

# Gemischte Modelle

Vitaly Belik

Institut für Veterinär-Epidemiologie und Biometrie, FU Berlin

05/01/2020

- ▶ Beispiel. Wann braucht man solche Modelle?

# Beispiel Kosmetische Chirurgie (nach A. Field)

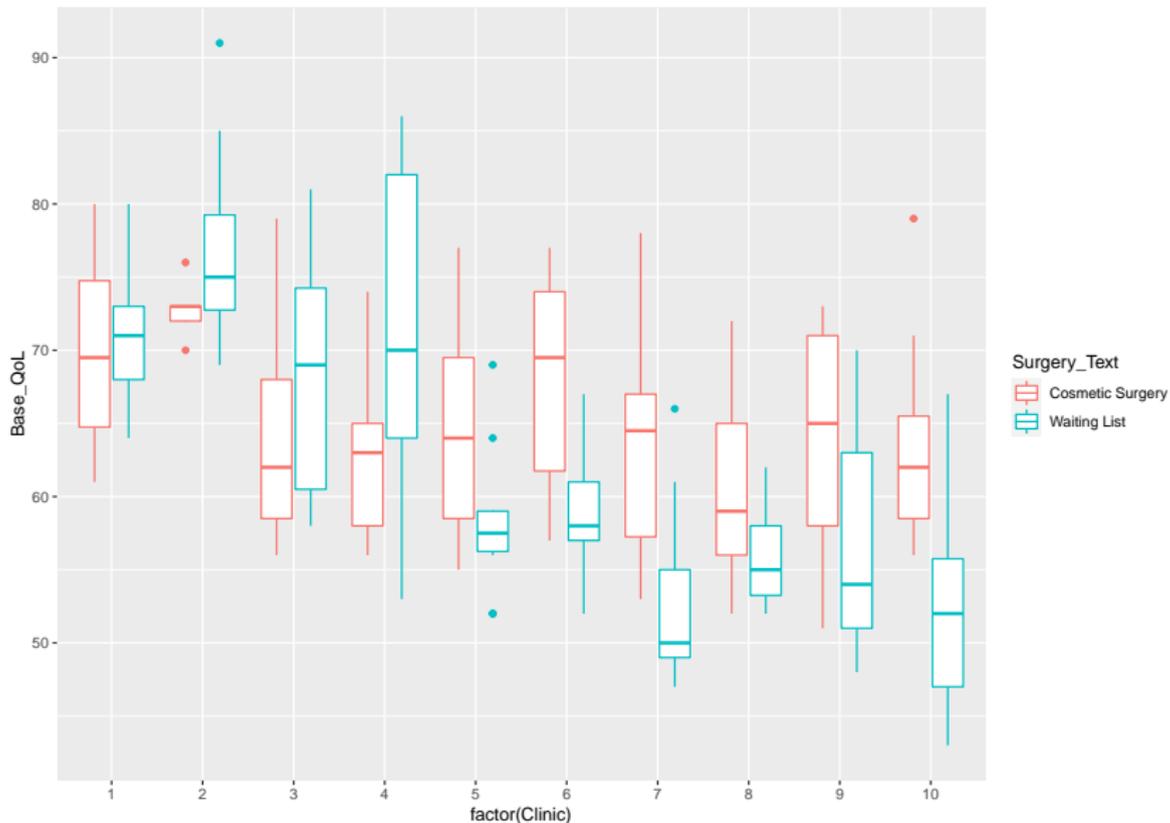
```
d <- read.table("Cosmetic Surgery.dat", sep = "\t", header = TRUE)
head(d)
```

```
##      particnu Post_QoL Base_QoL Clinic Surgery Reason Age Gender BDI Surgery_Text
## 1          1    71.3      73      1      0      0 31      0 12 Waiting List
## 2          2    77.0      74      1      0      0 32      0 16 Waiting List
## 3          3    73.0      80      1      0      0 33      0 13 Waiting List
## 4          4    68.9      76      1      0      0 59      1 11 Waiting List
## 5          5    69.0      71      1      0      0 61      1 11 Waiting List
## 6          6    68.5      72      1      0      1 32      0 10 Waiting List
##      Reason_Text Gender_Text
## 1 Change Appearance      Female
## 2 Change Appearance      Female
## 3 Change Appearance      Female
## 4 Change Appearance      Male
## 5 Change Appearance      Male
## 6 Physical reason      Female
```

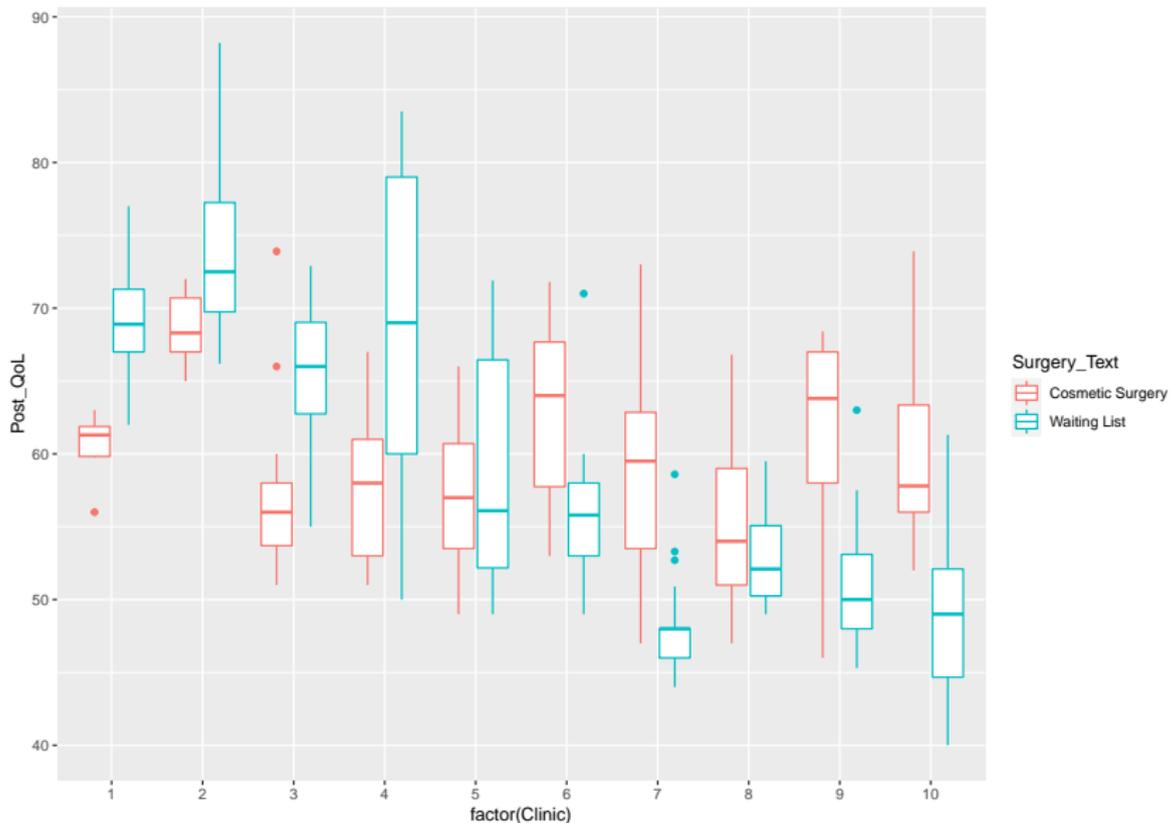
# Beschreibung des Datensatzes

- ▶ Post\_QoL: This is a measure of quality of life after the cosmetic surgery. This is our outcome variable.
- ▶ Base\_QoL: We need to adjust our outcome for quality of life before the surgery.
- ▶ Surgery: This variable is a dummy variable that specifies whether the person has undergone cosmetic surgery (1) or whether they are on the waiting list (0), which acts as our control group.
- ▶ Surgery\_Text: This variable is the same as above but specifies group membership as text (we will use this variable when we create graphs but not for the main analysis).
- ▶ Clinic: This variable specifies which of 10 clinics the person attended to have their surgery.
- ▶ Age: This variable tells us the person's age in years.
- ▶ BDI: It is becoming increasingly apparent that people volunteering for cosmetic surgery (especially when the surgery is purely for vanity) might have very different personality profiles than the general public (Cook, Rossera, Toone, James, & Salmon, 2006). In particular, these people might have low self-esteem or be depressed. When looking at quality of life it is important to assess natural levels of depression, and this variable used the Beck Depression Inventory (BDI) to do just that.
- ▶ Reason: This dummy variable specifies whether the person had/is waiting to have surgery purely to change their appearance (0), or because of a physical reason (1).
- ▶ Reason\_Text: This variable is the same as above but contains text to define each group rather than a number.
- ▶ Gender: This variable simply specifies whether the person was a man (1) or a woman (0).

```
library(ggplot2)
ggplot(data = d, aes(y = Base_QoL, x = factor(Clinic), color = Surgery_Text)) +
  geom_boxplot()
```

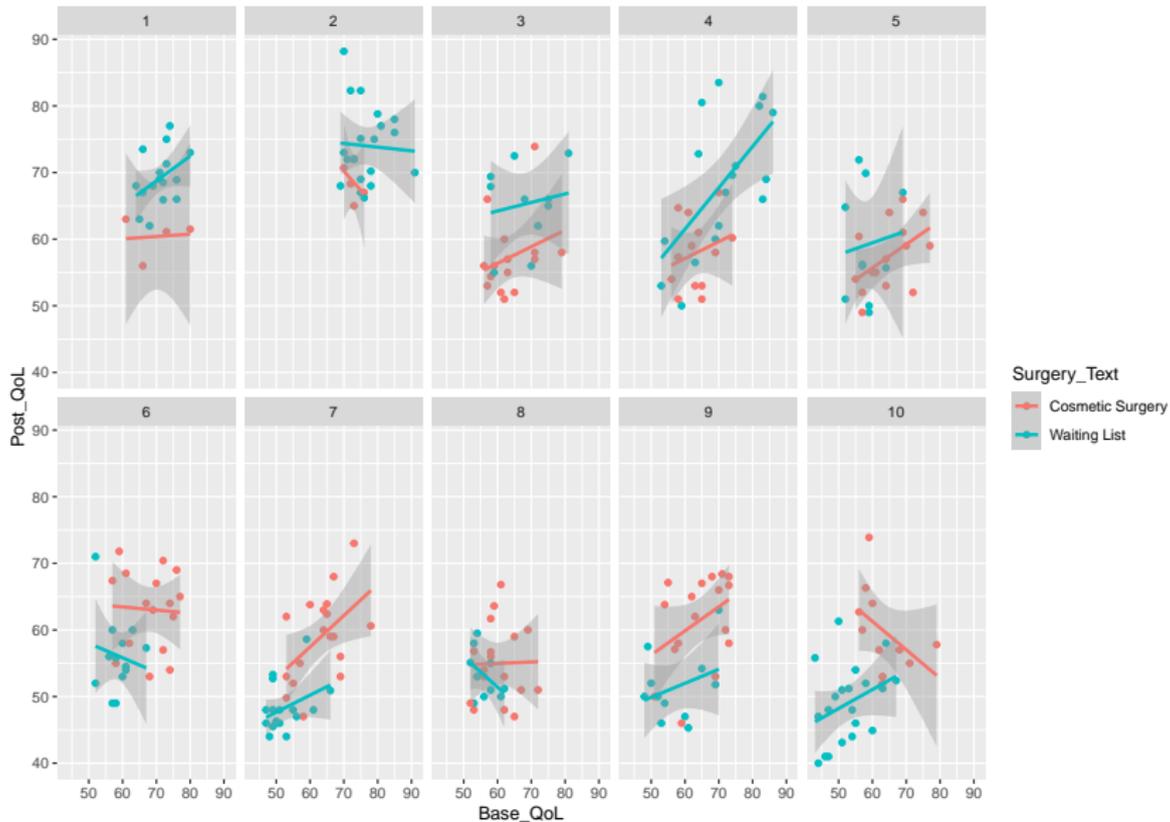


```
library(ggplot2)
ggplot(data = d, aes(y = Post_QoL, x = factor(Clinic), color = Surgery_Text)) +
  geom_boxplot()
```



```
library(ggplot2)
ggplot(data = d, aes(x = Base_QoL, y = Post_QoL, color = Surgery_Text)) +
  geom_point() + facet_wrap(~Clinic, ncol = 5) + geom_smooth(method = lm)
```

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



Ein gemischtes Modell ist hier eine gute Wahl:

- ▶ Es ermöglicht uns, alle uns vorliegenden Daten zu verwenden (höhere Stichprobengröße) und

Ein gemischtes Modell ist hier eine gute Wahl:

- ▶ Es ermöglicht uns, alle uns vorliegenden Daten zu verwenden (höhere Stichprobengröße) und
- ▶ die Korrelationen zwischen Daten von den gleichen Kliniken zu berücksichtigen.

Ein gemischtes Modell ist hier eine gute Wahl:

- ▶ Es ermöglicht uns, alle uns vorliegenden Daten zu verwenden (höhere Stichprobengröße) und
- ▶ die Korrelationen zwischen Daten von den gleichen Kliniken zu berücksichtigen.
- ▶ Wir werden auch weniger Parameter schätzen und Probleme mit mehreren Vergleichen vermeiden, die bei Verwendung separater Regressionen auftreten würden.

- ▶ Gewöhnliche Regression

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

- ▶ Gewöhnliche Regression

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon_i$$

Es ist wichtig, dass dieser Unterschied wenig mit den Variablen selbst und viel mit der Forschungsfrage zu tun hat! In vielen Fällen kann dieselbe Variable entweder als zufälliger oder als fester Effekt betrachtet werden (und manchmal sogar als beide gleichzeitig!). Beziehen Sie sich daher immer auf Ihre Fragen und Hypothesen, um Ihre Modelle entsprechend zu konstruieren.

$$Y_{ij} = \beta_{0j} + \beta_{1j} X_{ij} + \varepsilon_{ij}$$

$$b_{0j} = b_0 + u_{0j} \text{ (Random intercepts)}$$

$$b_{1j} = b_1 + u_{1j} \text{ (Random slopes)}$$

# Sollten meine Variablen feste oder zufällige Effekte sein?

Im Großen und Ganzen sind **feste Effekte** Variablen, von denen wir erwarten, dass sie sich auf die abhängige Variable auswirken: Sie werden in einer linearen Standardregression als erklärende Variablen bezeichnet. In unserem Fall sind wir daran interessiert, Schlussfolgerungen darüber zu ziehen, wie sich die schon vorhandene Lebensqualität auf die Lebensqualität nach der OP auswirkt. Die schon vorhandene Lebensqualität ist also ein fester Effekt.

# Sollten meine Variablen feste oder zufällige Effekte sein?

Im Großen und Ganzen sind **feste Effekte** Variablen, von denen wir erwarten, dass sie sich auf die abhängige Variable auswirken: Sie werden in einer linearen Standardregression als erklärende Variablen bezeichnet. In unserem Fall sind wir daran interessiert, Schlussfolgerungen darüber zu ziehen, wie sich die schon vorhandene Lebensqualität auf die Lebensqualität nach der OP auswirkt. Die schon vorhandene Lebensqualität ist also ein fester Effekt.

Auf der anderen Seite sind **zufällige Effekte** normalerweise *Gruppierungsfaktoren*, für die wir versuchen zu *kontrollieren*. Sie sind immer kategorisch, da Sie R nicht zwingen können, eine kontinuierliche Variable als zufälligen Effekt zu behandeln. In den meisten Fällen sind wir nicht speziell an ihren Auswirkungen auf die Antwortvariable interessiert, aber wir wissen, dass sie möglicherweise die Muster beeinflussen, die wir sehen.

# Sollten meine Variablen feste oder zufällige Effekte sein?

Im Großen und Ganzen sind **feste Effekte** Variablen, von denen wir erwarten, dass sie sich auf die abhängige Variable auswirken: Sie werden in einer linearen Standardregression als erklärende Variablen bezeichnet. In unserem Fall sind wir daran interessiert, Schlussfolgerungen darüber zu ziehen, wie sich die schon vorhandene Lebensqualität auf die Lebensqualität nach der OP auswirkt. Die schon vorhandene Lebensqualität ist also ein fester Effekt.

Auf der anderen Seite sind **zufällige Effekte** normalerweise *Gruppierungsfaktoren*, für die wir versuchen zu *kontrollieren*. Sie sind immer kategorisch, da Sie R nicht zwingen können, eine kontinuierliche Variable als zufälligen Effekt zu behandeln. In den meisten Fällen sind wir nicht speziell an ihren Auswirkungen auf die Antwortvariable interessiert, aber wir wissen, dass sie möglicherweise die Muster beeinflussen, die wir sehen.

Darüber hinaus sind die Daten für unseren Zufallseffekt nur ein Beispiel für alle Möglichkeiten. Und wir verallgemeinern die Ergebnisse anhand repräsentativer Stichproben auf eine ganze Population. Es ist uns egal, wie viel besser Patienten in Klinik A im Vergleich zu Patienten in Klinik B abgeschnitten haben, aber wir wissen, dass ihre jeweiligen Chirurgen ein Grund sein könnten, warum ihre Lebensqualität unterschiedlich wären, und wir möchten gerne wissen, wie viel Abweichungen sind darauf zurückzuführen, wenn wir die Lebensqualität für Patienten der Klinik Z vorhersagen.

## Mehr über zufällige Effekte

Beachten Sie, dass die goldene Regel lautet, dass Ihr zufälliger Effekt im Allgemeinen mindestens fünf Ebenen haben soll. Z.B.~würden wir Sex (ein zweistufiger Faktor: männlich oder weiblich) als festen, nicht zufälligen Effekt betrachten.

# Mehr über zufällige Effekte

Beachten Sie, dass die goldene Regel lautet, dass Ihr zufälliger Effekt im Allgemeinen mindestens fünf Ebenen haben soll. Z.B.~würden wir Sex (ein zweistufiger Faktor: männlich oder weiblich) als festen, nicht zufälligen Effekt betrachten.

Dies liegt einfach daran, dass die Schätzung der Varianz an wenigen Datenpunkten sehr ungenau ist. Mathematisch könnten Sie, aber Sie würden nicht viel Vertrauen in sie haben. Wenn Sie nur zwei oder drei Ebenen haben, wird das Modell Schwierigkeiten haben, die Varianz zu partitionieren - es gibt Ihnen eine Ausgabe, aber nicht unbedingt eine, der Sie vertrauen können.

# Mehr über zufällige Effekte

Beachten Sie, dass die goldene Regel lautet, dass Ihr zufälliger Effekt im Allgemeinen mindestens fünf Ebenen haben soll. Z.B.~würden wir Sex (ein zweistufiger Faktor: männlich oder weiblich) als festen, nicht zufälligen Effekt betrachten.

Dies liegt einfach daran, dass die Schätzung der Varianz an wenigen Datenpunkten sehr ungenau ist. Mathematisch könnten Sie, aber Sie würden nicht viel Vertrauen in sie haben. Wenn Sie nur zwei oder drei Ebenen haben, wird das Modell Schwierigkeiten haben, die Varianz zu partitionieren - es gibt Ihnen eine Ausgabe, aber nicht unbedingt eine, der Sie vertrauen können.

Der Name zufällig hat nicht viel mit mathematischer Zufälligkeit zu tun obwohl das verwirrend ist. Stellen Sie sie sich vorerst als *Gruppierungsvariablen* vor. Genau genommen geht es darum, unsere Modelle für unsere Fragen repräsentativ zu machen und bessere Schätzungen zu erhalten.

# Crossed and Nested random effects

Wiederholen wir dies mit einem anderen Beispiel: Ein Effekt wird (vollständig) gekreuzt, wenn alle Probanden alle Stufen dieses Effekts erfahren haben. Wenn Sie beispielsweise ein Düngungsexperiment an Sämlingen durchgeführt haben, die in einem saisonalen Wald wachsen, und in jeder Jahreszeit wiederholte Messungen über die Zeit (z. B. 3 Jahre) durchgeführt haben, möchten Sie möglicherweise einen gekreuzten Faktor namens Jahreszeit (Sommer1, Herbst1, Winter1, Frühling1, Summer2, . . . , Spring3), dh ein Faktor für jede Jahreszeit eines jeden Jahres. Dieser Gruppierungsfaktor würde die Tatsache erklären, dass alle Pflanzen im Experiment, unabhängig vom festen (Behandlungs-) Effekt (dh gedüngt oder nicht gedüngt), im zweiten Jahr einen sehr heißen Sommer oder im dritten einen sehr regnerischen Frühling erlebt haben könnten Jahr, und diese Bedingungen könnten Störungen in den erwarteten Mustern verursachen. Sie müssen nicht einmal Klimadaten zuordnen, um dies zu berücksichtigen! Sie wissen nur, dass alle Beobachtungen ab Frühjahr 3 einander ähnlicher sein können, weil sie dieselben Umwelteigenschaften hatten und nicht, weil sie auf Ihre Behandlung ansprechen.

## Nested random effects

*Pseudoreplikation* ist eine massive Erhöhung Ihrer Stichprobengröße durch Verwendung nicht unabhängiger Daten. Bei einer Stichprobengröße von 60.000 würden Sie mit ziemlicher Sicherheit einen „signifikanten“ Behandlungseffekt erzielen, der möglicherweise überhaupt keine ökologische Bedeutung hat. Und es verstößt gegen die Annahme der Unabhängigkeit von Beobachtungen, die für die lineare Regression von zentraler Bedeutung ist.

- ▶ Sie sollten die ML verwenden, wenn Sie Modelle mit verschiedenen festen Effekten vergleichen, da ML nicht auf den Koeffizienten der festen Effekte beruht

- ▶ Sie sollten die ML verwenden, wenn Sie Modelle mit verschiedenen festen Effekten vergleichen, da ML nicht auf den Koeffizienten der festen Effekte beruht
- ▶ Obwohl Sie ML zum Vergleichen von Modellen verwenden, sollten Sie Parameterschätzungen aus Ihrem endgültigen „besten“ REML-Modell melden, da ML die Varianz der zufälligen Effekte möglicherweise unterschätzt.

- ▶ Über experimentelles Design, Ihr System und die gesammelten Daten sowie Ihre Fragen nachdenken.

- ▶ Über experimentelles Design, Ihr System und die gesammelten Daten sowie Ihre Fragen nachdenken.
- ▶ Wenn Ihre zufälligen Effekte dazu dienen, sich mit Pseudoreplikation zu befassen, spielt es keine Rolle, ob sie “signifikant” sind oder nicht: Sie sind Teil Ihres Designs und müssen einbezogen werden.

- ▶ Über experimentelles Design, Ihr System und die gesammelten Daten sowie Ihre Fragen nachdenken.
- ▶ Wenn Ihre zufälligen Effekte dazu dienen, sich mit Pseudoreplikation zu befassen, spielt es keine Rolle, ob sie “signifikant” sind oder nicht: Sie sind Teil Ihres Designs und müssen einbezogen werden.
- ▶ Es wird empfohlen REML-Schätzer zum Vergleich von Modellen mit verschiedenen zufälligen Effekten (wir halten feste Effekte konstant) zu nutzen.

- ▶ Über experimentelles Design, Ihr System und die gesammelten Daten sowie Ihre Fragen nachdenken.
- ▶ Wenn Ihre zufälligen Effekte dazu dienen, sich mit Pseudoreplikation zu befassen, spielt es keine Rolle, ob sie “signifikant” sind oder nicht: Sie sind Teil Ihres Designs und müssen einbezogen werden.
- ▶ Es wird empfohlen REML-Schätzer zum Vergleich von Modellen mit verschiedenen zufälligen Effekten (wir halten feste Effekte konstant) zu nutzen.

HINWEIS: Variieren Sie NICHT zufällige und feste Effekte gleichzeitig - entweder mit Ihrer Zufallseffektstruktur oder mit Ihrer festen Effektstruktur an einem bestimmten Punkt.

HINWEIS 2: Vergleichen Sie keine lmer-Modelle mit lm-Modellen (oder glmer mit glm).

- ▶ Ein vollständiges Modell anpassen (sogar komplexer als erwartet oder gewünscht)
- ▶ Sortieren Sie die Zufallseffektstruktur (verwenden Sie REML-Likelihood oder REML AIC oder BIC).
- ▶ Struktur mit festen Effekten sortieren (entweder REML, die F-Statistik oder die t-Statistik verwenden oder verschachtelte ML-Modelle vergleichen – halten Sie Ihre zufälligen Effekte konstant)
- ▶ Sobald Sie das endgültige Modell erreicht haben, präsentieren Sie es mithilfe der REML-Schätzung

HINWEIS: Es besteht die Gefahr, dass es wie ein gebrochener Rekord klingt: Ich denke, es ist am besten, zu entscheiden, was Ihr Modell auf Biologie / Ökologie / Datenstruktur usw. basiert, als blind der Modellauswahl zu folgen. Nur weil etwas nicht signifikant ist, heißt das nicht unbedingt, dass Sie es immer loswerden sollten.

# Kovarianzstruktur

- ▶ Bei den zufälligen Effekten brauchen wir Annahmen über Korrelationen (variance–covariance matrix)
- ▶ Simple Struktur – Fehler I. Art (parameter überwiegend signifikant).  
Komplizierte Struktur – Fehler II. Art (parameter überwiegend nicht signifikant).
- ▶ Variance components

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

- ▶ Unstructured

$$\begin{pmatrix} \sigma_1^2 & \sigma_{21} & \sigma_{31} \\ \sigma_{21} & \sigma_2^2 & \sigma_{32} \\ \sigma_{31} & \sigma_{32} & \sigma_3^2 \end{pmatrix}$$

# Beispiel. Standard-Regression (ANOVA)

```
m <- lm(Post_QoL ~ factor(Surgery), data = d)
summary(m)

##
## Call:
## lm(formula = Post_QoL ~ factor(Surgery), data = d)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -19.916  -7.271  -1.271   7.084  28.284
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    59.9159    0.7731  77.498  <2e-16 ***
## factor(Surgery)1 -0.6449    1.1222  -0.575   0.566
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.31 on 274 degrees of freedom
## Multiple R-squared:  0.001204, Adjusted R-squared:  -0.002442
## F-statistic: 0.3302 on 1 and 274 DF, p-value: 0.566

anova(m)

## Analysis of Variance Table
##
## Response: Post_QoL
##              Df Sum Sq Mean Sq F value Pr(>F)
## factor(Surgery)  1    28.6   28.620  0.3302  0.566
## Residuals      274 23747.9  86.671
```

```

m1 <- lm(Post_QoL ~ factor(Surgery) + Base_QoL, data = d)
summary(m1)

##
## Call:
## lm(formula = Post_QoL ~ factor(Surgery) + Base_QoL, data = d)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -13.4142  -5.1326  -0.6495   4.0540  23.5005
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    18.14702    2.90767   6.241 1.65e-09 ***
## factor(Surgery)1 -1.69723    0.84404  -2.011 0.0453 *
## Base_QoL        0.66504    0.04537  14.659 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.977 on 273 degrees of freedom
## Multiple R-squared:  0.4411, Adjusted R-squared:  0.437
## F-statistic: 107.7 on 2 and 273 DF,  p-value: < 2.2e-16

anova(m1)

```

```

## Analysis of Variance Table
##
## Response: Post_QoL
##              Df    Sum Sq Mean Sq F value Pr(>F)
## factor(Surgery)  1    28.6    28.6    0.588 0.4439
## Base_QoL        1 10459.6 10459.6 214.888 <2e-16 ***
## Residuals      273 13288.3    48.7
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

```

# Nur intercept

```
library(lme4)

## Loading required package: Matrix

library(nlme)

##
## Attaching package: 'nlme'

## The following object is masked from 'package:lme4':
##
##      lmList
m_intercept <- gls(Post_QoL ~ 1, data = d, method = "ML") # uses ML-Estimation
summary(m_intercept)

## Generalized least squares fit by maximum likelihood
## Model: Post_QoL ~ 1
## Data: d
##      AIC      BIC    logLik
## 2017.124 2024.365 -1006.562
##
## Coefficients:
##              Value Std.Error  t-value p-value
## (Intercept) 59.60978 0.5596972 106.5036      0
##
## Standardized residuals:
##      Min      Q1      Med      Q3      Max
## -2.1127754 -0.7875625 -0.1734394 0.7962286 3.0803354
##
## Residual standard error: 9.281527
## Degrees of freedom: 276 total; 275 residual
```

# Zufallseffekt Clinic

```
# m_rand_intercept <- lmer(Post_QoL ~ 1|Clinic, data = d,  
# REML = FALSE) # uses ML-Estimation  
randomInterceptOnly <- lme(Post_QoL ~ 1, random = -1 | Clinic,  
  data = d, method = "ML") # uses ML-Estimation  
anova(m_intercept, randomInterceptOnly)
```

```
##           Model df      AIC      BIC    logLik  Test  L.Ratio  
## m_intercept      1  2 2017.124 2024.365 -1006.5622  
## randomInterceptOnly  2  3 1911.473 1922.334 -952.7364 1 vs 2 107.6517  
##           p-value  
## m_intercept  
## randomInterceptOnly <.0001
```

# Zufällige Effekte mit *Surgery* als fester Effekt

```
# Add surgery as a predictor
randomInterceptSurgery <- lme(Post_QoL ~ Surgery, data = d, random = ~1 |
  Clinic, method = "ML")
summary(randomInterceptSurgery)
```

```
## Linear mixed-effects model fit by maximum likelihood
## Data: d
##      AIC      BIC    logLik
## 1910.137 1924.619 -951.0686
##
## Random effects:
## Formula: ~1 | Clinic
##      (Intercept) Residual
## StdDev:      6.099513  7.18542
##
## Fixed effects: Post_QoL ~ Surgery
##              Value Std.Error DF t-value p-value
## (Intercept) 59.30517  2.0299632 265 29.21490  0.000
## Surgery      1.66583  0.9091314 265  1.83233  0.068
## Correlation:
##      (Intr)
## Surgery -0.21
##
## Standardized Within-Group Residuals:
##      Min      Q1      Med      Q3      Max
## -1.8904290 -0.7191399 -0.1420998  0.7177762  2.8644538
##
## Number of Observations: 276
## Number of Groups: 10
```

# Zufällige Effekte mit *Surgery* und *Base\_QoL* als feste Effekte

```
## Fit effect of surgery and baseline QoL- random intercepts
## across clinics
randomInterceptSurgeryQoL <- lme(Post_QoL ~ Surgery + Base_QoL,
  data = d, random = ~1 | Clinic, method = "ML")
summary(randomInterceptSurgeryQoL)

## Linear mixed-effects model fit by maximum likelihood
## Data: d
##      AIC      BIC    logLik
## 1847.49 1865.592 -918.745
##
## Random effects:
## Formula: ~1 | Clinic
##      (Intercept) Residual
## StdDev:      3.039264 6.518986
##
## Fixed effects: Post_QoL ~ Surgery + Base_QoL
##      Value Std.Error DF   t-value p-value
## (Intercept) 29.563601  3.471879 264   8.515160 0.0000
## Surgery      -0.312999  0.843145 264  -0.371228 0.7108
## Base_QoL      0.478630  0.052774 264   9.069465 0.0000
## Correlation:
##      (Intr) Surgery
## Surgery  0.102
## Base_QoL -0.947 -0.222
##
## Standardized Within-Group Residuals:
##      Min      Q1      Med      Q3      Max
## -1.8872666 -0.7537675 -0.0954987  0.5657241  3.0020852
##
## Number of Observations: 276
## Number of Groups: 10
```

# Modellenvergleich

```
anova(randomInterceptOnly, randomInterceptSurgery, randomInterceptSurgeryQoL)
```

```
##           Model df      AIC      BIC   logLik  Test L.Ratio
## randomInterceptOnly      1  3 1911.473 1922.334 -952.7364
## randomInterceptSurgery    2  4 1910.137 1924.619 -951.0686 1 vs 2  3.33564
## randomInterceptSurgeryQoL  3  5 1847.490 1865.592 -918.7450 2 vs 3 64.64721
##                               p-value
## randomInterceptOnly
## randomInterceptSurgery    0.0678
## randomInterceptSurgeryQoL <.0001
```

# Random slopes I

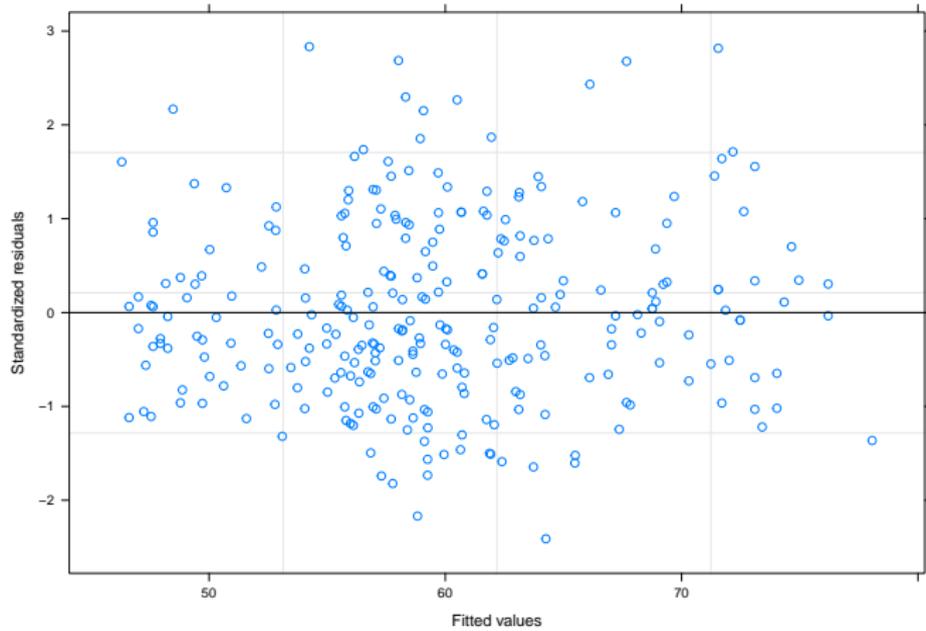
```
## Fit effect of surgery and baseline QoL- random slopes and
## intercepts across clinics
addRandomSlope <- lme(Post_QoL ~ Surgery + Base_QoL, data = d,
  random = ~Surgery | Clinic, method = "ML")
summary(addRandomSlope)
```

```
## Linear mixed-effects model fit by maximum likelihood
## Data: d
##      AIC      BIC    logLik
## 1812.624 1837.967 -899.3119
##
## Random effects:
## Formula: ~Surgery | Clinic
## Structure: General positive-definite, Log-Cholesky parametrization
##      StdDev   Corr
## (Intercept) 6.132655 (Intr)
## Surgery      6.197489 -0.965
## Residual     5.912335
##
## Fixed effects: Post_QoL ~ Surgery + Base_QoL
##      Value Std.Error DF t-value p-value
## (Intercept) 40.10253  3.892945 264 10.301334 0.0000
## Surgery     -0.65453  2.110917 264 -0.310069 0.7568
## Base_QoL    0.31022  0.053506 264  5.797812 0.0000
## Correlation:
##      (Intr) Surgery
## Surgery -0.430
## Base_QoL -0.855 -0.063
##
## Standardized Within-Group Residuals:
##      Min      Q1      Med      Q3      Max
## -2.4114778 -0.6628574 -0.1138411  0.6833110  2.8334730
##
## Number of Observations: 276
## Number of Groups: 10
```

# Random slopes II

```
anova(randomInterceptSurgeryQoL, addRandomSlope)
```

```
##           Model df      AIC      BIC   logLik   Test  L.Ratio
## randomInterceptSurgeryQoL  1  5 1847.490 1865.592 -918.7450
## addRandomSlope            2  7 1812.624 1837.966 -899.3119 1 vs 2 38.86626
##                               p-value
## randomInterceptSurgeryQoL
## addRandomSlope            <.0001
```



```
## Fit effect of surgery and baseline QoL, Reason and
## Reason*Surgery Interaction- random slopes and intercepts
## across clinics
```

```
addReason <- lme(Post_QoL ~ Surgery + Base_QoL + Reason, data = d,
  random = ~Surgery | Clinic, method = "ML")
addReason <- update(addRandomSlope, . - . + Reason)
summary(addReason)
```

```
## Linear mixed-effects model fit by maximum likelihood
## Data: d
##      AIC      BIC    logLik
## 1810.825 1839.788 -897.4124
##
## Random effects:
## Formula: ~Surgery | Clinic
## Structure: General positive-definite, Log-Cholesky parametrization
##           StdDev  Corr
## (Intercept) 5.838006 (Intr)
## Surgery      6.171602 -0.969
## Residual     5.886964
##
## Fixed effects: Post_QoL ~ Surgery + Base_QoL + Reason
##           Value Std.Error DF   t-value p-value
## (Intercept) 41.43182  3.929923 263 10.542653 0.0000
## Surgery     -0.56817  2.106684 263 -0.269697 0.7876
## Base_QoL     0.30535  0.053417 263  5.716335 0.0000
## Reason      -1.69137  0.854411 263 -1.979571 0.0488
## Correlation:
##           (Intr) Surgry Bas_QL
## Surgery   -0.400
## Base_QoL  -0.862 -0.064
## Reason    -0.231 -0.042  0.118
##
## Standardized Within-Group Residuals:
##           Min      Q1      Med      Q3      Max
## -2.2006350 -0.6830482 -0.1256909  0.6730347  2.9817687
##
## Number of Observations: 276
## Number of Groups: 10
```

```

finalModel <- lme(Post_QoL ~ Surgery + Base_QoL + Reason + Reason:Surgery,
  data = d, random = ~Surgery | Clinic, method = "ML")
summary(finalModel)

## Linear mixed-effects model fit by maximum likelihood
## Data: d
##      AIC      BIC    logLik
## 1807.045 1839.629 -894.5226
##
## Random effects:
## Formula: ~Surgery | Clinic
## Structure: General positive-definite, Log-Cholesky parametrization
##           StdDev  Corr
## (Intercept) 5.482359 (Intr)
## Surgery      5.417495 -0.946
## Residual     5.818911
##
## Fixed effects: Post_QoL ~ Surgery + Base_QoL + Reason + Reason:Surgery
##           Value Std.Error DF  t-value p-value
## (Intercept)  42.51781  3.875317 262 10.971441  0.0000
## Surgery      -3.18768  2.185367 262 -1.458646  0.1459
## Base_QoL      0.30536  0.053125 262  5.747835  0.0000
## Reason       -3.51515  1.140934 262 -3.080939  0.0023
## Surgery:Reason  4.22129  1.700269 262  2.482718  0.0137
## Correlation:
##           (Intr) Surgry Bas_QL Reason
## Surgery      -0.356
## Base_QoL     -0.865 -0.078
## Reason       -0.233  0.306  0.065
## Surgery:Reason  0.096 -0.505  0.024 -0.661
##
## Standardized Within-Group Residuals:
##           Min      Q1      Med      Q3      Max
## -2.2331483 -0.6972193 -0.1541073  0.6326387  3.1641799
##
## Number of Observations: 276
## Number of Groups: 10

intervals(finalModel, 0.95)

```

```

## Approximate 95% confidence intervals
##
## Fixed effects:
##           lower      est.      upper
## (Intercept) 34.9565174 42.5178133 50.0791092
## Surgery     -7.4516406 -3.1876776  1.0762854
## Base_QoL    0.2017009  0.3053562  0.4090115
## Reason      -5.7412741 -3.5151488 -1.2890236
## Surgery:Reason 0.9038215  4.2212894  7.5387574
## attr(,"label")
## [1] "Fixed effects:"
##
## Random Effects:
##   Level: Clinic
##           lower      est.      upper
## sd((Intercept)) 3.3130516 5.4823585 9.072076
## sd(Surgery)     3.1323585 5.4174947 9.369697
## cor((Intercept),Surgery) -0.9937882 -0.9455544 -0.598258
##
## Within-group standard error:
##   lower      est.      upper
## 5.331220 5.818911 6.351214
anova(addRandomSlope, addReason, finalModel)

##           Model df      AIC      BIC   logLik   Test  L.Ratio p-value
## addRandomSlope  1  7 1812.624 1837.966 -899.3119
## addReason       2  8 1810.825 1839.788 -897.4124 1 vs 2 3.798961 0.0513
## finalModel     3  9 1807.045 1839.629 -894.5226 2 vs 3 5.779555 0.0162

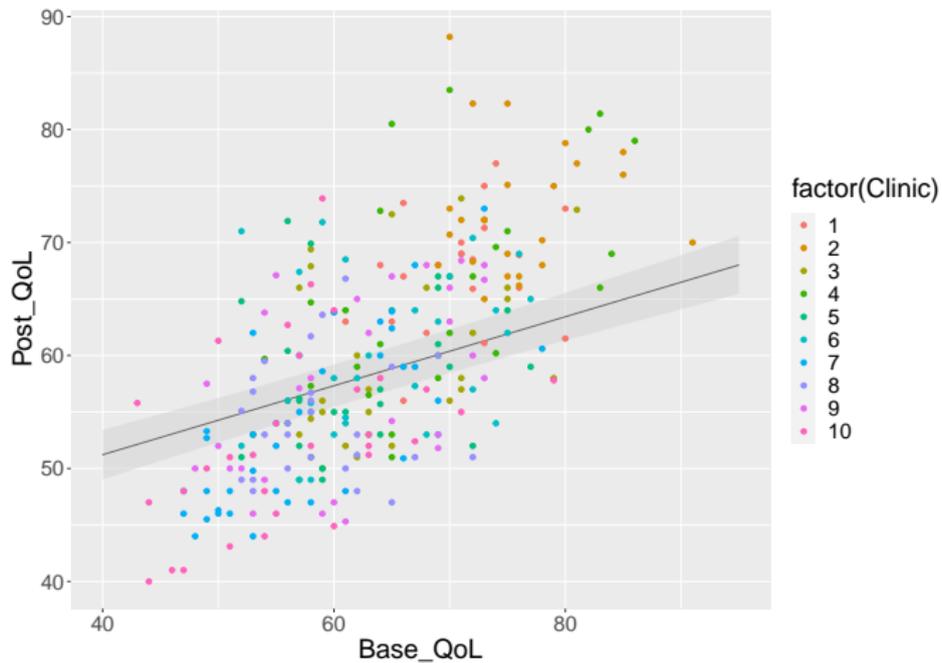
```

# Visualisierung der Ergebnisse I

```
library(ggeffects)
# Extract the prediction data frame
pred.mm <- ggpredict(finalModel, terms = c("Base_QoL"))

ggplot(pred.mm) +
  geom_line(aes(x = x, y = predicted)) +           # slope
  geom_ribbon(aes(x = x, ymin = predicted - std.error, ymax = predicted + std.error),
            fill = "lightgrey", alpha = 0.5) + # error band
  geom_point(data = d,                            # adding the raw data (scaled values)
            aes(x = Base_QoL, y = Post_QoL, colour = factor(Clinic))) +
  labs(x = "Base_QoL", y = "Post_QoL") +
  theme(text = element_text(size = 20))
```

# Visualisierung der Ergebnisse II



# References

<https://ourcodingclub.github.io/tutorials/mixed-models/>

A. Field

- ▶ Gemischte Modelle konvergieren manchmal nicht
- ▶ Da kann Reskalierung helfen

```
d$BDI_Centred <- scale(d$BDI, scale = F)
head(d$BDI)
```

```
## [1] 12 16 13 11 11 10
```

```
head(d$BDI_Centred)
```

```
##           [,1]
## [1,] -11.054348
## [2,]  -7.054348
## [3,] -10.054348
## [4,] -12.054348
## [5,] -12.054348
## [6,] -13.054348
```

```
# #-----  
# #OLIVER TWISTED: CENTRING #Grand Mean Centring surgeryData  
# = read.delim('Cosmetic Surgery.dat', header = TRUE)  
# surgeryData$BDI_Centred<-scale(surgeryData$BDI, scale = F)  
# print(surgeryData) #Group Mean Centring  
# groupMeans<-aggregate(surgeryData$BDI,  
# list(surgeryData$Clinic), mean)  
# names(groupMeans)<-c('Clinic', 'BDI_Mean')  
# print(groupMeans) surgeryData<-merge(surgeryData,  
# groupMeans, by = 'Clinic')  
# surgeryData$BDI_GpMC<-surgeryData$BDI-surgeryData$BDI_Mean
```