Sample R Demonstration Notes

Multilevel Models for Hierarchical Data

Daniel J. Bauer & Patrick J. Curran

CenterStat.org

Copyright © 2024 Curran-Bauer Analytics

All rights reserved

No part of these notes may be reproduced, translated, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission by the copyright holders.

Chapter 2
Basic Multilevel Models

Birth Order Data Analysis 2-3
Random Effects ANOVA 2-3
Adding Lower-Level Predictors to the Model 2-8
Adding Upper-Level Predictors to the Model 2-10

Birth Order Data Analysis

For this demonstration we will again use the **birth.csv** data file (after deleting the outlier as shown in Chapter 1). Accompanying **R** code is in the **ch02_birth.R** file. For brevity, the first few lines of syntax, loading packages and reading in the data, are not repeated here, as they are mostly the same as in Chapter 1. One exception is that we use some additional packages in this chapter. In particular, to fit most of the other linear multilevel models in this course, we will use the **lme4** package. We will also routinely use the **lmerTest** package to obtain p-values for fixed effect estimates and Type III tests for multiple degree of freedom main effects and interactions (using Satterthwaite degrees of freedom as the default). Lastly, we will use the **performance** package to compute the intra-class correlation (ICC).

Random Effects ANOVA

We can get a crude sense of how meaningful between-family differences in Math IQ are by computing and plotting the distribution of family mean IQ. We do this by first sorting the data my **mom_id**, then creating a new variable, **mmath**, containing the mean IQ for each family. We do this by using the **aggregate** function to generate a new data set called **family** to contain the mean of math (renamed to **mmath**) for each unique **mom_id** value. Then we merge this back with the original data using the **join** function of the **plyr** package. We will use these means in two plots, an index plot and a histogram of family means.

```
birth <- birth[order(birth$mom_id) , ]
means <- aggregate(math~mom_id,birth,mean)
names(means)<- c("mom_id","mmath")
birth <- join(birth, means, by="mom_id")

head(x = birth, n = 10)
```

This is some of the data including the new **mmath** variable containing the means:

```
   mom_id kid_id brthordr math cohort brthage mmath
1       3    301        1  102      1      19 102.0
2       4    401        1  105      0      18 105.0
3       8    802        2  102      1      17  99.5
4       8    803        3   97      1      17  99.5
5      20   2001        1  112      0      30 112.0
6      25   2501        1  105      0      30 105.0
7      43   4301        1   97      1      21  81.0
8      43   4302        2   65      0      21  81.0
9      44   4402        2  100      1      20 100.0
10     49   4901        1  126      1      24 107.5
```

Notice that kids within the same family (e.g., **kid_id 4301** and **4302**) have the same family mean **mmath** value.

Next, we produce our plots. For the index plot, we first used a for loop to create a variable called **first** that is equal to 1 for the first observation in each **mom_id** and equal to 0 otherwise. Then we used a second for loop to create a variable called **family** by computing a

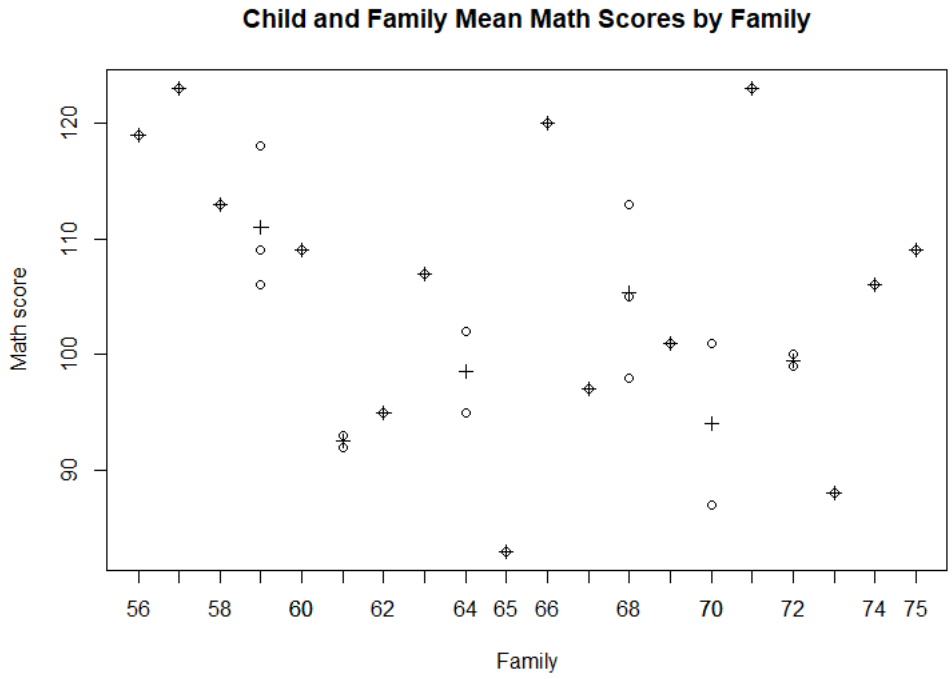
cumulative sum of the **`first`** variable. This produces family identifiers numbered 1 to 2207 from which we can select a subset of families of interest. Note that the index plot is made on a subset of the families in the data (in this case, the 56th through 75th families).

```
n_row <- length(birth$mom_id)
birth <- cbind(birth, first = 1)
for(i in 2:n_row)
{
  if (birth$mom_id[i]==birth$mom_id[i-1])
  {birth$first[i]=0}
}
birth <- cbind(birth, family = 1)
fam <- 0
for(i in 1:n_row)
{
  fam = fam + birth$first[i]
  birth$family[i] = fam
}
```

We then select only the 56th through 75th families and generate the plot via this code:

```
pltdata <- subset(birth, subset = (birth$family >= 56 & birth$family<=75))
attach(pltdata)

plot(family, math,
     main = "Child and Family Mean Math Scores by Family",
     ylab = "Math score",
     xlab = "Family",
     points(family, mmath, pch=3))
axis(1,at=seq(56,75,1))
```



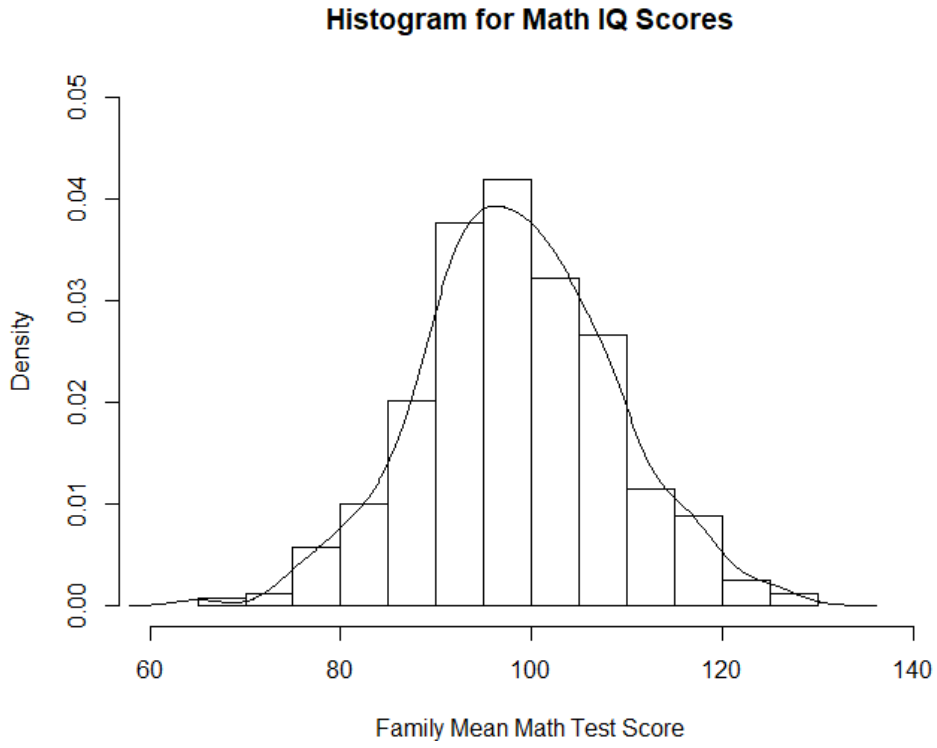
Within the plot, the circles are the individual children's scores whereas the plus signs indicate the means for each family. The index plot shows that there is both within- and between-family variation. There are individual differences among siblings within a family, but also differences in mean math IQ across families.

For the histogram, we selected only families with 2+ children in the data because otherwise the family mean and child score are identical. Thus, we first created the **`count`** variable containing **`N`**, the number of observations per **`mom_id`**.

```
birth <- cbind(birth, count = c(n_row))
birth$count <- as.numeric(ave(birth$mom_id, birth$mom_id, FUN = length))

pltdata2 <- subset(birth, subset = (birth$count >= 2 & birth$first==1))
attach(pltdata2)

hist(mmath,
     main="Histogram for Math IQ Scores",
     xlab="Family Mean Math Test Score",
     xlim=c(60,140),
     ylim=c(0,0.05),
     prob=TRUE,
     breaks=20)
lines(density(mmath))
```



Focusing on the across-family differences, the histogram shows a great deal of family-to-family variation in math IQ, and that this variation follows the form of a normal distribution. Of course, some of the variation is due to simple sampling error (since each family mean is

computed from very few observations), so we would like to more formally evaluate within- versus between-family IQ differences. The random-effects ANOVA model provides a way to decompose this variance.

The random-effects ANOVA model is

Level 1:

$$y_{ij} = \beta_{0j} + r_{ij} \qquad r_{ij} \sim N\left(0, \sigma^2\right)$$

Level 2:

$$\beta_{0j} = \gamma_{00} + u_{0j} \qquad u_{0j} \sim N\left(0, \tau_{00}\right)$$

Reduced-Form:

$$y_{ij} = \gamma_{00} + u_{0j} + r_{ij}$$

The code for fitting this model using the **lmer** function of **lme4** is shown here:

```
re_ANOVA <- lmer(math ~ (1|mom_id), data = birth)
summary(re_ANOVA)
icc(re_ANOVA)
```

Syntax Notes:

- Within the **lmer** function:
  - Similar to the **lm** function described in Chapter 1, the name of the outcome variable, **math**, is the first argument after the parenthesis and precedes the squiggly "regression operator" **~**, whereas predictors are listed immediately after the regression operator **~**. Here, there are no predictors included yet; a fixed intercept is estimated by default.
  - The syntax **(… | …)** after the regression operator **~** defines the random effect. The variables to the left of **|** indicate the effects that should be random; here it is a **1** for a random intercept (no other random effects are included). The variable to the right of **|** is the clustering variable; here it is **mom_id**.
  - The Restricted Maximum Likelihood Estimator (REML estimator) is the default within **lmer** to fit the multilevel model. You can also specify this with **reml = TRUE** in the model line.
  - The final argument in the model statement specifies the data set
- Results from fitting this model are stored in the **re_ANOVA** object for post-processing.
- To print results of the model, we specify **summary(re_ANOVA)**. When the **lmerTest** package has been loaded, the output will include t-tests with p-values for the fixed effects based on Satterthwaite degrees of freedom.
- **icc(re_ANOVA)** computes the ICC for the model. The **icc** function is part of the **performance** package.

When you run the model, make sure you do not receive a warning that estimation failed to converge. *Do not interpret your model estimates if the model fails to converge – this may indicate problems with your model specification (or a need to increase the number of iterations for estimation).*

The **summary** function generates the following output:

```
Linear mixed model fit by REML. t-tests use Satterthwaite's method
['lmerModLmerTest']
Formula: math ~ (1 | mom_id)
   Data: birth

REML criterion at convergence: 25942.8

Scaled residuals:
     Min       1Q   Median       3Q      Max
-2.88328 -0.51810  0.02905  0.52969  3.11466

Random effects:
 Groups   Name        Variance Std.Dev.
 mom_id   (Intercept)  52.15    7.222
 Residual             102.28   10.113
Number of obs: 3312, groups:  mom_id, 2207

Fixed effects:
             Estimate Std. Error        df t value Pr(>|t|)
(Intercept)   98.9115     0.2386 2157.0732   414.5   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

This portion of the output provides the variance component estimates:

```
Random effects:
 Groups   Name        Variance Std.Dev.
 mom_id   (Intercept)  52.15    7.222
 Residual             102.28   10.113
```

Here, **mom_id(Intercept)** and **Residual** label the estimates for the random intercept variance and the residual variance, respectively. Note that **lme4** does not report inference tests of the variance parameter estimates. You can use the **confint( )** function to obtain confidence intervals for the standard deviations of the random effects, by either likelihood profile or parametric bootstrap, if desired. We could also conduct a likelihood ratio test to see if including the random intercept in the model is really necessary, but given the very large estimate, we can guess that this test would be highly significant. We show this LRT in Chapter 3.

Next, we obtain the estimates for the fixed effects:

```
Fixed effects:
             Estimate Std. Error        df t value Pr(>|t|)
(Intercept)   98.9115     0.2386 2157.0732   414.5   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Recall that **(Intercept)** is the internal label that **R** uses for the intercept and that, in the absence of predictors, the intercept is interpretable as the mean of the outcome. Thus, this table shows that the average value of Math IQ is 98.91. Note that this will not exactly equal the grand mean ignoring clustering because estimation of the multilevel model implicitly adjusts for the partial redundancy in the data due to dependence.

Last, we obtain the ICC with the **icc** function of the **performance** package:

```
    Adjusted ICC: 0.338
  Conditional ICC: 0.338
```

We see that the model implies that 34% of the variance in math scores is associated with between-family differences. An equally valid interpretation is that, within a given family, math scores have a .34 correlation. Thus, the ICC is a direct index of the degree of dependence in the data.

Adding Lower-Level Predictors to the Model

We will now add fixed effects for birth order, cohort and birth order × cohort to the model at Level 1. Our model will now be:

Level 1:

$$Math_{ij} = \beta_{0j} + \beta_{1j}Second_{ij} + \beta_{2j}Third_{ij} + \beta_{3j}Fourth_{ij} + \beta_{4j}Old_{ij} + \beta_{5j}Second_{ij} \times Old_{ij} + \beta_{6j}Third_{ij} \times Old_{ij} + \beta_{7j}Fourth_{ij} \times Old_{ij} + r_{ij}$$

$$r_{ij} \sim N\left(0, \sigma^2\right)$$

Level 2:

$$\beta_{0j} = \gamma_{00} + u_{0j} \qquad u_{0j} \sim N\left(0, \tau_{00}\right)$$

$$\beta_{pj} = \gamma_{p0} \text{ for } p > 0$$

Reduced-Form:

$$Math_{ij} = \gamma_{00} + \gamma_{10}Second_{ij} + \gamma_{20}Third_{ij} + \gamma_{30}Fourth_{ij} + \gamma_{40}Old_{ij} + \gamma_{50}Second_{ij} \times Old_{ij} + \gamma_{60}Third_{ij} \times Old_{ij} + \gamma_{70}Fourth_{ij} \times Old_{ij} + u_{0j} + r_{ij}$$

Because **brthordr** and **cohort** are categorical variables, we first need to specify them as factor variables in the data frame (just as we did in Chapter 1):

```
birth$brthordr <- factor(birth$brthordr)
birth$cohort <- factor(birth$cohort)
```

To include the interaction between these variables we specify **cohort*brthordr.**

```
ll_predictors <- lmer(math ~ cohort + brthordr + cohort*brthordr +
                      (1|mom_id), data = birth)
```

By default, **R** will perform reference group coding for factor variables, using the first level of each factor as the reference group.

To obtain Type III tests of multiple degree of freedom main effects and interactions (tests of partial effects for each factor controlling for all others), we can use the **anova** function. With the **lmerTest** package loaded, this function is modified to be used with **lmer** output objects with an option to request Type III tests.

```
anova(ll_predictors, type="III")
```

```
Type III Analysis of Variance Table with Satterthwaite's method
                Sum Sq Mean Sq NumDF  DenDF F value    Pr(>F)
cohort          3049.8 3049.76     1 2844.7 30.0193 4.652e-08 ***
brthordr        3514.5 1171.50     3 2435.2 11.5313 1.665e-07 ***
cohort:brthordr  103.8   34.59     3 2740.2  0.3405    0.7961
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

We see that both main effects are significant but the interaction is not.

We can now use the **summary** function to obtain the estimates of the model parameters.

```
summary(ll_predictors)
```

Abridged results follow:

```
REML criterion at convergence: 25858.9

Random effects:
 Groups   Name        Variance Std.Dev.
 mom_id   (Intercept)  49.3     7.022
 Residual             101.6    10.079
Number of obs: 3312, groups:  mom_id, 2207

Fixed effects:
                    Estimate Std. Error         df t value Pr(>|t|)
(Intercept)        101.49825    0.47425 3293.27530 214.021  < 2e-16 ***
cohort1             -3.26858    0.60720 3294.67768  -5.383 7.84e-08 ***
brthordr2           -1.53032    0.66714 2996.66903  -2.294 0.021867 *
brthordr3           -2.73512    0.80364 2980.12398  -3.403 0.000674 ***
brthordr4           -3.49268    1.22636 2633.55797  -2.848 0.004433 **
cohort1:brthordr2    0.34211    0.92603 2991.86327   0.369 0.711829
cohort1:brthordr3   -0.99399    1.28080 2827.47143  -0.776 0.437770
cohort1:brthordr4   -0.08046    2.09163 2630.21807  -0.038 0.969319
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

The conditional (residual) variance estimates seen in the **Random effects** table can be compared to the marginal (total) variance estimates obtained from the random-effects ANOVA results to see variance explained at each level of the model. Note that it is the between-family variance that has been most reduced by the inclusion of the Level 1 predictors.

The **Fixed effects** table shows the individual effects of the coding variables. Within the table, the reference category for categorical variables is omitted from the output (here first born children and the younger cohort). Similar to the results we obtained in Chapter 1 when ignoring the nesting, we see there is a monotonic decrease in math IQ with birth order in both cohorts and the older cohort has somewhat lower scores across the board.

Adding Upper-Level Predictors to the Model

We now include the mother's age when she had her first child as a Level 2 predictor of intercepts, to undo confounding between maternal age and birth order:

Level 1:

$$Math_{ij} = \beta_{0j} + \beta_{1j}Second_{ij} + \beta_{2j}Third_{ij} + \beta_{3j}Fourth_{ij} + \beta_{4j}Old_{ij} + \beta_{5j}Second_{ij} \times Old_{ij} + \beta_{6j}Third_{ij} \times Old_{ij} + \beta_{7j}Fourth_{ij} \times Old_{ij} + r_{ij}$$

$$r_{ij} \sim N\left(0, \sigma^2\right)$$

Level 2:

$$\beta_{0j} = \gamma_{00} + \gamma_{01}Birthage_j + u_{0j} \qquad u_{0j} \sim N\left(0, \tau_{00}\right)$$

$$\beta_{pj} = \gamma_{p0} \text{ for } p > 0$$

Reduced-Form:

$$Math_{ij} = \gamma_{00} + \gamma_{01}BirthAge_j + \gamma_{10}Second_{ij} + \gamma_{20}Third_{ij} + \gamma_{30}Fourth_{ij} + \gamma_{40}Old_{ij} + \gamma_{50}Second_{ij} \times Old_{ij} + \gamma_{60}Third_{ij} \times Old_{ij} + \gamma_{70}Fourth_{ij} \times Old_{ij} + u_{0j} + r_{ij}$$

The corresponding **lmer** syntax follows. Note that **brthage** is first mean-centered to simplify interpretation of the model intercept.

```
birth$brthage.c <- birth$brthage - mean(birth$brthage)
ul_predictors <- lmer(math ~ cohort + brthordr + cohort*brthordr + brthage.c
                      + (1|mom_id), data = birth)
anova(ul_predictors, type="III")
summary(ul_predictors)
```

We first consider the multiple degree of freedom tests from the **anova** function:

```
Type III Analysis of Variance Table with Satterthwaite's method
                Sum Sq Mean Sq NumDF  DenDF  F value    Pr(>F)
cohort           742.1   742.1     1 2842.7   7.3006  0.006934 **
brthordr         412.7   137.6     3 2446.7   1.3533  0.255341
brthage.c      11492.6 11492.6     1 2703.8 113.0632 < 2.2e-16 ***
cohort:brthordr  560.4   186.8     3 2820.1   1.8377  0.138113
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Notably, both the main effect of birth order and the interaction of birth order with cohort are now non-significant.

We next consider the parameter estimates from the **summary** function (abridged output):

```
Random effects:
 Groups   Name        Variance Std.Dev.
 mom_id   (Intercept)  43.34    6.583
 Residual             101.65   10.082
Number of obs: 3312, groups:  mom_id, 2207

Fixed effects:
                   Estimate Std. Error         df t value Pr(>|t|)
(Intercept)         99.11332    0.51753 3302.28789 191.511   <2e-16 ***
cohort1             -0.27557    0.66041 3286.06285  -0.417   0.6765
brthordr2            0.01715    0.67462 3049.69365   0.025   0.9797
brthordr3            0.16913    0.83965 3014.70194   0.201   0.8404
brthordr4            0.11717    1.26265 2667.77061   0.093   0.9261
brthage.c            0.73253    0.06889 2703.81600  10.633   <2e-16 ***
cohort1:brthordr2   -0.58993    0.91860 3049.47064  -0.642   0.5208
cohort1:brthordr3   -2.79835    1.27750 2928.09095  -2.190   0.0286 *
cohort1:brthordr4   -2.46952    2.08247 2725.67705  -1.186   0.2358
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

In the fixed effects table, the effect each of the dummy variable for birth order is non-significant. One dummy variable for the birth order by cohort interaction is significant, suggesting a significant difference between first and third born children in the older cohort, but this should be interpreted cautiously (or not at all) given the lack of a significant overall cohort by birth order interaction in the **anova** function output.