R-Syntax zu Kapitel 13:
Klassische Testtheorie (KTT)

Jana Gäde & Karin Schermelleh-Engel

01.03.2021

Inhalt

Vorbemerkungen

Der hier verwendete empirische Datensatz (BFIK_Labels.dat) von *N*= 250 Personen erfasst die Skala *Neurotizismus* der Kurzversion des Big Five Inventory (BFI-K; Rammstedt & John, 2005) anhand von 4 Items. Die Items lauten:
Ich . . .
. .. . werde leicht deprimiert, niedergeschlagen.
. .. . bin entspannt, lasse mich durch Stress nicht aus der Ruhe bringen. (-)
. .. . mache mir viele Sorgen.
. . . werde leicht nervös und unsicher.
(-) Invers formuliertes Item

Nachfolgend werden drei auf der Klassischen Testtheorie beruhenden und ineinander geschachtelten Modelle der Messäquivalenz (τ-Kongenerität, essentielle τ-Äquivalenz und essentielle τ-Parallelität) anhand des empirischen Beispiels unter Verwendung des R-Packages lavaan (Rosseel, 2012) analysiert. Da die Daten nicht normalverteilt sind, wird die robuste Maximum-Likelihood-Methode (MLR) für die Parameter-schätzung verwendet.
Je nach Passung der Modelle zu den Daten kann entschieden werden, welches Reliabilitätsmaß für die vorliegenden Daten adäquat ist (vgl. Gäde, Schermelleh-Engel & Werner, 2020; Schermelleh-Engel & Gäde, 2020).

Literatur

Gäde, J.C., Schermelleh-Engel, K. & Werner, C.S. (2020). Klassische Methoden der Reliabilitätsschätzung (Kapitel 14). In H. Moosbrugger & A. Kelava (Hrsg.), *Testtheorie und Fragebogenkonstruktion* (3., vollständig neu bearbeitete, erweiterte und aktualisierte Auflage). Heidelberg: Springer.

Rammstedt, B. & John, O. P. (2005). Kurzversion des Big Five Inventory (BFI-K): Entwicklung und Validierung eines ökonomischen Inventars zur Erfassung der fünf Faktoren der Persönlichkeit. *Diagnostica, 51*, 195-206.

R Development Core Team (2021). *R: A Language and Environment for Statistical Computing* (version 4.0.4). Vienna, Austria: R Foundation for Statistical Computing.

Rosseel, Y. (2012). lavaan: An R Package for Structural Equation Modeling. *Journal of Statistical Software, 48*, 1–36.

Schermelleh-Engel, K. & Gäde, J.C. (2020). Modellbasierte Methoden der Reliabilitätsschätzung (Kapitel 15). In H. Moosbrugger & A. Kelava (Hrsg.), *Testtheorie und Fragebogenkonstruktion* (3., vollständig neu bearbeitete, erweiterte und aktualisierte Auflage). Heidelberg: Springer.

Modell τ-kongenerischer Variablen

R-Input zur Schätzung von Omega

```
# Kap. 13: Klassische Testtheorie
# BFI-K: Kongenerisches Messmodell, Omega mit Konfidenzintervall

# Vor der ersten Anwendung: R-Package lavaan zur Analyse von latenten
# Variablenmodellen installieren
# dazu einen nahegelegenen CRAN Mirror angeben
# Eine Liste der CRAN Mirrors: https://cran.r-project.org/mirrors.html
install.packages("lavaan", repos='https://cran.uni-muenster.de/')

# Vor jeder Anwendung muss das installierte Package lavaan zur Nutzung
# geladen werden
library(lavaan)

# Arbeitsverzeichnis auswählen ("Dateipfad/zum/Arbeitsverzeichnis")
setwd("C:/Daten/BFIK")

# Daten einlesen
# Trennzeichen: Tabstopps (sep="")
# Variablennamen stehen in erster Zeile (header=TRUE)
BFIK_Labels.dat <- read.table("BFIK_Labels.dat", sep="", header=TRUE)

# Daten ansehen
View(BFIK_Labels.dat)

# Definition des einfaktoriellen Faktormodells
# Eindimensionales tau-kongenerisches Modell 'Neurotizismus': neur
# Die Labels p1 bis p4 bezeichnen die unterschiedlichen Faktorladungen
# der Antwortvariablen Y4, Y9, Y14 und Y19
neur <- 'NEUR=~ p1*Y4 + p2*Y9 + p3*Y14 + p4*Y19

# Den Fehlervarianzen werden die Labels e1 bis e4 zugewiesen
Y4~~e1*Y4
Y9~~e2*Y9
Y14~~e3*Y14
Y19~~e4*Y19

# Zunächst Schätzung von McDonalds Omega
# Erst danach das asymmetrische Konfidenzintervall bestimmen,
# weil dort die geschätzten Werte per Hand eingegeben werden müssen
# Formeln aus Kap. 15, Tabelle 15.2
# Achtung: Nur bei gutem Modellfit darf die Reliabilität berechnet werden!

# True-Score-Varianz
true := (p1 + p2 + p3 + p4)^2 * 1.0

# Fehlervarianz
error := e1 + e2 + e3 + e4

# Berechnung von OMEGA der Skala NEUR (True-Score-Varianz / Totale Varianz)
omega := true / (true + error)
'

# Analyse des Modells neur
# MLR-Schätzung (estimator= "MLR")
# alle latenten Varianzen auf eins fixiert (std.lv=TRUE)
fit.neur <- sem(neur, data= BFIK_Labels.dat, estimator= "MLR", std.lv=TRUE)

# Ergebnisse anzeigen
summary(fit.neur)

# standardisierte Lösung
standardizedSolution(fit.neur)
```

R-Output zur Schätzung von Omega

```
lavaan 0.6-7 ended normally after 14 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of free parameters                          8

  Number of observations                           250

Model Test User Model:
                                              Standard      Robust
  Test Statistic                                 2.095       1.882
  Degrees of freedom                                 2           2
  P-value (Chi-square)                           0.351       0.390
  Scaling correction factor                                  1.113
       Yuan-Bentler correction (Mplus variant)
# Der Modellfit ist gut, die Parameter und die Reliabilität dürfen
# interpretiert werden

Parameter Estimates:

  Standard errors                             Sandwich
  Information bread                           Observed
  Observed information based on                Hessian

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  NEUR =~
    Y4        (p1)    1.012    0.061   16.681    0.000
    Y9        (p2)    0.615    0.072    8.493    0.000
    Y14       (p3)    0.940    0.062   15.188    0.000
    Y19       (p4)    0.780    0.069   11.317    0.000
# Unstandardisierte frei geschätzte Faktorladungen

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .Y4        (e1)    0.550    0.092    6.000    0.000
   .Y9        (e2)    0.894    0.088   10.164    0.000
   .Y14       (e3)    0.511    0.067    7.581    0.000
   .Y19       (e4)    0.764    0.082    9.270    0.000
    NEUR              1.000
# Unstandardisierte frei geschätzte Fehlervarianzen

Defined Parameters:
                   Estimate  Std.Err  z-value  P(>|z|)
    true             11.201    1.054   10.628    0.000
    error             2.719    0.147   18.548    0.000
    omega             0.805    0.020   40.422    0.000
# Die Reliabilität geschätzt anhand von McDonalds Omega beträgt 0.805

# Standardisierte Lösung
     lhs op               rhs est.std    se      z pvalue
1    NEUR =~               Y4   0.807 0.037 21.872      0
2    NEUR =~               Y9   0.545 0.058  9.476      0
3    NEUR =~              Y14   0.796 0.034 23.599      0
4    NEUR =~              Y19   0.666 0.048 13.841      0
# Standardisierte Faktorladungen
5      Y4 ~~               Y4   0.349 0.059  5.872      0
6      Y9 ~~               Y9   0.703 0.063 11.191      0
7     Y14 ~~              Y14   0.366 0.054  6.824      0
8     Y19 ~~              Y19   0.557 0.064  8.693      0
# Standardisierte Fehlervarianzen
9    NEUR ~~             NEUR   1.000 0.000     NA     NA
# Standardisierte Varianz der latenten Variablen
10     Y4 ~1                    2.330 0.094 24.751      0
11     Y9 ~1                    2.804 0.121 23.145      0
12    Y14 ~1                    2.985 0.145 20.595      0
13    Y19 ~1                    2.572 0.098 26.262      0
# Interzepte der Itemvariablen
```

```
                          -- Fortsetzung - OUTPUT --

14   NEUR ~1                       0.000 0.000      NA      NA
# Standardisierter Mittelwert der latenten Variablen

# Die neu definierten Parameter werden ebenfalls standardisiert
# aber in dieser Form nicht weiter interpretiert

10  true := (p1+p2+p3+p4)^2*1.0   7.917 0.516 15.351      0     6.906    8.928
11 error :=          e1+e2+e3+e4   1.975 0.120 16.397      0     1.739    2.211
12 omega :=    true/(true+error)   0.800 0.020 39.836      0     0.761    0.840

# Die standardisierten Parameterschätzungen wurden nur zum Vergleich mit
# den Ergebnissen in Kapitel 13 ausgegeben
```

R-Input und -Output zur Schätzung des Konfidenzintervalls für Omega

```
# Berechnung der Grenzen des Konfidenzintervalls von Omega
# Formeln aus Kap. 15, Exkurs 15.3
# s.a. Raykov & Marcoulides, 2011, S. 166; Eid & Schmidt, 2014, S. 285
# Die Reliabilität und ihr Standardfehler müssen vorher geschätzt worden sein
# Achtung: Nur bei gutem Modellfit darf das Konfidenzintervall um den
# Reliabilitätskoeffizienten berechnet werden!
# Die Werte für REL (Estimate von omega) und SE (Standardfehler Std.Err. von
# omega) werden aus der vorherigen Analyse übernommen

REL <- .805
SE <- .020

# Logit-Transformation
L <- log(REL/(1-REL))

# SE des Logits berechnen
SEL <- SE/(REL*(1-REL))

# CI-Low des Logits berechnen
CI_L_LO <- L-1.96*SEL

# CI-Up des Logits berechnen
CI_L_UP <- L+1.96*SEL

# Untere Grenze des KI fuer Rel
CI_R_UP <- 1/(1+exp(-CI_L_UP))

# Obere Grenze des KI fuer Rel
CI_R_LO <- 1/(1+exp(-CI_L_LO))

# Konfidenzintervall anzeigen
CI_R <- c(CI_R_LO, CI_R_UP)
CI_R
[1] 0.7628053 0.8412508

# Der Reliabilitätskoeffizient Omega von 0.805 darf interpretiert werden wegen
# des guten Modellfits (χ²=1.882, df=2, p=0.390).
# Das 95%-Konfidenzintervall hat die Grenzen [0.763,0.841].
```

Modell essentiell τ-äquivalenter Variablen

R-Input zur Schätzung von Alpha

```
# Kap. 13: Klassische Testtheorie
# BFI-K: Ess. Tau-Äquivalentes Messmodell, Alpha mit Konfidenzintervall

# Vor der ersten Anwendung: R-Package lavaan zur Analyse von latenten
# Variablenmodellen installieren
# dazu einen nahegelegenen CRAN Mirror angeben
# Eine Liste der CRAN Mirrors: https://cran.r-project.org/mirrors.html
install.packages("lavaan", repos='https://cran.uni-muenster.de/')

# Vor jeder Anwendung muss das installierte Package lavaan zur Nutzung
# geladen werden
library(lavaan)

# Arbeitsverzeichnis auswählen ("Dateipfad/zum/Arbeitsverzeichnis")
setwd("C:/Daten/BFIK")

# Daten einlesen
# Trennzeichen: Tabstopps (sep="")
# Variablennamen stehen in erster Zeile (header=TRUE)
BFIK_Labels.dat <- read.table("BFIK_Labels.dat", sep="", header=TRUE)

# Definition des einfaktoriellen Faktormodells
# Eindimensionales ess. tau-äquivalentes Modell 'Neurotizismus': neur
# Das gemeinsame Label p1 bezeichnet die gleichgesetzten Faktorladungen
neur <- 'NEUR=~ p1*Y4 + p1*Y9 + p1*Y14 + p1*Y19

# Den Fehlervarianzen werden die Labels e1 bis e4 zugewiesen
Y4~~e1*Y4
Y9~~e2*Y9
Y14~~e3*Y14
Y19~~e4*Y19

# Zunächst Schätzung von Cronbachs Alpha
# Erst danach das asymmetrische Konfidenzintervall bestimmen,
# weil dort die geschätzten Werte per Hand eingegeben werden müssen
# Formeln aus Kap. 15, Tabelle 15.2
# Achtung: Nur bei gutem Modellfit darf die Reliabilität berechnet werden!

# True-Score-Varianz
true := (p1*4)^2 * 1.0

# Fehlervarianz
error := e1 + e2 + e3 + e4

# Berechnung von ALPHA der Skala NEUR (True-Score-Varianz / Totale Varianz)
alpha := true / (true + error)
'

# Analyse des Modells neur
# MLR-Schätzung (estimator= "MLR")
# alle latenten Varianzen auf eins fixiert (std.lv=TRUE)
fit.neur <- sem(neur, data= BFIK_Labels.dat, estimator= "MLR", std.lv=TRUE)

# Ergebnisse anzeigen
summary(fit.neur)

# standardisierte Lösung
standardizedSolution(fit.neur)
```

R-Output zur Schätzung von Alpha

```
lavaan 0.6-7 ended normally after 9 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of free parameters                          8
  Number of equality constraints                     3

  Number of observations                           250

Model Test User Model:
                                              Standard      Robust
  Test Statistic                                23.730      23.726
  Degrees of freedom                                 5           5
  P-value (Chi-square)                           0.000       0.000
  Scaling correction factor                                  1.000
       Yuan-Bentler correction (Mplus variant)
# Der Modellfit ist nicht hinreichend gut, die Parameter und die Reliabilität
# dürften deshalb eigentlich nicht interpretiert werden.

Parameter Estimates:

  Standard errors                             Sandwich
  Information bread                           Observed
  Observed information based on                Hessian

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  NEUR =~
    Y4        (p1)    0.848    0.041   20.633    0.000
    Y9        (p1)    0.848    0.041   20.633    0.000
    Y14       (p1)    0.848    0.041   20.633    0.000
    Y19       (p1)    0.848    0.041   20.633    0.000
# Annahme der essent. Tau-Äquivalenz: Alle Faktorladungen wurden gleichgesetzt

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .Y4        (e1)    0.680    0.073    9.332    0.000
   .Y9        (e2)    0.841    0.093    9.023    0.000
   .Y14       (e3)    0.582    0.057   10.168    0.000
   .Y19       (e4)    0.707    0.070   10.050    0.000
    NEUR              1.000
# Alle Fehlervarianzen wurden frei geschätzt

Defined Parameters:
                   Estimate  Std.Err  z-value  P(>|z|)
    true             11.500    1.115   10.316    0.000
    error             2.810    0.156   18.016    0.000
    alpha             0.804    0.020   39.387    0.000
# Die Reliabilität geschätzt anhand von Cronbachs Alpha beträgt 0.804

# Standardisierte Lösung
     lhs op               rhs est.std    se      z pvalue ci.lower ci.upper
1    NEUR =~               Y4   0.717 0.030 23.816      0    0.658    0.776
2    NEUR =~               Y9   0.679 0.027 25.331      0    0.626    0.731
3    NEUR =~              Y14   0.743 0.026 28.464      0    0.692    0.795
4    NEUR =~              Y19   0.710 0.027 26.475      0    0.657    0.762
5      Y4 ~~               Y4   0.486 0.043 11.271      0    0.402    0.571
6      Y9 ~~               Y9   0.539 0.036 14.823      0    0.468    0.611
7     Y14 ~~              Y14   0.447 0.039 11.515      0    0.371    0.523
8     Y19 ~~              Y19   0.496 0.038 13.031      0    0.421    0.571
9    NEUR ~~             NEUR   1.000 0.000     NA     NA    1.000    1.000
10   true :=      (p1*4)^2*1.0  8.220 0.690 11.908      0    6.867    9.573
11  error :=       e1+e2+e3+e4  1.969 0.130 15.164      0    1.714    2.223
12  alpha := true/(true+error)  0.807 0.023 35.577      0    0.762    0.851

# Die standardisierten Parameterschätzungen wurden nur zum Vergleich mit
# den Ergebnissen in Kapitel 13 ausgegeben
```

R-Input und -Output zur Schätzung des Konfidenzintervalls für Alpha

```
# Berechnung der Grenzen des Konfidenzintervalls von Alpha
# Formeln aus Kap. 15, Exkurs 15.3
# s.a. Raykov & Marcoulides, 2011, S. 166; Eid & Schmidt, 2014, S. 285
# Die Reliabilität und ihr Standardfehler müssen vorher geschätzt worden sein
# Achtung: Nur bei gutem Modellfit darf das Konfidenzintervall um den
# Reliabilitätskoeffizienten berechnet werden!
# Die Werte für REL (Estimate von alpha) und SE (Standardfehler Std.Err. von
# alpha) werden aus der vorherigen Analyse übernommen

REL <- .804
SE <- .020

# Logit-Transformation
L <- log(REL/(1-REL))

# SE des Logits berechnen
SEL <- SE/(REL*(1-REL))

# CI-Low des Logits berechnen
CI_L_LO <- L-1.96*SEL

# CI-Up des Logits berechnen
CI_L_UP <- L+1.96*SEL

# Untere Grenze des KI fuer Rel
CI_R_UP <- 1/(1+exp(-CI_L_UP))

# Obere Grenze des KI fuer Rel
CI_R_LO <- 1/(1+exp(-CI_L_LO))

# Konfidenzintervall anzeigen
CI_R <- c(CI_R_LO, CI_R_UP)
CI_R
[1] 0.7618281 0.8402703

# Der Reliabilitätskoeffizient Alpha von 0.804 hätte ein 95%-Konfidenzintervall
# mit den Grenzen [0.762,0.840] gehabt, wenn der Modellfit zufriedenstellend
# gut gewesen wäre
```

Modell essentiell τ-paralleler Variablen

R-Input zur Schätzung der Spearman-Brown-Reliabilität

```
# Kap. 13: Klassische Testtheorie
# BFI-K: Ess. Tau-Paralleles Messmodell
# Spearman-Brown-Reliabilitaet mit Konfidenzintervall

# Vor der ersten Anwendung: R-Package lavaan zur Analyse von latenten
# Variablenmodellen installieren
# dazu einen nahegelegenen CRAN Mirror angeben
# Eine Liste der CRAN Mirrors: https://cran.r-project.org/mirrors.html
install.packages("lavaan", repos='https://cran.uni-muenster.de/')

# Vor jeder Anwendung muss das installierte Package lavaan zur Nutzung
# geladen werden
library(lavaan)

# Arbeitsverzeichnis auswählen ("Dateipfad/zum/Arbeitsverzeichnis")
setwd("C:/Daten/BFIK")

# Daten einlesen
# Trennzeichen: Tabstopps (sep="")
# Variablennamen stehen in erster Zeile (header=TRUE)
BFIK_Labels.dat <- read.table("BFIK_Labels.dat", sep="", header=TRUE)

# Definition des einfaktoriellen Faktormodells
# Eindimensionales ess. tau-paralleles Modell 'Neurotizismus': neur
# Das gemeinsame Label p1 bezeichnet die Faktorladungen
neur <- 'NEUR=~ p1*Y4 + p1*Y9 + p1*Y14 + p1*Y19

# Den Fehlervarianzen wird das gemeinsame Label e1 zugewiesen
Y4~~e1*Y4
Y9~~e1*Y9
Y14~~e1*Y14
Y19~~e1*Y19

# Zunächst Schätzung der Spearman-Brown-Reliabilitaet
# Erst danach das asymmetrische Konfidenzintervall bestimmen,
# weil dort die geschätzten Werte per Hand eingegeben werden müssen
# Formeln aus Kap. 15, Tabelle 15.2
# Achtung: Nur bei gutem Modellfit darf die Reliabilität berechnet werden!

# True-Score-Varianz
true := (p1*4)^2 * 1.0

# Fehlervarianz
error := (e1*4)

# Berechnung der Spearman-Brown-Reliabilitaet (SB) der Skala NEUR
# (True-Score-Varianz / Totale Varianz)
SB := true / (true + error)
'

# Analyse des Modells neur
# MLR-Schätzung (estimator= "MLR")
# alle latenten Varianzen auf eins fixiert (std.lv=TRUE)
fit.neur <- sem(neur, data= BFIK_Labels.dat, estimator= "MLR", std.lv=TRUE)

# Ergebnisse anzeigen
summary(fit.neur)

# standardisierte Lösung
standardizedSolution(fit.neur)
```

R-Output zur Schätzung der Spearman-Brown-Reliabilität

```
lavaan 0.6-7 ended normally after 5 iterations

  Estimator                                         ML
  Optimization method                           NLMINB
  Number of free parameters                          8
  Number of equality constraints                     6

  Number of observations                           250

Model Test User Model:
                                              Standard      Robust
  Test Statistic                                28.325      31.347
  Degrees of freedom                                 8           8
  P-value (Chi-square)                           0.000       0.000
  Scaling correction factor                                  0.904
       Yuan-Bentler correction (Mplus variant)
# Der Modellfit ist nicht hinreichend gut, die Parameter und die Reliabilität
# dürften deshalb eigentlich nicht interpretiert werden

Parameter Estimates:

  Standard errors                             Sandwich
  Information bread                           Observed
  Observed information based on                Hessian

Latent Variables:
                   Estimate  Std.Err  z-value  P(>|z|)
  NEUR =~
    Y4        (p1)    0.834    0.040   21.014    0.000
    Y9        (p1)    0.834    0.040   21.014    0.000
    Y14       (p1)    0.834    0.040   21.014    0.000
    Y19       (p1)    0.834    0.040   21.014    0.000
# Annahme der essent. Tau-Parallelität:
# Alle Faktorladungen wurden gleichgesetzt
    NEUR              0.000

Variances:
                   Estimate  Std.Err  z-value  P(>|z|)
   .Y4        (e1)    0.708    0.040   17.703    0.000
   .Y9        (e1)    0.708    0.040   17.703    0.000
   .Y14       (e1)    0.708    0.040   17.703    0.000
   .Y19       (e1)    0.708    0.040   17.703    0.000
    NEUR              1.000
# Annahme der essentiellen Tau-Parallelität:
# Alle Fehlervarianzen wurden gleichgesetzt

Defined Parameters:
                   Estimate  Std.Err  z-value  P(>|z|)
    true             11.123    1.059   10.507    0.000
    error             2.832    0.160   17.703    0.000
    SB                0.797    0.021   37.859    0.000
# Die Reliabilität geschätzt anhand von SB beträgt 0.797

# Standardisierte Lösung
     lhs op                  rhs est.std    se      z pvalue ci.lower ci.upper
1   NEUR =~                   Y4   0.704 0.023 30.454      0    0.659    0.749
2   NEUR =~                   Y9   0.704 0.023 30.454      0    0.659    0.749
3   NEUR =~                  Y14   0.704 0.023 30.454      0    0.659    0.749
4   NEUR =~                  Y19   0.704 0.023 30.454      0    0.659    0.749
5     Y4 ~~                   Y4   0.505 0.033 15.508      0    0.441    0.568
6     Y9 ~~                   Y9   0.505 0.033 15.508      0    0.441    0.568
7    Y14 ~~                  Y14   0.505 0.033 15.508      0    0.441    0.568
8    Y19 ~~                  Y19   0.505 0.033 15.508      0    0.441    0.568
9   NEUR ~~                 NEUR   1.000 0.000     NA     NA    1.000    1.000
10  true :=        (p1*4)^2*1.0   7.927 0.521 15.227      0    6.907    8.947
11 error :=              (e1*4)   2.018 0.130 15.508      0    1.763    2.273
12    SB := true/(true+error)    0.797 0.021 37.859      0    0.756    0.838
# Die standardisierten Parameterschätzungen wurden nur zum Vergleich mit
# den Ergebnissen in Kapitel 13 ausgegeben
```

R-Input und -Output zur Schätzung des Konfidenzintervalls der Spearman-Brown-Reliabilität

```
# Berechnung der Grenzen des Konfidenzintervalls von SB
# s.a. Raykov & Marcoulides, 2011, S. 166; Eid & Schmidt, 2014, S. 285
# Die Reliabilität und ihr Standardfehler müssen vorher geschätzt worden sein
# Achtung: Nur bei gutem Modellfit darf das Konfidenzintervall um den
# Reliabilitätskoeffizienten berechnet werden!
# Die Werte für REL (Estimate von SB) und SE (Standardfehler von SB)
# können aus der vorherigen Analyse übernommen werden

REL <- .797
SE <- .021

# Logit-Transformation
L <- log(REL/(1-REL))

# SE des Logits berechnen
SEL <- SE/(REL*(1-REL))

# CI-Low des Logits berechnen
CI_L_LO <- L-1.96*SEL

# CI-Up des Logits berechnen
CI_L_UP <- L+1.96*SEL

# Untere Grenze des KI fuer Rel
CI_R_UP <- 1/(1+exp(-CI_L_UP))

# Obere Grenze des KI fuer Rel
CI_R_LO <- 1/(1+exp(-CI_L_LO))

# Konfidenzintervall anzeigen
CI_R <- c(CI_R_LO, CI_R_UP)
CI_R
[1] 0.7527338 0.8350778

# Der Spearman-Brown-Reliabilitätskoeffizient von 0.797 hätte ein 95%-
# Konfidenzintervall mit den Grenzen [0.753, 0.835] gehabt, wenn der Modellfit
# hinreichend gut gewesen wäre
```