Exploring and Simulating With a Multilevel Model

DJ Passey

Jan 4, 2023

```
library(renv)
```

```
##
## Attaching package: 'renv'
```

```
## The following objects are masked from 'package:stats':
##
##     embed, update
```

```
## The following objects are masked from 'package:utils':
##
##     history, upgrade
```

```
## The following objects are masked from 'package:base':
##
##     autoload, load, remove
```

```
library(here)
```

```
## here() starts at /Users/djpassey/Code/graphinference
```

```
# Activate R virtual environment
# The `here()` function should locate the top level
# directory of the enclosing git repository called `graphinference`
proj_dir = here()
print(cat("Double check that project directory\n",
          "is named graphinference:\n",
          "proj_dir =", proj_dir, "\n"))
```

```
## Double check that project directory
##  is named graphinference:
##  proj_dir = /Users/djpassey/Code/graphinference
## NULL
```

```
renv::activate(proj_dir)

library(lme4)
```

```
## Loading required package: Matrix
```

Mixed Effect vs Multilevel Model

Mixed Effect Model

A mixed effect model can be written in the form:

$$Y = \mu + X\beta + Z\alpha + \epsilon$$

Here, β represents the fixed effects, and α represents the random effects. The matrix X contains all the factors from the data, either continuous or one hot encoded categorical values.

The matrix Z includes the random effect factors. If no random slopes are to be estimated, then Z will be a zero-one matrix. In this case, the α values are effectively intercepts corresponding to class inclusion.

However, when random slopes are included in the model we model class inclusion as having an impact on the slope of the individuals response to a given variable. For example, if we are measuring blood pressure and predicting mood, we might want to model each person in the study as having a random intercept (some are happier than others) and also having different relationships between blood pressure and mood. For some, higher blood pressure means they are excited and happy, for others, higher blood pressure means they are anxious.

In this case, a row of the Z matrix would have n entries where only one of them, entry i, is equal to one. This is the one hot encoding of person ID and corresponds to the random intercept. The next n entries would correspond to blood pressure, one hot encoded to align with person ID. That is, at the same relative location in the second half of the row, entry $n+i$, the Z matrix would have the person's blood pressure and every other entry between n and $2n$ would be zero.

Thus α_i is the random intercept of mood for person i and α_{n+i} is the random slope that models how the blood pressure of person i affects their mood.

However, the model does not estimate the random effects. It only estimates the *variance* of the random effects.

Therefore, to make a prediction, about person i you would need to draw a random intercept and slope from normal distributions with zero mean and the estimated variances of the random intercept and slope respectively. Then you could pass data to your model.

What is weird about this is that mixed effect models predict differently every time. In order to capture person specific information, you would probably want to fit a new regression without random effects just to that specific person.

Great video about this here.

Multilevel model

A multilevel model is a specific kind of mixed effect model. It is usually written as two "levels". The example equations given here represent the least parsimonious multilevel model, given the number of covariates considered. It is possible to specify multilevel models that omit some of the following terms.

Level one:

$$y_{ij} = \beta_{0j} + \beta_{1j}\, x_{1ij} + \beta_{2j}\, x_{2ij} + e_{ij}$$

Here y_ij represents the dependent variable. The index j denotes the group and ranges from $1, 2, ..., g$ and the variable i denotes the within-group index and ranges from $1, 2, ..., N_g$. Each group can have a different number of observed values of y, thus N_j, the total number of observations varies based on group.

For example, a modeler might try to predict income, grouping by race. If there are five races $g = 5$. If the first race, $j = 1$, is Native American and the second race, $j = 2$ is Asian and the data includes 300 Native American income values and 100 Asian income values, then $N_1 = 300$ and $N_2 = 100$.

The next value, β_{0j} is an intercept that varies depending on the group. Continuing with our previous example, β_{02} would be the intercept for Asian income.

The coefficient β_{1j} is a coefficient that varies based on group. It represent the effect of the first factor in the model x_{1ij} on income. If the first factor is education, then the group indexing of β_{1j} allows the effect of education on income to vary among the groups.

The first factor is represented by x_{1ij}. The first index is 1 and denotes that this is the first factor in the model. The index j denotes the group membership of this observation and i denotes the index of the observation as it relates to the total number of observations within the group j.

The next term $\beta_{2j}x_{2ij}$ has the same interpretation as the term above, except relating to the second factor in the model. Continuing with our example, x_{2ij} might represent age.

We assume $e_{ij} \sim \mathsf{N}(0, \sigma_e)$.

Level Two: In a multilevel model, we can describe each of the β values from the level one equation with another equation.

$$\beta_{0j} = \gamma_{00} + \gamma_{01}w_{1j} + \gamma_{02}w_{2j} + u_{0j}\beta_{1j} = \gamma_{10} + \gamma_{11}w_{1j} + \gamma_{12}w_{2j} + u_{1j}\beta_{2j} = \gamma_{20} + \gamma_{21}w_{1j} + \gamma_{22}w_{2j} + u_{2j}$$

In each of these equations, we describe β_{kj} a linear combination of an intercept, group specific covariates, $w1j$ and $w2j$, and an error term, u_{kj}.

It is important that the group specific covariates take on only one value per group. In our example where the data is grouped by race, $w1j$ might represent the percent of the United States that identifies as that race. Then, w_{11} be the percent of the U.S. that identifies as Native American because $j = 1$ is the group index for Native American. Notice that to know the value of w_{1j} all we need is a specific group index j. As another example is w_{2j} might equal the number of millionaires who identify as that race. Once again, the special covariates w_{1j} and w_{2j} are uniform accross the *entire* group and do not vary within the group.

As a counter-example, w_{1j} and w_{2j} cannot correspond to variables like age, because this covariate is not the same for everyone of a particular race. To see that this aligns with the notation, let's imagine that w_{1j} did correspond to age. If this were true we would need an additional index, e.g. w_{1ij} to denote variation in the value of w_{1j} that cannot be explained by group membership alone.

This requirement on the ws comes from the assumption that some covariates are "nested". (But if a covariate is not nested, it can still be included in the model as an additional covariate e.g. x_{3ij}.)

The variable u_{kj}s are modeled as correlated mean zero gaussian random variables.

Interactions

For the second level two equation, we recall that in our example, $beta_{1j}$ models the effect of education on income. Since

$$\beta_{1j}x_{1ij} = \big(\gamma_{10} + \gamma_{11}w_{1j} + \gamma_{12}w_{2j} + u_{1j}\big)x_{1ij}\beta_{1j}x_{1ij} = \gamma_{10}\,x_{1ij} + \gamma_{11}w_{1j}\,x_{1ij} + \gamma_{12}w_{2j}\,x_{1ij} + u_{1j}\,x_{1ij}$$

each sub-term of β_{1j} captures a different way that group, and group covariates relate to the dependent variable.

In our example, the first term $\gamma_{10}x_{1ij}$ is a fixed effect that models the overall linear relationship between education and income without taking race into account.

The next two terms, $\gamma_{11}w_{1j}\,x_{1ij} + \gamma_{12}w_{2j}\,x_{1ij}$ are interaction terms that represent the interaction between x_{1ij} and the group specific covariates w_{1j} and w_{2j}. In our example these represent the interaction between education and the percent of the population that identifies as race j and education and the number of CEOs that identify as race j. (Our example breaks dow here as these interaction terms seem sort of silly to model.)

The last term $u_{1j}\,x_{1ij}$ represents a random slope that describes the manner in which the relationship between education and income changes for each race. We model the standard deviation and correlation of us just like we do for normal random effects.

Reduced Equation

The reduced equation is when we replace each β with its level 2 equation. This gives:

$$ y_{ij} = \gamma

InstEval is a dataset of how students rate lectures. The dataset was created with the goal of identifying the most liked instructor. The columns are:

- `s` a factor with levels 1:2972 denoting individual students.
- `d` a factor with 1128 levels from 1:2160, denoting individual professors or lecturers.
- `studage` an ordered factor with levels $2 < 4 < 6 < 8$, denoting student's "age" measured in the semester number the student has been enrolled.
- `lectage` an ordered factor with 6 levels, $1 < 2 < \ldots < 6$, measuring how many semesters back the lecture rated had taken place.
- `service` a binary factor with levels 0 and 1; a lecture is a "service", if held for a different department than the lecturer's main one.
- `dept` a factor with 14 levels from 1:15, using a random code for the department of the lecture.
- `y` a numeric vector of ratings of lectures by the students, using the discrete scale 1:5, with meanings of 'poor' to 'very good'.

Each observation is one student's rating for a specific lecture (of one lecturer, during one semester in the past).

If our goal is to find the best liked lecturer, we are interested in the relative sizes of the effect of `d` on `y`. The best liked lecturer should have the biggest positive effect on `y`.

However, we would like to include in the regression the fact that some students may give mainly positive ratings while others may give more negative ones. Similarly, students who are older may give different types of ratings. We can control for this by including a random intercept for each student and a random intercept for student age.

Similarly, some departments may get higher ratings because their subjects are well liked. We can control for this so that the effect of the professor is department agnostic by including a random intercept for department.

There is a column for service denoting if a professor lectured in their main department or a different one. It is possible that this column has interactions with department and professor, that is, for some departments, service may improve your rating, and for others, service may decrease your rating. Similarly, there may be professors who do better outside their department, and other professors who don't.

My inclanation is to control for this with a random intercept on `service:d`

Check that instructor (`d` column) is nested in `dept`. In other words, each instructor belongs to a single department.

```
all(rowSums(xtabs(~ d + dept, InstEval) != 0L ) == 1)
```

```
## [1] TRUE
```

Check that each student is associated with exactly one studage. (Otherwise we would have ratings from students across multiple years.)

```
all(rowSums(xtabs(~ s + studage, InstEval) != 0L ) == 1)
```

```
## [1] TRUE
```

Regressions

```
instr_eval1 <- lm(y ~ d, data=InstEval)
summary(instr_eval1)
```

```
##
## Call:
## lm(formula = y ~ d, data = InstEval)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.3636 -0.9298  0.0588  0.9813  3.3333
```

```
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.727e+00  3.685e-01  10.113  < 2e-16 ***
## d6          -9.531e-01  4.290e-01  -2.222 0.026305 *
## d7           3.333e-01  4.256e-01   0.783 0.433468
## d8          -1.134e+00  4.014e-01  -2.825 0.004730 **
## d12         -1.606e-01  4.309e-01  -0.373 0.709324
## d13         -3.497e-02  4.173e-01  -0.084 0.933225
## ...
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.222 on 72293 degrees of freedom
## Multiple R-squared:  0.1725, Adjusted R-squared:  0.1596
## F-statistic: 13.37 on 1127 and 72293 DF,  p-value: < 2.2e-16
```

```
instr_eval2 <- lm(y ~ service + dept + d, data=InstEval)
summary(instr_eval2)
```

```
##
## Call:
## lm(formula = y ~ service + dept + d, data = InstEval)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4308 -0.9198  0.0595  0.9708  3.2984
##
## Coefficients: (13 not defined because of singularities)
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.7272727  0.3684220  10.117  < 2e-16 ***
## service1    -0.0973856  0.0136723  -7.123 1.07e-12 ***
## dept5       -0.5809547  0.4850535  -1.198 0.231033
## dept10       0.0505051  0.4209623   0.120 0.904503
## dept12      -0.7714910  0.4396144  -1.755 0.079276 .
## dept6       -1.5734266  0.4395008  -3.580 0.000344 ***
## dept7       -1.3272727  0.4850500  -2.736 0.006214 **
## dept4       -0.1939394  0.4850500  -0.400 0.689280
## dept8       -0.1071599  0.4121351  -0.260 0.794856
## dept9       -0.0877652  0.3809535  -0.230 0.817795
## dept14      -1.2209398  0.3836084  -3.183 0.001459 **
## dept1       -0.2272727  0.5338941  -0.426 0.670336
## dept3       -0.4965035  0.5005869  -0.992 0.321278
## dept11       0.0766488  0.3999897   0.192 0.848035
## dept2       -0.7936066  0.3869406  -2.051 0.040273 *
## d6          -0.1784467  0.3250867  -0.549 0.583062
## d7           0.3540701  0.2633004   1.345 0.178714
## d8          -0.3510072  0.2877547  -1.220 0.222539
## d12          1.1414884  0.2471467   4.619 3.87e-06 ***
## d13         -0.0754818  0.2824192  -0.267 0.789263
## ...
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.222 on 72292 degrees of freedom
```

```
## Multiple R-squared:  0.173,  Adjusted R-squared:  0.1601
## F-statistic: 13.41 on 1128 and 72292 DF,  p-value: < 2.2e-16
```

```
instr_eval3 <- lm(y ~ service + dept + d + service:d, data=InstEval)
summary(instr_eval3)
```

```
##
## Call:
## lm(formula = y ~ service + dept + d + service:d, data = InstEval)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4545 -0.9032  0.0556  0.9592  3.3846
##
## Coefficients: (479 not defined because of singularities)
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     3.727e+00  3.663e-01  10.175  < 2e-16 ***
## service1       -1.238e+00  4.748e-01  -2.606 0.009150 **
## dept5          -4.196e-01  4.977e-01  -0.843 0.399217
## dept10          5.051e-02  4.185e-01   0.121 0.903953
## dept12         -1.442e+00  5.874e-01  -2.454 0.014123 *
## dept6          -1.573e+00  4.370e-01  -3.601 0.000318 ***
## dept7          -1.327e+00  4.823e-01  -2.752 0.005921 **
## dept4          -1.939e-01  4.823e-01  -0.402 0.687577
## dept8           1.033e+00  6.270e-01   1.648 0.099453 .
## dept9          -5.622e-02  3.793e-01  -0.148 0.882177
## dept14         -1.760e-02  4.264e-01  -0.041 0.967082
## dept1          -2.273e-01  5.308e-01  -0.428 0.668542
## dept3          -4.965e-01  4.977e-01  -0.998 0.318486
## dept11          2.727e-01  9.339e-01   0.292 0.770262
## dept2           2.727e-01  5.874e-01   0.464 0.642432
## d6              4.143e-01  5.100e-01   0.812 0.416562
## d7              1.298e+00  1.004e+00   1.293 0.196029
## d8              1.951e-01  4.891e-01   0.399 0.690049
## d12            -1.310e+00  5.855e-01  -2.237 0.025297 *
## d13            -3.492e-02  2.884e-01  -0.121 0.903622
## ...
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.215 on 71631 degrees of freedom
## Multiple R-squared:   0.19,  Adjusted R-squared:  0.1698
## F-statistic: 9.392 on 1789 and 71631 DF,  p-value: < 2.2e-16
```

```
anova(instr_eval2, instr_eval3)
```

```
## Analysis of Variance Table
##
## Model 1: y ~ service + dept + d
## Model 2: y ~ service + dept + d + service:d
##   Res.Df    RSS  Df Sum of Sq      F    Pr(>F)
## 1  72292 107938
## 2  71631 105724 661    2213.8 2.2692 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Mixed Effect Models

```
me_instr_eval1 <- lmer(y ~ service + dept + d + d:service + (1 | studage / s), data=InstEval)
```

```
## fixed-effect model matrix is rank deficient so dropping 479 columns / coefficients
```

```
summary(me_instr_eval1)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: y ~ service + dept + d + d:service + (1 | studage/s)
##    Data: InstEval
##
## REML criterion at convergence: 234042
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.97146 -0.73061  0.03439  0.75432  3.04260
##
## Random effects:
##  Groups    Name        Variance  Std.Dev.
##  s:studage (Intercept) 0.1042016 0.32280
##  studage   (Intercept) 0.0001142 0.01069
##  Residual              1.3710352 1.17091
## Number of obs: 73421, groups:  s:studage, 2972; studage, 4
##
## Fixed effects:
##                  Estimate Std. Error t value
## (Intercept)     3.8136279  0.3570932  10.680
## service1       -1.0872627  0.4648478  -2.339
## dept5          -0.3588668  0.4856483  -0.739
## dept10         -0.0461133  0.4083123  -0.113
## dept12         -1.6111457  0.5720680  -2.816
## dept6          -1.5423850  0.4262494  -3.619
## dept7          -1.2465021  0.4699869  -2.652
## dept4          -0.1863719  0.4700122  -0.397
## dept8           0.8015823  0.6132148   1.307
## dept9          -0.1356342  0.3700464  -0.367
## dept14         -0.0698464  0.4162224  -0.168
## dept1          -0.3740825  0.5188163  -0.721
## dept3          -0.6951642  0.4867751  -1.428
## dept11          0.5016004  0.9113838   0.550
## dept2           0.0640467  0.5739985   0.112
## d6              0.4779126  0.4954382   0.965
## d7              0.7148472  0.9808910   0.729
## d8              0.3535770  0.4761726   0.743
## d12            -1.2978312  0.5725438  -2.267
## d13            -0.1556240  0.2820550  -0.552
## ...
##
## Correlation matrix not shown by default, as p = 1790 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

```
## fit warnings:
## fixed-effect model matrix is rank deficient so dropping 479 columns / coefficients
```

```
anova(me_instr_eval1)
```

```
## Analysis of Variance Table
##           npar  Sum Sq Mean Sq  F value
## service      1   258.8 258.832 188.7859
## dept        13   855.8  65.828  48.0135
## d         1114 21049.8  18.896  13.7820
## service:d  661  2043.8   3.092   2.2552
```

```
me_instr_eval2 <- lmer(y ~ service + dept + d + d:service + (1 | s), data=InstEval)
```

```
## fixed-effect model matrix is rank deficient so dropping 479 columns / coefficients
```

```
summary(me_instr_eval2)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: y ~ service + dept + d + d:service + (1 | s)
##    Data: InstEval
##
## REML criterion at convergence: 234042.3
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.97357 -0.73059  0.03469  0.75437  3.04387
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  s        (Intercept) 0.1043   0.323
##  Residual             1.3710   1.171
## Number of obs: 73421, groups:  s, 2972
##
## Fixed effects:
##                  Estimate Std. Error t value
## (Intercept)     3.8118440  0.3570497  10.676
## service1       -1.0858679  0.4648452  -2.336
## dept5          -0.3597357  0.4856457  -0.741
## dept10         -0.0450186  0.4083066  -0.110
## dept12         -1.6098739  0.5720656  -2.814
## dept6          -1.5402580  0.4262409  -3.614
## dept7          -1.2435485  0.4699753  -2.646
## dept4          -0.1857524  0.4700105  -0.395
## dept8           0.8013665  0.6132142   1.307
## dept9          -0.1352919  0.3700461  -0.366
## dept14         -0.0690918  0.4162202  -0.166
## dept1          -0.3717585  0.5188115  -0.717
## dept3          -0.6924945  0.4867664  -1.423
## dept11          0.5028416  0.9113822   0.552
## dept2           0.0641471  0.5739978   0.112
## d6              0.4772810  0.4954355   0.963
## d7              0.7136243  0.9808891   0.728
## d8              0.3531276  0.4761650   0.742
## d12            -1.2964509  0.5725443  -2.264
## d13            -0.1551402  0.2820446  -0.550
## ...
```

```
##
```

```
## Correlation matrix not shown by default, as p = 1790 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```

```
## fit warnings:
## fixed-effect model matrix is rank deficient so dropping 479 columns / coefficients
```

```
anova(me_instr_eval2, me_instr_eval1)
```

```
## refitting model(s) with ML (instead of REML)
```

```
## Data: InstEval
## Models:
## me_instr_eval2: y ~ service + dept + d + d:service + (1 | s)
## me_instr_eval1: y ~ service + dept + d + d:service + (1 | studage/s)
##                npar    AIC    BIC  logLik deviance Chisq Df Pr(>Chisq)
## me_instr_eval2 1792 236224 252718 -116320   232640
## me_instr_eval1 1793 236226 252729 -116320   232640     0  1          1
```

In conclusion, nesting students in student age doesn't really help.

When you look at the effect size and polarity of d12 in each of the models above, it varies wildly. This bothered me at first: shouldn't at least the direction of the effect be consistent? But then I realized that the regressions aren't giving us inconsistent information, they are just saying, if this is your model of how things work, this is the effect we see.

For example, when we don't include department, d12 cannot be differentiated from the mean score. However, when we include department and service in the regression, she appears to have a highly significant, highly positive effect on her rating. However, when we include dept:service, she suddenly has a highly *negative* effect on ratings. That is to say that her mix of department and service schedule explains her high ratings.

By which measure should she be evaluated? Which one is the *truth*? The answer is that none of them are true, they are simply metrics that fall out of how you choose to explain lecture ratings.

Mixed Effect Model Data Simulation

Our previous models were a little too complicated for data simulation so we will train a simple mixed effect model:

```
simple_me_reg <- lmer(y ~ service + d + (1 | dept), data=InstEval)
summary(simple_me_reg)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: y ~ service + d + (1 | dept)
##    Data: InstEval
##
## REML criterion at convergence: 238259.8
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -2.80774 -0.75272  0.04866  0.79446  2.69938
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  dept     (Intercept) 0.01162  0.1078
##  Residual             1.49308  1.2219
## Number of obs: 73421, groups:  dept, 14
##
## Fixed effects:
##               Estimate Std. Error t value
```

```
## (Intercept)  3.727e+00  3.839e-01   9.710
## service1    -9.739e-02  1.367e-02  -7.123
## d6          -9.499e-01  4.551e-01  -2.087
## d7           4.307e-01  4.521e-01   0.953
## d8          -1.122e+00  4.293e-01  -2.615
## d12         -7.945e-02  4.570e-01  -0.174
## d13         -2.498e-02  4.441e-01  -0.056
## ...
##
## Correlation matrix not shown by default, as p = 1129 > 12.
## Use print(x, correlation=TRUE)  or
##     vcov(x)        if you need it
```