

---

# Empirisch-experimentelle Forschungsmethoden in der Anwendung

Seminar

---

# Deskriptive Statistik und Boxplot

Von den Daten zu Kennzahlen und Diagrammen

# Themen heute:

---

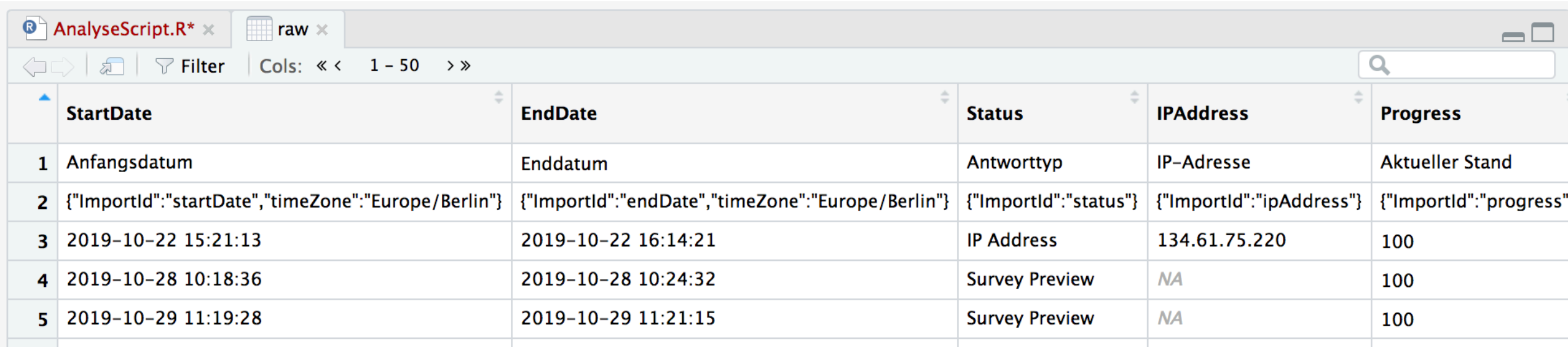
- Rückblick auf Data Cleaning
- Deskriptive Statistik
  - Lagemaße
  - Streumaße
  - Interpretation
- Erste Plots mit ggplot2

# Kurzer Rückblick

- AnalyseScript.R
  - Bibliotheken laden
  - erhobene Daten laden
  - Daten cleanen
  - Skalen berechnen
  - (Auswertung)
  - (Grafiken)

```
1 # Analyse Skript
2
3 ##### Bibliotheken laden
4 # install.packages("tidyverse")
5 # install.packages("psych")
6
7 library(tidyverse)
8 source("qualtrichelpers.R")
9
```

# Warum Data Cleaning?



	StartDate	EndDate	Status	IPAddress	Progress
1	Anfangsdatum	Enddatum	Antworttyp	IP-Adresse	Aktueller Stand
2	{"ImportId":"startDate","timeZone":"Europe/Berlin"}	{"ImportId":"endDate","timeZone":"Europe/Berlin"}	{"ImportId":"status"}	{"ImportId":"ipAddress"}	{"ImportId":"progress"}
3	2019-10-22 15:21:13	2019-10-22 16:14:21	IP Address	134.61.75.220	100
4	2019-10-28 10:18:36	2019-10-28 10:24:32	Survey Preview	NA	100
5	2019-10-29 11:19:28	2019-10-29 11:21:15	Survey Preview	NA	100

- .csv-Datei ist von Qualtrics erstellt
- Noch kein gutes Format für uns:
  - Überflüssige Spalten
  - Obere zwei Spalten: Mischung aus Fragetext, Variablenname und Itemtext

qualtrics-export.csv

# Warum Data Cleaning?

qualtrics-export.csv

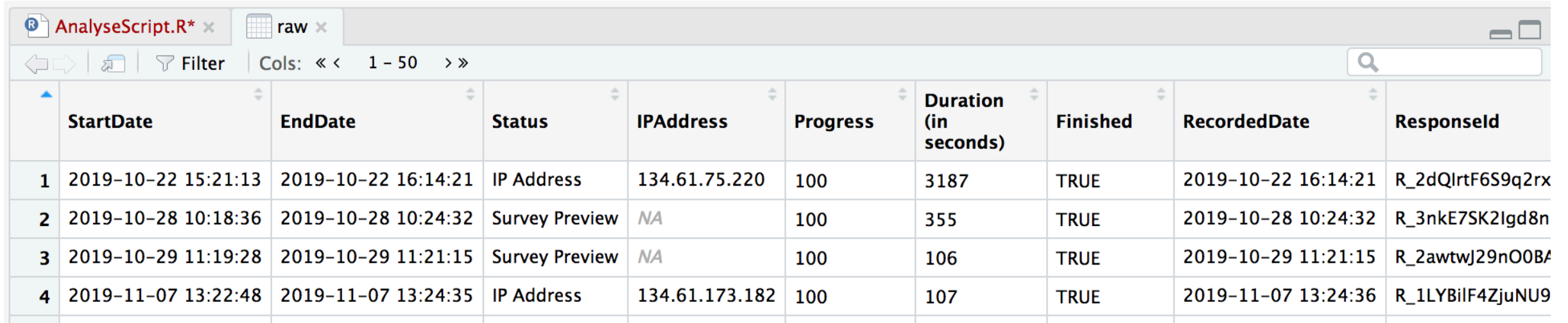
	StartDate	EndDate	Status	IPAddress	Progress
1	Anfangsdatum	Enddatum	Antworttyp	IP-Adresse	Aktueller Stand
2	{"ImportId":"startDate","timeZone":"Europe/Berlin"}	{"ImportId":"endDate","timeZone":"Europe/Berlin"}	{"ImportId":"status"}	{"ImportId":"ipAddress"}	{"ImportId":"progress"}
3	2019-10-22 15:21:13	2019-10-22 16:14:21	IP Address	134.61.75.220	100

```
raw <- load_qualtrics_csv(„data/qualtrics-export.csv“)
```

	StartDate	EndDate	Status	IPAddress	Progress	Duration (in seconds)	Finished	RecordedDate	ResponseId
1	2019-10-22 15:21:13	2019-10-22 16:14:21	IP Address	134.61.75.220	100	3187	TRUE	2019-10-22 16:14:21	R_2dQlrtF6S9q2rx
2	2019-10-28 10:18:36	2019-10-28 10:24:32	Survey Preview	NA	100	355	TRUE	2019-10-28 10:24:32	R_3nkE7SK2lgd8n
3	2019-10-29 11:19:28	2019-10-29 11:21:15	Survey Preview	NA	100	106	TRUE	2019-10-29 11:21:15	R_2awtwj29nO0BA
4	2019-11-07 13:22:48	2019-11-07 13:24:35	IP Address	134.61.173.182	100	107	TRUE	2019-11-07 13:24:36	R_1LYBilF4ZjuNU9

raw

# Warum Data Cleaning?



	StartDate	EndDate	Status	IPAddress	Progress	Duration (in seconds)	Finished	RecordedDate	Responseld
1	2019-10-22 15:21:13	2019-10-22 16:14:21	IP Address	134.61.75.220	100	3187	TRUE	2019-10-22 16:14:21	R_2dQlrtF6S9q2rx
2	2019-10-28 10:18:36	2019-10-28 10:24:32	Survey Preview	NA	100	355	TRUE	2019-10-28 10:24:32	R_3nkE7SK2lgd8n
3	2019-10-29 11:19:28	2019-10-29 11:21:15	Survey Preview	NA	100	106	TRUE	2019-10-29 11:21:15	R_2awtwJ29nO0B/
4	2019-11-07 13:22:48	2019-11-07 13:24:35	IP Address	134.61.173.182	100	107	TRUE	2019-11-07 13:24:36	R_1LYBiIF4ZjuNU9

- Besser: Jede Zeile entspricht einem Probanden
- Aber: Immer noch zu viele Spalten
- Variablennamen sind zu lang und nicht aussagekräftig

raw

# Warum Data Cleaning?

- Überflüssige Spalten entfernen:

```
raw <- raw[,c(-1:-17, -22:-28, -58:-118,  
-121:-125)]
```

*oder*

```
raw.short <- raw[,c(-1:-17, -22:-28,  
-58:-118, -121:-125)]
```

- Variablen umbenennen:

```
names(raw.short)[4] <- "ati_1"  
usw.
```

	gender	age	edu1	edu2	ATI_1	ATI_2
1	weiblich	24	Abitur/Fachabitur	Hochschulabschluss	NA	NA
2	männlich	29	Abitur/Fachabitur	Hochschulabschluss	Stimme zu	Stimme zu
3	männlich	20	Haupt- / Volksschulsabschluss	noch keine Ausbildung	Stimme gar nicht zu	Stimme nicht zu
4	männlich	34	NA	Hochschulabschluss	Stimme eher nicht zu	Stimme eher nicht zu

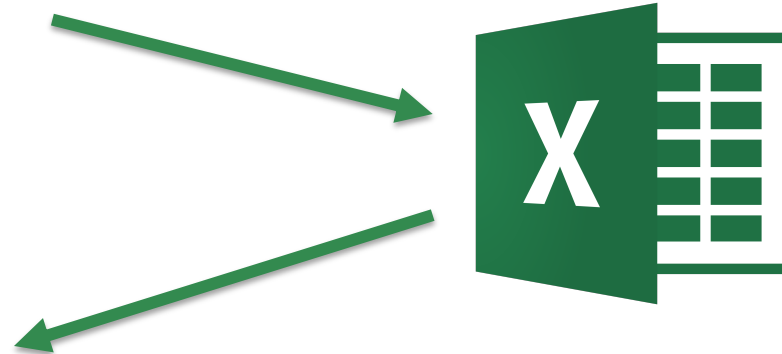
raw.short



# Codebook

- Elegantere Variante, um die Variablen umzubenennen:

```
generate_codebook(raw.short, filename, "data/codebook.csv")
```



```
codebook <- read_codebook("data/codebook_final.csv")
```

# Codebook\_final.csv

	variable	variable_old	text	info
1	gender	gender	Bitte geben Sie Ihr Geschlecht an:	{"ImportId":"QID4"}
2	age	age	Bitte geben Sie Ihr Alter in Jahren an:	{"ImportId":"QID5_TEXT"}
3	edu1	edu1	Bitte geben Sie Ihren höchsten Schulabschluss an:	{"ImportId":"QID6"}
4	edu2	edu2	Bitte geben Sie Ihren höchsten Ausbildungsabschluss ...	{"ImportId":"QID7"}
5	ati_1	ATI_1	Im Folgenden geht es um "technische Systeme" im All...	{"ImportId":"QID2_1"}
6	ati_2	ATI_2	Im Folgenden geht es um "technische Systeme" im All...	{"ImportId":"QID2_2"}

- Schlüsselt Variablennamen gegen den Fragetext auf.
- Nächster Schritt: Variablenname für raw.short übernehmen:  
`names(raw.short) <- codebook$variable`

codebook

# Zuweisung der Datentypen: as.factor

	gender	age	edu1	edu2	ati_1	ati_2
1	weiblich	24	Abitur/Fachabitur	Hochschulabschluss	NA	NA
2	männlich	29	Abitur/Fachabitur	Hochschulabschluss	Stimme zu	Stimme zu
3	männlich	20	Haupt-/ Volksschulsabschluss	noch keine Ausbildung	Stimme gar nicht zu	Stimme nicht zu
4	männlich	34	NA	Hochschulabschluss	Stimme eher nicht zu	Stimme eher nicht zu
5	männlich	22	(noch) kein Schulabschluss	noch keine Ausbildung	Stimme gar nicht zu	Stimme gar nicht zu
6	männlich	23	Realschulabschluss	Berufsausbildung	Stimme eher nicht zu	Stimme nicht zu

raw.short

- Variablennamen sind jetzt „selbsterklärend“
- Nächster Schritt: Zuweisung der Datentypen, falls Variable kategorial:

```
raw.short$gender <- as.factor(raw.short$gender)
```

# Zuweisung der Datentypen: ordered

	gender	age	edu1	edu2	ati_1	ati_2
1	weiblich	24	Abitur/Fachabitur	Hochschulabschluss	NA	NA
2	männlich	29	Abitur/Fachabitur	Hochschulabschluss	Stimme zu	Stimme zu
3	männlich	20	Haupt-/ Volksschulsabschluss	noch keine Ausbildung	Stimme gar nicht zu	Stimme nicht zu
4	männlich	34	NA	Hochschulabschluss	Stimme eher nicht zu	Stimme eher nicht zu

- Zuweisung der Datentypen, falls Variable ordinal:

```
scale.zustimmung <-c("Stimme gar nicht zu",  
  "Stimme nicht zu",  
  "Stimme eher nicht zu",  
  "Stimme eher zu",  
  "Stimme zu",  
  "Stimme völlig zu")
```

```
raw.short$ati_1 <- ordered(raw.short$ati_1, levels = scale.zustimmung)
```

```
raw.short$ati_2 <- ordered(raw.short$ati_2, levels = scale.zustimmung)
```

raw.short

# Challenge: Verschiedene Skalen

---

```
scale.zustimmung <-c("Stimme gar nicht zu",  
  "Stimme nicht zu",  
  "Stimme eher nicht zu",  
  "Stimme eher zu",  
  "Stimme zu",  
  "Stimme völlig zu")
```

```
scale.zutreffen <-c("trifft gar nicht zu",  
  "trifft nicht zu",  
  "trifft eher nicht zu",  
  "trifft eher zu",  
  "trifft zu",  
  "trifft völlig zu")
```

```
scale.zustimmung2 <-c("Stimme gar nicht zu",  
  "Stimme nicht zu",  
  "Stimme eher nicht zu",  
  "Stimme eher zu",  
  "Stimme zu",  
  "Stimme sehr zu")
```

```
scale.gerne <-c("Auf keinen Fall",  
  "ungerne",  
  "eher ungerne",  
  "eher gerne",  
  "gerne",  
  "Sehr gerne")
```

# Skalenbildung

```
schluesselliste <- list(ATI = c("ati_1", "ati_2", "-ati_3", "ati_4", "ati_5", "-ati_6", "ati_7", "-ati_8", "ati_9"),
  VBA = c("-vb_allg_1", "vb_allg_2", "-vb_allg_3", "vb_allg_4"),
  AAZ = c("-aaz_1", "aaz_2", "-aaz_3", "aaz_4", "aaz_5", "aaz_6", "aaz_7", "aaz_8"),
  PRO = c("pro_1", "pro_2", "pro_3", "pro_4"),
  PRE = c("pre_1", "pre_2", "pre_3", "pre_4")
)
```

▼ schluesselliste	list [5]	List of length 5
ATI	character [9]	'ati_1' 'ati_2' '-ati_3' 'ati_4' 'ati_5' '-ati_6' ...
VBA	character [4]	'-vb_allg_1' 'vb_allg_2' '-vb_allg_3' 'vb_allg_4'
AAZ	character [8]	'-aaz_1' 'aaz_2' '-aaz_3' 'aaz_4' 'aaz_5' 'aaz_6' ...
PRO	character [4]	'pro_1' 'pro_2' 'pro_3' 'pro_4'
PRE	character [4]	'pre_1' 'pre_2' 'pre_3' 'pre_4'

# Skalenberechnung

```
scores <- scoreItems(schluesselliste, raw.short, min = 1, max = 6)
```

▼ scores	list [16] (S3: psych, score.itemr	List of length 16
scores	double [11 x 5]	3.61 5.11 2.89 3.33 2.89 3.33 4.00 3.50 4.00 3.50 4.
missing	double [11 x 5]	9 0 0 0 0 0 4 0 4 0 0 0 8 8 8 0 0 1 4 4 4 0 0 0 4 4 4 0
alpha	double [1 x 5]	0.871 -0.539 0.890 0.554 0.853
av.r	double [1 x 5]	0.429 -0.096 0.503 0.237 0.592
sn	double [1 x 5]	6.77 -0.35 8.11 1.24 5.81
...	...	...

- Nächster Schritt: Aufbau eines Datasets data:

```
data <- bind_cols(raw.short, as.tibble(scores$scores))
```

# Erstellung der finalen Datenmatrix

```
data <- bind_cols(raw.short, as.tibble(scores$scores))
```

age	gender	ati_1	[...]	comments	ATI	[...]	PRE
25	männlich	Stimme nicht zu	[...]	NA	3.375	[...]	2.555
22	männlich	Stimme nicht zu	[...]	NA	3.500	[...]	3.225
1	männlich	Stimme völlig zu	[...]	NA	3.500	[...]	4.125

- Die roten Spalten sind in Skalen verrechnet und können entfernt werden:

```
data <- data %>%
```

```
  select(-starts_with("ati", ignore.case = F)) %>%
```

```
  select(-starts_with("vba", ignore.case = F))
```



# Der Pipe-Operator %>%

---

```
data <- data %>%
```

```
  select(-starts_with("ati", ignore.case = F)) %>%
```

```
  select(-starts_with("vb", ignore.case = F)) %>%
```

```
  select(-starts_with("aaz", ignore.case = F)) %>%
```

```
  select(-starts_with("pre", ignore.case = F)) %>%
```

```
  select(-starts_with("pro", ignore.case = F))
```

# Der Pipe-Operator

- Syntaktische (!) Funktion aus dem Tidyverse.
- Erlaubt uns, einen „Schachtelsatz“ als Kette aufzuschreiben.

*Peter, der einen schwarzen Rucksack dabei hatte, welcher goldene Knöpfe hatte, die schwer schließbar waren, und ein rotes Logo, und eine weiße Tasche, ging in die Vorlesung.*

*Oder*

*Peter ging in die Vorlesung.*

*Er hatte eine weiße Tasche und einen schwarzen Rucksack dabei.*

*Der Rucksack hatte ein rotes Logo und goldene Knöpfe.*

*Die Knöpfe waren schwer schließbar.*

# Der Pipe-Operator %>%

```
data <- select(select(select(data, -starts_with("ati", ignore.case = F))  
, -starts_with("vb", ignore.case = F)), -starts_with("aaz", ignore.case = F))
```

```
data <- data %>%
```

```
select(_____, -starts_with("ati", ignore.case = F)) %>%
```

```
select(_____, -starts_with("vb", ignore.case = F)) %>%
```

```
select(_____, -starts_with("aaz", ignore.case = F)))
```

# Der Pipe-Operator %>%

---

```
data <- select(select(select(data, -starts_with("ati", ignore.case = F))  
, -starts_with("vb", ignore.case = F)), -starts_with("aaz", ignore.case = F))
```

```
data <- data %>%
```

```
  select( -starts_with("ati", ignore.case = F)) %>%
```

```
  select(-starts_with("vb", ignore.case = F)) %>%
```

```
  select(-starts_with("aaz", ignore.case = F)))
```

# Finale Datenmatrix

	gender	age	edu1	edu2	textfeld	abschluss	ATI	VBA	AAZ	PRO	PRE
1	weiblich	24	Abitur/Fachabitur	Hochschulabschl...	NA	NA	3.611111	4.00	3.8125	2.5	3.125
2	männlich	29	Abitur/Fachabitur	Hochschulabschl...	NA	NA	5.111111	3.50	3.8125	2.5	3.125
3	männlich	20	Haupt-/ Volkssch...	noch keine Ausbi...	NA	NA	2.888889	4.00	3.8125	2.5	3.125
4	männlich	34	NA	Hochschulabschl...	asdasFS	NOP	3.333333	3.50	3.2500	3.0	3.000
5	männlich	22	(noch) kein Schul...	noch keine Ausbi...	NA	NA	2.888889	4.00	3.3750	2.5	2.750
6	männlich	23	Realschulabschluss	Berufsausbildung	Jjj	NA	3.333333	3.75	1.7500	1.5	1.750
7	männlich	22	Abitur/Fachabitur	noch keine Ausbi...	NA	NA	3.000000	3.25	3.6250	3.0	3.750
8	weiblich	22	Abitur/Fachabitur	Hochschulabschl...	Leider fällt mir im...	NA	4.333333	4.00	4.8750	2.5	3.375
9	weiblich	21	Abitur/Fachabitur	noch keine Ausbi...	xxx	NA	3.888889	4.00	4.6250	3.5	5.000
10	weiblich	26	Haupt-/ Volkssch...	Hochschulabschl...	Schwer zu sagen. ...	Tolle Umfrage. Vi...	2.666667	4.00	3.6250	3.0	2.750
11	männlich	29	Abitur/Fachabitur	Hochschulabschl...	Überhaupt keine ...	Bester Frageboge...	4.666667	4.75	4.5000	4.0	4.375

- Nächster Schritt: Deskriptive Statistik

## Live Demo

# Der nächste Termin

---

- Nächste Woche:  
Boxplot und Histogramm  
Hypothesen und Hypothesentests
- Hausaufgabe in Kleingruppe:
  - Siehe L2P und Slack

