SMiP 2024 MLM Example 2

Lesa Hoffman

May 2024

Table of Contents

```
knitr::opts_chunk$set(echo=TRUE)

# Working directory for all chunks below
  knitr::opts_knit$set(root.dir="C:/Dropbox/Papers and Data/SMiP/SMiP_2024_MLM_Part1")
```

Multilevel Models for Subjects Crossed with Items Predicting Log-Transformed Response Time (logRT)

This example will illustrate the estimation and interpretation of multilevel models with crossed random effects for an observed continuous outcome. The data are from chapter 12 of Hoffman (2015) Longitudinal Analysis: Modeling Within-Person Fluctuation and Change, as included in the Example 2 .zip folder. Log-transformed response time (logRT) outcomes for a change detection task (in which subjects must detect a change between two otherwise identical scenes presented using the "flicker paradigm") were collected for 51 items from 153 subjects (total possible observations = 7803; total actual observations = 7646 due to missing logRTs when the change was not detected). Items are natural scenes that varied systematically in two features: relevance to driving of the change (scaled from 0–5, to be centered at 3) and visual salience of the change scaled from 0–5, to be centered at 3). Subjects varied in age, as captured by two predictors: younger versus older adults (older: 0=younger, 1=older), and years over age 65 if older (yrs65, 0=age 65). Given the small samples of subjects and items, we will use REML estimation and Satterthwaite denominator degrees of freedom. The last model is also estimated using Bayesian estimation in Mplus 8.10 (see separate files in Example 2 .zip folder).

Loading Packages

First (below), we set global options to my preferred versions, and then we install and load the R packages to be used.

```r
# Set width of output and number of significant digits printed,
# number of digits before using scientific notation, shut off significance stars
options(width=120, digits=5, scipen=6, show.signif.stars=FALSE)

#####  Check to see if packages are downloaded, install if not, then load  #####

# To import xls or xlsx data as table
if (!require("readxl")) install.packages("readxl"); library(readxl)
## Loading required package: readxl
## Warning: package 'readxl' was built under R version 4.3.1

# To get compact data description
if (!require("psych")) install.packages("psych"); library(psych)
## Loading required package: psych

# To estimate MLMs using gls or lme
if (!require("nlme")) install.packages("nlme"); library(nlme)
## Loading required package: nlme

# To estimate MLMs using lmer
# Re-install to fix problem of matrix incompatibility
#oo <- options(repos = "https://cran.r-project.org/")
#utils::install.packages("Matrix")
#utils::install.packages("lme4")
#options(oo)
library(Matrix); library(lme4)
## Warning: package 'Matrix' was built under R version 4.3.3
## Warning: package 'lme4' was built under R version 4.3.3
##
## Attaching package: 'lme4'
## The following object is masked from 'package:nlme':
##
##     lmList

# To get Satterthwaite DDF in lmer
if (!require("lmerTest")) install.packages("lmerTest"); library(lmerTest)
## Loading required package: lmerTest
##
## Attaching package: 'lmerTest'
## The following object is masked from 'package:lme4':
##
##     lmer
## The following object is masked from 'package:stats':
##
##     step

# To get ICC in lmer
```

```r
if (!require("performance")) install.packages("performance"); library(performance)
## Loading required package: performance
## Warning: package 'performance' was built under R version 4.3.3

# Clear environment (re-run as needed for troubleshooting purposes)
#rm(list=ls())
```

Data Import, Manipulation, and Description

Next, we import the excel data file for this example and examine descriptive statistics. The data is in
"long" (or "stacked") format in which each row contains one trial (per subject per item).

```r
# Define variables for working directory and data name -- CHANGE THESE
filesave = "C:\\Dropbox/Papers and Data/SMiP/SMiP_2024_MLM_Part1/"
filename = "Example2.xlsx"; sheetname = "data"
setwd(dir=filesave)

# Import Example 2 excel trial-level dataset
Example2 = read_excel(paste0(filesave,filename), sheet=sheetname)
# Convert to data frame to use in analysis
Example2 = as.data.frame(Example2)

# Create predictors for analysis
# Log-transformed RT
Example2$logRT = log(Example2$RT)
# Piecewise slope for years over 65 if in older group
Example2$yrs65=NA
Example2$yrs65[which(Example2$older==0)]=0
Example2$yrs65[which(Example2$older==1)]=Example2$age[which(Example2$older==1)]-65
# Center item predictors near middle of 0-5 scale
Example2$rel3 = Example2$relevance-3
Example2$sal3 = Example2$salience-3

# Filter to only cases complete on all variables to be used below
Example2 = Example2[complete.cases(Example2[ ,
c("logRT","rel3","sal3","older","yrs65")]),]

print("Descriptives for logRT and predictors across all trials")

## [1] "Descriptives for logRT and predictors across all trials"

print(describe(x=Example2[,c("logRT","relevance","salience","age")], fast=TRUE),
digits=3)

##            vars    n   mean     sd    min    max  range    se
## logRT         1 7646  1.612  0.828  0.145  4.094  3.949 0.009
## relevance     2 7646  2.657  1.884  0.000  5.000  5.000 0.022
## salience      3 7646  3.026  1.091  1.000  5.000  4.000 0.012
## age           4 7646 40.100 27.165 18.000 86.000 68.000 0.311

# Add mean subject RT and number of responses per subject to dataset
Example2$SubjectMeanRT = ave(x=Example2$logRT, Example2$SubjectID, FUN=mean)
Example2$SubjectN = ave(x=Example2$logRT, Example2$SubjectID, FUN=length)
# Create subject-level dataset
```

```
Example2Subjects = unique(Example2[,c("SubjectID","SubjectMeanRT","SubjectN")])
print("Descriptives for SubjectMeanRT and SubjectN across subjects")

## [1] "Descriptives for SubjectMeanRT and SubjectN across subjects"

print(describe(x=Example2Subjects[,c("SubjectMeanRT","SubjectN")], fast=TRUE), digits=3)

##               vars   n   mean    sd    min    max range    se
## SubjectMeanRT    1 153  1.620 0.431  0.895  2.642 1.747 0.035
## SubjectN         2 153 49.974 1.573 43.000 51.000 8.000 0.127

# Add mean item RT and number of responses per item to dataset
Example2$ItemMeanRT = ave(x=Example2$logRT, Example2$ItemID, FUN=mean)
Example2$ItemN = ave(x=Example2$logRT, Example2$ItemID, FUN=length)
# Create item-level dataset
Example2Items = unique(Example2[,c("ItemID","ItemMeanRT","ItemN")])
print("Descriptives for ItemMeanRT and ItemN across items")

## [1] "Descriptives for ItemMeanRT and ItemN across items"

print(describe(x=Example2Items[,c("ItemMeanRT","ItemN")], fast=TRUE), digits=3)

##            vars  n    mean    sd     min     max  range    se
## ItemMeanRT    1 51   1.615 0.355   0.887   2.265  1.378 0.050
## ItemN         2 51 149.922 2.424 143.000 153.000 10.000 0.339
```

As shown in the descriptive tables above, there is a considerable range in mean RT across subjects as well as across items. Thus, our models will likely need to represent RT variability across both crossed level-2 dimensions (an empirical question to be answered below).

Empty Model 3: Partitioning logRT Variance into Three Components (or "Piles")

First, we will estimate an "empty means" (i.e., no-predictor) model for RT, in which the fixed effects in each model include only an intercept, denoted in each as γ_{000}. We are skipping empty models 1 and 2 (as were shown in Example 1 for pedagogical purposes). In the notation below, the subscripts are t = level-1 trial, i = level-2 item, and s = level-2 subject. Three variance components are included: a level-1 e_{tis} residual with estimated variance denoted as σ_e^2, a level-2 subject random intercept U_{00s} with estimated variance denoted as $\tau_{U_{00s}}^2$, and a level-2 item random intercept U_{0i0} with estimated variance denoted as $\tau_{U_{0I0}}^2$:

$$\text{Empty Model 3: } logRT_{tis} = \gamma_{000} + U_{00s} + U_{0i0} + e_{tis}$$
$$e_{tis} \sim N(0, \sigma_e^2)$$
$$U_{00s} \sim N\left(0, \tau_{U_{00s}}^2\right)$$
$$U_{0i0} \sim N\left(0, \tau_{U_{0I0}}^2\right)$$

```
# Skipping empty models 1 and 2 (were only for pedagogical purposes in Example 1)

print("Empty Model 3: Two-level crossed for trials nested in subjects and in items")

## [1] "Empty Model 3: Two-level crossed for trials nested in subjects and in items"

Empty3 = lmer(data=Example2, REML=TRUE, formula=logRT~1+(1|SubjectID)+(1|ItemID))
print("Show results using Satttherwaite DDF")
```

```
## [1] "Show results using Satttherwaite DDF"

summary(Empty3, ddf="Satterthwaite")

## Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
## Formula: logRT ~ 1 + (1 | SubjectID) + (1 | ItemID)
##    Data: Example2
##
## REML criterion at convergence: 15182
##
## Scaled residuals:
##    Min      1Q Median      3Q     Max
## -3.144 -0.649 -0.115   0.553   4.190
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  SubjectID (Intercept) 0.180    0.425
##  ItemID    (Intercept) 0.126    0.355
##  Residual              0.390    0.624
## Number of obs: 7646, groups:  SubjectID, 153; ItemID, 51
##
## Fixed effects:
##             Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)   1.6226     0.0608 99.8179    26.7   <2e-16
```

As shown above, the two-level crossed model returns a fixed intercept $\gamma_{000} = 1.623$ that is very close to the original RT mean = 1.612, but it now represents the sample mean of the subject means AND the item means (and is thus a weighted mean).

```
print("Show intraclass correlations for proportion of variance due to each sampling
dimension")

## [1] "Show intraclass correlations for proportion of variance due to each sampling
dimension"

icc(Empty3, by_group=TRUE)

## # ICC by Group
##
## Group     |   ICC
## -----------------
## SubjectID | 0.259
## ItemID    | 0.181
```

Here is how the ICCs above were computed:

$$ICC_S = \frac{\tau^2_{U_{00S}}}{\tau^2_{U_{00S}} + \tau^2_{U_{0I0}} + \sigma^2_e} = \frac{0.180}{0.180 + 0.126 + .390} = .259$$

$$ICC_I = \frac{\tau^2_{U_{0I0}}}{\tau^2_{U_{00S}} + \tau^2_{U_{0I0}} + \sigma^2_e} = \frac{0.126}{0.180 + 0.126 + .390} = .181$$

As shown above, the two-level crossed model partitions the observed variance in logRT into between-subject mean differences (25.9% as given by $ICC_S = 0.259$, between-item mean differences (18.1% as

given by $ICC_I = 0.181$, and trial-specific deviations from their subject and item means (the remaining 56.0%).

Here is how these ICCs can be computed manually using saved output:

```
print("Print stored variance components table for reference")

## [1] "Print stored variance components table for reference"

as.data.frame(VarCorr(Empty3))

##         grp        var1 var2    vcov    sdcor
## 1 SubjectID (Intercept) <NA> 0.18026 0.42457
## 2    ItemID (Intercept) <NA> 0.12592 0.35485
## 3  Residual        <NA> <NA> 0.38987 0.62440

# Compute model-implied total variance
Empty3TotVar = as.data.frame(VarCorr(Empty3))[1,4] +
               as.data.frame(VarCorr(Empty3))[2,4] +
               as.data.frame(VarCorr(Empty3))[3,4]
print("Total model-implied variance = "); Empty3TotVar

## [1] "Total model-implied variance = "

## [1] 0.69605

# Save each variance as an object
Empty3SubIntVar  = as.data.frame(VarCorr(Empty3))[1,4]
Empty3ItemIntVar = as.data.frame(VarCorr(Empty3))[2,4]
Empty3ResVar     = as.data.frame(VarCorr(Empty3))[3,4]
# Manual computation of subject and item ICCs
print("Subject ICC = "); Empty3ICC_S = Empty3SubIntVar /Empty3TotVar; Empty3ICC_S

## [1] "Subject ICC = "

## [1] 0.25897

print("Item ICC = ");    Empty3ICC_I = Empty3ItemIntVar/Empty3TotVar; Empty3ICC_I

## [1] "Item ICC = "

## [1] 0.18091
```

The `ranova` command then conducts a likelihood ratio test comparing the log-likelihood from the empty models with vs without each level-2 random intercept variance:

```
print("Show intraclass correlation LRTs"); ranova(Empty3)

## [1] "Show intraclass correlation LRTs"

## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## logRT ~ (1 | SubjectID) + (1 | ItemID)
##                npar logLik   AIC  LRT Df Pr(>Chisq)
## <none>            4  -7591 15190
```

```
## (1 | SubjectID)    3  -8771 17548 2360  1     <2e-16
## (1 | ItemID)       3  -8540 17087 1899  1     <2e-16
```

The LRT from `ranova` indicates that both random intercept variances are significantly > 0, subjects: $-2\Delta LL(1) = 2359.7, p < .001$, and items: $-2\Delta LL(1) = 1898.7, p < .001$. The conclusion would be the same using a mixture of χ^2 distributions with $df = 0,1$ instead of $df = 1$. For more elaboration, please see SMiP 2024 MLM Example 1.

To help convey effect size of these piles of variance in a more meaningful metric, we can use the results of our model to compute 95% random effects confidence intervals for the subject and item random intercepts. These indicate the expected range of the subject intercepts and item intercepts for 95% of our sample as follows:

$$\text{Subject Random Intercept 95\% CI} = \gamma_{000} \pm 1.96 \times \sqrt{\tau^2_{U_{00S}}}$$

$$\text{Subject Random Intercept 95\% CI} = 1.622 \pm 1.96 \times \sqrt{0.180} = 0.790 \text{ to } 2.455$$

$$\text{Item Random Intercept 95\% CI} = \gamma_{000} \pm 1.96 \times \sqrt{\tau^2_{U_{0I0}}}$$

$$\text{Item Random Intercept 95\% CI} = 1.622 \pm 1.96 \times \sqrt{0.126} = 0.927 \text{ to } 2.318$$

Here is how the random effect confidence intervals can be computed using saved output:

```
# Save fixed intercept for use below
Empty3FixInt = fixef(Empty3)
print("95% random intercept confidence interval for subjects")

## [1] "95% random intercept confidence interval for subjects"

SubInt_LCI = Empty3FixInt - 1.96*sqrt(Empty3SubIntVar)
SubInt_UCI = Empty3FixInt + 1.96*sqrt(Empty3SubIntVar)
print("Subject intercept lower CI = "); SubInt_LCI

## [1] "Subject intercept lower CI = "

## (Intercept)
##     0.79045

print("Subject intercept upper CI = "); SubInt_UCI

## [1] "Subject intercept upper CI = "

## (Intercept)
##     2.4548

print("95% random intercept confidence interval for items")

## [1] "95% random intercept confidence interval for items"

ItemInt_LCI = Empty3FixInt - 1.96*sqrt(Empty3ItemIntVar)
ItemInt_UCI = Empty3FixInt + 1.96*sqrt(Empty3ItemIntVar)
print("Item intercept lower CI = "); ItemInt_LCI

## [1] "Item intercept lower CI = "
```

```
## (Intercept)
##     0.9271
```

```
print("Item intercept upper CI = "); ItemInt_UCI
```

```
## [1] "Item intercept upper CI = "
```

```
## (Intercept)
##     2.3181
```

Conditional Models Including Item Predictors of logRT

Conditional Model 1: Add Fixed Slopes of Item Predictors

Next, we add fixed slopes for the item predictors (rel3 = change relevance -3 and sal3 = change salience -3) to the two-level crossed model to predict logRT (in which the index in the second subscript keeps track of which item predictor each is). The fixed effects in each model include the intercept γ_{000}, the fixed slopes γ_{010} and γ_{020} for the main effects of rel3 and sal3, respectively, and a fixed slope for their interaction term, γ_{030}:

Conditional Model 1: $logRT_{tis} = \gamma_{000} + \gamma_{010}(\text{rel}_i - 3) + \gamma_{020}(\text{sal}_i - 3) + \gamma_{030}(\text{rel}_i - 3)(\text{sal}_i - 3)$
$+ U_{00s} + U_{0i0} + e_{tis}$
$e_{tis} \sim N(0, \sigma_e^2)$
$U_{00s} \sim N(0, \tau_{U_{00S}}^2)$
$U_{0i0} \sim N(0, \tau_{U_{0I0}}^2)$

```
print("Conditional Model 1: Add fixed slopes for item predictors")
```

```
## [1] "Conditional Model 1: Add fixed slopes for item predictors"
```

```
Cond1 = lmer(data=Example2, REML=TRUE, formula=
             logRT~1+rel3+sal3+rel3:sal3+(1|SubjectID)+(1|ItemID))
print("Show results using Sattherwaite DDF")
```

```
## [1] "Show results using Sattherwaite DDF"
```

```
summary(Cond1, ddf="Satterthwaite")
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
## Formula: logRT ~ 1 + rel3 + sal3 + rel3:sal3 + (1 | SubjectID) + (1 |      ItemID)
##    Data: Example2
##
## REML criterion at convergence: 15181
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -3.151 -0.649 -0.115  0.553  4.178
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  SubjectID (Intercept) 0.1803   0.425
##  ItemID    (Intercept) 0.0941   0.307
##  Residual              0.3899   0.624
## Number of obs: 7646, groups:  SubjectID, 153; ItemID, 51
```

```
##
## Fixed effects:
##               Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)     1.6126     0.0569 104.4498   28.32   <2e-16
## rel3           -0.0503     0.0237  47.0087   -2.12    0.040
## sal3           -0.1377     0.0421  47.0047   -3.27    0.002
## rel3:sal3      -0.0115     0.0200  47.0202   -0.57    0.569


## Correlation of Fixed Effects:
##           (Intr) rel3    sal3
## rel3       0.166
## sal3      -0.087 -0.232
## rel3:sal3 -0.171 -0.100  0.252
```

```
# Btw, here is shorter code to include interactions with all lower-order main effects:
#Cond1 = lmer(data=Example1, REML=TRUE,
#            LogRT~1+rel3*sal3+(1|SubjectID)+(1|ItemID))
```

Let's interpret the results for the fixed effects in the model above:

fixed intercept γ_{000} =

fixed slope for rel3 γ_{010} =

fixed slope for sal3 γ_{020} =

fixed slope for interaction γ_{030} =

To get an effect size for the contribution of our item predictors, we can compute a pseudo-R^2 value for the proportion reduction in item random intercept variance relative to the empty means model:

$$\text{Pseudo-R}_I^2 = \frac{\text{Empty3 Variance}_I - \text{Cond1 Variance}_I}{\text{Empty3 Variance}_I} = \frac{0.126 - 0.094}{0.126} = .256$$

Here is how to compute pseudo-R^2 values using saved output:

```
print("Print stored variance components table for reference")
```

```
## [1] "Print stored variance components table for reference"
```

```
as.data.frame(VarCorr(Cond1))
```

```
##         grp         var1 var2      vcov    sdcor
## 1 SubjectID (Intercept) <NA> 0.180280 0.42459
## 2    ItemID (Intercept) <NA> 0.094113 0.30678
## 3  Residual             <NA> <NA> 0.389870 0.62440
```

```
# Save each variance as an object
Cond1SubIntVar  = as.data.frame(VarCorr(Cond1))[1,4]
Cond1ItemIntVar = as.data.frame(VarCorr(Cond1))[2,4]
Cond1ResVar     = as.data.frame(VarCorr(Cond1))[3,4]
# Compute pseudo-R2 for each variance
Cond1SubIntR2  = (Empty3SubIntVar  - Cond1SubIntVar) /Empty3SubIntVar;  Cond1SubIntR2
```

```
## [1] -0.00011821
```

```
Cond1ItemIntR2 = (Empty3ItemIntVar - Cond1ItemIntVar)/Empty3ItemIntVar; Cond1ItemIntR2

## [1] 0.25259

Cond1ResR2     = (Empty3ResVar     - Cond1ResVar)    /Empty3ResVar;    Cond1ResR2

## [1] 0.0000019061
```

We can get a significance test for the item random intercept pseudo-R^2 using the function `contestMD` from the `lmerTest` package, which allows us to obtain a joint hypothesis test for multiple slopes at once:

```
print("Omnibus F-test for model prediction")

## [1] "Omnibus F-test for model prediction"

contestMD(Cond1, ddf="Satterthwaite", L=rbind(c(0,1,0,0),c(0,0,1,0),c(0,0,0,1)))

##   Sum Sq Mean Sq NumDF DenDF F value      Pr(>F)
## 1  7.579  2.5263     3 47.01  6.4799 0.00092287
```

The two item features and their interaction significantly predicted logRT, $F(3,\ 47.0) = 6.48, p = .001$, and accounted for 25.3% of the item random intercept variance. The other variances remained unchanged relative to the empty means model, as expected given that no subject-level or trial-level predictors were added.

We can also see whether the item random intercept variance that remains is significantly > 0 using an LRT against a model without it via `ranova`:

```
print("LRT for remaining item random intercept variance"); ranova(Cond1)

## [1] "LRT for remaining item random intercept variance"

## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## logRT ~ rel3 + sal3 + (1 | SubjectID) + (1 | ItemID) + rel3:sal3
##                 npar logLik   AIC  LRT Df Pr(>Chisq)
## <none>             7  -7591 15195
## (1 | SubjectID)    6  -8771 17553 2360  1      <2e-16
## (1 | ItemID)       6  -8270 16551 1358  1      <2e-16
```

Significant item random intercept variance remained, $-2\Delta LL(1) = 1357.9, p < .001$, so we will retain the level-2 item random intercept variance in our model.

Conditional Model 2: Add Fixed Slopes of Subject Predictors

Next, we add fixed slopes for the subject predictors (older, 0=younger 1=older) and (yrs65 = years over age 65 if older) to the two-level crossed model to predict logRT (in which the index in the first subscript keeps track of which subject predictor each is). The new fixed slopes are γ_{001} and γ_{002} for the main effects of older and yrs65, respectively:

Conditional Model 1: $logRT_{tis} = \gamma_{000} + \gamma_{010}(\text{rel}_i - 3) + \gamma_{020}(\text{sal}_i - 3) + \gamma_{030}(\text{rel}_i - 3)(\text{sal}_i - 3)$
$+ \gamma_{001}(\text{older}_s) + \gamma_{002}(\text{yrs}_s - 65) + U_{00s} + U_{0i0} + e_{tis}$

$e_{tis} \sim N(0, \sigma_e^2)$

$U_{00s} \sim N(0, \tau_{U_{00S}}^2)$

$U_{0i0} \sim N(0, \tau_{U_{0I0}}^2)$

```
print("Conditional Model 2: Add fixed slopes for subject predictors")

## [1] "Conditional Model 2: Add fixed slopes for subject predictors"

Cond2 = lmer(data=Example2, REML=TRUE, formula=
             logRT~1+rel3+sal3+older+yrs65+ rel3:sal3+
               (1|SubjectID)+(1|ItemID)) # Moved interaction to end to match output
print("Show results using Sattherwaite DDF")

## [1] "Show results using Sattherwaite DDF"

summary(Cond2, ddf="Satterthwaite")

## Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
## Formula: logRT ~ 1 + rel3 + sal3 + older + yrs65 + rel3:sal3 + (1 | SubjectID) +
(1 | ItemID)
##    Data: Example2
##
## REML criterion at convergence: 14920
##
## Scaled residuals:
##    Min    1Q Median    3Q    Max
## -3.224 -0.651 -0.115  0.553  4.207
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  SubjectID (Intercept) 0.0233   0.153
##  ItemID    (Intercept) 0.0941   0.307
##  Residual              0.3899   0.624
## Number of obs: 7646, groups:  SubjectID, 153; ItemID, 51
##
## Fixed effects:
##             Estimate Std. Error        df t value  Pr(>|t|)
## (Intercept)  1.31231    0.04833  59.74052   27.15   < 2e-16
## rel3        -0.05031    0.02374  47.00822   -2.12     0.039
## sal3        -0.13774    0.04214  47.00414   -3.27     0.002
## older        0.58991    0.05556 150.48302   10.62   < 2e-16
## yrs65        0.02021    0.00441 151.60904    4.59 0.0000093
## rel3:sal3   -0.01149    0.01997  47.01962   -0.58     0.568
##
## Correlation of Fixed Effects:
##           (Intr) rel3   sal3   older  yrs65
## rel3       0.195
## sal3      -0.103 -0.232
## older     -0.120  0.000  0.000
## yrs65      0.000  0.000  0.000 -0.847
## rel3:sal3 -0.202 -0.100  0.252  0.000  0.000
```

Let's interpret the results for the modified or fixed effects in the model above:

fixed intercept γ_{000} =

fixed slope for older γ_{001} =

fixed slope for yrs65 γ_{002} =

To get an effect size for the contribution of our subject predictors, we can compute a pseudo-R^2 for the proportion reduction in subject random intercept variance relative to the empty means model:

$$\text{Pseudo-R}_S^2 = \frac{\text{Empty3 Variance}_S - \text{Cond2 Variance}_S}{\text{Empty3 Variance}_S} = \frac{0.180 - 0.023}{0.180} = .871$$

Here is how to compute pseudo-R^2 values and their change across conditional models using saved output:

```
print("Print stored variance components table for reference")

## [1] "Print stored variance components table for reference"

as.data.frame(VarCorr(Cond2))

##          grp          var1 var2    vcov    sdcor
## 1 SubjectID (Intercept) <NA> 0.023328 0.15274
## 2    ItemID (Intercept) <NA> 0.094077 0.30672
## 3  Residual            <NA> <NA> 0.389870 0.62440

# Save each variance as an object
Cond2SubIntVar  = as.data.frame(VarCorr(Cond2))[1,4]
Cond2ItemIntVar = as.data.frame(VarCorr(Cond2))[2,4]
Cond2ResVar     = as.data.frame(VarCorr(Cond2))[3,4]
# Compute pseudo-R2 for each variance and change relative to Cond1
Cond2SubIntR2        = (Empty3SubIntVar  - Cond2SubIntVar) /Empty3SubIntVar;
Cond2SubIntR2

## [1] 0.87059

Cond2ItemIntR2       = (Empty3ItemIntVar - Cond2ItemIntVar)/Empty3ItemIntVar;
Cond2ItemIntR2

## [1] 0.25288

Cond2ResR2           = (Empty3ResVar     - Cond2ResVar)    /Empty3ResVar;    Cond2ResR2

## [1] 0.0000016704

Cond2SubIntR2Change  =  Cond2SubIntR2  - Cond1SubIntR2;  Cond2SubIntR2Change

## [1] 0.8707

Cond2ItemIntR2Change =  Cond2ItemIntR2 - Cond1ItemIntR2; Cond2ItemIntR2Change

## [1] 0.00028896

Cond2ResR2Change     =  Cond2ResR2     - Cond1ResR2;     Cond2ResR2Change

## [1] -0.00000023567
```

We can get a significance test for that subject random intercept pseudo-R^2 value using the function contestMD from the lmerTest package, which allows us to obtain a joint hypothesis test for multiple slopes at once:

```
print("Omnibus F-test for subject fixed slopes")

## [1] "Omnibus F-test for subject fixed slopes"

contestMD(Cond2, ddf="Satterthwaite", L=rbind(c(0,0,0,1,0,0),c(0,0,0,0,1,0)))

##    Sum Sq Mean Sq NumDF  DenDF F value      Pr(>F)
## 1 298.32  149.16     2 151.05  382.59 7.4709e-60
```

The two subject characteristics predicted logRT, $F(2,\ 151.0) = 382.59, p < .001$, and accounted for 87.1% of the subject random intercept variance. The other variances remained unchanged relative to the first conditional model, as expected given that no additional item-level or trial-level predictors were added.

We can also see whether the subject random intercept variance that remains is significantly > 0 using an LRT against a model without it via ranova:

```
print("LRT for remaining subject random intercept variance"); ranova(Cond2)

## [1] "LRT for remaining subject random intercept variance"

## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## logRT ~ rel3 + sal3 + older + yrs65 + (1 | SubjectID) + (1 | ItemID) + rel3:sal3
##                  npar logLik   AIC   LRT Df Pr(>Chisq)
## <none>              9  -7460 14938
## (1 | SubjectID)     8  -7574 15163   227  1     <2e-16
## (1 | ItemID)        8  -8139 16293  1358  1     <2e-16
```

Significant subject random intercept variance remained, $-2\Delta LL(1) = 227.3, p < .001$, so we will retain the level-2 subject random intercept variance in our model.

Conditional Model 3: Add Random Slope over Subjects for the Salience Item Predictor

Next, we add to the two-level crossed model a subject random slope for the item predictor sal3, U_{02s}, whose variance across subjects is then estimated to form the level-2 subject random sal3 slope variance denoted as $\tau^2_{U_{02S}}$:

Conditional Model 3: $logRT_{tis} = \gamma_{000} + \gamma_{010}(\text{rel}_i - 3) + \gamma_{020}(\text{sal}_i - 3) + \gamma_{030}(\text{rel}_i - 3)(\text{sal}_i - 3)$
$+\gamma_{001}(\text{older}_s) + \gamma_{002}(\text{yrs}_s - 65) + U_{00s} + U_{02s}(\text{sal}_i - 3) + U_{0i0} + e_{tis}$
$e_{tis} \sim N(0, \sigma_e^2)$
$\begin{bmatrix} U_{00s} \\ U_{02s} \end{bmatrix} \sim MVN \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \tau^2_{U_{00S}} & \tau_{U_{00S,012}} \\ \tau_{U_{00S,02S}} & \tau^2_{U_{02S}} \end{bmatrix} \right)$
$U_{0i0} \sim N(0, \tau^2_{U_{0I0}})$

This new random slope allow individual differences in the extent of the RT difference by increasing change salience. Also added is the covariance between the subject random intercepts and the subject random sal3 slopes (which is provided as a correlation in the output below):

```
print("Conditional Model 3: Add random subject slope for item predictor sal3")

## [1] "Conditional Model 3: Add random subject slope for item predictor sal3"

Cond3 = lmer(data=Example2, REML=TRUE, formula=
            logRT~1+rel3+sal3+older+yrs65+ rel3:sal3+
                (1+sal3|SubjectID)+(1|ItemID))
print("Show results using Sattherwaite DDF")

## [1] "Show results using Sattherwaite DDF"

summary(Cond3, ddf="Satterthwaite")

## Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
## Formula: logRT ~ 1 + rel3 + sal3 + older + yrs65 + rel3:sal3 + (1 + sal3 |
## SubjectID) + (1 | ItemID)
##     Data: Example2
##
## REML criterion at convergence: 14912
##
## Scaled residuals:
##     Min      1Q Median      3Q     Max
## -3.252 -0.652 -0.112   0.549   4.222
##
## Random effects:
##  Groups    Name        Variance Std.Dev. Corr
##  SubjectID (Intercept) 0.02341  0.1530
##            sal3        0.00241  0.0491   -0.06
##  ItemID    (Intercept) 0.09417  0.3069
##  Residual              0.38694  0.6220
## Number of obs: 7646, groups:  SubjectID, 153; ItemID, 51
##
## Fixed effects:
##               Estimate Std. Error       df t value  Pr(>|t|)
## (Intercept)    1.3116     0.0483  59.7540   27.13   < 2e-16
## rel3          -0.0503     0.0238  47.0086   -2.12    0.0393
## sal3          -0.1377     0.0423  47.8246   -3.25    0.0021
## older          0.5897     0.0555 150.4455   10.62   < 2e-16
## yrs65          0.0204     0.0044 151.5489    4.64 0.0000076
## rel3:sal3     -0.0115     0.0200  47.0203   -0.58    0.5673
##
## Correlation of Fixed Effects:
##           (Intr) rel3   sal3   older  yrs65
## rel3       0.195
## sal3      -0.104 -0.231
## older     -0.120  0.000  0.000
## yrs65      0.000  0.000  0.000 -0.847
## rel3:sal3 -0.202 -0.100  0.251  0.000  0.000

print("LRT for subject item salience slope variance"); ranova(Cond3)

## [1] "LRT for subject item salience slope variance"

## ANOVA-like table for random-effects: Single term deletions
##
```

```
## Model:
## logRT ~ rel3 + sal3 + older + yrs65 + (1 + sal3 | SubjectID) + (1 | ItemID) +
rel3:sal3
##                              npar logLik   AIC  LRT Df Pr(>Chisq)
## <none>                         11  -7456 14934
## sal3 in (1 + sal3 | SubjectID)  9  -7460 14938    8  2       0.015
## (1 | ItemID)                    10 -8138 16297 1365  1      <2e-16
```

As shown above, the fixed sal3 slope is now $\gamma_{020} = -0.138$, which is the same to several decimal places as it was previously: $\gamma_{020} = -0.138$. However, it has changed interpretation: It is now the mean of the subject-specific sal3 slopes. The new subject random sal3 slope variance $\tau^2_{U_{02S}} = 0.002$. The subject random slopes were negatively correlated $r = -0.064$ with the subject random intercepts, indicating that subjects who had slower response times to medium-salience changes (i.e., higher intercepts at sal3=0) tended to have slightly less of a difference between low- and high-salience changes (i.e., flatter sal3 slopes).

The LRT generated by `ranova` indicates that the model fit improved significantly from adding the new random slope variance and covariance, $-2\Delta LL(2) = 8.367, p = .015$. Thus, we should retain the two new parameters.

To use the more correct (and less conservative) mixture χ^2 distribution instead, we would use $df = 1$ for when the random slope variance would have become negative and $df = 2$ for when it would have been positive; the covariance does not have a boundary at 0 so we keep its $df = 1$ regardless. The mixture critical value at $p < .05$ for $df = 1,2$ is 5.14, and we can compute an exact p-value for the mixture by weighting each p-value by 0.5 and then summing them, as shown below:

```
print("LRT for random slope variance using a mixture-chi-square test")

## [1] "LRT for random slope variance using a mixture-chi-square test"

Cond3Diff2LL = -2*(logLik(Cond2)-logLik(Cond3)); Cond3Diff2LL

## 'log Lik.' 8.3671 (df=9)

Cond3DiffP1 = pchisq(Cond3Diff2LL, df=1, lower.tail=FALSE); Cond3DiffP1

## 'log Lik.' 0.0038207 (df=9)

Cond3DiffP2 = pchisq(Cond3Diff2LL, df=2, lower.tail=FALSE); Cond3DiffP2

## 'log Lik.' 0.015244 (df=9)

Cond3DiffP12 = (.5*Cond3DiffP1) + (.5*Cond3DiffP2)
print("Test statistic and mixture p-values for df=1,2")

## [1] "Test statistic and mixture p-values for df=1,2"

Cond3Diff2LL; Cond3DiffP12

## 'log Lik.' 8.3671 (df=9)

## 'log Lik.' 0.0095324 (df=9)
```

To get a sense of the degree of slope differences (to help convey effect size), we can compute a 95% random slope confidence interval as follows:

$$\text{Subject Random Sal3 Slope 95\% CI} = \gamma_{020} \pm 1.96 \times \sqrt{\tau^2_{U_{02S}}}$$

$$\text{Subject Random Sal3 Slope 95\% CI} = -0.138 \pm 1.96 \times \sqrt{0.0024} = -0.234 \text{ to } -0.041$$

This indicates the expected variability in the slope for change salience, in which 95% of the sample subjects would be expected to have a negative item predictor slope.

Below is how the random effect confidence intervals can be computed using saved output:

```r
# Print stored fixed effects table for reference
as.data.frame(fixef(Cond3))

##              fixef(Cond3)
## (Intercept)      1.311608
## rel3            -0.050355
## sal3            -0.137657
## older            0.589685
## yrs65            0.020416
## rel3:sal3       -0.011507

# Save fixed sal3 slope for use below
Cond3SalSlp = as.data.frame(fixef(Cond3))[3,1]

# Print stored variance components table for reference
as.data.frame(VarCorr(Cond3))

##          grp        var1 var2        vcov      sdcor
## 1 SubjectID (Intercept) <NA>   0.02340754   0.152995
## 2 SubjectID        sal3 <NA>   0.00241121   0.049104
## 3 SubjectID (Intercept) sal3 -0.00048115  -0.064045
## 4    ItemID (Intercept) <NA>   0.09416920   0.306870
## 5  Residual        <NA> <NA>   0.38693877   0.622044

# Save each variance as an object
Cond3SubIntVar  = as.data.frame(VarCorr(Cond3))[1,4]
Cond3SubSalVar  = as.data.frame(VarCorr(Cond3))[2,4]
Cond3ItemIntVar = as.data.frame(VarCorr(Cond3))[4,4]
Cond3ResVar     = as.data.frame(VarCorr(Cond3))[5,4]

print("95% random salience slope confidence interval for subjects")

## [1] "95% random salience slope confidence interval for subjects"

SubSal_LCI = Cond3SalSlp - 1.96*sqrt(Cond3SubSalVar)
SubSal_UCI = Cond3SalSlp + 1.96*sqrt(Cond3SubSalVar)
print("Subject sal3 slope lower CI = "); SubSal_LCI

## [1] "Subject sal3 slope lower CI = "

## [1] -0.2339

print("Subject sal3 slope upper CI = "); SubSal_UCI

## [1] "Subject sal3 slope upper CI = "

## [1] -0.041413
```

Conditional Model 4: Add Random Slope over Subjects for the Relevance Item Predictor

Next, we add to the two-level crossed model a subject random slope for the item predictor rel3, U_{01s}, whose variance across subjects is then estimated to form the level-2 subject random rel3 slope variance denoted as $\tau^2_{U_{01S}}$:

Conditional Model 4: $logRT_{tis} = \gamma_{000} + \gamma_{010}(rel_i - 3) + \gamma_{020}(sal_i - 3) + \gamma_{030}(rel_i - 3)(sal_i - 3)$
$+\gamma_{001}(older_s) + \gamma_{002}(yrs_s - 65) + U_{00s} + U_{010s}(rel_i - 3) + U_{020s}(sal_i - 3) + U_{0i0} + e_{tis}$
$e_{tis} \sim N(0, \sigma^2_e)$

$$\begin{bmatrix} U_{00s} \\ U_{01s} \\ U_{02s} \end{bmatrix} \sim MVN \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \tau^2_{U_{00S}} & \tau_{U_{00S,01S}} & \tau_{U_{00S,02S}} \\ \tau_{U_{00S,01S}} & \tau^2_{U_{01S}} & \tau_{U_{01S,02S}} \\ \tau_{U_{00S,02S}} & \tau_{U_{01S,02S}} & \tau^2_{U_{02S}} \end{bmatrix} \right)$$

$U_{0i0} \sim N\left(0, \tau^2_{U_{0I0}}\right)$

This new random slope allow individual differences in the extent of the RT difference by increasing change relevance. Also added are the covariances of the subject random rel3 slopes with the subject random intercepts and random sal3 slopes (which are provided as correlations in the output below):

```
print("Conditional Model 4: Add random subject slope for item predictor rel3")

## [1] "Conditional Model 4: Add random subject slope for item predictor rel3"

Cond4 = lmer(data=Example2, REML=TRUE, formula=
             logRT~1+rel3+sal3+older+yrs65+ rel3:sal3+
               (1+sal3+rel3|SubjectID)+(1|ItemID))

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model
## failed to converge with max|grad|
## = 0.00644277 (tol = 0.002, component 1)

print("Show results using Sattherwaite DDF")

## [1] "Show results using Sattherwaite DDF"

summary(Cond4, ddf="Satterthwaite")

## Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
## Formula: logRT ~ 1 + rel3 + sal3 + older + yrs65 + rel3:sal3 + (1 + sal3 +     rel3 |
## SubjectID) + (1 | ItemID)
##    Data: Example2
##
## REML criterion at convergence: 14909
##
## Scaled residuals:
##    Min     1Q  Median     3Q     Max
## -3.191 -0.650 -0.110  0.549  4.171
##
## Random effects:
##  Groups     Name        Variance Std.Dev. Corr
##  SubjectID (Intercept) 0.023705 0.1540
##            sal3        0.002313 0.0481   -0.07
##            rel3        0.000426 0.0206    0.14  0.08
##  ItemID    (Intercept) 0.094198 0.3069
```

```
##  Residual                   0.385449 0.6208
## Number of obs: 7646, groups:  SubjectID, 153; ItemID, 51
##
## Fixed effects:
##              Estimate Std. Error       df t value  Pr(>|t|)
## (Intercept)   1.3132     0.0484  59.8342   27.15   < 2e-16
## rel3         -0.0503     0.0238  47.4546   -2.11    0.0400
## sal3         -0.1378     0.0423  47.7861   -3.25    0.0021
## older         0.5884     0.0555 150.4282   10.60   < 2e-16
## yrs65         0.0201     0.0044 151.6242    4.57 0.0000099
## rel3:sal3    -0.0115     0.0200  47.0171   -0.58    0.5669
##
## Correlation of Fixed Effects:
##           (Intr) rel3   sal3   older  yrs65
## rel3       0.197
## sal3      -0.104 -0.230
## older     -0.120  0.000  0.000
## yrs65      0.000 -0.001  0.000 -0.847
## rel3:sal3 -0.202 -0.100  0.251  0.000  0.000
## optimizer (nloptwrap) convergence code: 0 (OK)
## Model failed to converge with max|grad| = 0.00644277 (tol = 0.002, component 1)
```

```
print("LRT for subject item salience slope variance"); ranova(Cond4)
```

```
## [1] "LRT for subject item salience slope variance"

## boundary (singular) fit: see help('isSingular')

## ANOVA-like table for random-effects: Single term deletions
##
## Model:
## logRT ~ rel3 + sal3 + older + yrs65 + (1 + sal3 + rel3 | SubjectID) + (1 | ItemID) +
rel3:sal3
##                                        npar logLik   AIC   LRT Df Pr(>Chisq)
## <none>                                   14  -7454 14937
## sal3 in (1 + sal3 + rel3 | SubjectID)    11  -7458 14939     8  3      0.044
## rel3 in (1 + sal3 + rel3 | SubjectID)    11  -7456 14934     3  3      0.432
## (1 | ItemID)                             13  -8138 16302  1367  1     <2e-16
```

As shown above, the fixed rel3 slope is now $\gamma_{010} = -0.05$, which is the same to several decimal places as it was previously: $\gamma_{010} = -0.05$. However, it has changed interpretation: It is now the mean of the subject-specific rel3 slopes. The new subject random rel3 slope variance $\tau^2_{U_{01S}} = 0.002$. However, the model appears to not have converged properly (as indicated by the error messages). In addition, the LRT generated by ranova indicates that the model fit did not improve significantly from adding the new random slope variance and two covariances, $-2\Delta LL(2) = 2.751, p = .432$. Thus, all evidence suggests that we should not retain the three new parameters.

To use the more correct (and less conservative) mixture χ^2 distribution instead, we would use $df = 2$ for when the random slope variance would have become negative and $df = 1$ for when it would have been positive; the covariances do not have a boundary at 0 so we keep their $df = 2$ regardless. The mixture critical value at $p < .05$ for $df = 2,3$ is 7.05, and we can compute an exact p-value for the mixture by weighting each p-value by 0.5 and then summing them, as shown below:

```
print("LRT for random slope variance using a mixture-chi-square test")

## [1] "LRT for random slope variance using a mixture-chi-square test"

Cond4Diff2LL = -2*(logLik(Cond3)-logLik(Cond4)); Cond4Diff2LL

## 'log Lik.' 2.7511 (df=11)

Cond4DiffP2 = pchisq(Cond4Diff2LL, df=2, lower.tail=FALSE); Cond4DiffP2

## 'log Lik.' 0.2527 (df=11)

Cond4DiffP3 = pchisq(Cond4Diff2LL, df=3, lower.tail=FALSE); Cond4DiffP3

## 'log Lik.' 0.43161 (df=11)

Cond4DiffP23 = (.5*Cond4DiffP2) + (.5*Cond4DiffP3)
print("Test statistic and mixture p-values for df=2,3")

## [1] "Test statistic and mixture p-values for df=2,3"

Cond4Diff2LL; Cond4DiffP23

## 'log Lik.' 2.7511 (df=11)

## 'log Lik.' 0.34216 (df=11)
```

To get a sense of the degree of slope differences (to help convey effect size), we can still compute a 95% random slope confidence interval as follows:

$$\text{Subject Random Rel3 Slope 95\% CI} = \gamma_{010} \pm 1.96 \times \sqrt{\tau^2_{U_{01S}}}$$

$$\text{Subject Random Rel3 Slope 95\% CI} = -0.050 \pm 1.96 \times \sqrt{0.0004} = -0.091 \text{ to } -0.001$$

This indicates the expected variability in the slope for change relevance, in which 95% of the sample subjects would be expected to have a negative item predictor slope.

Below is how the random effect confidence intervals can be computed using saved output:

```
# Print stored fixed effects table for reference
as.data.frame(fixef(Cond4))

##                fixef(Cond4)
## (Intercept)      1.313220
## rel3            -0.050278
## sal3            -0.137787
## older            0.588418
## yrs65            0.020131
## rel3:sal3       -0.011519

# Save fixed rel3 slope for use below
Cond4RelSlp = as.data.frame(fixef(Cond4))[2,1]

# Print stored variance components table for reference
as.data.frame(VarCorr(Cond4))
```

```
##        grp         var1 var2        vcov       sdcor
## 1 SubjectID (Intercept) <NA>  0.023705052  0.153964
## 2 SubjectID         sal3 <NA>  0.002313408  0.048098
## 3 SubjectID         rel3 <NA>  0.000425815  0.020635
## 4 SubjectID (Intercept) sal3 -0.000527161 -0.071186
## 5 SubjectID (Intercept) rel3  0.000459417  0.144603
## 6 SubjectID         sal3 rel3  0.000074865  0.075430
## 7    ItemID (Intercept) <NA>  0.094197532  0.306916
## 8  Residual         <NA> <NA>  0.385448813  0.620845
```

```
# Save subject random rel3 slope variance for use below
Cond4SubRelVar = as.data.frame(VarCorr(Cond4))[3,4]

print("95% random relevance slope confidence interval for subjects")

## [1] "95% random relevance slope confidence interval for subjects"

SubRelSlp_LCI = Cond4RelSlp - 1.96*sqrt(Cond4SubRelVar)
SubRelSlp_UCI = Cond4RelSlp + 1.96*sqrt(Cond4SubRelVar)
print("Subject rel3 slope lower CI = "); SubRelSlp_LCI

## [1] "Subject rel3 slope lower CI = "

## [1] -0.090724

print("Subject rel3 slope upper CI = "); SubRelSlp_UCI

## [1] "Subject rel3 slope upper CI = "

## [1] -0.0098332
```

Conditional Model 5: Add All Possible Fixed Slope Interactions

Next, we add to model 3 (with only a random subject sal3 slope) all possible fixed slope interactions between the item predictors and the subject predictors (six new interactions in total):

Conditional Model 5: $logRT_{tis} = \gamma_{000} + \gamma_{010}(rel_i - 3) + \gamma_{020}(sal_i - 3) + \gamma_{030}(rel_i - 3)(sal_i - 3)$
$+\gamma_{001}(older_s) + \gamma_{002}(yrs_s - 65)$
$+\gamma_{011}(rel_i - 3)(older_s) + \gamma_{021}(sal_i - 3)(older_s) + \gamma_{031}(rel_i - 3)(sal_i - 3)(older_s)$
$+\gamma_{012}(rel_i - 3)(yrs_s - 65) + \gamma_{022}(sal_i - 3)(yrs_s - 65) + \gamma_{032}(rel_i - 3)(sal_i - 3)(yrs_s - 65)$
$+U_{00s} + U_{020s}(sal_i - 3) + U_{0i0} + e_{tis}$
$e_{tis} \sim N(0, \sigma_e^2)$
$\begin{bmatrix} U_{00s} \\ U_{02s} \end{bmatrix} \sim MVN\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \tau_{U_{00S}}^2 & \tau_{U_{00S,012}} \\ \tau_{U_{00S,02S}} & \tau_{U_{02S}}^2 \end{bmatrix} \right)$
$U_{0i0} \sim N\left(0, \tau_{U_{0I0}}^2\right)$

```
print("Conditional Model 5: Add all possible fixed slope interactions")

## [1] "Conditional Model 5: Add all possible fixed slope interactions"

Cond5 = lmer(data=Example2, REML=TRUE, formula=
          logRT~1+rel3+sal3+older+yrs65+ rel3:sal3+
             rel3:older+ sal3:older+ rel3:sal3:older+
```

```
                      rel3:yrs65+ sal3:yrs65+ rel3:sal3:yrs65+
                      (1+sal3|SubjectID)+(1|ItemID))
print("Show results using Sattherwaite DDF")

## [1] "Show results using Sattherwaite DDF"

summary(Cond5, ddf="Satterthwaite")

## Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
## Formula: logRT ~ 1 + rel3 + sal3 + older + yrs65 + rel3:sal3 + rel3:older +
##      sal3:older + rel3:sal3:older + rel3:yrs65 + sal3:yrs65 +
##      rel3:sal3:yrs65 + (1 + sal3 | SubjectID) + (1 | ItemID)
##    Data: Example2
##
## REML criterion at convergence: 14924
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -2.959 -0.653 -0.110  0.553  4.160
##
## Random effects:
##  Groups     Name        Variance Std.Dev. Corr
##  SubjectID (Intercept) 0.02341  0.1530
##            sal3        0.00205  0.0453   -0.06
##  ItemID    (Intercept) 0.09432  0.3071
##  Residual              0.38532  0.6207
## Number of obs: 7646, groups:  SubjectID, 153; ItemID, 51
##
## Fixed effects:
##                     Estimate    Std. Error           df t value  Pr(>|t|)
## (Intercept)        1.3038060    0.0484104   59.8617518   26.93   < 2e-16
## rel3              -0.0639845    0.0239464   48.4374144   -2.67    0.0102
## sal3              -0.1425470    0.0427487   49.5386348   -3.33    0.0016
## older              0.6067037    0.0561958  157.5463875   10.80   < 2e-16
## yrs65              0.0208350    0.0044567  158.8448366    4.68 0.0000062
## rel3:sal3         -0.0027290    0.0201368   48.4382233   -0.14    0.8928
## rel3:older         0.0142439    0.0152275 7300.7837458    0.94    0.3496
## sal3:older        -0.0148719    0.0305667  184.2334347   -0.49    0.6272
## rel3:yrs65         0.0022248    0.0012180 7308.7729232    1.83    0.0678
## sal3:yrs65         0.0025720    0.0024302  187.6586721    1.06    0.2913
## rel3:sal3:older   -0.0253048    0.0128736 7309.4171463   -1.97    0.0494
## rel3:sal3:yrs65    0.0000419    0.0010274 7314.1630283    0.04    0.9675
##
## Correlation of Fixed Effects:
##            (Intr) rel3   sal3   older  yrs65  rl3:s3 rl3:ld sl3:ld rl3:65 sl3:65
rl3:3:
## rel3        0.197
## sal3       -0.105 -0.231
## older      -0.121 -0.007  0.005
## yrs65       0.000  0.000  0.000 -0.847
## rel3:sal3  -0.203 -0.100  0.251  0.007  0.000
## rel3:older -0.013 -0.065  0.015  0.108 -0.091  0.006
## sal3:older  0.009  0.013 -0.073 -0.080  0.069 -0.014 -0.214
## rel3:yrs65  0.000  0.000  0.000 -0.090  0.105  0.000 -0.847  0.184
```

```
## sal3:yrs65    0.000  0.000  0.000  0.068 -0.081  0.000  0.185 -0.848 -0.221
## rl3:sl3:ldr  0.013  0.006 -0.016 -0.116  0.101 -0.064 -0.115  0.224  0.099 -0.188
## rl3:sl3:y65  0.000  0.000  0.000  0.100 -0.120  0.000  0.099 -0.187 -0.115  0.217 -
0.848

# Btw, here is shorter code to include interactions with all lower-order main effects:
#Cond5 = lmer(data=Example2, REML=TRUE, formula=
#            logRT~1+rel3*sal3*older + rel3*sal3*yrs65+
#            (1+sal3|SubjectID)+(1|ItemID))

print("Omnibus F-test for fixed slopes for interactions with older")

## [1] "Omnibus F-test for fixed slopes for interactions with older"

contestMD(Cond5, ddf="Satterthwaite", L=rbind(c(0,0,0,0,0,0,1,0,0,0,0,0),
            c(0,0,0,0,0,0,0,1,0,0,0,0),c(0,0,0,0,0,0,0,0,0,0,1,0)))

##    Sum Sq Mean Sq NumDF  DenDF F value  Pr(>F)
## 1 1.6882 0.56273     3 579.83  1.4604 0.22431

print("Omnibus F-test for fixed slopes for interactions with yrs65")

## [1] "Omnibus F-test for fixed slopes for interactions with yrs65"

contestMD(Cond5, ddf="Satterthwaite", L=rbind(c(0,0,0,0,0,0,0,0,1,0,0,0),
            c(0,0,0,0,0,0,0,0,0,1,0,0),c(0,0,0,0,0,0,0,0,0,0,0,1)))

##    Sum Sq Mean Sq NumDF  DenDF F value Pr(>F)
## 1 2.1526 0.71754     3 591.35  1.8622 0.1348
```

As shown above, none of the interactions with yrs65 are significant, and the joint F-test of those three slopes is also nonsignificant. However, while the same appears to be true for the interactions with older, they are specific to an older adult who is 65. Thus, we should re-estimate the model excluding the three interactions with yrs65 to see what happens to the interactions with older, as shown next.

Conditional Model 6: Retain Significant Fixed Slope Interactions

Next, we remove the three interactions with yrs65 from the model to see if the three interactions with older are significant without them. This is the final model we will consider, as it is the most parsimonious model that retains all significant fixed effects and random slopes:

Conditional Model 6: $logRT_{tis} = \gamma_{000} + \gamma_{010}(\text{rel}_i - 3) + \gamma_{020}(\text{sal}_i - 3) + \gamma_{030}(\text{rel}_i - 3)(\text{sal}_i - 3)$
$+\gamma_{001}(\text{older}_s) + \gamma_{002}(\text{yrs}_s - 65)$
$+\gamma_{011}(\text{rel}_i - 3)(\text{older}_s) + \gamma_{021}(\text{sal}_i - 3)(\text{older}_s) + \gamma_{031}(\text{rel}_i - 3)(\text{sal}_i - 3)(\text{older}_s)$
$+U_{00s} + U_{020s}(\text{sal}_i - 3) + U_{0i0} + e_{tis}$
$e_{tis} \sim N(0, \sigma_e^2)$
$\begin{bmatrix} U_{00s} \\ U_{02s} \end{bmatrix} \sim MVN \left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \tau_{U_{00S}}^2 & \tau_{U_{00S,012}} \\ \tau_{U_{00S,02S}} & \tau_{U_{02S}}^2 \end{bmatrix} \right)$
$U_{0i0} \sim N(0, \tau_{U_{0I0}}^2)$

```r
print("Conditional Model 6: Remove 3 nonsignificant yrs65 interactions")

## [1] "Conditional Model 6: Remove 3 nonsignificant yrs65 interactions"

Cond6 = lmer(data=Example2, REML=TRUE, formula=
             logRT~1+rel3+sal3+older+yrs65+ rel3:sal3+
                 rel3:older+ sal3:older+ rel3:sal3:older+
                 (1+sal3|SubjectID)+(1|ItemID))
print("Show results using Sattherwaite DDF")

## [1] "Show results using Sattherwaite DDF"

summary(Cond6, ddf="Satterthwaite")

## Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
## Formula: logRT ~ 1 + rel3 + sal3 + older + yrs65 + rel3:sal3 + rel3:older +
##     sal3:older + rel3:sal3:older + (1 + sal3 | SubjectID) + (1 |      ItemID)
##    Data: Example2
##
## REML criterion at convergence: 14896
##
## Scaled residuals:
##    Min    1Q Median    3Q    Max
## -3.047 -0.651 -0.109  0.554  4.159
##
## Random effects:
##  Groups     Name        Variance Std.Dev. Corr
##  SubjectID (Intercept) 0.02341  0.1530
##            sal3        0.00213  0.0462   -0.06
##  ItemID    (Intercept) 0.09433  0.3071
##  Residual              0.38538  0.6208
## Number of obs: 7646, groups:  SubjectID, 153; ItemID, 51
##
## Fixed effects:
##                   Estimate Std. Error        df t value  Pr(>|t|)
## (Intercept)        1.30381    0.04841   59.86445   26.93  < 2e-16
## rel3              -0.06398    0.02395   48.43985   -2.67   0.01025
## sal3              -0.14255    0.04276   49.58601   -3.33   0.00163
## older              0.61200    0.05571  152.49346   10.99  < 2e-16
## yrs65              0.02036    0.00440  151.56236    4.62 0.0000080
## rel3:sal3         -0.00273    0.02014   48.44067   -0.14   0.89274
## rel3:older         0.03786    0.00809 7304.74966    4.68 0.0000029
## sal3:older         0.01276    0.01629  184.38108    0.78   0.43447
## rel3:sal3:older   -0.02481    0.00683 7314.00392   -3.63   0.00028
##
## Correlation of Fixed Effects:
##             (Intr) rel3   sal3   older  yrs65  rl3:s3 rl3:ld sl3:ld
## rel3         0.197
## sal3        -0.105 -0.231
## older       -0.122 -0.007  0.005
## yrs65        0.000  0.000  0.000 -0.844
## rel3:sal3   -0.203 -0.100  0.251  0.007  0.000
## rel3:older  -0.024 -0.122  0.028  0.060 -0.004  0.012
## sal3:older   0.017  0.025 -0.138 -0.041 -0.001 -0.027 -0.212
## rl3:sl3:ldr  0.025  0.012 -0.030 -0.059 -0.003 -0.122 -0.109  0.218
```

Let's interpret the results for the fixed effects in the model above:

fixed intercept γ_{000} =

fixed slope for rel3 γ_{010} =

fixed slope for sal3 γ_{020} =

fixed slope for rel3:sal3 γ_{030} =

fixed slope for older γ_{001} =

fixed slope for yrs65 γ_{002} =

fixed slope for rel3:older γ_{011} =

fixed slope for sal3:older γ_{021} =

fixed slope for rel3:sal3:older γ_{031} =

As we can see, only some of the possible simple slopes are given directly as model parameters (and thus are shown in the default output). To get the others, we can use the `contest1D` function from the `lmerTest` package to generate linear combinations of our fixed effects (and their corresponding standard errors and hypothesis tests using Satterthwaite denominator degrees of freedom here). In the `c=(_)` calls below, the blanks provide multipliers for each fixed effect in the order they appear in the output.

To get predicted slopes for any item and any subject, we use only the fixed effects that involve the predictor the slope is for, then factor that predictor out of the equation, as follows:

$$\text{Predicted logRT} = \gamma_{000} + \gamma_{010}(\text{rel}_i - 3) + \gamma_{020}(\text{sal}_i - 3) + \gamma_{030}(\text{rel}_i - 3)(\text{sal}_i - 3)$$
$$+\gamma_{001}(\text{older}_s) + \gamma_{002}(\text{yrs}_s - 65)$$
$$+\gamma_{011}(\text{rel}_i - 3)(\text{older}_s) + \gamma_{021}(\text{sal}_i - 3)(\text{older}_s) + \gamma_{031}(\text{rel}_i - 3)(\text{sal}_i - 3)(\text{older}_s)$$

$$\text{Predicted rel3 slope} = \gamma_{010}(\text{rel}_i - 3) + \gamma_{030}(\text{rel}_i - 3)(\text{sal}_i - 3)$$
$$+\gamma_{011}(\text{rel}_i - 3)(\text{older}_s) + \gamma_{031}(\text{rel}_i - 3)(\text{sal}_i - 3)(\text{older}_s)$$
$$\text{Predicted rel3 slope} = \gamma_{010} + \gamma_{030}(\text{sal}_i - 3) + \gamma_{011}(\text{older}_s) + \gamma_{031}(\text{sal}_i - 3)(\text{older}_s)$$

$$\text{Predicted sal3 slope} = \gamma_{020}(\text{sal}_i - 3) + \gamma_{030}(\text{sal}_i - 3)(\text{rel}_i - 3)$$
$$+\gamma_{021}(\text{sal}_i - 3)(\text{older}_s) + \gamma_{031}(\text{sal}_i - 3)(\text{rel}_i - 3)(\text{older}_s)$$
$$\text{Predicted sal3 slope} = \gamma_{020}(1) + \gamma_{030}(\text{rel}_i - 3) + \gamma_{021}(\text{older}_s) + \gamma_{031}(\text{rel}_i - 3)(\text{older}_s)$$

$$\text{Predicted rel3:sal3 slope} = \gamma_{030}(\text{rel}_i - 3)(\text{sal}_i - 3) + \gamma_{031}(\text{rel}_i - 3)(\text{sal}_i - 3)(\text{older}_s)$$
$$\text{Predicted rel3:sal3 slope} = \gamma_{030}(1) + \gamma_{031}(\text{older}_s)$$

```
# Get simple slopes to understand pattern of three-way interaction
print("Rel slope at sal=3 for younger"); contest1D(Cond6, L=c(0,1,0,0,0,0,0,0,0))

## [1] "Rel slope at sal=3 for younger"

##     Estimate Std. Error     df t value Pr(>|t|)
## 1 -0.063984   0.023948  48.44 -2.6718 0.010246

print("Rel slope at sal=3 for older");    contest1D(Cond6, L=c(0,1,0,0,0,0,1,0,0))
```

```
## [1] "Rel slope at sal=3 for older"

##    Estimate Std. Error     df t value Pr(>|t|)
## 1 -0.026126   0.024324 51.557 -1.0741  0.28779

print("Sal slope at rel=3 for younger"); contest1D(Cond6, L=c(0,0,1,0,0,0,0,0,0))

## [1] "Sal slope at rel=3 for younger"

##   Estimate Std. Error     df t value  Pr(>|t|)
## 1 -0.14255   0.042761 49.586 -3.3336 0.0016275

print("Sal slope at rel=3 for older");    contest1D(Cond6, L=c(0,0,1,0,0,0,0,1,0))

## [1] "Sal slope at rel=3 for older"

##   Estimate Std. Error     df t value Pr(>|t|)
## 1 -0.12979   0.043601 53.518 -2.9767 0.004368

print("Rel*sal slope for younger");       contest1D(Cond6, L=c(0,0,0,0,0,1,0,0,0))

## [1] "Rel*sal slope for younger"

##      Estimate Std. Error     df  t value Pr(>|t|)
## 1 -0.0027296   0.020138 48.441 -0.13555  0.89274

print("Rel*sal slopefor older");          contest1D(Cond6, L=c(0,0,0,0,0,1,0,0,1))

## [1] "Rel*sal slopefor older"

##    Estimate Std. Error     df t value Pr(>|t|)
## 1 -0.027537   0.020463 51.647 -1.3457  0.18428
```

We can visualize the interaction by creating fake cases to plot predicted outcomes from the fixed effects:

```
############## Make a plot of predicted outcomes ################
# Import fake cases with centered predictor values to be plotted:
#     rel3=-2to2; sal3=-2to1, older=0or1, yrs65=0or15
Fake = read_excel(paste0(filesave,filename),sheet="fake")
# Convert to data frame to use in analysis
Fake = as.data.frame(Fake)
# Merge predicted values from main-effects-only model into FI data
Fake = data.frame(Fake, yhat=predict(object=Cond6, newdata=Fake, re.form=NA))
# Create proper x-axis for plot
Fake$rel = Fake$rel3 + 3

# Make plot
plot(y=Fake$yhat, x=Fake$rel, type="n", ylim=c(-1,3), xlim=c(1,5),
     xlab="Relevance", ylab="Predicted Log-Transformed RT")  #lty=linetype
lines(x=Fake$rel[1:5],   y=Fake$yhat[1:5],   type="l", lty=1, lwd=2, col="purple1")
lines(x=Fake$rel[6:10],  y=Fake$yhat[6:10],  type="l", lty=1, lwd=2, col="blue1")
lines(x=Fake$rel[11:15], y=Fake$yhat[11:15], type="l", lty=1, lwd=2, col="red1")
lines(x=Fake$rel[16:20], y=Fake$yhat[16:20], type="l", lty=2, lwd=2, col="purple1")
lines(x=Fake$rel[21:25], y=Fake$yhat[21:25], type="l", lty=2, lwd=2, col="blue1")
lines(x=Fake$rel[26:30], y=Fake$yhat[26:30], type="l", lty=2, lwd=2, col="red1")
legend(col=c("purple1","blue1","red1","purple1","blue1","red1"),
       lty=c(1,1,1,2,2,2), lwd=c(2,2,2,2,2,2),"bottomleft",
```

```
      legend=c("Age=80 Salience=1",  "Age=65 Salience=1",
               "Younger Salience=1", "Age=80 Salience=4",
               "Age=65 Salience=4",  "Younger Salience=4"))
```
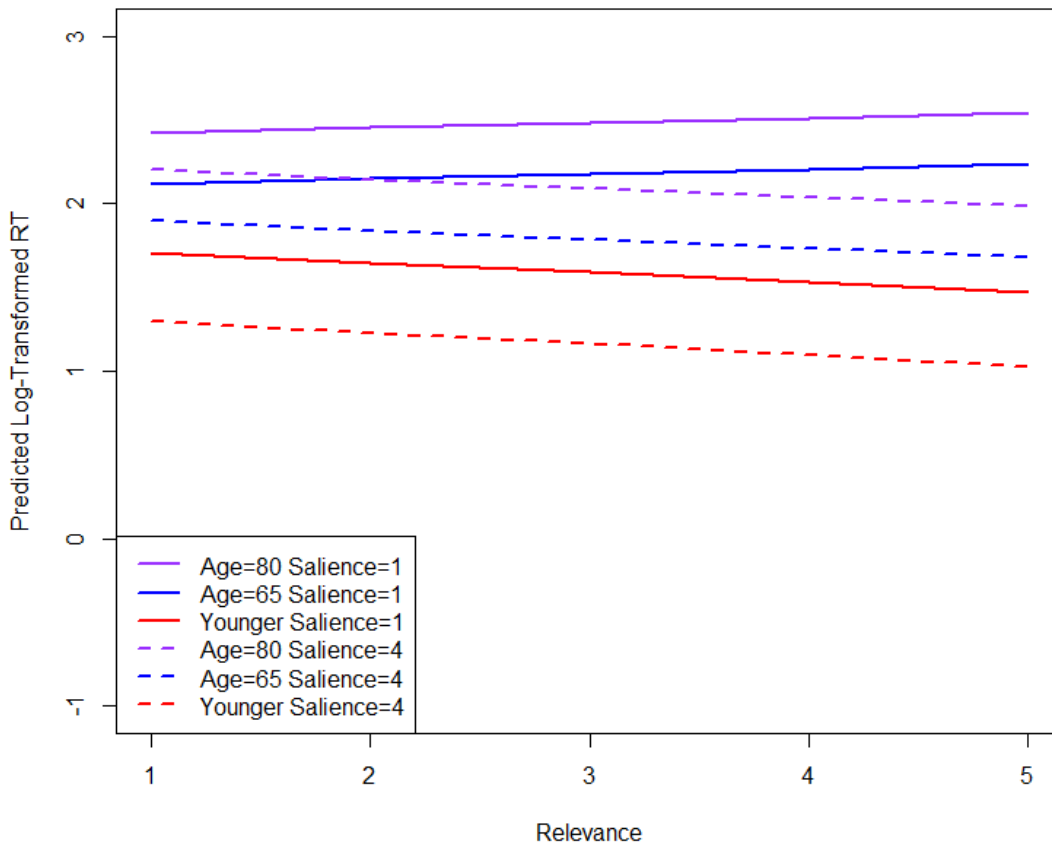

To get an effect size for the contribution of our interactions, we can compute pseudo-R^2 values for the proportion reduction in each variance relative conditional model 3 as a new baseline, as done using saved output below:

```
print("Print stored variance components table for reference")

## [1] "Print stored variance components table for reference"

as.data.frame(VarCorr(Cond6))

##         grp        var1 var2       vcov       sdcor
## 1 SubjectID (Intercept) <NA>  0.02340770  0.152996
## 2 SubjectID        sal3 <NA>  0.00213001  0.046152
## 3 SubjectID (Intercept) sal3 -0.00040937 -0.057976
## 4    ItemID (Intercept) <NA>  0.09432834  0.307129
## 5  Residual        <NA> <NA>  0.38537588  0.620787

# Save each variance as an object
Cond6SubIntVar  = as.data.frame(VarCorr(Cond6))[1,4]
Cond6SubSalVar  = as.data.frame(VarCorr(Cond6))[2,4]
Cond6ItemIntVar = as.data.frame(VarCorr(Cond6))[4,4]
Cond6ResVar     = as.data.frame(VarCorr(Cond6))[5,4]
```

```
# Compute pseudo-R2 for each variance and change relative to Cond3 as new baseline
Cond6SubIntR2        = (Cond3SubIntVar  - Cond6SubIntVar) /Cond3SubIntVar;  Cond6SubIntR2

## [1] -0.0000064235

Cond6SubSalR2        = (Cond3SubSalVar  - Cond6SubSalVar) /Cond3SubSalVar;  Cond6SubSalR2

## [1] 0.11662

Cond6ItemIntR2       = (Cond3ItemIntVar - Cond6ItemIntVar)/Cond3ItemIntVar;
Cond6ItemIntR2

## [1] -0.0016899

Cond6ResR2           = (Cond3ResVar     - Cond6ResVar)    /Cond3ResVar;      Cond6ResR2

## [1] 0.0040391
```

As shown, the three interactions with older reduced two of the four piles of variance. The subject and item random intercept variances were unchanged because we did not add new main effects of either subject or item predictors, respectively. The interaction of sal3:older reduced the subject random sal3 slope variance by 11.7%, and the interactions of rel3:older and rel3:sal3:older reduced the residual variance by 0.4% (as expected given that neither rel3 nor rel3:sal had subject random slope variances estimated in the model).

Lastly, we can get a composite measure of effect size by computing total-R2 as the squared correlation between the observed logRT outcome and that predicted by the model fixed effects, as shown below:

```
# Compute totalR2: squared correlation between y and yhat from fixed effects
print("Save yhat and correlation of yhat with y")

## [1] "Save yhat and correlation of yhat with y"

Example2$Yhat = predict(Cond6, re.form=NA)
Cond6r = cor.test(Example2$Yhat, Example2$logRT, method="pearson")
print("Total R2 = "); Cond6r$estimate^2

## [1] "Total R2 = "

##      cor
## 0.27629
```

As shown, the model fixed slopes accounted for 27.6% of the variance in the observed logRT outcome.

In chapter 12, I then added heterogeneous level-1 residual variance and level-2 random subject variances and covariances by age group, but I was unable to figure out how to do so in R. This is because the function lmer in lmerTest does not allow for the specification of a grouping variable for the level-1 residual variance. While this is possible to do using the function lme in the nlme package, I could not figure out how to get the lme function to fit models with crossed random effects. So if anyone knows how to solve this problem, please let me know! A sample results section is available at the end of chapter 12.

Conditional Model 6R: Remove Subject Random Sal3 Slope for Comparison with Mplus

Mplus does not allow random slopes of one crossed dimension over the other, and so below we re-estimate conditional model 6 without the subject random sal3 slope to compare with Mplus:

```r
# Example of how to export a .csv file for use in Mplus
# Copy data, replace all missing values with -999 for Mplus
Example2_Mplus = subset(Example2,
select=c("SubjectID","ItemID","logRT","older","yrs65","rel3","sal3"))
Example2_Mplus[is.na(Example2_Mplus)] <- -999

# Write to .csv file without column names
write.table(x=Example2_Mplus, col.names=FALSE, row.names=FALSE, sep=",",
            file=paste0(filesave,"Example2_Mplus.csv"))

print("Conditional Model 6R: Remove random subject salience slope for Mplus comparison")
```

```
## [1] "Conditional Model 6R: Remove random subject salience slope for Mplus comparison"
```

```r
Cond6Mplus = lmer(data=Example2, REML=TRUE, formula=
              logRT~1+rel3+sal3+older+yrs65+ rel3:sal3+
              rel3:older+ sal3:older+ rel3:sal3:older+
              (1|SubjectID)+(1|ItemID))
print("Show results using Sattherwaite DDF")
```

```
## [1] "Show results using Sattherwaite DDF"
```

```r
summary(Cond6Mplus, ddf="Satterthwaite")
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method ['lmerModLmerTest']
## Formula: logRT ~ 1 + rel3 + sal3 + older + yrs65 + rel3:sal3 + rel3:older +
##     sal3:older + rel3:sal3:older + (1 | SubjectID) + (1 | ItemID)
##    Data: Example2
##
## REML criterion at convergence: 14903
##
## Scaled residuals:
##    Min     1Q Median     3Q    Max
## -3.134 -0.654 -0.108  0.554  4.143
##
## Random effects:
##  Groups    Name        Variance Std.Dev.
##  SubjectID (Intercept) 0.0233   0.153
##  ItemID    (Intercept) 0.0942   0.307
##  Residual              0.3880   0.623
## Number of obs: 7646, groups:  SubjectID, 153; ItemID, 51
##
## Fixed effects:
##                  Estimate Std. Error        df t value  Pr(>|t|)
## (Intercept)       1.30380    0.04839  59.84436   26.94   < 2e-16
## rel3             -0.06399    0.02394  48.45094   -2.67    0.0102
## sal3             -0.14251    0.04249  48.44190   -3.35    0.0016
## older             0.61385    0.05571 152.45257   11.02   < 2e-16
## yrs65             0.02017    0.00440 151.60394    4.58 0.0000096
## rel3:sal3        -0.00272    0.02013  48.45145   -0.14    0.8929
## rel3:older        0.03796    0.00811 7442.18741    4.68 0.0000029
## sal3:older        0.01272    0.01439 7442.71851    0.88    0.3770
## rel3:sal3:older  -0.02480    0.00685 7443.04242   -3.62    0.0003
##
```

```
## Correlation of Fixed Effects:
##            (Intr) rel3   sal3   older  yrs65  rl3:s3 rl3:ld sl3:ld
## rel3        0.197
## sal3       -0.104 -0.232
## older      -0.122 -0.007  0.004
## yrs65       0.000  0.000  0.000 -0.844
## rel3:sal3  -0.203 -0.100  0.252  0.007  0.000
## rel3:older -0.024 -0.122  0.028  0.060 -0.004  0.012
## sal3:older  0.013  0.028 -0.122 -0.033 -0.001 -0.031 -0.241
## rl3:sl3:ldr 0.025  0.012 -0.030 -0.060 -0.003 -0.122 -0.109  0.248
```

Please see the Mplus results in a separate .out file.