

R-Syntax zu Kapitel 15: Modellbasierte Methoden der Reliabilitätsschätzung

Karin Schermelleh-Engel & Jana C. Gåde

01.03.2021

Inhalt

1	Cronbachs Alpha.....	2
1.1	Modellbasierte Schätzung von Cronbachs Alpha – R-Input	2
1.2	Modellbasierte Schätzung von Cronbachs Alpha – R-Output	3
1.3	Konfidenzintervall zu Cronbachs Alpha	4
2	McDonalds Omega	5
2.1	Modellbasierte Schätzung von McDonalds Omega – R-Input.....	5
2.2	Modellbasierte Schätzung von McDonalds Omega – R-Output.....	6
2.3	Konfidenzintervall zu McDonalds Omega	7
3	Bollens Omega.....	8
3.1	Modellbasierte Schätzung von Bollens Omega – R-Input	8
3.2	Modellbasierte Schätzung von Bollens Omega – R-Output	9
3.3	Konfidenzintervall zu Bollens Omega.....	10
4	Omega-Koeffizienten im Bifaktormodell.....	11
4.1	Modellbasierte Schätzung der Omega-Koeffizienten im Bifaktormodell – R-Input.....	11
4.2	Modellbasierte Schätzung der Omega-Koeffizienten im Bifaktormodell – R-Output.....	13
4.3	Konfidenzintervall zu den Omega-Koeffizienten im Bifaktormodell.....	15

1 Cronbachs Alpha

1.1 Modellbasierte Schätzung von Cronbachs Alpha – R-Input

```
# Kap. 15: Eindimensionale Modelle
# Cronbachs Alpha mit Konfidenzintervall

# Vor erster 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
setwd("Dateipfad/zum/Arbeitsverzeichnis")

# Daten einlesen
# Trennzeichen: Tabstopps (sep="")
# Variablennamen stehen in erster Zeile (header=TRUE)
Kapitel_15_Daten_R <- read.table("Kapitel_15_Daten_R.dat", sep="", header=TRUE)

# Definition des einfaktoriellen Modells
# Eindimensionales tau-äquivalentes Modell 'Concern over Mistakes': cm
# Alle Faktorladungen erhalten dasselbe Label p und werden damit
# gleichgesetzt
cm <- 'CM=~ p*CM09 + p*CM21 + p*CM23 + p*CM25 + p*CM34

# Den Fehlervarianzen werden die Labels e1 bis e5 zugewiesen
CM09~~e1*CM09
CM21~~e2*CM21
CM23~~e3*CM23
CM25~~e4*CM25
CM34~~e5*CM34

# Schätzung von Cronbachs Alpha
# Zunächst die Reliabilität schätzen
# 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
cm.true := (p*5)^2 * 1.0

# Fehlervarianz
cm.error := e1 + e2 + e3 + e4 + e5

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

# Analyse des Modells cm
# MLR-Schätzung (estimator= "MLR")
# alle latenten Varianzen auf eins fixiert (std.lv=TRUE)
fit.cm <- sem(cm, data= Kapitel_15_Daten_R, estimator= "MLR", std.lv=TRUE)

# Ergebnisse anzeigen
summary(fit.cm)
```

1.2 Modellbasierte Schätzung von Cronbachs Alpha – R-Output

```
lavaan 0.6-7 ended normally after 10 iterations

Estimator ML
Optimization method NLMINB
Number of free parameters 10
Number of equality constraints 4

Number of observations 250

Model Test User Model:
Test Statistic Standard Robust
Degrees of freedom 32.435 29.095
P-value (Chi-square) 9 9
Scaling correction factor 0.000 0.001
Yuan-Bentler correction (Mplus variant) 1.115

Parameter Estimates:
Standard errors Sandwich
Information bread Observed
Observed information based on Hessian

Latent Variables:
Estimate Std.Err z-value P(>|z|)
CM =~
CM09 (p) 0.882 0.046 19.321 0.000
CM21 (p) 0.882 0.046 19.321 0.000
CM23 (p) 0.882 0.046 19.321 0.000
CM25 (p) 0.882 0.046 19.321 0.000
CM34 (p) 0.882 0.046 19.321 0.000

Variances:
Estimate Std.Err z-value P(>|z|)
.CM09 (e1) 0.791 0.080 9.893 0.000
.CM21 (e2) 0.554 0.062 8.878 0.000
.CM23 (e3) 0.445 0.060 7.366 0.000
.CM25 (e4) 0.509 0.063 8.134 0.000
.CM34 (e5) 0.487 0.056 8.669 0.000
CM 1.000

Defined Parameters:
Estimate Std.Err z-value P(>|z|)
cm.true 19.461 2.015 9.660 0.000
cm.error 2.785 0.168 16.572 0.000
cm.alpha 0.875 0.015 58.745 0.000
# Cronbachs Alpha wurde auf 0.875 geschätzt
```

1.3 Konfidenzintervall zu Cronbachs 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 cm.alpha) und
# SE (Standardfehler Std.Err. von cm.alpha)
# werden aus vorheriger Analyse übernommen
REL <- .875
SE <- .015

# 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] 0.8425215 0.9015627
# Cronbachs Alpha wurde auf 0.875 geschätzt mit einem 95%-Konfidenzintervall von [0.843; 0.902]
```

2 McDonalds Omega

2.1 Modellbasierte Schätzung von McDonalds Omega – R-Input

```
# Kap. 15: Eindimensionale Modelle
# McDonalds Omega mit Konfidenzintervall

# Vor erster 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
setwd("Dateipfad/zum/Arbeitsverzeichnis")

# Daten einlesen
# Trennzeichen: Tabstopps (sep="")
# Variablennamen stehen in erster Zeile (header=TRUE)
Kapitel_15_Daten_R <- read.table("Kapitel_15_Daten_R.dat", sep="", header=TRUE)

# Definition des einfaktoriellen Modells
# Eindimensionales tau-kongenerisches Modell 'Concern over Mistakes': cm
# Die Labels p1 bis p5 bezeichnen die unterschiedlichen Faktorladungen
cm <- 'CM=~ p1*CM09 + p2*CM21 + p3*CM23 + p4*CM25 + p5*CM34

# Den Fehlervarianzen werden die Labels e1 bis e5 zugewiesen
CM09~~e1*CM09
CM21~~e2*CM21
CM23~~e3*CM23
CM25~~e4*CM25
CM34~~e5*CM34

# Schätzung von McDonalds Omega
# Zunächst die Reliabilität schätzen
# 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
cm.true := (p1 + p2 + p3 + p4 + p5)^2 * 1.0

# Fehlervarianz
cm.error := e1 + e2 + e3 + e4 + e5

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

# Analyse des Modells cm
# MLR-Schätzung (estimator= "MLR")
# alle latenten Varianzen auf eins fixiert (std.lv=TRUE)
fit.cm <- sem(cm, data= Kapitel_15_Daten_R, estimator= "MLR", std.lv=TRUE)

# Ergebnisse anzeigen
summary(fit.cm)
```

2.2 Modellbasierte Schätzung von McDonalds Omega – R-Output

```
lavaan 0.6-7 ended normally after 13 iterations

Estimator ML
Optimization method NLMINB
Number of free parameters 10

Number of observations 250

Model Test User Model:
Test Statistic Standard Robust
Degrees of freedom 24.616 17.841
P-value (Chi-square) 5 5
Scaling correction factor 0.000 0.003
Yuan-Bentler correction (Mplus variant) 1.380

Parameter Estimates:
Standard errors Sandwich
Information bread Observed
Observed information based on Hessian

Latent Variables:
Estimate Std.Err z-value P(>|z|)
CM =~
CM09 (p1) 0.824 0.072 11.530 0.000
CM21 (p2) 0.818 0.056 14.614 0.000
CM23 (p3) 0.902 0.065 13.936 0.000
CM25 (p4) 0.990 0.053 18.661 0.000
CM34 (p5) 0.853 0.063 13.543 0.000

Variances:
Estimate Std.Err z-value P(>|z|)
.CM09 (e1) 0.801 0.086 9.311 0.000
.CM21 (e2) 0.565 0.064 8.755 0.000
.CM23 (e3) 0.448 0.067 6.702 0.000
.CM25 (e4) 0.458 0.069 6.670 0.000
.CM34 (e5) 0.492 0.059 8.294 0.000
CM 1.000

Defined Parameters:
Estimate Std.Err z-value P(>|z|)
cm.true 19.253 1.974 9.756 0.000
cm.error 2.764 0.166 16.665 0.000
cm.omega 0.874 0.015 58.965 0.000
# McDonalds Omega wurde auf 0.874 geschätzt
```

2.3 Konfidenzintervall zu McDonalds 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 Die Werte für REL (Estimate von cm.omega) und
# SE (Standardfehler Std.Err. von cm.omega)
# werden aus vorheriger Analyse übernommen
REL <- .874
SE <- .015

# 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] 0.8415527 0.9005876
# McDonalds Omega wurde auf 0.874 geschätzt mit einem 95%-Konfidenzintervall von [0.842; 0.900]
```

3 Bollens Omega

3.1 Modellbasierte Schätzung von Bollens Omega – R-Input

```

# Kap. 15: Eindimensionale Modelle
# Bollens Omega mit Konfidenzintervall

# Vor erster 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
setwd("Dateipfad/zum/Arbeitsverzeichnis")

# Daten einlesen
# Trennzeichen: Tabstopps (sep="")
# Variablennamen stehen in erster Zeile (header=TRUE)
Kapitel_15_Daten_R <- read.table("Kapitel_15_Daten_R.dat", sep="", header=T
RUE)

# Definition des einfaktoriellen Modells
# Eindimensionales tau-kongenerisches Modell 'Concern over Mistakes': cm
# Die Labels p1 bis p5 bezeichnen die unterschiedlichen Faktorladungen
cm <- 'CM~ p1*CM09 + p2*CM21 + p3*CM23 + p4*CM25 + p5*CM34

# Den Fehlervarianzen werden die Labels e1 bis e5 zugewiesen
CM09~~e1*CM09
CM21~~e2*CM21
CM23~~e3*CM23
CM25~~e4*CM25
CM34~~e5*CM34

# Eine Fehlerkovarianz wird zugelassen:
# Der Fehlerkovarianz zwischen Item CM09 und CM23 wird
# das Label e6 zugewiesen
CM09~~e6*CM23

# Schätzung von Bollens Omega
# Zunächst die Reliabilität schätzen
# 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
cm.true := (p1 + p2 + p3 + p4 + p5)^2 * 1.0

# Fehlervarianz
cm.error := (e1 + e2 + e3 + e4 + e5) + 2*e6

# Berechnung von BOLLENS OMEGA der Skala CM
# (True-Score-Varianz / Totale Varianz)
cm.B.omega := cm.true / (cm.true + cm.error)

# Analyse des Modells cm
# MLR-Schätzung (estimator="MLR")
# alle latenten Varianzen auf eins fixiert (std.lv=TRUE)
fit.cm <- sem(cm, data= Kapitel_15_Daten_R, estimator="MLR", std.lv=TRUE)

# Ergebnisse anzeigen
summary(fit.cm)

```


3.2 Modellbasierte Schätzung von Bollen's Omega – R-Output

```
lavaan 0.6-7 ended normally after 16 iterations

Estimator                      ML
Optimization method             NLMINB
Number of free parameters       11

Number of observations           250

Model Test User Model:
Test Statistic                  Standard      Robust
Degrees of freedom              6.249        4.564
P-value (Chi-square)           0.181        4
Scaling correction factor       0.335        1.369
Yuan-Bentler correction (Mplus variant)

Parameter Estimates:
Standard errors                 Sandwich
Information bread              Observed
Observed information based on  Hessian

Latent Variables:
Estimate  Std.Err  z-value  P(>|z|)
CM =~
CM09      (p1)    0.759    0.074    10.291   0.000
CM21      (p2)    0.832    0.055    15.052   0.000
CM23      (p3)    0.859    0.068    12.616   0.000
CM25      (p4)    1.008    0.052    19.347   0.000
CM34      (p5)    0.868    0.063    13.813   0.000

Covariances:
Estimate  Std.Err  z-value  P(>|z|)
.CM09 ~~
.CM23      (e6)    0.218    0.069    3.148    0.002

Variances:
Estimate  Std.Err  z-value  P(>|z|)
.CM09      (e1)    0.905    0.087    10.349   0.000
.CM21      (e2)    0.542    0.065    8.383    0.000
.CM23      (e3)    0.522    0.077    6.774    0.000
.CM25      (e4)    0.423    0.068    6.255    0.000
.CM34      (e5)    0.466    0.060    7.783    0.000
CM         1.000

Defined Parameters:
Estimate  Std.Err  z-value  P(>|z|)
cm.true   18.715   1.992    9.393    0.000
cm.error   3.294    0.273   12.080    0.000
cm.B.omega 0.850    0.020   43.323    0.000
# Bollen's Omega wurde auf 0.850 geschätzt
```

3.3 Konfidenzintervall zu Bollen's Omega

```
# Berechnung der Grenzen des Konfidenzintervalls von Bollen's Omega
# Die Werte für REL (= cm.B.omega) und SE (Standardfehler von cm.B.omega)
# können aus vorheriger Analyse übernommen werden

# Berechnung der Grenzen des Konfidenzintervalls von Bollen's 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 Die Werte für REL (Estimate von cm.B.omega)
# und SE (Standardfehler Std.Err. von cm.B.omega)
# werden aus vorheriger Analyse übernommen
REL <- .85
SE <- .02

# 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] 0.8064569 0.8851421
# Bollen's Omega wurde auf 0.850 geschätzt mit einem 95%-Konfidenzintervall
# von [0.806; 0.885]
```

4 Omega-Koeffizienten im Bifaktormodell

4.1 Modellbasierte Schätzung der Omega-Koeffizienten im Bifaktormodell – R-Input

```
# Kap. 15: Bifaktormodell
# McDonalds Omega im mehrdimensionalen Modell

# Vor erster 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
setwd("Dateipfad/zum/Arbeitsverzeichnis")

# Daten einlesen
# Trennzeichen: Tabstopps (sep="")
# Variablennamen stehen in erster Zeile (header=TRUE)
Kapitel_15_Daten_R <- read.table("Kapitel_15_Daten_R.dat", sep="", header
=TRUE)

# Definition des Bifaktormodells
# Annahme der Kongenerität:
# Alle Faktorladungen erhalten unterschiedliche Labels
# Die Labels können frei gewählt werden, z.B. p für Parameter

bifaktor <- '
# Spezifische Faktoren
CM=~ p1*CM09 + p2*CM21 + p3*CM25 + p4*CM34      # Concern over Mistakes
DA=~ p5*DA17 + p6*DA28 + p7*DA32                # Doubts about Actions
PS=~ p8*PS12 + p9*PS19 + p10*PS24               # Personal Standards

# Generalfaktor
GEN=~ p11*CM09 + p12*CM21 + p13*CM25 + p14*CM34 # CM-Items
+ p15*DA17 + p16*DA28 + p17*DA32                # DA-Items
+ p18*PS12 + p19*PS19 + p20*PS24                # PS-Items

# Den Fehlervarianzen werden die Labels e1 bis e10 zugewiesen
CM09~~e1*CM09 #CM-Fehlervarianzen
CM21~~e2*CM21
CM25~~e3*CM25
CM34~~e4*CM34
DA17~~e5*DA17 #DA-Fehlervarianzen
DA28~~e6*DA28
DA32~~e7*DA32
PS12~~e8*PS12 #PS-Fehlervarianzen
PS19~~e9*PS19
PS24~~e10*PS24

# Schätzung der Omegakoeffizienten
# Für die Subskalen CM, DA und PS:
# Omega-CM-total, Omega-CM-hierarchisch, Omega-CM-spezifisch
# Omega-DA-total, Omega-DA-hierarchisch, Omega-DA-spezifisch
# Omega-PS-total, Omega-PS-hierarchisch, Omega-PS-spezifisch
# Für den Gesamttest: Omega-total, Omega-hierarchisch, Omega-spezifisch
# Die Multiplikation der jeweiligen wahren Varianzanteile mit
# den Faktorvarianzen (1.0) wird zur Vereinfachung weggelassen

# Einzelskalen
# Teilberechnungen

# CM
cm.g := (p11 + p12 + p13 + p14)^2
```

```

# True-Score-Varianz Generalfaktor
cm.spez := (p1 + p2 + p3 + p4)^2
# True-Score-Varianz spezifischer Faktor
cm.error := e1 + e2 + e3 + e4
# Fehlervarianz

# DA
da.g := (p15 + p16 + p17)^2
# True-Score-Varianz Generalfaktor
da.spez := (p5 + p6 + p7)^2
# True-Score-Varianz spezifischer Faktor
da.error := e5 + e6 + e7
# Fehlervarianz

# PS
ps.g := (p18 + p19 + p20)^2
# True-Score-Varianz Generalfaktor
ps.spez := (p8 + p9 + p10)^2
# True-Score-Varianz spezifischer Faktor
ps.error := e8 + e9 + e10
# Fehlervarianz

# Berechnung der Omegas für CM
om.cm.t := (cm.g + cm.spez) / (cm.g + cm.spez + cm.error)
# OMEGA-CM-TOTAL
om_cm_h := cm.g / (cm.g + cm.spez + cm.error)
# OMEGA-CM-Hierarchisch
om_cm_s := cm.spez / (cm.g + cm.spez + cm.error)
# OMEGA-CM-Spezifisch

# Berechnung der Omegas für DA
om.da.t := (da.g + da.spez) / (da.g + da.spez + da.error)
# OMEGA-DA-TOTAL
om.da.h := da.g / (da.g + da.spez + da.error)
# OMEGA-DA-Hierarchisch
om_da_s := da.spez / (da.g + da.spez + da.error)
# OMEGA-DA-Spezifisch

# Berechnung der Omegas für PS
om.ps.t := (ps.g + ps.spez) / (ps.g + ps.spez + ps.error)
# OMEGA-PS-TOTAL
om.ps.h := ps.g / (ps.g + ps.spez + ps.error)
# OMEGA-PS-Hierarchisch
om_ps_s := ps.spez / (ps.g + ps.spez + ps.error)
# OMEGA-PS-Spezifisch

# Gesamtmodell
# Teilberechnungen
H := (p11+p12+p13+p14+p15+p16+p17+p18+p19+p20)^2
# True-Score-Varianz Generalfaktor
S := cm.spez + da.spez + ps.spez
# True-Score-Varianz aller spezifischen Faktoren
E := cm.error + da.error + ps.error
# Fehlervarianz gesamt

# Berechnung der Omegas für das Gesamtmodell
OM_T := (H + S) / (H + S + E) # Omega-Total
OM_H := H / (H + S + E) # Omega-Hierarchisch
OM_S := S / (H + S + E) # Omega-Spezifisch

# Analyse des Bifaktormodells bifaktor:
# MLR-Schätzung (estimator="MLR"),
# alle latenten Varianzen auf eins fixiert (std.lv=TRUE),
# alle Faktoren unkorreliert (orthogonal=TRUE)
fit.bifaktor <- sem(bifaktor, data= Kapitel_15_Daten_R, estimator="MLR",
orthogonal=TRUE, std.lv=TRUE)

# Ergebnisse anzeigen
summary(fit.bifaktor)

```

4.2 Modellbasierte Schätzung der Omega-Koeffizienten im Bifaktormodell –
R-Output

```
lavaan 0.6-7 ended normally after 33 iterations

Estimator                               ML
Optimization method                     NLMINB
Number of free parameters                30

Number of observations                    250

Model Test User Model:

Test Statistic                           Standard      Robust
Degrees of freedom                       35.039       32.444
P-value (Chi-square)                      25           25
Scaling correction factor                 0.088       0.146
Yuan-Bentler correction (Mplus variant)  1.080

Parameter Estimates:

Standard errors                           Sandwich
Information bread                         Observed
Observed information based on             Hessian

Latent Variables:

Estimate  Std.Err  z-value  P(>|z|)
CM =~
  CM09    (p1)   -0.102   0.352   -0.291   0.771
  CM21    (p2)    0.282   0.147    1.917   0.055
  CM25    (p3)    0.306   0.215    1.425   0.154
  CM34    (p4)    0.438   0.238    1.837   0.066
DA =~
  DA17    (p5)    0.601   0.148    4.047   0.000
  DA28    (p6)    0.511   0.114    4.483   0.000
  DA32    (p7)    0.455   0.126    3.611   0.000
PS =~
  PS12    (p8)    0.841   0.072   11.649   0.000
  PS19    (p9)    0.846   0.070   12.075   0.000
  PS24    (p10)   0.616   0.073    8.398   0.000
GEN =~
  CM09    (p11)   0.857   0.097    8.797   0.000
  CM21    (p12)   0.793   0.074   10.782   0.000
  CM25    (p13)   0.962   0.084   11.477   0.000
  CM34    (p14)   0.759   0.109    6.957   0.000
  DA17    (p15)   0.697   0.095    7.328   0.000
  DA28    (p16)   0.881   0.081   10.824   0.000
  DA32    (p17)   0.444   0.079    5.659   0.000
  PS12    (p18)   0.411   0.085    4.816   0.000
  PS19    (p19)   0.386   0.073    5.271   0.000
  PS24    (p20)   0.376   0.082    4.608   0.000

Covariances:

Estimate  Std.Err  z-value  P(>|z|)
CM ~~
  DA      0.000
  PS      0.000
  GEN     0.000
DA ~~
  PS      0.000
  GEN     0.000
PS ~~
  GEN     0.000

Variances:

Estimate  Std.Err  z-value  P(>|z|)
.CM09    (e1)   0.736   0.203    3.630   0.000
.CM21    (e2)   0.525   0.071    7.388   0.000
```

.CM25	(e3)	0.420	0.067	6.242	0.000
.CM34	(e4)	0.453	0.131	3.472	0.001
.DA17	(e5)	0.675	0.144	4.681	0.000
.DA28	(e6)	0.658	0.092	7.121	0.000
.DA32	(e7)	0.938	0.115	8.148	0.000
.PS12	(e8)	0.441	0.097	4.542	0.000
.PS19	(e9)	0.392	0.111	3.519	0.000
.PS24	(e10)	0.691	0.082	8.418	0.000
CM		1.000			
DA		1.000			
PS		1.000			
GEN		1.000			
Defined Parameters:					
		Estimate	Std.Err	z-value	P(> z)
cm.g		11.361	1.957	5.807	0.000
cm.spez		0.854	1.462	0.584	0.559
cm.error		2.134	0.222	9.610	0.000
da.g		4.089	0.802	5.099	0.000
da.spez		2.456	0.677	3.628	0.000
da.error		2.271	0.156	14.584	0.000
ps.g		1.376	0.478	2.881	0.004
ps.spez		5.304	0.628	8.440	0.000
ps.error		1.524	0.133	11.418	0.000
om.cm.t		0.851	0.021	40.915	0.000
om.cm.h		0.792	0.112	7.080	0.000
om.cm.s		0.059	0.102	0.585	0.559
om.da.t		0.742	0.028	26.328	0.000
om.da.h		0.464	0.076	6.118	0.000
om.da.s		0.279	0.075	3.718	0.000
om.ps.t		0.814	0.023	35.105	0.000
om.ps.h		0.168	0.053	3.154	0.002
om.ps.s		0.647	0.054	12.025	0.000
H		43.111	4.937	8.733	0.000
S		8.614	1.138	7.568	0.000
E		5.929	0.328	18.050	0.000
OM_T		0.897	0.010	85.553	0.000
OM_H		0.748	0.029	25.550	0.000
OM_S		0.149	0.023	6.543	0.000
# Omega_Total wurde auf 0.897 geschätzt					
# Omega_Hierarchisch wurde auf 0.748 geschätzt					
# Omega_Spezifisch wurde auf 0.149 geschätzt					

4.3 Konfidenzintervall zu den Omega-Koeffizienten im Bifaktormodell

```
# Berechnung der Grenzen des Konfidenzintervalls von Omega
# Die Werte für REL (Wert eines Omega-Koeffizienten) und
# SE (Standardfehler des jeweiligen Omega-Koeffizienten)
# können aus vorheriger Analyse übernommen werden

#REL <- ???
#SE <- ???

## 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
```