Linear mixed models in R
Day 5

JONAS WALTHER

Evaluation

Please give your feedback for the course at:

https://evaluation.zeq.uni-tuebingen.de/evasys/online/

Summary-Output of an LME

- You prepared your final model
- You checked the assumptions
- You understood and interpreted the outcome

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: RT ~ Group + Context + Group:Context + Age + AoA + Trial + (1 +
##     Context | Subject) + (1 + Context | ItemNr)
##    Data: PN_Data
##
## Random effects:
##  Groups   Name        Variance Std.Dev. Corr
##  ItemNr   (Intercept) 24021    154.99
##           ContextUK    3449     58.73   -0.15
##  Subject  (Intercept) 19522    139.72
##           ContextUK   16426    128.16   -0.41
##  Residual             55688    235.98
## Number of obs: 7227, groups:  ItemNr, 210; Subject, 74
##
## Fixed effects:
##                              Estimate Std. Error t value
## (Intercept)                 1019.6555    79.0877  12.893
## GroupExperimental            -59.6657    37.2576  -1.601
## ContextUK                    -11.2579    23.2922  -0.483
## Age                           -1.9501     2.4100  -0.809
## AoA                            3.2835     3.9845   0.824
## Trial                          0.4629     0.2875   1.610
## GroupExperimental:ContextUK   45.7348    32.1411   1.423
##
## Correlation of Fixed Effects:
##             (Intr) GrpExp CntxUK Age    AoA    Trial
## GrpExprmntl  0.196
## ContextUK   -0.127  0.290
## Age         -0.810 -0.309 -0.012
## AoA         -0.269 -0.217  0.001 -0.234
## Trial       -0.110 -0.001 -0.006  0.000  0.001
## GrpExpr:CUK  0.088 -0.404 -0.704  0.010 -0.001  0.000
```

Reporting linear mixed-effects models

"A one-way ANOVA demonstrated that the effect of leadership style was significant for employee engagement, $F(2, 78) = 4.58$, $p = .013$."

There are no standardized guidelines for LMEs yet

The complexity, large structure (and a lack of p-values) make reporting LMEs more effortful

- Especially for short-form texts (i.e. abstracts)

Kim, Dedrick, Cao, Ferron (2008) Multilevel Factor Analysis: Reporting Guidelines and a Review of Reporting Practices

Guidelines and checklist for reporting statistical models

- Multilevel factor analysis, not LMEs
- Might still be helpful to remind you about reporting different aspects of your statistical tests

Reporting LME results

As many details as possible

All relevant statistical measures:

- Data structure and size
- Variable transformations and contrast coding
- Maximal model structure
- Final model results for all fixed and random effects
- Post-hoc tests for relevant effects

Reporting as a table

Effect	Estimate	SE	t	by-Picture SD	by-Participant SD
Intercept	−1.14	0.03	−38.00***	0.15	0.18
Group	0.00	0.05	−0.02		
Context	0.02	0.04	0.65	−0.05	−0.11
Word-lexical frequency	0.00	0.01	−0.08	–	
Age	−0.03	0.02	−1.30		–
Age of L2 acquisition	0.03	0.02	1.15		–
log (Trial number)	0.00	0.01	0.78		
Group:Context	0.03	0.07	0.48		–
Group:Word-lexical frequency	0.00	0.00	−0.28	–	–
Control Group:Context:Word-lexical frequency	0.00	0.01	−0.11	–	–
Mig. Group:Context:Word-lexical frequency	0.01	0.01	1.94′	–	–

(Intercept)	Estimate	SE
GroupExperimental	-59.666	(79.088)
ContextUK	-11.258	(37.258)
Age	-1.950	(23.292)
AoA	3.283	(2.410)
Trial	0.463	(3.984)
GroupExperimental × ContextUK	45.735	(0.288)
SD (Intercept ItemNr)	154.988	(32.141)
SD (ContextUK ItemNr)	58.732	
Cor (Intercept~ContextUK ItemNr)	-0.153	
SD (Intercept Subject)	139.721	
SD (ContextUK Subject)	128.163	
Cor (Intercept~ContextUK Subject)	-0.413	
SD (Observations)	235.982	
Num.Obs.	7227	
R2 Marg.	0.008	
R2 Cond.	0.450	
AIC	100519.6	
BIC	100616.0	
ICC	0.4	
RMSE	229.36	

Model summary package

Creates table outputs for a large variety of models to various output formats

library(modelsummary)

```
Modelsummary(model_GeneralModel, output = "markdown")

Modelsummary(model_GeneralModel, output = "latex")
```

sjPlot package

Creates a html table as output

```
library(sjPlot)

tab_model(model_GeneralModel)
```

	RT		
Predictors	*Estimates*	*CI*	*p*
(Intercept)	1019.66	864.62 – 1174.69	**<0.001**
Group [Experimental]	-59.67	-132.70 – 13.37	0.109
Context [UK]	-11.26	-56.92 – 34.40	0.629
Age	-1.95	-6.67 – 2.77	0.418
AoA	3.28	-4.53 – 11.09	0.410
Trial	0.46	-0.10 – 1.03	0.108
Group [Experimental] × Context [UK]	45.73	-17.27 – 108.74	0.155
Random Effects			
σ^2	55687.53		
τ_{00} ItemNr	24021.24		
τ_{00} Subject	19521.93		
τ_{11} ItemNr.ContextUK	3449.39		
τ_{11} Subject.ContextUK	16425.87		
ρ_{01} ItemNr	-0.15		
ρ_{01} Subject	-0.41		
ICC	0.45		
$N_{Subject}$	74		
N_{ItemNr}	210		

predict_response()

ggeffects package

Integrated with the ggplots package

Calculates marginal means

Provides them in easily usable format for plots

predict_response()

```
model_Large8 %>%
  predict_response(c("Group", "Context"))
## # Predicted values of RT


## Context: PL
## Group        | Predicted |           95% CI
## -----------------------------------------
## Control      |    994.54 | 945.32, 1043.76
## Experimental |    955.29 | 906.37, 1004.20
```

```
## Context: UK
## Group        | Predicted |           95% CI
## -----------------------------------------
## Control      |   1006.62 | 957.25, 1055.99
## Experimental |    967.37 | 918.04, 1016.69


## Adjusted for:
## * Subject = 0 (population-level)
## *  ItemNr = 0 (population-level)
```

predict_response()

X - the values of the first term, x-values

group - levels of the second term

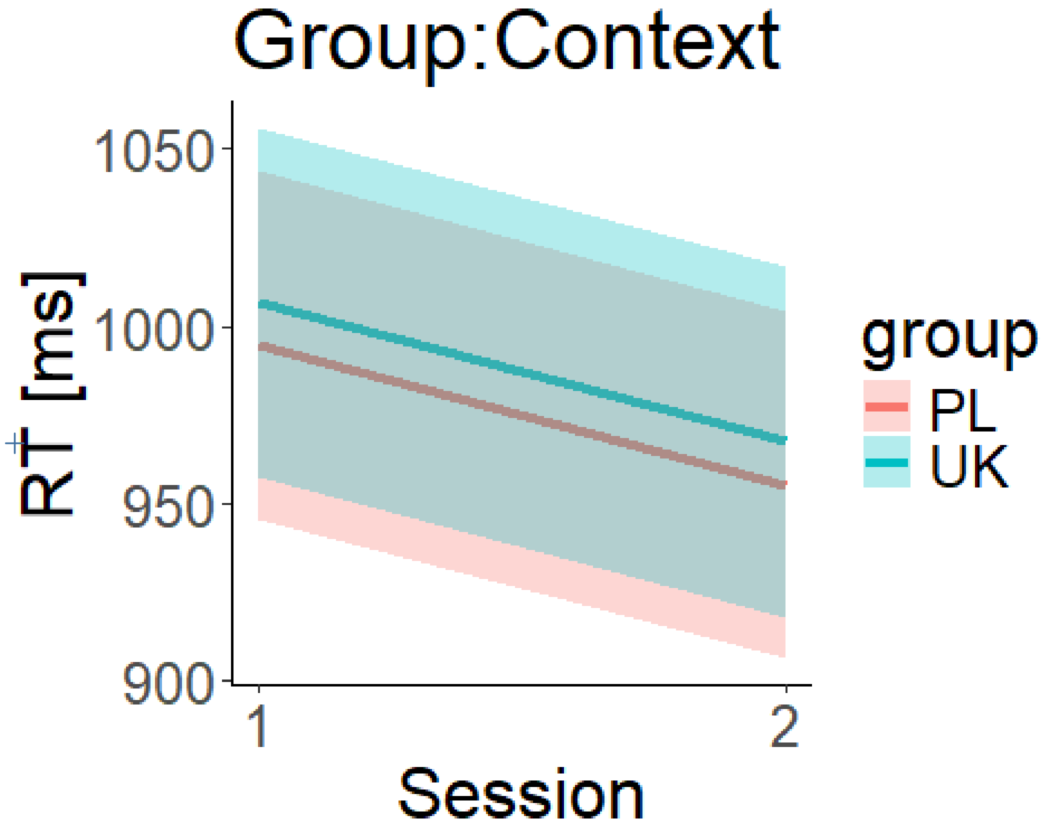
facet – levels of third term

predicted - the predicted y-values

conf.low and conf.high

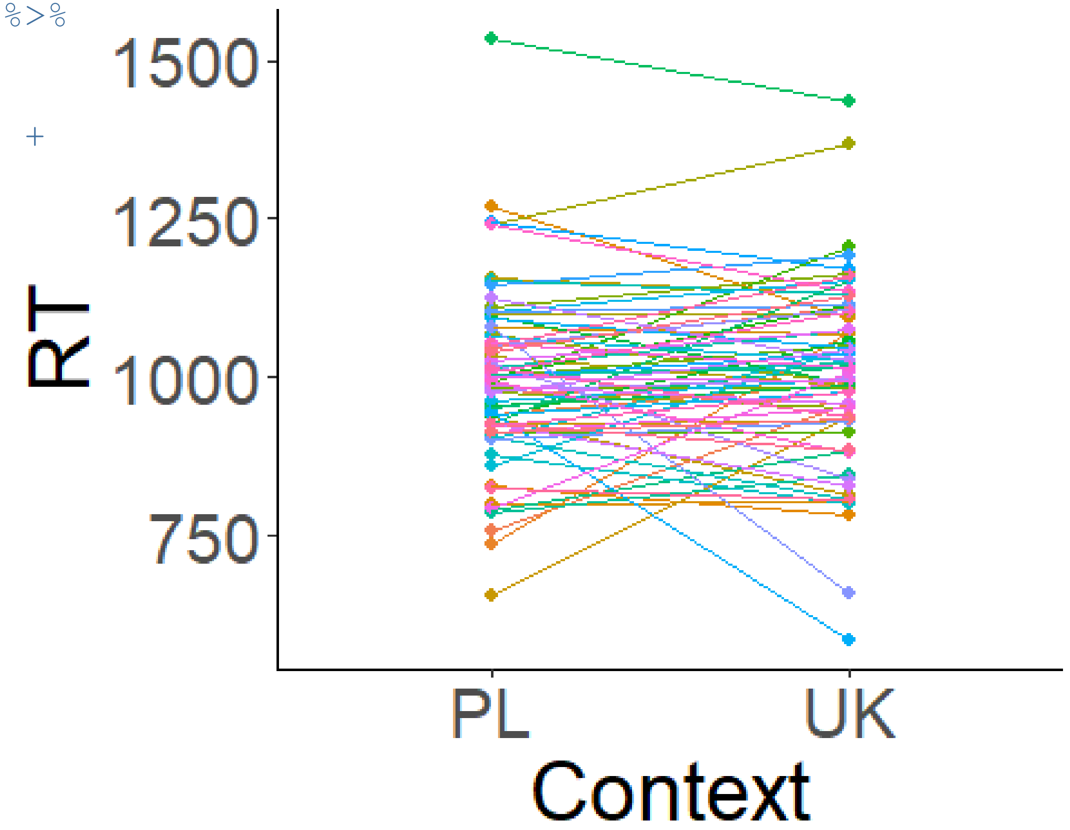
predict_response() for fixed effects

```
model_Large8 %>%
  predict_response(c("Group", "Context")) %>%
  ggplot(aes(x=as.numeric(x), y=predicted,
      color=group, fill=group)) +
  geom_line(linewidth=1.7) +
  geom_ribbon(aes(ymin=conf.low, ymax=conf.high),
      alpha = 0.3, colour = NA) +
  scale_x_continuous(name =
"Session",breaks=c(1,2)) +
  scale_y_continuous(name = "RT [ms]") +
  ggtitle("Predicted effects for Group:Context") +
  theme_classic() +
  theme(text = element_text(size=24),
element_line(size = 2))
```



predict_response() for random effects

```
model_Large8 %>%
  ggpredict(c("Context", "Subject"), type = "re") %>%
    ggplot(aes(x=x, y=predicted, group=group)) +
    geom_point(size=1.7, aes(color=group)) +
    geom_line(aes(color=group)) +  theme_classic() +
    scale_color_discrete(guide="none") +
    scale_y_continuous(name="RT") +
    scale_x_discrete(labels = c("PL",
"UK"),name="Context")+
    theme(text = element_text(size=28),
element_line(size = 2))
```



Power analysis

Statistical tool that calculates the minimum sample size for a given study and analysis

It needs four primary components:

- Statistical power: the likelihood that a test will detect an effect of a certain size if there is one, usually set at 80% or higher
- Sample size: the minimum number of observations needed to observe an effect of a certain size with a given power level
- Alpha: usually 0.05
- Expected effect size

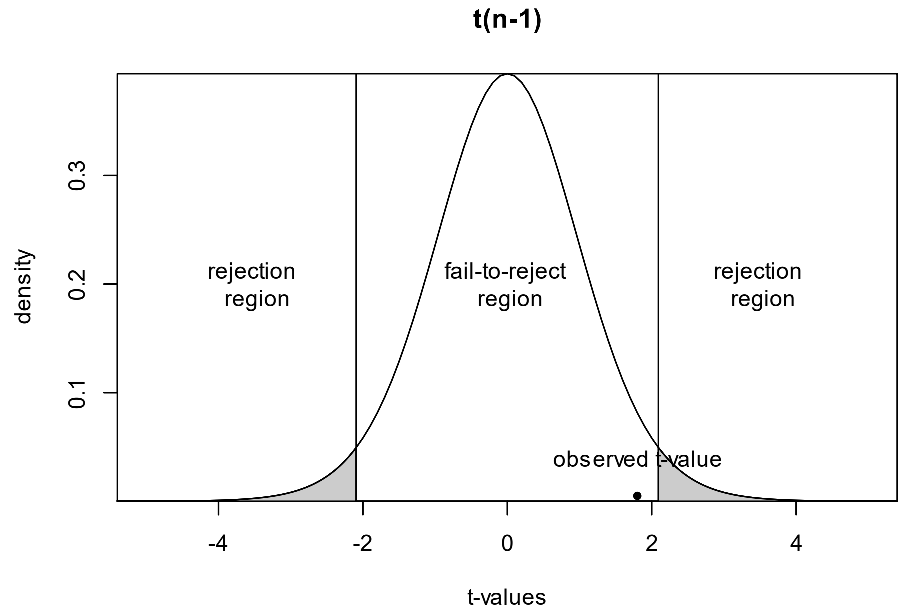
If you have three of those components you can calculate the fourth

- Determine power from existing sample
- Determine necessary sample size to reach a given power

Statistical power

Probability of not rejecting H1, if H1 is true

	$If\ H_0\ is\ True$	$If\ H_1\ is\ True$
$Probability\ to\ reject\ H0$	α	$1-\alpha$
$Probability\ to\ not\ reject\ H_0$	$1-\beta\ (power)$	β



Simulate a model

```
summary(model_Large3)

## Fixed effects:

##                              Estimate Std. Error t value

## (Intercept)                  1020.739     23.344  43.726

## GroupExperimental             -76.150     29.565  -2.576

## ContextUK                     -18.660      8.471  -2.203

## GroupExperimental:ContextUK    53.089     11.706   4.535
```

Take details from previous data or publications

$$Reactiontime \sim Group * Context + (1 + Context|Subject)$$

Simulating a model

```
Subject <- factor(1:40)
Group <- c("Control", "Experimental")

subj_full <- rep(Subject, 20)
group_full <- rep(rep(Group, each=20), 20)
context_full <- c(rep("PL", each=400), rep("UK", each=400))

covars <- data.frame(Subject=subj_full, Group=group_full, Context=context_full)

covars

##     Subject       Group Context

## 1         1     Control      PL

## 2         2     Control      PL

## 3         3     Control      PL

## 4         4     Control      PL

## 5         5     Control      PL
```

Simulate a model

```
summary(model_Large3)

## Fixed effects:

##                              Estimate Std. Error t value

## (Intercept)                  1020.739     23.344  43.726

## GroupExperimental             -76.150     29.565  -2.576

## ContextUK                     -18.660      8.471  -2.203

## GroupExperimental:ContextUK    53.089     11.706   4.535
```

Take details from previous data or publications

$$Reactiontime \sim Group * Context + (1 + Context|Subject)$$

Simulating a model

```
## Intercept and slopes for intervention, time1, time2, intervention:time1, intervention:time2
fixed <- c(1000, -80, -20, 50)


## Random intercepts for participants clustered by class
rand <- 0.5

## residual variance
res <- 2


library(simr)


model <- makeLmer(RT ~ Group*Context + (1|Subject), fixef=fixed, VarCorr=rand, sigma=res, data=covars)
```

Simulating a model

```
## Linear mixed model fit by REML ['lmerMod']

## Formula: RT ~ Group * Context + (1 |
Subject)

##    Data: covars

## REML criterion at convergence: 3522.113

## Random effects:

##  Groups   Name        Std.Dev.

##  Subject  (Intercept) 0.7071

##  Residual             2.0000

## Number of obs: 800, groups:  Subject, 40
```

```
## Fixed Effects:

##(Intercept)            GroupExperimental

##   1000                              -80

##ContextUK  GroupExperimental:ContextUK

##   -20                               50
```

Calculate power of a given sample

```
powerSim(model, nsim=20, test = fixed(xname="ContextUK", method = "t"))

Power for predictor 'ContextUK', (95% confidence interval):==========================|
      100.0% (83.16, 100.0)

Test: t-test with Satterthwaite degrees of freedom (package lmerTest)
      Effect size for ContextUK is -20.

Based on 20 simulations, (0 warnings, 0 errors)
alpha = 0.05, nrow = 800

Time elapsed: 0 h 0 m 2 s
```

Calculate power of a given sample

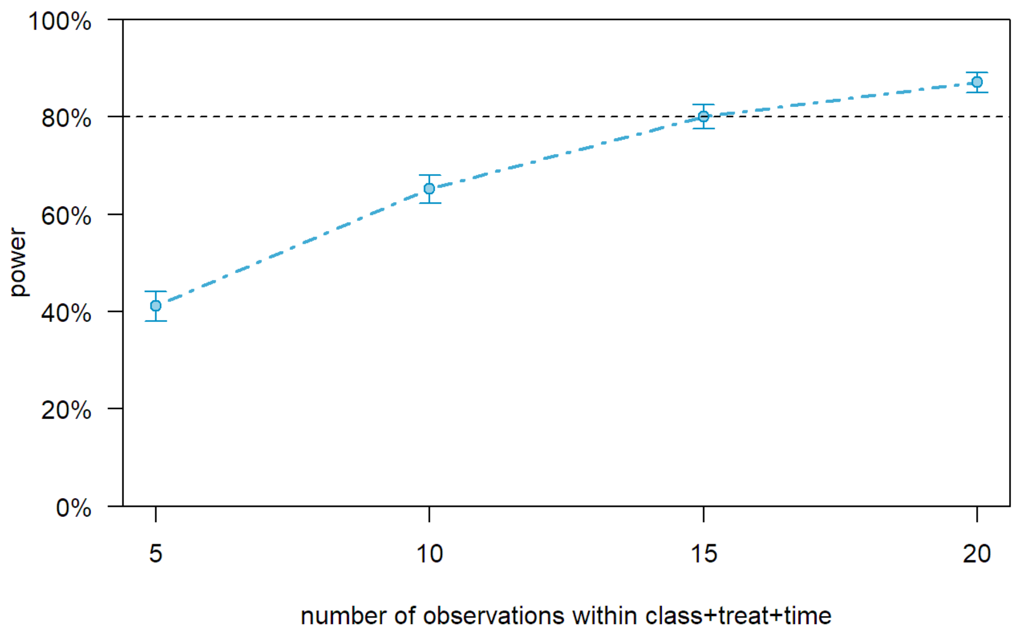
```
model_small <- modelfixef(model_small)['ContextUK'] <- -0.5

powerSim(model_small, nsim=20, test = fixed(xname="ContextUK", method = "t"))


Power for predictor 'ContextUK', (95% confidence interval):==========================|

      75.00% (50.90, 91.34)


Test: t-test with Satterthwaite degrees of freedom (package lmerTest)

      Effect size for ContextUK is -0.50


Based on 20 simulations, (0 warnings, 0 errors)

alpha = 0.05, nrow = 800


Time elapsed: 0 h 0 m 2 s
```

Calculate necessary sample size to get a desired result

p_curve_treat <-

powerCurve(model, test=fixed(xname="ContextUK"), within="Subject", breaks=c(5,10,15,20)))

plot(p_curve_treat)



Thank you for your attention!