
Aufgabenblatt 4

UK Erweiterungen des linearen Modells

Cordula Eggerth

Matrikelnummer: 00750881

Kursleiter:

Prof. Dr. Marcus Hudec &

Prof. Dr. Wilfried Grossmann

Sommersemester 2019

Aufgabe 1 (Zufallseffekte):

Um die Variation der Milchfettleistung von Bullen zu untersuchen werden zufällig zwei Bullen ausgewählt und je drei ihrer Töchter untersucht. Die Ergebnisse sind durch folgende Werte gegeben:

Bulle 1: Milchfettwerte der Töchter: 19,17, 15

Bulle 2: Milchfettwerte der Töchter: 25, 5, 15

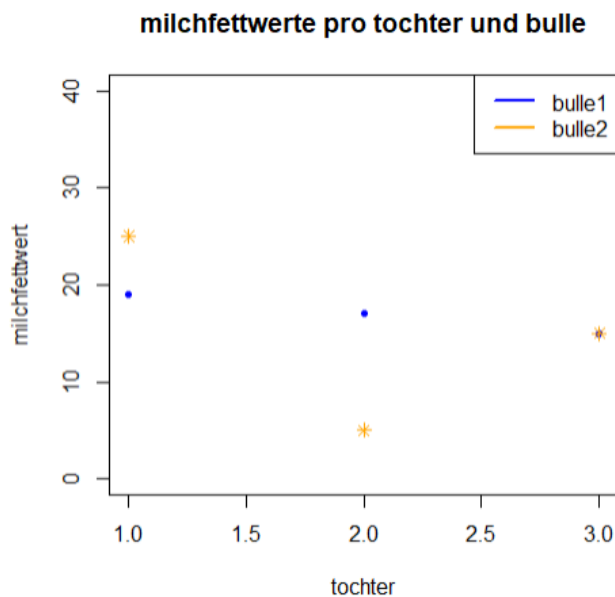
a) Berechne nach den Formeln die Variation der Milchfettleistungen zwischen den Klassen (Bullen) und die Variation der Milchfettleistungen innerhalb der Klassen. Ermittle daraus die Varianzschätzung für die Variation zwischen den Klassen. Ist diese Schätzung sinnvoll?

b) Ermittle die Schätzung der Varianzen mittels REML mit der Maximum Likelihood-methode und der REML Methode und bestimme die Intraclasskorrelation.

Für die Daten wurde das folgende data.frame angelegt:

```
> milchfett_df
      tochter1 tochter2 tochter3
bulle1       19       17       15
bulle2       25        5       15
```

Im untenstehenden Plot wurden die Milchfettwerte pro Tochter und Bulle eingezeichnet:



Deskriptive Statistiken zum Datensatz:

```
> summary(t(milchfett_df))
      bulle1      bulle2
Min.   :15    Min.    : 5
1st Qu.:16    1st Qu.:10
Median :17    Median :15
Mean   :17    Mean   :15
3rd Qu.:18    3rd Qu.:20
Max.   :19    Max.   :25
```

```
> apply(milchfett_df, 1, sd)
bulle1 bulle2
      2      10
```

1.a.

m ... Anzahl der Gruppen (i.e. 2 Bullen)

n ... Anzahl Beobachtungen pro Gruppe (i.e. 3 Töchter pro Bulle)

Variation der Milchfettwerte ZWISCHEN den Klassen (Bullen)

```
> variation_between <- sum((rowMeans(milchfett)-mean(milchfett))^2)
> variation_between
[1] 2
```

Variation der Milchfettwerte INNERHALB der Klassen (Bullen)

```
> variation_within <- sum((milchfett[1, ]-rowMeans(milchfett)[1])^2)+
  sum((milchfett[2, ]-rowMeans(milchfett)[2])^2)
> variation_within
[1] 208
```

MSE:

```
> mse <- variation_within / (m*(n-1))
> mse
[1] 52
```

MSM:

```
> msm <- variation_between / (m-1)
> msm
[1] 2
```

Varianzschätzung:

```
> varianz_schaetzung <- (msm-mse) / n
> varianz_schaetzung
[1] -16.66667
```

Ergebnisinterpretation:

Die Varianzschätzung ist in diesem Beispiel negativ, d.h. sie ist demnach keine sinnvolle Schätzung.

1.b.

In der Folge wurde eine Schätzung der Varianzen mittels REML mit der Maximum-Likelihood-methode und der REML-Methode durchgeführt.

```
> data
  bulle tochter milchfett
1     1         1        19
2     1         2        17
3     1         3        15
4     2         1        25
5     2         2         5
6     2         3        15
```

```
> # ML:
> ML <- lmer(data$milchfett~tochter+(1|bulle), REML=FALSE)
> summary(ML)
Linear mixed model fit by maximum likelihood ['lmerMod']
Formula: data$milchfett ~ tochter + (1 | bulle)
```

AIC	BIC	logLik	deviance	df.resid
43.3	42.2	-16.6	33.3	1

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.5492	-0.5809	0.0000	0.5809	1.5492

Random effects:

Groups	Name	Variance	Std.Dev.
bulle	(Intercept)	0	0.000
Residual		15	3.873

Number of obs: 6, groups: bulle, 2

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	22.000	2.739	8.033
tochter2	-11.000	3.873	-2.840
tochter3	-7.000	3.873	-1.807

Correlation of Fixed Effects:

	(Intr)	tchtr2
tochter2	-0.707	
tochter3	-0.707	0.500

convergence code: 0

```
> coef(ML) # koeffizienten
$`bulle`
  (Intercept) tochter2 tochter3
1           22         -11        -7
2           22         -11        -7

attr(,"class")
[1] "coef.mer"
> fixef(ML) # feste effekte
(Intercept)  tochter2  tochter3
          22          -11          -7
> ranef(ML) # zufaellige effekte
$`bulle`
  (Intercept)
1           0
2           0
```

with conditional variances for "bulle"

```
> # REML:
> REML <- lmer(data$milchfett~tochter+(1|bulle), REML=TRUE)
> summary(REML)
Linear mixed model fit by REML ['lmerMod']
Formula: data$milchfett ~ tochter + (1 | bulle)
```

REML criterion at convergence: 20.8

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.0955	-0.4108	0.0000	0.4108	1.0955

Random effects:

Groups	Name	Variance	Std.Dev.
bulle	(Intercept)	0	0.000
Residual		30	5.477

Number of obs: 6, groups: bulle, 2

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	22.000	3.873	5.680
tochter2	-11.000	5.477	-2.008
tochter3	-7.000	5.477	-1.278

Correlation of Fixed Effects:

(Intr)	tchtr2
tochter2	-0.707
tochter3	-0.707 0.500

convergence code: 0

```
> coef(REML) # koeffizienten
$`bulle`
  (Intercept) tochter2 tochter3
1           22         -11        -7
2           22         -11        -7

attr(,"class")
[1] "coef.mer"
> fixef(REML) # feste effekte
(Intercept)   tochter2   tochter3
           22         -11         -7
> ranef(REML) # zufaellige effekte
$`bulle`
  (Intercept)
1           0
2           0
```

with conditional variances for "bulle"

Ergebnisinterpretation:

Die Ergebnisse, die mittels der ML- und der REML-Methode sind sehr ähnlich. Die Koeffizientenschätzungen für die festen Effekte sind gleich, während die Schätzung der zufälligen Effekte sich unterscheidet.

Basierend auf den Ergebnissen der 'summary(ML)' wurde die *Intra-Class-Korrelation* berechnet:

```
> # basierend auf 'summary(ML)' ergebnissen
> sigma <- 15
> sigma.a <- 0
> intraclass_correlation <- sigma.a / (sigma.a+sigma)
> intraclass_correlation
[1] 0
```

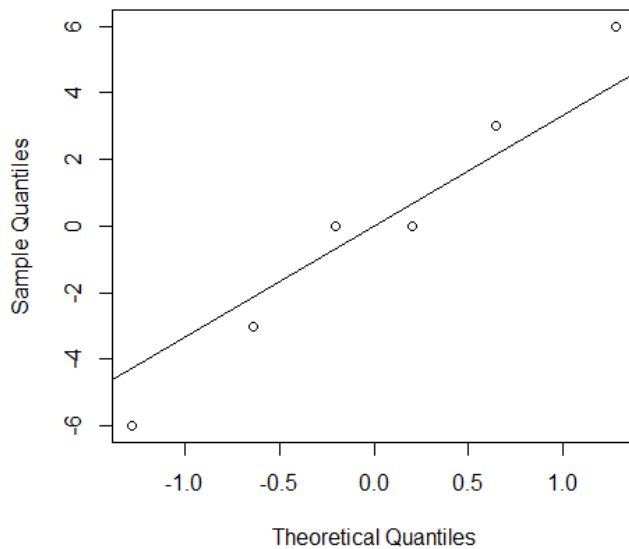
Ergebnisinterpretation:

In diesem Beispiel ist die Intra-Class-Korrelation 0, d.h. es gibt keine Korrelation zwischen verschiedenen Beobachtungen in derselben Klasse (also keine Korrelation zwischen verschiedenen Töchtern desselben Bullen).

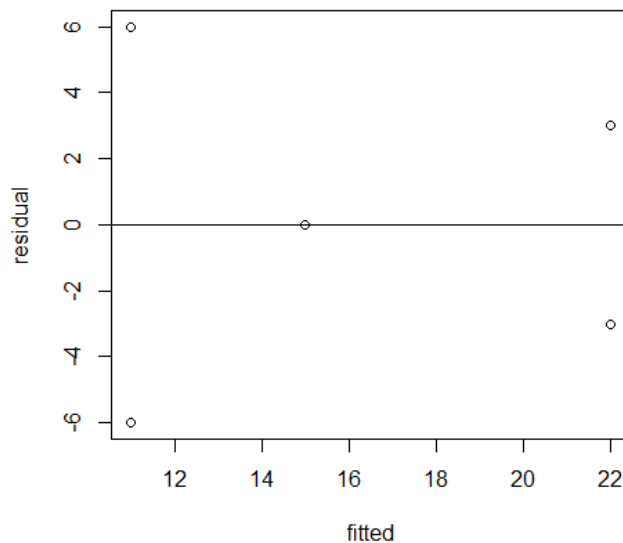
Diagnostic Plots:

Da es sich hier um sehr wenige Beobachtungen handelt, sind die Diagnostic Plots nicht so aussagekräftig, wie im Fall von einer größeren Stichprobe. Aber im QQ-Plot ist zu erkennen, dass die Punkte relativ nahe an der QQ-Linie liegen. Im Plot „Residuals vs. Fitted“ sehen die Residuen so aus als wären sie zufällig um die horizontale Nulllinie verstreut.

QQ-Plot:



Residuals vs. Fitted:



Modellvergleiche:

Verglichen werden die obigen Modell mit dem Nullmodell, wobei im Nullmodell die Milchfettwerte (i.e. die abhängige Variable) nur durch eine Konstante erklärt werden. Hierfür wird ein lineares Modell (`lm()`) verwendet:

```
> nullmodell <- lm(data$milchfett~1)
> summary(nullmodell)

Call:
lm(formula = data$milchfett ~ 1)

Residuals:
    1     2     3     4     5     6 
    3     1    -1     9   -11    -1 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    16.000      2.671   5.991  0.00186 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 6.542 on 5 degrees of freedom
```

Der Vergleich des REML-/ML-Modells mit dem Nullmodell erfolgt mittels `anova()`. Im vorliegenden Fall wurde das REML-Modell automatisch von R „refitted“ mit ML. Somit wurde eigentlich das ML-Modell herangezogen zum Vergleich.

```
> anova(REML, nullmodell)
refitting model(s) with ML (instead of REML)
Data: NULL
Models:
nullmodell: data$milchfett ~ 1
REML: data$milchfett ~ tochter + (1 | bulle)
      Df    AIC    BIC logLik deviance Chisq Chi Df Pr(>Chisq)
nullmodell  2 42.473 42.056 -19.236   38.473      3      0.1579
REML       5 43.276 42.234 -16.638   33.276 5.197      3      0.1579

> anova(ML, nullmodell)
Data: NULL
Models:
nullmodell: data$milchfett ~ 1
ML: data$milchfett ~ tochter + (1 | bulle)
      Df    AIC    BIC logLik deviance Chisq Chi Df Pr(>Chisq)
nullmodell  2 42.473 42.056 -19.236   38.473      3      0.1579
ML         5 43.276 42.234 -16.638   33.276 5.197      3      0.1579
```

Berechnung der Log-Likelihood der beiden Modelle:

```
> reml.LL <- as.numeric(2*(logLik(REML)-logLik(nullmodell)))
> reml.LL
[1] 17.6759
> ml.LL <- as.numeric(2*(logLik(ML)-logLik(nullmodell)))
> ml.LL
[1] 5.196998
```

Berechnung der p-Werte der Chi-Quadrat-Verteilung auf dem 5% Signifikanzniveau:

```
> pchisq(reml.LL,1,lower=FALSE) # hochsignifikant auf 0.05 level  
[1] 2.619247e-05  
> pchisq(ml.LL,1,lower=FALSE) # signifikant auf 0.05 level  
[1] 0.02262593
```


R-Code zu Aufgabe 1 (Zufallseffekte):

```
#####
# AUFGABE 1 (Zufallseffekte)
#####
# Um die Variation der Milchfettleistung von Bullen zu untersuchen werden zufällig
# zwei Bullen ausgewählt und je drei ihrer Töchter untersucht.
# Die Ergebnisse sind durch folgende Werte gegeben:
# Bulle 1: Milchfettwerte der Töchter: 19,17, 15
# Bulle 2: Milchfettwerte der Töchter: 25, 5, 15
# a) Berechne nach den Formeln die Variation der Milchfettleistungen zwischen den Kl
#     (Bullen) und die Variation der Milchfettleistungen innerhalb der Klassen. Ermitt
#     daraus die Varianzschätzung für die Variation zwischen den Klassen.
#     Ist diese Schätzung sinnvoll?
# b) Ermittle die Schätzung der Varianzen mittels REML mit der Maximum-Likelihoodmet
#     und der REML Methode und bestimme die Intraclasskorrelation.

milchfett_df <- data.frame(tochter1=c(19,25),
                           tochter2=c(17,5),
                           tochter3=c(15,15))
rownames(milchfett_df) <- c("bulle1","bulle2")
milchfett_df[1,] <- as.numeric(milchfett_df[1,])
milchfett_df[2,] <- as.numeric(milchfett_df[2,])

# plot: werte pro tochter und bulle
plot(milchfett_df[1, ], xlim=c(1,3), ylim=c(0,40),
     type="p", ylab="milchfettwert", xlab="tochter", col="blue",
     pch=20)
points(milchfett_df[2, ], col="orange", pch=8)
legend("topright", legend=c("bulle1", "bulle2"),
     col=c("blue", "orange"), lwd=2)

# deskriptive statistiken
summary(t(milchfett_df))
apply(milchfett_df,1,sd)

# 1.a.

# variation der milchfettwerte ZWISCHEN den klassen (bullen)
# m ... anzahl gruppen (i.e. 2 bullen)
# n ... anzahl beobachtungen pro gruppe (i.e. 3 toechter pro bulle)
m <- 2
n <- 3
milchfett <- as.matrix(milchfett_df)
|
variation_between <- sum((rowMeans(milchfett)-mean(milchfett))^2)
variation_between

# variation der milchfettwerte INNERHALB der klassen (bullen)
variation_within <- sum((milchfett[1, ]-rowMeans(milchfett)[1])^2)+
  sum((milchfett[2, ]-rowMeans(milchfett)[2])^2)
variation_within

# daraus varianzschätzung fuer die variation zwischen den klassen
mse <- variation_within / (m*(n-1))
msm <- variation_between / (m-1)

varianz_schaetzung <- (msm-mse) / n
varianz_schaetzung

# ergebnisinterpretation:
# die varianzschätzung ist hier negativ, d.h. sie ist demnach
# keine sinnvolle schätzung
```

```
# 1.b.

tochter <- factor(c(rep(1:3, 2)))
bulle <- factor(c(1,1,1,2,2,2))
data <- data.frame(bulle, tochter,
                  milchfett=c(19,17,15,25,5,15))

# ML:
ML <- lmer(data$milchfett~tochter+(1|bulle), REML=FALSE)
summary(ML)
coef(ML) # koeffizienten
fixef(ML) # feste effekte
ranef(ML) # zufaellige effekte

# REML:
REML <- lmer(data$milchfett~tochter+(1|bulle), REML=TRUE)
summary(REML)
coef(REML) # koeffizienten
fixef(REML) # feste effekte
ranef(REML) # zufaellige effekte

# ergebnisinterpretation von ML und REML:
# die ergebnisse sind in diesem bsp. nicht unterschiedlich

# intraclasskorrelation:
# basierend auf 'summary(ML)' ergebnissen
sigma <- 15
sigma.a <- 0
intraclass_correlation <- sigma.a / (sigma.a+sigma)

# ergebnisinterpretation:
# es gibt hier keine intraclass correlation, d.h. keine
# korrelation zwischen verschiedenen beobachtungen in
# derselben klasse (bulle)

# vergleich von modellen zum beispiel:
# diagnostic plots:
qqnorm(resid(REML),main="")
qqline(resid(REML))

plot(fitted(REML),resid(REML),
     xlab="fitted",ylab="residual")
abline(0,0)

# vergleiche mit dem nullmodell:
nullmodell <- lm(data$milchfett~1) # erklarung nur durch konstante
summary(nullmodell)

# vgl. mit anova (ML vs. nullmodell)
# hier: REML wurde automatisch "refitted" mit ML,
# daher ML vergleichen
anova(REML, nullmodell)
anova(ML, nullmodell)

reml.LL <- as.numeric(2*(logLik(REML)-logLik(nullmodell)))
ml.LL <- as.numeric(2*(logLik(ML)-logLik(nullmodell)))

pchisq(reml.LL,1,lower=FALSE) # hochsignifikant auf 0.05 level
pchisq(ml.LL,1,lower=FALSE) # signifikant auf 0.05 level
```

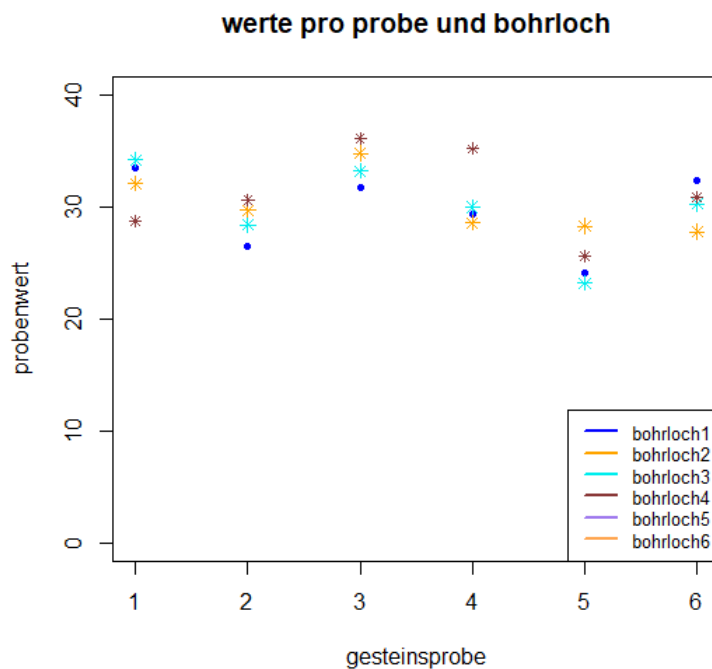
Aufgabe 2 (Zufallseffekte):

Um die Variation des Metallgehaltes in einem Erzlager zu untersuchen, werden aus I=6 Bohrlöchern jeweils J=5 Gesteinsproben entnommen und auf ihren Metallgehalt analysiert. Die Ergebnisse der Proben sind in folgender Tabelle dargestellt.

Probe	Bohrloch					
	1	2	3	4	5	6
1	33,5	26,5	31,7	29,4	24,1	32,4
2	29,4	31,3	35,4	36,7	26,7	38,6
3	32,1	29,7	34,8	28,6	28,3	27,8
4	34,2	28,4	33,2	30	23,2	30,3
5	28,7	30,6	36,1	35,2	25,6	30,9

Man analysiere die Daten mit einem Modell das das Bohrloch als Zufallseffekt auffasst und vergleiche die Ergebnisse mit jenen einer einfachen Varianzanalyse. Weiter bestimme man auch die Intraclass Korrelation.

Der folgende Plot veranschaulicht die Probenwerte je Bohrloch:

**Deskriptive Statistiken:**

```
> summary(metall_df) # summary pro bohrloch
```

bohrloch1	bohrloch2	bohrloch3	bohrloch4	bohrloch5	bohrloch6
Min. :28.70	Min. :26.5	Min. :31.70	Min. :28.60	Min. :23.20	Min. :27.8
1st Qu.:29.40	1st Qu.:28.4	1st Qu.:33.20	1st Qu.:29.40	1st Qu.:24.10	1st Qu.:30.3
Median :32.10	Median :29.7	Median :34.80	Median :30.00	Median :25.60	Median :30.9
Mean :31.58	Mean :29.3	Mean :34.24	Mean :31.98	Mean :25.58	Mean :32.0
3rd Qu.:33.50	3rd Qu.:30.6	3rd Qu.:35.40	3rd Qu.:35.20	3rd Qu.:26.70	3rd Qu.:32.4
Max. :34.20	Max. :31.3	Max. :36.10	Max. :36.70	Max. :28.30	Max. :38.6

```
> summary(t(metall_df)) # summary pro probe
      v1      v2      v3      v4      v5
Min.  :24.10  Min.  :26.70  Min.  :27.80  Min.  :23.20  Min.  :25.60
1st Qu.:27.23  1st Qu.:29.88  1st Qu.:28.38  1st Qu.:28.80  1st Qu.:29.18
Median :30.55  Median :33.35  Median :29.15  Median :30.15  Median :30.75
Mean   :29.60  Mean   :33.02  Mean   :30.22  Mean   :29.88  Mean   :31.18
3rd Qu.:32.23  3rd Qu.:36.38  3rd Qu.:31.50  3rd Qu.:32.48  3rd Qu.:34.12
Max.   :33.50  Max.   :38.60  Max.   :34.80  Max.   :34.20  Max.   :36.10

> apply(metall_df,2,sd) # sd pro bohrloch
bohrloch1 bohrloch2 bohrloch3 bohrloch4 bohrloch5 bohrloch6
 2.442744  1.903943  1.778482  3.696214  2.031502  4.045368

> apply(metall_df,1,sd) # sd pro probe
[1] 3.670422 4.611905 2.718394 3.916333 3.951413
```

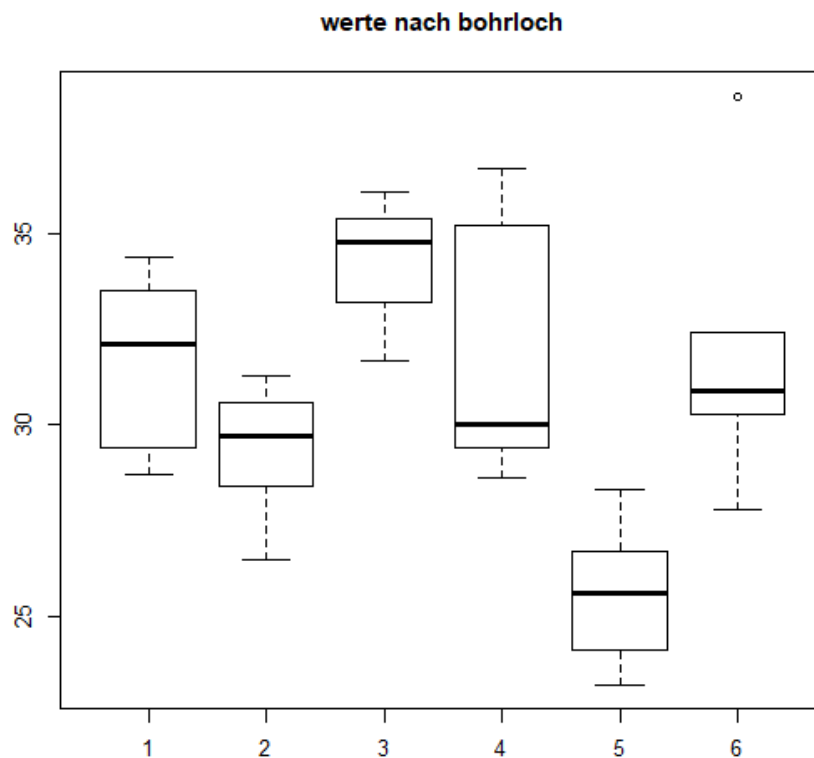
Mittelwerte pro Bohrloch:

1	2	3	4	5	6
31.62	29.30	34.24	31.98	25.58	32.00

Standardabweichung pro Bohrloch:

1	2	3	4	5	6
2.497399	1.903943	1.778482	3.696214	2.031502	4.045368

Boxplots pro Bohrloch:



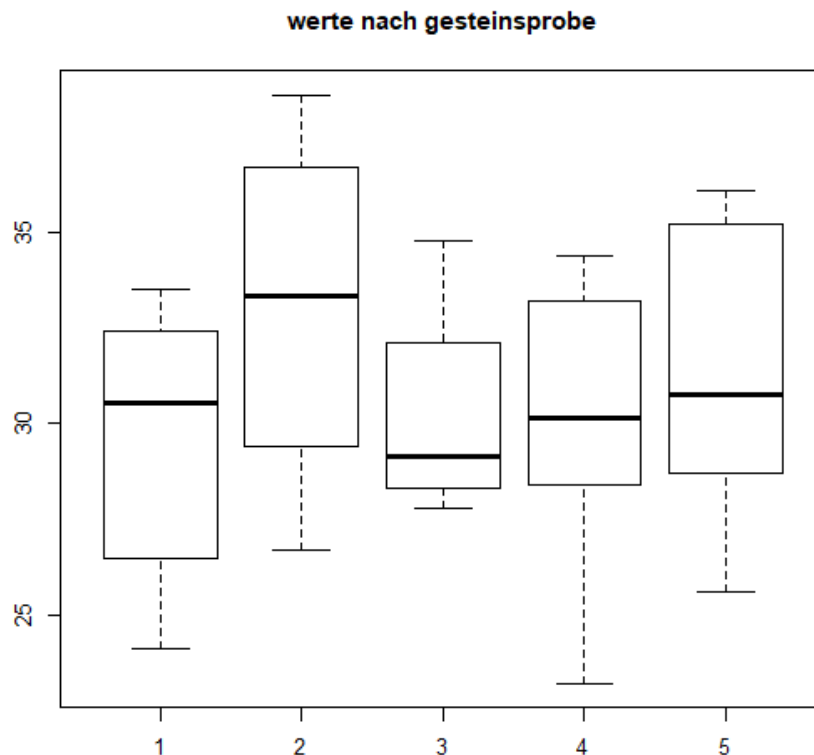
Mittelwerte pro Gesteinsprobe:

1	2	3	4	5
29.60000	33.01667	30.21667	29.91667	31.18333

Standardabweichung pro Gesteinsprobe:

1 2 3 4 5
3.670422 4.611905 2.718394 3.961018 3.951413

Boxplots pro Gesteinsprobe:



Modelle (Bohrloch als Zufallseffekt (*bohr.mod1*) vs. Einfache Varianzanalyse (*bohr.mod2*)):

EINFACHE VARIANZANALYSE:

Darin war der Test auf das Bohrloch signifikant, der Test auf die Gesteinsprobe jedoch nicht.

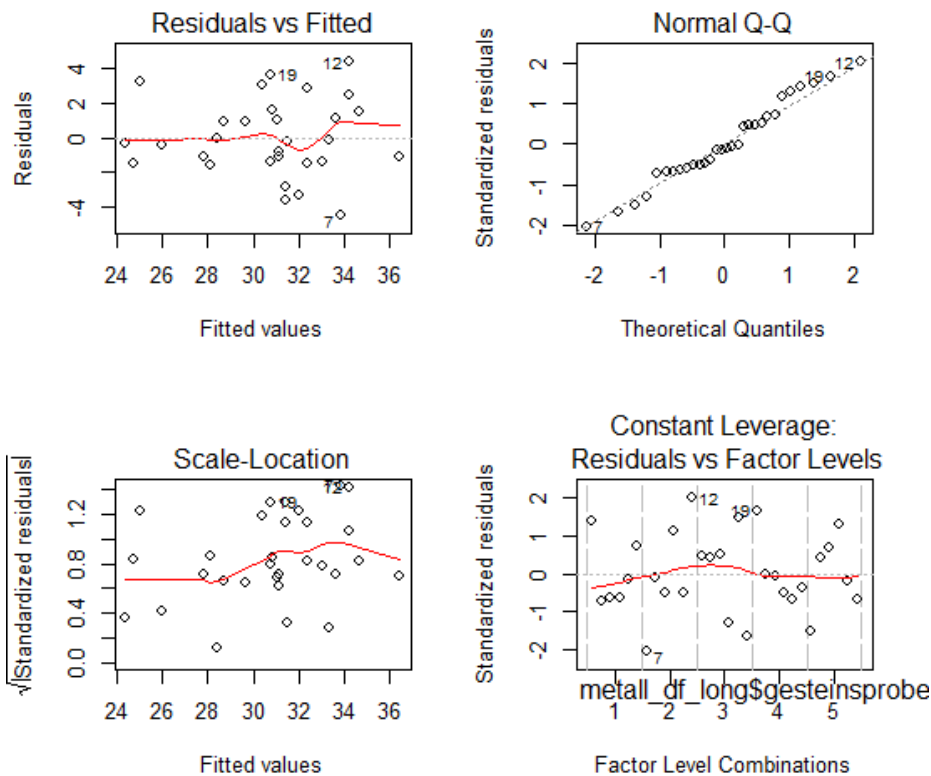
```
> bohr.mod1 <- aov(metall_df_long$messung ~ metall_df_long$gesteinsprobe
+                  metall_df_long$bohrloch)
> summary(bohr.mod1)
```

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
metall_df_long\$gesteinsprobe	4	45.72	11.43	1.599	0.21354
metall_df_long\$bohrloch	5	224.18	44.84	6.271	0.00118 **
Residuals	20	142.99	7.15		

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Diagnostic Plots für das Modell mit der einfachen Varianzanalyse:

Die Annahmen, dass der Erwartungswert der Fehler 0 ist und die Daten normalverteilt sind, konnten nicht verworfen werden, wie man aus den Plot „Residuals vs. Fitted“ und „Normal Q-Q“ sehen kann. Hinsichtlich Leverage sind die Punkte 12, 18 und 7 zu beachten, aber diese sind auch noch keine extremen Ausreißer außerhalb der Cook's Distance, sind also daher nicht als extrem bedrohlich einzustufen.



Bartlett-Test auf Varianzhomogenität:

Dieser ergab, dass die Varianzen in den Gruppen homogen sind, denn der p-Wert war nicht signifikant.

```
> bartlett.test(metall_df_long$messung ~ metall_df_long$bohrloch)
```

Bartlett test of homogeneity of variances

data: metall_df_long\$messung by metall_df_long\$bohrloch
Bartlett's K-squared = 4.6516, df = 5, p-value = 0.4599

BOHRLOCH ALS ZUFALLSEFFEKT:

```
> bohr.mod2 <- lmer(metall_df_long$messung ~ metall_df_long$gesteinsprobe +  
+                   (1|metall_df_long$bohrloch), REML = TRUE)
```

```
> summary(bohr.mod2)
```

Linear mixed model fit by REML ['lmerMod']

Formula: metall_df_long\$messung ~ metall_df_long\$gesteinsprobe + (1 | metall_df_long\$bohrloch)

REML criterion at convergence: 138.3

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.6145	-0.4784	-0.1845	0.6564	1.7067

Random effects:

Groups	Name	Variance	Std.Dev.
metall_df_long\$bohrloch	(Intercept)	7.537	2.745
Residual		7.150	2.674

Number of obs: 30, groups: metall_df_long\$bohrloch, 6

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	29.6000	1.5646	18.919
metall_df_long\$gesteinsprobe2	3.4167	1.5438	2.213
metall_df_long\$gesteinsprobe3	0.6167	1.5438	0.399
metall_df_long\$gesteinsprobe4	0.3167	1.5438	0.205
metall_df_long\$gesteinsprobe5	1.5833	1.5438	1.026

Correlation of Fixed Effects:

	(Intr)	mt__\$2	mt__\$3	mt__\$4
mtll_df_l\$2	-0.493			
mtll_df_l\$3	-0.493	0.500		
mtll_df_l\$4	-0.493	0.500	0.500	
mtll_df_l\$5	-0.493	0.500	0.500	0.500

Zusätzlich wurde die **Intra-Class-Korrelation** aus dem Modell mit Zufallseffekten berechnet, wobei sie bei ca. 0.51 liegt:

```
> sigma.a <- 7.537
> sigma <- 7.150
> intraclass_correlation_bohr <- sigma.a / (sigma.a + sigma)
> intraclass_correlation_bohr
[1] 0.5131749
```

Modellvergleich:

Koeffizientenschätzung (basierend auf den Ergebnissen der ANOVA):

```
> coef(bohr.mod1)
              (Intercept) metall_df_long$gesteinsprobe2
              30.4333333              3.4166667
metall_df_long$gesteinsprobe3 metall_df_long$gesteinsprobe4
              0.6166667              0.3166667
metall_df_long$gesteinsprobe5 metall_df_long$bohrloch2
              1.5833333              -2.3200000
              metall_df_long$bohrloch3 metall_df_long$bohrloch4
              2.6200000              0.3600000
              metall_df_long$bohrloch5 metall_df_long$bohrloch6
              -6.0400000              0.3800000
```

Schätzung der Zufallseffekte (basierend auf bohrloch als Zufallsvariable):

```
> ranef(bohr.mod2)
$`metall_df_long$bohrloch`
  (Intercept)
1  0.7004458
2 -1.2495954
3  2.9026476
4  1.0030384
5 -4.3763856
6  1.0198491
with conditional variances for "metall_df_long$bohrloch"
```

Schätzung der festen Effekte:

```
> fixef(bohr.mod2) # basierend auf bohrloch als ZV
      (Intercept) metall_df_long$gesteinsprobe2
      29.6000000      3.4166667
metall_df_long$gesteinsprobe3 metall_df_long$gesteinsprobe4
      0.6166667      0.3166667
metall_df_long$gesteinsprobe5
      1.5833333
```

Schätzung der Koeffizienten:

```
> coef(bohr.mod2) # basierend auf bohrloch als ZV
$`metall_df_long$bohrloch`
      (Intercept) metall_df_long$gesteinsprobe2 metall_df_long$gesteinsprobe3
1      30.30045      3.416667      0.6166667
2      28.35040      3.416667      0.6166667
3      32.50265      3.416667      0.6166667
4      30.60304      3.416667      0.6166667
5      25.22361      3.416667      0.6166667
6      30.61985      3.416667      0.6166667
      metall_df_long$gesteinsprobe4 metall_df_long$gesteinsprobe5
1      0.3166667      1.583333
2      0.3166667      1.583333
3      0.3166667      1.583333
4      0.3166667      1.583333
5      0.3166667      1.583333
6      0.3166667      1.583333

attr(,"class")
[1] "coef.mer"
```

Testen der festen Effekte:

Das *Nullmodell* (bestehend aus nur einer Konstante als Fixed Term zur Erklärung der abhängigen Variable) wird mit dem Alternativmodell verglichen. Das Alternativmodell enthält abgesehen vom Zufallseffektteil die fixe Komponente der gesteinsprobe.

```
> summary(bohr.nullmodell)
Linear mixed model fit by maximum likelihood ['lmerMod']
Formula: metall_df_long$messung ~ 1 + (1 | metall_df_long$bohrloch)
```

AIC	BIC	logLik	deviance	df.resid
162.4	166.6	-78.2	156.4	27

Scaled residuals:

Min	1Q	Median	3Q	Max
-1.40673	-0.70852	-0.05153	0.59605	2.44473

Random effects:

Groups	Name	Variance	Std.Dev.
metall_df_long\$bohrloch	(Intercept)	5.900	2.429
Residual		7.863	2.804

Number of obs: 30, groups: metall_df_long\$bohrloch, 6

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	30.787	1.116	27.59

Das *Alternativmodell* ist ein Mixed Model mit festem Effekt auf die Gesteinsprobe (anstatt der Konstante im Nullmodell), und der Zufallseffekt ist ebenso das Bohrloch:

```
> summary(bohr.alternativmodell)
Linear mixed model fit by maximum likelihood ['lmerMod']
Formula: metall_df_long$messung ~ metall_df_long$gesteinsprobe + (1 |
    metall_df_long$bohrloch)

      AIC      BIC    logLik deviance df.resid
    163.7    173.5     -74.8    149.7      23

Scaled residuals:
    Min       1Q   Median       3Q      Max
-1.7686 -0.5240 -0.2021  0.7191  1.8696

Random effects:
 Groups                Name      Variance Std.Dev.
metall_df_long$bohrloch (Intercept)  6.281    2.506
Residual                        5.958    2.441
Number of obs: 30, groups: metall_df_long$bohrloch, 6

Fixed effects:
              Estimate Std. Error t value
(Intercept)    29.6000     1.4282  20.725
metall_df_long$gesteinsprobe2    3.4167     1.4093   2.424
metall_df_long$gesteinsprobe3    0.6167     1.4093   0.438
metall_df_long$gesteinsprobe4    0.3167     1.4093   0.225
metall_df_long$gesteinsprobe5    1.5833     1.4093   1.124

Correlation of Fixed Effects:
      (Intr) mt__$2 mt__$3 mt__$4
mtll_df_l$2 -0.493
mtll_df_l$3 -0.493  0.500
mtll_df_l$4 -0.493  0.500  0.500
mtll_df_l$5 -0.493  0.500  0.500  0.500
```

Modellvergleich mittels ANOVA:

Das jeweilige Alternativmodell wird mittels ANOVA mit dem Nullmodell verglichen, einerseits für den festen Effekt der Gesteinsprobe, andererseits für den Zufallseffekt des Bohrlochs. Beim Vergleich in Bezug auf den festen Effekt stellte sich heraus, dass die Gesteinsprobe nicht signifikant war. Der Test auf den Zufallseffekt des Bohrlochs war hingegen signifikant.

```
> anova(bohr.alternativmodell,bohr.nullmodell)
Data: NULL
Models:
bohr.nullmodell: metall_df_long$messung ~ 1 + (1 | metall_df_long$bohrloch)
bohr.alternativmodell: metall_df_long$messung ~ metall_df_long$gesteinsprobe + (1 |
bohr.alternativmodell: metall_df_long$bohrloch)
              Df    AIC    BIC  logLik deviance Chisq Chi Df Pr(>Chisq)
bohr.nullmodell    3 162.35 166.56 -78.176    156.35      NA   NA
bohr.alternativmodell 7 163.69 173.50 -74.847    149.69    6.65   4    0.1551
```

(ergebnis: bohrloch-effekt ist signifikant)

```
> anova(bohr.modellMitZufall,bohr.modellNull)
refitting model(s) with ML (instead of REML)
Data: NULL
Models:
bohr.modellNull: metall_df_long$messung ~ 1 + metall_df_long$gesteinsprobe
bohr.modellMitZufall: metall_df_long$messung ~ metall_df_long$gesteinsprobe
+ (1 | metall_df_long$bohrloch)

          Df      AIC      BIC  logLik deviance  Chisq Chi Df Pr(>Chisq)
bohr.modellNull      6 172.28 180.68 -80.138   160.28
bohr.modellMitZufall  7 163.69 173.50 -74.847   149.69 10.581      1 0.001143

bohr.modellNull
bohr.modellMitZufall **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Direkter Vergleich der Log-Likelihood für die Zufallseffekte ergibt dasselbe Ergebnis und dieselbe Folgerung, wie zuvor beschrieben:

```
> LL <- as.numeric(2*(logLik(bohr.modellMitZufall) - logLik(bohr.modellNull, REML = TRUE)))
> pchisq(LL, 1, lower.tail = FALSE)
[1] 0.002983671
```

R-Code zu Aufgabe 2 (Zufallseffekte):

```

#####
# AUFGABE 2 (Zufallseffekte)
#####
# Um die Variation des Metallgehaltes in einem Erzlager zu untersuchen, werden aus I=6
# Bohrlöchern jeweils J=5 Gesteinsproben entnommen und auf ihren Metallgehalt analysiert.
# Die Ergebnisse der Proben sind in folgender Tabelle dargestellt.
# Man analysiere die Daten mit einem Modell das das Bohrloch als Zufallseffekt auffasst
# und vergleiche die Ergebnisse mit jenen einer einfachen Varianzanalyse.
# Weiter bestimme man auch die Intraclass Korrelation.

# balanciertes design (weil jeweils 5 gesteinsproben pro bohrloch)
metall_df <- data.frame(bohrloch1=c(33.5,29.4,32.1,34.2,28.7),
                        bohrloch2=c(26.5,31.3,29.7,28.4,30.6),
                        bohrloch3=c(31.7,35.4,34.8,33.2,36.1),
                        bohrloch4=c(29.4,36.7,28.6,30.0,35.2),
                        bohrloch5=c(24.1,26.7,28.3,23.2,25.6),
                        bohrloch6=c(32.4,38.6,27.8,30.3,30.9))

metall_df

# plot: werte pro gesteinsprobe und bohrloch
plot(as.numeric(metall_df[,1]), xlim=c(1,6), ylim=c(0,40),
     ylab="probenwert", xlab="gesteinsprobe", col="blue",
     pch=20, main="werte pro probe und bohrloch")
points(as.numeric(metall_df[,3]), col="orange", pch=8)
points(as.numeric(metall_df[,4]), col="cyan2", pch=8)
points(as.numeric(metall_df[,5]), col="indianred4", pch=8)
points(as.numeric(metall_df[,6]), col="mediumpurple2", pch=8)
points(as.numeric(metall_df[,6]), col="tan1", pch=8)
par(cex=0.8)
legend("bottomright", legend=c("bohrloch1", "bohrloch2",
                              "bohrloch3", "bohrloch4", "bohrloch5", "bohrloch6"),
      col=c("blue", "orange", "cyan2", "indianred4",
            "mediumpurple2", "tan1"), lwd=2)

# deskriptive statistiken
summary(metall_df) # summary pro bohrloch
summary(t(metall_df)) # summary pro probe
apply(metall_df,2,sd) # sd pro bohrloch
apply(metall_df,1,sd) # sd pro probe

bohrloch <- factor(rep(1:6,5))
gesteinsprobe <- factor(rep(1:5,each=6))
messungen <- c(33.5, 26.5, 31.7, 29.4, 24.1, 32.4, # probe 1
              29.4, 31.3, 35.4, 36.7, 26.7, 38.6, # probe 2
              32.1, 29.7, 34.8, 28.6, 28.3, 27.8, # probe 3
              34.4, 28.4, 33.2, 30.0, 23.2, 30.3, # probe 4
              28.7, 30.6, 36.1, 35.2, 25.6, 30.9) # probe 5
metall_df_long <- data.frame(gesteinsprobe=gesteinsprobe,
                           bohrloch=bohrloch,
                           messung=messungen)

metall_df_long

# means pro bohrloch
tapply(metall_df_long$messung, metall_df_long$bohrloch, mean)

# standardabweichung pro bohrloch
tapply(metall_df_long$messung, metall_df_long$bohrloch, sd)

# boxplot: werte nach bohrloch
boxplot(metall_df_long$messung ~ metall_df_long$bohrloch,
       main="werte nach bohrloch")

# means pro probe
tapply(metall_df_long$messung, metall_df_long$gesteinsprobe, mean)

# standardabweichung pro probe
tapply(metall_df_long$messung, metall_df_long$gesteinsprobe, sd)

```

```
# boxplot: werte nach probe
boxplot(metall_df_long$messung ~ metall_df_long$gesteinsprobe,
        main="werte nach gesteinsprobe")

# modelle: bohrloch als zufallseffekt (bohr.mod1) vs.
#         einfache varianzanalyse (bohr.mod2)

# EINFACHE VARIANZANALYSE:
bohr.mod1 <- aov(metall_df_long$messung ~ metall_df_long$gesteinsprobe +
                metall_df_long$bohrloch)

summary(bohr.mod1)

# standard diagnostic plots fuer das modell:
# (ergebnis: E[u_t]=0, NV daten)
par(mfrow=c(2,2))
plot(bohr.mod1)

# bartlett-test auf varianzhomogenitaet
# (ergebnis: varianzen sind homogen)
bartlett.test(metall_df_long$messung ~ metall_df_long$bohrloch)

# BOHRLOCH ALS ZUFALLSEFFEKT:

bohr.mod2 <- lmer(metall_df_long$messung ~ metall_df_long$gesteinsprobe +
                 (1|metall_df_long$bohrloch), REML = TRUE)

summary(bohr.mod2)

# INTRA-CLASS-CORRELATION:
sigma.a <- 7.537
sigma <- 7.150
intraclass_correlation_bohr <- sigma.a / (sigma.a + sigma)

# MODELLVERGLEICH/-BEURTEILUNG
coef(bohr.mod1) # basierend auf anova
ranef(bohr.mod2) # basierend auf bohrloch als ZV
fixef(bohr.mod2) # basierend auf bohrloch als ZV
coef(bohr.mod2) # basierend auf bohrloch als ZV

# testen der festen effekte
# nullmod. und alternativmod. mit ML-methode vergleichen
# nullmodell (nur konstante als fixed term)
bohr.nullmodell <- lmer(metall_df_long$messung ~ 1 + (1|metall_df_long$bohrloch),
                       REML=FALSE)

summary(bohr.nullmodell)
# alternativmodell (mixed model mit fixed effect "gesteinsprobe")
bohr.alternativmodell <- lmer(metall_df_long$messung ~ metall_df_long$gesteinsprobe +
                             (1|metall_df_long$bohrloch), REML=FALSE)

summary(bohr.alternativmodell)

# vgl. der beiden modelle mit anova
anova(bohr.alternativmodell,bohr.nullmodell)

# test, ob bohrloch signifikant
# (ergebnis: bohrloch-effekt ist signifikant)
bohr.modellMitZufall <-lmer(metall_df_long$messung ~ metall_df_long$gesteinsprobe +
                          (1|metall_df_long$bohrloch))

bohr.modellNull <-lm(metall_df_long$messung ~ 1 + metall_df_long$gesteinsprobe)
anova(bohr.modellMitZufall,bohr.modellNull)
# direkt likelihood vergleichen
LL <- as.numeric(2*(logLik(bohr.modellMitZufall) - logLik(bohr.modellNull, REML = TRUE)))
pchisq(LL, 1, lower.tail = FALSE)
```

Aufgabe 3 (Longitudinaldaten):

Die Daten „phosphate“ im Package HSAUR enthalten die anorganischen Phosphatlevels im Plasma von 33 Personen, von denen 20 zu einer Kontrollgruppe gehören und 13 als adipös klassifiziert sind. Stelle die Profile der Personen in den einzelnen Gruppen dar und erstelle ein geeignetes Modell mit Zufallseffekten für den Verlauf der Phosphatwerte.

Überblick über die Daten:

```
> head(phosphate, n=11)
  group t0 t0.5 t1 t1.5 t2 t3 t4 t5
1 control 4.3 3.3 3.0 2.6 2.2 2.5 3.4 4.4
2 control 3.7 2.6 2.6 1.9 2.9 3.2 3.1 3.9
3 control 4.0 4.1 3.1 2.3 2.9 3.1 3.9 4.0
4 control 3.6 3.0 2.2 2.8 2.9 3.9 3.8 4.0
5 control 4.1 3.8 2.1 3.0 3.6 3.4 3.6 3.7
```

Deskriptive Statistiken (gesamt):

```
> summary(phosphate[])
  group      t0      t0.5      t1      t1.5      t2
control:20  Min.   :3.000  Min.   :2.200  Min.   :2.000  Min.   :1.900  Min.   :1.500
obese :13    1st Qu.:3.800  1st Qu.:3.100  1st Qu.:2.900  1st Qu.:2.600  1st Qu.:2.800
          Median :4.400  Median :3.900  Median :3.200  Median :3.100  Median :3.200
          Mean   :4.358  Mean   :3.794  Mean   :3.364  Mean   :3.145  Mean   :3.136
          3rd Qu.:4.800  3rd Qu.:4.300  3rd Qu.:3.900  3rd Qu.:3.700  3rd Qu.:3.700
          Max.   :6.600  Max.   :6.100  Max.   :5.200  Max.   :4.600  Max.   :4.700

      t3      t4      t5
Min.   :1.900  Min.   :2.300  Min.   :2.700
1st Qu.:3.100  1st Qu.:3.200  1st Qu.:3.700
Median :3.600  Median :3.600  Median :3.900
Mean   :3.364  Mean   :3.624  Mean   :3.985
3rd Qu.:3.900  3rd Qu.:4.000  3rd Qu.:4.300
Max.   :4.300  Max.   :4.800  Max.   :5.000
```

Profile der Personen in den einzelnen Gruppen:

```
> summary(phosphate[group=="control",])
  group      t0      t0.5      t1      t1.5      t2
control:20  Min.   :3.10  Min.   :2.200  Min.   :2.000  Min.   :1.900  Min.   :1.500
obese : 0    1st Qu.:3.70  1st Qu.:3.075  1st Qu.:2.600  1st Qu.:2.450  1st Qu.:2.825
          Median :4.20  Median :3.300  Median :3.000  Median :2.700  Median :3.000
          Mean   :4.15  Mean   :3.505  Mean   :2.950  Mean   :2.785  Mean   :3.000
          3rd Qu.:4.45  3rd Qu.:3.925  3rd Qu.:3.225  3rd Qu.:3.125  3rd Qu.:3.600
          Max.   :5.00  Max.   :5.000  Max.   :4.100  Max.   :3.900  Max.   :3.800

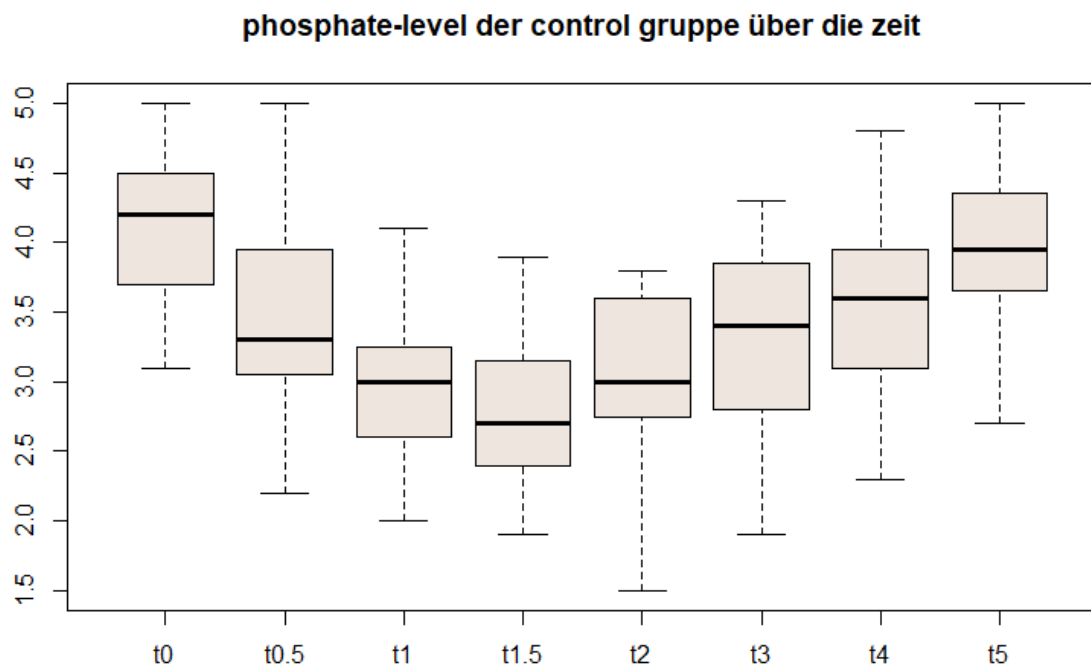
      t3      t4      t5
Min.   :1.900  Min.   :2.300  Min.   :2.700
1st Qu.:2.950  1st Qu.:3.100  1st Qu.:3.675
Median :3.400  Median :3.600  Median :3.950
Mean   :3.300  Mean   :3.565  Mean   :3.960
3rd Qu.:3.825  3rd Qu.:3.925  3rd Qu.:4.325
Max.   :4.300  Max.   :4.800  Max.   :5.000
```

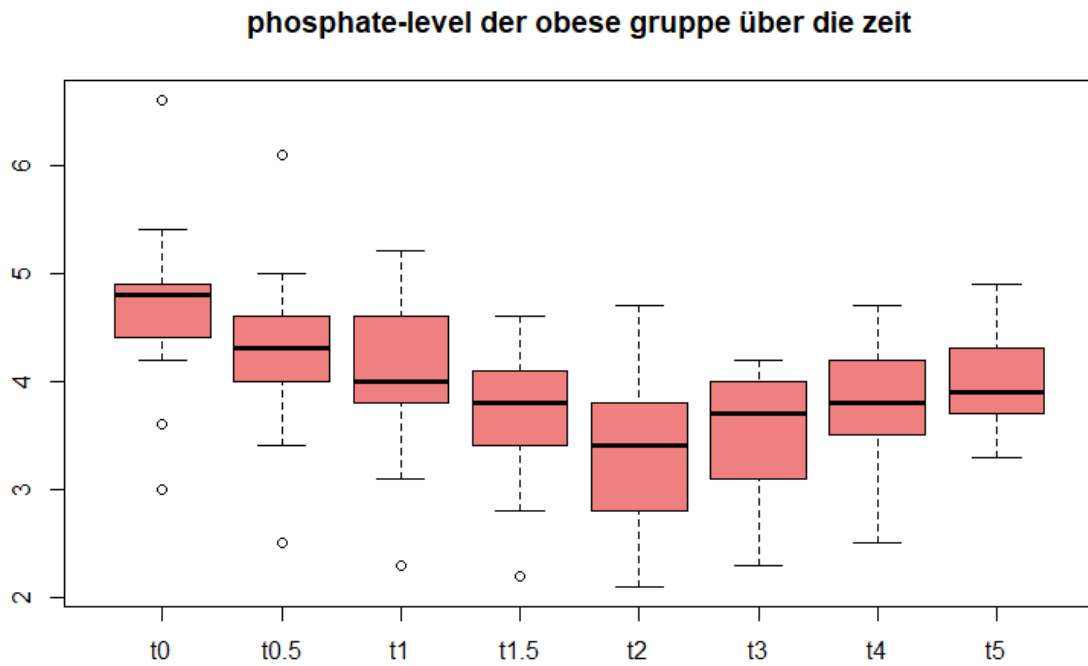
```
> summary(phosphate[group=="obese",])
  group      t0      t0.5      t1      t1.5      t2
control: 0  Min.   :3.000  Min.   :2.500  Min.   :2.3  Min.   :2.2  Min.   :2.100
obese   :13  1st Qu.:4.400  1st Qu.:4.000  1st Qu.:3.8  1st Qu.:3.4  1st Qu.:2.800
          Median :4.800  Median :4.300  Median :4.0  Median :3.8  Median :3.400
          Mean   :4.677  Mean   :4.238  Mean   :4.0  Mean   :3.7  Mean   :3.346
          3rd Qu.:4.900  3rd Qu.:4.600  3rd Qu.:4.6  3rd Qu.:4.1  3rd Qu.:3.800
          Max.   :6.600  Max.   :6.100  Max.   :5.2  Max.   :4.6  Max.   :4.700

      t3      t4      t5
Min.   :2.300  Min.   :2.500  Min.   :3.300
1st Qu.:3.100  1st Qu.:3.500  1st Qu.:3.700
Median :3.700  Median :3.800  Median :3.900
Mean   :3.462  Mean   :3.715  Mean   :4.023
3rd Qu.:4.000  3rd Qu.:4.200  3rd Qu.:4.300
Max.   :4.200  Max.   :4.700  Max.   :4.900

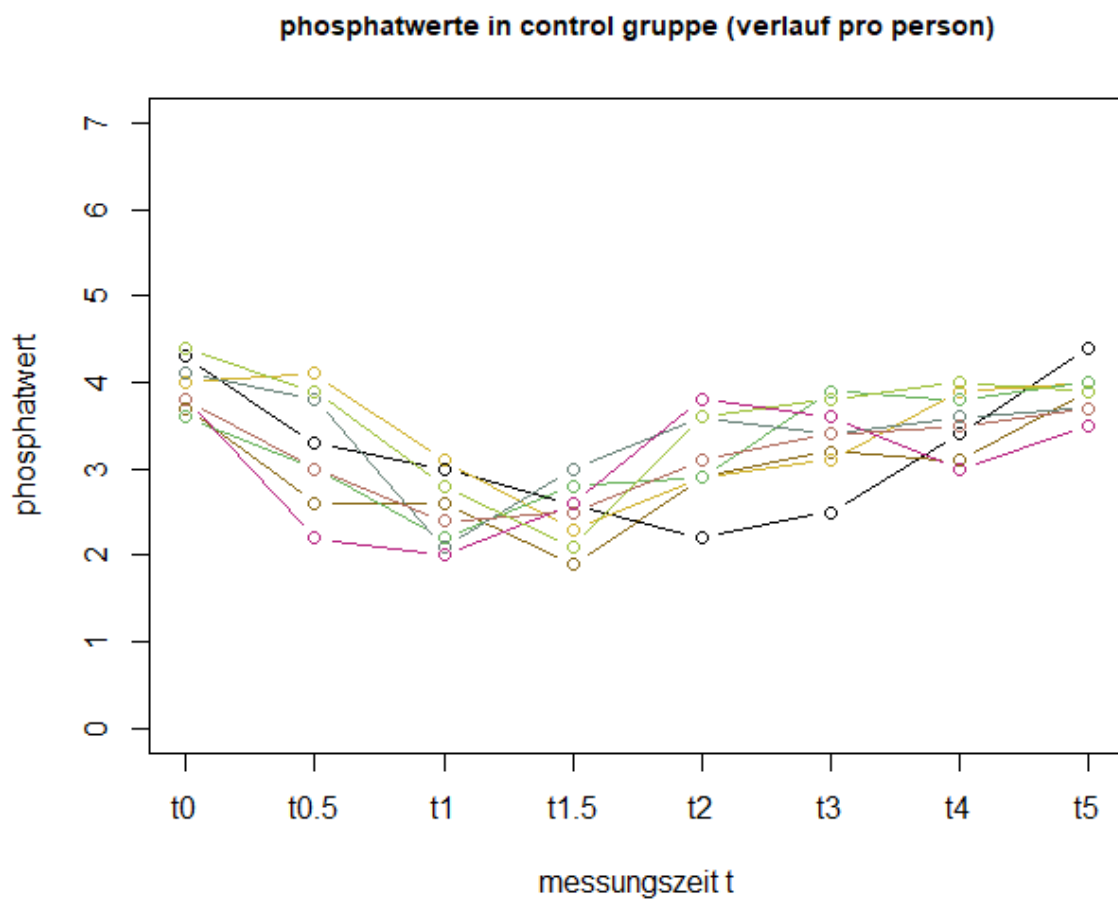
> apply(phosphate[-1],2,sd) # sd pro zeitpunkt
      t0      t0.5      t1      t1.5      t2      t3      t4      t5
0.7088857 0.8591200 0.8069302 0.7314758 0.7131875 0.6891200 0.6339511 0.5750659
> apply(phosphate_data[group=="control",],2,sd)
      t0      t0.5      t1      t1.5      t2      t3      t4      t5
0.5155477 0.7472581 0.5414406 0.5294237 0.6223893 0.7159829 0.6698586 0.6081897
> apply(phosphate_data[group=="obese",],2,sd)
      t0      t0.5      t1      t1.5      t2      t3      t4      t5
0.8584422 0.8549254 0.7404953 0.6582806 0.8150822 0.6614863 0.5885663 0.5418392
```

Verlauf der Phosphatwerte pro Gruppe über die Zeit:

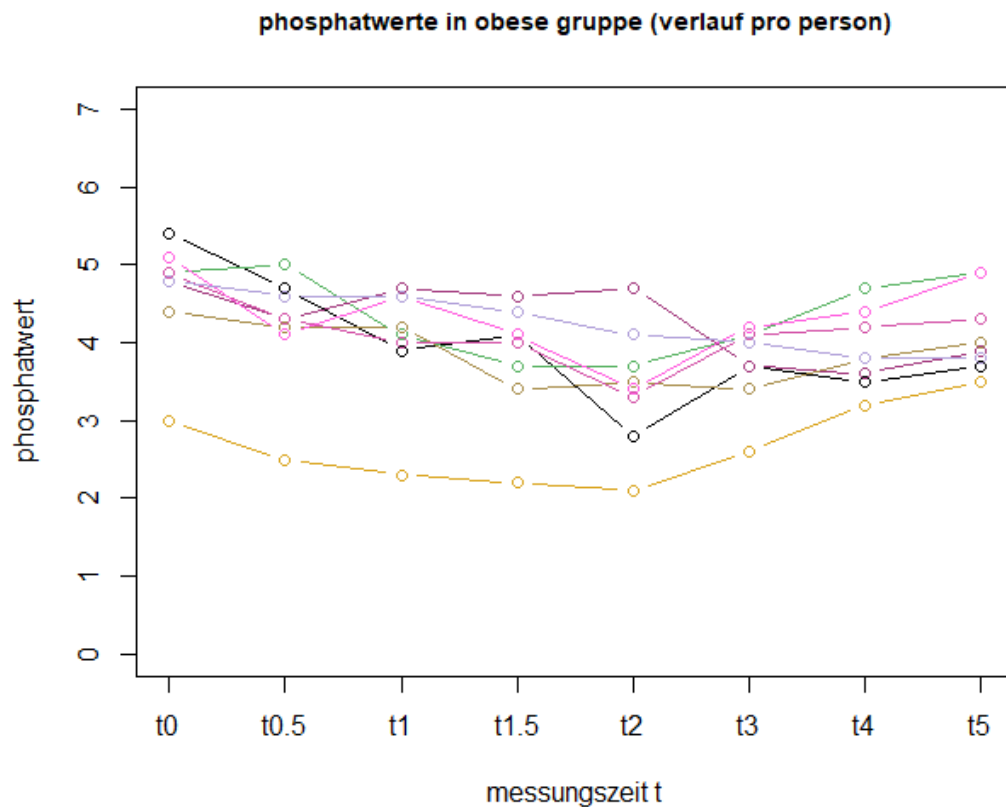




Profile der einzelnen Personen in der *Control*-Gruppe (über die Zeit):



Profile der einzelnen Personen in der *Obese*-Gruppe (über die Zeit):



Modell mit Zufallseffekten für den Verlauf der Phosphatwerte:

(mit Library `furniture`, <https://cran.r-project.org/web/packages/furniture/furniture.pdf>)

Der Datensatz wurde mittels der Funktion `long()` der Library `furniture` umorganisiert, sodass die Daten in der Langform vorliegen. Der Ausschnitt zeigt einen Teil der Daten für subject 1-3:

```
> subset(long_df, subject %in% c("1","2","3"))
```

	group	time	value	subject
1.1	control	0.0	4.3	1
2.1	control	0.0	3.7	2
3.1	control	0.0	4.0	3
1.2	control	0.5	3.3	1
2.2	control	0.5	2.6	2
3.2	control	0.5	4.1	3
1.3	control	1.0	3.0	1
2.3	control	1.0	2.6	2
3.3	control	1.0	3.1	3
1.4	control	1.5	2.6	1
2.4	control	1.5	1.9	2
3.4	control	1.5	2.3	3

Modell 1:

```
Linear mixed model fit by maximum likelihood ['lmerMod']
Formula: long_df$value ~ long_df$time + long_df$group + (1 | subject)
Data: long_df
```

AIC	BIC	logLik	deviance	df.resid
562.5	580.4	-276.3	552.5	259

```
Scaled residuals:
      Min       1Q   Median       3Q      Max
-2.33724 -0.68569 -0.08461  0.73489  2.99361
```

```
Random effects:
Groups   Name             Variance Std.Dev.
subject  (Intercept)  0.2227   0.4719
Residual                0.3822   0.6183
Number of obs: 264, groups:  subject, 33
```

```
Fixed effects:
              Estimate Std. Error t value
(Intercept)    3.43885    0.12638  27.211
long_df$time   -0.01740    0.02328  -0.748
long_df$groupobese  0.49332    0.18528   2.662
```

```
Correlation of Fixed Effects:
          (Intr) lng_df$t
long_df$tim -0.391
lng_df$grpb -0.578  0.000
```

Modell 2:

```
Linear mixed model fit by maximum likelihood ['lmerMod']
Formula: long_df$value ~ long_df$time + long_df$group + (time | subject)
Data: long_df
```

AIC	BIC	logLik	deviance	df.resid
564.6	589.7	-275.3	550.6	257

```
Scaled residuals:
      Min       1Q   Median       3Q      Max
-2.37973 -0.68784 -0.02457  0.70339  2.61858
```

```
Random effects:
Groups   Name             Variance Std.Dev. Corr
subject  (Intercept)  0.292107 0.54047
          time         0.006452 0.08032 -0.52
Residual                0.362536 0.60211
Number of obs: 264, groups:  subject, 33
```

```
Fixed effects:
              Estimate Std. Error t value
(Intercept)    3.46282    0.13352  25.935
long_df$time   -0.01740    0.02664  -0.653
long_df$groupobese  0.43249    0.18448   2.344
```

```
Correlation of Fixed Effects:
```

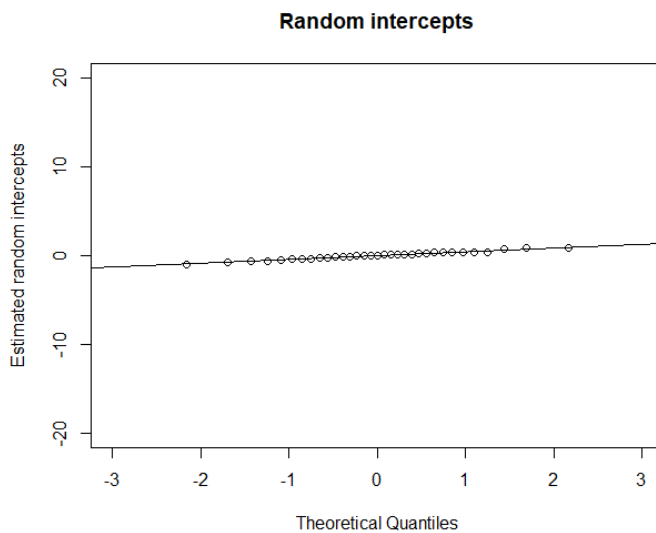
```
(Intr) lng_df$t
long_df$tim -0.498
lng_df$grpb -0.544  0.000
```

ANOVA zum Modellvergleich:

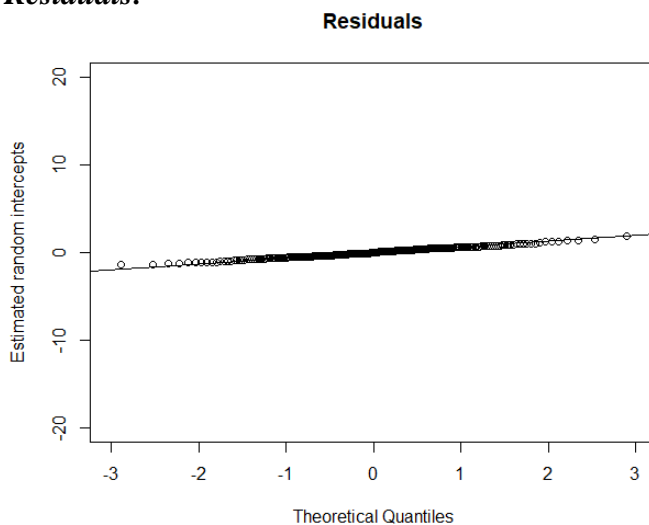
```
> anova(phosphate.mod1, phosphate.mod2)
Data: long_df
Models:
phosphate.mod1: long_df$value ~ long_df$time + long_df$group + (1 | subject)
phosphate.mod2: long_df$value ~ long_df$time + long_df$group + (time | subject)
      Df    AIC    BIC logLik deviance  Chisq Chi Df Pr(>Chisq)
phosphate.mod1  5 562.51 580.39 -276.26   552.51
phosphate.mod2  7 564.64 589.67 -275.32   550.64  1.8724      2    0.3921
```

In den beiden Modellen wurde der Unterschied durch den Effekt des zeitlichen Verlaufs der Phosphatmesswerte verglichen. Der Vergleich von Modell 1 und Modell2 ergibt, dass der p-Wert des Tests nicht signifikant ist.

Random Effect Prediction:



Residuals:



R-Code zu Aufgabe 3 (Longitudinaldaten):

```
#####
# AUFGABE 3 (Longitudinaldaten)
#####
# Die Daten "phosphate" im Package HSAUR enthalten die anorganischen Phosphat-levels im
# Plasma von 33 Personen, von denen 20 zu einer Kontrollgruppe gehören und 13 als adipös
# klassifiziert sind. Stelle die Profile der Personen in den einzelnen Gruppen dar und
# erstelle ein geeignetes Modell mit Zufallseffekten für den Verlauf der Phosphatwerte.

phosphate
attach(phosphate)
head(phosphate, n=11)

# profile der personen in den einzelnen gruppen
summary(phosphate[])
summary(phosphate[group=="control",])
summary(phosphate[group=="obese",])

apply(phosphate[-1],2,sd) # sd pro zeitpunkt
phosphate_data <- phosphate[-1]
apply(phosphate_data[group=="control",],2,sd)
apply(phosphate_data[group=="obese",],2,sd)

# verlauf pro gruppe
boxplot(phosphate_data[group=="control",], col="seashell2",
        main="phosphate-level der control gruppe über die zeit")
boxplot(phosphate_data[group=="obese",], col="lightcoral",
        main="phosphate-level der obese gruppe über die zeit")

control <- phosphate_data[group=="control",]
obese <- phosphate_data[group=="obese",]

# profile der personen in control gruppe
colfunc <- colorRampPalette(c("black", "goldenrod2", "mediumseagreen",
                             "mediumvioletred", "olivedrab3", "magenta1",
                             "rosybrown1", "royalblue1", "turquoise3",
                             "steelblue1", "tomato1", "paleturquoise1"))
plot(rep(1,10),col=colfunc(20),pch=19,cex=3) # test colorRampPalette look
plot(as.numeric(control[1,]), type="b", xaxt = "n",
     xlab="messungszeit t", ylim=c(0,7),
     ylab="phosphatwert",
     main="phosphatwerte in control gruppe (verlauf pro person)",
     cex.main=0.9)
axis(1, at=1:8, labels=colnames(control))
for (i in 2:length(control)){
  lines(as.numeric(control[i,]), type="b", col=colfunc(nrow(control))[i])
}

# profile der personen in obese gruppe
plot(as.numeric(obese[1,]), type="b", xaxt = "n",
     xlab="messungszeit t", ylim=c(0,7),
     ylab="phosphatwert",
     main="phosphatwerte in obese gruppe (verlauf pro person)",
     cex.main=0.9)
axis(1, at=1:8, labels=colnames(obese))
for (i in 2:length(obese)){
  lines(as.numeric(obese[i,]), type="b", col=colfunc(nrow(obese))[i])
}
```

```
# modell mit zufallseffekten für den verlauf der phosphatwerte
# mit library furniture:
# https://cran.r-project.org/web/packages/furniture/furniture.pdf
phosphate_baseData <- phosphate
long_df <- long(phosphate_baseData,
               c("t0", "t0.5", "t1", "t1.5", "t2", "t3", "t4", "t5"),
               v.names = "value")
nobs <- nrow(phosphate_baseData)
long_df$time <- rep(c(0, 0.5, 1, 1.5, 2, 3, 4, 5), rep(nobs, 8))
colnames(long_df)[4] <- "subject"
long_df
subset(long_df, subject %in% c("1", "2", "3")) # test subject 1-3

# modell 1
phosphate.mod1 <- lmer(long_df$value ~ long_df$time + long_df$group + (1|subject),
                      data=long_df, REML=FALSE, na.action=na.omit)
summary(phosphate.mod1)

# modell 2
phosphate.mod2 <- lmer(long_df$value ~ long_df$time + long_df$group + (time|subject),
                      data=long_df, REML=FALSE, na.action=na.omit)
summary(phosphate.mod2)

# anova (zum modellvergleich)
anova(phosphate.mod1, phosphate.mod2)

# random effects prediction
qint<-ranef(phosphate.mod1)$subject[["(Intercept)"]]
qres<-residuals(phosphate.mod1)
qqnorm(qint, ylab="Estimated random intercepts",
       xlim=c(-3, 3), ylim=c(-20, 20), main = "Random intercepts")
qqline(qint)

qqnorm(qres, ylab="Estimated random intercepts",
       xlim=c(-3, 3), ylim=c(-20, 20), main = "Residuals")
qqline(qres)
```

Aufgabe 4 (Longitudinaldaten):

Die Daten „ratdrink“ im Package faraway enthalten für 27 Ratten die Gewichtsmessung in fünf aufeinander folgenden Wochen. Die ersten zehn Ratten sind gehören zur Kontrollgruppe, bei sieben Ratten wurden Thyroxin in das Trinkwasser hinzugefügt und bei 10 Ratten wurde Thiouracil hinzugefügt. Stelle die Daten in geeigneter Form dar und bestimme ein Modell für den zeitlichen Verlauf.

Überblick über die einen Ausschnitt des Datensatz und *deskriptive Statistiken*:

```
> head(ratdrink,n=10)
```

	wt	weeks	subject	treat
1	57	0	1	control
2	86	1	1	control
3	114	2	1	control
4	139	3	1	control
5	172	4	1	control
6	60	0	2	control
7	93	1	2	control
8	123	2	2	control
9	146	3	2	control
10	177	4	2	control

Die folgenden Zahlen zeigen die Mittelwerte pro Subjekt (subject), pro Behandlung (treat) und pro Woche (week).

```
> means_per_subject
```

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
113.6	119.8	113.8	101.8	102.6	100.4	93.4	110.0	91.6	111.4	118.4	92.6	115.8	117.0	98.0
16	17	18	19	20	21	22	23	24	25	26	27			
95.6	107.2	101.0	94.6	94.2	96.0	98.0	87.4	88.0	95.2	76.4	88.0			

```
> means_per_treat
```

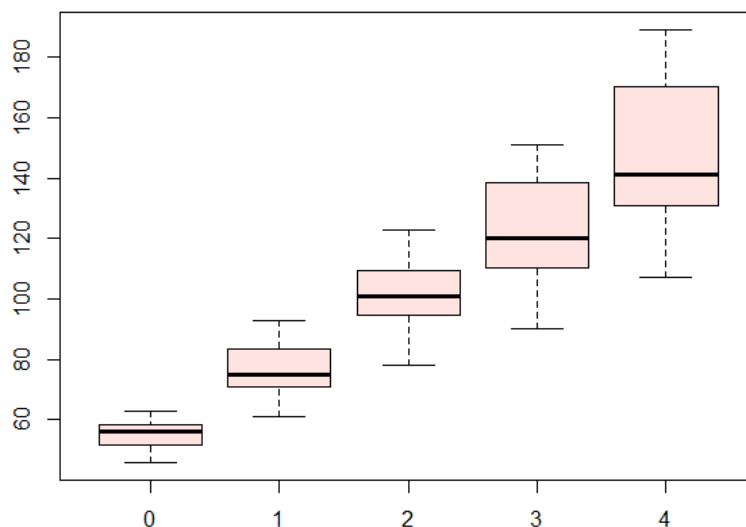
control	thiouracil	thyroxine
105.8400	91.8800	106.3714

```
> means_per_week
```

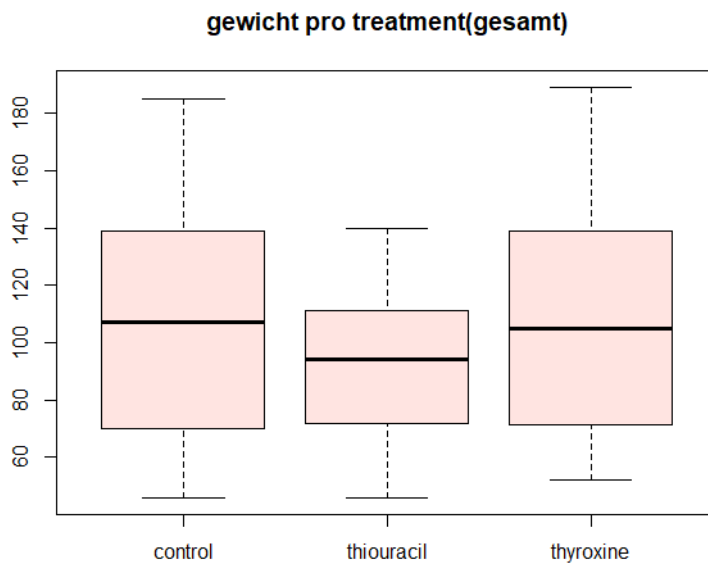
0	1	2	3	4
54.66667	77.00000	101.92593	122.74074	147.70370

Der folgende Boxplot zeigt insgesamt die Entwicklung des Gewichts der Ratten über die Wochen hinweg:

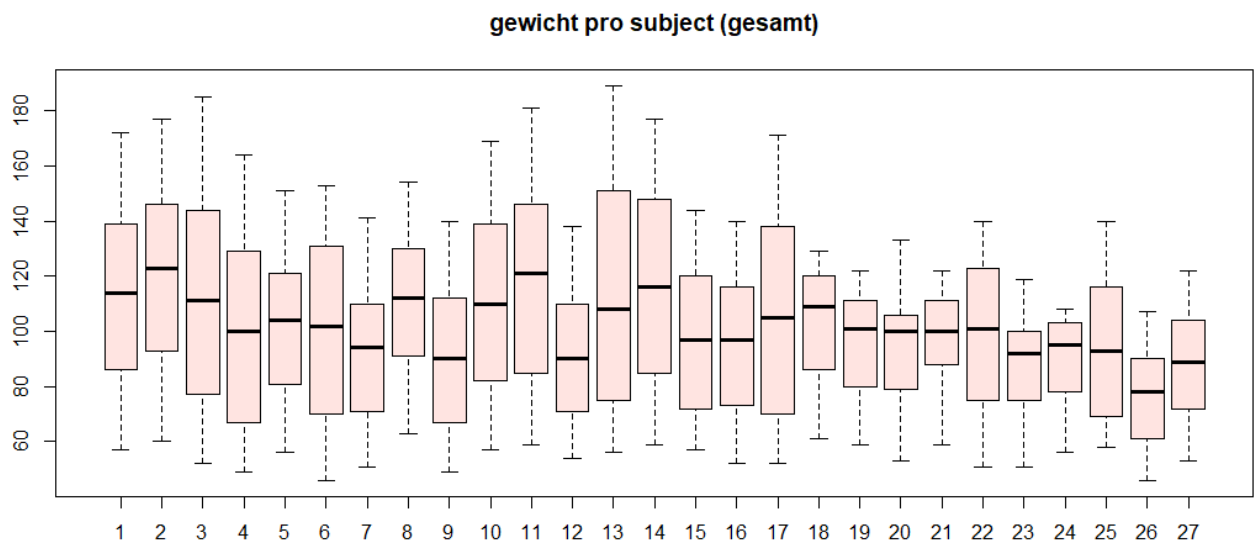
entwicklung des gewichts pro woche (gesamt)



Der folgende Boxplot zeigt das Gewicht aufgeschlüsselt nach Behandlungen:

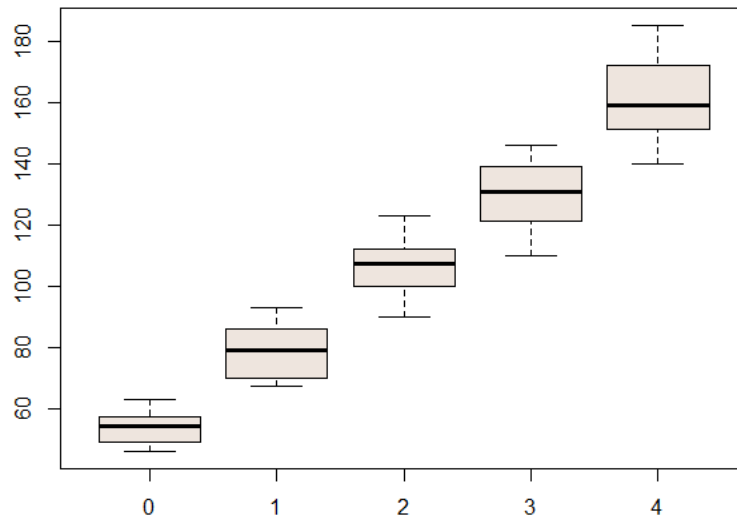


Der folgende Boxplot zeigt das Gewicht der einzelnen Ratten (insgesamt über den gesamten Beobachtungszeitraum):

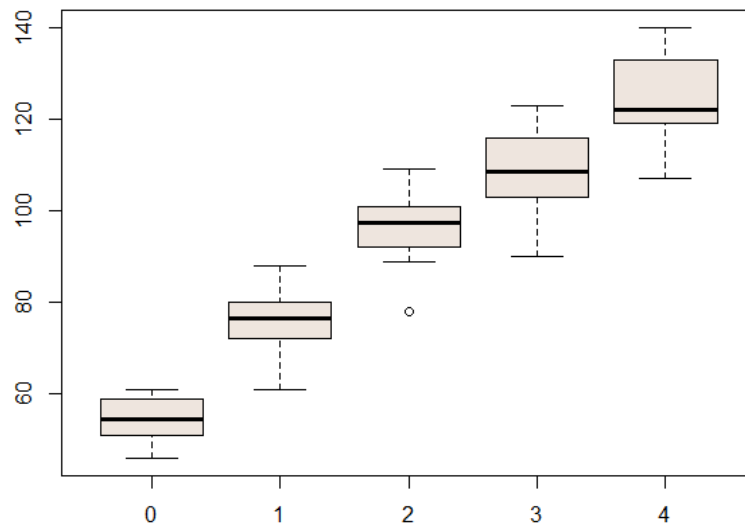


Gewicht im zeitlichen Verlauf pro betrachteter `treat`-Gruppe:

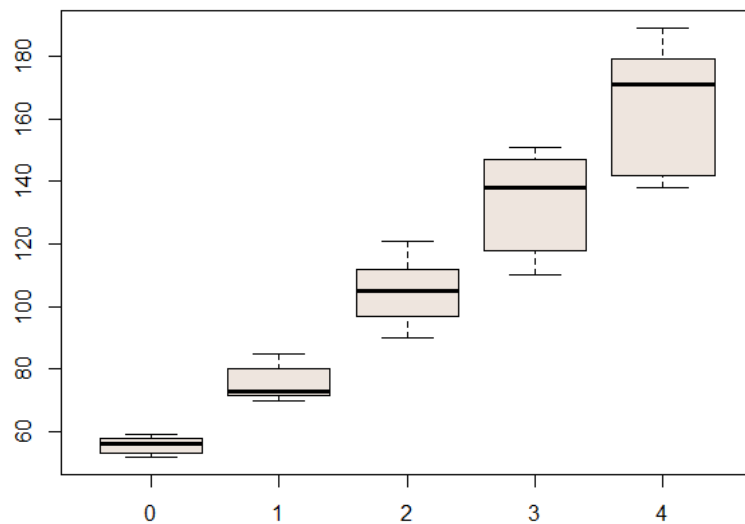
gewicht der control gruppe über die wochen



gewicht der thiouracil gruppe über die wochen



gewicht der thyroxine gruppe über die wochen



Modell mit Zufallseffekten für den Verlauf der Gewichtswerte der Ratten:

MODELL 1:

```
> summary(rat.mod1)
Linear mixed model fit by maximum likelihood ['lmerMod']
Formula: ratdrink$wt ~ ratdrink$weeks + ratdrink$treat + (1 | subject)
Data: ratdrink
```

	AIC	BIC	logLik	deviance	df.resid
	1056.0	1073.4	-522.0	1044.0	129

Scaled residuals:

	Min	1Q	Median	3Q	Max
	-2.69230	-0.60919	-0.05952	0.59404	2.89519

Random effects:

Groups	Name	Variance	Std.Dev.
subject	(Intercept)	51.57	7.181
Residual		104.18	10.207

Number of obs: 135, groups: subject, 27

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	59.4770	2.9637	20.069
ratdrink\$weeks	23.1815	0.6212	37.320
ratdrink\$treatthiouracil	-13.9600	3.8053	-3.669
ratdrink\$treatthyroxine	0.5314	4.1932	0.127

Correlation of Fixed Effects:

	(Intr)	rtdrnk\$w	rtdrnk\$trtthr
ratdrnk\$wks	-0.419		
rtdrnk\$trtthr	-0.642	0.000	
rtdrnk\$trtthy	-0.583	0.000	0.454

MODELL 2:

```
> summary(rat.mod2)
Linear mixed model fit by maximum likelihood ['lmerMod']
Formula: ratdrink$wt ~ ratdrink$weeks + ratdrink$treat + (weeks | subject)
Data: ratdrink
```

	AIC	BIC	logLik	deviance	df.resid
	936.7	960.0	-460.4	920.7	127

Scaled residuals:

	Min	1Q	Median	3Q	Max
	-1.83603	-0.56182	0.00687	0.57235	2.06117

Random effects:

Groups	Name	Variance	Std.Dev.	Corr
subject	(Intercept)	31.34	5.598	
	weeks	34.11	5.840	-0.34
Residual		18.90	4.348	

Number of obs: 135, groups: subject, 27

Fixed effects:

	Estimate	Std. Error	t value
--	----------	------------	---------

(Intercept)	54.2435	1.9700	27.534
ratdrink\$weeks	23.1815	1.1547	20.076
ratdrink\$treatthiouracil	0.9069	2.7031	0.336
ratdrink\$treatthyroxine	-0.5203	2.9786	-0.175

Correlation of Fixed Effects:

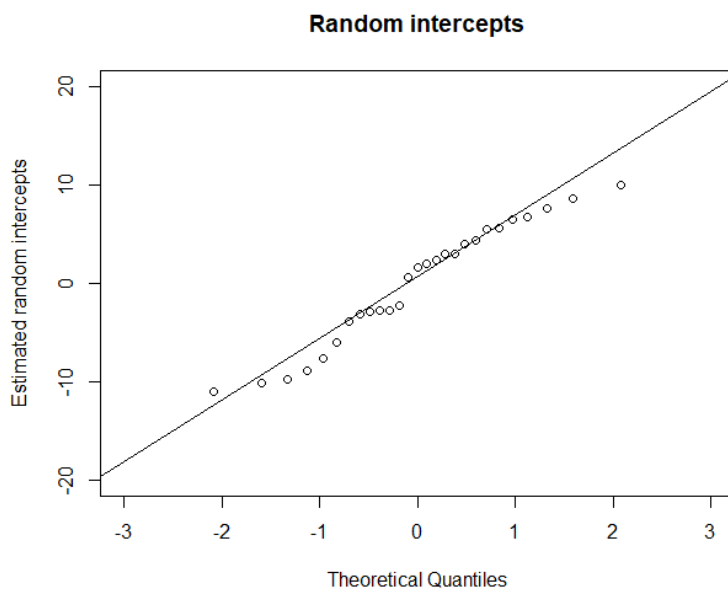
	(Intr)	rtdrnk\$w	rtdrnk\$trtthr
ratdrnk\$wks	-0.242		
rtdrnk\$trtthr	-0.686	0.000	
rtdrnk\$trtthy	-0.623	0.000	0.454

Modellvergleich mittels ANOVA:

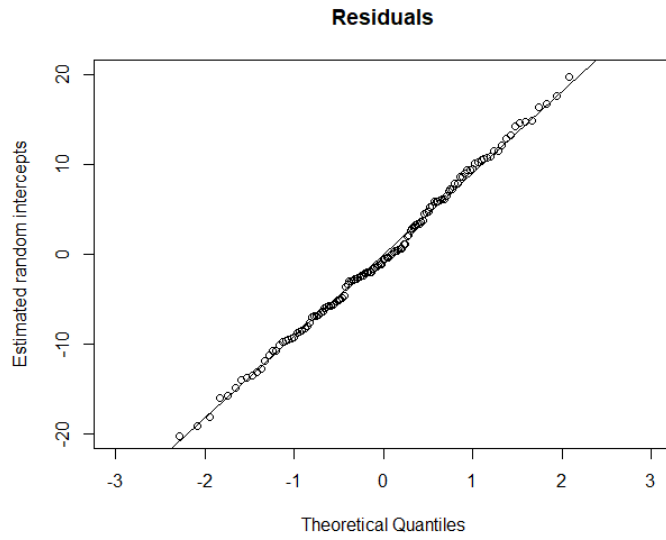
```
> anova(rat.mod1, rat.mod2)
Data: ratdrink
Models:
rat.mod1: ratdrink$wt ~ ratdrink$weeks + ratdrink$treat + (1 | subject)
rat.mod2: ratdrink$wt ~ ratdrink$weeks + ratdrink$treat + (weeks | subject)
      Df    AIC    BIC logLik deviance  Chisq Chi Df Pr(>Chisq)
rat.mod1  6 1055.97 1073.40 -521.98  1043.97
rat.mod2  8  936.74  959.98 -460.37   920.74 123.23    2 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Mittels ANOVA wurden das Modell 1 mit dem Modell 2 verglichen hinsichtlich des Zufallseffekt auf den Verlauf der Gewichtsmesswerte der Ratten. Der p-Wert des Tests mittels ANOVA war hierbei signifikant.

Random Intercepts:



Residuals:



R-Code zu Aufgabe 4 (Longitudinaldaten):

```
#####
# AUFGABE 4 (Longitudinaldaten)
#####
# Die Daten "ratdrink" im Package faraway enthalten für 27 Ratten die Gewichtsmessung
# in fünf aufeinander folgenden Wochen. Die ersten zehn Ratten sind gehören zur
# Kontrollgruppe, bei sieben Ratten wurden Thyroxin in das Trinkwasser hinzugefügt
# und bei 10 Ratten wurde Thiouracil hinzugefügt. Stelle die Daten in geeigneter Form
# dar und bestimme ein Modell für den zeitlichen Verlauf.

data(ratdrink)
ratdrink
head(ratdrink,n=10)

# profile der ratten in den einzelnen gruppen
means_per_subject <- tapply(ratdrink$wt, ratdrink$subject, mean)
means_per_treat <- tapply(ratdrink$wt, ratdrink$treat, mean)
means_per_week <- tapply(ratdrink$wt, ratdrink$weeks, mean)

boxplot(ratdrink$wt ~ ratdrink$weeks)
boxplot(ratdrink$wt ~ ratdrink$treat)
boxplot(ratdrink$wt ~ ratdrink$subject)

control_rat <- ratdrink[ratdrink$treat=="control",]
thiouracil_rat <- ratdrink[ratdrink$treat=="thiouracil",]
thyroxine_rat <- ratdrink[ratdrink$treat=="thyroxine",]

boxplot(control_rat$wt ~ control_rat$weeks, col="seashell2",
        main="gewicht der control gruppe über die wochen")
boxplot(thiouracil_rat$wt ~ thiouracil_rat$weeks, col="seashell2",
        main="gewicht der thiouracil gruppe über die wochen")
boxplot(thyroxine_rat$wt ~ thyroxine_rat$weeks, col="seashell2",
        main="gewicht der thyroxine gruppe über die wochen")

# modell mit zufallseffekten für den verlauf der gewichtswerte

# modell 1
rat.mod1 <- lmer(ratdrink$wt ~ ratdrink$weeks + ratdrink$treat + (1|subject),
                data=ratdrink, REML=FALSE, na.action=na.omit)
summary(rat.mod1)

# modell 2
rat.mod2 <- lmer(ratdrink$wt ~ ratdrink$weeks + ratdrink$treat + (weeks|subject),
                data=ratdrink, REML=FALSE, na.action=na.omit)
summary(rat.mod2)

# anova (zum modellvergleich)
anova(rat.mod1, rat.mod2)

# random effects prediction
qint<-ranef(rat.mod1)$subject[["(Intercept)"]]
qres<-residuals(rat.mod1)
qqnorm(qint,ylab="Estimated random intercepts",
       xlim=c(-3,3), ylim=c(-20,20), main = "Random intercepts")
qqline(qint)

qqnorm(qres,ylab="Estimated random intercepts",
       xlim=c(-3,3), ylim=c(-20,20), main = "Residuals")
qqline(qres)
```

Aufgabe 5 (MANOVA):

Im Datensatz „students“ im Package HSAUR findet man die Ergebnisse von 35 Studierenden, zur Risikobereitschaft in zwei verschiedenen Tests, die als „hih“ und „low“ bezeichnet werden. Die Studierenden wurden dabei zufällig den drei Behandlungsgruppen mit den Bezeichnungen „AA“, „C“ und „NC“ zugeordnet. **Vergleiche die Ergebnisse in den Gruppen in den beiden Variablen.**

Der Datensatz `students` stammt aus einem Test mit 2 parallelen Formularen, in dem 3 (zufällig zugeteilte) Behandlungsgruppen (nämlich „AA“, „C“ und „NC“) geführt wurden. Er umfasst 35 Beobachtungen. Die Variable `low` ist das Ergebnis des ersten Tests und die Variable `high` jenes des zweiten Tests.

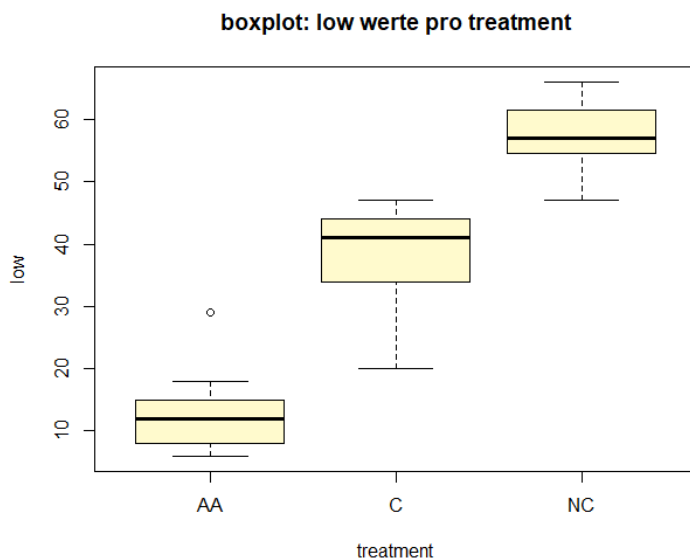
Überblick über den Datensatz und deskriptive Statistiken:

```
> students
  treatment low high
1         AA   8  28
2         AA  18  28
3         AA   8  23
4         AA  12  20
5         AA  15  30
6         AA  12  32
7         AA  18  31
8         AA  29  25
9         AA   6  28
10        AA   7  28

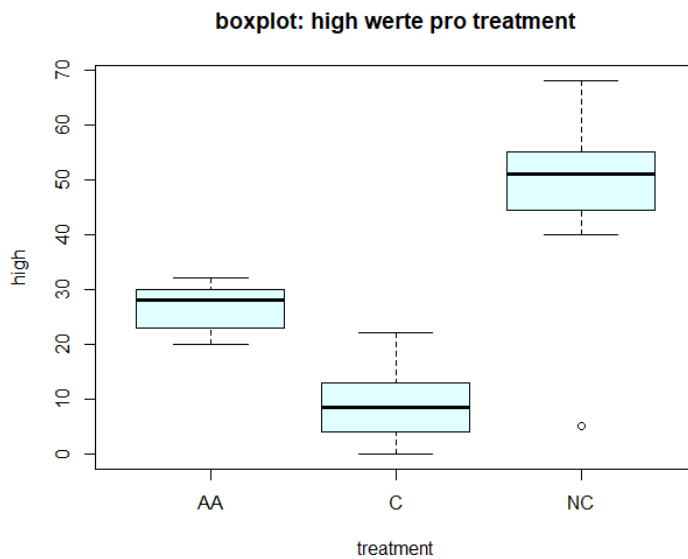
> str(students)
'data.frame':   35 obs. of  3 variables:
 $ treatment: Factor w/ 3 levels "AA","C","NC": 1 1 1 1 1 1 1 1 1 1 ...
 $ low      : int   8 18 8 12 15 12 18 29 6 7 ...
 $ high     : int  28 28 23 20 30 32 31 25 28 28 ...

> summary(students)
  treatment      low      high
AA:14   Min.   : 6.0   Min.   : 0.00
C :10   1st Qu.:13.0   1st Qu.:13.50
NC:11   Median :34.0   Median :28.00
        Mean   :33.8   Mean   :28.06
        3rd Qu.:51.5   3rd Qu.:40.00
        Max.   :66.0   Max.   :68.00
```

Boxplot der `low` Werte pro Treatment:

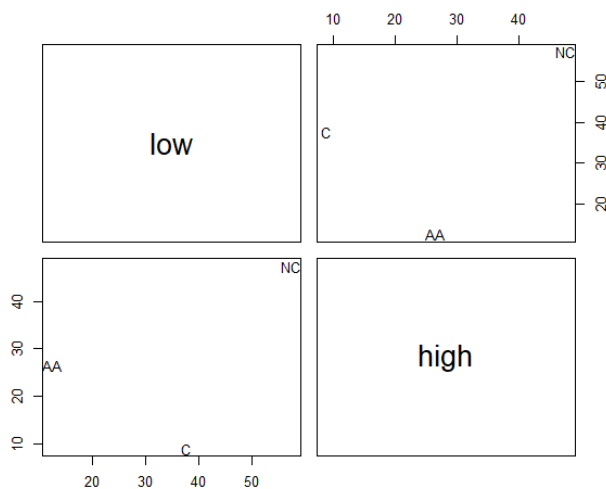


Boxplot der high Werte pro Treatment:



Mittelwerte der einzelnen Gruppen:

```
treatment    low    high
AA 12.57143 26.42857
C  37.70000  9.00000
NC 57.27273 47.45455
```



Annahmen prüfen:

- Test auf Normalverteilung der abhängigen variablen in den Gruppen mittels multivariatem Shapiro-Wilk-Test:

```
> mshapiro.test(test_nv[, 1:14]) # gr. AA: hyp. d. mv. NV nicht verworfen
```

Shapiro-Wilk normality test

```
data:  Z
W = 0.86742, p-value = 0.03861
```

```
> mshapiro.test(test_nv[, 15:24]) # gr. C: hyp. d. mv. NV nicht verworfen

Shapiro-Wilk normality test

data:  Z
W = 0.8651, p-value = 0.0876

> mshapiro.test(test_nv[, 25:35]) # gr. NC: hyp. d. mv. NV nicht verworfen

Shapiro-Wilk normality test

data:  Z
W = 0.80106, p-value = 0.009713
```

Die Tests ergeben, dass die Nullhypothese über die multivariate Normalverteilung nicht verworfen wird.

- Test auf Varianzhomogenität mittels Bartlett-Test:

```
> bartlett.test(high~treatment, students) # H0 nicht verworfen

Bartlett test of homogeneity of variances

data:  high by treatment
Bartlett's K-squared = 21.635, df = 2, p-value = 2.005e-05

> bartlett.test(low~treatment, students) # H0 nicht verworfen

Bartlett test of homogeneity of variances

data:  low by treatment
Bartlett's K-squared = 2.5405, df = 2, p-value = 0.2808
```

Die Tests ergeben, dass die Nullhypothese über die Varianzhomogenität nicht verworfen wird.

MANOVA durchführen:

Die MANOVA wurde hier in mehreren Varianten durchgeführt (hinsichtlich der Tests), nämlich mit Hilfe von Wilks Lambda, Roys größtem Eigenwert, Pillais Spurkriterium und dem Hotelling-Lawley-Kriterium. All diese Tests ergaben, dass der Effekt signifikant ist.

```
> students.result <- manova(cbind(students$low,students$high)~students$treatment,
+                             data=students)
> summary(students.result, test="Wilks") # wilks lambda
```

	Df	Wilks	approx F	num Df	den Df	Pr(>F)
students\$treatment	2	0.035085	67.251	4	62	< 2.2e-16 ***
Residuals	32					

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> summary(students.result, test="Roy") # roys groesster eigenwert
              Df      Roy approx F num Df den Df      Pr(>F)
students$treatment  2  8.1239   129.98      2      32 4.336e-16 ***
Residuals          32
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> summary(students.result, test="Pillai") # pillais spurkriterium
              Df Pillai approx F num Df den Df      Pr(>F)
students$treatment  2  1.5703   58.468      4      64 < 2.2e-16 ***
Residuals          32
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> summary(students.result, test="Hotelling-Lawley") # hotelling-lawley kriterium
              Df Hotelling-Lawley approx F num Df den Df      Pr(>F)
students$treatment  2      10.248   76.858      4      60 < 2.2e-16 ***
Residuals          32
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Univariate Tests:

Zur Untersuchung der einzelnen, einfachen Effekte wurden univariate Tests durchgeführt, die ebenfalls jeweils signifikant waren.

```
> summary.aov(students.result) # vgl. welche verschieden sind
Response 1 :
              Df  Sum Sq Mean Sq F value    Pr(>F)
students$treatment  2 12521.9  6260.9  128.62 5.039e-16 ***
Residuals          32  1557.7    48.7
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Response 2 :
              Df Sum Sq Mean Sq F value    Pr(>F)
students$treatment  2  7807.7  3903.9  39.233 2.459e-09 ***
Residuals          32  3184.2    99.5
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Paarweise multivariate Tests:

Zusätzlich wurden paarweise multivariate Tests (für die Kombination von Treatments) durchgeführt, die auch signifikant waren.

```
> summary(manova(cbind(students$low,students$high)~students$treatment,
+                    data=students, subset=treatment %in% c("AA","NC")))
              Df  Pillai approx F num Df den Df      Pr(>F)
students$treatment  1 0.93788   166.07      2    22 5.317e-14 ***
Residuals          23
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> summary(manova(cbind(students$low,students$high)~students$treatment,
+                    data=students, subset=treatment %in% c("AA","C")))
              Df  Pillai approx F num Df den Df      Pr(>F)
students$treatment  1 0.9211   122.58      2    21 2.626e-12 ***
Residuals          22
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> summary(manova(cbind(students$low,students$high)~students$treatment,
+                    data=students, subset=treatment %in% c("C","NC")))
              Df  Pillai approx F num Df den Df      Pr(>F)
students$treatment  1 0.76211   28.833      2    18 2.44e-06 ***
Residuals          19
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```


R-Code zu Aufgabe 5 (MANOVA):

```
#####
# AUFGABE 5 (MANOVA)
#####
# Im Datensatz „students“ im Package HSAUR findet man die Ergebnisse von 35 Studierenden,
# zur Risikobereitschaft in zwei verschiedenen Tests, die als „high“ und „low“ bezeichnet
# werden. Die Studierenden wurden dabei zufällig den drei Behandlungsgruppen mit den
# Bezeichnungen „AA“, „C“ und „NC“ zugeordnet. Vergleiche die Ergebnisse in den Gruppen
# in den beiden Variablen.

# deskriptiv/darstellung:
data(students)
students
str(students)
summary(students)

boxplot(students$low ~ students$treatment, data=students,
        xlab="treatment", ylab="low", col="lemonchiffon1")
boxplot(students$high ~ students$treatment, data=students,
        xlab="treatment", ylab="high", col="lightcyan")

means <- aggregate(students[,c("low", "high")],
                    list(treatment=students$treatment),
                    mean)
pairs(means[, -1], panel=function(x, y) {text(x, y,
        levels(students$treatment))})

# annahmen pruefen:

# normalverteilung in den gruppen
test_nv <- rbind(as.numeric(t(students)[2,]), as.numeric(t(students)[3,]))
rownames(test_nv) <- c("low", "high")
mshapiro.test(test_nv[, 1:14]) # gr. AA: hyp. d. mv. NV nicht verworfen
mshapiro.test(test_nv[, 15:24]) # gr. C: hyp. d. mv. NV nicht verworfen
mshapiro.test(test_nv[, 25:35]) # gr. NC: hyp. d. mv. NV nicht verworfen

# varianzhomogenitaet in gruppen:
bartlett.test(high~treatment, students) # H0 nicht verworfen
bartlett.test(low~treatment, students) # H0 nicht verworfen

# MANOVA durchfuehren:
students.result <- manova(cbind(students$low, students$high)~students$treatment,
                          data=students)
summary(students.result, test="Wilks") # wilks lambda
summary(students.result, test="Roy") # roys groesster eigenwert
summary(students.result, test="Pillai") # pillais spurkriterium
summary(students.result, test="Hotelling-Lawley") # hotelling-lawley kriterium

# univariate tests:
summary.aov(students.result) # vgl. welche verschieden sind
# (ergebnis: sind beide verschieden unter den treatments)

# paarweise multivariate tests:
summary(manova(cbind(students$low, students$high)~students$treatment,
                    data=students, subset=treatment %in% c("AA", "NC")))

summary(manova(cbind(students$low, students$high)~students$treatment,
                    data=students, subset=treatment %in% c("AA", "C")))

summary(manova(cbind(students$low, students$high)~students$treatment,
                    data=students, subset=treatment %in% c("C", "NC")))
```

Aufgabe 6 (MANOVA):

Der Datensatz „water“ im Package HSAUR enthält die Daten über die Wasserhärte und die Mortalitätsrate in 61 Städten in England und Wales, die nach ihrer Lage in „north“ und „south“ eingeteilt werden. Untersuche die Unterschiede zwischen den beiden Regionen („north“-„south“) und stelle die Daten in geeigneter Form dar.

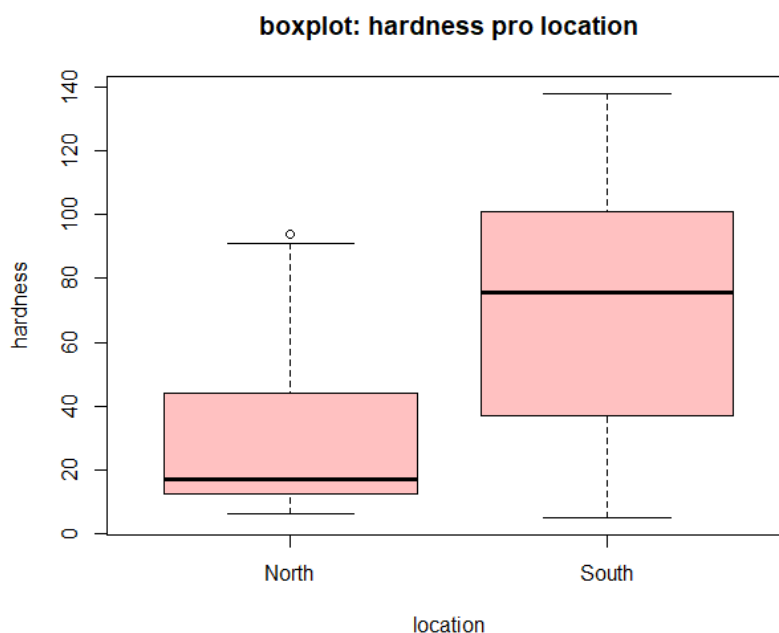
Überblick über den Datensatz und deskriptive Statistiken:

```
> water
  location      town mortality hardness
1   South      Bath      1247      105
2   North  Birkenhead      1668       17
3   South  Birmingham      1466        5
4   North  Blackburn      1800       14
5   North  Blackpool      1609       18
6   North   Bolton      1558       10
7   North   Bootle      1807       15
8   South  Bournemouth      1299       78

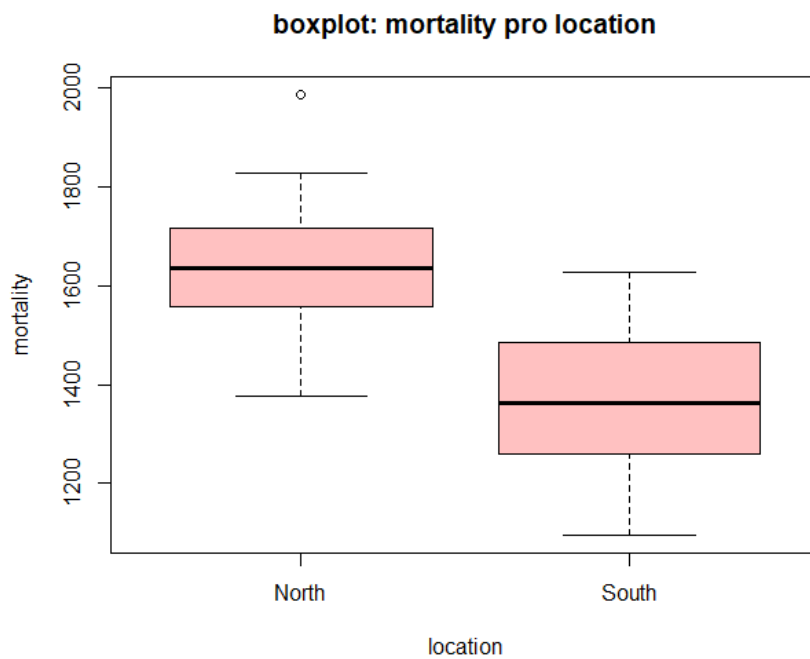
> str(water)
'data.frame':   61 obs. of  4 variables:
 $ location : Factor w/ 2 levels "North","South": 2 1 2 1 1 1 1 2 1 2
 $ town      : chr  "Bath" "Birkenhead" "Birmingham" "Blackburn" ...
 $ mortality : int  1247 1668 1466 1800 1609 1558 1807 1299 1637 1359
 $ hardness  : int   105  17  5  14  18  10  15  78  10  84 ...

> summary(water)
  location      town      mortality      hardness
North:35   Length:61   Min.      :1096   Min.      :  5.00
South:26   Class :character 1st Qu.:1379   1st Qu.: 14.00
          Mode  :character Median :1555   Median : 39.00
          Mean   :1524   Mean   : 47.18
          3rd Qu.:1668   3rd Qu.: 75.00
          Max.   :1987   Max.   :138.00
```

Boxplot der Wasserhärte pro Location:



Boxplot der Mortalität pro Location:



Mittelwerte pro Gruppe:

```
> means
  location mortality hardness
1   North    1633.6     30.4
2   South    1376.8     69.8
```

Paare der Mittelwerte:



Annahmen prüfen:

- Test auf Normalverteilung der abhängigen Variablen in den Gruppen mittels multivariatem Shapiro-Wilk-Test:

```
> mshapiro.test(water_nv[, 1:35]) # gr. North: hyp. d. mv. NV nicht verworfen

      Shapiro-Wilk normality test

data:  Z
W = 0.94435, p-value = 0.07602

> mshapiro.test(water_nv[, 36:61]) # gr. North: hyp. d. mv. NV nicht verworfen

      Shapiro-Wilk normality test

data:  Z
W = 0.97747, p-value = 0.8162
```

Im Zuge des Tests auf (multivariate) Normalverteilung wurde die Nullhypothese nicht verworfen.

- Test auf Varianzhomogenität mittels Bartlett-Test:

```
> bartlett.test(mortality~location, water) # H0 nicht verworfen

      Bartlett test of homogeneity of variances

data:  mortality by location
Bartlett's K-squared = 0.016408, df = 1, p-value = 0.8981

> bartlett.test(hardness~location, water) # H0 nicht verworfen

      Bartlett test of homogeneity of variances

data:  hardness by location
Bartlett's K-squared = 5.4191, df = 1, p-value = 0.01992
```

Im Zuge des Tests auf Varianzhomogenität wurde die Nullhypothese nicht verworfen.

MANOVA durchführen:

Die MANOVA wurde auf die Variablen mortality und hardness durchgeführt. Dabei zeichnete sich nach allen ausgeführten Testmethoden eine Signifikanz der Effekte ab.

```
> water.result <- manova(cbind(water$mortality, water$hardness)~water$location, data=water)

> summary(water.result, test="Wilks") # wilks lambda
```

	Df	Wilks	approx F	num Df	den Df	Pr(>F)
water\$location	1	0.52626	26.106	2	58	8.217e-09 ***
Residuals	59					

```
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> summary(water.result, test="Roy") # roys groesster eigenwert
              Df      Roy approx F num Df den Df      Pr(>F)
water$location 1 0.90021   26.106      2    58 8.217e-09 ***
Residuals      59
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> summary(water.result, test="Pillai") # pillais spurkriterium
              Df  Pillai approx F num Df den Df      Pr(>F)
water$location 1 0.47374   26.106      2    58 8.217e-09 ***
Residuals      59
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

> summary(water.result, test="Hotelling-Lawley") # hotelling-lawley kriterium
              Df Hotelling-Lawley approx F num Df den Df      Pr(>F)
water$location 1              0.90021   26.106      2    58 8.217e-09 ***
Residuals      59
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Univariate Tests:

Die univariaten Tests ergaben ebenfalls jeweils signifikante Effekte.

```
> summary.aov(water.result) # vgl. welche verschieden sind
Response 1 :
              Df  Sum Sq Mean Sq F value      Pr(>F)
water$location 1  983729  983729  51.388 1.402e-09 ***
Residuals      59 1129444   19143
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Response 2 :
              Df Sum Sq Mean Sq F value      Pr(>F)
water$location 1  23122 23122.0  21.333 2.14e-05 ***
Residuals      59  63947  1083.8
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

R-Code zu Aufgabe 6 (MANOVA):

```
#####
# AUFGABE 6 (MANOVA)
#####
# Der Datensatz „water“ im Package HSAUR enthält die Daten über die Wasserhärte und
# die Mortalitätsrate in 61 Städten in England und Wales, die nach ihrer Lage in „north“
# und „south“ eingeteilt werden. Untersuche die Unterschiede zwischen den beiden Regionen
# („north“-„south“) und stelle die Daten in geeigneter Form dar.

data(water)
water
str(water)
summary(water)

boxplot(water$hardness ~ water$location, data=water,
        xlab="location", ylab="hardness", col="rosybrown1",
        main="boxplot: hardness pro location")

boxplot(water$mortality ~ water$location, data=water,
        xlab="location", ylab="mortality", col="rosybrown1",
        main="boxplot: mortality pro location")

means <- aggregate(water[,c("mortality","hardness")],
                   list(location=water$location))
means[1:2,2:3] <- round(means[1:2,2:3],digits=1)

pairs(means[, -1], panel=function(x,y){text(x,y,
      levels(water$location))})

# annahmen pruefen:

# normalverteilung in den gruppen
water_nv <- water[order(water$location), ]
water_nv <- t(water_nv)
water_nv <- water_nv[-(1:2), ]
water_1 <- as.numeric(water_nv[1, ])
water_2 <- as.numeric(water_nv[2, ])
water_nv <- rbind(water_1,water_2)
rownames(water_nv) <- c("mortality","hardness")
mshapiro.test(water_nv[, 1:35]) # gr. North: hyp. d. mv. NV nicht verworfen
mshapiro.test(water_nv[, 36:61]) # gr. North: hyp. d. mv. NV nicht verworfen

# varianzhomogenitaet in gruppen:
bartlett.test(mortality~location, water) # H0 nicht verworfen
bartlett.test(hardness~location, water) # H0 nicht verworfen

# MANOVA durchfuehren:
water.result <- manova(cbind(water$mortality,water$hardness)~water$location,
                      data=water)
summary(water.result, test="Wilks") # wilks lambda
summary(water.result, test="Roy") # roys groesster eigenwert
summary(water.result, test="Pillai") # pillais spurkriterium
summary(water.result, test="Hotelling-Lawley") # hotelling-lawley kriterium

# univariate tests:
summary.aov(water.result) # vgl. welche verschieden sind
# (ergebnis: sind beide verschieden unter den treatments)
```

Literaturquellen:

- Folien und R-Codes zu den bisher vorgetragenen Kapiteln aus UK Erweiterungen des linearen Modells (Prof. Wilfried Grossmann).
- R Package *furniture*: <https://cran.r-project.org/web/packages/furniture/furniture.pdf>.
- N.H. Timm (2002): Applied Multivariate Statistics. Springer Verlag, NY, USA. <https://app.quadstat.net/dataset/r-dataset-package-hsaur-students>.
- STHDA (Statistical Tools for High-Throughput Data Analysis. MANOVA Analysis: <http://www.sthda.com/english/wiki/manova-test-in-r-multivariate-analysis-of-variance>.
- <http://dwoell.de/rexrepos/posts/multMANOVA.html>.