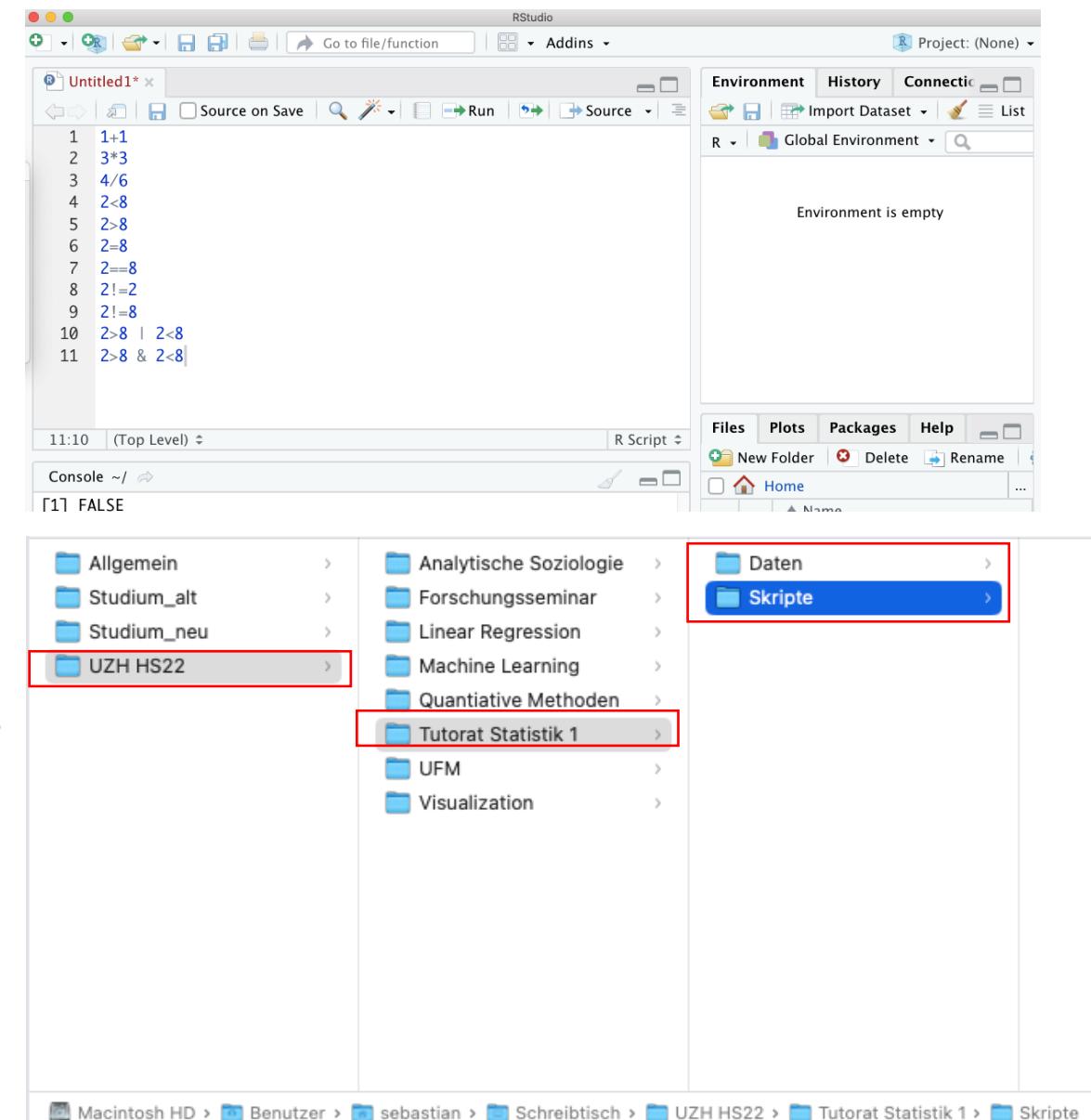
Arbeiten mit Skripten

Programmierung über die Konsole ist eher unüblich!

→ Befehlssequenzen sind nicht wiederherstellbar und kopierbar, Analysen damit nicht mehr replizierbar oder modifizierbar!

Besser: Programmierung über Skripte.

Schritt 1: Computer Set-Up / Ordnerstruktur

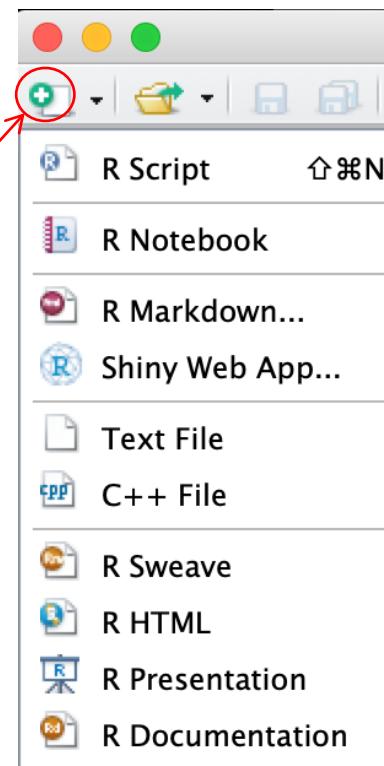


2.2 Arbeiten mit Skripten

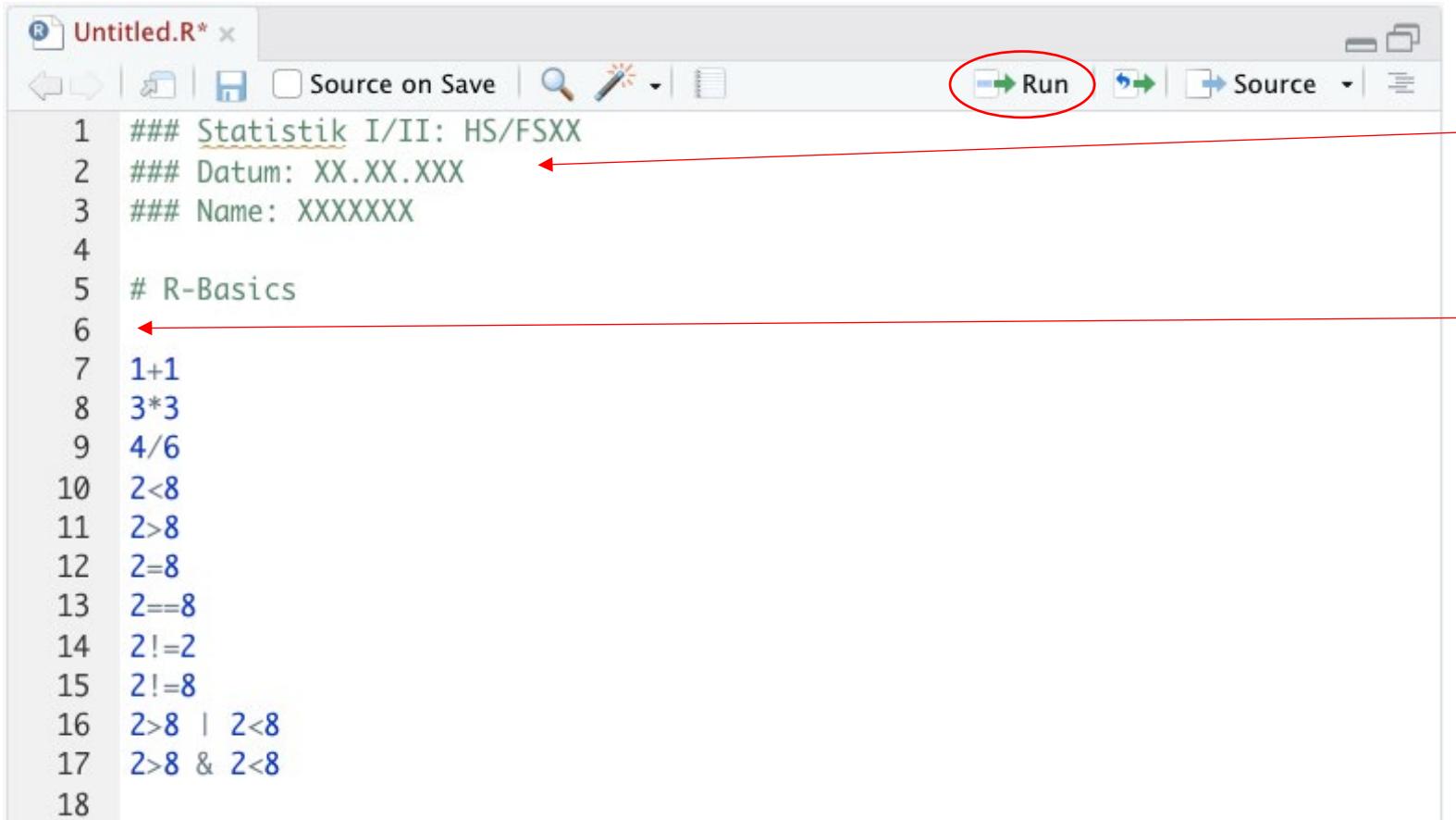
Schritt 2.) Sobald wir die Ordnerstruktur eingerichtet haben, können wir ein neues Skript anlegen...

...über File/New File/R Script

...über das Dokument-Icon



2.2 Arbeiten mit Skripten



The screenshot shows the RStudio interface with a script file named "Untitled.R". The code in the editor is:

```
1 ### Statistik I/II: HS/FSXX
2 ### Datum: XX.XX.XXX
3 ### Name: XXXXXXX
4
5 # R-Basics
6
7 1+1
8 3*3
9 4/6
10 2<8
11 2>8
12 2=8
13 2==8
14 2!=2
15 2!=8
16 2>8 | 2<8
17 2>8 & 2<8
18
```

A red arrow points from the text "laufen zu lassen:" to the line number 6. Another red arrow points from the text "Markieren und..." to the "Run" button in the toolbar, which is highlighted with a red circle.

**Wichtig: Beschriftung
des Skriptes**

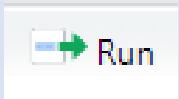
Tipp: Leerzeilen

Um einen Teil des Skripts
laufen zu lassen:

a) Markieren und...

 ⌘ + ⌂
 CTRL + ⌂

b) Oder alternativ:



2.3 R-Syntax Basics: Wir generieren einen (Phantasie-) Datensatz

Erstellen von Wertelisten bzw. (Phantasie-)Variablen

```
# Personennummer  
1:5  
  
pid <- 1:5  
  
pid  
  
c(1, 2, 3, 4, 5)  
  
pid2 <- c(1, 2, 3, 4, 5)  
  
pid2  
  
seq (from = 1, to = 5, by = 1)  
  
pid3 <- seq (from = 1, to = 5, by = 1)
```

```
# Kanton
```

```
c("ZH", "BE", "LU", "UR", "SZ")  
  
kid <- c("ZH", "BE", "LU", "UR", "SZ")  
  
kid
```

Weitere Variablen...

```
# Körpergrösse Eltern  
vg <- seq(from = 176, to = 184, by = 2)  
mg <- seq(from = 171, to = 175, by = 1)  
  
# Geburtsjahr und Befragungsjahr  
yrbrn <- 2001:2005  
date <- 2022  
  
# Monatliches Einkommen in CHF  
minc <- seq(from = 4500, to = 9000, by = 1000)
```

- Code ins Skript schreiben und aktivieren
- Beschreibe die Funktionalität von «:», «c» und «seq»
- Was macht die Anweisung «<-»
- Unterscheiden sich die Variablen pid, pid2 und pid3?
- Grundlegender Unterschied zwischen pid und kid?

2.3 R-Syntax Basics: Wir generieren einen (Phantasie-) Datensatz

Variablenklassen

```
# Attribute der Variablen  
class(pid)  
class(minc)  
class(kid)  
kid_f<-as.factor(kid)  
class(kid_f)
```

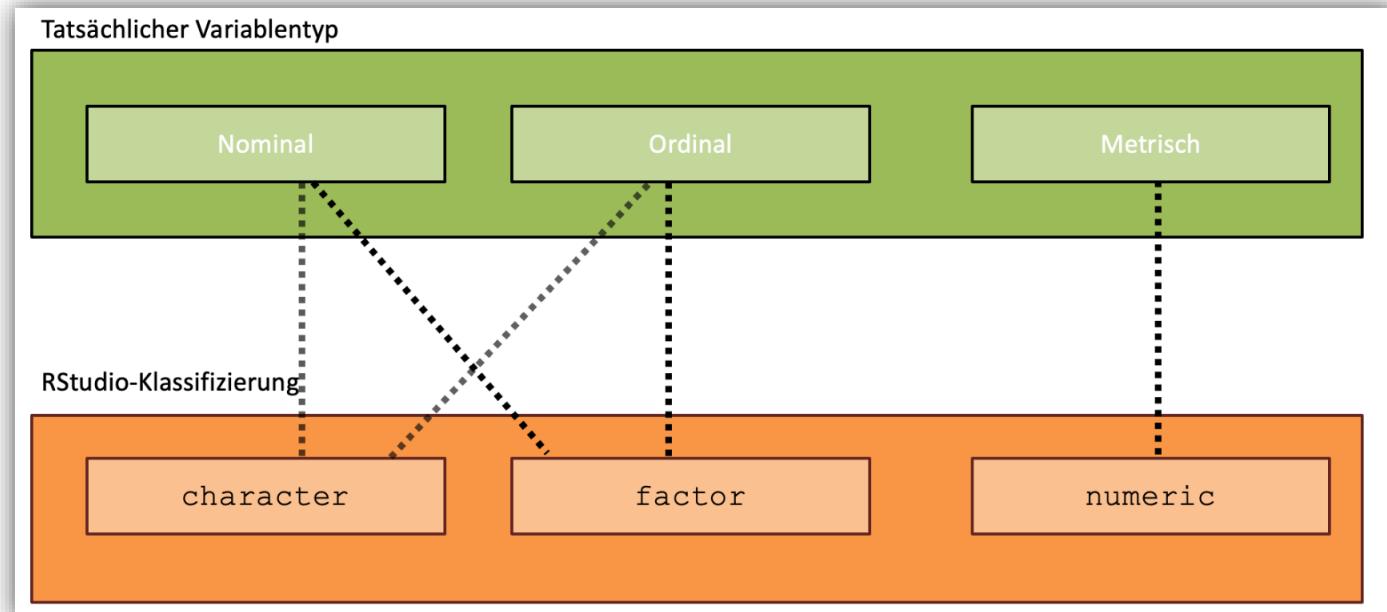
- Ins Skript schreiben und aktivieren
- Was bedeuten «integer», «numeric», «character» und «factor»?

2.3 R-Syntax Basics: Wir generieren einen (Phantasie-) Datensatz

Variablenklassen

```
# Attribute der Variablen  
class(pid)  
class(minc)  
class(kid)  
kid_f<-as.factor(kid)  
class(kid_f)
```

- Ins Skript schreiben und aktivieren
- Was bedeuten «integer», «numeric», «character» und «factor»?



- «factors» und «characters» repräsentieren beide kategoriale Variablen
- «factors» haben eine andere Speicherlogik als «characters» und sind für viele Anwendungen/Packages besser geeignet.
- «numerics» repräsentieren metrische Variablen
- «integers» sind eine Untergruppe der «numerics»

2.3 R-Syntax Basics: Wir generieren einen (Phantasie-) Datensatz

Variablenklassen

```
# Attribute der Variablen  
class(pid)  
class(minc)  
class(kid)  
kid_f<-as.factor(kid)  
class(kid_f)
```

- Ins Skript schreiben und aktivieren
- Was bedeuten «integer», «numeric», «character» und «factor»?

Rechnen mit Variablen

```
# Monatliches Einkommen in Jahreseinkommen  
minc * 12  
yinc <- minc * 12  
yinc  
  
# Alter zum Zeitpunkt der Befragung  
age <- date - yrbrn  
age  
  
# Durchschnittliche elterliche Körpergrösse  
pg <- (vg + mg)/2  
pg
```

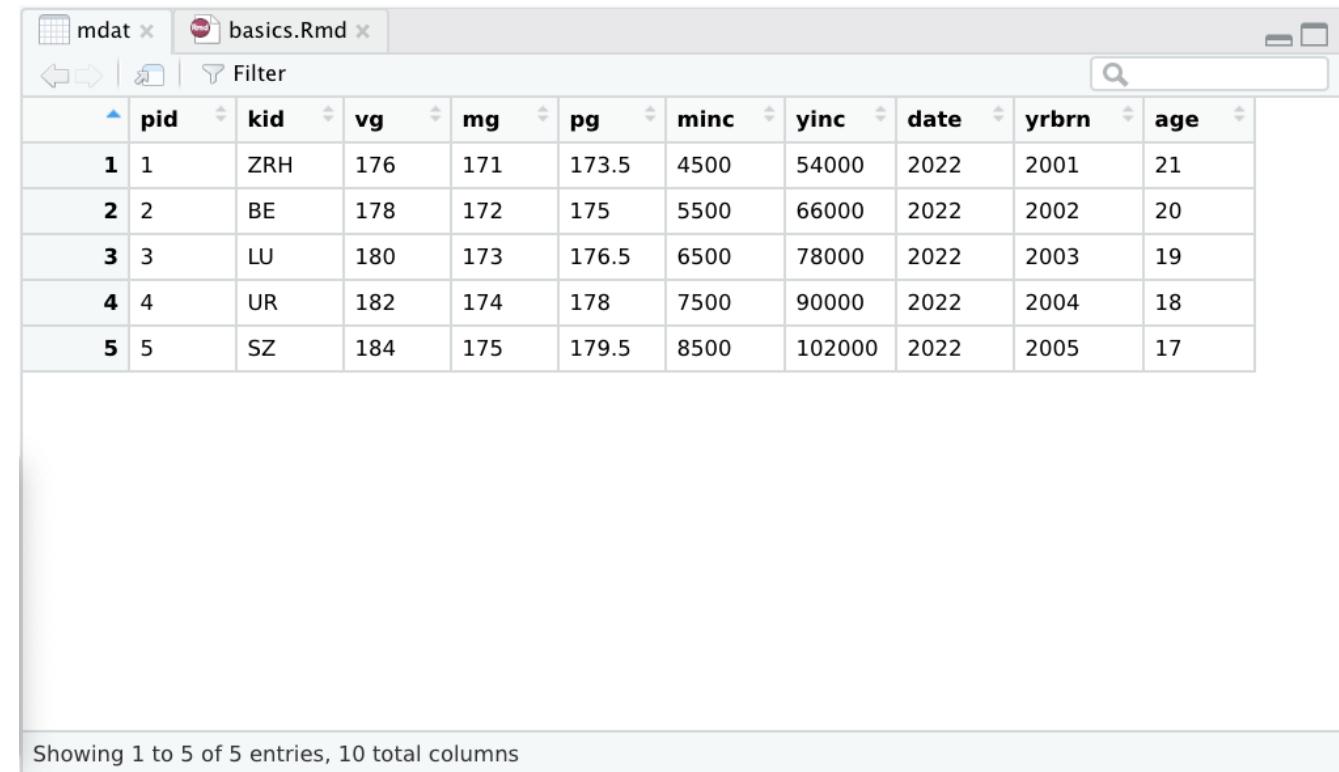
Wie verrechnet R Variablen?

2.3 R-Syntax Basics

Verbindung von Variablen zu Datenmatrizen

```
m <- cbind(pid, kid, vg, mg, pg, minc, yinc, date, yrbrn, age)
class(m)
mdat <- as.data.frame(m)
class(mdat)
View(mdat)
```

- Ins Skript schreiben und aktivieren
- Wie wurde hier Information kombiniert?
- Beschreibe die Datenmatrix: Welche Eigenschaften hat die Person mit pid=1



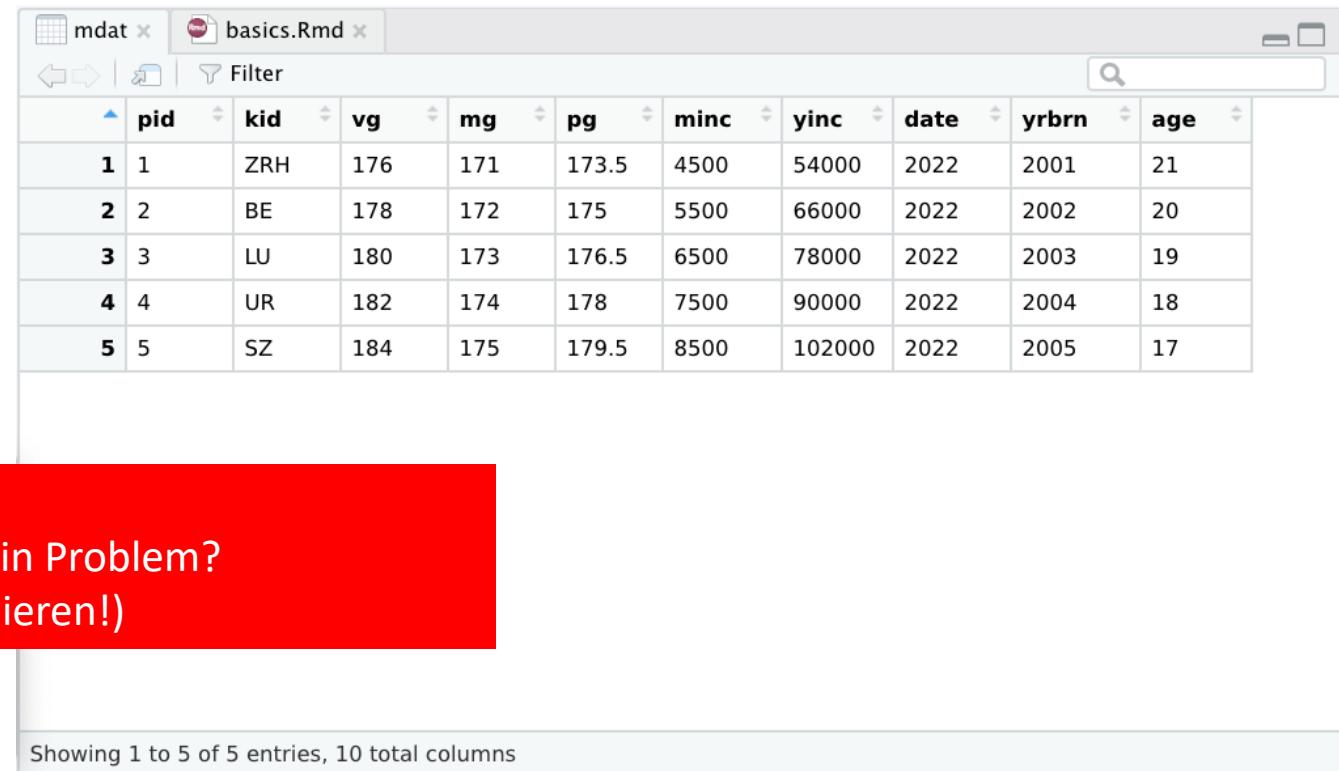
	pid	kid	vg	mg	pg	minc	yinc	date	yrbrn	age
1	1	ZRH	176	171	173.5	4500	54000	2022	2001	21
2	2	BE	178	172	175	5500	66000	2022	2002	20
3	3	LU	180	173	176.5	6500	78000	2022	2003	19
4	4	UR	182	174	178	7500	90000	2022	2004	18
5	5	SZ	184	175	179.5	8500	102000	2022	2005	17

Showing 1 to 5 of 5 entries, 10 total columns

2.3 R-Syntax Basics

Verbindung von Variablen zu Datenmatrizen

```
m <- cbind(pid, kid, vg, mg, pg, minc, yinc, date, yrbrn, age)
class(m)
mdat <- as.data.frame(m)
class(mdat)
View(mdat)
```



	pid	kid	vg	mg	pg	minc	yinc	date	yrbrn	age
1	1	ZRH	176	171	173.5	4500	54000	2022	2001	21
2	2	BE	178	172	175	5500	66000	2022	2002	20
3	3	LU	180	173	176.5	6500	78000	2022	2003	19
4	4	UR	182	174	178	7500	90000	2022	2004	18
5	5	SZ	184	175	179.5	8500	102000	2022	2005	17

Showing 1 to 5 of 5 entries, 10 total columns

Variablenmanagement in der Datenmatrix

```
class (mdat$minc)
mean (mdat$minc)
```

- Was macht «\$»?
- Was ist mit der Variable «minc» passiert, warum ist das ein Problem?
- Wie können wir sie wieder «numerisieren? (ggf. recherchieren!)

Meine Variable mdat\$inc in R soll numerisch sein, ist aber ein Faktor. Was nun?

Überlegt, wie man einen Faktor in eine Zahl umwandelt für einige wenige Sekunden >

In R solltest du den Faktor nicht direkt in numerisch umwandeln, da das zu den internen Integer-Codes führt. Stattdessen wandelst du zuerst in einen Character um und dann in numerisch. Beispielsweise so:

```
r
```

 Kopieren

```
# Schritt 1: Faktor in Character umwandeln  
mdat$inc <- as.character(mdat$inc)  
  
# Schritt 2: Character in numerisch umwandeln  
mdat$inc <- as.numeric(mdat$inc)
```

Dadurch wird zuerst der Faktor in einen Character konvertiert und anschließend in einen numerischen Vektor umgewandelt.

2.3 R-Syntax Basics

Verbindung von Variablen zu Datenmatrizen

```
m <- cbind(pid, kid, vg, mg, pg, minc, yinc, date, yrbrn, age)
class(m)
mdat <- as.data.frame(m)
class(mdat)
View(mdat)
```

	pid	kid	vg	mg	pg	minc	yinc	date	yrbrn	age
1	1	ZRH	176	171	173.5	4500	54000	2022	2001	21
2	2	BE	178	172	175	5500	66000	2022	2002	20
3	3	LU	180	173	176.5	6500	78000	2022	2003	19
4	4	UR	182	174	178	7500	90000	2022	2004	18
5	5	SZ	184	175	179.5	8500	102000	2022	2005	17

Variablenmanagement in der Datenmatrix

```
class (mdat$minc)
mean (mdat$minc)
```

- Was macht «\$»?
- Was ist mit der Variable «minc» passiert, warum ist das ein Problem?
- Wie können wir sie wieder «numerisieren»? (ggf. recherchieren!)

```
# Notwendiger Zwischenschritt, nur falls Variable als Faktor angelegt wurde:
# mdat$minc<-as.character(mdat$minc)
mdat$minc<-as.numeric(mdat$minc)
class (mdat$minc)
mean (mdat$minc)
```

2.3 R-Syntax Basics

Variablenmanagement in der Datenmatrix

```
# Generiere Stundenlohn aus Monatslohn  
mdat$hour_wage <- mdat$minc/160  
mdat$hour_wage  
View(mdat)
```

- Beschreibe die Funktion des Codes
- Bilde nun nach gleichem Muster eine Variable, die die elterliche Durchschnittsgrösse in Zoll angibt (*Umrechnungsquotient: 2.54*)

```
# Generiere Grösse in Zoll aus Grösse in Zentimetern  
mdat$pg <- as.character(mdat$pg)  
mdat$pg <- as.numeric(mdat$pg)  
mdat$pgz <- mdat$pg/2.54  
mdat$pgz|  
View(mdat)
```

	pid	kid	vg	mg	pg	minc	yinc	date	yrbrn	age	hour_wage	pgi	pgz
1	1	ZH	176	171	173.5	4500	54000	2022	2001	21	28.125	68.30709	5.692257
2	2	BE	178	172	175.0	5500	66000	2022	2002	20	34.375	68.89764	5.741470
3	3	LU	180	173	176.5	6500	78000	2022	2003	19	40.625	69.48819	5.790682
4	4	UR	182	174	178.0	7500	90000	2022	2004	18	46.875	70.07874	5.839895
5	5	SZ	184	175	179.5	8500	102000	2022	2005	17	53.125	70.66929	5.889108

Showing 1 to 5 of 5 entries, 13 total columns

2.3 R-Syntax Basics

Codeverschachtelungen in R

Hier: 3 sequentielle Arbeitsschritte, die sich auf eine Variable beziehen

```
# Grösse in Zoll  
mdat$pg <- as.character(mdat$pg)  
mdat$pg <- as.numeric(as.character(mdat$pg))  
mdat$pgi <- mdat$pg/2.54
```

Solche Befehlssequenzen *müssen nicht*, aber *können* effizienter – nämlich in einer Codezeile – dargestellt werden.

Möglichkeit 1:
Verschachtelung mit Klammern
Erläutere die Logik des Codes!

```
# Grösse in Zoll: Alternative Codeorganisation "Klammer"  
mdat$pgi <- as.numeric(as.character(mdat$pg))/2.54
```

Möglichkeit 2:
Verschachtelung mit «Pipe»-Operator
Erläutere die Logik des Codes!

```
# Grösse in Zoll: Alternative Codeorganisation "Pipe"  
library(tidyverse)  
mdat$pgi <- as.character(mdat$pg) %>% as.numeric %>% {. / 2.54}
```

Alle drei Varianten führen zum selben Ergebnis und sind korrekt!

Ab hier nächste Woche weiter

2.5 Einlesen der Daten (<https://ess.sikt.no/en/?tab=overview>)

ESS Data Portal Log in  Menu 

ESS Data Portal

- Search, download or visualize data from the European Social Survey.
- Access data from **EOSC Future Project** and **CRoss-National Online Survey (CRONOS)**.
- Create your own datasets using the **Datafile Builder (Wizard)**.

Search for ESS data e.g. trust politicians, election 

European Social Survey 2002 - 2022

Search and download European Social Survey data for 18,000 questions and variables contained in 60 downloadable data files. This service is a work in progress, to improve your access to ESS data.

Overview **Datafile Builder (Wizard)**

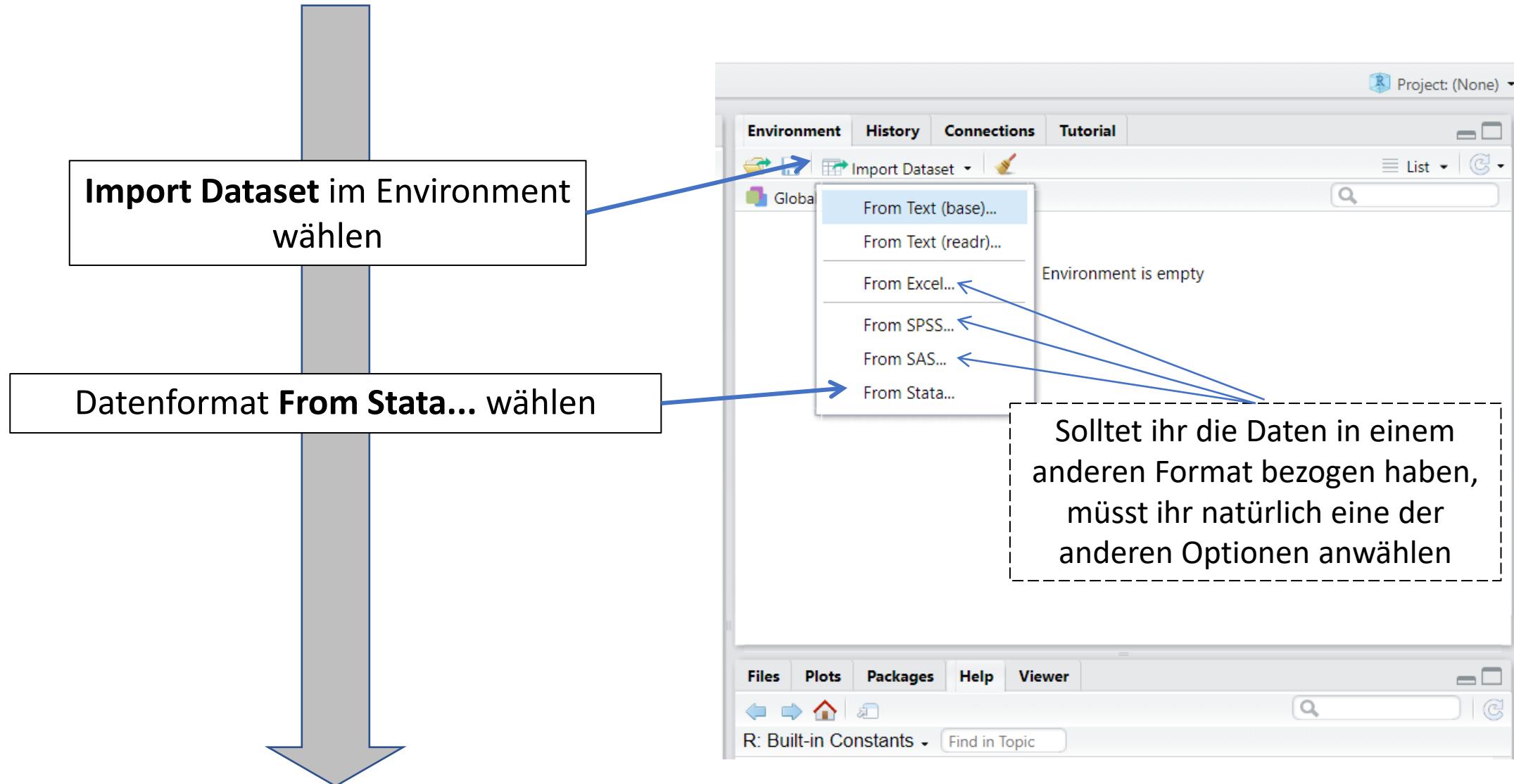
ESS round 10 - 2020. Democracy, Digital social contacts 

ESS round 9 - 2018. Timing of life, Justice and fairness 

ESS round 8 - 2016. Welfare attitudes, Attitudes to climate change  

2.5

Einlesen der Daten



2.5 Einlesen der Daten

Import Statistical Data

File/URL:
C:/Daten/ESS/ESS8e02_2.dta

Data Preview:

name	essround	edition	proddate	idno	cntry	nwspol	netusoft
Title of dataset	ESS round	Edition	Production date	Respondent's identification number	Country	News about politics and current affairs, watching, reading or listen...	Internet use,
ESS8e02_2	8	2.2	10.12.2020	1	AT	120	4
ESS8e02_2	8	2.2	10.12.2020	2	AT	120	5
ESS8e02_2	8	2.2	10.12.2020	4	AT	30	2
ESS8e02_2	8	2.2	10.12.2020	6	AT	30	5
ESS8e02_2	8	2.2	10.12.2020	10	AT	30	5
ESS8e02_2	8	2.2	10.12.2020	11	AT	60	5
ESS8e02_2	8	2.2	10.12.2020	12	AT	15	2
ESS8e02_2	8	2.2	10.12.2020	13	AT	45	4

Previewing first 50 entries.

Import Options:

Name: ESS8e02_2
Model:
Format: DTA Open Data Viewer

Code Preview:

```
library(haven)
ESS8e02_2 <- read_dta("c:/Daten/ESS/ESS8e02_2.dta")
View(ESS8e02_2)
```

Import Cancel

Navigiere zum Datenordner und wähle den Datensatz

R erkennt die Datenstruktur im Ausgangsfile (...kann aber ggf. korrigiert werden)

R übersetzt Importvorgang in Befehlssyntax

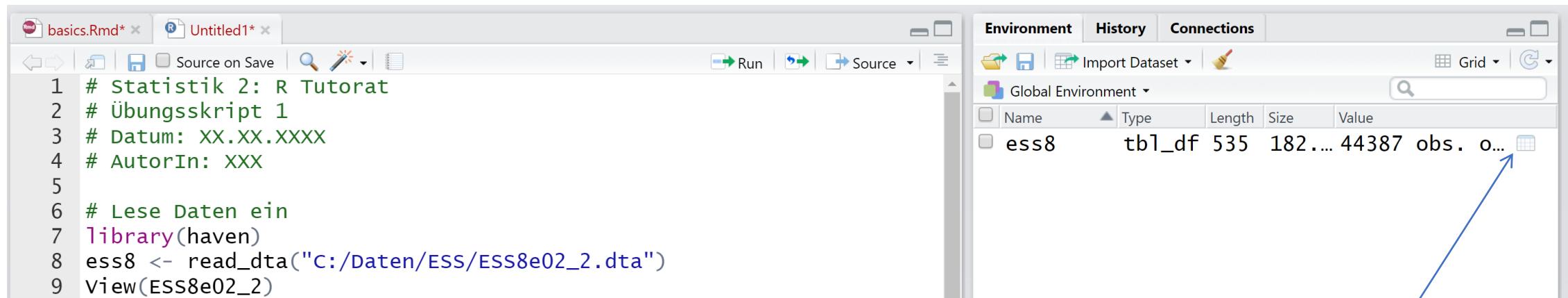
Die Befehlssyntax sollte dann per **copy-paste** direkt ins Skript übertragen werden.

```
basics.Rmd* Untitled1* 5 # Lese Daten ein 6 library(haven) 7 ess8 <- read_dta("C:/Daten/ESS/ESS8e02_2.dta") 8 View(ess8) 9
```

Hier könnt Ihr dann weitere Änderungen vornehmen (z.B. **Name ändern**, Pfad auf WD, etc.)

...und schliesslich den Code aktivieren

2.6 Dateninspektion



The screenshot shows the RStudio interface. In the top-left corner, there are two tabs: 'basics.Rmd*' and 'Untitled1*'. Below the tabs are standard file operations like Open, Save, and Print. The main area contains R code:

```
1 # Statistik 2: R Tutorat
2 # Übungsskript 1
3 # Datum: XX.XX.XXXX
4 # AutorIn: XXX
5
6 # Lese Daten ein
7 library(haven)
8 ess8 <- read_dta("C:/Daten/ESS/ESS8e02_2.dta")
9 View(ESS8e02_2)
```

To the right of the code editor is the 'Environment' tab of the global environment pane. It lists a single object:

Name	Type	Length	Size	Value
ess8	tbl_df	535	182...	44387 obs. o...

A blue arrow points from the 'View' command in the code editor to the 'View' button in the environment pane. Another blue arrow points from the 'Value' column of the 'ess8' row in the environment pane to the matrix icon in the same row.

Visuelle Inspektion der Datenmatrix entweder mit «View»...

oder durch Klick auf das Matrixsymbol
zum Datensatz im Environment

2.5 Dateninspektion

(a) Checkliste: Visuelle Inspektion

- Matrix ok (*tidydata*)?

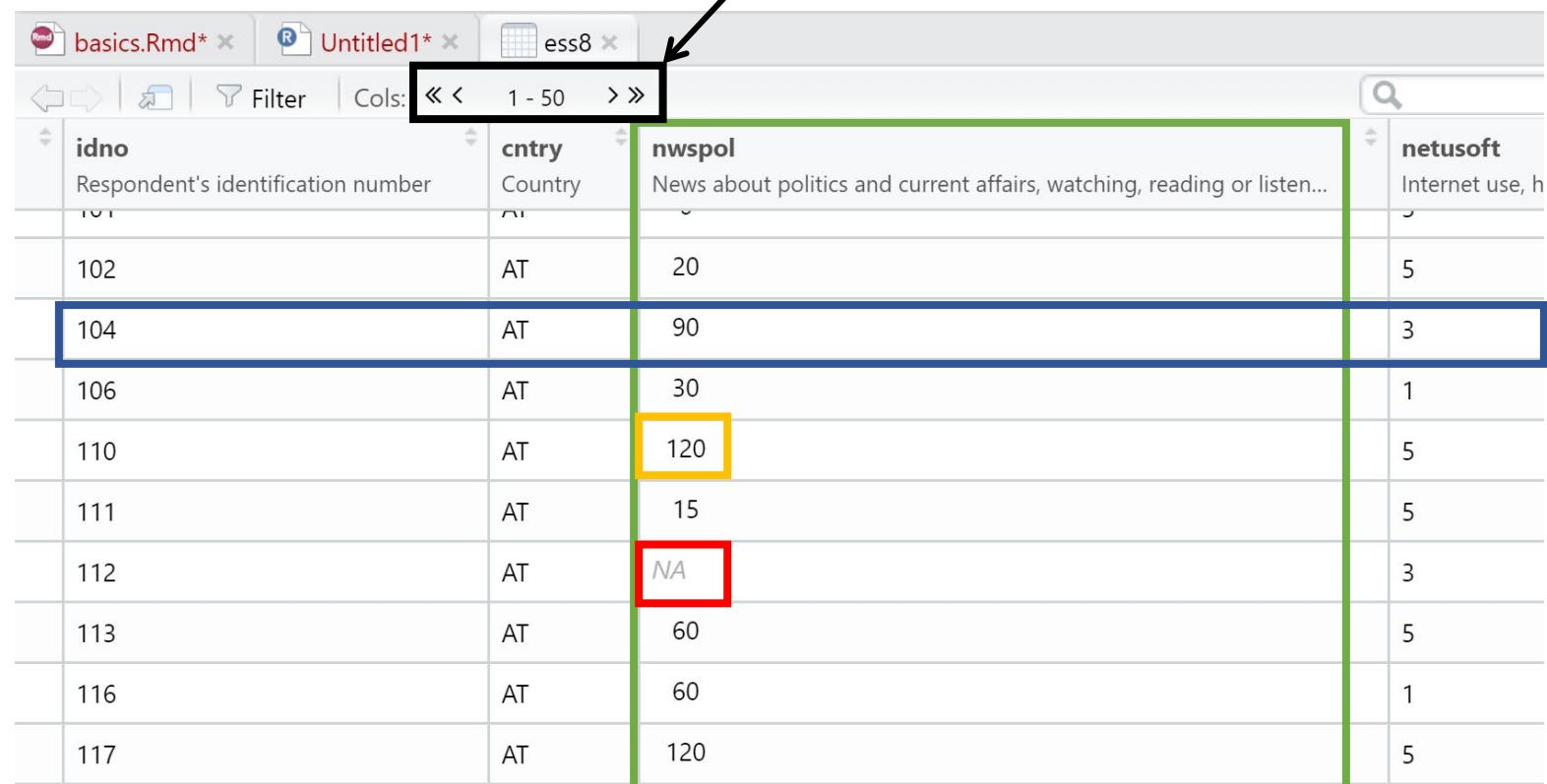
→ Merkmale in Spalten

→ Merkmalsträger in Zeilen

→ Werte in Zellen

→ Fehlende Werte == «NA»?

Achtung: in der Matrixansicht sind die Variablen in 50er Blöcken organisiert. Die Pfeilsymbole oben dienen der Navigation



idno	cntry	nwspol	netusoft
101	DE	10	5
102	AT	20	5
104	AT	90	3
106	AT	30	1
110	AT	120	5
111	AT	15	5
112	AT	NA	3
113	AT	60	5
116	AT	60	1
117	AT	120	5

(b) Systematische Inspektion

```
12 # Anzahl Fälle und Variablen  
13 dim(ess8)  
[1] 44387 535
```

2.5 Dateninspektion

(c) Variablensuche

- Finde im ESS eine Variable, welche die Zufriedenheit mit der nationalen Regierung misst. Beschreibe sowohl deine Suchstrategie als auch die Verteilung der Variable (z.B. mit «summary()»)
- Finde im ESS eine geeignete Variable zur Messung von Fremdenfeindlichkeit. Beschreibe deine Schwierigkeiten bei der Suche nach der Variable

2.6 Dateninspektion

(c) Variablersuche

Problem: Das ESS enthält über 500 Variablen. Wie finde ich die für mich relevanten Merkmale?

- Möglichkeit 1: Suche im Codebook auf der HP zum ESS
- Möglichkeit 2: Datenorientierte Suche, z.B. mit „look_for()“

```
16 install.packages("labelled")
17 library(labelled)
18 # Generiere Codebook
19 varlist <- look_for(ess8)
20 view(varlist)
```

Welche Informationen enthält die mit «look_for» generierte Variablenliste?

2.6 Dateninspektion

(c) Variablersuche

Welche Informationen enthält die mit «look_for» generierte Variablenliste?



A screenshot of the RStudio interface showing a data frame named 'varlist'. The table has columns: pos, variable, label, col_type, missing, levels, and value_labels. The 'value_labels' column contains character vectors representing survey responses. A red arrow points from a text box containing a tip to the search bar.

pos	variable	label	col_type	missing	levels	value_labels
6	cntry	Country	chr	0	NULL	NULL
7	nwspol	News about politics and current affairs, watching, reading or...	dbl+lbl	524	NULL	c(Refusal = NA, 'Don't know' = NA, 'No answer' = N [...])
8	netusoft	Internet use, how often	dbl+lbl	49	NULL	c(Never = 1, 'Only occasionally' = 2, 'A few times [...]')
9	netustm	Internet use, how much time on typical day, in minutes	dbl+lbl	14274	NULL	c('Not applicable' = NA, Refusal = NA, 'Don't know [...]')
10	ppltrst	Most people can be trusted or you can't be too careful	dbl+lbl	115	NULL	c('You can't be too careful' = 0, '1' = 1, '2' = 2 [...])
11	pplfair	Most people try to take advantage of you, or try to be fair	dbl+lbl	315	NULL	c('Most people try to take advantage of me' = 0, ' [...]')

Tipp: Mit dem Lupen-Icon-Feld nach spezifischen Variablen suchen..

- Finde eine geeignete Variable zur Messung von Fremdenfeindlichkeit. Untersuche die Verteilungseigenschaften dieser Variable (z.B. mit «summary()»)
- Finde eine geeignete Variable zur Messung des Zivilstandes. Untersuche auch die Häufigkeitsverteilung dieser Variable (z.B. mit «table()»)

2.6 Dateninspektion

(d) Variableninspektion

```
> # Variableninspektion "immigration bad"
> class (ess8$imbgeco)
[1] "haven_labelled" "vctrs_vctr"      "double"
> summary (ess8$imbgeco)
   Min. 1st Qu. Median      Mean 3rd Qu.      Max.    NA's
0.000  3.000  5.000  5.006  7.000 10.000

```

- Plausible Werte, Variable ok?

2.6 Dateninspektion

(d) Variableninspektion **marsts**

```
> class (ess8$marsts)
[1] "haven_labelled" "vctrs_vctr"     "double"
> table(as_factor(ess8$marsts))
```

	Legally married	813
	In a legally registered civil union	196
	Legally separated	258
	Legally divorced/civil union dissolved	4246
	widowed/civil partner died	3756
None of these (NEVER married or in legally registered civil union)		13040
	Not applicable	21213
	Refusal	215
	Don't know	54
	No answer	596

- Plausible Werte, Variable ok?

- Antwort: nein.

- Diese Variable bezieht sich nur auf Personen, die nicht mit einer Partnerin im Haushalt zusammenleben, siehe Codebook auf der HP:

marsts - Legal marital status	
Type	Code
Location	F11
Pre-Question Text	ASK IF NOT LIVING WITH A HUSBAND/WIFE/PARTNER OR ARE COHABITING CARD 60

Achtung: Manchmal befinden sich Phantomvariablen im Datensatz, die ihr meist über einfache Inspektion der Verteilung identifizieren könnt. Für Eure mündl. Prüfung solltet Ihr zudem alle verwendeten Variablen zusätzlich auf Basis des ESS-Codebooks checken!

2.6 Dateninspektion

(d) Variableninspektion maritalb

```
> class (ess8$maritalb)
[1] "haven_labelled" "vctrs_vctr"      "double"
> table(as_factor(ess8$maritalb))
```

Category	Count
Legally married	21711
In a legally registered civil union	443
Legally separated	648
Legally divorced/civil union dissolved	3912
Widowed/Civil partner died	3756
None of these (NEVER married or in legally registered civil union)	13039
Refusal	229
Don't know	61
No answer	588

- Plausible Werte, Variable ok?

- Antwort: ja.

- Diese Variable bezieht sich sowohl auf Personen, die mit, als auch solche, die ohne eine Partnerin im Haushalt zusammenleben. Also alle, siehe Codebook:

maritalb - Legal marital status, post coded	
Type	Code
Location	F11b
Question	POST CODE: MARITAL STATUS
Note	Post coded variable based on F6 (RSHPSTS) and F11 (MARSTS).

Aufgabe für nächste Woche

- Falls ihr merkt, dass ihr irgendwo noch unsicher seid, dann schaut euch nochmals die Seiten zur Statistik 1 an. (<http://www.suz.uzh.ch/dataforstat/>)
- Danke fürs Mitmachen uns bis nächste Woche! ☺