Vertiefte Forschungsmethodik der Psychologie und Psychotherapie I

II. Multivariate Statistische Verfahren

Dr. Leonard Pleschberger

Heinrich-Heine-Universität Düsseldorf

WS 25/26

Übersicht

II.1. Multivariate Statistik

- Bisher haben wir uns **univariat** mit einzelnen Merkmalen oder höchstens **bivariat** mit der Beziehung zweier Variablen beschäftigt.
- In der **multivariaten Statistik** betrachten wir nun die Beziehungen *mehrerer* Variablen gleichzeitig.
- Damit werden unsere statistischen Modelle größer und komplexer – aber auch **realitätsnäher**, denn psychologische Konstrukte hängen selten nur von einer einzigen Einflussgröße ab.

II.2. Abhängigkeitsmodelle

content...

II.2.1. Multiple Regression

Fragestellung: Wie kann der lineare Einfluss gleich mehrerer Prädiktoren $X_1, \ldots, X_p$ auf eine Zielvariable Y erklärt werden?

Das Modell der multiplen (linearen) Regression lautet also

$$Y = b_0 + b_1 X_1 + \cdots + b_p X_p + \varepsilon.$$

Hier bezeichnen

- b_0 den Achsenabschnitt (Intercept).
- $b_1, \ldots, b_p$ die Koeffizienten – sie bestimmen den individuellen Einfluss von $X_1, \ldots, X_p$.
- ε einen zufälligen Fehlerterm.

Ziel: Wir möchten die Koeffizienten $b_0, \ldots, b_p$ mit Hilfe von R so schätzen, dass sie unsere Daten am besten fitten.

Multiple Regression: Statistik

Bei n Messungen mit 1 Zielvariable und p Prädikatoren ergibt sich

$$\begin{pmatrix} Y_1 \\ \vdots \\ Y_n \end{pmatrix} = \begin{pmatrix} 1 & X_{11} & \dots & X_{1p} \\ \vdots & \ddots & \ddots & \vdots \\ 1 & X_{n1} & \dots & X_{np} \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_p \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

was wir in fetter Vektorschreibweise noch kompakter schreiben können als

$$\mathbf{Y} = \mathbf{X}\mathbf{b} + \varepsilon.$$

Hier wird $\mathbf{X}$ als *Designmatrix* bezeichnet. Nun kommen in jeder Zeile immer genau die gleichen Koeffizienten $\mathbf{b}$ vor und der Rest besteht ausschließlich aus Messungen und Residuen.

Multiple Regression: Statistik

Wir wollen einen Schätzer $\hat{\mathbf{b}}$ bestimmen, sodass die vorhergesagten Werte $\hat{\mathbf{Y}} = \mathbf{X}\hat{\mathbf{b}}$ möglichst nahe bei den gemessenen Werten $\mathbf{Y}$ liegen. Dazu rechnen wir das Minimum der Summe der quadrierten Vorhersagefehler Q mittels Kurvendiskussion aus:

$$Q\left(\hat{\mathbf{b}}\right) = \left|\mathbf{Y} - \hat{\mathbf{Y}}\right|^2 = \left|\mathbf{Y} - \mathbf{X}\hat{\mathbf{b}}\right|^2 = \left(\mathbf{Y} - \mathbf{X}\hat{\mathbf{b}}\right)^\top \left(\mathbf{Y} - \mathbf{X}\hat{\mathbf{b}}\right)$$

Wie leiten nach dem Schätzer ab und setzen die Ableitung gleich null:

$$\frac{\mathrm{d}Q}{\mathrm{d}\hat{\mathbf{b}}} = -2\mathbf{X}^\top\mathbf{Y} + 2\mathbf{X}^\top\mathbf{X}\hat{\mathbf{b}} = 0 \quad \Leftrightarrow \quad \mathbf{X}^\top\mathbf{X}\hat{\mathbf{b}} = \mathbf{X}^\top\mathbf{Y}.$$

Die rechte Gleichung heißt *Normalengleichung*. Wir invertieren noch Matrix auf der linken Seite und erhalten den Schätzer

$$\hat{\mathbf{b}} = \left(\mathbf{X}^\top\mathbf{X}\right)^{-1}\mathbf{X}^\top\mathbf{Y}.$$

Beispiel Multiple Regression: Depressionsschwere

Fragestellung: Wie beeinflussen die Prädiktoren *Stresslevel* (X_1), *Schlafqualität* (X_2) und *Soziale Unterstützung*(X_3) die Zielvariable *Depressionsschwere* (Y) linear?

Wir stellen das folgende multiple lineare Regressionsmodel auf:

$$\text{Depression} = b_0 + b_1 \cdot \text{Stress} + b_2 \cdot \text{Schlaf} + b_3 \cdot \text{Soziales} + \varepsilon$$

Dies schreiben wir als

$$\mathbf{Y} = \mathbf{Xb} + \varepsilon.$$

- $\mathbf{Y}$ $(n \times 1)$: Vektor der Zielvariablen *Depression*.
- $\mathbf{X}$ $(n \times 4)$: Designmatrix für *Intercept*, *Stress*, *Schlaf* und *Soziales*.
- $\mathbf{b}$ (4×1): Koeffizienten.
- ε $(n \times 1)$: Residuen.

Beispiel Multiple Regression: Daten

Bei $n = 8$ Personen wurden die folgenden Daten erhoben:

ID	Stress	Schlaf	Soziales	Depression
1	80	4	2	30
2	65	6	3	22
3	40	8	5	15
4	90	3	1	35
5	55	7	4	18
6	70	5	2	25
7	85	4	2	29
8	60	6	3	21

Beispiel Multiple Regression: Modell in R

Wir laden die Daten in R und verwenden das *lm*()-Modell wie folgt:

```r
# Import the data
data <- data.frame(
stress     = c(80, 65, 40, 90, 55, 70, 85, 60),
sleep      = c(4, 6, 8, 3, 7, 5, 4, 6),
social     = c(2, 3, 5, 1, 4, 2, 2, 3),
depression = c(30, 22, 15, 35, 18, 25, 29, 21)
)

# Multiple lineare Regression
modell <- lm(depression ~ stress + sleep + social,
    data = data)

# Results
summary(modell)
```

Beispiel Multiple Regression: Ergebnisse in R

R liefert die folgende Ausgabe:

```
Residuals:
    1     2     3     4     5     6     7     8
-0.30  0.65  0.30  1.15  0.20 -0.35 -0.85 -0.80

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  58.1000    16.8919   3.440   0.0263 *
stress       -0.0900     0.1378  -0.653   0.5492
sleep        -5.8500     1.7250  -3.391   0.0275 *
social        1.4000     1.2546   1.116   0.3270
---
Signif. codes:
0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Beispiel Multiple Regression: Interpretation

Des Weiteren:

```
Residual standard error: 0.9287 on 4 degrees of
freedom
Multiple R-squared:  0.9889, Adjusted R-squared:
0.9806
F-statistic: 119.2 on 3 and 4 DF, p-value: 0.0002286
```

- **Koeffizienten**: Die Schätzungen ergeben die Regressionsgleichung

$$\text{Depression} = 58.10 - 0.1 \cdot \text{Stress} - 5.9 \cdot \text{Schlaf} + 1.4 \cdot \text{Soziales}.$$

Lediglich zwei Merkmale stehen signifikant zum Niveau $\alpha = 5\%$ im Zusammenhang mit Depression (gekennzeichnet mit *): Das Intercept mit $p = 0.03 < 0.05$ und der Schlaf mit $p = 0.03 < 0.05$. Stress und soziale Unterstützung haben keinen signifikanten Einfluss in diesem Modell.

Multiple Regression: Interpretation

- **Residuen**: Die Werte geben die Abweichung zwischen den prognostizierten und den tatsächlichen Werten an. Die Vorzeichen sind so zu deuten: Bei -0.30 überschätzt das Modell den Wert, i.e. man muss vom Prognosewert 0.30 abziehen, um auf den tatsächlichen Messwert zu kommen. Analog deutet man ein positives Vorzeichen. Deren Durchschnitt ergibt den *Residuen-Standardfehler* 0.93 mit

$$\begin{aligned} \text{df} &= \#\{\text{Beobachtungen}\} - \#\{\text{Prädikatoren}\} - \#\{\text{Intercept}\} \\ &= 8 - 3 - 1 = 4 \end{aligned}$$

Freiheitsgraden. Die Residuen sind relativ klein $\Rightarrow$ das Model fittet die Daten gut.

Beispiel Multiple Regression: Interpretation

- **Bestimmtheitsmaße**: Mit Pearson's Korrelationskoeffizienten r ergibt sich

$$R^2 = r^2 = 0.99.$$

Das sagt aus, dass das Modell extrem gut fittet. Unter Beachtung der Anzahl der Prädikatoren ergibt sich

$$R^2_{\text{adjusted}} = 0.98,$$

woraus ein sehr geringes Overfitting resultiert.

- **Statistiken**: Mit t-Tests wird die Signifikanz für jeden einzelnen Prädiktor getestet und ein F-Test klärt die Signifikanz des gesamten Regressionsmodells. Deshalb sind die empirischen t- und F-Statistikenaufgeführt. Der Wert $F = 119.2$ zusammen mit $p = 0.0002$ sagt aus, dass das Modell insgesamt hochsignifikant ist.

II.2.2. Mehrebenenanalyse (HLM)

- Die **Mehrebenenanalyse** (engl. *Multilevel Modeling* oder *Hierarchical Linear Modeling*, HLM) ist eine statistische Methode, die verwendet wird, wenn Daten hierarchisch oder geschachtelt organisiert sind – etwa Personen innerhalb von Gruppen, Patient:innen innerhalb von Kliniken, Messzeitpunkte innerhalb von Individuen, etc.
- Die Mehrebenenanalyse ist eine Erweiterung der multiplen Regression, die die Abhängigkeit von Beobachtungen innerhalb übergeordneter Gruppen berücksichtigt. Dabei werden gruppenspezifische Intercepts modelliert, um Unterschiede zwischen Gruppen zu erfassen. Das Residuum eines Patienten setzen sich aus dem individuellen Residuum und einem Gruppenresiduum zusammen.

Beispiel Mehrebenenanalyse: Depression in Kliniken

Fragestellung: Wie erstelle ich eine multiple Regression mit der Zusatzinformation, dass die Patienten in Untergruppen (wie Kliniken) aufgeteilt sind?

Level 1: Personenebene

- **Zielvariable:** Depressionswert via BDI-II (0-63)
- **Prädiktoren**: Stress im Alltag (0–100), Schlafqualität (1–10)

Level 2: Klinikebene (Untergruppen)

- Patienten stammen aus verschiedenen Kliniken (e.g. 5 Kliniken in Deutschland).
- Jede Klinik hat eigene Strukturen (Therapiephilosophie, Personal, Ressourcen).
- **Prädiktor**: Durchschnittliche Anzahl an Psychotherapeut:innen pro 10 Patienten (als Indikator für Therapieintensität).

Beispiel Mehrebenenanalyse: Depression in Kliniken

Gemeinsamkeit zur multiplen Regression: Der Depressionswert des i-ten Patienten aus der j-ten Klinik wird u.a. durch eine Linearkombination von durchschnittlichem Intercept, sowie den Prädikatoren Stress, Schlaf und Therapieintensität (gleich für Patienten der Klinik j) ermittelt:

$$\textbf{Depression}_{ij} = b_0 + b_1 \cdot \textbf{Stress}_{ij} + b_2 \cdot \textbf{Schlaf}_{ij} + b_3 \cdot \textbf{Intensität}_j + \textbf{Fehlerterm}$$

Unterschied zur multiplen Regression: Durch die verfeinerte Struktur wird es möglich den Fehlerterm durch Residuen auf Personenebene ε_{ij} (Level 1) und Residuen auf Klinikebene κ_j (Level 2) zu modellieren.

$\Rightarrow$ In Vektorschreibweise ergibt sich für alle Patienten der Klinik j gerade

$$\boldsymbol{Y}_j = \boldsymbol{X}_j \boldsymbol{b} + \boldsymbol{\varepsilon} + \boldsymbol{\kappa}_j.$$

Beispiel Mehrebenenanalyse: Daten erzeugen

Erzeuge `n` normalverteilte Zufallszahlen mit Mittelwert `mu` und Standardabweichung `s`:

```r
rnorm(n, mean = mu, sd = s) # Default: rnorm(n, 0, 1)
```

```r
set.seed(12)  # Reproducibility

n_clinics  <- 5          # Number of clinics (level 2)
n_patients <- 10         # Patients per clinic
N <- n_clinics * n_patients

# Clinic IDs
clinic <- factor(rep(1:n_clinics, each = n_patients))

# Level-2 predictor: Therapy intensity per clinic
therapy_intensity <- rep(c(1.5, 2.0, 2.5, 3.0, 3.5),
                         each = n_patients)
```

Beispiel Mehrebenenanalyse: Daten erzeugen

```r
# Level-1 predictors (patient characteristics)
stress <- round(rnorm(N, mean = 60, sd = 15), 1)
          # Daily stress (0-100)
sleep  <- round(rnorm(N, mean = 6.5, sd = 1.2), 1)
          # Sleep quality (1-10)

# True effects for simulation
b0  <- 30        # Baseline depression score
b1  <- 0.25      # Effect of stress (per stress point)
b2  <- -1.5      # Effect of sleep (per hour)
g01 <- -2.0      # Effect of th. intensity (level 2)

# Random intercepts for clinics (level 2)
u0 <- rnorm(n_clinics, 0, 2)    # each clinic has a
                                # different baseline
rand_intercepts <- rep(u0, each = n_patients)
```

Beispiel Mehrebenenanalyse: Daten erzeugen

```
# Simulate depression
depression <- b0 + b1 * stress + b2 * sleep + g01 *
 therapy_intensity + rand_intercepts + rnorm(N, 0 ,3)
# Clinical and individual error

# Build final dataset
data <- data.frame(clinic, therapy_intensity, stress,
    sleep, depression)

head(data, 3)
```

Ausgabe:

```
  clinic therapy_intensity stress sleep depression
1      1               1.5   80.6   6.9     36.800
2      1               1.5   51.5   5.6     31.475
3      1               1.5   65.4   8.4     30.750
```

Beispiel Mehrebenenanalyse: Daten auswerten

Zunächst installieren wir **einmal** das Package für die Mehrebenenanalyse:

```
> install.packages("lme4")
```

```
# Load package
library(lme4)

# Random intercept model with a Level-2 predictor
model <- lmer(depression ~ stress + sleep +
        therapy_intensity + (1 | clinic), data = data)

summary(model)
```

- Zielvariable: `depression`.
- Prädikatoren: `stress`, `sleep`, `therapy_intensity`.
- `(1 | clinic)` → Füfälliger Interceopt für jedes `clinic`.

Beispiel Mehrebenenanalyse: Output und Interpretation

```
Linear mixed model fit by REML ['lmerMod']
Formula:
depression ~ stress + sleep + therapy_intensity +
             (1 | clinic)
Data: data

REML criterion at convergence: 255.3

Scaled residuals:
     Min       1Q   Median       3Q      Max
-1.86553 -0.66244  0.03221  0.68334  1.93486
```

- Die Residuen sollten idealerweise gleichmäßig um 0 verteilt sein.
- Median $\approx 0 \Rightarrow$ Die Modellfehler mitteln sich nahezu aus.
- Fazit: Das Modell passt die Daten gut an.

Beispiel Mehrebenenanalyse: Output und Interpretation

Abweichungen zwischen den Kliniken (Level 2) und auf Patientenebene (Level 1):

```
Random effects:
Groups   Name          Variance Std.Dev.
clinic   (Intercept) 3.999    2.000
Residual             8.722    2.953
Number of obs: 50, groups:  clinic, 5
```

- `clinic (Intercept)`: Die Kliniken (Level 2) unterscheiden sich leicht im Grundniveau der Depression (zwischen-Gruppen-Varianz).
- `Residual`: Die gruppenunspezifische Reststreuung auf Patientenebene (Level 1); das, was durch die Prädikatoren nicht erklärt wird.

Beispiel Mehrebenenanalyse: Output und Interpretation

Nun die Schätzungen der Koeffizienten und deren Signifikanz (Daumenregel: $|\text{t-Wert}| > 2 \Rightarrow$ Merkmal ist signifikant):

```
Fixed effects:
                   Estimate Std. Error t value
(Intercept)         38.1546     5.1406   7.422
stress               0.2768     0.0335   8.263
sleep               -1.9853     0.4335  -4.580
therapy_intensity   -4.0916     1.4002  -2.922
```

- Pro 1 Stresspunkt: Zunahme von 0.3 Depressionspunkten.
- Pro 1 Stunde Schlaf: Abnahme von 2 Depressionspunkten.
- Pro 1 Therapeut: Abnahme von 4 Depressionspunkten.

Beispiel Mehrebenenanalyse: Output und Interpretation

Zuletzt betrachten wir noch die Korrelationen zwischen den geschätzten Prädiktoren:

```
Correlation of Fixed Effects:
                    (Intr) stress sleep
stress              -0.376
sleep               -0.618  0.049
therapy_intensity   -0.701 -0.041  0.065
```

Eine geringe Korrelation bedeutet eine gute Trennbarkeit der Prädiktoren.

II.2.3. Multivariate Varianzanalyse (MANOVA)

Fragestellung: Wie beeinflussen Stress und Schlaf die gleichzeitigen Ausprägungen mehrerer psychologischer Zielgrößen wie z.B. Depressivität, Ängstlichkeit und Erschöpfung?

content...

II.3. Interdependenzmodelle

Interdependenzmodelle decken Zusammenhänge zwischen Items oder Fällen auf, ohne zwischen unabhängigen und abhängigen Variablen zu unterscheiden. Wir betrachten Faktoren- und Clusteranalysen.

Es gibt zwei Arten von Faktorenanalysen:

- **Exploratorische Faktorenanalyse (EFA)**: Man möchte die Dimensionsstruktur von gegebenen Items aufdecken und darüber Hypothesen generieren (Modellgewinnung).
- **Konfirmatorische Faktorenanalyse (CFA)**: Es werden vorab konkrete Hypothesen über die Dimensionsstruktur der Items formuliert und diese werden dann getestet (Modellprüfung).

Die **Clusteranalyse** hat das Ziel, Fälle (v.a. Personen) anhand von Ähnlichkeiten zu gruppieren – *ohne* vorherige Annahmen.

II.3.1. Exploratorische Faktorenanalyse (EFA)

Fragestellung: Welche übergeordneten, latenten Konstrukte (Faktoren) liegen den beobachteten Merkmalen (Items) zu Grunde? *Oder*: Welche Items laden auf denselben Faktor und messen somit dasselbe latente Merkmal?

Ziele:

- **Skalenbildung**: Welche Items messen tatsächlich denselben Faktor und können somit sinnvoll zu einer Skala zusammengefasst werden?
- **Theoriegewinn**: Die EFA generiert Hypothesen; durch die mathematische Reduktion können sich neue, strukturelle Zusammenhänge ergeben.
- **Konstruktvalidierung**: Messen die Items tatsächlich das, was sie messen sollen? Welche Dimensionsstruktur ergibt sich für die Faktoren?

Exploratorische Faktorenanalyse (EFA): Statistik

Die EFA geht davon aus, dass n beobachtete Items $X_1, \ldots, X_n$ durch eine Linearkombination aus p gemeinsamen Faktoren $F_1, \ldots, F_p$ und einem spezifischen Fehlerterm $\varepsilon_1, \ldots, \varepsilon_n$ erklärt werden. In Zeichen:

$$X_i = \lambda_{i1} F_1 + \cdots + \lambda_{ip} F_p + \varepsilon_i,$$

was gerade

$$\begin{pmatrix} X_1 \\ \vdots \\ X_n \end{pmatrix} = \begin{pmatrix} \lambda_{11} & \cdots & \lambda_{1p} \\ \vdots & \ddots & \vdots \\ \lambda_{n1} & \cdots & \lambda_{np} \end{pmatrix} \begin{pmatrix} F_1 \\ \vdots \\ F_p \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix}$$

ergibt. In fetter Vektorschreibweise mit der **Ladungsmatrix $\mathbf{\Lambda}$** können wir das noch kompakter schreiben als

$$\mathbf{X} = \mathbf{\Lambda F} + \varepsilon.$$

Exploratorische Faktorenanalyse (EFA): Statistik

Wir interessieren dafür, wie stark die Items miteinander zusammenhängen. Wir berechnen also Ihre **empirischen Kovarianzen**. Diese werden in der Kovarianzmatrix

$$\text{Cov}(\mathbf{X}) = \Sigma = \begin{pmatrix} \text{Var}(X_1) & \text{Cov}(X_1, X_2) & \cdots \\ \text{Cov}(X_1, X_2) & \text{Var}(X_1) & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}$$

zusammengefasst. Wir setzen nun $\mathbf{X} = \mathbf{\Lambda F} + \varepsilon$ oben ein. Man erhält

$$\Sigma = \text{Cov}(X) = \text{Cov}(\mathbf{\Lambda F} + \varepsilon) = \mathbf{\Lambda}\text{Cov}(F)\mathbf{\Lambda}^\top + \text{Cov}(\varepsilon) = \mathbf{\Lambda \Phi \Lambda}^\top + \mathbf{\Psi}$$

mit

- $\mathbf{\Phi}$: Kovarianzmatrix der Faktoren.
- $\mathbf{\Psi}$: Diagonalmatrix der Fehlervarianzen der Items.

Exploratorische Faktorenanalyse (EFA): Statistik

Nimmt man an, dass die Faktoren unabhängig voneinander sind, so ergibt sich die Einheitsmatrix $\mathbf{\Phi} = \mathrm{I}_n$. In diesem Fall gilt:

$$\mathbf{\Sigma} = \mathbf{\Lambda}\mathbf{\Lambda}^\top + \mathbf{\Psi}$$

Das bedeutet: Die beobachtete Gesamtvarianz lässt sich zerlegen in

- eine *gemeinsame Varianz* (erklärt durch Faktoren),
- eine *spezifische Varianz* (nicht erklärbar, z.B. Messfehler).

Das wäre der Idealfall: Wir könnten die Gesamtvarianz durch möglichst wenige Ursachen (Faktoren und Residuen) erklären. Also versuchen wir in der EFA mathematisch $\hat{\mathbf{\Lambda}}$ und $\hat{\mathbf{\Psi}}$ so zu schätzen, dass sich gerade

$$\mathbf{\Sigma} \approx \hat{\mathbf{\Lambda}}\hat{\mathbf{\Lambda}}^\top + \hat{\mathbf{\Psi}}$$

ergibt.

Exploratorische Faktorenanalyse (EFA): Statistik

„Ein Vergleich mit der Multiplen Regression macht die unterschiedlichen Denkweisen bei Abhängigkeits- und Interdependenzmodell deutlich: Bei der Multiplen Regression ist die Zielvariable (latentes Merkmal) *zusammengesetzt aus* gewichteten Prädikatoren (manifeste Merkmale); bei der EFA hingegen *lädt* das Item (manifestes Merkmal) *auf* die Faktoren (latenten Merkmale) – eine inverse Logik!“

II.3.2. Konfirmatorische Faktorenanalyse (CFA)

Notation nach Jöreskog (1960er).

Es werden vorab konkrete Hypothesen über die Dimensionsstruktur der Items formuliert und diese werden dann getestet (Modellprüfung).

content...

Konfirmatorische Faktorenanalyse (CFA): Statistik

II.3.3. Clusteranalyse

Die **Clusteranalyse** hat das Ziel, Personen oder Items anhand von Ähnlichkeiten zu gruppieren – *ohne* vorherige Annahmen.

Wir wollen anhand von standardisierten Beobachtungen u.a.

- Items zu Symptomkomplexen gruppieren,
- Persönlichkeitsprofile per Big-Five-Dimensionen extrahieren,
- Antwortmuster auf Skalen entdecken,
- klinische Subtypen mit unterschiedlichen Therapieansprachen finden.

content...

Clusteranalyse: Beispiel Depressionsschwere

Wir betrachten fünf Patienten: *Alice*, *Bob* und *Chris*, *Daniel* und *Elias*. Wir möchten sie entsprechend einer Erhebung via BDI-II entsprechend Ihrer Depressionsschwere in zwei Cluster aufteilen. Wir erheben die kognitiv-affektiven Items *Traurigkeit* (Item 1) und *Schuldgefühle* (Item 5) sowie die somatisch-affektiven Items *Müdigkeit / Energieverlust* (Item 15) und *Schlafprobleme* (Item 16), je (0-3). Wir erhalten die folgenden Rohdaten.

Items =	(Traurigkeit,	Schuldgefühle,	Müdigkeit,	Schlafprobleme)
Alice	3	3	2	3
Bob	1	0	1	1
Chris	0	0	0	1
Daniel	2	1	3	2
Elias	2	2	3	3

Clusteranalyse: Beispiel Depressionsschwere

Zunächst standardisieren wir die Daten spaltenweise. Wir erhalten die folgenden Werte als Datenmatrix, deren Zeilen wir als Personen schreiben:

$$\mathbf{Z} = \begin{pmatrix} 1.37 & 1.54 & 0.17 & 1.12 \\ -0.59 & -1.03 & -0.69 & -1.12 \\ -1.57 & -1.03 & -1.54 & -1.12 \\ 0.39 & -0.17 & 1.03 & 0.00 \\ 0.39 & 0.69 & 1.03 & 1.12 \end{pmatrix} = \begin{pmatrix} \textbf{Alice} \\ \textbf{Bob} \\ \textbf{Chris} \\ \textbf{Daniel} \\ \textbf{Elias} \end{pmatrix}$$

Clusteranalyse: Hierarchisch agglomerativ mit Ward

Wir führen eine **hierarchisch agglomerative Clusteranalyse** durch, speziell: **Ward's Methode**. Wir starten mit je genau einer Person als eigenes Cluster. Pro Schritt fusionieren wir immer je zwei Cluster so, dass der Anstieg der unverzerrten Varianz innerhalb der Cluster minimal ist.

Schritt 1: Nur Singletons. Es ergibt sich die symmetrische Distanzmatrix

	Alice	**Bob**	**Chris**	**Daniel**	**Elias**
Alice	0	8.10	11.60	2.94	1.21
Bob		0	**0.84**	2.96	5.95
Chris			0	6.22	9.21
Daniel				0	1.00
Elias					0

Clusteranalyse: Hierarchisch agglomerativ mit Ward

Schritt 2: Wegen des minimalen Varianzzuwachses ergibt sich das erste Cluster zu $\{\textbf{Bob}, \textbf{Chris}\}$. Wir erhalten die neue Distanzmatrix

	Alice	**{Bob, Chris}**	**Daniel**	**Elias**
Alice	0	4.47	2.94	**1.21**
{Bob, Chris}		0	1.97	3.41
Daniel			0	**1.00**
Elias				0

Clusteranalyse: Hierarchisch agglomerativ mit Ward

Schritt 3: Da die minimalen Distanzen bei zwei Singletons auftauchen, können wir diese direkt zu {**Alice, Daniel, Elias**} clustern. Jetzt kann nur noch das Gesamtcluster gebildet werden und der Algorithmus hält. Wir könnnen nun einfach unsere $k = 2$ Cluster auswählen.

Durch Summation der bisherigen Distanzen innerhalb der beiden Cluster und unter Berücksichtigung derer Größe erhalten wir die **unverzerrten within-Varianzen**

$$\hat{\sigma}^2_{\text{within}}(\{\textbf{Bob, Chris}\}) = 0.84$$

sowie

$$\hat{\sigma}^2_{\text{within}}(\{\textbf{Alice, Daniel, Elias}\}) = 1.72.$$

Clusteranalyse: Hierarchisch divisiv mit Single Linkage

Wir führen eine **hierarchisch divisive Clusteranalyse** mittels **Single Linkage** durch. Zu Beginn bilden alle Personen ein großes Cluster. Pro Schritt teilen wir ein Cluster in zwei Teilcluster auf, sodass die Distanz zwischen ihnen möglichst groß wird.

Schritt 1: Ein Gesamtcluster. Die Euklidische Distanzmatrix lautet

	Alice	**Bob**	**Chris**	**Daniel**	**Elias**
Alice	0	4.01	**4.82**	2.42	1.56
Bob		0	1.30	2.43	3.45
Chris			0	3.53	4.29
Daniel				0	1.41
Elias					0

Clusteranalyse: Hierarchisch divisiv mit Single Linkage

Die größte Distanz tritt bei dem Paar {**Alice**, **Chris**} auf. Ob wir nun Alice oder Chris ausclustern, hängt nun vom Single Linkage ab: Das minimale Linkage bei Alice beträgt 1.56, das bei Chris gerade 1.30. Damit hat Chris das stärkere Linkage und Alice scheidet aus dem Cluster aus.

Schritt 2: Wir wiederholen das Verfahren beim Cluster {**Bob**, **Chris**, **Daniel**, **Elias**}. Unsere Distanzmatrix besitzt weiterhin Gültigkeit. Zwischen Chris und Elias herrscht nun die größte Distanz. Chris hat zu Bob mit 1.30 ein besseres minimales Linkage als Elias ein solches mit 1.41 zu Daniel besitzt. Somit scheidet Elias aus dem Cluster aus. Übrig bleibt das Cluster {**Bob**, **Chris**, **Daniel**}.

Schritt 3: Eine weitere Iteration liefert das Restcluster {**Bob**, **Daniel**}. Dies kann nun nur noch in Singletons gespalten werden und der Algorithmus hält.

Clusteranalyse: Partitionierend mit k-Means & k-Means++

Wir führen eine k-**Means Clusteranalyse** durch mit der k-**Means++** Methode. Dazu geben wir vorab eine fixe Anzahl von k (hier: $k = 2$) gewünschten Clustern vor. Mit k-Means++, einem probabilistischen Verfahren, das Distanzen mit einbezieht, wählen wir k Clusterzentren aus (Schritte 1-3). Gemäß ihrem Abstand werden die Namen den k Clustern zugeordnet. Aus diesen Clustern werden neue Clusterzentren berechnet und es erfolgt eine Cluster-Zuordnung der Namen zu den neuen Zentren (Schritte 4-6). Die letzten beiden Schritte werden einer gewissen Anzahl gemäß wiederholt oder der Algorithmus hält, wenn die Zuweisungen unverändert bleiben.

Schritt 1: Wir wählen einen zufälligen Personenvektor als Clusterzentrum aus. Dies ergibt in unserem Fall

$$\mu_1^{(0)} = \textbf{Alice}.$$

Clusteranalyse: Partitionierend mit k-Means & k-Means++

Schritt 2: Wir berechnen die quadrierten Abstände $D(\boldsymbol{z})^2$ der Personen zu μ_1. Dies ergibt die Distanzmatrix

$D(\boldsymbol{z})^2$	**Alice**	**Bob**	**Chris**	**Daniel**	**Elias**	$\sum$
$\mu_1 =$ **Alice**	0	16.20	23.19	5.88	2.42	47.69

Den Personen ordnen wir Wahrscheinlichkeiten proportional zu $D(\boldsymbol{z})^2$ zu, i.e. weiter entfernte Personen werden mit einer größeren Wahrscheinlichkeit ausgewählt. Wir erhalten

	Alice	**Bob**	**Chris**	**Daniel**	**Elias**	$\sum$
$\mathbb{P}(\boldsymbol{z})$	0	0.34	0.49	0.12	0.05	1

Clusteranalyse: Partitionierend mit k-Means & k-Means++

Schritt 3: Gemäß den Wahrscheinlichkeiten ziehen wir eine Person als weiteres Clusterzentrum. Für uns ergibt sich

$$\mu_2^{(0)} = \textbf{Chris}.$$

Für $k = 2$ ist damit die Initialisierung abgeschlossen.

Schritt 4: Clustern. Berechne die Euklidischen Distanzen der Personen zu den Zentren. Dann weisen wir einen Namen dem Cluster $c^{(0)}$ mit dem nächsten Zentrum zu. Wir erhalten die Distanz- und Clustermatrix

	Alice	**Bob**	**Chris**	**Daniel**	**Elias**
$\mu_1^{(0)}$	**0**	4.03	4.82	**2.42**	**1.56**
$\mu_2^{(0)}$	4.82	**1.30**	**0**	3.53	4.29
$c^{(0)}$	1	2	2	1	1

Clusteranalyse: Partitionierend mit k-Means & k-Means++

Schritt 5: Zentren updaten. Berechne komponentenweise das arithmetische Mittel der Personen pro Cluster {**Alice**, **Daniel**, **Elias**} und {**Bob**, **Chris**}. Wir erhalten neue Zentren

$$\mu_1^{(1)} = (0.72, 0.69, 0.74, 0.75) \quad \text{und} \quad \mu_2^{(1)} = (-1.08, -1.03, -1.12, -1.12).$$

Schritt 6: Clustern. Wie in Schritt 4 erhalten die wir die Matrix

	Alice	**Bob**	**Chris**	**Daniel**	**Elias**
$\mu_1^{(1)}$	**1.27**	3.19	4.11	**1.22**	**0.57**
$\mu_2^{(1)}$	4.39	**0.65**	**0.65**	2.96	3.84
$c^{(1)}$	1	2	2	1	1

Die Cluster bleiben unverändert und der k-Means Algorithmus hält.

Clusteranalyse: Latent Profile Analysis (LPA) nach Bayes

Wir führen ein modellbasiertes **Latent Profile Analysis (LPA)** durch. Hierbei wird davon ausgegangen, dass unsere Messwerte aus verschiedenen „versteckten“ Profilen stammen. Für jede Person berechnet das Modell die Wahrscheinlichkeiten, mit denen sie zu jedem Profil gehört. Es ergibt probabilistische Zuordnungen jeder Person zu den Profilen.

Wir nehmen $k = 2$ Profile an. Jedes Profil besitze eine Gaußsche Verteilung mit typischen Mittelwerten $\mu_1,\ \mu_2$ und Streuungen $\Sigma_1,\ \Sigma_2$.

Die Gesamtverteilung ist eine Mischung dieser 2 Profile, ein **Gaußsches Mischmodell**: Jede Person $\boldsymbol{z}$ landet mit einer gewissen Wahrscheinlichkeit $p_{z,i}$ in jedem Profil i für $i = 1, 2$.

Clusteranalyse: Latent Profile Analysis (LPA) nach Bayes

Wir wollen die Wahrscheinlichkeit berechnen, dass Person **x** in Profil 1 oder 2 landet. Nach dem Satz von Bayes und dem Satz von der totalen Wahrscheinlichkeit haben wir also für $i = 1, 2$ gerade

$$\boldsymbol{p}_i(\boldsymbol{z}) = \overbrace{\mathbb{P}(\text{Profil } i \mid \boldsymbol{z})}^{\text{Posterior-Wahrscheinlichkeit}}$$

$$= \frac{\overbrace{\mathbb{P}(\text{Profil } i)}^{\text{Prior-Wahrscheinlichkeit}} \cdot \overbrace{\mathbb{P}(\boldsymbol{z} \mid \text{Profil } i)}^{\text{Likelihood von } \boldsymbol{z} \text{ gegeben Profil } i}}{\mathbb{P}(\boldsymbol{z})}$$

$$= \frac{\mathbb{P}(\text{Profil } i) \cdot \mathbb{P}(\boldsymbol{z} \mid \text{Profil } i)}{\mathbb{P}(\text{Profil } 1) \cdot \mathbb{P}(\boldsymbol{z} \mid \text{Profil } 1) + \mathbb{P}(\text{Profil } 2) \cdot \mathbb{P}(\boldsymbol{z} \mid \text{Profil } 2)}$$

$$= \frac{\pi_i \cdot f_i(\boldsymbol{z})}{\pi_1 \cdot f_1(\boldsymbol{z}) + \pi_2 \cdot f_2(\boldsymbol{z})}$$

Clusteranalyse: Latent Profile Analysis (LPA) nach Bayes

Schritt 1 (Initialisierung): Zuerst gehen wir von gleichen Mischungsgewichten $\pi_{1,2}$ aus. Dann lassen wir den $(k = 2)$-*Means++* Algorithmus laufen, um die initialen Mittelwerte aus der ersten Iteration zu gewinnen. Typische Startwerte sind daher

$$\pi_1^{(0)} = \pi_2^{(0)} = \tfrac{1}{2},$$

$$\mu_1^{(0)} = (0.72,\, 0.69,\, 0.74,\, 0.75), \quad \mu_2^{(0)} = (-1.08,\, -1.03,\, -1.12,\, -1.12).$$

Die Kovarianzen schätzen wir mit $\hat{\Sigma}^{(0)} = \frac{1}{n} Z^\top Z$ für $n = 5$ Personen zu

$$\hat{\Sigma}^{(0)} = \begin{pmatrix} 1.00 & 0.91 & 0.77 & 0.88 \\ 0.91 & 1.00 & 0.62 & 0.96 \\ 0.77 & 0.62 & 1.00 & 0.77 \\ 0.88 & 0.96 & 0.77 & 1.00 \end{pmatrix}$$

Clusteranalyse: Latent Profile Analysis (LPA) nach Bayes

Schritt 2 (Expectation): Da wir von Gauß-verteilten Profilen ausgehen, können wir mit den Mittelwerten und der Kovarianzmatrix die Likelihoods für die Profile 1 und 2 ausrechnen. Die Formel der multivariaten Dichte der Gauß-Verteilung lautet

$$f_{1,2}(\mathbf{z}) = \frac{1}{(2\pi)^{p/2}\,|\hat{\Sigma}^{(0)}|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{z}-\boldsymbol{\mu}_{1,2})^{\top}\left(\hat{\Sigma}^{(0)}\right)^{-1}(\mathbf{z}-\boldsymbol{\mu}_{1,2})\right)$$

Eingesetzt ergeben sich die Werte

	Alice	**Bob**	**Chris**	**Daniel**	**Elias**
$f_1^{(0)}(\mathbf{z})$	0.3849	0.0679	0.0659	0.2683	0.2739
$f_2^{(0)}(\mathbf{z})$	0.0439	0.5556	0.5009	0.0291	0.0386

Clusteranalyse: Latent Profile Analysis (LPA) nach Bayes

Daraus lassen sich für $i = 1, 2$ die Posterior-Wahrscheinlichkeiten $\boldsymbol{p}_i^{(0)}(\mathbf{z}) = \mathbb{P}(\text{Profil } i \mid \boldsymbol{z})$ berechnen. Wir erhalten

	Alice	**Bob**	**Chris**	**Daniel**	**Elias**
$\boldsymbol{p}_1^{(0)}(\mathbf{z}) =$	(0.8920,	0.1139,	0.1120,	0.8904,	0.8923)
$\boldsymbol{p}_2^{(0)}(\mathbf{z}) =$	(0.1081,	0.8861,	0.8880,	0.1096,	0.1077)

Schritt 3 (Maximization): Mit den gewonnenen $\boldsymbol{p}_i^{(0)}(\mathbf{z})$ berechnen wir für $i = 1, 2$ neue Mischungsgewichte

$$\pi_1^{(1)} = 0.5801, \quad \pi_2^{(1)} = 0.4200.$$

Clusteranalyse: Latent Profile Analysis (LPA) nach Bayes

Ferner ergeben sich neue Mittelwerte

$$\mu_1^{(1)} = (\ \ 0.5772,\ \ 0.5535,\ \ 0.5988,\ \ 0.6018),$$
$$\mu_2^{(1)} = (-0.8023, -0.7646, -0.8273, -0.8314)$$

sowie die Updates der Kovarianzmatrizen

$$\begin{pmatrix} 0.4463 & 0.4780 & 0.0822 & 0.3527 \\ 0.4780 & 0.6609 & 0.0031 & 0.5249 \\ 0.0822 & 0.0031 & 0.4126 & 0.1500 \\ 0.3527 & 0.5249 & 0.1500 & 0.5072 \end{pmatrix}, \begin{pmatrix} 0.6583 & 0.4460 & 0.5844 & 0.4571 \\ 0.4460 & 0.4598 & 0.3791 & 0.4676 \\ 0.5844 & 0.3791 & 0.6309 & 0.4369 \\ 0.4571 & 0.4676 & 0.4369 & 0.4977 \end{pmatrix},$$

Clusteranalyse: Latent Profile Analysis (LPA) nach Bayes

Dies ergibt schließlich

	Alice	**Bob**	**Chris**	**Daniel**	**Elias**
$\boldsymbol{p}_1^{(1)}(\mathsf{z}) =$	(0.9997,	0.0000,	0.0000,	0.9997,	0.9997)
$\boldsymbol{p}_2^{(1)}(\mathsf{z}) =$	(0.0003,	1.0000,	1.0000,	0.0003,	0.0003)

Wir sehen, dass sich die Wahrscheinlichkeiten viel stärker der 0, bzw. der 1 angenähert haben. Die Näherungen sind hinreichend genau, sodass der Algorithmus halten kann und wir die Cluster

$$\{\textbf{Alice}, \textbf{Daniel}, \textbf{Elias}\}, \quad \{\textbf{Bob}, \textbf{Chris}\}$$

erhalten.

Clusteranalyse: Statistik (Datenaufbereitung)

Datenbasis: Wir tragen pro Person (Zeilen) die p gemessenen Variablen in eine Matrix $\boldsymbol{X}$ ein.

$$\boldsymbol{X} = \begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix} = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_n \end{pmatrix} \Bigg\} \; n \text{ Personen}$$

Oft führt man spaltenweise eine Standardisierung

$$z_{ij} = \frac{x_{ij} - \bar{x}_j}{s_j}$$

durch, damit Personen gleich gewichtet werden.

Clusteranalyse: Statistik (Datenaufbereitung)

Es ergeben sich die standardisierten Daten

$$\boldsymbol{Z} = \begin{pmatrix} z_{11} & z_{12} & \cdots & z_{1p} \\ z_{21} & z_{22} & \cdots & z_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ z_{n1} & z_{n2} & \cdots & z_{np} \end{pmatrix} = \begin{pmatrix} \boldsymbol{z}_1 \\ \boldsymbol{z}_2 \\ \vdots \\ \boldsymbol{z}_n \end{pmatrix}$$

Nun müssen wir definieren, wie wir die Distanz (Metrik) zwischen zwei standardisierten Personen $\boldsymbol{z_a}, \boldsymbol{z_b}$ messen. Wir nehmen die zwei betreffenden Zeilen und bilden spaltenweise die Differenz beider Items, i.e. $z_{a1} - z_{b1}, \ldots, z_{ap} - z_{bp}$. Diese Differenzen wollen wir nun auf gewisse Arten summieren. Möglichkeiten hierfür sind:

Clusteranalyse: Statistik (Distanzen)

Euklidische Metrik:

$$d(\boldsymbol{z_a}, \boldsymbol{z_b}) = \sqrt{\sum_{j=1}^{p} (z_{aj} - z_{bj})^2} = \|\boldsymbol{z_a} - \boldsymbol{z_b}\|_2$$

Konkret an unserer Matrix $\mathbf{Z}$:

$$d(\boldsymbol{z}_1, \boldsymbol{z}_2) = \sqrt{(z_{11} - z_{21})^2 + (z_{12} - z_{22})^2 + \cdots + (z_{1p} - z_{2p})^2}$$

Die Differenzen werden hier durch Quadrieren stets positiv gemacht und wir summieren diese Quadrate zu einer Gesamtdistanz auf. Das Quadrieren machen wir anschließend noch durch eine Wurzel wett. Die Notation auf der rechten Seite wird als **Euklidische Norm** des Distanzvektors bezeichnet und misst lediglich dessen Größe.

Clusteranalyse: Statistik (Distanzen)

Manhattan-Metrik:

$$d\left(\boldsymbol{z_a}, \boldsymbol{z_b}\right) = \sum_{j=1}^{p} |z_{aj} - z_{bj}| = \|\boldsymbol{z_a} - \boldsymbol{z_b}\|_1$$

Konkret an unserer Matrix $\mathbf{Z}$:

$$d\left(\boldsymbol{z}_1, \boldsymbol{z}_2\right) = |z_{11} - z_{21}| + |z_{12} - z_{22}| + \cdots + |z_{1p} - z_{2p}|$$

Die Differenzen werden hier durch den Betrag stets positiv gemacht und wir summieren diese Beträge zu einer Gesamtdistanz auf. Die Notation auf der rechten Seite wird als **Manhattan-Norm** des Distanzvektors bezeichnet und misst ebenfalls dessen Größe.

Clusteranalyse: Statistik (Distanzen)

Korrelationsbasierte Distanz:

$$d\left(\boldsymbol{z_a}, \boldsymbol{z_b}\right) = 1 - r_{\boldsymbol{z_a}\boldsymbol{z_b}}$$

Wird verwendet, um Personen zu gruppieren, deren Profilform ähnlich ist – unabhängig von der Höhe der Werte. Mit **Pearson's Korrelationskoeffizienten** $r \in [-1, 1]$ gilt gerade $0 \leq d \leq 2$ mit

Distanz	Korrelation r	Interpretation
$d = 0$	$r = 1$	perfekte positive Korrelation
$d = 1$	$r = 0$	keine Korrelation
$d = 2$	$r = -1$	perfekte negative Korrelation

Clusteranalyse: Statistik (Distanzen)

Auf diese Weise bestimmen wir sämtliche paarweisen Distanzen von Personen. Es ergibt sich die Distanzmatrix

$$\mathbf{D} = \begin{pmatrix} d(\boldsymbol{z}_1, \boldsymbol{z}_1) = 0 & d(\boldsymbol{z}_1, \boldsymbol{z}_2) & \cdots & d(\boldsymbol{z}_1, \boldsymbol{z}_p) \\ d(\boldsymbol{z}_2, \boldsymbol{z}_1) & d(\boldsymbol{z}_2, \boldsymbol{z}_2) = 0 & \cdots & d(\boldsymbol{z}_2, \boldsymbol{z}_p) \\ \vdots & \vdots & \ddots & \vdots \\ d(\boldsymbol{z}_n, \boldsymbol{z}_1) & d(\boldsymbol{z}_n, \boldsymbol{z}_2) & \cdots & d(\boldsymbol{z}_n, \boldsymbol{z}_n) = 0 \end{pmatrix}$$

Wir können jetzt Daten systematisch in Personen speichern und deren Distanzen bestimmen. Nun benötigen wir nur noch geeignete Algorithmen, um die Daten sukzessive zu clustern. Wir unterscheiden:

Clusteranalyse: Statistik (Algorithmen)

Hierarchische Verfahren
Es wird eine Baumstruktur (Dendrogramm) erstellt, die alle Objekte und ihre Fusionsschritte zeigt.

- **Agglomerativ:** Jedes Merkmal bildet zunächst ein einzelnes Cluster. Schrittweise erweitern wir die Cluster, bis alle Merkmale in einem einzigen großen Cluster sind. Pro Schritt werden jeweils die zwei ähnlichsten Cluster zusammengefasst.
- **Divisiv:** Man beginnt mit einem einzigen großen Cluster und teilt es schrittweise in kleinere Cluster auf.

Partitionierende Verfahren (e.g. k-Means)

- Wähle k Startzentren. Weise Punkte dem nächsten Zentrum zu. Berechne neue Zentren. Wiederholen bis Konvergenz.

Clusteranalyse: Statistik (Algorithmen)

II.4. Strukturgleichungsmodelle (SEM)

Ein Strukturgleichungsmodell besteht aus zwei Teilen:

- **Messmodel**: Wie werden latente Konstrukte / Faktoren durch beobachtbare Items gemessen? Prinzip: **Konfirmatorische Faktorenanalyse (CFA)**.
- **Strukturmodell**: Wie werden Pfade (gerichtete Zusammenhänge) zwischen latenten Konstrukten beschrieben? Prinzip: **(Multivariate) Regression** mit latenten Konstrukten als Prädikatoren und Zielvariablen.

content...

II.5. Meta-Analyse

content...