Advanced mixed-models workshop: Session 6

Dale Barr

University of Glasgow

Bremen March 2015

Determining Maximal Random Effects

Barr, Levy, Scheepers, & Tily (2013); Barr (2013)

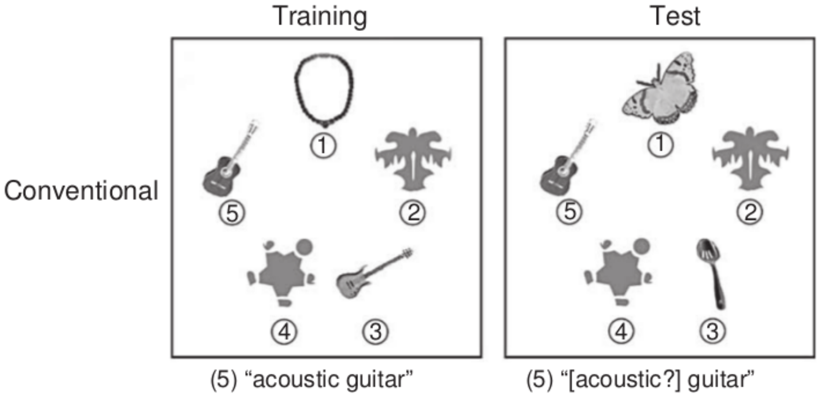
- Random intercept is needed whenever there are multiple observations per unit
- Any within-unit factor gets a random slope, *unless there is only one observation per level per unit*
- Between-unit factors do not get a random slope
- For each interaction, include a slope for the highest order combination of within-subject factors subsumed by the interaction
- For time-series data, include random slopes for time predictors if you have more than one time series per unit

Audience design in language production

Gann & Barr (2014), *Language, Cognition, & Neuroscience*

- How do we explain referential misspecification?
- Retrieval of past descriptions from memory is an obligatory consequence of attending to a referent with a referential goal in mind
  - Instance Theory of Automaticity (Logan, 1988)
- Retrieved descriptions checked against context for adequacy, and re-shaped if necessary

Design



- Addressee: New or Old (Between)
- Novelty (of referent): New or Old
- Feedback: Yes or No

- 16 dyads (1 spkr, 1 addr)
- 16 triads (1 spkr, 2 addr)

Experimental Items

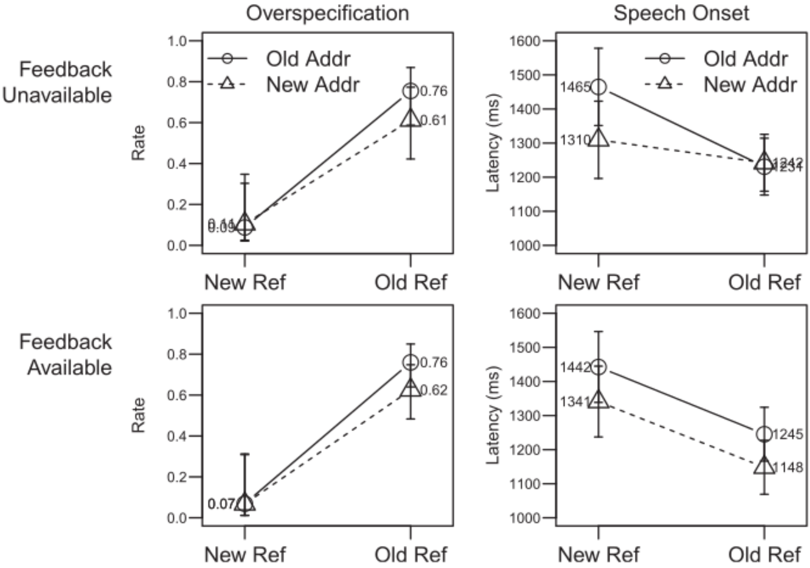
Target	*Competitor*	*Modal response*
Candle	(Not melted) candle	Unmelted candle
Key	(Old-fashioned) key	Modern/new/gold key
Knife	(Swiss Army) knife	Knife with the brown/wooden handle
Trash can	(Metal) trash can	Plastic/white trash can
Spoon	(Large slotted) spoon	Small spoon
Guitar	(Electric) guitar	Acoustic guitar
Carrot	(Cartoon) carrot	Little/real carrot
Gorilla	(Young/brown) gorilla	Black gorilla
Gun	(Toy) gun	Real gun
Leaf	(Dark green three pointed) leaf	Dark green leaf
Rose	(Wilted; stem not visible) rose	Rose with a stem
Marker	Marker (with cap off)	Marker with a cap
Screwdriver	Screwdriver (with black on handle)	Screwdriver, all red handle
Backpack	(Blue) backpack	Purple backpack
Clamp	(Open) clamp	Closed clamp

Before doing analysis: *look at your data*

```
library(lme4)

crefs <- readRDS("crefs.rds")
head(crefs, 3)

crefs_mit <- subset(crefs, ModInTrain)

with(crefs_mit, aggregate(Modifier ~ Novelty+Addressee+Feedback, FUN=mean))
```

```
  SessionID ItemID RespID Novelty Addressee Feedback ModInTrain Modifier  SOT
1        56      2   6314     New       New       No       TRUE        0 1462
2        56      3   6319     New       New       No       TRUE        0 1126
3        56      7   6315     Old       New       No       TRUE        1 1695
  Novelty Addressee Feedback   Modifier
1     New       New       No 0.10526316
2     Old       New       No 0.61224490
3     New       Old       No 0.08771930
4     Old       Old       No 0.75510204
5     New       New      Yes 0.06779661
6     Old       New      Yes 0.62500000
7     New       Old      Yes 0.06896552
8     Old       Old      Yes 0.76000000
```

Plot cell means



Specifying the random effects

	Subjects	Items
Addressee	Between	Within
Novelty	Within	Within
Feedback	Within	Between

```
xtabs(~Addressee+SessionID, crefs)
xtabs(~Addressee+ItemID, crefs)
```

```
         SessionID
Addressee 56 57 60 61 62 63 66 67 68 69 70 71 72 73 74 75 77 78 79 80 81 82 83
      New 15  0 15 15  0  0 15  0 15 15  0 15 15  0  0  0  0  0  0 15  0 15  0
      Old  0 15  0  0 15 15  0 15  0  0 15  0  0 15 15 15 15 15 15  0 15  0 15
         SessionID
Addressee 84 85 89 90 91 92 95 96 97
      New 15 15  0 15 15  0  0 15 15
      Old  0  0 15  0  0 15 15  0  0
         ItemID
Addressee  2  3  7  8 12 13 17 18 22 23 27 28 32 33 37 38
      New 16 16 16  7 16 16 16 16 16 16 16 16 16  9 16 16
      Old 16 16 16  8 16 16 16 16 16 16 16 16 16  8 16 16
```

Creating predictor variables

```
crefs_mit2 <- transform(crefs_mit, N=ifelse(Novelty=="New",1,0),
                        A=ifelse(Addressee=="New",1,0),
                        F=ifelse(Feedback=="Yes",1,0))
crefs_mit2c <- transform(crefs_mit2,
                         Nc=N-mean(N),
                         Ac=A-mean(A),
                         Fc=F-mean(F))

head(crefs_mit2c)
```

```
 SessionID ItemID RespID Novelty Addressee Feedback ModInTrain Modifier  SOT N
1        56      2   6314     New       New       No       TRUE        0 1462 1
2        56      3   6319     New       New       No       TRUE        0 1126 1
3        56      7   6315     Old       New       No       TRUE        1 1695 0
4        56      8   6318     Old       New       No       TRUE        1 1137 0
5        56     13   6382     New       New       No       TRUE        0  986 1
6        56     12   6384     New       New       No       TRUE        0  928 1
  A F         Nc        Ac         Fc
1 1 0  0.4530892 0.5057208 -0.4988558
2 1 0  0.4530892 0.5057208 -0.4988558
3 1 0 -0.5469108 0.5057208 -0.4988558
4 1 0 -0.5469108 0.5057208 -0.4988558
5 1 0  0.4530892 0.5057208 -0.4988558
6 1 0  0.4530892 0.5057208 -0.4988558
```

Specifying the model

Our model:

$$\log\left(\frac{p_{ij}}{1-p_{ij}}\right) = \beta_0 + \beta_1 A + \beta_2 N + \beta_3 F + \beta_4 AN + \beta_5 AF + \beta_6 NF + \beta_7 ANF$$

$$\begin{aligned}
\beta_0 &= \gamma_0 + S_{0i} + I_{0j} \\
(A)\beta_1 &= \gamma_1 + I_{1j} \\
(N)\beta_2 &= \gamma_2 + S_{2i} + I_{2j} \\
(F)\beta_3 &= \gamma_3 + S_{3i} \\
(AN)\beta_4 &= \gamma_4 + I_{4j} \\
(AF)\beta_5 &= \gamma_5 \\
(NF)\beta_6 &= \gamma_6 + S_{6i} \\
(ANF)\beta_7 &= \gamma_7
\end{aligned}$$

	Subjects	Items
Addressee	Between	Within
Novelty	Within	Within
Feedback	Within	Between

```
m1 <- glmer(Modifier ~ Ac*Nc*Fc + (1 + Nc*Fc | SessionID) + (1 + Ac*Nc | ItemID),
                    data=crefs_mit2c, family=binomial(link="logit"),
                    control=glmerControl(optimizer="bobyqa"))
```

Fitting the model

```
# takes a long time but doesn't converge
m1 <- glmer(Modifier ~ Ac*Nc*Fc +
            (1 + Nc*Fc | SessionID) + (1 + Ac*Nc | ItemID),
            crefs_mit2c, family=binomial(link="logit"),
            control=glmerControl(optimizer="bobyqa"))
```

```
:  Warning messages:
: 1: In checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv,  :
:   Model failed to converge with max|grad| = 0.01537 (tol = 0.001, component 20)
: 2: In checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv,  :
:   Model is nearly unidentifiable: very large eigenvalue
:  - Rescale variables?;Model is nearly unidentifiable: large eigenvalue ratio
:  - Rescale variables?
```

Diagonal model (covariance parameters fixed to zero)

```
# no-random-correlations model
m2 <- glmer(Modifier ~ Ac*Nc*Fc +
            (1 + Nc*Fc || SessionID) + (1 + Ac*Nc || ItemID),
            crefs_mit2c, family=binomial(link="logit"),
            control=glmerControl(optimizer="bobyqa")) # converges
```

Diagonal model (output)

```
Random effects:
 Groups       Name        Variance  Std.Dev.
 SessionID    (Intercept) 6.566e-01 8.103e-01
 SessionID.1  Nc          1.132e+00 1.064e+00
 SessionID.2  Fc          0.000e+00 0.000e+00
 SessionID.3  Nc:Fc       4.297e-15 6.555e-08
 ItemID       (Intercept) 1.026e+00 1.013e+00
 ItemID.1     Ac          1.580e-01 3.975e-01
 ItemID.2     Nc          1.216e+00 1.103e+00
 ItemID.3     Ac:Nc       0.000e+00 0.000e+00
Number of obs: 427, groups:  SessionID, 32; ItemID, 16

Fixed effects:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.13298    0.35710  -3.173  0.00151 **
Ac          -0.29116    0.46433  -0.627  0.53063
Nc          -4.32096    0.61720  -7.001 2.54e-12 ***
Fc          -0.38034    0.62448  -0.609  0.54249
Ac:Nc        1.05604    0.78602   1.344  0.17910
Ac:Fc       -0.07195    0.71227  -0.101  0.91954
Nc:Fc       -0.32425    0.89398  -0.363  0.71683
Ac:Nc:Fc    -0.53686    1.33216  -0.403  0.68695
```

Clean up the random effects

- get rid of REPs estimated to be zero because we're using model comparison, and those could slow down the estimation procedure

```
m3 <- glmer(Modifier ~ Ac*Nc*Fc +
                (1 | SessionID) +
                (0 + Nc | SessionID) +
                (1 | ItemID) +
                (0 + Ac | ItemID) +
                (0 + Nc | ItemID),
            crefs_mit2c, family=binomial(link="logit"),
            control=glmerControl(optimizer="bobyqa")) # converges
```

Perform tests using model comparison

```r
m3_noA <- update(m3, . ~ . - Ac)
m3_noN <- update(m3, . ~ . - Nc)
m3_noF <- update(m3, . ~ . - Fc)
m3_noAN <- update(m3, . ~ . - Ac:Nc)
m3_noAF <- update(m3, . ~ . - Ac:Fc)
m3_noNF <- update(m3, . ~ . - Nc:Fc)
m3_noANF <- update(m3, . ~ . - Ac:Nc:Fc)

anova(m3, m3_noA)
anova(m3, m3_noN)
anova(m3, m3_noF)
anova(m3, m3_noAN)
anova(m3, m3_noAF)
anova(m3, m3_noNF)
anova(m3, m3_noANF)
```

	Chisq	Df	p
A	.382	1	.537
N	32.693	1	$<.001$
F	.372	1	.542
AN	1.843	1	.175
AF	.010	1	.920
NF	.131	1	.717
ANF	.162	1	.687

Implications

- Little evidence that the speaker took partner's perspective into account
  - Use of modifier driven by listener's own experience
- Supports idea that modifier use is (at least partly) based on memory retrieval

Final thoughts

- When performing (reviewing) analyses, it is of utmost importance to ensure random effects are appropriately specified
- Random-intercept-only models are rarely appropriate
- Use design-driven rather than data-driven random effects
- Development of `lme4` is rapid, and there are many "tricks of the trade"; tune into blogs, mailing lists, and social media to keep up
- Don't give up! It will make sense... at some point...