# Evaluation und Metaanalyse

MSc Klinische Psychologie und Psychotherapie

SoSe 2024

Prof. Dr. Dirk Ostwald

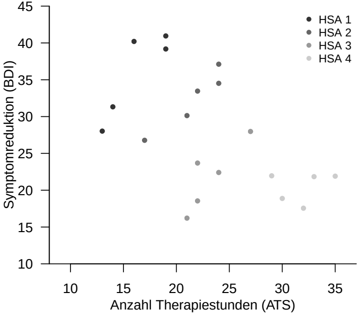
# (6) Mehrebenenanalyse

## Multizentrenstudie

- Studie zum Effekt von Therapiestundenanzahl auf Reduktion der Depressionssymptomatik
- Multizentrendesign mit $m = 4$ Hochschulambulanzen mit jeweils $n_i = 5$ Patient:innen
- Pre-Post-BDI-II Differenz `dBDI` als primäres Ergebnismaß (Positive Werte = Reduktion)
- `HSA`: Hochschulambulanz, `ATS`: Anzahl Therapiestunden, `BDI`: Pre-Post-BDI-II Differenz

```
   HSA ATS BDI
1    1  19  39
2    1  13  28
3    1  14  31
4    1  16  40
5    1  19  41
6    2  17  27
7    2  24  37
8    2  24  35
9    2  22  33
10   2  21  30
11   3  21  16
12   3  22  24
13   3  22  19
14   3  27  28
15   3  24  22
16   4  33  22
17   4  30  19
18   4  32  18
19   4  35  22
20   4  29  22
```

Multizentrenstudie




Zur Analyse wollen wir das R Paket `nlme` nach Pinheiro and Bates (2000) nutzen.

```r
install.packages("nlme")                # Installation des R Pakets
library(nlme)                           # Invokation der R Pakets
```

# Analyse einer Multizentrenstudie mit `nlme`

## `lm()` in R

- `lm(formula, data)` erlaubt das evaluieren linearer Modelle der Form

$$y = X\beta + \varepsilon \text{ mit } \varepsilon \sim N(0_n, \sigma^2 I_n): \quad (1)$$

- `formula` spezifiziert die Form und die Variablen des Modells als R Formel.
- `data` spezifiziert die Daten der Variablen des Modells als R Dataframe.

## `lmList()` in `nlme`

- `lmList(formula | grouping, data)` erlaubt das evaluieren gruppenspezifischer linearer Modelle

$$y_i = X_i\beta_i + \varepsilon_i \text{ mit } \varepsilon_i \sim N(0_{n_i}, \sigma_i^2 I_n) \text{ für } i = 1, ..., m: \quad (2)$$

- `formula` spezifiziert die Form und die Variablen jedes der $m$ Modelle als R Formel.
- `grouping` bildet die Datengruppierungen als R Faktor ab.
- `data` spezifiziert die Daten der Variablen des Modells als R Dataframe.

## `lme()` in `nlme`

- `lme(fixed, data, random)` erlaubt das evaluieren von Linear Mixed Models der form

$$y = X_f\beta_f + X_r\beta_r + \varepsilon \text{ mit } \beta_r \sim N(0_q, \Sigma_{\beta_r}) \text{ und } \varepsilon \sim N(0_n, \Sigma_\varepsilon) \quad (3)$$

- `fixed` spezifiert die Form der Fixed-Effects-Modellkomponenten als R Formel.
- `data` spezifiziert die Daten der Variablen des Modells als R Dataframe.
- `random` spezifiert die Form der Random-Effects-Modellkomponenten als erweiterte R Formel.

# Analyse einer Multizentrenstudie mit `nlme`

## Evaluation eines einfachen linearen Regressionsmodells

### Strukturelle Form

- Wir vernachlässigen zunächst die Gruppenstruktur (Hochschulambulanzvariation) des Datensatzes.
- Wir modellieren den $i$ten `BDI`-Wert $y_i$ als Funktion des $i$ten `ATS`-Wertes $x_i$ als

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \text{ mit } \varepsilon_i \sim N(0, \sigma^2) \text{ für } i = 1, ..., n \tag{4}$$

### Designmatrixform

- In Matrixform betrachten wir also das Modell

$$y = X\beta + \varepsilon \Leftrightarrow \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_n \end{pmatrix} \text{ mit } \varepsilon \sim N(0_{20}, \sigma^2 I_{20}) \tag{5}$$

Wir fassen den Datensatz also mithilfe folgender Parameterschätzer zusammen:

| Parameterschätzer | Interpretation |
| --- | --- |
| $\hat{\beta}_0$ | Interzeptparameterschätzer/Symptomreduktion bei 0 Therapiestunden |
| $\hat{\beta}_1$ | Steigungsparameterschätzer/Symptomreduktion pro 1 Therapiestunde |
| $\hat{\sigma}^2$ | Varianzparameterschätzer/Residuelle, durch das Modell unerklärte, Varianz |

# Analyse einer Multizentrenstudie mit nlme

## Evaluation eines einfachen linearen Regressionsmodells

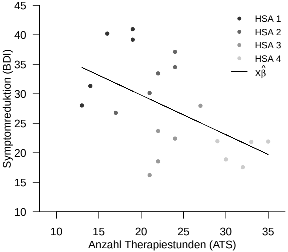
```
D           = read.csv("./6_Daten/BDI.csv")            # Einlesen des Datensatzes
M           = lm(BDI ~ ATS, D)                          # Einfache lineare Regression
S           = summary(M)                                # Modellschätzungsresultate
X           = model.matrix(M)                           # Designmatrix
beta_hat    = as.matrix(M$coefficients)                 # Betaparameterschätzer
sigsqr_hat  = S$sigma**2                                # Varianzparameterschätzer
```

Designmatrix

```
   (Intercept) ATS
1            1  19
2            1  13
3            1  14
4            1  16
5            1  19
6            1  17
7            1  24
8            1  24
9            1  22
10           1  21
11           1  21
12           1  22
13           1  22
14           1  27
15           1  24
16           1  33
17           1  30
18           1  32
19           1  35
20           1  29
attr(,"assign")
[1] 0 1
```

# Analyse einer Multizentrenstudie mit `nlme`

Evaluation eines einfachen linearen Regressionsmodells




```
Betaparameterschätzer    :  43.2 -0.67
Varianzparameterschätzer :  47.88
```

- Keine Berücksichtigung der Gruppenstruktur (Hochschulambulanzvariation)
- Steigungsparameterschätzer $\hat{\beta}_1$ legt Abnahme der Symptomoreduktion mit Therapiedauer nahe
- Dateninspektion zeigt aber einen Anstieg der Symptomreduktion mit Therapiestunden für jede HSA
- Das Modell ist für den betrachteten Anwendungsfall sicherlich nicht ideal

# Analyse einer Multizentrenstudie mit `nlme`

## Evaluation eines einfaktoriellen Varianzanalysemodells

### Strukturelle Form

- Wir vernachlässigen den Effekt der Therapiedauer und betrachten lediglich den Effekt der Hochschulambulanz
- Wir modellieren den $j$ten Datenwert in der $i$ten Gruppe als

$$y_{1j} = \mu_0 + \varepsilon_{1j} \text{ und } y_{ij} = \mu_0 + \alpha_i + \varepsilon_{ij} \text{ für } i = 2,3,4 \text{ mit } \varepsilon_{ij} \sim N(0,\sigma^2) \text{ für } i = 1,,...,m, j = 1,...,n_i$$

### Designmatrixform

- In Designmatrixform betrachten wir also das Modell

$$y = X\beta + \varepsilon \Leftrightarrow \begin{pmatrix} y_{11} \\ \vdots \\ y_{15} \\ y_{21} \\ \vdots \\ y_{25} \\ y_{31} \\ \vdots \\ y_{35} \\ y_{41} \\ \vdots \\ y_{45} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \mu_0 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{pmatrix} + \begin{pmatrix} \varepsilon_{11} \\ \vdots \\ \varepsilon_{15} \\ \varepsilon_{21} \\ \vdots \\ \varepsilon_{25} \\ \varepsilon_{31} \\ \vdots \\ \varepsilon_{35} \\ \varepsilon_{41} \\ \vdots \\ \varepsilon_{45} \end{pmatrix} \text{ mit } \varepsilon \sim N(0_{20}, \sigma^2 I_{20}) \tag{6}$$

# Analyse einer Multizentrenstudie mit `nlme`

## Evaluation eines einfaktoriellen Varianzanalysemodells

Wir fassen den Datensatz also mithilfe folgender Parameterschätzer zusammen:

| Parameterschätzer | Interpretation |
|---|---|
| $\hat{\mu}_0$ | Erwartungswertschätzer für die Symptomreduktion in der Referenzgruppe HSA 1 |
| $\hat{\alpha}_2$ | Schätzer für die Erwartungswertdifferenz für die Symptomreduktion zwischen HSA 2 und HSA 1 |
| $\hat{\alpha}_3$ | Schätzer für die Erwartungswertdifferenz für die Symptomreduktion zwischen HSA 3 und HSA 1 |
| $\hat{\alpha}_4$ | Schätzer für die Erwartungswertdifferenz für die Symptomreduktion zwischen HSA 4 und HSA 1 |
| $\hat{\sigma}^2$ | Varianzparameterschätzer/Residuelle, durch das Modell unerklärte, Varianz |

```r
D           = read.csv("./6_Daten/BDI.csv")           # Einlesen des Datensatzes
D$HSA       = as.factor(D$HSA)                        # Umwandlung numerischer Werte in R factor
M           = lm(BDI ~ HSA, D)                        # Einfache lineare Regression
S           = summary(M)                              # Modellschätzungsresultate
X           = model.matrix(M)                         # Designmatrix
beta_hat    = as.matrix(M$coefficients)               # Betaparameterschätzer
sigsqr_hat  = S$sigma**2                              # Varianzparameterschätzer
```

# Analyse einer Multizentrenstudie mit nlme

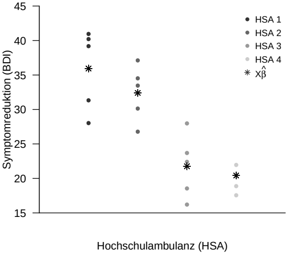
Evaluation eines einfaktoriellen Varianzanalysemodells

Designmatrix

```
   (Intercept) HSA2 HSA3 HSA4
1            1    0    0    0
2            1    0    0    0
3            1    0    0    0
4            1    0    0    0
5            1    0    0    0
6            1    1    0    0
7            1    1    0    0
8            1    1    0    0
9            1    1    0    0
10           1    1    0    0
11           1    0    1    0
12           1    0    1    0
13           1    0    1    0
14           1    0    1    0
15           1    0    1    0
16           1    0    0    1
17           1    0    0    1
18           1    0    0    1
19           1    0    0    1
20           1    0    0    1
attr(,"assign")
[1] 0 1 1 1
attr(,"contrasts")
attr(,"contrasts")$HSA
[1] "contr.treatment"
```

# Analyse einer Multizentrenstudie mit `nlme`

Evaluation eines einfaktoriellen Varianzanalysemodells




```
Betaparameterschätzer    :  35.94 -3.54 -14.17 -15.51
Varianzparameterschätzer :  18.97
```

- Keine Berücksichtigung der Anzahl an Therapiestunden
- Das Modell ist vor dem Hintergrund der Frage nach dem Therapiestundeneffekt sicherlich nicht ideal
- HSA 2, HSA 3 und HSA 4 haben einer geringere Symptomereduktionserwartung als HSA 1
- Dateninspektion zeigt unterschiedliche mittlere Therapiestundenanzahlen pro HSA

# Analyse einer Multizentrenstudie mit `nlme`

## Evaluation gruppenspezifischer einfacher linearen Regressionsmodelle

### Strukturelle Form

- Wir betrachten jede der Gruppen (Hochschulambulanzen) einzeln.
- Wir modellieren den $j$ten `BDI`-Wert der $i$ten Gruppe $y_{i_j}$ als Funktion des $i,j$ten `ATS`-Wertes $x_{i_j}$

$$y_{i_j} = \beta_{i_0} + \beta_{i_1} x_{i_j} + \varepsilon_{i_j} \text{ mit } \varepsilon_{i_j} \sim N(0, \sigma_i^2) \text{ für } i = 1, ..., m, j = 1, ..., n_i. \quad (7)$$

- In Matrixform betrachten wir also für $i = 1, ..., k$

$$y_i = X_i \beta_i + \varepsilon_i \Leftrightarrow \begin{pmatrix} y_{i_1} \\ \vdots \\ y_{i_{n_i}} \end{pmatrix} = \begin{pmatrix} 1 & x_{i_1} \\ \vdots & \vdots \\ 1 & x_{i_{n_i}} \end{pmatrix} \begin{pmatrix} \beta_{i_0} \\ \beta_{i_1} \end{pmatrix} + \begin{pmatrix} \varepsilon_{i_1} \\ \vdots \\ \varepsilon_{i_{n_i}} \end{pmatrix} \text{ mit } \varepsilon_i \sim N(0_{n_i}, \sigma^2 I_{n_i}) \quad (8)$$

Wir fassen den Datensatz also mithilfe folgender Parameterschätzer für jede Gruppe $i = 1, ..., m$ zusammen:

| Parameterschätzer | Interpretation |
|---|---|
| $\hat{\beta}_{i_0}$ | Interzeptparameterschätzer/Symptomreduktion bei 0 Therapiestunden in HSA $i$ |
| $\hat{\beta}_{i_1}$ | Steigungsparameterschätzer/Symptomreduktion pro 1 Therapiestunde in HSA $i$ |
| $\hat{\sigma}_i^2$ | Varianzparameterschätzer/Residuelle, durch das Modell unerklärte, Varianz in HSA $i$ |

# Analyse einer Multizentrenstudie mit `nlme`

Evaluation gruppenspezifischer einfacher linearen Regressionsmodelle

```
library(nlme)
D           = read.csv("./6_Daten/BDI.csv")              # Einlesen des Datensatzes
M           = lmList(BDI ~ ATS | HSA, D)                 # gruppenspezifische ELRs
S           = list()                                     # Liste für Modelschätzungsresultate
X           = list()                                     # Liste für Designmatrizen
beta_hat    = list()                                     # Liste für Betaparameterschätzer
sigsqr_hat  = list()                                     # Liste für Varianzparameterschätzer
for(i in 1:length(M)){                                   # Gruppeniterationen
    S[[i]]            = summary(M[[i]])                  # Modellschätzungsresultate
    X[[i]]            = model.matrix(M[[i]])             # Designmatrix
    beta_hat[[i]]     = as.matrix(M[[i]]$coefficients)   # Betaparameterschätzer
    sigsqr_hat[[i]]   = S[[i]]$sigma**2}                 # Varianzparameterschätzer
```

`BDI ~ ATS | HSA` ist eine erweiterte R Formel, die `BDI` und `ATS` anhand der Level des Faktors `HSA` partitioniert!

# Analyse einer Multizentrenstudie mit nlme

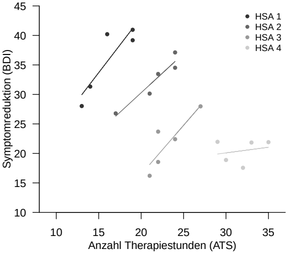
Evaluation gruppenspezifischer einfacher linearen Regressionsmodelle

Designmatrizen

```
  (Intercept) ATS
1           1  19
2           1  13
3           1  14
4           1  16
5           1  19
attr(,"assign")
[1] 0 1
   (Intercept) ATS
6            1  17
7            1  24
8            1  24
9            1  22
10           1  21
attr(,"assign")
[1] 0 1
   (Intercept) ATS
11           1  21
12           1  22
13           1  22
14           1  27
15           1  24
attr(,"assign")
[1] 0 1
   (Intercept) ATS
16           1  33
17           1  30
18           1  32
19           1  35
20           1  29
attr(,"assign")
[1] 0 1
```

# Analyse einer Multizentrenstudie mit `nlme`

Evaluation gruppenspezifischer einfacher linearen Regressionsmodelle




```
HSA 1 Betaparameterschätzer    :  5.57 1.87  Varianzparameterschätzer:  9.84
HSA 2 Betaparameterschätzer    :  3.75 1.33  Varianzparameterschätzer:  2.05
HSA 3 Betaparameterschätzer    :  -16.8 1.66  Varianzparameterschätzer:  6.98
HSA 4 Betaparameterschätzer    :  14.35 0.19  Varianzparameterschätzer:  5.45
```

- Es wird deutlich, dass die Symptomreduktion in jeder HSA mit steigender Therapiestundenanzahl zunimmt.
- Dies entspricht der Intuition bei HSA-spezifischer Dateninspektion.
- Es wird allerdings kein HSA-übergreifender Therapiestundenanzahleffekt bestimmt.
- Es wird auch kein Maß für die Varianz der Effekt zwischen den Hochschulambulanzen bestimmt.

# Analyse einer Multizentrenstudie mit `nlme`

## Evaluation eines Einfache-Lineare-Regression-LMMs mit zufälligem Interzeptparameter

Strukturelle Form

- Wir modellieren den $j$ten Datenwert in der $i$ten Gruppe mit $j = 1, ..., 5$ und $i = 1, ..., 4$ als

$$y_{ij} = \beta_0 + \beta_1 x_{ij} + \beta_{0i} + \varepsilon_{ij} \text{ mit } \beta_{0i} \sim N(0, \sigma^2_{\beta_r}) \text{ mit } \varepsilon_{ij} \sim N(0, \sigma^2) \tag{9}$$

Designmatrixform

$$y = X_f \beta_f + X_r \beta_r + \varepsilon \Leftrightarrow \begin{pmatrix} y_{11} \\ \vdots \\ y_{15} \\ y_{21} \\ \vdots \\ y_{25} \\ y_{31} \\ \vdots \\ y_{35} \\ y_{41} \\ \vdots \\ y_{45} \end{pmatrix} = \begin{pmatrix} 1 & x_{11} \\ \vdots & \vdots \\ 1 & x_{15} \\ 1 & x_{21} \\ \vdots & \vdots \\ 1 & x_{25} \\ 1 & x_{31} \\ \vdots & \vdots \\ 1 & x_{35} \\ 1 & x_{41} \\ \vdots & \vdots \\ 1 & x_{45} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} 1 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \beta_{01} \\ \beta_{02} \\ \beta_{03} \\ \beta_{04} \end{pmatrix} + \begin{pmatrix} \varepsilon_{11} \\ \vdots \\ \varepsilon_{15} \\ \varepsilon_{21} \\ \vdots \\ \varepsilon_{25} \\ \varepsilon_{31} \\ \vdots \\ \varepsilon_{35} \\ \varepsilon_{41} \\ \vdots \\ \varepsilon_{45} \end{pmatrix} \tag{10}$$

mit

$$\beta_r \sim N(0_4, \sigma^2_{\beta_r} I_4) \text{ und } \varepsilon \sim N(0_{20}, \sigma^2 I_{20}) \tag{11}$$

# Analyse einer Multizentrenstudie mit `nlme`

## Evaluation eines Einfache-Lineare-Regression-LMMs mit zufälligem Interzeptparameter

Wir fassen den Datensatz als für $i = 1, ..., 4$ mithilfe folgender Parameterschätzer zusammen:

| Parameterschätzer | Interpretation |
|---|---|
| $\hat{\beta}_0$ | FFX-Interzept-Schätzer/HSA-übergreifende Symptomreduktion bei 0 Therapiestunden |
| $\hat{\beta}_1$ | FFX-Steigungs-Schätzer/HSA-übergreifende Symptomreduktion pro 1 Therapiestunde |
| $\hat{\beta}_{0i}$ | RFX-Interzept-Schätzer/HSA $i$-spezifische Symptomreduktion bei 0 Therapiestunden |
| $\hat{\sigma}^2_{\beta_r}$ | RFX-Varianz-Schätzer/HSA-übergreifende Interzeptvarianz |
| $\hat{\sigma}^2$ | Datenvarianz-Schätzer/Residuelle, durch das Modell unerklärte, Varianz |

```
library(nlme)                                                          # R Paket
D               = read.csv("./6_Daten/BDI.csv")                        # Einlesen des Datensatzes
D$HSA           = as.factor(D$HSA)                                     # Umwandlung numerischer Werte in R factor
M               = lme(BDI ~ ATS, D, random = ~ 1 | HSA)                # LMM vom Einfache Lineare Regressionstyp
X_f             = model.matrix(M, D)                                   # Fixed-Effects-Designmatrix
X_r             = model.matrix(~ M$groups[[1]] - 1)                    # Random-Effects-Designmatrix
beta_f_hat      = M$coefficients$fixed                                 # Fixed-Effects-Parameterschätzer
beta_r_hat      = M$coefficients$random$HSA                            # Random-Effects-Parameterschätzer
s_eps_hat       = M$sigma**2                                           # Datenvarianzkomponente
s_beta_r_hat    = diag(getVarCov(M))                                   # Random-Effects-Varianzkomponente
```

`random = ~ 1 | HSA` enkodiert einen gruppenspezifischen zufälligen Interzeptparameter

Evaluation eines Einfache-Lineare-Regression-LMMs mit zufälligem Interzeptparameter

Fixed-Effects-Designmatrix

```
   (Intercept) ATS
1            1  19
2            1  13
3            1  14
4            1  16
5            1  19
6            1  17
7            1  24
8            1  24
9            1  22
10           1  21
11           1  21
12           1  22
13           1  22
14           1  27
15           1  24
16           1  33
17           1  30
18           1  32
19           1  35
20           1  29
attr(,"assign")
[1] 0 1
```

# Analyse einer Multizentrenstudie mit nlme

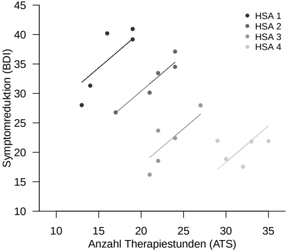
Evaluation eines Einfache-Lineare-Regression-LMMs mit zufälligem Interzeptparameter

Random-Effects-Designmatrix

```
   M$groups[[1]]1 M$groups[[1]]2 M$groups[[1]]3 M$groups[[1]]4
1               1              0              0              0
2               1              0              0              0
3               1              0              0              0
4               1              0              0              0
5               1              0              0              0
6               0              1              0              0
7               0              1              0              0
8               0              1              0              0
9               0              1              0              0
10              0              1              0              0
11              0              0              1              0
12              0              0              1              0
13              0              0              1              0
14              0              0              1              0
15              0              0              1              0
16              0              0              0              1
17              0              0              0              1
18              0              0              0              1
19              0              0              0              1
20              0              0              0              1
attr(,"assign")
[1] 1 1 1 1
attr(,"contrasts")
attr(,"contrasts")$`M$groups[[1]]`
[1] "contr.treatment"
```

# Analyse einer Multizentrenstudie mit `nlme`

Evaluation eines Einfache-Lineare-Regression-LMMs mit zufälligem Interzeptparameter




```
Fixed-Effects-Parameterschätzer  :  -0.87 1.23
Residuelle Varianzschätzer       :  7.65
Random-Effects-Varianzschätzer   :  228.4

[1] "Random-Effects-Parameterschätzer"

  (Intercept)
1   16.792256
2    6.689060
3   -5.828815
4  -17.652501
```

- Es wird deutlich, dass die Symptomreduktion über HSAs mit steigender Therapiestundenanzahl zunimmt.
- Die Annahme, dass die Symptomreduktionsrate in jeder HSA gleich ist, mag etwas stark sein (vgl. HSA 4).

### Evaluation eines ELR-LMMs mit zufälligen Interzept- und Steigungsparametern

Strukturelle Form

- Wir modellieren den $j$ten Datenwert in der $i$ten Gruppe mit $j = 1, ..., 5$ und $i = 1, ..., 4$ als

$$y_{ij} = \beta_0 + \beta_1 x_{ij} + \beta_{0i} + \beta_{1i} x_{ij} + \varepsilon_{ij} \text{ mit } \beta_{ki} \sim N(0, \sigma^2_{\beta_r}) \text{ für } k = 0, 1 \text{ und } \varepsilon_{ij} \sim N(0, \sigma^2) \qquad (12)$$

Designmatrixform

$$y = X_f \beta_f + X_r \beta_r + \varepsilon \Leftrightarrow \qquad (13)$$

$$\begin{pmatrix} y_{11} \\ \vdots \\ y_{15} \\ y_{21} \\ \vdots \\ y_{25} \\ y_{31} \\ \vdots \\ y_{35} \\ y_{41} \\ \vdots \\ y_{45} \end{pmatrix} = \begin{pmatrix} 1 & x_{11} \\ \vdots & \vdots \\ 1 & x_{15} \\ 1 & x_{21} \\ \vdots & \vdots \\ 1 & x_{25} \\ 1 & x_{31} \\ \vdots & \vdots \\ 1 & x_{35} \\ 1 & x_{41} \\ \vdots & \vdots \\ 1 & x_{45} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \end{pmatrix} + \begin{pmatrix} 1 & x_{11} & 0 & 0 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{15} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & x_{21} & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 1 & x_{25} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & x_{31} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 1 & x_{35} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & x_{41} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & x_{45} \end{pmatrix} \begin{pmatrix} \beta_{01} \\ \beta_{11} \\ \beta_{02} \\ \beta_{12} \\ \beta_{03} \\ \beta_{13} \\ \beta_{04} \\ \beta_{14} \end{pmatrix} + \begin{pmatrix} \varepsilon_{11} \\ \vdots \\ \varepsilon_{15} \\ \varepsilon_{21} \\ \vdots \\ \varepsilon_{25} \\ \varepsilon_{31} \\ \vdots \\ \varepsilon_{35} \\ \varepsilon_{41} \\ \vdots \\ \varepsilon_{45} \end{pmatrix} \qquad (14)$$

mit

$$\beta_r \sim N(0_8, \sigma^2_{\beta_r} I_8) \text{ und } \varepsilon \sim N(0_{20}, \sigma^2 I_{20}) \qquad (15)$$

## Evaluation eines ELR-LMMs mit zufälligen Interzept- und Steigungsparametern

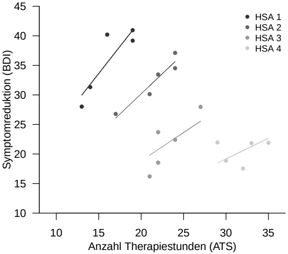
Wir fassen den Datensatz als für $i = 1, ..., 4$ mithilfe folgender Parameterschätzer zusammen:

| Parameterschätzer | Interpretation |
| --- | --- |
| $\hat{\beta}_0$ | FFX-Interzept-Schätzer/HSA-übergreifende Symptomreduktion bei 0 Therapiestunden |
| $\hat{\beta}_1$ | FFX-Steigungs-Schätzer/HSA-übergreifende Symptomreduktion pro 1 Therapiestunde |
| $\hat{\beta}_{0i}$ | RFX-Interzept-Schätzer/HSA $i$-spezifische Symptomreduktion bei 0 Therapiestunden |
| $\hat{\beta}_{1i}$ | RFX-Steigungs-Schätzer/HSA $i$-spezifische Symptomreduktion pro 1 Therapiestunde |
| $\hat{\sigma}^2_{\beta_r}$ | RFX-Varianz-Schätzer/HSA-übergreifende Interzept- und Steigungsvarianz |
| $\hat{\sigma}^2$ | Datenvarianz-Schätzer/Residuelle, durch das Modell unerklärte, Varianz |

```r
library(nlme)                                                          # R Paket
ctrl             = lmeControl(opt = "optim", maxIter = 100)            # Anzahl an ReML Iterationen
D                = read.csv("./6_Daten/BDI.csv")                       # Einlesen des Datensatzes
D$HSA            = as.factor(D$HSA)                                    # Umwandlung numerischer Werte in R factor
M                = lme(BDI ~ ATS, D, random = ~ ATS | HSA, control = ctrl) # LMM vom Einfache Lineare Regressionstyp
X_f              = model.matrix(M,D)                                   # Fixed-Effects-Designmatrix
beta_f_hat       = M$coefficients$fixed                                # Fixed-Effects-Parameterschätzer
beta_r_hat       = M$coefficients$random$HSA                           # Random-Effects-Parameterschätzer
sigsqr_eps_hat   = M$sigma**2                                          # Datenvarianzkomponente
```

# Analyse einer Multizentrenstudie mit `nlme`

Evaluation eines ELR-LMMs mit zufälligen Interzept- und Steigungsparametern




```
Fixed-Effects-Parameterschätzer  :  1.74 1.21
Residuelle Varianzschätzer       :  7.65
Random-Effects-Varianzschätzer   :  228.4

[1] "Random-Effects-Parameterschätzer"

  (Intercept)        ATS
1    4.438574  0.6167610
2    1.116603  0.1531704
3   -2.062339 -0.2545769
4   -3.492838 -0.5153544
```

- Es wird deutlich, dass die Symptomreduktion über HSAs mit steigender Therapiestundenanzahl zunimmt.
- Die Symptomreduktionsrate varriieren nicht unabhängig über HSAs, sondern durch andere HSAs informiert.

# Appendix

## R Formeln

Modelle der Form $y = X\beta + \varepsilon$ mit $\varepsilon \sim N(0_n, \sigma^2 I_n)$ werden in R symbolisch durch `formulas` dargestellt

```
Daten ~ Term 1 + Term 2 + ... + Term k
```

- Der `~` Operator trennt die linke Seite und rechte Seite einer `formula`
- `Daten` wird zur Identifikation der abhängigen Variable $y$ genutzt
- `Term 1 + Term 2 + ... + Term k` dient der Spezifikation der Spalten der Designmatrix $X$
- Die `formulas` Syntax geht zurück auf Wilkinson and Rogers (1973) und Chambers and Hastie (1992)

Terme können numerische Prädiktoren oder kategoriale Faktoren (R `factors`) sein

Die `formula` Syntax ist symbolisch, zur Laufzeit müssen die Prädiktoren und Faktoren nicht spezifiziert sein

Essentielle Operatoren in `formulas` sind `+` und `:`

- `+` fügt der Designmatrix Prädiktoren hinzu, `:` dient der Spezifikation von Interaktionen

Nichtessentielle Operatoren in `formulas` sind `*`, `/`, `%in%`, `-` und `^`

- Für zwei Faktoren `f1` und `f2` gilt beispielsweise `f1*f2 = f1 + f2 + f1:f2`

## R Formeln

Beispiele mit `x1`, `x2` als numerische Vektoren und `f1`, `f2` als R Faktoren

```
formula(y ~ x1)          # Spezifikation einer einfachen linearen Regresssion mithilfe der formula() Funktion
y ~ x1                   # Aufruf der formula() Funktion ist aber nicht nötig, R erkennt formulas auch so
y ~ 1 + x1               # Explizite Interzeptdefinition bei einfacher linearer Regression, nicht nötig
y ~ 0 + x1               # Verzicht auf Interzeptdefinition bei einfacher lineare Regression
y ~ 1 + x1 + x2          # Multiple Regression mit zwei Regressoren und expliziter Interzeptdefinition
y ~ f1                   # Einfaktorielles ANOVA Design
y ~ f1 + f2              # Additives zweifaktorielles ANOVA Design
y ~ f1 + f2 + f1:f2      # Zweifaktorielles ANOVA Design mit Interaktion
y ~ f1 + x1              # Additives einfaktorielles ANCOVA Design mit einer Kovariate
y ~ f1 + x1 + f1:x1      # Einfaktorielles ANCOVA Design mit einer Kovariate und Interaktion
```

Wir betrachten im Folgenden die durch diese `formulas` erzeugten Designmatrizen $X \in \mathbb{R}^{n \times p}$.

## R Formeln

### Einfache Lineare Regression

```
n   = 12                                          # Anzahl Datenpunkte
x1  = 1:n                                         # Regressor
y   = rnorm(n)                                    # Beispieldaten
D   = data.frame(y = y, x1 = x1)                  # Dataframe
M   = lm(y ~ x1, D)                               # Modellevaluation
X   = model.matrix(M)                             # Designmatrix
```

```
   (Intercept) x1
1            1  1
2            1  2
3            1  3
4            1  4
5            1  5
6            1  6
7            1  7
8            1  8
9            1  9
10           1 10
11           1 11
12           1 12
attr(,"assign")
[1] 0 1
```

## R Formeln

Einfache Lineare Regression mit expliziter Interzeptdefinition

```
n   = 12                                  # Anzahl Datenpunkte
x1  = 1:n                                 # Regressor
y   = rnorm(n)                            # Beispieldaten
D   = data.frame(y = y, x1 = x1)          # Dataframe
M   = lm(y ~ 1 + x1, D)                   # Modellevaluation
X   = model.matrix(M)                     # Designmatrix
```

```
   (Intercept) x1
1            1  1
2            1  2
3            1  3
4            1  4
5            1  5
6            1  6
7            1  7
8            1  8
9            1  9
10           1 10
11           1 11
12           1 12
attr(,"assign")
[1] 0 1
```

## R Formeln

Einfache Lineare Regression mit Verzicht auf Interzeptdefinition

```
n   = 12                                                       # Anzahl Datenpunkte
x1  = 1:n                                                      # Regressor
y   = rnorm(n)                                                 # Beispieldaten
D   = data.frame(y = y, x1 = x1)                               # Dataframe
M   = lm(y ~ 0 + x1, D)                                        # Modellevaluation
X   = model.matrix(M)                                          # Designmatrix
```

```
   x1
1   1
2   2
3   3
4   4
5   5
6   6
7   7
8   8
9   9
10 10
11 11
12 12
attr(,"assign")
[1] 1
```

## R Formeln

Multiple Regression mit zwei Regressoren und expliziter Interzeptdefinition

```
n   = 12                                            # Anzahl Datenpunkte
x1  = 1:n                                           # Regressor 1
x2  = (1:n)+5                                       # Regressor 2
y   = rnorm(n)                                      # Beispieldaten
D   = data.frame(y = y, x1 = x1, x2 = x2)           # Dataframe
M   = lm(y ~ 1 + x1 + x2, D)                        # Modellevaluation
X   = model.matrix(M)                               # Designmatrix
```

```
   (Intercept) x1 x2
1            1  1  6
2            1  2  7
3            1  3  8
4            1  4  9
5            1  5 10
6            1  6 11
7            1  7 12
8            1  8 13
9            1  9 14
10           1 10 15
11           1 11 16
12           1 12 17
attr(,"assign")
[1] 0 1 2
```

## R Formeln

Einfaktorielles ANOVA Design mit drei Faktorleveln

```
n   = 12                                        # Anzahl Datenpunkte
f1  = as.factor(c(1,1,1,1,2,2,2,2,3,3,3,3))    # Faktorlevel der Datenpunkte
y   = rnorm(n)                                  # Beispieldaten
D   = data.frame(y = y, f1 = f1)                # Dataframe
M   = lm(y ~ f1, D)                             # Modellevaluation
X   = model.matrix(M)                           # Designmatrix
```

```
   (Intercept) f12 f13
1            1   0   0
2            1   0   0
3            1   0   0
4            1   0   0
5            1   1   0
6            1   1   0
7            1   1   0
8            1   1   0
9            1   0   1
10           1   0   1
11           1   0   1
12           1   0   1
attr(,"assign")
[1] 0 1 1
attr(,"contrasts")
attr(,"contrasts")$f1
[1] "contr.treatment"
```

# Appendix

## R Formeln

Zweifaktorielles additives ANOVA Design mit jeweils zwei Faktorleveln

```
n   = 12                                                # Anzahl Datenpunkte
f1  = as.factor(c(1,1,1,1,1,1,2,2,2,2,2,2))             # Faktor-1-Level der Datenpunkte
f2  = as.factor(c(1,1,1,2,2,2,1,1,1,2,2,2))             # Faktor-2-Level der Datenpunkte
y   = rnorm(n)                                          # Beispieldaten
D   = data.frame(y = y, f1 = f1, f2 = f2)               # Dataframe
M   = lm(y ~ f1 + f2, D)                                # Modellevaluation
X   = model.matrix(M)                                   # Designmatrix
```

```
   (Intercept) f12 f22
1            1   0   0
2            1   0   0
3            1   0   0
4            1   0   1
5            1   0   1
6            1   0   1
7            1   1   0
8            1   1   0
9            1   1   0
10           1   1   1
11           1   1   1
12           1   1   1
attr(,"assign")
[1] 0 1 2
attr(,"contrasts")
attr(,"contrasts")$f1
[1] "contr.treatment"

attr(,"contrasts")$f2
[1] "contr.treatment"
```

# Appendix

## R Formeln

### Zweifaktorielles additives ANOVA Design mit Interaktion

```
n   = 12                                          # Anzahl Datenpunkte
f1  = as.factor(c(1,1,1,1,1,1,2,2,2,2,2,2))       # Faktor-1-Level der Datenpunkte
f2  = as.factor(c(1,1,1,2,2,2,1,1,1,2,2,2))       # Faktor-2-Level der Datenpunkte
y   = rnorm(n)                                    # Beispieldaten
D   = data.frame(y = y, f1 = f1, f2 = f2)         # Dataframe
M   = lm(y ~ f1 + f2 + f1:f2, D)                  # Modellevaluation
X   = model.matrix(M)                             # Designmatrix
```

```
   (Intercept) f12 f22 f12:f22
1            1   0   0       0
2            1   0   0       0
3            1   0   0       0
4            1   0   1       0
5            1   0   1       0
6            1   0   1       0
7            1   1   0       0
8            1   1   0       0
9            1   1   0       0
10           1   1   1       1
11           1   1   1       1
12           1   1   1       1
attr(,"assign")
[1] 0 1 2 3
attr(,"contrasts")
attr(,"contrasts")$f1
[1] "contr.treatment"

attr(,"contrasts")$f2
[1] "contr.treatment"
```

## R Formeln

Additives einfaktorielles ANCOVA Design mit einer Kovariate

```
n   = 12                                         # Anzahl Datenpunkte
f1  = as.factor(c(1,1,1,1,1,1,2,2,2,2,2,2))      # Faktorlevel der Datenpunkte
x1  = 1:n                                        # Kovariatenwerte der Datenpunkte
y   = rnorm(n)                                   # Beispieldaten
D   = data.frame(y = y, f1 = f1, x1 = x1)        # Dataframe
M   = lm(y ~ f1 + x1, D)                         # Modellevaluation
X   = model.matrix(M)                            # Designmatrix
```

```
   (Intercept) f12 x1
1            1   0  1
2            1   0  2
3            1   0  3
4            1   0  4
5            1   0  5
6            1   0  6
7            1   1  7
8            1   1  8
9            1   1  9
10           1   1 10
11           1   1 11
12           1   1 12
attr(,"assign")
[1] 0 1 2
attr(,"contrasts")
attr(,"contrasts")$f1
[1] "contr.treatment"
```

## R Formeln

### Einfaktorielles ANCOVA Design mit einer Kovariate und Interaktion

```r
n   = 12                                    # Anzahl Datenpunkte
f1  = as.factor(c(1,1,1,1,1,1,2,2,2,2,2,2)) # Faktorlevel der Datenpunkte
x1  = 1:n                                   # Kovariatenwerte der Datenpunkte
y   = rnorm(n)                              # Beispieldaten
D   = data.frame(y = y, f1 = f1, x1 = x1)  # Dataframe
M   = lm(y ~ f1 + x1 + f1:x1, D)           # Modellevaluation
X   = model.matrix(M)                       # Designmatrix
```

```
   (Intercept) f12 x1 f12:x1
1            1   0  1      0
2            1   0  2      0
3            1   0  3      0
4            1   0  4      0
5            1   0  5      0
6            1   0  6      0
7            1   1  7      7
8            1   1  8      8
9            1   1  9      9
10           1   1 10     10
11           1   1 11     11
12           1   1 12     12
attr(,"assign")
[1] 0 1 2 3
attr(,"contrasts")
attr(,"contrasts")$f1
[1] "contr.treatment"
```

# Referenzen I


Chambers, John M., and Trevor Hastie, eds. 1992. *Statistical Models in S*. Wadsworth & Brooks/Cole Computer Science Series. Pacific Grove, Calif: Wadsworth & Brooks/Cole Advanced Books & Software.

Pinheiro, José C., and Douglas M. Bates. 2000. *Mixed-Effects Models in S and S-PLUS*. Statistics and Computing. New York: Springer.

Wilkinson, G. N., and C. E. Rogers. 1973. "Symbolic Description of Factorial Models for Analysis of Variance." *Applied Statistics* 22 (3): 392. https://doi.org/10.2307/2346786.