

Linear Mixed-Effects Models (aka Statistics III)

Bernd Figner
b.figner@psych.ru.nl

Week 4: February 24, 2014

1

Today: Models, Models, Models

- **Questions: General, homework-related?**
- **All involved lmer modeling steps**
- **Theoretical and practical considerations**
 - Packages
 - Preparing the data frame
 - Contrast settings
 - Pre-model checks and diagnostics
 - The lme4 syntax
 - Interpreting the output
 - Post-model checks and diagnostics
 - Getting p values
 - Post-hocs
- **Homework/lab session**

2

Important Stuff from Papers

(Baayen et al., 2008; Barr et al., 2013; Barr, 2013)

- **How to specify random effects**
 - Intercepts
 - Slopes
 - Correlation (aka Covariance) terms
- **How to set up your model**
 - Fixed and random effects syntax
 - What should be a:
 - fixed effect
 - random intercept
 - random slope
 - **Non-convergence**

3

Overview

Important Data Analysis/Modeling Steps

recommended not only for mixed models

4

0. What packages do we need?

Matrix '1.1.1.1' (or higher)

```
install.packages('Matrix', repos='http://cran.us.r-project.org')
```

pbkrtest '0.3.8' (or higher)

```
install.packages('pbkrtest', repos='http://cran.us.r-project.org')
```

lme4 '1.1.2' (or higher)

```
install.packages('lme4', repos = c("http://lme4.r-forge.r-project.org/repos", getOption("repos")))
```

5

```
install.packages("car") → Anova()
```

```
install.packages("boot") → boot.ci()
```

```
install.packages("reshape") → melt()
```

```
install.packages("lattice") → densityplot()
```

```
install.packages("lsmeans") → lsmeans()
```

```
install.packages("multcomp") → glht()
```

6

1. Preparatory Steps

- Read/import data
- Reshaping data (if necessary) → **long** format
- Merging with other data frames (if necessary)
- Centering or scaling of continuous variables
- Coding of categorical (=unordered) and ordered factors
- Contrast settings

7

2. Understand the raw data

- Look at descriptives (means, medians, range, ...)
- Plot raw data
 - Distribution of dependent variable (normal? transform?)
 - Aggregated (e.g., group differences)
 - Individual differences (in case of repeated measures)
 - Similar means across participants or big differences?
→ random intercepts
 - Do within-subject effects look similar across participants
or do participants differ a lot? → random slopes

8

3. Set up your mixed-model

- **Which fixed effects?** Main effects, interactions
- **Which random effects? Barr et al. (2013)**
 - **Theoretical** = "design-driven"
 - based on experimental design
 - **Empirical** = "data-driven"
 - Generally not recommended (but common in some fields)
 - **Exception:** non-convergence can force simplification
 - Should be principled!
 - Barr et al. (2013): concrete step-by-step guidelines

9

4. Run and check your model

- `summary(mymodel)`
- **Everything ok?**
 - Converged without warnings or errors
 - Number of observations, number of groups ok?
- **Check some diagnostic plots**
 - `plot(mymodel)` → fitted vs. residuals
 - `densityplot(resid(mymodel))` → distribution of the residuals
 - `plot(DV, fitted(mymodel))` → raw vs. fitted

10

5. Significance tests

- **Typically only for fixed effects of interest**
- **Several different options**
- Tests of **coefficients** vs. test of **effects**
- **Coefficients** → like regression: coefficient significantly different from 0?
- **Effects** → like ANOVA: Is whole predictor significant?
 - factor with more than 2 levels
 - whole interaction term
- Post-hoc tests

11

6. Report model in text and figures

- Describe model in sufficient detail
- Report results
- Create plots showing the results
 - Plot raw data?
 - Plot model results? (e.g., marginal means)

12

Mixed-model analysis step-by-step

Bill's repeated-measure ANOVA example

1. Preparatory Steps

- Read/import data
- Reshaping data (if necessary) → **long** format
- Merging with other data frames (if necessary)
- Centering or scaling of continuous variables
- Coding of categorical (=unordered) and ordered factors
- Contrast settings

1. Preparatory Steps

- Read/import data
- Reshaping data (if necessary) → **long** format
- Merging with other data frames (if necessary)
- Centering or scaling of continuous variables
- Coding of categorical (=unordered) and ordered factors
- Contrast settings

15

Creating the data frame

```

participant<-gl(20, 9, labels = c("P01", "P02", "P03", "P04", "P05",
  "P06", "P07", "P08", "P09", "P10", "P11", "P12", "P13", "P14",
  "P15", "P16", "P17", "P18", "P19", "P20" ))
drink<-gl(3, 3, 180, labels = c("Beer", "Wine", "Water"))
imagery<-gl(3, 1, 180, labels = c("Positive", "Negative", "Neutral"))
groups<-gl(9, 1, 180, labels = c("beerpos", "beerneg", "beerneut",
  "winepos", "wineneg", "wineneut", "waterpos", "waterneg",
  "waterneut"))
attitude<-c(1, 6, 5, 38, -5, 4, 10, -14, -2, 26, 27, 27, 23, -15, 14,
  21, -6, 0, 1, -19, -10, 28, -13, 13, 33, -2, 9, 7, -18, 6, 26, -16,
  19, 23, -17, 5, 22, -8, 4, 34, -23, 14, 21, -19, 0, 30, -6, 3, 32,
  -22, 21, 17, -11, 4, 40, -6, 0, 24, -9, 19, 15, -10, 2, 15, -9, 4,
  29, -18, 7, 13, -17, 8, 20, -17, 9, 30, -17, 12, 16, -4, 10, 9, -12,
  -5, 24, -15, 18, 17, -4, 8, 14, -11, 7, 34, -14, 20, 19, -1, 12, 43,
  30, 8, 20, -12, 4, 9, -10, -13, 15, -6, 13, 23, -15, 15, 29, -1, 10,
  15, 15, 12, 20, -15, 6, 6, -16, 1, 40, 30, 19, 28, -4, 0, 20, -10,
  2, 8, 12, 8, 11, -2, 6, 27, 5, -5, 17, 17, 15, 17, -6, 6, 9, -6,
  -13, 30, 21, 21, 15, -2, 16, 19, -20, 3, 34, 23, 28, 27, -7, 7, 12,
  -12, 2, 34, 20, 26, 24, -10, 12, 12, -9, 4)
longAttitude<-data.frame(participant, drink, imagery, groups, attitude)

```


Data Structure

- 20 participants
- DV: attitude; continuous: range -23 to +43
- Each pp contributes 9 data points: 3 x 3 design
 - drink: Beer, Water, Wine
 - imagery: positive, negative, neutral
- Each of the 9 unique factor combinations is measured once in each participant (no replication)

17

```

longAttitude[1:20,]
  participant drink  imagery  groups attitude
1          P01  Beer  Positive  beerpos      1
2          P01  Beer  Negative  beerneg      6
3          P01  Beer   Neutral  beerneut      5
4          P01  Wine  Positive  winepos     38
5          P01  Wine  Negative  wineneg     -5
6          P01  Wine   Neutral  wineneut      4
7          P01  Water  Positive  waterpos     10
8          P01  Water  Negative  waterneg    -14
9          P01  Water   Neutral  waterneut     -2
10         P02  Beer  Positive  beerpos     26
11         P02  Beer  Negative  beerneg     27
12         P02  Beer   Neutral  beerneut     27
13         P02  Wine  Positive  winepos     23
14         P02  Wine  Negative  wineneg    -15
15         P02  Wine   Neutral  wineneut     14
16         P02  Water  Positive  waterpos     21
17         P02  Water  Negative  waterneg     -6
18         P02  Water   Neutral  waterneut      0
19         P03  Beer  Positive  beerpos      1
20         P03  Beer  Negative  beerneg    -19

```

```
with(longAttitude, table(imagery, drink))
```

	drink		
imagery	Beer	Wine	Water
Positive	20	20	20
Negative	20	20	20
Neutral	20	20	20

19

```
with(longAttitude, table(imagery, drink, participant))
```

```
, , participant = P01
      drink
imagery  Beer Wine Water
Positive  1    1    1
Negative  1    1    1
Neutral   1    1    1
```

```
, , participant = P02
      drink
imagery  Beer Wine Water
Positive  1    1    1
Negative  1    1    1
Neutral   1    1    1
```

```
, , participant = P03
      drink
imagery  Beer Wine Water
Positive  1    1    1
Negative  1    1    1
Neutral   1    1    1
```

```
, , participant = P04
      drink
imagery  Beer Wine Water
Positive  1    1    1
Negative  1    1    1
Neutral   1    1    1
```

20

1. Preparatory Steps

- Read/import data
- Reshaping data (if necessary) → **long** format
- Merging with other data frames (if necessary)
- Centering or scaling of continuous variables
- Coding of categorical (=unordered) and ordered factors
- Contrast settings

21

Factor Coding

- Both factors (drink, imagery) are categorical
- Let's make them explicit factors (not strictly necessary here, but good practice)

```
longAttitude$f_imagery <- as.factor(longAttitude$imagery)
longAttitude$f_drink <- as.factor(longAttitude$drink)
```

What contrast setting? (aka contrast coding)

22

Contrast Settings

R Default

- treatment for unordered factors (=categorical)
- polynomial for ordered factors

```
options(contrasts=c("contr.treatment", "contr.poly"))
```

My recommendation (based on Barr et al., 2013)

- "sum-to-zero" (=deviation) for unordered factors
- polynomial for ordered factors

```
options(contrasts=c("contr.sum", "contr.poly"))
```

I have that statement virtually always at the top of each script

23

Treatment coding

```
contrasts(longAttitude$f_drink)
```

	Wine	Water
Beer	0	0
Wine	1	0
Water	0	1

Sum-to-zero (deviation) coding

```
contrasts(longAttitude$f_drink)
```

	[,1]	[,2]
Beer	1	0
Wine	0	1
Water	-1	-1

24

Does it matter?

For the lmer models

- Typically not
- BUT: Barr et al. (2013): in rare cases, it **does** matter
→ Deviation coding recommended!

Interpretation

- Significance of factor as a whole: Doesn't matter
- Significance of coefficients associated with a factor
(number coeffs = factor levels – 1): **DOES** matter!

25

Interpretation

"Naive" interpretation is wrong!

Treatment coding

	Wine	Water
Beer	0	0
Wine	1	0
Water	0	1

Naive (wrong!) interpretation

- First contrast compares Wine against two others
- Second contrast: Water versus two others

→ **Wrong!!**

26

Correct Interpretation

(1) Save your contrast matrix into a variable

```
Treat_cont_f_drink <- contrasts(longAttitude$f_drink)
```

(2) Add a first column of 1's (for the intercept)

```
Treat_cont_f_drink <- cbind(c(1,1,1), Treat_cont_f_drink)
```

		Wine	Water
Beer	1	0	0
Wine	1	1	0
Water	1	0	1

(3) Invert (=solve) the matrix, to get the actual weights

```
solve(Treat_cont_f_drink)
```

	Beer	Wine	Water
	1	0	0
Wine	-1	1	0
Water	-1	0	1

27

Correct Interpretation cont.

```
solve(Treat_cont_f_drink)
```

	Beer	Wine	Water
	1	0	0
Wine	-1	1	0
Water	-1	0	1

28

Correct Interpretation cont.

```
solve(Treat_cont_f_drink)
      Beer Wine Water
Wine    -1    1    0
Water   -1    0    1
```

First row

- Intercept
- Category Beer

29

Correct Interpretation cont.

```
solve(Treat_cont_f_drink)
      Beer Wine Water
      1    0    0
Wine   -1    1    0
Water  -1    0    1
```

Second row (Contrast "Wine")

- Comparison Wine (+1) versus Beer (-1)

30

Correct Interpretation cont.

```
solve(Treat_cont_f_drink)
      Beer Wine Water
      1    0    0
Wine  -1    1    0
Water -1    0    1
```

Third row (contrast "Water")

- Comparison Water (+1) versus Beer (-1)

31

What about Deviation Coding?

```
options(contrasts=c("contr.sum", "contr.poly"))
Dev_cont_f_drink <- contrasts(longAttitude$f_drink)
Dev_cont_f_drink <- cbind(c(1,1,1), Dev_cont_f_drink)
solve(Dev_cont_f_drink)
```

```
      Beer      Wine      Water
[1,] 0.3333333 0.3333333 0.3333333
[2,] 0.6666667 -0.3333333 -0.3333333
[3,] -0.3333333 0.6666667 -0.3333333
```

32

What about Deviation Coding?

```
options(contrasts=c("contr.sum", "contr.poly"))
Dev_cont_f_drink <- contrasts(longAttitude$f_drink)
Dev_cont_f_drink <- cbind(c(1,1,1), Dev_cont_f_drink)
solve(Dev_cont_f_drink)
```

	Beer	Wine	Water
[1,]	0.3333333	0.3333333	0.3333333
[2,]	0.6666667	-0.3333333	-0.3333333
[3,]	-0.3333333	0.6666667	-0.3333333

First row

- Intercept
- Average across all 3 drinks

33

What about Deviation Coding?

```
options(contrasts=c("contr.sum", "contr.poly"))
Dev_cont_f_drink <- contrasts(longAttitude$f_drink)
Dev_cont_f_drink <- cbind(c(1,1,1), Dev_cont_f_drink)
solve(Dev_cont_f_drink)
```

	Beer	Wine	Water
[1,]	0.3333333	0.3333333	0.3333333
[2,]	0.6666667	-0.3333333	-0.3333333
[3,]	-0.3333333	0.6666667	-0.3333333

Second row (contrast 1!)

- Beer (+2/3) compared to the average of Wine and Water (both -1/3)

34

What about Deviation Coding?

```
options(contrasts=c("contr.sum", "contr.poly"))
Dev_cont_f_drink <- contrasts(longAttitude$f_drink)
Dev_cont_f_drink <- cbind(c(1,1,1), Dev_cont_f_drink)
solve(Dev_cont_f_drink)
```

	Beer	Wine	Water
[1,]	0.3333333	0.3333333	0.3333333
[2,]	0.6666667	-0.3333333	-0.3333333
[3,]	-0.3333333	0.6666667	-0.3333333

Third row (contrast 2!)

- Wine (+2/3) compared to the average of Beer and Water (both -1/3)

35

2. Understand the raw data

- Look at descriptives (means, medians, range, ...)
- Plot raw data
 - Distribution of dependent variable (normal? transform?)
 - Aggregated (e.g., group differences)
 - Individual differences (in case of repeated measures)
 - Similar means across participants or big differences?
 - random intercepts
 - Do within-subject effects look similar across participants or do participants differ a lot?
 - random slopes

36

Some Descriptives

```
library(psych)
describe(longAttitude)
```

	var	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se
participant*	1	180	10.50	5.78	10.5	10.50	7.41	1	20	19	0.00	-1.23	0.43
drink*	2	180	2.00	0.82	2.0	2.00	1.48	1	3	2	0.00	-1.52	0.06
imagery*	3	180	2.00	0.82	2.0	2.00	1.48	1	3	2	0.00	-1.52	0.06
groups*	4	180	5.00	2.59	5.0	5.00	2.97	1	9	8	0.00	-1.25	0.19
attitude	5	180	7.89	15.37	8.0	7.89	17.79	-23	43	66	-0.02	-0.86	1.15
f_imagery*	7	180	2.00	0.82	2.0	2.00	1.48	1	3	2	0.00	-1.52	0.06
f_drink*	8	180	2.00	0.82	2.0	2.00	1.48	1	3	2	0.00	-1.52	0.06

Factors indicated by *

→ descriptives don't make that much sense for factors...

37

```
with(longAttitude, describeBy(attitude, group =
list(imagery, drink), mat = TRUE))
```

item	group1	group2	var	n	mean	sd	median	trimmed	mad	min	max	range	skew	kurtosis	se	
11	1	Positive	Beer	1	20	21.05	13.007993	18.5	21.0000	16.3086	1	43	42	0.13940484	-1.29076740	2.908676
12	2	Negative	Beer	1	20	4.45	17.303711	0.0	4.1250	23.7216	-19	30	49	0.12623981	-1.66479365	3.869227
13	3	Neutral	Beer	1	20	10.00	10.295630	8.0	10.0000	7.4130	-10	28	38	0.16850893	-0.76357245	2.302173
14	4	Positive	Wine	1	20	25.35	6.737757	25.0	25.5625	6.6717	11	38	27	-0.19313927	-0.57484950	1.506608
15	5	Negative	Wine	1	20	-12.00	6.181466	-13.5	-11.9375	5.9304	-23	-2	21	0.08001792	-1.09341404	1.382218
16	6	Neutral	Wine	1	20	11.65	6.243101	12.5	11.7500	8.8956	0	21	21	-0.12751348	-1.36166606	1.396000
17	7	Positive	Water	1	20	17.40	7.074044	17.0	16.9375	6.6717	6	33	27	0.43047448	-0.62080268	1.581804
18	8	Negative	Water	1	20	-9.20	6.802476	-10.0	-9.3125	8.8956	-20	5	25	0.17181762	-0.96767898	1.521080
19	9	Neutral	Water	1	20	2.35	6.838552	2.5	3.1875	5.1891	-13	12	25	-0.82311850	0.06250553	1.529147

Number observations and means per cell in the 3x3 design

- Means can be used for plotting
- SEs are wrong! They don't take into account repeated-measures nature

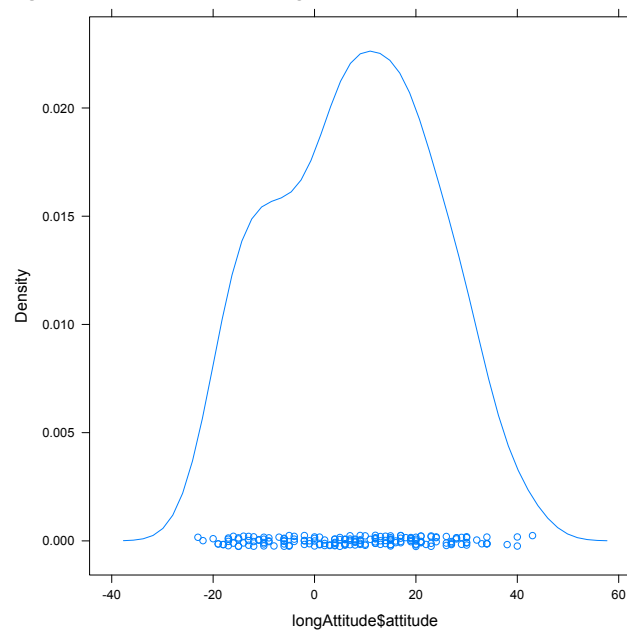
38

2. Understand the raw data

- Look at descriptives (means, medians, range, ...)
- Plot raw data
 - Distribution of dependent variable (normal? transform?)
 - Aggregated (e.g., group differences)
 - Individual differences (in case of repeated measures)
 - Similar means across participants or big differences?
 - random intercepts
 - Do within-subject effects look similar across participants
 - or do participants differ a lot? → random slopes

39

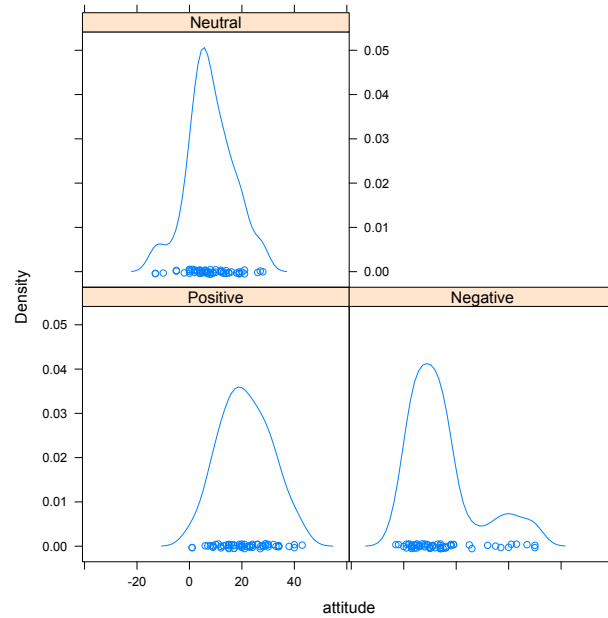
```
library(lattice)
with(longAttitude, densityplot(attitude))
```



40

Separate per condition

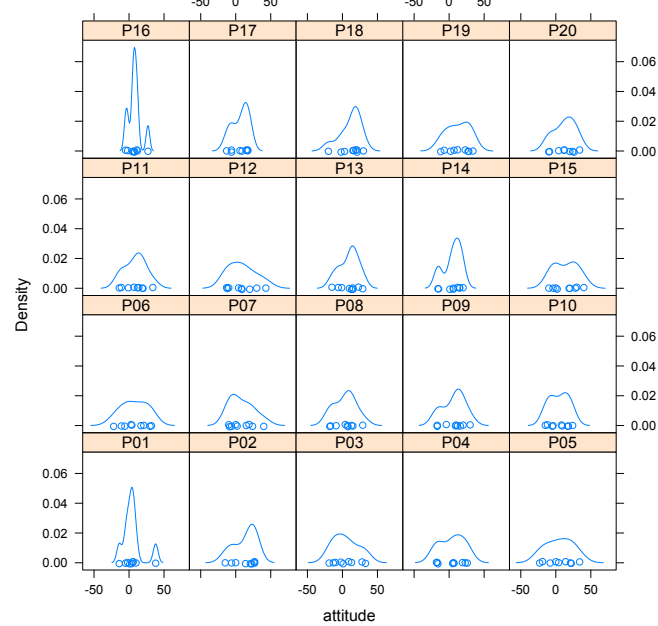
```
with(longAttitude, densityplot(~attitude | imagery))
```



41

Or per participant

```
with(longAttitude, densityplot(~attitude | participant))
```



42

3. Set up your mixed-model

- **Which fixed effects?** Main effects, interactions
- **Which random effects? Barr et al. (2013)**
 - **Theoretical** = "design-driven"
 - based on experimental design
 - **Empirical** = "data-driven"
 - Generally not recommended (but common in some fields)
 - **Exception:** non-convergence can force simplification
 - Should be principled!
 - Barr et al. (2013): concrete step-by-step guidelines

43

3. Set up your mixed-model

- **Which fixed effects? Main effects, interactions**
- **Which random effects? Barr et al. (2013)**
 - **Theoretical** = "design-driven"
 - based on experimental design
 - **Empirical** = "data-driven"
 - Generally not recommended (but common in some fields)
 - **Exception:** non-convergence can force simplification
 - Should be principled!
 - Barr et al. (2013): concrete step-by-step guidelines

44

Fixed Effects

- Typically effects of interest
- Those, we want to test for significance
 - Main effect of imagery
 - Main effect of drink
 - Interaction between imagery and drink

Remember?

- $a + b \rightarrow$ main effect of a and main effect of b
- $a : b \rightarrow$ the interaction of a and b
- $a * b \rightarrow$ main effects a and b AND interaction
the same as $a + b + a : b$

45

3. Set up your mixed-model

- Which fixed effects? Main effects, interactions
- Which random effects? Barr et al. (2013)
 - Theoretical = "design-driven"
→ based on experimental design
 - Empirical = "data-driven"
 - Generally not recommended (but common in some fields)
 - Exception: non-convergence can force simplification
 - Should be principled!
 - Barr et al. (2013): concrete step-by-step guidelines

46

Random Effects

- **What's our "unit," our "grouping" variable?**
 - Do we have 1 or more grouping variables?
 - Only 1: participant → random per-participant intercept
- **Which effects are within units? in our case, within-subject**
 - drink, imagery, interaction drink x imagery
 - Thus, random per-participant slopes for all 3
 - **Sure?**
 - No "replication:" each participant only 1 observation per 3 x 3 cell!
 - No random slope for the interaction

47

Random Effects cont.

- **All random effects...**
 - Per-participant intercept
 - Per-participant "drink" slope
 - Per-participant "imagery" slope
- **... have the same grouping unit**
- **Random correlation (aka covariance) terms among all random effects**

48

A quick detour:

Random Effects Syntax

Random Effects Syntax

Always in parentheses and with a „|“ somewhere in there, like (xxx | yyy)

For the examples on the following slides

- **group**: a grouping factor (e.g., participant)
- **pred**: a predictor (e.g., drink; could also be a continuous variable, doesn't matter)

See: <http://glmm.wikidot.com/faq>
[scroll down to „Model specification etc“]

R Syntax	What it stands for
(1 group)	random group intercept
(pred group)	random slope of pred per group AND correlated random group intercept
(1+pred group)	same as the one before
(0+pred group)	random slope of pred per group WITHOUT random group intercept
(0+pred group) + (1 group)	random slope of pred per group AND uncorrelated random group intercept
pred + (0+pred group) + (1 group)	fixed slope of pred which is allowed to vary over group AND uncorrelated random group intercept
(1 group) + (1 group2)	intercept varying among crossed random effects group and group2

**End of the detour,
back to our model**

Our lmer model

Fixed effects

- drink, imagery, their interaction

• Random effects

- Random per-participant intercept
- Random per-participant slope for drink
- Random per-participant slope for imagery
- Random correlations between all random effects

```
att_1 <- lmer(attitude ~ f_imagery *
f_drink + (1 + f_imagery + f_drink |
participant), data = longAttitude)
```

53

lmer: General Syntax

```
MyModel <- lmer(DV ~ IV1 + IV2 + (1 + IV1
| GroupingFactor), data = MyData)
```

Maximum number of iterations (e.g., 30,000)

```
MyModel <- lmer(DV ~ IV1 [...], control =
lmerControl(optCtrl = list(maxfun = 30000))
```

Optimizer (e.g., bobyqa)

```
MyModel <- lmer(DV ~ IV1 [...], control =
lmerControl(optimizer = "bobyqa"))
```

54

Our lmer model

```
att_1 <- lmer(attitude ~ f_imagery *  
f_drink + (1 + f_imagery + f_drink |  
participant), data = longAttitude)
```

```
summary(att_1)
```

55

4. Run and check your model

- `summary(mymodel)`
- **Everything ok?**
 - Converged without warnings or errors
 - Number of observations, number of groups ok?
- **Check some diagnostic plots**
 - `plot(mymodel)` → fitted vs. residuals
 - `densityplot(resid(mymodel))` → distribution of the residuals
 - `plot(DV, fitted(mymodel))` → raw vs. fitted

56

```

Linear mixed model fit by REML ['lmerMod']
Formula: attitude ~ f_imagery * f_drink + (1 + f_imagery + f_drink | participant)
Data: longAttitude

REML criterion at convergence: 1206.6

Scaled residuals:
    Min       1Q   Median       3Q      Max
-2.33976 -0.58251 -0.00033  0.68843  3.12033

Random effects:
Groups           Name          Variance Std.Dev. Corr
participant (Intercept)  8.049    2.837
              f_imagery1  5.702    2.388   -0.41
              f_imagery2 23.694    4.868    0.47 -0.95
              f_drink1   75.921    8.713    0.85 -0.50  0.68
              f_drink2   12.560    3.544   -0.94  0.64 -0.73 -0.94
Residual                33.401    5.779
Number of obs: 180, groups: participant, 20

Fixed effects:
              Estimate Std. Error t value
(Intercept)    7.8944    0.7668  10.295
f_imagery1     13.3722    0.8101  16.507
f_imagery2    -13.4778    1.2473 -10.805
f_drink1        3.9389    2.0414   1.930
f_drink2        0.4389    0.9996   0.439
f_imagery1:f_drink1 -4.1556    0.8615  -4.823
f_imagery2:f_drink1  6.0944    0.8615   7.074
f_imagery1:f_drink2  3.6444    0.8615   4.230
f_imagery2:f_drink2 -6.8556    0.8615  -7.957

Correlation of Fixed Effects:
(Intr) f_mgr1 f_mgr2 f_drn1 f_drn2 f_1:_1 f_2:_1 f_1:_2
f_imagery1 -0.222
f_imagery2  0.340 -0.728
f_drink1    0.675 -0.314  0.562
f_drink2   -0.619  0.333 -0.503 -0.802
f_mgr1:f_1  0.000  0.000  0.000  0.000  0.000
f_mgr2:f_1  0.000  0.000  0.000  0.000  0.000 -0.500
f_mgr1:f_2  0.000  0.000  0.000  0.000  0.000 -0.500  0.250
f_mgr2:f_2  0.000  0.000  0.000  0.000  0.000  0.250 -0.500 -0.500

```

57

```

Linear mixed model fit by REML ['lmerMod']
Formula: attitude ~ f_imagery * f_drink + (1 +
f_imagery + f_drink | participant)
Data: longAttitude

```

REML criterion at convergence: 1206.6

```

Scaled residuals:
    Min       1Q   Median       3Q      Max
-2.33976 -0.58251 -0.00033  0.68843  3.12033

```

Note: If you want to use ML instead of REML

```

att_1_ML <- lmer(attitude ~ f_imagery * f_drink + (1 +
f_imagery + f_drink | participant), data = longAttitude,
REML = FALSE)

```

58

Random effects:

Groups	Name	Variance	Std.Dev.	Corr
participant	(Intercept)	8.049	2.837	
	f_imagery1	5.702	2.388	-0.41
	f_imagery2	23.694	4.868	0.47 -0.95
	f_drink1	75.921	8.713	0.85 -0.50 0.68
	f_drink2	12.560	3.544	-0.94 0.64 -0.73 -0.94
Residual		33.401	5.779	

Number of obs: 180, groups: participant, 20

	Beer	Wine	Water	
[1,]	0.3333333	0.3333333	0.3333333	
[2,]	0.6666667	-0.3333333	-0.3333333	→ f_drink1
[3,]	-0.3333333	0.6666667	-0.3333333	→ f_drink2

59

Fixed effects:

	Estimate	Std. Error	t value
(Intercept)	7.8944	0.7668	10.295
f_imagery1	13.3722	0.8101	16.507
f_imagery2	-13.4778	1.2473	-10.805
f_drink1	3.9389	2.0414	1.930
f_drink2	0.4389	0.9996	0.439
f_imagery1:f_drink1	-4.1556	0.8615	-4.823
f_imagery2:f_drink1	6.0944	0.8615	7.074
f_imagery1:f_drink2	3.6444	0.8615	4.230
f_imagery2:f_drink2	-6.8556	0.8615	-7.957

Notes

- R default: type 1 → Order matters
- no *p* values
- for large data sets: rule of thumb $t > 2$

60

```

Correlation of Fixed Effects:
(Intr) f_mgr1 f_mgr2 f_drn1 f_drn2 f_1:_1 f_2:_1 f_1:_2
f_imagery1 -0.222
f_imagery2  0.340 -0.728
f_drink1    0.675 -0.314  0.562
f_drink2   -0.619  0.333 -0.503 -0.802
f_mgr1:f_1  0.000  0.000  0.000  0.000  0.000
f_mgr2:f_1  0.000  0.000  0.000  0.000  0.000 -0.500
f_mgr1:f_2  0.000  0.000  0.000  0.000  0.000 -0.500  0.250
f_mgr2:f_2  0.000  0.000  0.000  0.000  0.000  0.250 -0.500 -0.500

```

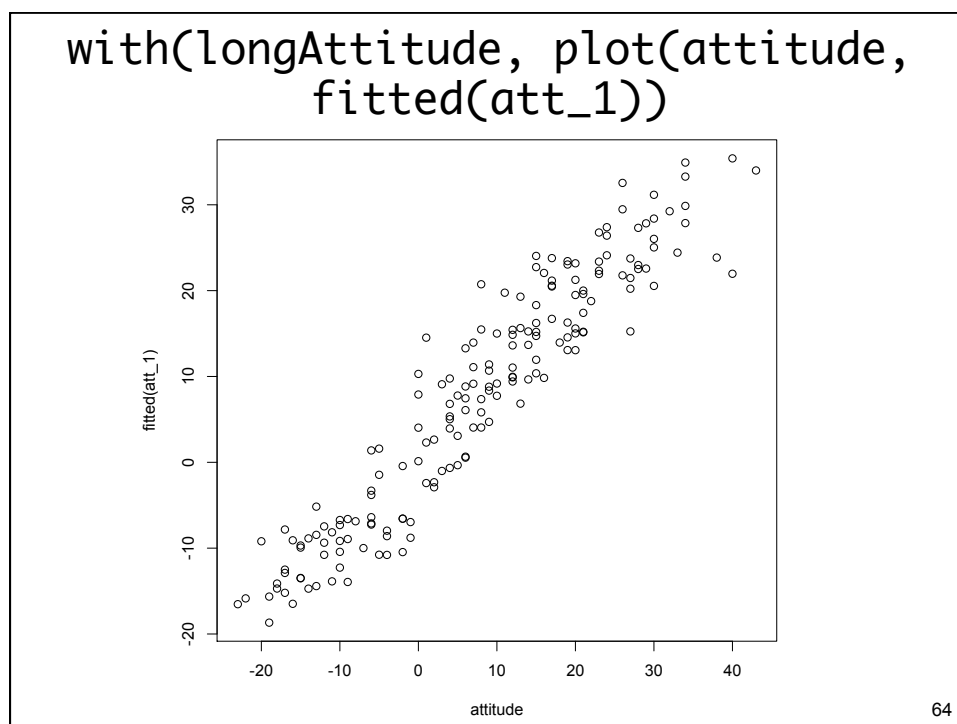
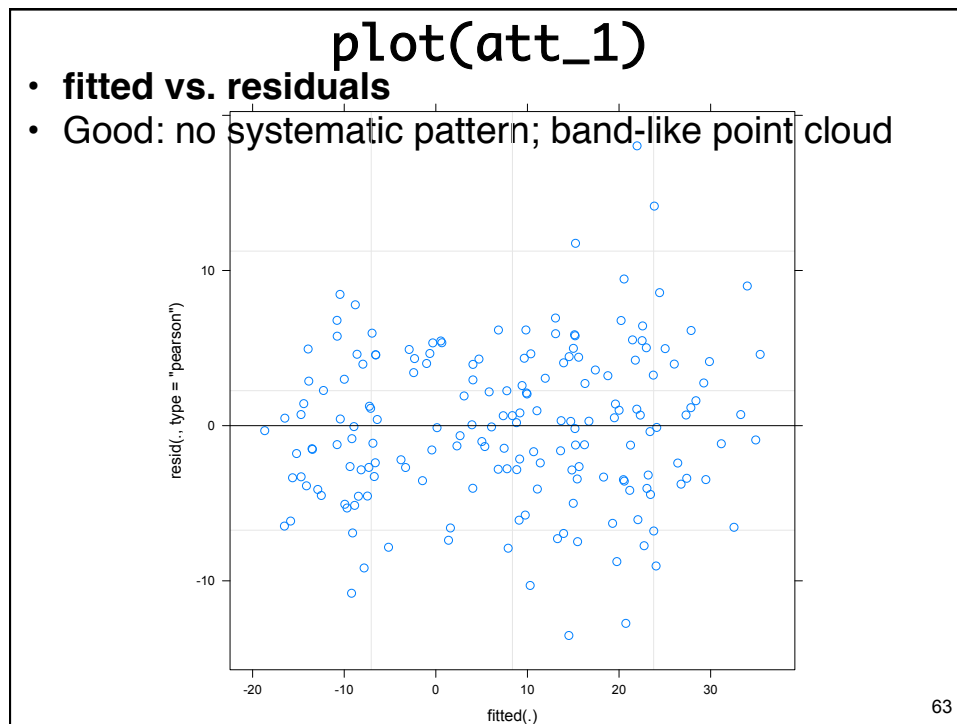
- What is it??!! Nobody talks about it...
- It is something ... complicated. It can be related to multicollinearity.
- Douglas Bates' explanation:
<https://stat.ethz.ch/pipermail/r-sig-mixed-models/2009q1/001941.html>
- Take-home message: Ignore it
`print(summary(mymodel), corr = FALSE)`

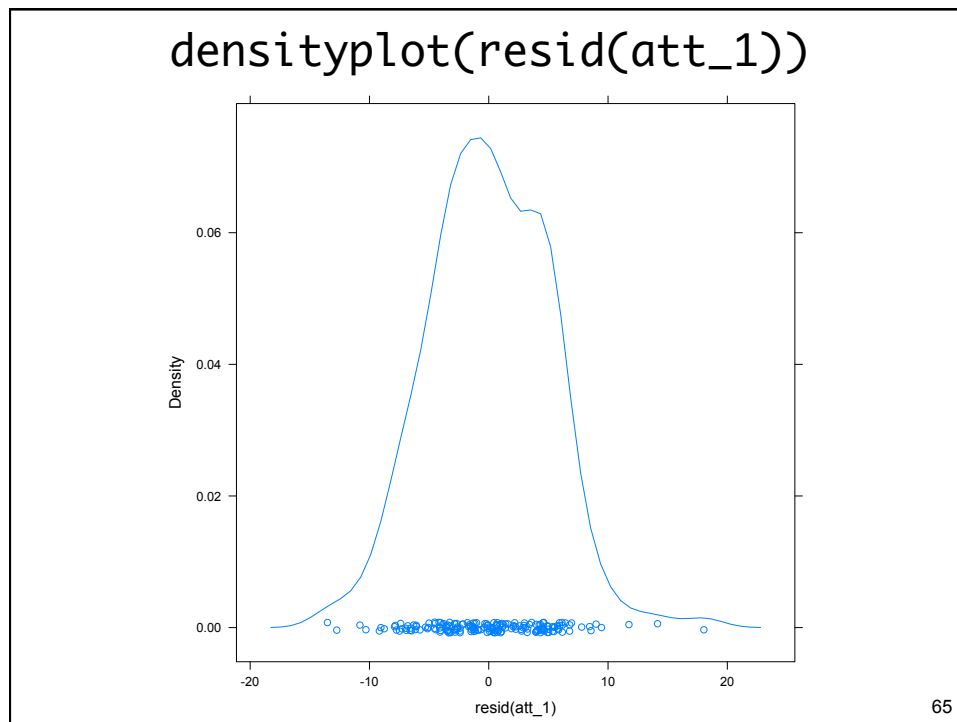
61

4. Run and check your model

- Check `summary(mymodel)`
- Everything ok?
 - Converged without warnings or errors
 - Number of observations, number of groups ok?
- Check some diagnostic plots
 - `plot(mymodel)` → fitted vs. residuals
 - `densityplot(resid(mymodel))` → distribution of the residuals
 - `plot(DV, fitted(mymodel))` → raw vs. fitted

62





5. Significance tests

- **Typically only for fixed effects of interest**
- **Several different options**
- Tests of **coefficients** vs. test of **effects**
- **Coefficients** → like regression: coefficient significantly different from 0?
- **Effects** → like ANOVA: Is whole predictor significant?
 - factor with more than 2 levels
 - whole interaction term
- Post-hoc tests

66

Tests of Coefficients and „Effects“

- **Example: 3 experimental conditions**
 - neutral
 - positive mood induction
 - negative mood induction
- **Effects:** familiar from ANOVA framework
 - Is the **whole factor** significant?
→ „Effect of mood induction“
 - Similarly: interaction between categorical factors: Is **whole interaction** significant?

67

- **Coefficients:** familiar from regression framework
 - Is the coefficient significantly different from 0?
 - Continuous predictors (→ 1 df)
 - Interactions between continuous predictors (→ 1 df)
 - Factors with 2 levels (→ 1 df)
 - **Factors with more levels** (mood induction example)
 - Neutral condition vs average of positive and negative mood significant?
 - Positive vs negative mood condition significant?
 - ...
- **How you set up contrasts matters!**

68

Getting p values

→ Different methods (usually similar results)

(Douglas Bates: <https://stat.ethz.ch/pipermail/r-help/2006-May/094765.html>)

List ordered according to recommendations

<http://glmm.wikidot.com/faq>

69

<http://glmm.wikidot.com/faq>

What is the best way to test hypotheses on effects in GLMMs?

- This question is not completely answerable, but there is generally a tradeoff between computational difficulty and accuracy.

70

Three "most recommended" ones

(1) (non)Parametric bootstrap

- coefficients: `bootMer()` → `boot.ci()`
- effects: `PBmodcomp()` (package `pbkrtest`)

(2) Conditional F-tests with df correction

- `Anova(..., test="F")` (package `car`)
- `KRmodcomp()` (package `pbkrtest`)

(3) Likelihood Ratio Tests

- `anova()` or `drop1()`

71

Conditional F test with df correction (Kenward-Roger)

```
library(car)
```

```
Anova(mymodel, type = 3, test = "F")
```

To get type 2 test: `type = 2`

`Anova()` calls `KRmodcomp` from `pbkrtest`

```
KRmodcomp(LargeModel, SmallModel)
```

`SmallModel`: same as `LargeModel`, but without the fixed effect of interest

Notes

- Typically fast; widely accepted
- NOT available for generalized mixed models

72

Repeated-measures ANOVA Example

```
Anova(att_1, type = 3, test = "F")
```

[approx. 7 sec on this laptop]

Analysis of Deviance Table (Type III Wald F tests with Kenward-Roger df)

Response: attitude

	F	Df	Df.res	Pr(>F)
(Intercept)	105.9886	1	19	3.288e-09 ***
f_imagery	130.5585	2	18	1.929e-11 ***
f_drink	6.9885	2	18	0.005674 **
f_imagery:f_drink	19.6430	4	76	3.906e-11 ***

73

Bootstrap methods: bootMer()

```
library(lme4)
```

(1) Run your lmer model (mymodel)

(2) Define a function (only once per R session)

```
FUN_bootMer <- function(fit) {
  return(fixef(fit))
}
```

(3) Run bootstrap (can take a while!)

```
boot_mymodel <- bootMer(mymodel, FUN_bootMer,
  nsim = 1000, type = "parametric", .progress =
  "txt", PBargs = list(style = 3))
```

→ Look up ?bootMer

→ Possible to use several cores → much faster!

74

(4) Get Confidence Intervals**(4.1.) look at boot_att_1**

```
head(as.data.frame(boot_att_1))
```

(4.2.) get CIs at boot_att_1

For intercept

```
boot.ci(boot_att_1, index = 1, conf =  
0.95, type=c("norm", "basic", "perc"))
```

99% CI

```
boot.ci(boot_att_1, index = 1, conf =  
0.99, type=c("norm", "basic", "perc"))
```

75

For second column (= first coefficient after intercept)

```
boot.ci(boot_att_1, index = 2, conf =  
0.95, type=c("norm", "basic", "perc"))
```

etc for 3rd column, 4th column,

When a CI does NOT include 0 → significant!

- 95% CI does NOT include 0 → $p < .05$
- 99% CI does NOT include 0 → $p < .01$
- 95% CI DOES include 0 → $ns, p > .05$
- 90% CI DOES include 0 → $ns, p > .10$

76

Bootstrap method to test EFFECTS

- PBmodcomp(LargeModel, SmallModel) from pbkrtest → test of **effects**

Same idea as KRmodcomp

- First, fit model with and without effect of interest
 - Then, compare the fit of the 2 models via PBmodcomp()
- PBmodcomp(LargeModel, SmallModel)

Notes

- Can take quite some time
- Look up ?PBmodcomp → possible to distribute across several clusters! Much faster!!

77

LRTs

Same model comparison idea

- First, fit model with and without effect of interest
- Then, compare the fit of the 2 models via anova()

anova(SmallModel, LargeModel)

Notes

- Often NOT recommended (at least for smaller data sets)
- BUT: Barr et al. (2013) → worked very well!

78

**Reporting results: next
time...**

**Questions,
comments, ...?**

Important Notes re Homework (more details in BB version)

Homework

- **Go through all the steps we covered today**
- **Use the intertemporal choice valuation ratings**
- **Time and Amount: continuous predictors!**
 - Always center continuous predictors
- **Convergence problems**
 - Check your data and model (e.g., outliers? odd participants?)
 - Increase number of iterations
 - Try scaling (= standardizing) predictors

82

Our Data

- **Risky choice**
 - hot and cold version
 - questionnaire items
 - demographics
- **Intertemporal choice**
 - Binary choices, each between an SS and LL
 - **Valuation ratings (heelemaal niet goed/heel goed)**
 - Titrator (table with SS/LL options)
 - questionnaire items

83

Geef voor elk van de onderstaande opties aan hoe goed je die optie vindt.

Let op: Linker en rechter uiteindes van de schaal betekenen:

Helemaal niet goed € 15 over 42 dagen
 Heel goed € 85 vandaag

1. € 78 over 28 dagen

Helemaal niet goed |-----| Heel goed

2. € 18 over 3 dagen

Helemaal niet goed |-----| Heel goed

3. € 50 over 42 dagen

Helemaal niet goed |-----| Heel goed

84

Important steps to do

- Load valuation rating file (→ BlackBoard)
- Load file with item information (time and amount for q1 to q15) (→ BlackBoard)
- Several preprocessing steps necessary
 - Create participant code variable
 - Merge data frames
 - Center (scale) predictors
 - ...
- Do checks before running the model

85

- Run model
- Convergence problems?
 - Data frame and model ok? Outliers, odd participants?
 - Increase number of iterations?
 - Predictors centered? Predictors scaled?
- Do checks after having run the model (plots)
- Get p values
 - Try Anova(..., test = "F")
 - Try others, if you want

86

More details in BB homework slides!!

What you have to hand in

- Sworn statement (as usual...)
- R script with all commands
- Word or pdf document
 - Figures you created (pre and post model)
 - Results that you get
 - Comment things a bit
"Participants showed a significant effect of amount:
As the amount of money increased, the
attractiveness ratings increased (decreased?)
significantly (coeff = xx; $p < .05$)..."

87

That's it for today's lecture!

Questions or comments?

See you in a bit downstairs!

-1.55A/B