```
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ dplyr     1.1.3     ✓ readr     2.1.4
## ✓ forcats   1.0.0     ✓ stringr   1.5.0
## ✓ ggplot2   3.4.4     ✓ tibble    3.2.1
## ✓ lubridate 1.9.3     ✓ tidyr     1.3.0
## ✓ purrr     1.0.2
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(readxl)
library(skimr)
library(corrr)
```

```
##
## Attache Paket: 'corrr'
##
## Das folgende Objekt ist maskiert 'package:skimr':
##
##     focus
```

```
library(rstanarm)
```

```
## Lade nötiges Paket: Rcpp
## This is rstanarm version 2.26.1
## - See https://mc-stan.org/rstanarm/articles/priors for changes to default priors!
## - Default priors may change, so it's safest to specify priors, even if equivalent to the defaults.
## - For execution on a local, multicore CPU with excess RAM we recommend calling
##   options(mc.cores = parallel::detectCores())
```

```
library(bayestestR)
library(bayesplot)
```

```
## This is bayesplot version 1.10.0
## - Online documentation and vignettes at mc-stan.org/bayesplot
## - bayesplot theme set to bayesplot::theme_default()
##    * Does _not_ affect other ggplot2 plots
##    * See ?bayesplot_theme_set for details on theme setting
```

```
library(tidymodels)
```

```
## ── Attaching packages ────────────────────────────────────── tidymodels 1.1.1 ──
## ✓ broom        1.0.5     ✓ rsample      1.2.0
## ✓ dials        1.2.0     ✓ tune         1.1.2
## ✓ infer        1.0.5     ✓ workflows    1.1.3
## ✓ modeldata    1.2.0     ✓ workflowsets 1.0.1
## ✓ parsnip      1.1.1     ✓ yardstick    1.2.0
## ✓ recipes      1.0.8
## ── Conflicts ───────────────────────────────────────── tidymodels_conflicts() ──
## ✕ scales::discard()   masks purrr::discard()
## ✕ dplyr::filter()     masks stats::filter()
## ✕ recipes::fixed()    masks stringr::fixed()
## ✕ dplyr::lag()        masks stats::lag()
## ✕ rsample::populate() masks Rcpp::populate()
## ✕ yardstick::spec()   masks readr::spec()
## ✕ recipes::step()     masks stats::step()
## • Use suppressPackageStartupMessages() to eliminate package startup messages
```

```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
library(QuantPsyc)
```

```
## Warning: Paket 'QuantPsyc' wurde unter R Version 4.3.3 erstellt
```

```
## Lade nötiges Paket: boot
##
## Attache Paket: 'boot'
##
## Das folgende Objekt ist maskiert 'package:rstanarm':
##
##     logit
##
## Lade nötiges Paket: MASS
##
## Attache Paket: 'MASS'
##
## Das folgende Objekt ist maskiert 'package:dplyr':
##
##     select
##
##
## Attache Paket: 'QuantPsyc'
##
## Das folgende Objekt ist maskiert 'package:base':
##
##     norm
```

```
options(mc.cores = parallel::detectCores())
```

```
conflicted::conflict_prefer("select", "dplyr")
```

```
## [conflicted] Will prefer dplyr::select over any other package.
```

```
conflicted::conflict_prefer("filter", "dplyr")
```

```
## [conflicted] Will prefer dplyr::filter over any other package.
```

1. Vorbereitung der Daten

Daten einlesen

```
d_raw <- read_excel("raw_data.xlsx")
```

Unnötige Variablen entfernen

```
raw1 <- d_raw %>%
  select(-c(CASE, SERIAL, REF, QUESTNNR, MODE, STARTED, TIME001:TIME012, MAILSENT, LASTDATA, FINISHED, Q_VIEW
ER, LASTPAGE, MAXPAGE, MISSREL, MISSING))
```

Variablenbeschreibung (Zeile) entfernen

```
raw1 = raw1[-1,]
```

Spaltennamen umbenennen

```
names(raw1) <- c("Alter", "Geschlecht", "Bildung", "Bildung_sonstig", "BZG_J", "BZG_M", "Beschaeftigungsart",
"Arbeitszeitmodell", "Gehalt", "Position_im_Unternehmen", "Remotarbeit", "IS_K01", "IS_K02", "IS_K03", "IS_K0
4", "IS_E05", "IS_E06", "IS_E07", "IS_E08", "OC_A01", "OC_A02", "OC_A03", "OC_A04", "OC_A05", "CPE_C01", "CPE
_C02", "CPE_C03", "CPE_C04", "CPE_I05", "CPE_I06", "CPE_I07", "CPE_I08", "CPE_M09", "CPE_M10", "CPE_M11", "CP
E_M12", "CPE_E13", "CPE_E14", "CPE_E15", "CPE_E16", "PS_01", "PS_02", "PS_03", "PS_04", "PS_05", "PS_06", "PS
_07", "QQ_01", "QQ_02", "QQ_03", "QQ_04", "QQ_05", "QQ_06", "QQ_07", "QQ_N08", "QQ_N09", "QQ_N10", "QQ_N11",
"QQ_N12", "QQ_N13", "QQ_N14", "SL_01", "SL_02", "SL_03", "SL_04", "SL_05", "SL_06", "SL_07", "ZE01", "ZE04",
"TIME_SUM", "TIME_RSI")
```

ID-Spalte einfügen

```
raw2 <- raw1 %>%
  mutate(ID = row_number()) %>%
  select(ID, everything())
```

Kompletten Datensatz auf NA's prüfen

```
raw2 %>%
  summarise(across(everything(),
                   ~sum(is.na(.))))
```

```
## # A tibble: 1 × 73
##      ID Alter Geschlecht Bildung Bildung_sonstig BZG_J BZG_M Beschaeftigungsart
##   <int> <int>      <int>   <int>           <int> <int> <int>              <int>
## 1     0     2          0       0             189     0     5                  0
## # ℹ 65 more variables: Arbeitszeitmodell <int>, Gehalt <int>,
## #   Position_im_Unternehmen <int>, Remotarbeit <int>, IS_K01 <int>,
## #   IS_K02 <int>, IS_K03 <int>, IS_K04 <int>, IS_E05 <int>, IS_E06 <int>,
## #   IS_E07 <int>, IS_E08 <int>, OC_A01 <int>, OC_A02 <int>, OC_A03 <int>,
## #   OC_A04 <int>, OC_A05 <int>, CPE_C01 <int>, CPE_C02 <int>, CPE_C03 <int>,
## #   CPE_C04 <int>, CPE_I05 <int>, CPE_I06 <int>, CPE_I07 <int>, CPE_I08 <int>,
## #   CPE_M09 <int>, CPE_M10 <int>, CPE_M11 <int>, CPE_M12 <int>, …
```

```
skim(raw2)
```

Data summary

Name	raw2
Number of rows	192
Number of columns	73
_______________________	
Column type frequency:	
character	72
numeric	1
________________________	
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Alter	2	0.99	2	2	0	39	0
Geschlecht	0	1.00	1	1	0	2	0
Bildung	0	1.00	1	1	0	8	0
Bildung_sonstig	189	0.02	12	17	0	3	0
BZG_J	0	1.00	1	2	0	38	0
BZG_M	5	0.97	1	2	0	12	0
Beschaeftigungsart	0	1.00	1	1	0	2	0
Arbeitszeitmodell	0	1.00	1	5	0	40	0
Gehalt	2	0.99	1	1	0	7	0
Position_im_Unternehmen	0	1.00	1	1	0	4	0
Remotarbeit	0	1.00	1	1	0	6	0
IS_K01	0	1.00	1	1	0	7	0
IS_K02	0	1.00	1	1	0	7	0
IS_K03	0	1.00	1	1	0	7	0

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
IS_K04	0	1.00	1	1	0	7	0
IS_E05	0	1.00	1	1	0	7	0
IS_E06	0	1.00	1	1	0	7	0
IS_E07	0	1.00	1	1	0	7	0
IS_E08	0	1.00	1	1	0	7	0
OC_A01	0	1.00	1	1	0	5	0
OC_A02	0	1.00	1	1	0	5	0
OC_A03	0	1.00	1	1	0	5	0
OC_A04	0	1.00	1	1	0	5	0
OC_A05	0	1.00	1	1	0	5	0
CPE_C01	0	1.00	1	1	0	7	0
CPE_C02	0	1.00	1	1	0	7	0
CPE_C03	0	1.00	1	1	0	7	0
CPE_C04	0	1.00	1	1	0	7	0
CPE_I05	0	1.00	1	1	0	7	0
CPE_I06	0	1.00	1	1	0	7	0
CPE_I07	0	1.00	1	1	0	7	0
CPE_I08	0	1.00	1	1	0	7	0
CPE_M09	0	1.00	1	1	0	7	0
CPE_M10	0	1.00	1	1	0	7	0
CPE_M11	1	0.99	1	1	0	7	0
CPE_M12	0	1.00	1	1	0	7	0
CPE_E13	0	1.00	1	1	0	7	0
CPE_E14	0	1.00	1	1	0	7	0
CPE_E15	0	1.00	1	1	0	7	0
CPE_E16	0	1.00	1	1	0	7	0
PS_01	0	1.00	1	1	0	7	0
PS_02	0	1.00	1	1	0	7	0
PS_03	0	1.00	1	1	0	7	0
PS_04	0	1.00	1	1	0	7	0
PS_05	0	1.00	1	1	0	7	0
PS_06	0	1.00	1	1	0	7	0
PS_07	0	1.00	1	1	0	7	0
QQ_01	0	1.00	1	1	0	5	0
QQ_02	0	1.00	1	1	0	5	0

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
QQ_03	0	1.00	1	1	0	5	0
QQ_04	0	1.00	1	1	0	5	0
QQ_05	0	1.00	1	1	0	5	0
QQ_06	0	1.00	1	1	0	5	0
QQ_07	0	1.00	1	1	0	5	0
QQ_N08	0	1.00	1	1	0	5	0
QQ_N09	0	1.00	1	1	0	5	0
QQ_N10	0	1.00	1	1	0	5	0
QQ_N11	0	1.00	1	1	0	5	0
QQ_N12	0	1.00	1	1	0	5	0
QQ_N13	0	1.00	1	1	0	5	0
QQ_N14	0	1.00	1	1	0	5	0
SL_01	0	1.00	1	1	0	7	0
SL_02	0	1.00	1	1	0	7	0
SL_03	0	1.00	1	1	0	7	0
SL_04	0	1.00	1	1	0	7	0
SL_05	1	0.99	1	1	0	7	0
SL_06	0	1.00	1	1	0	7	0
SL_07	0	1.00	1	1	0	7	0
ZE01	0	1.00	1	1	0	1	0
ZE04	0	1.00	1	1	0	1	0
TIME_SUM	0	1.00	3	4	0	167	0
TIME_RSI	0	1.00	1	4	0	111	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
ID	0	1	96.5	55.57	1	48.75	96.5	144.25	192	▇▇▇▇▇

> NA bei CPE_M11 & SL_05 –> später prüfen, wenn Werte numerisch sind! (vgl. 4.!)

2. Negative gepolte Items prüfen

Recodieren der Psychologische Sicherheit Items

```
raw3 <- raw2 %>%
  mutate(across(
    .cols = c(PS_03, PS_05, PS_06),
    .fns = ~ dplyr::recode(.,
                    `1` = 7,
                    `2` = 6,
                    `3` = 5,
                    `4` = 4,
                    `5` = 3,
                    `6` = 2,
                    `7` = 1)))
```

Recodieren des organisationalen Commitments

```
raw4 <- raw3 %>%
  mutate(across(
    .cols = OC_A02,
    .fns = ~dplyr::recode(.,
                    `1` = 5,
                    `2` = 4,
                    `3` = 3,
                    `4` = 2,
                    `5` = 1)))
```

3. Variablen umkodieren

```
raw4$Geschlecht <- as.numeric(raw4$Geschlecht)
raw4$Alter <- as.numeric(raw4$Alter)
raw4$Bildung <- as.numeric(raw4$Bildung)
raw4$BZG_J <- as.numeric(raw4$BZG_J)
raw4$BZG_M <- as.numeric(raw4$BZG_M)
raw4$Arbeitszeitmodell <- as.numeric(raw4$Arbeitszeitmodell)
raw4$Beschaeftigungsart <- as.numeric(raw4$Beschaeftigungsart)
raw4$Gehalt <- as.numeric(raw4$Gehalt)
raw4$Position_im_Unternehmen <- as.numeric(raw4$Position_im_Unternehmen)
raw4$Remotarbeit <- as.numeric(raw4$Remotarbeit)
```

```
raw4$IS_K01 <- as.numeric(raw4$IS_K01)
raw4$IS_K02 <- as.numeric(raw4$IS_K02)
raw4$IS_K03 <- as.numeric(raw4$IS_K03)
raw4$IS_K04 <- as.numeric(raw4$IS_K04)
raw4$IS_E05 <- as.numeric(raw4$IS_E05)
raw4$IS_E06 <- as.numeric(raw4$IS_E06)
raw4$IS_E07 <- as.numeric(raw4$IS_E07)
raw4$IS_E08 <- as.numeric(raw4$IS_E08)
```

```
raw4$OC_A01 <- as.numeric(raw4$OC_A01)
raw4$OC_A02 <- as.numeric(raw4$OC_A02)
raw4$OC_A03 <- as.numeric(raw4$OC_A03)
raw4$OC_A04 <- as.numeric(raw4$OC_A04)
raw4$OC_A05 <- as.numeric(raw4$OC_A05)
```

```
raw4$CPE_C01 <- as.numeric(raw4$CPE_C01)
raw4$CPE_C02 <- as.numeric(raw4$CPE_C02)
raw4$CPE_C03 <- as.numeric(raw4$CPE_C03)
raw4$CPE_C04 <- as.numeric(raw4$CPE_C04)
raw4$CPE_I05 <- as.numeric(raw4$CPE_I05)
raw4$CPE_I06 <- as.numeric(raw4$CPE_I06)
raw4$CPE_I07 <- as.numeric(raw4$CPE_I07)
raw4$CPE_I08 <- as.numeric(raw4$CPE_I08)
raw4$CPE_M09 <- as.numeric(raw4$CPE_M09)
raw4$CPE_M10 <- as.numeric(raw4$CPE_M10)
raw4$CPE_M11 <- as.numeric(raw4$CPE_M11)
raw4$CPE_M12 <- as.numeric(raw4$CPE_M12)
raw4$CPE_E13 <- as.numeric(raw4$CPE_E13)
raw4$CPE_E14 <- as.numeric(raw4$CPE_E14)
raw4$CPE_E15 <- as.numeric(raw4$CPE_E15)
raw4$CPE_E16 <- as.numeric(raw4$CPE_E16)
```

```
raw4$PS_01 <- as.numeric(raw4$PS_01)
raw4$PS_02 <- as.numeric(raw4$PS_02)
raw4$PS_03 <- as.numeric(raw4$PS_03)
raw4$PS_04 <- as.numeric(raw4$PS_04)
raw4$PS_05 <- as.numeric(raw4$PS_05)
raw4$PS_06 <- as.numeric(raw4$PS_06)
raw4$PS_07 <- as.numeric(raw4$PS_07)
```

```
raw4$QQ_01 <- as.numeric(raw4$QQ_01)
raw4$QQ_02 <- as.numeric(raw4$QQ_02)
raw4$QQ_03 <- as.numeric(raw4$QQ_03)
raw4$QQ_04 <- as.numeric(raw4$QQ_04)
raw4$QQ_05 <- as.numeric(raw4$QQ_05)
raw4$QQ_06 <- as.numeric(raw4$QQ_06)
raw4$QQ_07 <- as.numeric(raw4$QQ_07)
```

```
raw4$QQ_N08 <- as.numeric(raw4$QQ_N08)
raw4$QQ_N09 <- as.numeric(raw4$QQ_N09)
raw4$QQ_N10 <- as.numeric(raw4$QQ_N10)
raw4$QQ_N11 <- as.numeric(raw4$QQ_N11)
raw4$QQ_N12 <- as.numeric(raw4$QQ_N12)
raw4$QQ_N13 <- as.numeric(raw4$QQ_N13)
raw4$QQ_N14 <- as.numeric(raw4$QQ_N14)
```

```
raw4$SL_01 <- as.numeric(raw4$SL_01)
raw4$SL_02 <- as.numeric(raw4$SL_02)
raw4$SL_03 <- as.numeric(raw4$SL_03)
raw4$SL_04 <- as.numeric(raw4$SL_04)
raw4$SL_05 <- as.numeric(raw4$SL_05)
raw4$SL_06 <- as.numeric(raw4$SL_06)
raw4$SL_07 <- as.numeric(raw4$SL_07)
```

```
raw4$ZE01 <- as.numeric(raw4$ZE01)
raw4$ZE04 <- as.numeric(raw4$ZE04)
raw4$TIME_SUM <- as.numeric(raw4$TIME_SUM)
raw4$TIME_RSI <- as.numeric(raw4$TIME_RSI)
```

Bearbeitungszeit

```
raw4_1 <- raw4 %>%
  mutate(Time_sum_m = TIME_SUM / 60)
```

```
raw4_1 %>%
  summarise(median = median(Time_sum_m),
            sd = sd(Time_sum_m))
```

```
## # A tibble: 1 × 2
##   median    sd
##    <dbl> <dbl>
## 1   9.99  2.77
```

> Alles was größer oder kleiner als 2 SD ist wird nochmal genauer betrachtet!

```
raw4_1 %>%
  filter(Time_sum_m < 4.450805 | Time_sum_m > 15.53253) %>%
  select(ID, Alter, Geschlecht, Bildung, Time_sum_m, everything())
```

```
## # A tibble: 7 × 74
##      ID Alter Geschlecht Bildung Time_sum_m Bildung_sonstig BZG_J BZG_M
##   <int> <dbl>      <dbl>   <dbl>      <dbl> <chr>           <dbl> <dbl>
## 1    15    26          2       2       2.43 <NA>                0     1
## 2    40    53          2       5      16.8  <NA>               15     0
## 3    42    55          1       2      15.5  <NA>                9    NA
## 4    91    48          2       5      16.8  <NA>                2     0
## 5    93    51          2       5      16.9  <NA>               17     8
## 6   106    24          1       5       4.25 <NA>                6     4
## 7   185    55          2       3      15.7  <NA>               39     9
## # ℹ 66 more variables: Beschaeftigungsart <dbl>, Arbeitszeitmodell <dbl>,
## #   Gehalt <dbl>, Position_im_Unternehmen <dbl>, Remotarbeit <dbl>,
## #   IS_K01 <dbl>, IS_K02 <dbl>, IS_K03 <dbl>, IS_K04 <dbl>, IS_E05 <dbl>,
## #   IS_E06 <dbl>, IS_E07 <dbl>, IS_E08 <dbl>, OC_A01 <dbl>, OC_A02 <dbl>,
## #   OC_A03 <dbl>, OC_A04 <dbl>, OC_A05 <dbl>, CPE_C01 <dbl>, CPE_C02 <dbl>,
## #   CPE_C03 <dbl>, CPE_C04 <dbl>, CPE_I05 <dbl>, CPE_I06 <dbl>, CPE_I07 <dbl>,
## #   CPE_I08 <dbl>, CPE_M09 <dbl>, CPE_M10 <dbl>, CPE_M11 <dbl>, …
```

> Wie sich zeigt, haben zwei Probanden den Fragebogen unter 2 SD vom Median beantwortet, weswegen diese aus dem Datensatz entfernt werden. Da weitere 5 Probanden länger als 2 SD vom Median zur Beantwortung benötigt haben, diese jedoch alle um die 50 Jahre oder älter sind, wird es von der gesamten Gruppe als logisch empfunden, dass diese Gruppe gegebenenfalls eher länger benötigt.

Personen mit einer unterdurchschnittlichen Bearbeitungsdauer rausschmeißen

```
drops <-c(15, 106)
```

```
raw4_2 <- raw4_1[-drops,]
```

Daten neu sortieren

```
raw4_3 <- raw4_2 %>%
  select(-ID) %>%
  mutate(ID = row_number()) %>%
  select(ID, Alter, Geschlecht, Bildung, Bildung_sonstig, BZG_J, BZG_M, Beschaeftigungsart, Arbeitszeitmodell
, Gehalt, Position_im_Unternehmen, Remotarbeit, everything())
```

Daten checken

```
raw4_3 %>%
  filter(Time_sum_m < 4.450805 | Time_sum_m > 15.53253) %>%
  select(ID, Alter, Geschlecht, Bildung, Time_sum_m, everything())
```

```
## # A tibble: 5 × 74
##      ID Alter Geschlecht Bildung Time_sum_m Bildung_sonstig BZG_J BZG_M
##   <int> <dbl>      <dbl>   <dbl>      <dbl> <chr>           <dbl> <dbl>
## 1    39    53          2       5       16.8 <NA>               15     0
## 2    41    55          1       2       15.5 <NA>                9    NA
## 3    90    48          2       5       16.8 <NA>                2     0
## 4    92    51          2       5       16.9 <NA>               17     8
## 5   183    55          2       3       15.7 <NA>               39     9
## # ℹ 66 more variables: Beschaeftigungsart <dbl>, Arbeitszeitmodell <dbl>,
## #   Gehalt <dbl>, Position_im_Unternehmen <dbl>, Remotarbeit <dbl>,
## #   IS_K01 <dbl>, IS_K02 <dbl>, IS_K03 <dbl>, IS_K04 <dbl>, IS_E05 <dbl>,
## #   IS_E06 <dbl>, IS_E07 <dbl>, IS_E08 <dbl>, OC_A01 <dbl>, OC_A02 <dbl>,
## #   OC_A03 <dbl>, OC_A04 <dbl>, OC_A05 <dbl>, CPE_C01 <dbl>, CPE_C02 <dbl>,
## #   CPE_C03 <dbl>, CPE_C04 <dbl>, CPE_I05 <dbl>, CPE_I06 <dbl>, CPE_I07 <dbl>,
## #   CPE_I08 <dbl>, CPE_M09 <dbl>, CPE_M10 <dbl>, CPE_M11 <dbl>, …
```

4. Mittelwerte der jeweiligen Items mit NA’s anschauen

```
raw4_3 %>%
  summarise(mean_CPE_M11 = mean(CPE_M11, na.rm=TRUE),
            mean_SL_05 = mean(SL_05, na.rm=TRUE))
```

```
## # A tibble: 1 × 2
##   mean_CPE_M11 mean_SL_05
##          <dbl>      <dbl>
## 1         4.66       3.42
```

Spalten mit NA’s heraussuchen

```
raw4_3 %>%
  filter(is.na(CPE_M11 | SL_05))
```

```
## # A tibble: 1 × 74
##      ID Alter Geschlecht Bildung Bildung_sonstig BZG_J BZG_M Beschaeftigungsart
##   <int> <dbl>      <dbl>   <dbl> <chr>           <dbl> <dbl>              <dbl>
## 1    60    48          2       7 <NA>               11     7                  2
## # ℹ 66 more variables: Arbeitszeitmodell <dbl>, Gehalt <dbl>,
## #   Position_im_Unternehmen <dbl>, Remotarbeit <dbl>, IS_K01 <dbl>,
## #   IS_K02 <dbl>, IS_K03 <dbl>, IS_K04 <dbl>, IS_E05 <dbl>, IS_E06 <dbl>,
## #   IS_E07 <dbl>, IS_E08 <dbl>, OC_A01 <dbl>, OC_A02 <dbl>, OC_A03 <dbl>,
## #   OC_A04 <dbl>, OC_A05 <dbl>, CPE_C01 <dbl>, CPE_C02 <dbl>, CPE_C03 <dbl>,
## #   CPE_C04 <dbl>, CPE_I05 <dbl>, CPE_I06 <dbl>, CPE_I07 <dbl>, CPE_I08 <dbl>,
## #   CPE_M09 <dbl>, CPE_M10 <dbl>, CPE_M11 <dbl>, CPE_M12 <dbl>, …
```

> Eine Person (ID = 60) hat also weder eine Angabe zu CPE_M11 und SL_05 gemacht! Da ansonsten alle Werte vorhanden sind, werden diese NA’s durch die Mittelwerte ausgetauscht!

Austauschen der Werte

```
raw4_3$CPE_M11[60] <- 4.656085
raw4_3$SL_05[60] <- 3.42328
```

NA’s checken

```
skim(raw4_3)
```

Data summary

Name	raw4_3
Number of rows	190
Number of columns	74
_______________________	
Column type frequency:	
character	1
numeric	73
________________________	
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Bildung_sonstig	187	0.02	12	17	0	3	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
ID	0	1.00	95.50	54.99	1.00	48.25	95.50	142.75	190.00	▇▇▇▇▇
Alter	2	0.99	39.10	13.52	21.00	27.00	33.50	53.00	63.00	▇▂▁▂▅

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Geschlecht	0	1.00	1.69	0.46	1.00	1.00	2.00	2.00	2.00	
Bildung	0	1.00	5.41	1.50	2.00	4.00	6.00	6.00	9.00	
BZG_J	0	1.00	9.97	12.08	0.00	1.00	4.50	13.75	45.00	
BZG_M	5	0.97	5.15	3.40	0.00	2.00	5.00	8.00	11.00	
Beschaeftigungsart	0	1.00	1.85	0.36	1.00	2.00	2.00	2.00	2.00	
Arbeitszeitmodell	0	1.00	31.77	9.83	0.00	25.00	35.50	40.00	56.00	
Gehalt	2	0.99	3.04	1.46	1.00	2.00	3.00	4.00	7.00	
Position_im_Unternehmen	0	1.00	1.32	0.71	1.00	1.00	1.00	1.00	4.00	
Remotarbeit	0	1.00	2.24	1.49	1.00	1.00	2.00	3.00	6.00	
IS_K01	0	1.00	3.52	1.87	1.00	2.00	3.00	5.00	7.00	
IS_K02	0	1.00	4.22	1.79	1.00	3.00	5.00	6.00	7.00	
IS_K03	0	1.00	2.66	1.45	1.00	2.00	2.00	3.00	7.00	
IS_K04	0	1.00	3.51	1.96	1.00	2.00	3.00	5.00	7.00	
IS_E05	0	1.00	2.73	1.74	1.00	1.00	2.00	4.00	7.00	
IS_E06	0	1.00	3.07	1.65	1.00	2.00	3.00	4.00	7.00	
IS_E07	0	1.00	3.33	1.71	1.00	2.00	3.00	5.00	7.00	
IS_E08	0	1.00	2.60	1.56	1.00	1.00	2.00	3.00	7.00	
OC_A01	0	1.00	3.61	1.15	1.00	3.00	4.00	4.00	5.00	
OC_A02	0	1.00	3.43	1.24	1.00	2.00	4.00	4.00	5.00	
OC_A03	0	1.00	3.60	1.11	1.00	3.00	4.00	4.00	5.00	
OC_A04	0	1.00	3.49	1.14	1.00	3.00	4.00	4.00	5.00	
OC_A05	0	1.00	3.54	1.05	1.00	3.00	4.00	4.00	5.00	
CPE_C01	0	1.00	4.91	1.53	1.00	4.00	5.00	6.00	7.00	
CPE_C02	0	1.00	4.92	1.62	1.00	4.00	5.00	6.00	7.00	
CPE_C03	0	1.00	5.08	1.48	1.00	4.00	5.00	6.00	7.00	
CPE_C04	0	1.00	4.74	1.54	1.00	4.00	5.00	6.00	7.00	
CPE_I05	0	1.00	5.06	1.43	1.00	4.00	5.00	6.00	7.00	
CPE_I06	0	1.00	4.83	1.44	1.00	4.00	5.00	6.00	7.00	
CPE_I07	0	1.00	5.34	1.35	1.00	4.00	6.00	6.00	7.00	
CPE_I08	0	1.00	4.82	1.49	1.00	4.00	5.00	6.00	7.00	
CPE_M09	0	1.00	4.18	1.60	1.00	3.00	4.00	5.00	7.00	
CPE_M10	0	1.00	4.76	1.55	1.00	4.00	5.00	6.00	7.00	
CPE_M11	0	1.00	4.66	1.35	1.00	4.00	5.00	6.00	7.00	
CPE_M12	0	1.00	5.17	1.38	1.00	4.00	5.00	6.00	7.00	
CPE_E13	0	1.00	4.67	1.59	1.00	4.00	5.00	6.00	7.00	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
CPE_E14	0	1.00	4.91	1.50	1.00	4.00	5.00	6.00	7.00	
CPE_E15	0	1.00	4.93	1.43	1.00	4.00	5.00	6.00	7.00	
CPE_E16	0	1.00	4.77	1.64	1.00	4.00	5.00	6.00	7.00	
PS_01	0	1.00	5.45	1.46	1.00	5.00	6.00	6.00	7.00	
PS_02	0	1.00	5.89	1.70	1.00	6.00	7.00	7.00	7.00	
PS_03	0	1.00	5.36	1.68	1.00	4.25	6.00	7.00	7.00	
PS_04	0	1.00	4.87	1.46	1.00	4.00	5.00	6.00	7.00	
PS_05	0	1.00	5.15	1.74	1.00	4.00	6.00	7.00	7.00	
PS_06	0	1.00	5.88	1.47	1.00	5.00	6.00	7.00	7.00	
PS_07	0	1.00	5.42	1.43	1.00	5.00	6.00	6.00	7.00	
QQ_01	0	1.00	2.66	1.36	1.00	2.00	2.00	4.00	5.00	
QQ_02	0	1.00	1.52	0.83	1.00	1.00	1.00	2.00	5.00	
QQ_03	0	1.00	2.53	1.18	1.00	2.00	2.00	3.00	5.00	
QQ_04	0	1.00	2.13	1.03	1.00	1.00	2.00	3.00	5.00	
QQ_05	0	1.00	2.39	1.17	1.00	2.00	2.00	3.00	5.00	
QQ_06	0	1.00	2.15	1.11	1.00	1.00	2.00	3.00	5.00	
QQ_07	0	1.00	2.49	1.21	1.00	2.00	2.00	3.00	5.00	
QQ_N08	0	1.00	2.75	1.05	1.00	2.00	3.00	4.00	5.00	
QQ_N09	0	1.00	2.08	1.10	1.00	1.00	2.00	3.00	5.00	
QQ_N10	0	1.00	2.25	1.06	1.00	2.00	2.00	3.00	5.00	
QQ_N11	0	1.00	2.19	0.96	1.00	2.00	2.00	3.00	5.00	
QQ_N12	0	1.00	2.10	1.21	1.00	1.00	2.00	2.00	5.00	
QQ_N13	0	1.00	1.75	0.87	1.00	1.00	2.00	2.00	5.00	
QQ_N14	0	1.00	2.33	1.01	1.00	2.00	2.00	3.00	5.00	
SL_01	0	1.00	4.69	1.69	1.00	3.25	5.00	6.00	7.00	
SL_02	0	1.00	3.98	1.76	1.00	3.00	4.00	5.00	7.00	
SL_03	0	1.00	4.56	2.10	1.00	3.00	5.00	6.00	7.00	
SL_04	0	1.00	4.12	1.80	1.00	3.00	4.00	6.00	7.00	
SL_05	0	1.00	3.42	1.63	1.00	2.00	3.21	4.00	7.00	
SL_06	0	1.00	5.44	1.45	1.00	5.00	6.00	6.00	7.00	
SL_07	0	1.00	5.34	1.71	1.00	4.00	6.00	7.00	7.00	
ZE01	0	1.00	1.00	0.00	1.00	1.00	1.00	1.00	1.00	
ZE04	0	1.00	1.00	0.00	1.00	1.00	1.00	1.00	1.00	
TIME_SUM	0	1.00	607.62	161.73	277.00	488.25	602.00	710.00	1016.00	
TIME_RSI	0	1.00	1.07	0.37	0.22	0.83	1.01	1.30	2.15	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
Time_sum_m	0	1.00	10.13	2.70	4.62	8.14	10.03	11.83	16.93	▂▅▇▂▁

5. `BZG_J` und `BZG_M` zusammenrechnen

BZG_M NA’s mit Nullen austauschen

```
raw4_3 %>%
  filter(is.na(BZG_M))
```

```
## # A tibble: 5 × 74
##      ID Alter Geschlecht Bildung Bildung_sonstig BZG_J BZG_M Beschaeftigungsart
##   <int> <dbl>      <dbl>   <dbl> <chr>           <dbl> <dbl>              <dbl>
## 1    38    53          2       3 <NA>               33    NA                  2
## 2    41    55          1       2 <NA>                9    NA                  2
## 3   105    59          2       6 <NA>               32    NA                  2
## 4   147    56          1       2 <NA>               16    NA                  2
## 5   150    53          2       5 <NA>               30    NA                  2
## # ℹ 66 more variables: Arbeitszeitmodell <dbl>, Gehalt <dbl>,
## #   Position_im_Unternehmen <dbl>, Remotarbeit <dbl>, IS_K01 <dbl>,
## #   IS_K02 <dbl>, IS_K03 <dbl>, IS_K04 <dbl>, IS_E05 <dbl>, IS_E06 <dbl>,
## #   IS_E07 <dbl>, IS_E08 <dbl>, OC_A01 <dbl>, OC_A02 <dbl>, OC_A03 <dbl>,
## #   OC_A04 <dbl>, OC_A05 <dbl>, CPE_C01 <dbl>, CPE_C02 <dbl>, CPE_C03 <dbl>,
## #   CPE_C04 <dbl>, CPE_I05 <dbl>, CPE_I06 <dbl>, CPE_I07 <dbl>, CPE_I08 <dbl>,
## #   CPE_M09 <dbl>, CPE_M10 <dbl>, CPE_M11 <dbl>, CPE_M12 <dbl>, …
```

```
raw4_3$BZG_M[38] <- 0
raw4_3$BZG_M[41] <- 0
raw4_3$BZG_M[105] <- 0
raw4_3$BZG_M[147] <- 0
raw4_3$BZG_M[150] <- 0
```

```
raw4_3 %>%
  filter(is.na(BZG_M))
```

```
## # A tibble: 0 × 74
## # ℹ 74 variables: ID <int>, Alter <dbl>, Geschlecht <dbl>, Bildung <dbl>,
## #   Bildung_sonstig <chr>, BZG_J <dbl>, BZG_M <dbl>, Beschaeftigungsart <dbl>,
## #   Arbeitszeitmodell <dbl>, Gehalt <dbl>, Position_im_Unternehmen <dbl>,
## #   Remotarbeit <dbl>, IS_K01 <dbl>, IS_K02 <dbl>, IS_K03 <dbl>, IS_K04 <dbl>,
## #   IS_E05 <dbl>, IS_E06 <dbl>, IS_E07 <dbl>, IS_E08 <dbl>, OC_A01 <dbl>,
## #   OC_A02 <dbl>, OC_A03 <dbl>, OC_A04 <dbl>, OC_A05 <dbl>, CPE_C01 <dbl>,
## #   CPE_C02 <dbl>, CPE_C03 <dbl>, CPE_C04 <dbl>, CPE_I05 <dbl>, …
```

BZG-Variablen umrechnen

```
raw4a <- raw4_3 %>%
  mutate(BZG_JM = (BZG_J*12)) %>%
  mutate(BZG = (BZG_JM+BZG_M)/12) %>%
  select(-c(BZG_J, BZG_M, BZG_JM)) %>%
  select(ID, Alter, Geschlecht, Bildung, Bildung_sonstig, BZG, everything())
```

BZG genauer betrachten

```
raw4a %>%
  filter(BZG < 1) %>%
  count()
```

```
## # A tibble: 1 × 1
##       n
##   <int>
## 1    30
```

> 30 Personen sind weniger als 1 Jahr bei ihrem aktuellen Unternehmen beschäftigt

Deskriptive Werte von BZG betrachten

```
raw4a %>%
  summarise(mean = mean(BZG),
            median = median(BZG),
         sd = sd(BZG),
         max = max(BZG),
         min = min(BZG))
```

```
## # A tibble: 1 × 5
##    mean median    sd   max    min
##   <dbl>  <dbl> <dbl> <dbl>  <dbl>
## 1  10.4   4.92  12.1  45.7 0.0833
```

6. Verteilungen der Soziodemografika

Geschlecht

```
raw4a %>%
  count(Geschlecht) %>%
  mutate(prob = n/sum(n)) %>%
  round(2)
```

```
## # A tibble: 2 × 3
##   Geschlecht     n  prob
##        <dbl> <dbl> <dbl>
## 1          1    58  0.31
## 2          2   132  0.69
```

> 1 = männlich 2 = weiblich

Alter

```
raw4a %>%
  summarise(mean(Alter, na.rm=TRUE),
            median(Alter, na.rm=TRUE),
         sd(Alter, na.rm=TRUE),
         max(Alter, na.rm =TRUE),
         min(Alter, na.rm=TRUE))
```

```
## # A tibble: 1 × 5
##   `mean(Alter, na.rm = TRUE)` median(Alter, na.rm = TRU…¹ sd(Alter, na.rm = TR…²
##                         <dbl>                       <dbl>                  <dbl>
## 1                        39.1                        33.5                   13.5
## # ℹ abbreviated names: ¹`median(Alter, na.rm = TRUE)`,
## #   ²`sd(Alter, na.rm = TRUE)`
## # ℹ 2 more variables: `max(Alter, na.rm = TRUE)` <dbl>,
## #   `min(Alter, na.rm = TRUE)` <dbl>
```

```
raw4a %>%
  filter(Alter > 60)
```

```
## # A tibble: 6 × 73
##      ID Alter Geschlecht Bildung Bildung_sonstig   BZG Beschaeftigungsart
##   <int> <dbl>      <dbl>   <dbl> <chr>           <dbl>              <dbl>
## 1     8    63          1       6 <NA>             41.5                  2
## 2    15    61          2       3 <NA>             23.2                  2
## 3    16    62          1       5 <NA>             44.5                  2
## 4    47    62          1       3 <NA>             45.7                  2
## 5   160    62          2       7 <NA>             15.7                  2
## 6   180    61          2       6 <NA>             11.3                  2
## # ℹ 66 more variables: Arbeitszeitmodell <dbl>, Gehalt <dbl>,
## #   Position_im_Unternehmen <dbl>, Remotarbeit <dbl>, IS_K01 <dbl>,
## #   IS_K02 <dbl>, IS_K03 <dbl>, IS_K04 <dbl>, IS_E05 <dbl>, IS_E06 <dbl>,
## #   IS_E07 <dbl>, IS_E08 <dbl>, OC_A01 <dbl>, OC_A02 <dbl>, OC_A03 <dbl>,
## #   OC_A04 <dbl>, OC_A05 <dbl>, CPE_C01 <dbl>, CPE_C02 <dbl>, CPE_C03 <dbl>,
## #   CPE_C04 <dbl>, CPE_I05 <dbl>, CPE_I06 <dbl>, CPE_I07 <dbl>, CPE_I08 <dbl>,
## #   CPE_M09 <dbl>, CPE_M10 <dbl>, CPE_M11 <dbl>, CPE_M12 <dbl>, …
```

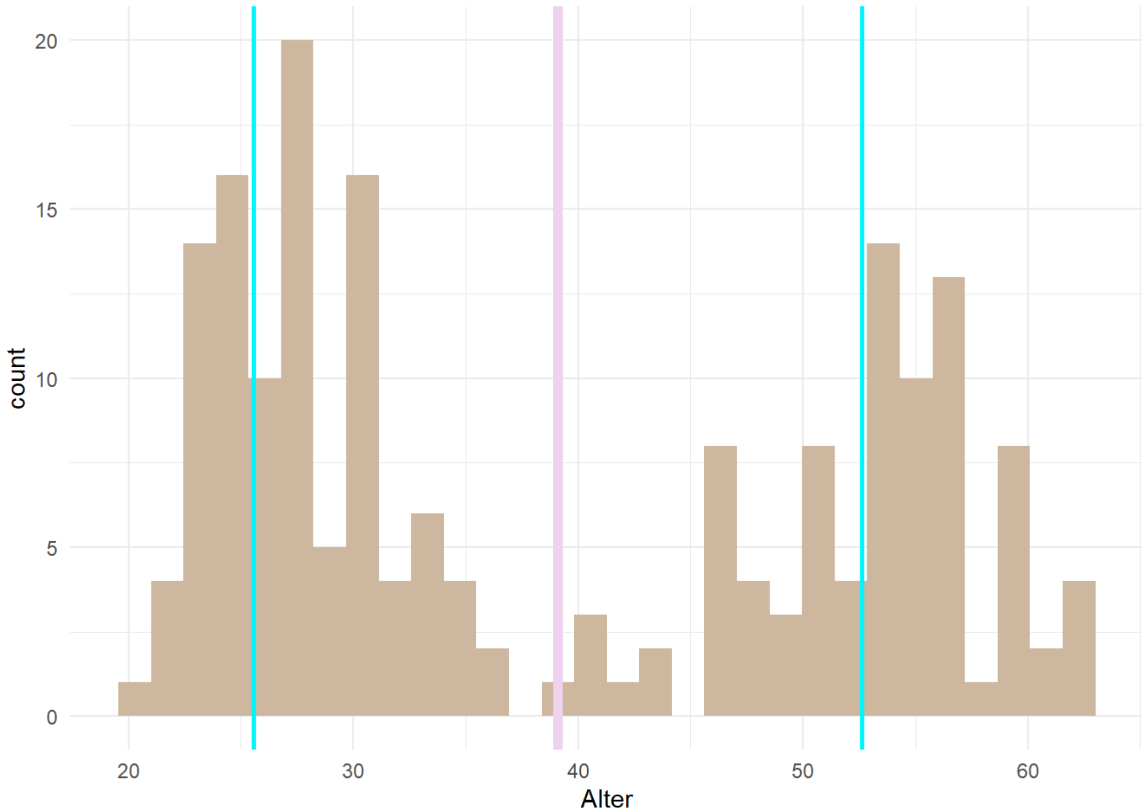
visuelle Darstellung des Alters

```
raw4a %>%
  ggplot()+
  aes(x=Alter)+
  geom_histogram(fill = "bisque3")+
  geom_vline(xintercept=39.10106 , color="thistle2", size=2)+
  geom_vline(xintercept=52.62487, color="turquoise1", size=1)+
  geom_vline(xintercept=25.57725, color="turquoise1", size=1)+
  theme_minimal()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```



Bildung

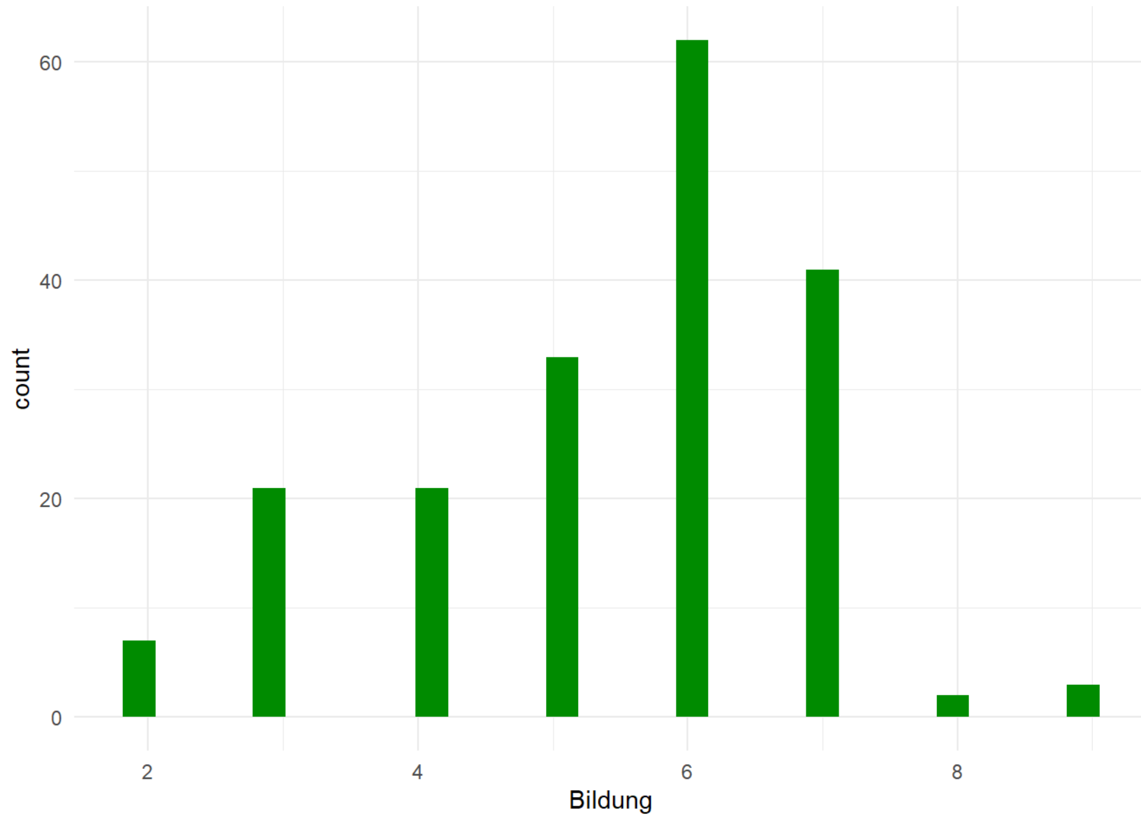
```
raw4a %>%
  count(Bildung) %>%
  mutate(prob = n/sum(n)) %>%
  round(2)
```

```
## # A tibble: 8 × 3
##   Bildung     n  prob
##     <dbl> <dbl> <dbl>
## 1       2     7  0.04
## 2       3    21  0.11
## 3       4    21  0.11
## 4       5    33  0.17
## 5       6    62  0.33
## 6       7    41  0.22
## 7       8     2  0.01
## 8       9     3  0.02
```

Visuelle Darstellung der Bildung

```
raw4a %>%
  ggplot()+
  aes(x=Bildung)+
  geom_histogram(fill = "green4")+
  theme_minimal()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Arbeitszeitmodell = Stunden/Woche

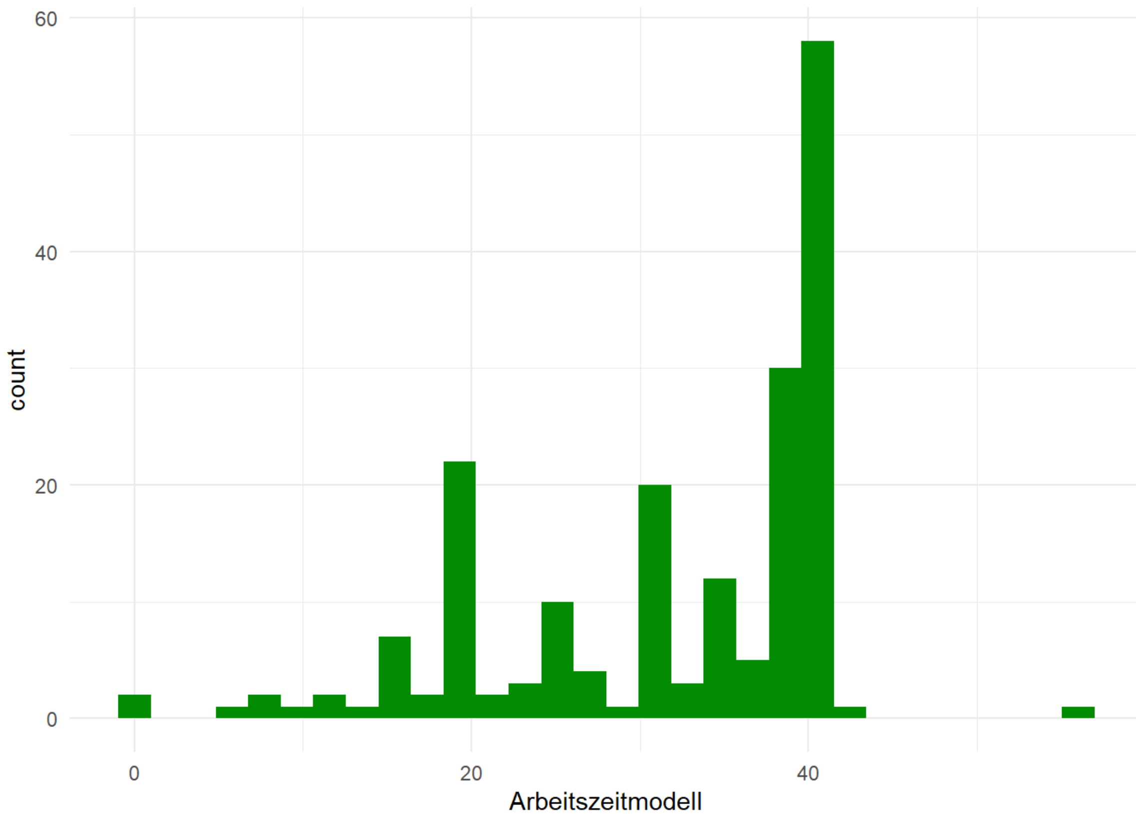
```
raw4a %>%
  summarise(mean = mean(Arbeitszeitmodell, na.rm=TRUE),
            median = median(Arbeitszeitmodell, na.rm=TRUE),
         sd = sd(Arbeitszeitmodell, na.rm=TRUE),
         min = min(Arbeitszeitmodell, na.rm=TRUE),
         max = max(Arbeitszeitmodell, na.rm=TRUE))
```

```
## # A tibble: 1 × 5
##    mean median    sd   min   max
##   <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1  31.8   35.5  9.83     0    56
```

Visuelle Darstellung der verschiedenen Arbeitsmodelle

```
raw4a %>%
  ggplot()+
  aes(x=Arbeitszeitmodell)+
  geom_histogram(fill = "green4")+
  theme_minimal()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



###

Arbeitszeitmodell Extremwerte betrachten

```
raw4a %>%
  filter(Arbeitszeitmodell < 10 | Arbeitszeitmodell > 45) %>%
  arrange(desc(Arbeitszeitmodell)) %>%
  select(ID, Alter, Geschlecht, Bildung, Arbeitszeitmodell, everything())
```

```
## # A tibble: 7 × 73
##      ID Alter Geschlecht Bildung Arbeitszeitmodell Bildung_sonstig    BZG
##   <int> <dbl>      <dbl>   <dbl>             <dbl> <chr>            <dbl>
## 1   101    29          1       5                56  <NA>            2.67
## 2    52    24          1       6                 9  <NA>            5
## 3   182    22          2       6                 8  <NA>            0.167
## 4    63    23          1       6               7.5 <NA>            0.583
## 5    73    23          1       4                 5  <NA>            0.583
## 6   135    46          1       7                 0  <NA>            3.75
## 7   167    59          2       6                 0  <NA>           39.8
## # ℹ 66 more variables: Beschaeftigungsart <dbl>, Gehalt <dbl>,
## #   Position_im_Unternehmen <dbl>, Remotarbeit <dbl>, IS_K01 <dbl>,
## #   IS_K02 <dbl>, IS_K03 <dbl>, IS_K04 <dbl>, IS_E05 <dbl>, IS_E06 <dbl>,
## #   IS_E07 <dbl>, IS_E08 <dbl>, OC_A01 <dbl>, OC_A02 <dbl>, OC_A03 <dbl>,
## #   OC_A04 <dbl>, OC_A05 <dbl>, CPE_C01 <dbl>, CPE_C02 <dbl>, CPE_C03 <dbl>,
## #   CPE_C04 <dbl>, CPE_I05 <dbl>, CPE_I06 <dbl>, CPE_I07 <dbl>, CPE_I08 <dbl>,
## #   CPE_M09 <dbl>, CPE_M10 <dbl>, CPE_M11 <dbl>, CPE_M12 <dbl>, …
```

Gehalt

```
raw4a %>%
  count(Gehalt) %>%
  mutate(prob = n/sum(n)) %>%
  round(2)
```

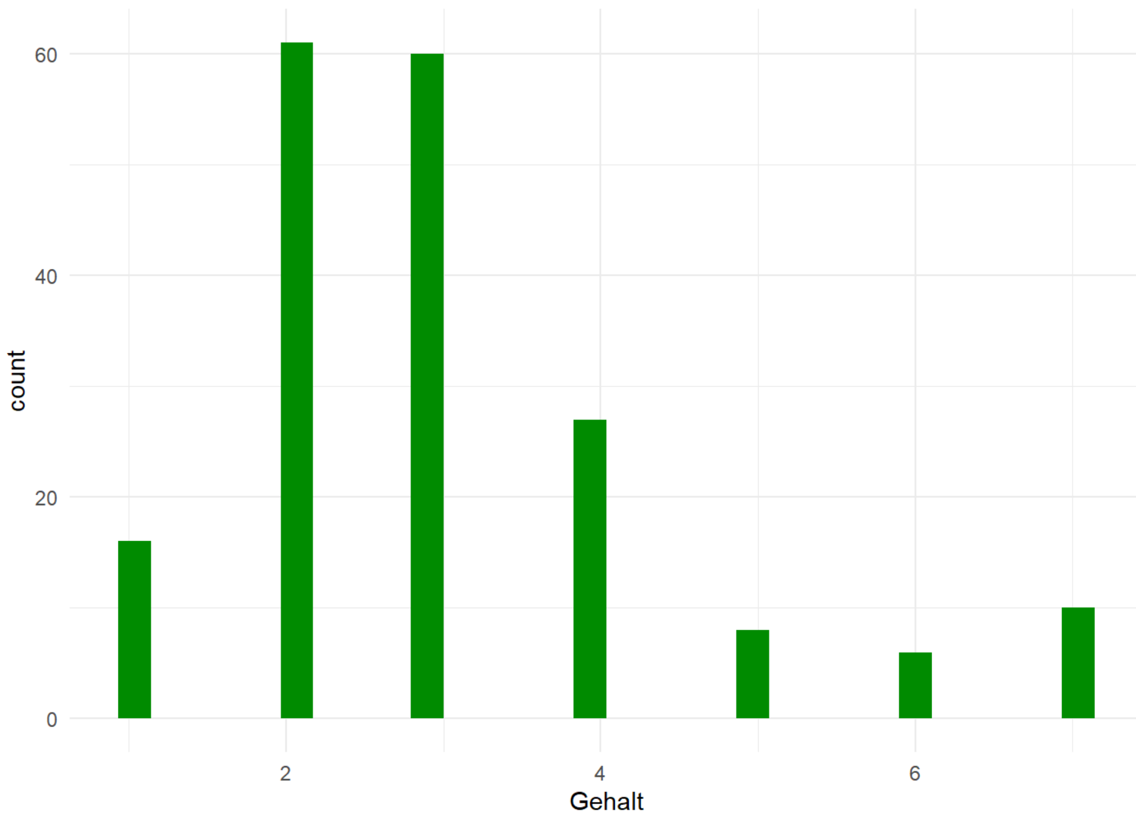
```
## # A tibble: 8 × 3
##   Gehalt     n  prob
##    <dbl> <dbl> <dbl>
## 1      1    16  0.08
## 2      2    61  0.32
## 3      3    60  0.32
## 4      4    27  0.14
## 5      5     8  0.04
## 6      6     6  0.03
## 7      7    10  0.05
## 8     NA     2  0.01
```

Visuelle Darstellung des Gehalts

```
raw4a %>%
  ggplot()+
  aes(x=Gehalt)+
  geom_histogram(fill = "green4")+
  theme_minimal()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 2 rows containing non-finite values (`stat_bin()`).
```



Position im Unternehmen

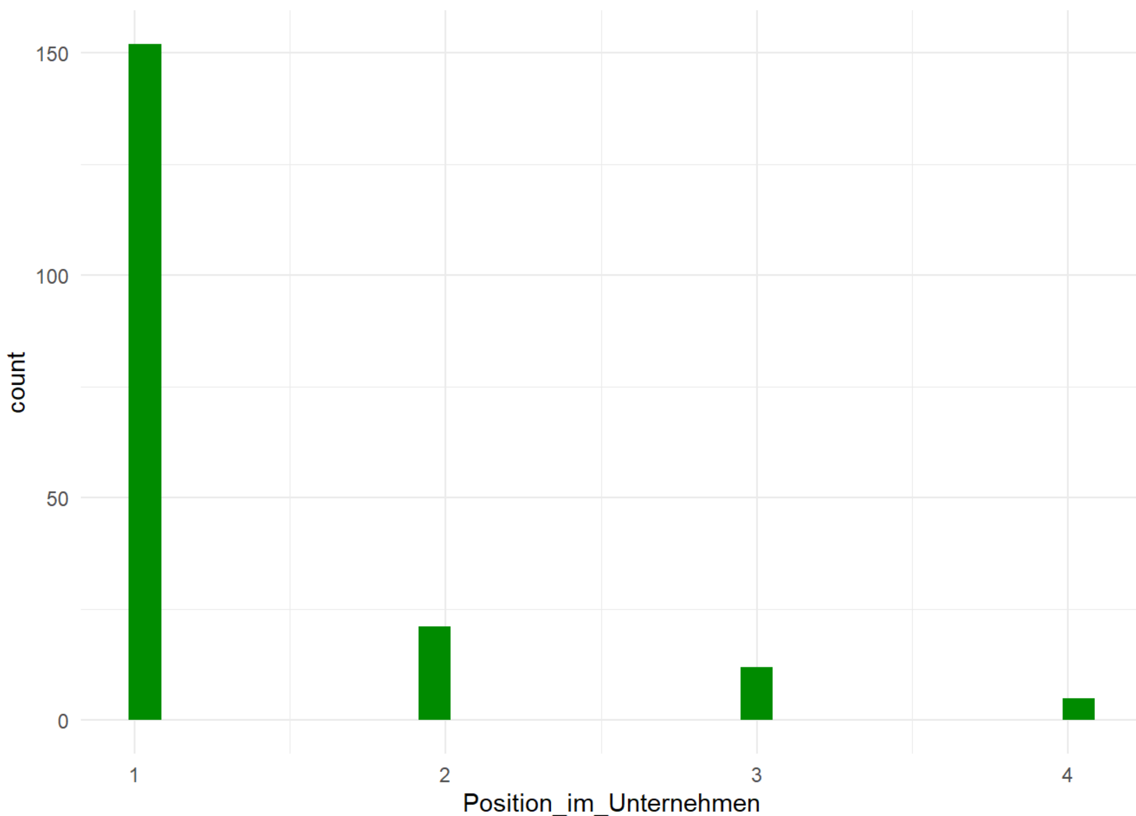
```
raw4a %>%
  count(Position_im_Unternehmen) %>%
  mutate(prob = n/sum(n)) %>%
  round(2)
```

```
## # A tibble: 4 × 3
##   Position_im_Unternehmen     n  prob
##                     <dbl> <dbl> <dbl>
## 1                       1   152  0.8
## 2                       2    21  0.11
## 3                       3    12  0.06
## 4                       4     5  0.03
```

Visuelle Darstellung der Position im Unternehmen

```
raw4a %>%
  ggplot()+
  aes(x=Position_im_Unternehmen)+
  geom_histogram(fill = "green4")+
  theme_minimal()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



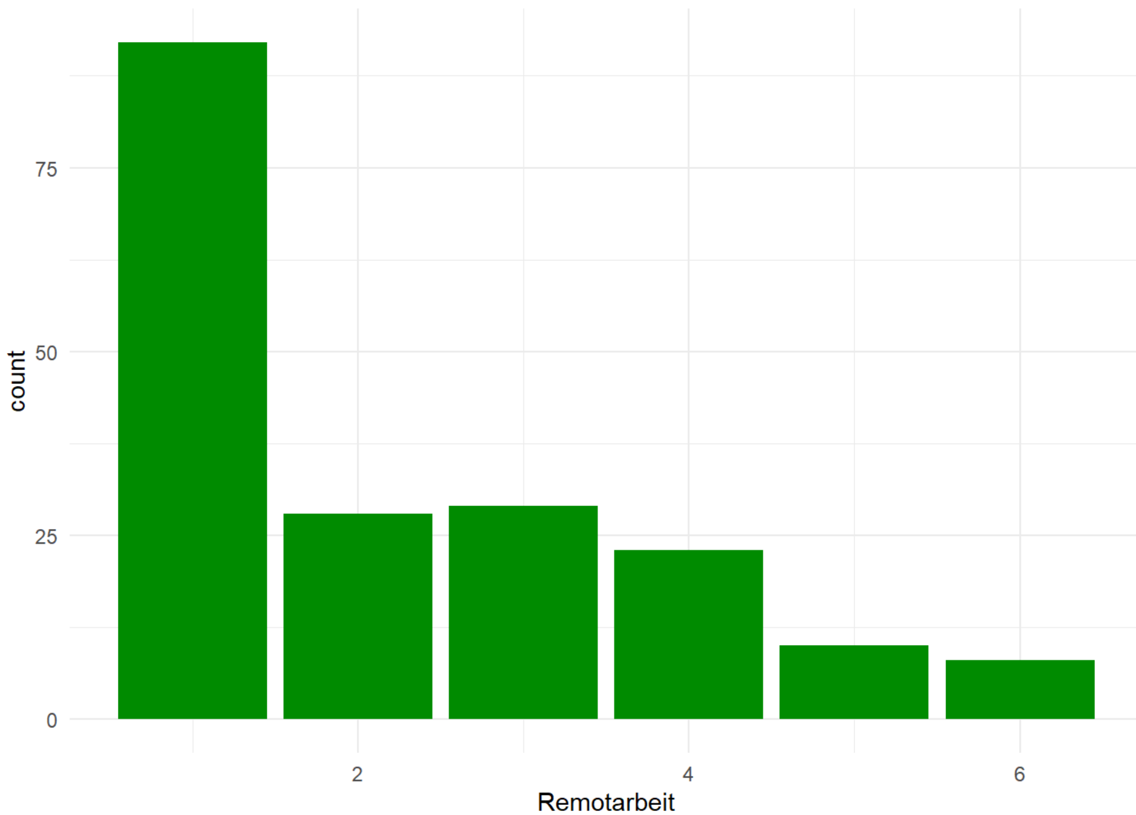
Remotearbeit

```
raw4a %>%
  count(Remotarbeit) %>%
  mutate(prob = n/sum(n)) %>%
  round(2)
```

```
## # A tibble: 6 × 3
##   Remotarbeit     n  prob
##         <dbl> <dbl> <dbl>
## 1           1    92  0.48
## 2           2    28  0.15
## 3           3    29  0.15
## 4           4    23  0.12
## 5           5    10  0.05
## 6           6     8  0.04
```

Visuelle Darstellung der Remotearbeit

```
raw4a %>%
  ggplot()+
  aes(x=Remotarbeit)+
  geom_bar(fill = "green4")+
  theme_minimal()
```



7. Kurzen Datensatz `raw` abspeichern

```
write.csv(raw4a, file="raw4.csv")
```

8. Cronbachs Alpha berechnen

Irritation

```
psych::alpha(subset(raw4a, select=c(IS_K01:IS_E08)), check.keys=TRUE)
```

```
##
## Reliability analysis
## Call: psych::alpha(x = subset(raw4a, select = c(IS_K01:IS_E08)), check.keys = TRUE)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean  sd median_r
##       0.86      0.86    0.89      0.43   6 0.016  3.2 1.2     0.42
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.82  0.86  0.89
## Duhachek  0.83  0.86  0.89
##
##  Reliability if an item is dropped:
##        raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## IS_K01      0.83      0.83    0.86      0.41 4.9    0.020 0.027  0.40
## IS_K02      0.84      0.84    0.86      0.43 5.3    0.018 0.022  0.44
## IS_K03      0.85      0.85    0.88      0.45 5.7    0.017 0.030  0.44
## IS_K04      0.84      0.84    0.87      0.43 5.3    0.018 0.024  0.44
## IS_E05      0.84      0.84    0.88      0.43 5.3    0.018 0.039  0.40
## IS_E06      0.84      0.84    0.87      0.43 5.3    0.018 0.035  0.40
## IS_E07      0.85      0.85    0.87      0.45 5.6    0.017 0.031  0.44
## IS_E08      0.83      0.83    0.87      0.41 4.9    0.019 0.037  0.36
##
##  Item statistics
##          n raw.r std.r r.cor r.drop mean  sd
## IS_K01 190  0.80  0.79  0.78   0.72  3.5 1.9
## IS_K02 190  0.72  0.70  0.68   0.62  4.2 1.8
## IS_K03 190  0.61  0.64  0.57   0.50  2.7 1.5
## IS_K04 190  0.73  0.70  0.67   0.61  3.5 2.0
## IS_E05 190  0.70  0.71  0.64   0.59  2.7 1.7
## IS_E06 190  0.69  0.70  0.66   0.58  3.1 1.7
## IS_E07 190  0.63  0.65  0.60   0.51  3.3 1.7
## IS_E08 190  0.78  0.78  0.74   0.70  2.6 1.6
##
## Non missing response frequency for each item
##           1    2    3    4    5    6    7 miss
## IS_K01 0.15 0.22 0.20 0.09 0.16 0.09 0.08    0
## IS_K02 0.06 0.18 0.14 0.09 0.25 0.17 0.10    0
## IS_K03 0.22 0.33 0.24 0.08 0.09 0.02 0.03    0
## IS_K04 0.18 0.22 0.16 0.06 0.17 0.13 0.08    0
## IS_E05 0.33 0.24 0.14 0.09 0.12 0.06 0.03    0
## IS_E06 0.19 0.25 0.19 0.12 0.17 0.04 0.03    0
## IS_E07 0.16 0.24 0.17 0.11 0.22 0.07 0.03    0
## IS_E08 0.33 0.21 0.22 0.09 0.10 0.04 0.02    0
```

CPE

```
psych::alpha(subset(raw4a, select=c(CPE_C01:CPE_E16)), check.keys=TRUE)
```

```
##
## Reliability analysis
## Call: psych::alpha(x = subset(raw4a, select = c(CPE_C01:CPE_E16)),
##     check.keys = TRUE)
##
##   raw_alpha std.alpha G6(smc) average_r S/N    ase mean  sd median_r
##       0.95      0.95    0.96      0.55  20 0.0051  4.9 1.1     0.57
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.94  0.95  0.96
## Duhachek  0.94  0.95  0.96
##
##  Reliability if an item is dropped:
##         raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r med.r
## CPE_C01      0.95      0.95    0.96      0.54  18   0.0056 0.0146  0.55
## CPE_C02      0.95      0.95    0.96      0.55  19   0.0053 0.0141  0.58
## CPE_C03      0.95      0.95    0.96      0.55  18   0.0055 0.0148  0.56
## CPE_C04      0.95      0.95    0.96      0.54  18   0.0055 0.0141  0.56
## CPE_I05      0.95      0.95    0.96      0.54  18   0.0056 0.0146  0.56
## CPE_I06      0.95      0.95    0.96      0.55  18   0.0055 0.0149  0.57
## CPE_I07      0.95      0.95    0.96      0.59  21   0.0049 0.0052  0.58
## CPE_I08      0.95      0.95    0.96      0.55  18   0.0054 0.0152  0.57
## CPE_M09      0.95      0.95    0.96      0.55  18   0.0055 0.0142  0.57
## CPE_M10      0.95      0.95    0.96      0.55  18   0.0054 0.0142  0.57
## CPE_M11      0.95      0.95    0.96      0.55  18   0.0055 0.0145  0.56
## CPE_M12      0.95      0.95    0.96      0.55  18   0.0054 0.0147  0.57
## CPE_E13      0.95      0.95    0.96      0.54  18   0.0056 0.0138  0.55
## CPE_E14      0.95      0.95    0.96      0.55  18   0.0055 0.0139  0.56
## CPE_E15      0.95      0.95    0.96      0.55  18   0.0055 0.0146  0.56
## CPE_E16      0.95      0.95    0.96      0.56  19   0.0053 0.0140  0.58
##
##  Item statistics
##           n raw.r std.r r.cor r.drop mean  sd
## CPE_C01 190  0.82  0.82  0.81   0.79  4.9 1.5
## CPE_C02 190  0.72  0.72  0.70   0.68  4.9 1.6
## CPE_C03 190  0.79  0.79  0.78   0.76  5.1 1.5
## CPE_C04 190  0.81  0.80  0.79   0.77  4.7 1.5
## CPE_I05 190  0.82  0.83  0.82   0.80  5.1 1.4
## CPE_I06 190  0.78  0.78  0.77   0.75  4.8 1.4
## CPE_I07 190  0.45  0.46  0.40   0.39  5.3 1.4
## CPE_I08 190  0.76  0.77  0.75   0.73  4.8 1.5
## CPE_M09 190  0.78  0.78  0.76   0.74  4.2 1.6
## CPE_M10 190  0.77  0.77  0.76   0.73  4.8 1.6
## CPE_M11 190  0.77  0.78  0.76   0.74  4.7 1.4
## CPE_M12 190  0.74  0.74  0.72   0.70  5.2 1.4
## CPE_E13 190  0.84  0.84  0.83   0.81  4.7 1.6
## CPE_E14 190  0.80  0.79  0.79   0.76  4.9 1.5
## CPE_E15 190  0.79  0.79  0.78   0.76  4.9 1.4
## CPE_E16 190  0.71  0.71  0.69   0.67  4.8 1.6
##
## Non missing response frequency for each item
##            1    2    3    4 4.656085    5    6    7 miss
## CPE_C01 0.03 0.05 0.13 0.10     0.00 0.29 0.26 0.14    0
## CPE_C02 0.04 0.05 0.11 0.13     0.00 0.25 0.25 0.17    0
## CPE_C03 0.04 0.02 0.08 0.17     0.00 0.24 0.28 0.17    0
## CPE_C04 0.05 0.04 0.13 0.14     0.00 0.29 0.26 0.09    0
## CPE_I05 0.02 0.03 0.09 0.22     0.00 0.19 0.31 0.15    0
## CPE_I06 0.03 0.04 0.12 0.19     0.00 0.26 0.25 0.11    0
```

```
## CPE_I07 0.01 0.03 0.07 0.15     0.00 0.20 0.34 0.20    0
## CPE_I08 0.02 0.05 0.15 0.16     0.00 0.22 0.28 0.12    0
## CPE_M09 0.07 0.08 0.17 0.22     0.00 0.22 0.19 0.05    0
## CPE_M10 0.04 0.06 0.11 0.16     0.00 0.27 0.26 0.11    0
## CPE_M11 0.02 0.03 0.16 0.20     0.01 0.30 0.22 0.07    0
## CPE_M12 0.02 0.03 0.07 0.15     0.00 0.25 0.32 0.16    0
## CPE_E13 0.04 0.05 0.14 0.18     0.00 0.25 0.20 0.13    0
## CPE_E14 0.03 0.04 0.12 0.16     0.00 0.24 0.28 0.13    0
## CPE_E15 0.02 0.05 0.12 0.16     0.00 0.25 0.30 0.11    0
## CPE_E16 0.06 0.06 0.09 0.15     0.00 0.26 0.26 0.13    0
```

PS

```
psych::alpha(subset(raw4a, select=c(PS_01:PS_07)), check.keys=TRUE)
```

```
## 
## Reliability analysis   
## Call: psych::alpha(x = subset(raw4a, select = c(PS_01:PS_07)), check.keys = TRUE)
## 
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean  sd median_r
##       0.89      0.89    0.88      0.53 7.9 0.013  5.4 1.2     0.54
## 
##     95% confidence boundaries 
##          lower alpha upper
## Feldt     0.86  0.89  0.91
## Duhachek  0.86  0.89  0.91
## 
##  Reliability if an item is dropped:
##       raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r med.r
## PS_01      0.86      0.86    0.85      0.51 6.2    0.016 0.0050  0.50
## PS_02      0.88      0.88    0.87      0.56 7.6    0.013 0.0027  0.57
## PS_03      0.86      0.87    0.85      0.52 6.4    0.016 0.0060  0.53
## PS_04      0.87      0.87    0.85      0.52 6.6    0.015 0.0054  0.54
## PS_05      0.87      0.87    0.85      0.53 6.8    0.015 0.0040  0.54
## PS_06      0.87      0.88    0.86      0.54 7.1    0.014 0.0053  0.56
## PS_07      0.87      0.87    0.86      0.53 6.8    0.015 0.0061  0.54
## 
##  Item statistics 
##         n raw.r std.r r.cor r.drop mean  sd
## PS_01 190  0.83  0.84  0.81   0.76  5.4 1.5
## PS_02 190  0.70  0.69  0.62   0.57  5.9 1.7
## PS_03 190  0.81  0.81  0.77   0.73  5.4 1.7
## PS_04 190  0.78  0.79  0.74   0.70  4.9 1.5
## PS_05 190  0.78  0.77  0.73   0.67  5.1 1.7
## PS_06 190  0.74  0.74  0.68   0.64  5.9 1.5
## PS_07 190  0.76  0.77  0.72   0.68  5.4 1.4
## 
## Non missing response frequency for each item
##          1    2    3    4    5    6    7 miss
## PS_01 0.02 0.03 0.11 0.04 0.17 0.43 0.22    0
## PS_02 0.05 0.02 0.07 0.06 0.05 0.20 0.56    0
## PS_03 0.03 0.05 0.10 0.07 0.18 0.23 0.34    0
## PS_04 0.03 0.03 0.12 0.21 0.22 0.29 0.11    0
## PS_05 0.03 0.06 0.15 0.07 0.14 0.28 0.27    0
## PS_06 0.02 0.03 0.05 0.01 0.16 0.27 0.45    0
## PS_07 0.02 0.03 0.07 0.10 0.16 0.41 0.21    0
```

OCA

```
psych::alpha(subset(raw4a, select=c(OC_A01:OC_A05)), check.keys=TRUE)
```

```
##
## Reliability analysis
## Call: psych::alpha(x = subset(raw4a, select = c(OC_A01:OC_A05)), check.keys = TRUE)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.91      0.91     0.9      0.67  10 0.011  3.5 0.98     0.67
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.89  0.91  0.93
## Duhachek  0.89  0.91  0.93
##
##  Reliability if an item is dropped:
##        raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r med.r
## OC_A01      0.90      0.90    0.88      0.69 9.0    0.012 0.0056  0.69
## OC_A02      0.90      0.90    0.88      0.69 8.8    0.012 0.0043  0.67
## OC_A03      0.87      0.87    0.85      0.63 6.9    0.015 0.0037  0.64
## OC_A04      0.87      0.88    0.85      0.64 7.1    0.015 0.0035  0.66
## OC_A05      0.89      0.90    0.88      0.68 8.6    0.013 0.0079  0.68
##
##  Item statistics
##          n raw.r std.r r.cor r.drop mean  sd
## OC_A01 190  0.82  0.82  0.75   0.71  3.6 1.2
## OC_A02 190  0.84  0.83  0.77   0.73  3.4 1.2
## OC_A03 190  0.90  0.90  0.89   0.84  3.6 1.1
## OC_A04 190  0.90  0.90  0.88   0.83  3.5 1.1
## OC_A05 190  0.83  0.83  0.77   0.73  3.5 1.1
##
## Non missing response frequency for each item
##           1    2    3    4    5 miss
## OC_A01 0.07 0.07 0.27 0.33 0.25    0
## OC_A02 0.08 0.19 0.14 0.39 0.20    0
## OC_A03 0.06 0.10 0.24 0.38 0.22    0
## OC_A04 0.07 0.13 0.22 0.41 0.18    0
## OC_A05 0.06 0.09 0.24 0.46 0.15    0
```

QQ

QQ (Anand)

```
psych::alpha(subset(raw4a, select=c(QQ_01:QQ_07), check.keys=TRUE))
```

```
## Warning: In subset.data.frame(raw4a, select = c(QQ_01:QQ_07), check.keys = TRUE) :
##  zusätzliches Argument 'check.keys' wird verworfen
```

```
##
## Reliability analysis
## Call: psych::alpha(x = subset(raw4a, select = c(QQ_01:QQ_07), check.keys = TRUE))
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.76      0.77    0.78      0.33 3.4 0.027  2.3 0.73     0.32
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.70  0.76  0.81
## Duhachek  0.71  0.76  0.81
##
##  Reliability if an item is dropped:
##       raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## QQ_01      0.75      0.76    0.76      0.35 3.2    0.028 0.017  0.34
## QQ_02      0.72      0.73    0.72      0.31 2.7    0.031 0.011  0.32
## QQ_03      0.71      0.73    0.73      0.31 2.7    0.033 0.021  0.30
## QQ_04      0.71      0.72    0.72      0.30 2.6    0.033 0.017  0.32
## QQ_05      0.75      0.77    0.77      0.35 3.3    0.028 0.019  0.36
## QQ_06      0.71      0.72    0.72      0.30 2.6    0.033 0.017  0.30
## QQ_07      0.75      0.77    0.75      0.35 3.3    0.028 0.014  0.35
##
##  Item statistics
##         n raw.r std.r r.cor r.drop mean   sd
## QQ_01 190  0.60  0.57  0.46   0.38  2.7 1.36
## QQ_02 190  0.65  0.69  0.65   0.54  1.5 0.83
## QQ_03 190  0.71  0.70  0.63   0.56  2.5 1.18
## QQ_04 190  0.71  0.74  0.69   0.59  2.1 1.03
## QQ_05 190  0.57  0.56  0.43   0.38  2.4 1.17
## QQ_06 190  0.72  0.73  0.68   0.58  2.1 1.11
## QQ_07 190  0.58  0.56  0.46   0.39  2.5 1.21
##
## Non missing response frequency for each item
##          1    2    3    4    5 miss
## QQ_01 0.25 0.28 0.15 0.20 0.12    0
## QQ_02 0.62 0.31 0.03 0.03 0.02    0
## QQ_03 0.18 0.42 0.16 0.17 0.07    0
## QQ_04 0.29 0.44 0.16 0.08 0.03    0
## QQ_05 0.25 0.36 0.23 0.09 0.07    0
## QQ_06 0.32 0.41 0.13 0.11 0.04    0
## QQ_07 0.22 0.37 0.19 0.13 0.08    0
```

QQ-Neu

```
psych::alpha(subset(raw4a, select=c(QQ_N08:QQ_N14), check.keys=TRUE))
```

```
## Warning: In subset.data.frame(raw4a, select = c(QQ_N08:QQ_N14), check.keys = TRUE) :
##  zusätzliches Argument 'check.keys' wird verworfen
```

```
##
## Reliability analysis
## Call: psych::alpha(x = subset(raw4a, select = c(QQ_N08:QQ_N14), check.keys = TRUE))
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.74      0.74    0.73      0.29 2.8 0.029  2.2 0.65     0.31
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.67  0.74  0.79
## Duhachek  0.68  0.74  0.79
##
##  Reliability if an item is dropped:
##        raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r med.r
## QQ_N08      0.74      0.75    0.73      0.33 2.9    0.029 0.0066  0.32
## QQ_N09      0.69      0.69    0.68      0.27 2.3    0.034 0.0127  0.29
## QQ_N10      0.69      0.69    0.68      0.27 2.3    0.035 0.0113  0.28
## QQ_N11      0.71      0.72    0.70      0.30 2.5    0.032 0.0110  0.31
## QQ_N12      0.69      0.69    0.68      0.27 2.2    0.035 0.0130  0.29
## QQ_N13      0.69      0.69    0.68      0.27 2.2    0.034 0.0147  0.29
## QQ_N14      0.71      0.71    0.70      0.29 2.4    0.032 0.0137  0.31
##
##  Item statistics
##          n raw.r std.r r.cor r.drop mean   sd
## QQ_N08 190  0.48  0.48  0.33   0.27  2.8 1.05
## QQ_N09 190  0.68  0.67  0.60   0.51  2.1 1.10
## QQ_N10 190  0.67  0.67  0.60   0.51  2.3 1.06
## QQ_N11 190  0.57  0.59  0.49   0.40  2.2 0.96
## QQ_N12 190  0.70  0.67  0.60   0.52  2.1 1.21
## QQ_N13 190  0.65  0.67  0.59   0.51  1.7 0.87
## QQ_N14 190  0.61  0.61  0.51   0.43  2.3 1.01
##
## Non missing response frequency for each item
##           1    2    3    4    5 miss
## QQ_N08 0.12 0.32 0.27 0.27 0.02    0
## QQ_N09 0.35 0.38 0.13 0.10 0.04    0
## QQ_N10 0.23 0.47 0.16 0.09 0.05    0
## QQ_N11 0.22 0.52 0.16 0.07 0.03    0
## QQ_N12 0.38 0.37 0.06 0.13 0.06    0
## QQ_N13 0.45 0.44 0.05 0.06 0.01    0
## QQ_N14 0.17 0.52 0.16 0.11 0.04    0
```

QQ-Gesamt

```
psych::alpha(subset(raw4a, select=c(QQ_01:QQ_N14), check.keys=TRUE))
```

```
## Warning: In subset.data.frame(raw4a, select = c(QQ_01:QQ_N14), check.keys = TRUE) :
##  zusätzliches Argument 'check.keys' wird verworfen
```

```
## 
## Reliability analysis   
## Call: psych::alpha(x = subset(raw4a, select = c(QQ_01:QQ_N14), check.keys = TRUE))
## 
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.84      0.85    0.87      0.29 5.6 0.017  2.2 0.62     0.29
## 
##     95% confidence boundaries 
##          lower alpha upper
## Feldt     0.81  0.84  0.87
## Duhachek  0.81  0.84  0.88
## 
##  Reliability if an item is dropped:
##        raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## QQ_01       0.84      0.84    0.86      0.29 5.4    0.017 0.017  0.29
## QQ_02       0.83      0.83    0.85      0.27 4.8    0.019 0.014  0.28
## QQ_03       0.83      0.84    0.86      0.28 5.1    0.018 0.018  0.29
## QQ_04       0.83      0.84    0.85      0.28 5.1    0.018 0.017  0.29
## QQ_05       0.84      0.84    0.86      0.29 5.4    0.017 0.017  0.29
## QQ_06       0.82      0.83    0.85      0.27 4.8    0.019 0.016  0.27
## QQ_07       0.84      0.85    0.86      0.30 5.6    0.017 0.016  0.32
## QQ_N08      0.85      0.86    0.87      0.31 5.9    0.016 0.013  0.32
## QQ_N09      0.83      0.84    0.86      0.29 5.2    0.018 0.017  0.29
## QQ_N10      0.83      0.83    0.85      0.28 5.1    0.018 0.018  0.29
## QQ_N11      0.84      0.84    0.86      0.29 5.4    0.017 0.018  0.30
## QQ_N12      0.83      0.83    0.86      0.28 5.0    0.018 0.018  0.28
## QQ_N13      0.83      0.84    0.86      0.28 5.1    0.018 0.018  0.29
## QQ_N14      0.83      0.84    0.86      0.29 5.2    0.018 0.018  0.29
## 
##  Item statistics 
##          n raw.r std.r r.cor r.drop mean   sd
## QQ_01  190  0.55  0.52  0.47   0.42  2.7 1.36
## QQ_02  190  0.71  0.72  0.72   0.66  1.5 0.83
## QQ_03  190  0.62  0.61  0.57   0.53  2.5 1.18
## QQ_04  190  0.62  0.62  0.60   0.54  2.1 1.03
## QQ_05  190  0.52  0.51  0.47   0.41  2.4 1.17
## QQ_06  190  0.73  0.73  0.72   0.66  2.1 1.11
## QQ_07  190  0.46  0.44  0.38   0.34  2.5 1.21
## QQ_N08 190  0.32  0.33  0.25   0.21  2.8 1.05
## QQ_N09 190  0.59  0.59  0.55   0.50  2.1 1.10
## QQ_N10 190  0.64  0.64  0.62   0.56  2.3 1.06
## QQ_N11 190  0.52  0.53  0.48   0.43  2.2 0.96
## QQ_N12 190  0.66  0.65  0.62   0.57  2.1 1.21
## QQ_N13 190  0.62  0.64  0.61   0.55  1.7 0.87
## QQ_N14 190  0.57  0.58  0.53   0.48  2.3 1.01
## 
## Non missing response frequency for each item
##           1    2    3    4    5 miss
## QQ_01  0.25 0.28 0.15 0.20 0.12    0
## QQ_02  0.62 0.31 0.03 0.03 0.02    0
## QQ_03  0.18 0.42 0.16 0.17 0.07    0
## QQ_04  0.29 0.44 0.16 0.08 0.03    0
## QQ_05  0.25 0.36 0.23 0.09 0.07    0
## QQ_06  0.32 0.41 0.13 0.11 0.04    0
## QQ_07  0.22 0.37 0.19 0.13 0.08    0
## QQ_N08 0.12 0.32 0.27 0.27 0.02    0
## QQ_N09 0.35 0.38 0.13 0.10 0.04    0
## QQ_N10 0.23 0.47 0.16 0.09 0.05    0
## QQ_N11 0.22 0.52 0.16 0.07 0.03    0
```

```
## QQ_N12 0.38 0.37 0.06 0.13 0.06    0
## QQ_N13 0.45 0.44 0.05 0.06 0.01    0
## QQ_N14 0.17 0.52 0.16 0.11 0.04    0
```

> Wie bereits im der oberen Analyse der neuen Fragen bereits gesehen, fällt auch in der Gesamtbetrachtung auf, dass gerade das Item `QQ_N08` eine Item-Rest-Korrelation von 0.207 hat, weswegen dieses Item aus dem Fragebogen fällt

QQ_N08 aus dem Datensatz werfen

```
raw4b <- raw4a %>%
  select(-QQ_N08)
```

QQ_Gesamt prüfen

```
psych::alpha(subset(raw4b, select=c(QQ_01:QQ_N14), check.keys=TRUE))
```

```
## Warning: In subset.data.frame(raw4b, select = c(QQ_01:QQ_N14), check.keys = TRUE) :
##  zusätzliches Argument 'check.keys' wird verworfen
```

```
## 
## Reliability analysis   
## Call: psych::alpha(x = subset(raw4b, select = c(QQ_01:QQ_N14), check.keys = TRUE))
## 
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd median_r
##       0.85      0.86    0.87      0.31 5.9 0.016  2.2 0.65     0.32
## 
##     95% confidence boundaries 
##          lower alpha upper
## Feldt     0.81  0.85  0.88
## Duhachek  0.82  0.85  0.88
## 
##  Reliability if an item is dropped:
##        raw_alpha std.alpha G6(smc) average_r S/N alpha se var.r med.r
## QQ_01       0.85      0.85    0.87      0.32 5.7    0.017 0.013  0.32
## QQ_02       0.83      0.84    0.85      0.30 5.1    0.018 0.010  0.30
## QQ_03       0.84      0.84    0.86      0.31 5.4    0.018 0.014  0.32
## QQ_04       0.84      0.84    0.86      0.31 5.4    0.018 0.013  0.32
## QQ_05       0.84      0.85    0.86      0.32 5.7    0.017 0.014  0.32
## QQ_06       0.83      0.84    0.85      0.30 5.1    0.019 0.012  0.29
## QQ_07       0.85      0.86    0.87      0.33 6.0    0.016 0.011  0.34
## QQ_N09      0.84      0.85    0.86      0.32 5.6    0.017 0.013  0.32
## QQ_N10      0.83      0.84    0.86      0.31 5.3    0.018 0.014  0.31
## QQ_N11      0.84      0.85    0.86      0.32 5.7    0.017 0.013  0.32
## QQ_N12      0.83      0.84    0.86      0.31 5.3    0.018 0.013  0.31
## QQ_N13      0.84      0.84    0.86      0.31 5.4    0.017 0.013  0.31
## QQ_N14      0.84      0.85    0.86      0.32 5.5    0.017 0.014  0.32
## 
##  Item statistics 
##          n raw.r std.r r.cor r.drop mean   sd
## QQ_01  190  0.55  0.53  0.47   0.42  2.7 1.36
## QQ_02  190  0.71  0.73  0.73   0.66  1.5 0.83
## QQ_03  190  0.63  0.62  0.58   0.54  2.5 1.18
## QQ_04  190  0.63  0.64  0.61   0.55  2.1 1.03
## QQ_05  190  0.55  0.55  0.49   0.44  2.4 1.17
## QQ_06  190  0.73  0.74  0.72   0.66  2.1 1.11
## QQ_07  190  0.46  0.44  0.38   0.34  2.5 1.21
## QQ_N09 190  0.56  0.57  0.52   0.47  2.1 1.10
## QQ_N10 190  0.65  0.65  0.62   0.57  2.3 1.06
## QQ_N11 190  0.53  0.54  0.49   0.44  2.2 0.96
## QQ_N12 190  0.66  0.65  0.62   0.57  2.1 1.21
## QQ_N13 190  0.61  0.63  0.59   0.54  1.7 0.87
## QQ_N14 190  0.57  0.58  0.53   0.48  2.3 1.01
## 
## Non missing response frequency for each item
##           1    2    3    4    5 miss
## QQ_01  0.25 0.28 0.15 0.20 0.12    0
## QQ_02  0.62 0.31 0.03 0.03 0.02    0
## QQ_03  0.18 0.42 0.16 0.17 0.07    0
## QQ_04  0.29 0.44 0.16 0.08 0.03    0
## QQ_05  0.25 0.36 0.23 0.09 0.07    0
## QQ_06  0.32 0.41 0.13 0.11 0.04    0
## QQ_07  0.22 0.37 0.19 0.13 0.08    0
## QQ_N09 0.35 0.38 0.13 0.10 0.04    0
## QQ_N10 0.23 0.47 0.16 0.09 0.05    0
## QQ_N11 0.22 0.52 0.16 0.07 0.03    0
## QQ_N12 0.38 0.37 0.06 0.13 0.06    0
## QQ_N13 0.45 0.44 0.05 0.06 0.01    0
## QQ_N14 0.17 0.52 0.16 0.11 0.04    0
```

SL

```
psych::alpha(subset(raw4b, select=c(SL_01:SL_07)), check.keys=TRUE)
```

```
##
## Reliability analysis
## Call: psych::alpha(x = subset(raw4b, select = c(SL_01:SL_07)), check.keys = TRUE)
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean  sd median_r
##       0.86      0.86    0.85      0.46   6 0.016  4.5 1.3     0.46
##
##     95% confidence boundaries
##          lower alpha upper
## Feldt     0.82  0.86  0.89
## Duhachek  0.83  0.86  0.89
##
##  Reliability if an item is dropped:
##       raw_alpha std.alpha G6(smc) average_r S/N alpha se  var.r med.r
## SL_01      0.84      0.84    0.83      0.47 5.4    0.018 0.0071  0.46
## SL_02      0.83      0.83    0.81      0.45 4.8    0.019 0.0037  0.42
## SL_03      0.83      0.83    0.81      0.45 4.9    0.019 0.0057  0.43
## SL_04      0.84      0.84    0.83      0.47 5.3    0.018 0.0069  0.46
## SL_05      0.83      0.83    0.81      0.44 4.8    0.019 0.0037  0.41
## SL_06      0.84      0.84    0.83      0.48 5.5    0.017 0.0071  0.46
## SL_07      0.85      0.85    0.84      0.49 5.7    0.017 0.0063  0.48
##
##  Item statistics
##         n raw.r std.r r.cor r.drop mean  sd
## SL_01 190  0.71  0.71  0.63   0.59  4.7 1.7
## SL_02 190  0.79  0.79  0.76   0.69  4.0 1.8
## SL_03 190  0.80  0.77  0.73   0.68  4.6 2.1
## SL_04 190  0.73  0.72  0.65   0.61  4.1 1.8
## SL_05 190  0.79  0.79  0.76   0.70  3.4 1.6
## SL_06 190  0.68  0.70  0.62   0.57  5.4 1.5
## SL_07 190  0.66  0.67  0.57   0.53  5.3 1.7
##
## Non missing response frequency for each item
##          1    2    3 3.42328    4    5    6    7 miss
## SL_01 0.07 0.06 0.12    0.00 0.09 0.27 0.28 0.11    0
## SL_02 0.12 0.12 0.16    0.00 0.16 0.26 0.11 0.08    0
## SL_03 0.15 0.06 0.10    0.00 0.11 0.17 0.17 0.24    0
## SL_04 0.10 0.12 0.14    0.00 0.21 0.17 0.16 0.10    0
## SL_05 0.18 0.10 0.22    0.01 0.26 0.12 0.08 0.03    0
## SL_06 0.03 0.03 0.06    0.00 0.07 0.22 0.36 0.23    0
## SL_07 0.05 0.02 0.09    0.00 0.13 0.16 0.21 0.34    0
```

9. Skalen zusammenfassen

Irriation (IS)

```
d_work <- raw4b
d_work$IS_mean <- rowMeans(subset(raw4b, select = c(IS_K01, IS_K02, IS_K03, IS_K04, IS_E05, IS_E06, IS_E07, IS_E08)))
d_work$IS_K_mean <- rowMeans(subset(raw4b, select = c(IS_K01, IS_K02, IS_K03, IS_K04)))
d_work$IS_E_mean <- rowMeans(subset(raw4b, select = c(IS_E05, IS_E06, IS_E07, IS_E08)))
```

Kultur für psychologisches Empowerment in Organisationen (CPE)

```
d_work$CPE_mean <- rowMeans(subset(raw4b, select = c(CPE_C01, CPE_C02, CPE_C03, CPE_C04, CPE_I05, CPE_I06, CPE_I07, CPE_I08, CPE_M09, CPE_M10, CPE_M11, CPE_M12, CPE_E13, CPE_E14, CPE_E15, CPE_E16)))
d_work$CPE_C_mean <- rowMeans(subset(raw4b, select = c(CPE_C01, CPE_C02, CPE_C03, CPE_C04)))
d_work$CPE_I_mean <- rowMeans(subset(raw4b, select = c(CPE_I05, CPE_I06, CPE_I07, CPE_I08)))
d_work$CPE_M_mean <- rowMeans(subset(raw4b, select = c(CPE_M09, CPE_M10, CPE_M11, CPE_M12)))
d_work$CPE_E_mean <- rowMeans(subset(raw4b, select = c(CPE_E13, CPE_E14, CPE_E15, CPE_E16)))
```

Psychologische Sicherheit (PS)

```
d_work$PS_mean <- rowMeans(subset(raw4b, select = c(PS_01, PS_02, PS_03, PS_04, PS_05, PS_06, PS_07)))
```

Organisationelles Commitment (OCA)

```
d_work$OC_A_mean <- rowMeans(subset(raw4b, select = c(OC_A01, OC_A02, OC_A03, OC_A04, OC_A05)))
```

Quiet Quitting

```
d_work$QQ_Anand_mean <- rowMeans(subset(raw4b, select = c(QQ_01, QQ_02, QQ_03, QQ_04, QQ_05, QQ_06, QQ_07)))
d_work$QQ_Eigene_mean <- rowMeans(subset(raw4b, select = c(QQ_N09, QQ_N10, QQ_N11, QQ_N12, QQ_N13, QQ_N14)))
d_work$QQ_Gesamt_mean <- rowMeans(subset(raw4b, select = c(QQ_01, QQ_02, QQ_03, QQ_04, QQ_05, QQ_06, QQ_07, QQ_N09, QQ_N10, QQ_N11, QQ_N12, QQ_N13, QQ_N14)))
```

Servant Leadership

```
d_work$SL_mean <- rowMeans(subset(raw4b, select = c(SL_01, SL_02, SL_03, SL_04, SL_05, SL_06, SL_07)))
```

Einzelne Items aus dem Datensatz entfernen

```
d_work_s <- d_work %>%
  select(-c(IS_K01:IS_E08, OC_A01:OC_A05, CPE_C01:CPE_E16, PS_01:PS_07, QQ_01:QQ_07, QQ_N09:QQ_N14, SL_01:SL_07, TIME_SUM, TIME_RSI)) %>%
  select(ID, everything())
```

10. Kurzen Datensatz `d_work` abspeichern

```
write.csv(d_work_s, file="d_work_s.csv")
```

Auf NA’s im Datensatz prüfen

```
skim(d_work_s)
```

Data summary

Name	d_work_s
Number of rows	190
Number of columns	28
________________________	
Column type frequency:	
character	1
numeric	27
________________________	
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
Bildung_sonstig	187	0.02	12	17	0	3	0

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
ID	0	1.00	95.50	54.99	1.00	48.25	95.50	142.75	190.00	
Alter	2	0.99	39.10	13.52	21.00	27.00	33.50	53.00	63.00	
Geschlecht	0	1.00	1.69	0.46	1.00	1.00	2.00	2.00	2.00	
Bildung	0	1.00	5.41	1.50	2.00	4.00	6.00	6.00	9.00	
BZG	0	1.00	10.39	12.10	0.08	1.52	4.92	14.23	45.67	
Beschaeftigungsart	0	1.00	1.85	0.36	1.00	2.00	2.00	2.00	2.00	
Arbeitszeitmodell	0	1.00	31.77	9.83	0.00	25.00	35.50	40.00	56.00	
Gehalt	2	0.99	3.04	1.46	1.00	2.00	3.00	4.00	7.00	
Position_im_Unternehmen	0	1.00	1.32	0.71	1.00	1.00	1.00	1.00	4.00	
Remotarbeit	0	1.00	2.24	1.49	1.00	1.00	2.00	3.00	6.00	
ZE01	0	1.00	1.00	0.00	1.00	1.00	1.00	1.00	1.00	
ZE04	0	1.00	1.00	0.00	1.00	1.00	1.00	1.00	1.00	
Time_sum_m	0	1.00	10.13	2.70	4.62	8.14	10.03	11.83	16.93	
IS_mean	0	1.00	3.20	1.22	1.12	2.25	3.00	4.00	7.00	
IS_K_mean	0	1.00	3.48	1.40	1.00	2.25	3.25	4.50	7.00	
IS_E_mean	0	1.00	2.93	1.30	1.00	2.00	2.75	3.75	7.00	

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
CPE_mean	0	1.00	4.86	1.14	1.06	4.08	5.06	5.69	7.00	▁▂▅▇▂
CPE_C_mean	0	1.00	4.91	1.34	1.00	4.25	5.00	6.00	7.00	▁▂▅▇▇
CPE_I_mean	0	1.00	5.01	1.16	1.00	4.25	5.25	6.00	7.00	▁▂▇▇▅
CPE_M_mean	0	1.00	4.69	1.27	1.00	3.75	4.75	5.75	7.00	▁▃▇▇▅
CPE_E_mean	0	1.00	4.82	1.35	1.00	4.00	5.00	5.75	7.00	▁▂▆▇▅
PS_mean	0	1.00	5.43	1.21	1.29	4.86	5.71	6.43	7.00	▁▁▂▇▇
OC_A_mean	0	1.00	3.53	0.98	1.00	3.00	3.80	4.20	5.00	▁▂▂▇▂
QQ_Anand_mean	0	1.00	2.27	0.73	1.00	1.86	2.14	2.68	4.57	▅▇▂▁▁
QQ_Eigene_mean	0	1.00	2.12	0.69	1.00	1.67	2.00	2.33	4.50	▅▇▂▁▁
QQ_Gesamt_mean	0	1.00	2.20	0.65	1.08	1.77	2.08	2.54	4.46	▅▇▂▁▁
SL_mean	0	1.00	4.51	1.28	1.00	3.86	4.71	5.29	7.00	▁▃▇▇▂

11. Korrelationen zwischen den Skalen

```
d_work_s1 <- d_work_s %>%
  select(ID, IS_mean, IS_K_mean, IS_E_mean, CPE_mean, CPE_C_mean, CPE_I_mean, CPE_M_mean, CPE_E_mean, PS_mean
, OC_A_mean, QQ_Anand_mean, QQ_Eigene_mean, QQ_Gesamt_mean, SL_mean)
```

```
d_work_s1 %>%
  select(where(is.numeric)) %>%
  correlate() %>%
  shave()
```

```
## Correlation computed with
## • Method: 'pearson'
## • Missing treated using: 'pairwise.complete.obs'
```

```
## # A tibble: 15 × 16
##    term           ID IS_mean IS_K_mean IS_E_mean CPE_mean CPE_C_mean CPE_I_mean
##    <chr>       <dbl>   <dbl>     <dbl>     <dbl>    <dbl>      <dbl>      <dbl>
##  1 ID        NA       NA       NA         NA       NA         NA         NA
##  2 IS_mean   -0.0625   NA       NA         NA       NA         NA         NA
##  3 IS_K_mean  0.00529   0.910   NA         NA       NA         NA         NA
##  4 IS_E_mean -0.123     0.894    0.627     NA       NA         NA         NA
##  5 CPE_mean   0.112    -0.167   -0.0743    -0.233   NA         NA         NA
##  6 CPE_C_me…  0.0649   -0.200   -0.111     -0.256    0.903     NA         NA
##  7 CPE_I_me…  0.0854   -0.154   -0.0761    -0.207    0.876      0.750     NA
##  8 CPE_M_me…  0.146    -0.135   -0.0546    -0.195    0.889      0.744      0.683
##  9 CPE_E_me…  0.103    -0.106   -0.0247    -0.172    0.894      0.716      0.715
## 10 PS_mean    0.0675   -0.393   -0.321     -0.390    0.585      0.553      0.505
## 11 OC_A_mean  0.0841   -0.222   -0.172     -0.230    0.667      0.583      0.564
## 12 QQ_Anand… -0.0226    0.200    0.100      0.268   -0.603     -0.577     -0.488
## 13 QQ_Eigen… -0.0515    0.136   -0.0181     0.274   -0.355     -0.340     -0.297
## 14 QQ_Gesam… -0.0387    0.187    0.0513     0.294   -0.535     -0.512     -0.438
## 15 SL_mean    0.0636   -0.270   -0.229     -0.259    0.670      0.629      0.543
## # ℹ 8 more variables: CPE_M_mean <dbl>, CPE_E_mean <dbl>, PS_mean <dbl>,
## #   OC_A_mean <dbl>, QQ_Anand_mean <dbl>, QQ_Eigene_mean <dbl>,
## #   QQ_Gesamt_mean <dbl>, SL_mean <dbl>
```

```
subset_cor <- subset(d_work,
                     select = c(ID, IS_mean, IS_K_mean, IS_E_mean, CPE_mean, CPE_C_mean, CPE_I_mean, CPE_M_me
an, CPE_E_mean, PS_mean, OC_A_mean, QQ_Anand_mean, QQ_Eigene_mean, QQ_Gesamt_mean, SL_mean))
```

```
library(apaTables)
```

```
## Warning: Paket 'apaTables' wurde unter R Version 4.3.2 erstellt
```

```
apa.cor.table(subset_cor, filename = "corr-com.doc", table.number = 2)
```

```
## 
## 
## Table 2 
## 
## Means, standard deviations, and correlations with confidence intervals
## 
## 
##   Variable           M     SD    1            2            3           
##   1. ID              95.50 54.99                                         
##                                                                          
##   2. IS_mean         3.20  1.22  -.06                                    
##                                  [-.20, .08]                             
##                                                                          
##   3. IS_K_mean       3.48  1.40  .01          .91**                      
##                                  [-.14, .15] [.88, .93]                  
##                                                                          
##   4. IS_E_mean       2.93  1.30  -.12         .89**        .63**         
##                                  [-.26, .02] [.86, .92]   [.53, .71]     
##                                                                          
##   5. CPE_mean        4.86  1.14  .11          -.17*        -.07          
##                                  [-.03, .25] [-.30, -.03] [-.21, .07]    
##                                                                          
##   6. CPE_C_mean      4.91  1.34  .06          -.20**       -.11          
##                                  [-.08, .21] [-.33, -.06] [-.25, .03]    
##                                                                          
##   7. CPE_I_mean      5.01  1.16  .09          -.15*        -.08          
##                                  [-.06, .23] [-.29, -.01] [-.22, .07]    
##                                                                          
##   8. CPE_M_mean      4.69  1.27  .15*         -.14         -.05          
##                                  [.00, .28]  [-.27, .01]  [-.20, .09]    
##                                                                          
##   9. CPE_E_mean      4.82  1.35  .10          -.11         -.02          
##                                  [-.04, .24] [-.24, .04]  [-.17, .12]    
##                                                                          
##   10. PS_mean        5.43  1.21  .07          -.39**       -.32**        
##                                  [-.08, .21] [-.51, -.27] [-.44, -.19]   
##                                                                          
##   11. OC_A_mean      3.53  0.98  .08          -.22**       -.17*         
##                                  [-.06, .22] [-.35, -.08] [-.31, -.03]   
##                                                                          
##   12. QQ_Anand_mean  2.27  0.73  -.02         .20**        .10           
##                                  [-.16, .12] [.06, .33]   [-.04, .24]    
##                                                                          
##   13. QQ_Eigene_mean 2.12  0.69  -.05         .14          -.02          
##                                  [-.19, .09] [-.01, .27]  [-.16, .12]    
##                                                                          
##   14. QQ_Gesamt_mean 2.20  0.65  -.04         .19**        .05           
##                                  [-.18, .10] [.05, .32]   [-.09, .19]    
##                                                                          
##   15. SL_mean        4.51  1.28  .06          -.27**       -.23**        
##                                  [-.08, .20] [-.40, -.13] [-.36, -.09]   
##                                                                          
##   4            5            6            7            8            9           
## 
## 
## 
## 
## 
## 
```

```
##
##
##
##
##
##   -.23**
##   [-.36, -.09]
##
##   -.26**       .90**
##   [-.38, -.12] [.87, .93]
##
##   -.21**       .88**        .75**
##   [-.34, -.07] [.84, .91]   [.68, .81]
##
##   -.20**       .89**        .74**        .68**
##   [-.33, -.05] [.85, .92]   [.67, .80]   [.60, .75]
##
##   -.17*        .89**        .72**        .71**        .74**
##   [-.31, -.03] [.86, .92]   [.64, .78]   [.64, .78]   [.66, .80]
##
##   -.39**       .59**        .55**        .50**        .50**        .53**
##   [-.50, -.26] [.48, .67]   [.45, .64]   [.39, .60]   [.38, .60]   [.42, .62]
##
##   -.23**       .67**        .58**        .56**        .58**        .64**
##   [-.36, -.09] [.58, .74]   [.48, .67]   [.46, .65]   [.48, .67]   [.55, .72]
##
##   .27**        -.60**       -.58**       -.49**       -.56**       -.52**
##   [.13, .40]   [-.69, -.50] [-.66, -.47] [-.59, -.37] [-.65, -.45] [-.62, -.41]
##
##   .27**        -.35**       -.34**       -.30**       -.30**       -.32**
##   [.14, .40]   [-.47, -.22] [-.46, -.21] [-.42, -.16] [-.42, -.16] [-.45, -.19]
##
##   .29**        -.54**       -.51**       -.44**       -.48**       -.47**
##   [.16, .42]   [-.63, -.43] [-.61, -.40] [-.55, -.31] [-.58, -.36] [-.58, -.35]
##
##   -.26**       .67**        .63**        .54**        .59**        .62**
##   [-.39, -.12] [.58, .74]   [.53, .71]   [.43, .64]   [.49, .68]   [.52, .70]
##
##   10           11           12           13           14
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
##
```

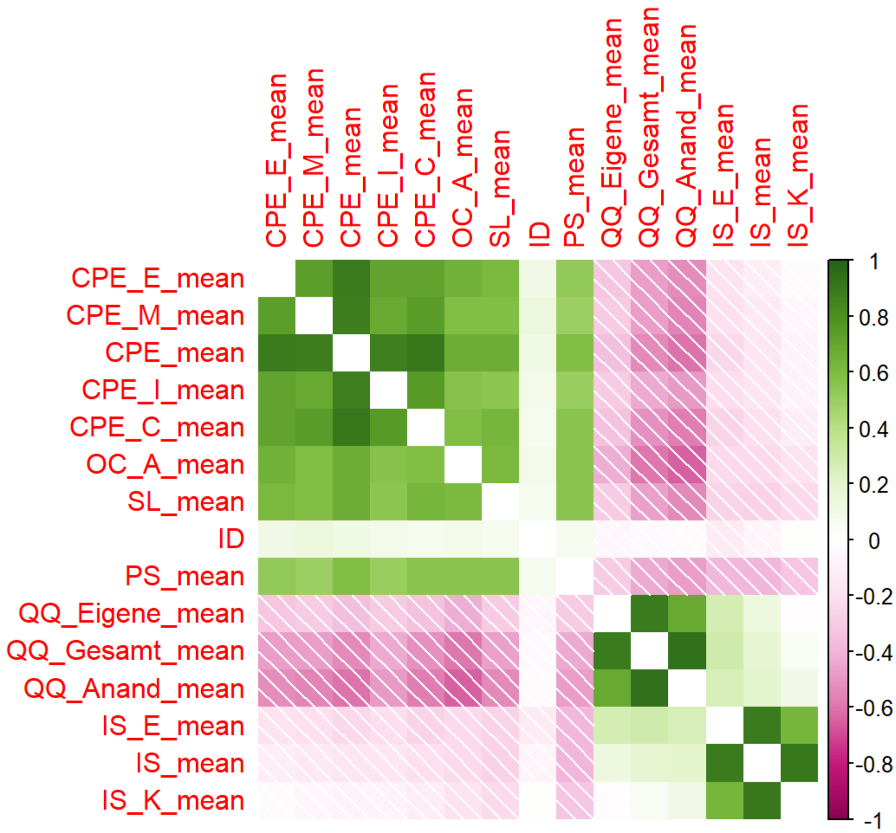
```
##
##
##
##
##
##
##
##
##
##   .55**
##   [.45, .65]
##
##   -.48**       -.65**
##   [-.58, -.36] [-.73, -.56]
##
##   -.30**       -.43**       .69**
##   [-.42, -.16] [-.54, -.30] [.60, .75]
##
##   -.43**       -.60**       .93**        .90**
##   [-.54, -.31] [-.68, -.50] [.91, .95]   [.87, .92]
##
##   .55**        .60**        -.53**       -.29**       -.46**
##   [.45, .64]   [.50, .68]   [-.63, -.42] [-.42, -.16] [-.57, -.34]
##
##
## Note. M and SD are used to represent mean and standard deviation, respectively.
## Values in square brackets indicate the 95% confidence interval.
## The confidence interval is a plausible range of population correlations
## that could have caused the sample correlation (Cumming, 2014).
##  * indicates p < .05. ** indicates p < .01.
##
```

```
M = cor(subset_cor)
M
```

```
##                         ID     IS_mean    IS_K_mean  IS_E_mean    CPE_mean
## ID              1.000000000 -0.06246959  0.005294973 -0.1228372  0.11169338
## IS_mean        -0.062469589  1.00000000  0.909680982  0.8941406 -0.16688901
## IS_K_mean       0.005294973  0.90968098  1.000000000  0.6274135 -0.07427676
## IS_E_mean      -0.122837175  0.89414055  0.627413495  1.0000000 -0.23282509
## CPE_mean        0.111693379 -0.16688901 -0.074276762 -0.2328251  1.00000000
## CPE_C_mean      0.064877070 -0.19999742 -0.110522468 -0.2558218  0.90342756
## CPE_I_mean      0.085409596 -0.15412408 -0.076068594 -0.2069594  0.87597791
## CPE_M_mean      0.145754832 -0.13544632 -0.054622113 -0.1950629  0.88860769
## CPE_E_mean      0.102519289 -0.10571401 -0.024706727 -0.1715709  0.89351884
## PS_mean         0.067458324 -0.39280596 -0.321442616 -0.3899156  0.58510783
## OC_A_mean       0.084083763 -0.22163863 -0.172140409 -0.2299616  0.66732545
## QQ_Anand_mean  -0.022589451  0.20046427  0.100304741  0.2677140 -0.60337318
## QQ_Eigene_mean -0.051498829  0.13582045 -0.018051046  0.2741209 -0.35485688
## QQ_Gesamt_mean -0.038710443  0.18656086  0.051315179  0.2944663 -0.53516238
## SL_mean         0.063642896 -0.26969692 -0.228858472 -0.2589154  0.66950746
##                 CPE_C_mean  CPE_I_mean  CPE_M_mean  CPE_E_mean     PS_mean
## ID              0.06487707  0.08540960  0.14575483  0.10251929  0.06745832
## IS_mean        -0.19999742 -0.15412408 -0.13544632 -0.10571401 -0.39280596
## IS_K_mean      -0.11052247 -0.07606859 -0.05462211 -0.02470673 -0.32144262
## IS_E_mean      -0.25582183 -0.20695936 -0.19506295 -0.17157087 -0.38991561
## CPE_mean        0.90342756  0.87597791  0.88860769  0.89351884  0.58510783
## CPE_C_mean      1.00000000  0.75004270  0.74395491  0.71582639  0.55321446
## CPE_I_mean      0.75004270  1.00000000  0.68261970  0.71469291  0.50481742
## CPE_M_mean      0.74395491  0.68261970  1.00000000  0.73705903  0.49520093
## CPE_E_mean      0.71582639  0.71469291  0.73705903  1.00000000  0.52819872
## PS_mean         0.55321446  0.50481742  0.49520093  0.52819872  1.00000000
## OC_A_mean       0.58347126  0.56392120  0.58337136  0.64206031  0.55401571
## QQ_Anand_mean  -0.57713309 -0.48785188 -0.55530328 -0.52426866 -0.47974571
## QQ_Eigene_mean -0.34032665 -0.29702437 -0.29944552 -0.32420787 -0.29966373
## QQ_Gesamt_mean -0.51232959 -0.43764042 -0.47926338 -0.47275813 -0.43407030
## SL_mean         0.62906337  0.54286467  0.58951074  0.61655350  0.55277011
##                  OC_A_mean QQ_Anand_mean QQ_Eigene_mean QQ_Gesamt_mean
## ID              0.08408376   -0.02258945    -0.05149883    -0.03871044
## IS_mean        -0.22163863    0.20046427     0.13582045     0.18656086
## IS_K_mean      -0.17214041    0.10030474    -0.01805105     0.05131518
## IS_E_mean      -0.22996159    0.26771396     0.27412091     0.29446632
## CPE_mean        0.66732545   -0.60337318    -0.35485688    -0.53516238
## CPE_C_mean      0.58347126   -0.57713309    -0.34032665    -0.51232959
## CPE_I_mean      0.56392120   -0.48785188    -0.29702437    -0.43764042
## CPE_M_mean      0.58337136   -0.55530328    -0.29944552    -0.47926338
## CPE_E_mean      0.64206031   -0.52426866    -0.32420787    -0.47275813
## PS_mean         0.55401571   -0.47974571    -0.29966373    -0.43407030
## OC_A_mean       1.00000000   -0.64977514    -0.42803591    -0.59874435
## QQ_Anand_mean  -0.64977514    1.00000000     0.68556745     0.93457206
## QQ_Eigene_mean -0.42803591    0.68556745     1.00000000     0.89971882
## QQ_Gesamt_mean -0.59874435    0.93457206     0.89971882     1.00000000
## SL_mean         0.60020417   -0.53177398    -0.29176394    -0.46140270
##                  SL_mean
## ID              0.0636429
## IS_mean        -0.2696969
## IS_K_mean      -0.2288585
## IS_E_mean      -0.2589154
## CPE_mean        0.6695075
## CPE_C_mean      0.6290634
## CPE_I_mean      0.5428647
## CPE_M_mean      0.5895107
## CPE_E_mean      0.6165535
## PS_mean         0.5527701
```

```
## OC_A_mean       0.6002042
## QQ_Anand_mean   -0.5317740
## QQ_Eigene_mean -0.2917639
## QQ_Gesamt_mean -0.4614027
## SL_mean         1.0000000
```

```
corrplot(M, method = 'shade', order = 'AOE', diag = FALSE, col=COL2('PiYG'))
```



12. Überprüfung der Hypothesen

Organisationales Commitment - affektiv `OCA`

QQ-Anand

```
lm1 <- lm(QQ_Anand_mean ~ OC_A_mean + Alter + Geschlecht + BZG + Gehalt, data=d_work_s)
summary(lm1)
```

```
## 
## Call:
## lm(formula = QQ_Anand_mean ~ OC_A_mean + Alter + Geschlecht + 
##     BZG + Gehalt, data = d_work_s)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.11526 -0.36806 -0.01861  0.29209  1.90326 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  4.4469392  0.2649841  16.782   <2e-16 ***
## OC_A_mean   -0.4730074  0.0415828 -11.375   <2e-16 ***
## Alter       -0.0038309  0.0041052  -0.933    0.352    
## Geschlecht  -0.0837481  0.0942494  -0.889    0.375    
## BZG          0.0002857  0.0046106   0.062    0.951    
## Gehalt      -0.0717429  0.0319550  -2.245    0.026 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.548 on 181 degrees of freedom
##   (3 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.4526, Adjusted R-squared:  0.4375 
## F-statistic: 29.93 on 5 and 181 DF,  p-value: < 2.2e-16
```

```
d_work %>%
  filter(is.na(Gehalt & Alter))
```

```
## # A tibble: 3 × 86
##      ID Alter Geschlecht Bildung Bildung_sonstig   BZG Beschaeftigungsart
##   <int> <dbl>      <dbl>   <dbl> <chr>           <dbl>              <dbl>
## 1   115    31          1       2 <NA>              1.5                  2
## 2   142    NA          2       7 <NA>             11                    2
## 3   154    NA          2       7 <NA>              7                    2
## # ℹ 79 more variables: Arbeitszeitmodell <dbl>, Gehalt <dbl>,
## #   Position_im_Unternehmen <dbl>, Remotarbeit <dbl>, IS_K01 <dbl>,
## #   IS_K02 <dbl>, IS_K03 <dbl>, IS_K04 <dbl>, IS_E05 <dbl>, IS_E06 <dbl>,
## #   IS_E07 <dbl>, IS_E08 <dbl>, OC_A01 <dbl>, OC_A02 <dbl>, OC_A03 <dbl>,
## #   OC_A04 <dbl>, OC_A05 <dbl>, CPE_C01 <dbl>, CPE_C02 <dbl>, CPE_C03 <dbl>,
## #   CPE_C04 <dbl>, CPE_I05 <dbl>, CPE_I06 <dbl>, CPE_I07 <dbl>, CPE_I08 <dbl>,
## #   CPE_M09 <dbl>, CPE_M10 <dbl>, CPE_M11 <dbl>, CPE_M12 <dbl>, …
```

> Wie im Abschnit 4 gesehen haben wir bei Alter und Gehalt jeweils 2 NA’s, jetzt zeigt sich, dass diese drei Personen sich daraus ergeben, dass eine Person fehlende Werte im Alter sowie Gehalt hat.

Gesamte QQ-Skala

```
lm2 <- lm(QQ_Gesamt_mean ~ OC_A_mean + Alter + Geschlecht + BZG + Gehalt, data=d_work_s)
summary(lm2)
```

```
##
## Call:
## lm(formula = QQ_Gesamt_mean ~ OC_A_mean + Alter + Geschlecht +
##     BZG + Gehalt, data = d_work_s)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.19541 -0.28365 -0.06557  0.26425  2.15441
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.073316   0.247445  16.461  < 2e-16 ***
## OC_A_mean   -0.387574   0.038831  -9.981  < 2e-16 ***
## Alter       -0.005953   0.003833  -1.553  0.12219
## Geschlecht  -0.031366   0.088011  -0.356  0.72197
## BZG          0.002949   0.004305   0.685  0.49423
## Gehalt      -0.082043   0.029840  -2.749  0.00658 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5117 on 181 degrees of freedom
##   (3 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.4082, Adjusted R-squared:  0.3918
## F-statistic: 24.97 on 5 and 181 DF,  p-value: < 2.2e-16
```

Irritation `IS`

QQ-Anand

```
lm3 <- lm(QQ_Anand_mean ~ IS_mean + Alter + Geschlecht + BZG + Gehalt, data=d_work_s)
summary(lm3)
```

```
##
## Call:
## lm(formula = QQ_Anand_mean ~ IS_mean + Alter + Geschlecht + BZG +
##     Gehalt, data = d_work_s)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.42948 -0.47547 -0.05748  0.39416  2.14894
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.627191   0.328234   8.004  1.4e-13 ***
## IS_mean      0.110707   0.042873   2.582   0.0106 *
## Alter       -0.005239   0.005295  -0.989   0.3238
## Geschlecht  -0.132148   0.121648  -1.086   0.2788
## BZG         -0.001302   0.005935  -0.219   0.8266
## Gehalt      -0.090335   0.041037  -2.201   0.0290 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7047 on 181 degrees of freedom
##   (3 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.09468,    Adjusted R-squared:  0.06968
## F-statistic: 3.786 on 5 and 181 DF,  p-value: 0.002762
```

Gesamte QQ-Skala

```
lm4 <- lm(QQ_Gesamt_mean ~ IS_mean + Alter + Geschlecht + BZG + Gehalt, data=d_work_s)
summary(lm4)
```

```
##
## Call:
## lm(formula = QQ_Gesamt_mean ~ IS_mean + Alter + Geschlecht +
##     BZG + Gehalt, data = d_work_s)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.42365 -0.41154 -0.05904  0.26311  2.19960
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.590090   0.292477   8.856 7.57e-16 ***
## IS_mean      0.088271   0.038203   2.311  0.02198 *
## Alter       -0.007136   0.004718  -1.512  0.13219
## Geschlecht  -0.070421   0.108396  -0.650  0.51673
## BZG          0.001663   0.005288   0.315  0.75346
## Gehalt      -0.097309   0.036567  -2.661  0.00849 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6279 on 181 degrees of freedom
##   (3 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.1087, Adjusted R-squared:  0.08408
## F-statistic: 4.415 on 5 and 181 DF,  p-value: 0.0008107
```

Kultur für psychologisches Empowerment - `CPE`

QQ-Anand

```
lm5 <- lm(QQ_Anand_mean ~ CPE_mean + Alter + Geschlecht + BZG + Gehalt, data=d_work_s)
summary(lm5)
```

```
##
## Call:
## lm(formula = QQ_Anand_mean ~ CPE_mean + Alter + Geschlecht +
##     BZG + Gehalt, data = d_work_s)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.50289 -0.33767 -0.02374  0.30041  1.69036
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.743983   0.292477  16.220   <2e-16 ***
## CPE_mean    -0.382623   0.036527 -10.475   <2e-16 ***
## Alter       -0.006345   0.004235  -1.498   0.1358
## Geschlecht  -0.092870   0.097372  -0.954   0.3415
## BZG         -0.003953   0.004774  -0.828   0.4087
## Gehalt      -0.055934   0.033145  -1.688   0.0932 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5662 on 181 degrees of freedom
##   (3 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.4156, Adjusted R-squared:  0.3995
## F-statistic: 25.75 on 5 and 181 DF,  p-value: < 2.2e-16
```

Gesamte QQ-Skala

```
lm6 <- lm(QQ_Gesamt_mean ~ CPE_mean + Alter + Geschlecht + BZG + Gehalt, data=d_work_s)
summary(lm6)
```

```
##
## Call:
## lm(formula = QQ_Gesamt_mean ~ CPE_mean + Alter + Geschlecht +
##     BZG + Gehalt, data = d_work_s)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.14996 -0.36084 -0.03586  0.22999  2.07595
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.2591366  0.2760972  15.426  < 2e-16 ***
## CPE_mean    -0.3010045  0.0344810  -8.730 1.66e-15 ***
## Alter       -0.0080196  0.0039974  -2.006   0.0463 *
## Geschlecht  -0.0392306  0.0919191  -0.427   0.6700
## BZG         -0.0004145  0.0045066  -0.092   0.9268
## Gehalt      -0.0702610  0.0312886  -2.246   0.0259 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5345 on 181 degrees of freedom
##   (3 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.3543, Adjusted R-squared:  0.3364
## F-statistic: 19.86 on 5 and 181 DF,  p-value: 9.26e-16
```

Psychologische Sicherheit `PS`

QQ-Anand

```
lm7 <- lm(QQ_Anand_mean ~ PS_mean + Alter + Geschlecht + BZG + Gehalt, data=d_work_s)
summary(lm7)
```

```
## 
## Call:
## lm(formula = QQ_Anand_mean ~ PS_mean + Alter + Geschlecht + BZG + 
##     Gehalt, data = d_work_s)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -1.18597 -0.39872 -0.02222  0.35526  2.34868 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  4.3994910  0.3298862  13.336  < 2e-16 ***
## PS_mean     -0.2790443  0.0382586  -7.294 9.11e-12 ***
## Alter       -0.0065421  0.0047180  -1.387   0.1673    
## Geschlecht  -0.0859853  0.1085131  -0.792   0.4292    
## BZG         -0.0008736  0.0053072  -0.165   0.8694    
## Gehalt      -0.0682820  0.0368728  -1.852   0.0657 .  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 0.6308 on 181 degrees of freedom
##   (3 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.2745, Adjusted R-squared:  0.2545 
## F-statistic:  13.7 on 5 and 181 DF,  p-value: 2.409e-11
```

Gesamte QQ-Skala

```
lm8 <- lm(QQ_Gesamt_mean ~ PS_mean + Alter + Geschlecht + BZG + Gehalt, data=d_work_s)
summary(lm8)
```

```
##
## Call:
## lm(formula = QQ_Gesamt_mean ~ PS_mean + Alter + Geschlecht +
##     BZG + Gehalt, data = d_work_s)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.1329 -0.3914 -0.0705  0.3059  2.3605
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.010999   0.300657  13.341  < 2e-16 ***
## PS_mean     -0.224026   0.034869  -6.425 1.13e-09 ***
## Alter       -0.008175   0.004300  -1.901   0.0589 .
## Geschlecht  -0.033510   0.098899  -0.339   0.7351
## BZG          0.002004   0.004837   0.414   0.6792
## Gehalt      -0.079596   0.033606  -2.369   0.0189 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5749 on 181 degrees of freedom
##   (3 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.2528, Adjusted R-squared:  0.2322
## F-statistic: 12.25 on 5 and 181 DF,  p-value: 3.096e-10
```

Servant Leadership `SL`

QQ-Anand

```
lm9 <- lm(QQ_Anand_mean ~ SL_mean + Alter + Geschlecht + BZG + Gehalt, data=d_work_s)
summary(lm9)
```

```
##
## Call:
## lm(formula = QQ_Anand_mean ~ SL_mean + Alter + Geschlecht + BZG +
##     Gehalt, data = d_work_s)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.2286 -0.3841 -0.0329  0.2746  1.8034
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  4.423002   0.300459  14.721  < 2e-16 ***
## SL_mean     -0.309026   0.034759  -8.891 6.09e-16 ***
## Alter       -0.010633   0.004501  -2.363   0.0192 *
## Geschlecht  -0.085881   0.102972  -0.834   0.4054
## BZG          0.002785   0.005051   0.551   0.5820
## Gehalt      -0.074466   0.034913  -2.133   0.0343 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5987 on 181 degrees of freedom
##   (3 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.3466, Adjusted R-squared:  0.3286
## F-statistic: 19.21 on 5 and 181 DF,  p-value: 2.595e-15
```

Gesamte QQ-Skala

```
lm10 <- lm(QQ_Gesamt_mean ~ SL_mean + Alter + Geschlecht + BZG + Gehalt, data=d_work_s)
summary(lm10)
```

```
##
## Call:
## lm(formula = QQ_Gesamt_mean ~ SL_mean + Alter + Geschlecht +
##     BZG + Gehalt, data = d_work_s)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.2210 -0.3654 -0.0326  0.2557  2.0685
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.998383   0.279654  14.298  < 2e-16 ***
## SL_mean     -0.241339   0.032352  -7.460 3.49e-12 ***
## Alter       -0.011370   0.004189  -2.714  0.00729 **
## Geschlecht  -0.033841   0.095842  -0.353  0.72443
## BZG          0.004867   0.004701   1.035  0.30193
## Gehalt      -0.084939   0.032496  -2.614  0.00971 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5572 on 181 degrees of freedom
##   (3 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.2982, Adjusted R-squared:  0.2788
## F-statistic: 15.38 on 5 and 181 DF,  p-value: 1.356e-12
```

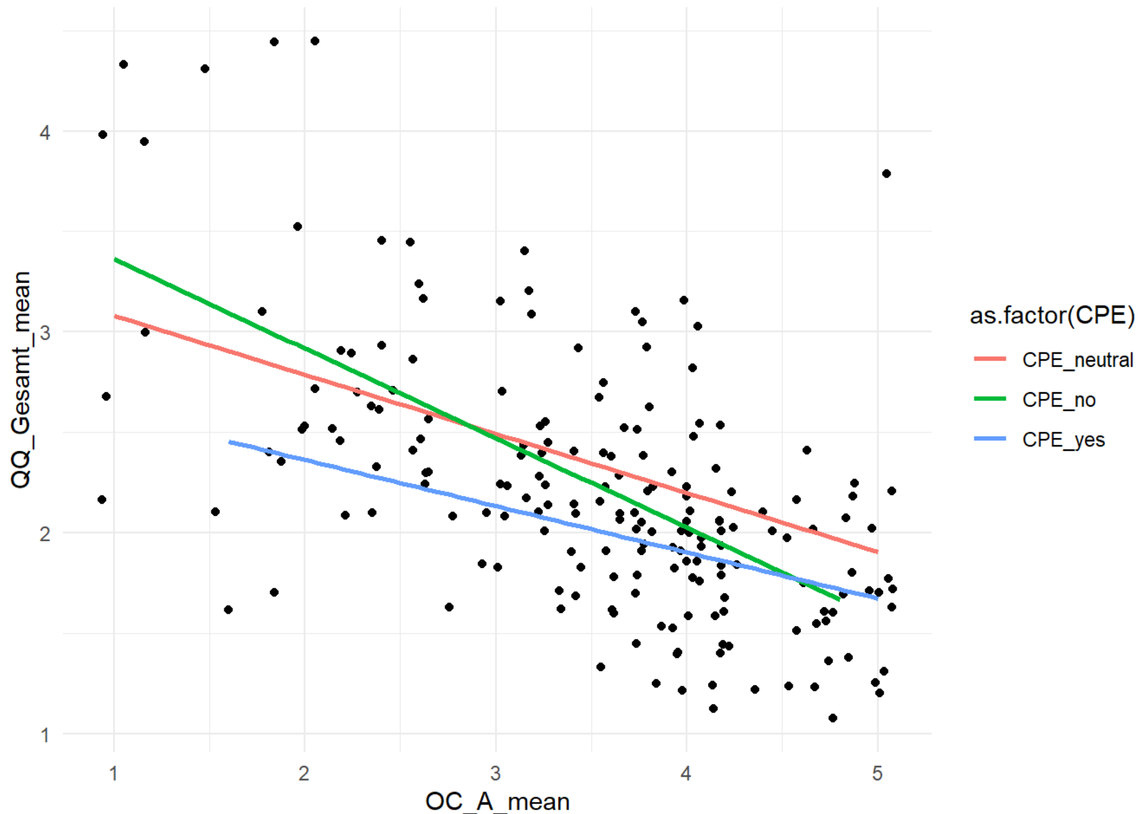
Moderation: CPE moderiert den Zusammenhang von OCA und QQ

```
d_work_plot_lm<- d_work_s %>%
  mutate(CPE = case_when(
    CPE_mean <= 3.99 ~ "CPE_no",
    CPE_mean > 3.99 & CPE_mean < 4.99 ~ "CPE_neutral",
    CPE_mean >= 4.99 ~ "CPE_yes"
  ))
```

Visuelle Darstellung des möglichen Zusammenhangs

```
ggplot(d_work_plot_lm)+
  geom_jitter(mapping = aes(x=OC_A_mean, y=QQ_Gesamt_mean))+
  geom_smooth(mapping = aes(x=OC_A_mean, y=QQ_Gesamt_mean, color=as.factor(CPE)), method = "lm", se=FALSE)+
  theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



QQ_Anand

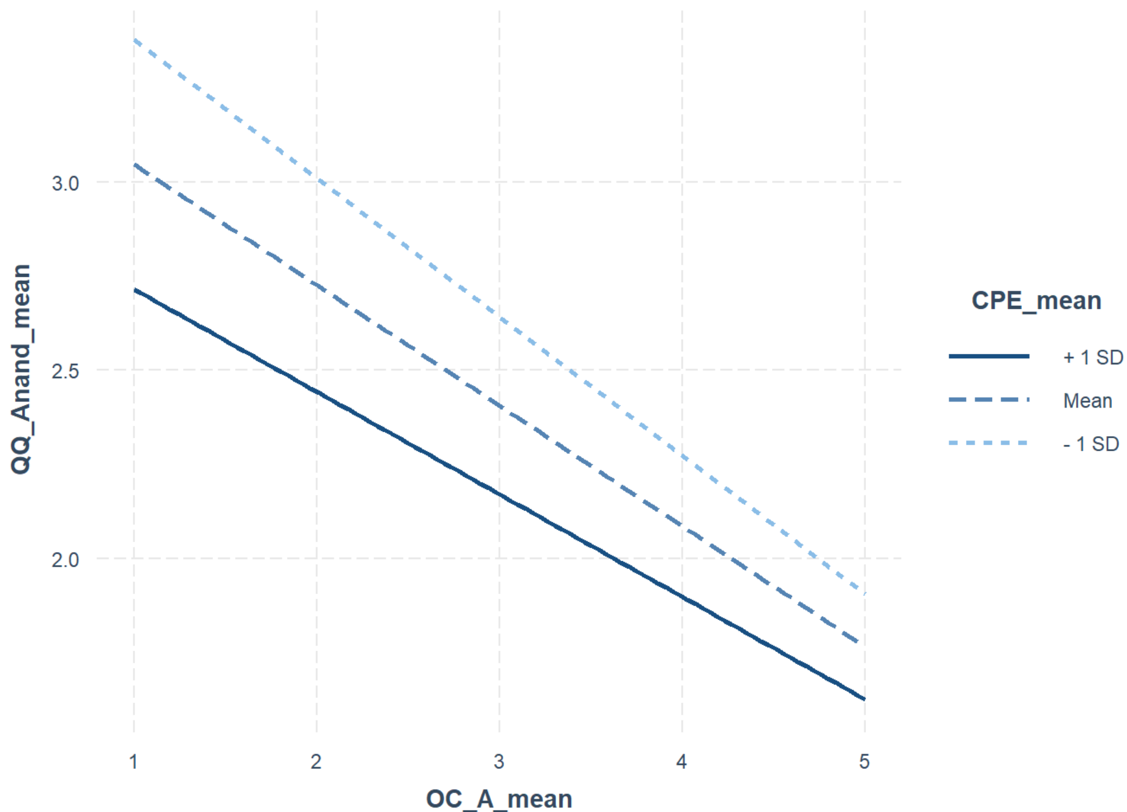
```
lm11 <- lm(QQ_Anand_mean ~ OC_A_mean*CPE_mean, data=d_work_s)
summary(lm11)
```

```
##
## Call:
## lm(formula = QQ_Anand_mean ~ OC_A_mean * CPE_mean, data = d_work_s)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.59046 -0.35593 -0.02728  0.31523  1.79251
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          4.98111    0.49461  10.071  < 2e-16 ***
## OC_A_mean           -0.52378    0.15856  -3.303  0.00115 **
## CPE_mean            -0.33238    0.11605  -2.864  0.00466 **
## OC_A_mean:CPE_mean   0.04197    0.03262   1.287  0.19972
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.528 on 186 degrees of freedom
## Multiple R-squared:  0.4788, Adjusted R-squared:  0.4704
## F-statistic: 56.96 on 3 and 186 DF,  p-value: < 2.2e-16
```

```
library(interactions)
```

```
## Warning: Paket 'interactions' wurde unter R Version 4.3.3 erstellt
```

```
interact_plot(model=lm11, pred=OC_A_mean, modx=CPE_mean)
```

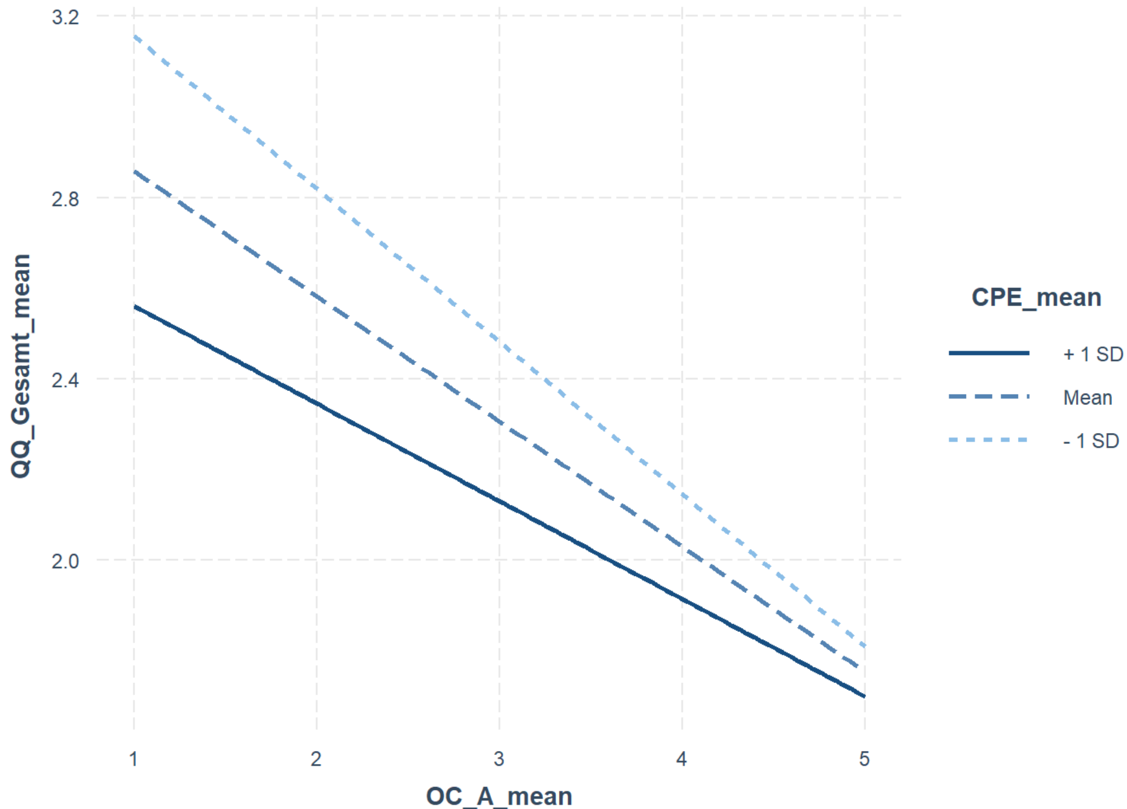


QQ_Gesamt

```
lm12 <- lm(QQ_Gesamt_mean ~ OC_A_mean*CPE_mean, data=d_work_s)
summary(lm12)
```

```
##
## Call:
## lm(formula = QQ_Gesamt_mean ~ OC_A_mean * CPE_mean, data = d_work_s)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.48917 -0.35063 -0.03933  0.24299  2.08243
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)         4.66278    0.47625   9.791  < 2e-16 ***
## OC_A_mean          -0.53529    0.15268  -3.506  0.00057 ***
## CPE_mean           -0.31443    0.11175  -2.814  0.00542 **
## OC_A_mean:CPE_mean  0.05330    0.03141   1.697  0.09135 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5084 on 186 degrees of freedom
## Multiple R-squared:  0.4009, Adjusted R-squared:  0.3913
## F-statistic: 41.49 on 3 and 186 DF,  p-value: < 2.2e-16
```

```
interact_plot(model=lm12, pred=OC_A_mean, modx=CPE_mean)
```



Moderation: Alter moderiert den Zusammenhang von OCA und QQ-Gesamt

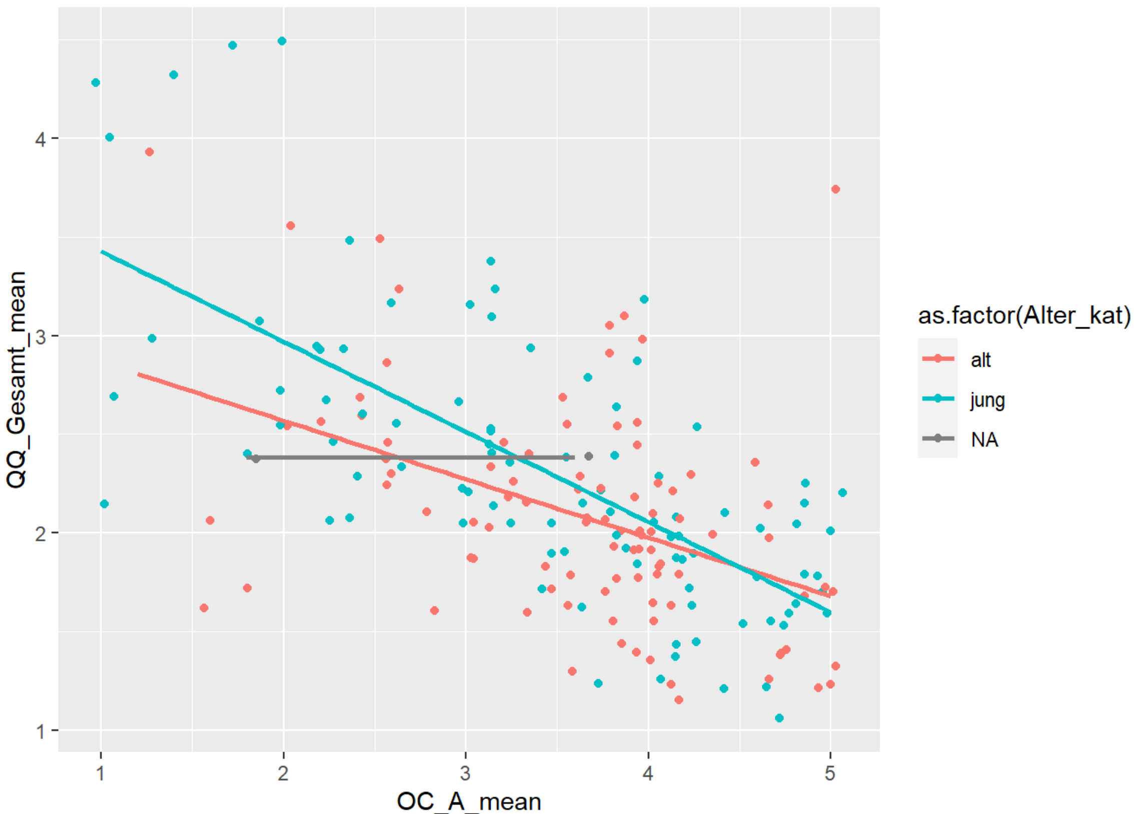
Alter in Kategorien unterteilen

```
d_work_plot_lm_1 <- d_work_plot_lm %>%
  mutate(Alter_kat = case_when(
    Alter <= 33.5 ~ "jung",
    Alter > 33.5 ~ "alt"
  ))
```

> Hierfür wurde der mediane Wert des Alters genutzt

```
ggplot(d_work_plot_lm_1)+
  geom_jitter(mapping = aes(x=OC_A_mean, y=QQ_Gesamt_mean, color=as.factor(Alter_kat)))+
  geom_smooth(mapping = aes(x=OC_A_mean, y=QQ_Gesamt_mean, color=as.factor(Alter_kat)), method = "lm", se=FAL
SE)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

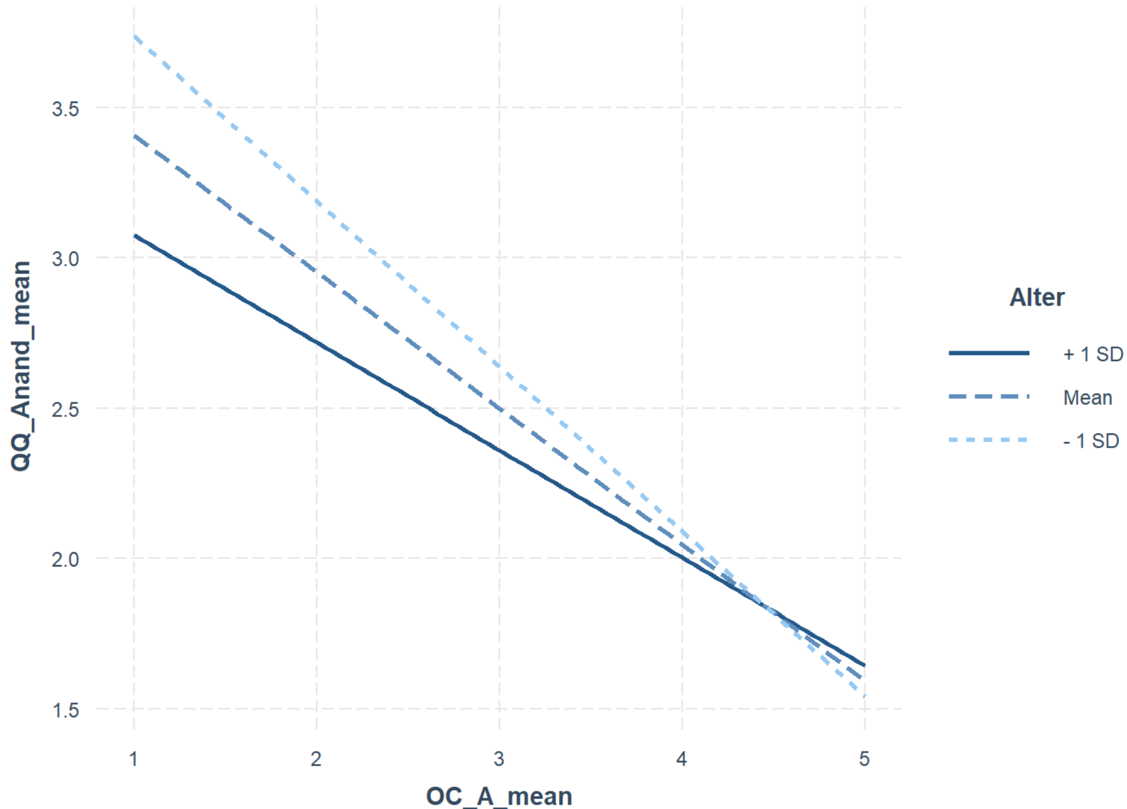


QQ_Anand

```
lm13 <- lm(QQ_Anand_mean ~ OC_A_mean*Alter, data=d_work_s)
summary(lm13)
```

```
##
## Call:
## lm(formula = QQ_Anand_mean ~ OC_A_mean * Alter, data = d_work_s)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.15814 -0.35842  0.00708  0.27445  1.95353
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      5.091634   0.446264  11.409  < 2e-16 ***
## OC_A_mean       -0.729646   0.122054  -5.978 1.15e-08 ***
## Alter           -0.031496   0.011988  -2.627  0.00933 **
## OC_A_mean:Alter  0.007063   0.003233   2.185  0.03017 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5442 on 184 degrees of freedom
##   (2 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.4517, Adjusted R-squared:  0.4428
## F-statistic: 50.53 on 3 and 184 DF,  p-value: < 2.2e-16
```

```
interact_plot(model=lm13, pred=OC_A_mean, modx=Alter)
```

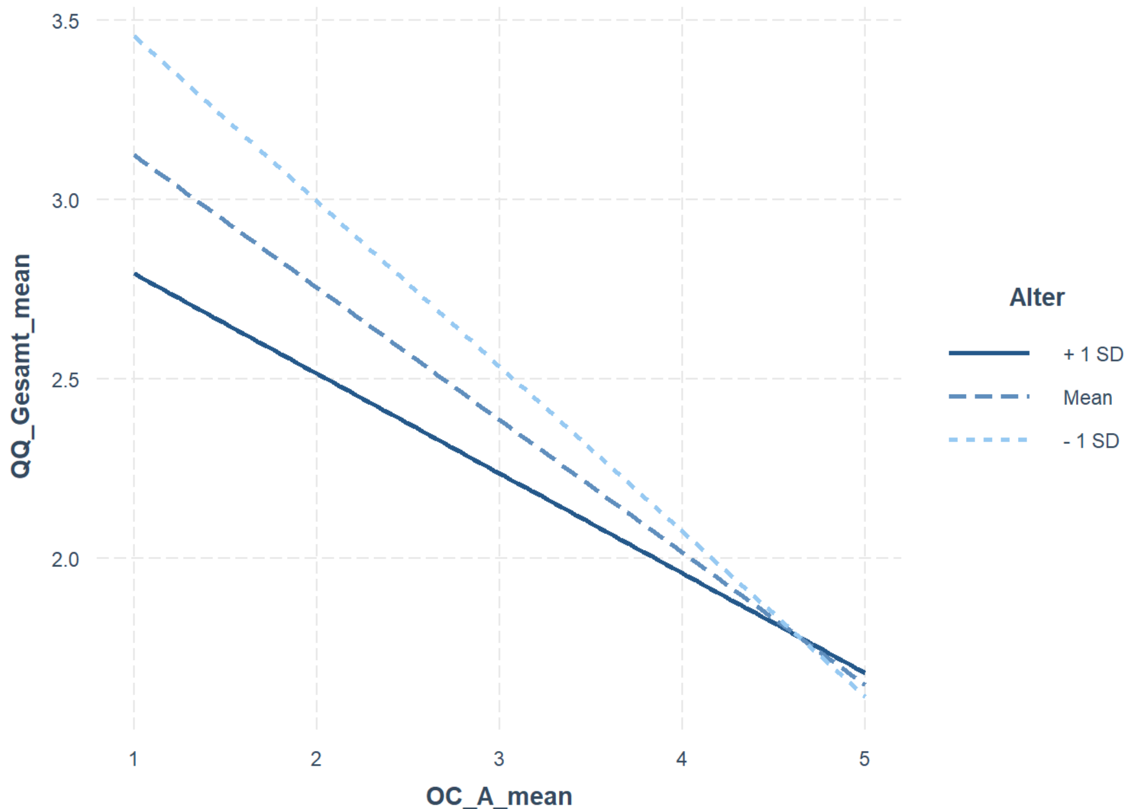


QQ_Gesamt

```
lm14 <- lm(QQ_Gesamt_mean ~ OC_A_mean*Alter, data=d_work_s)
summary(lm14)
```

```
##
## Call:
## lm(formula = QQ_Gesamt_mean ~ OC_A_mean * Alter, data = d_work_s)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.29249 -0.32551 -0.03449  0.20391  2.10053
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      4.716389   0.419832  11.234  < 2e-16 ***
## OC_A_mean       -0.632934   0.114825  -5.512 1.18e-07 ***
## Alter           -0.031240   0.011278  -2.770  0.00618 **
## OC_A_mean:Alter  0.006735   0.003041   2.215  0.02801 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.512 on 184 degrees of freedom
##   (2 Beobachtungen als fehlend gelöscht)
## Multiple R-squared:  0.3986, Adjusted R-squared:  0.3888
## F-statistic: 40.66 on 3 and 184 DF,  p-value: < 2.2e-16
```

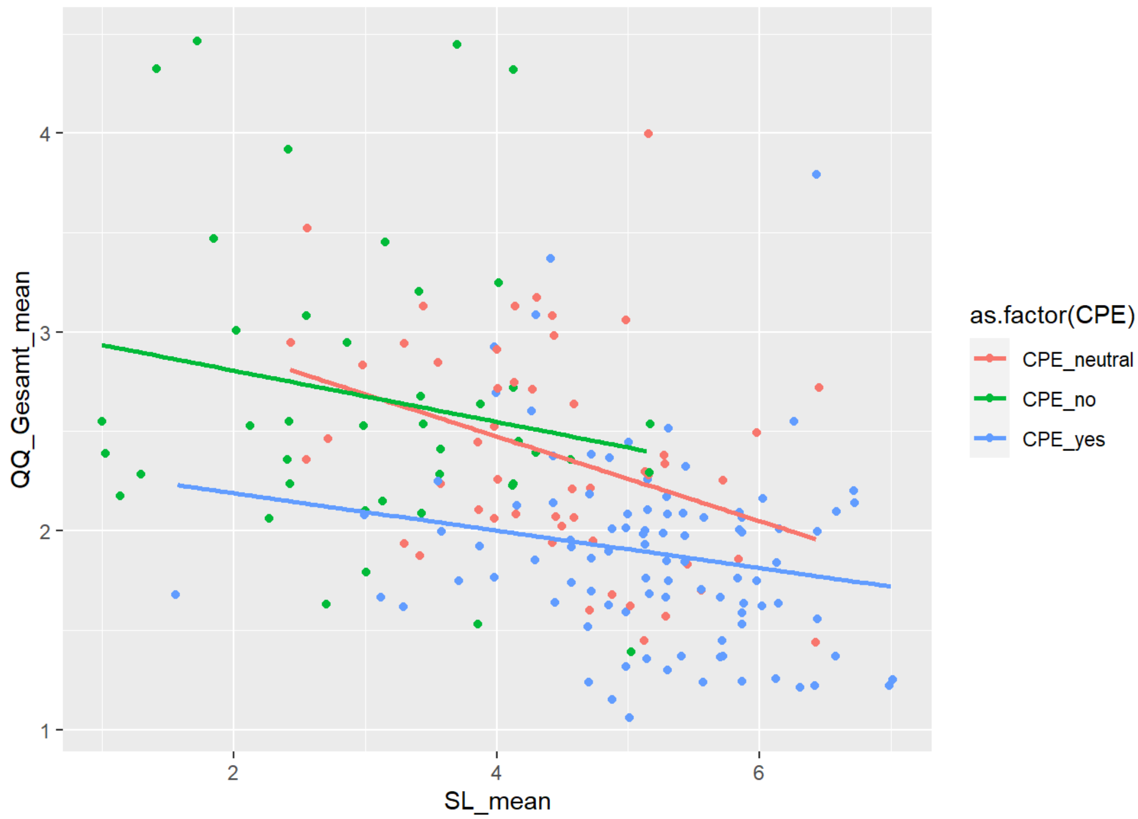
```
interact_plot(model=lm14, pred=OC_A_mean, modx=Alter)
```



Moderation: CPE moderiert den Zusammenhang von SL und QQ

```
ggplot(d_work_plot_lm)+
  geom_jitter(mapping = aes(x=SL_mean, y=QQ_Gesamt_mean, color=as.factor(CPE)))+
  geom_smooth(mapping = aes(x=SL_mean, y=QQ_Gesamt_mean, color=as.factor(CPE)), method = "lm", se=FALSE)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

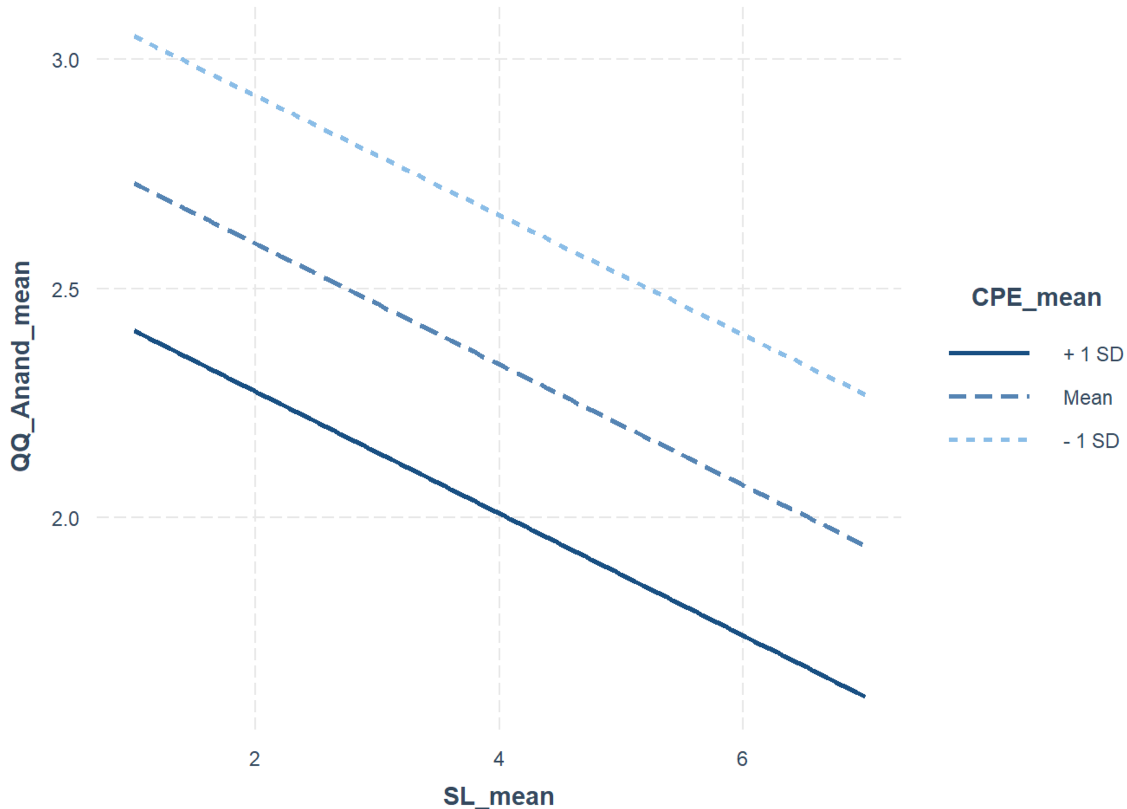


QQ_Anand

```
lm15 <- lm(QQ_Anand_mean ~ SL_mean*CPE_mean, data=d_work_s)
summary(lm15)
```

```
##
## Call:
## lm(formula = QQ_Anand_mean ~ SL_mean * CPE_mean, data = d_work_s)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.62130 -0.38599 -0.01402  0.28996  1.67173
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       4.224329   0.463541   9.113   <2e-16 ***
## SL_mean          -0.125660   0.120666  -1.041   0.2990
## CPE_mean         -0.280460   0.107839  -2.601   0.0101 *
## SL_mean:CPE_mean -0.001236   0.024008  -0.051   0.9590
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5695 on 186 degrees of freedom
## Multiple R-squared:  0.3937, Adjusted R-squared:  0.3839
## F-statistic: 40.26 on 3 and 186 DF,  p-value: < 2.2e-16
```

```
interact_plot(model=lm15, pred=SL_mean, modx=CPE_mean)
```

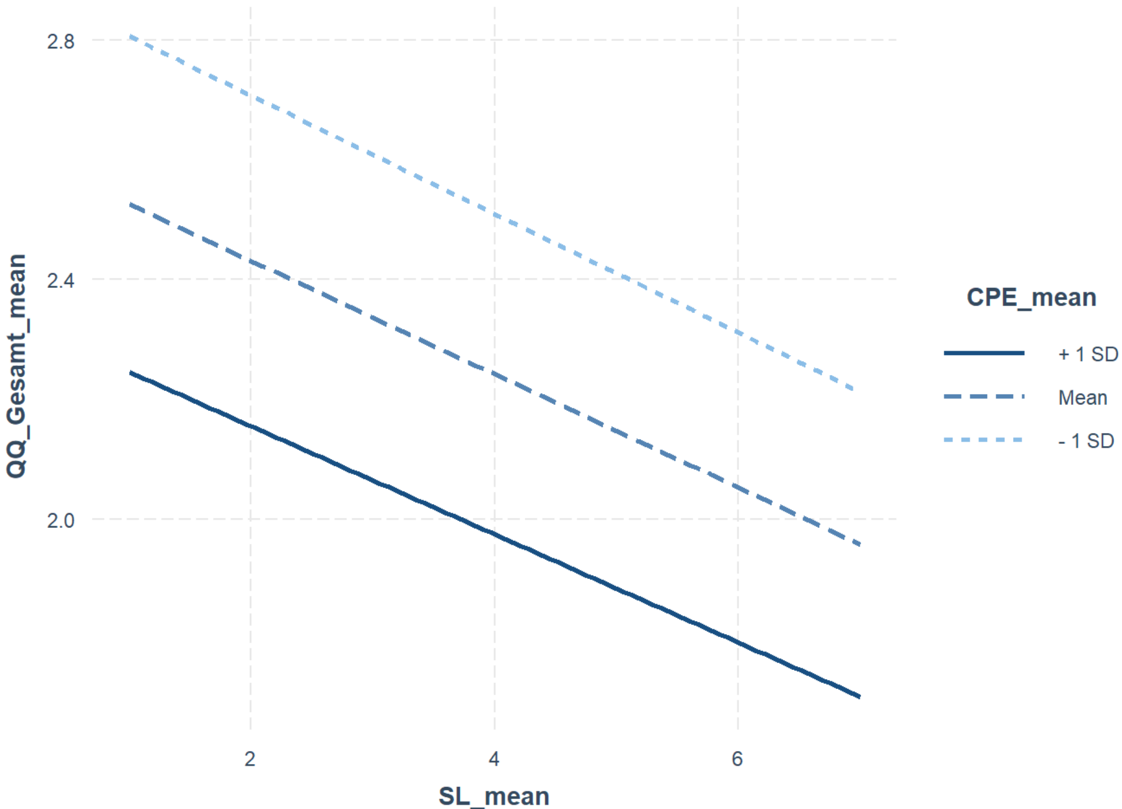


QQ_Gesamt

```
lm16 <- lm(QQ_Gesamt_mean ~ SL_mean*CPE_mean, data=d_work_s)
summary(lm16)
```

```
##
## Call:
## lm(formula = QQ_Gesamt_mean ~ SL_mean * CPE_mean, data = d_work_s)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.17852 -0.37456 -0.06911  0.28846  2.06996
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       3.83396    0.44548   8.606 3.13e-15 ***
## SL_mean          -0.11309    0.11596  -0.975   0.3307
## CPE_mean         -0.24972    0.10364  -2.410   0.0169 *
## SL_mean:CPE_mean  0.00381    0.02307   0.165   0.8690
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5473 on 186 degrees of freedom
## Multiple R-squared:  0.3058, Adjusted R-squared:  0.2946
## F-statistic: 27.31 on 3 and 186 DF,  p-value: 1.112e-14
```

```
interact_plot(model=lm16, pred=SL_mean, modx=CPE_mean)
```


2.8
2.4
2.0
QQ_Gesamt_mean
2
4
6
SL_mean
CPE_mean
+ 1 SD
Mean
- 1 SD