Multilevel Multilingual

Andrew Grogan-Kaylor

2023-11-10

Table of contents

List of Figures

List of Tables

1 Multilevel Models in Stata, R and Julia

1.1 Introduction

Below, I describe the use of Stata, R, and Julia to estimate multilevel models. Because this document is built by `Quarto`, I describe calling these programs from within a `Quarto` environment. However, each piece of software could be used individually and separately.

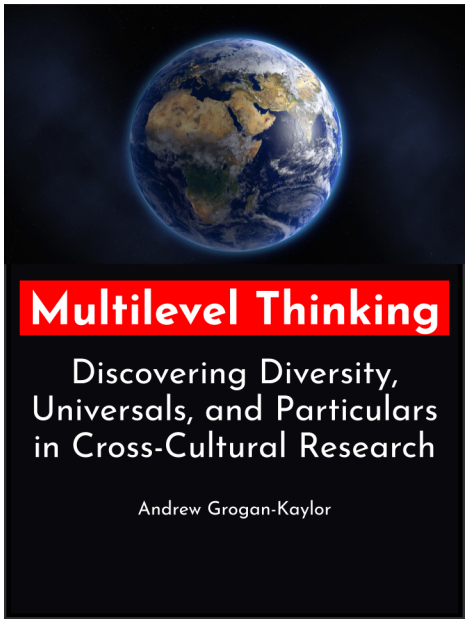


Figure 1.1: Book Cover For Multilevel Thinking

1.2 The Data

The examples below use the `simulated_multilevel_data.dta` file from *Multilevel Thinking*. Here is a direct link to download the data.

Table 1.1: Sample of Simulated Multilevel Data

country	HDI	family	id	group	physical_punishment	warmth	outcome
1	69	1	1.1	2	2	3	59.18
1	69	2	1.2	2	4	0	61.54
1	69	3	1.3	1	4	4	51.87
1	69	4	1.4	2	0	6	51.71
1	69	5	1.5	2	3	2	55.88
1	69	6	1.6	1	5	3	60.78

1.3 The Equation

$$\text{outcome}_{ij} = \beta_0 + \beta_1 \text{warmth}_{ij} + \beta_2 \text{physical punishment}_{ij} + \beta_3 \text{group}_{ij} + \beta_4 \text{HDI}_{ij} + u_{0j} + u_{1j} \times \text{warmth}_{ij} + e_{ij} \tag{1.1}$$

1.4 Setup

1.4.1 Stata

I need to use the library `Statamarkdown` to call Stata, or I could run Stata on its own

```
library(Statamarkdown)
```

1.4.2 R

In R, I use the library `lme4` to run multilevel models.

```
library(lme4)
```

1.4.3 Julia

I need to call Julia from R.

```
library(JuliaCall)

julia_setup(JULIA_HOME = "/Applications/Julia-1.8.app/Contents/Resources/julia/bin")
```

1.5 Get Data & Run Models

To explain statistical syntax for each software, I consider the more general case of a multilevel model with dependent variable `y`, independent variables `x` and `z`, clustering variable `group`, and a random slope for `x`. i is the index for the person, while j is the index for the `group`.

$$y = \beta_0 + \beta_1 x_{ij} + \beta_2 z_{ij} + u_{0j} + u_{1j} \times x_{ij} + e_{ij} \qquad (1.2)$$

1.5.1 Stata

In Stata `mixed`, the syntax for a multilevel model of the form described in Equation 1.2 is:

```
mixed y x || group: x
```

1.5.1.1 Get The Data

> **Tip For Running Stata From Quarto**
>
> Because I am running Stata from inside a Quarto document, and running Stata in multiple chunks, I need to use the `collectcode=TRUE` option in the first Stata chunk. i.e. my Quarto chunk needs to begin with "`{stata, collectcode=TRUE}`
> See Doug Hemken's excellent documentation on `Statamarkdown` here.

```
use simulated_multilevel_data.dta
```

1.5.1.2 Graph

```
twoway scatter outcome warmth, xtitle("warmth") ytitle("outcome") title("Outcome by Parental

quietly graph export scatter.png, replace
```

```
> arental Warmth")
```

1.5.1.3 Run The Model

```
mixed outcome warmth physical_punishment group HDI || country: warmth
```

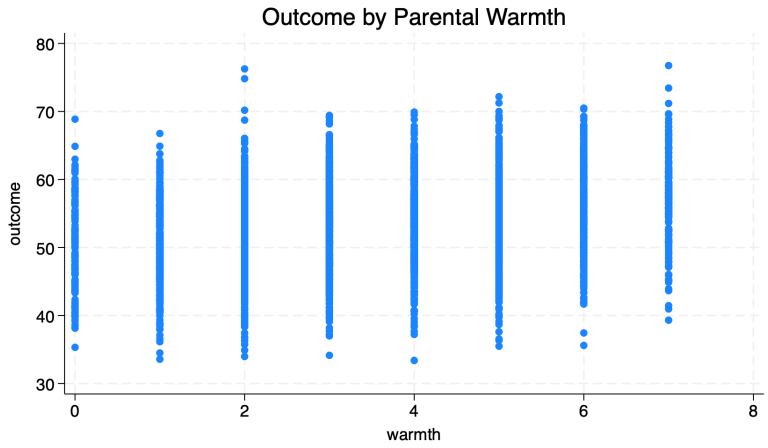


Figure 1.2: Outcome by Parental Warmth

```
Performing EM optimization ...

Performing gradient-based optimization:
Iteration 0:  Log likelihood =  -9668.198
Iteration 1:  Log likelihood = -9667.9551
Iteration 2:  Log likelihood = -9667.9534
Iteration 3:  Log likelihood = -9667.9533
Iteration 4:  Log likelihood = -9667.9532

Computing standard errors ...

Mixed-effects ML regression                     Number of obs     =  3,000
Group variable: country                         Number of groups  =     30
                                                Obs per group:
                                                              min =    100
                                                              avg =  100.0
                                                              max =    100
                                                Wald chi2(4)      = 401.26
Log likelihood = -9667.9532                     Prob > chi2       = 0.0000

------------------------------------------------------------------------------
             outcome | Coefficient  Std. err.      z    P>|z|     [95% conf. interval]
---------------------+--------------------------------------------------------------
              warmth |   .9616447   .0581825    16.53   0.000     .8476091     1.07568
```

```
physical_punishment |  -.8453802   .0798155   -10.59   0.000    -1.001816   -.6889448
              group |   1.084344   .2200539     4.93   0.000     .6530461    1.515642
                HDI |    .010557   .0204522     0.52   0.606    -.0295286    .0506426
              _cons |   49.87963   1.436612    34.72   0.000     47.06392    52.69534
------------------------------------------------------------------------------

------------------------------------------------------------------------------
  Random-effects parameters  |   Estimate   Std. err.      [95% conf. interval]
-----------------------------+------------------------------------------------
country: Independent         |
                  var(warmth) |   1.83e-06   .0000173      1.76e-14    190.9774
                   var(_cons) |   3.370262   .9633726      1.924651    5.901676
-----------------------------+------------------------------------------------
               var(Residual) |   36.01906   .9346936      34.23291    37.89842
------------------------------------------------------------------------------
LR test vs. linear model: chi2(2) = 198.01                Prob > chi2 = 0.0000

Note: LR test is conservative and provided only for reference.
```

1.5.2 R

In R `lme4`, the general syntax for a multilevel model of the form described in Equation 1.2 is:

```
lmer(y ~ x + z + (1 + x || group), data = ...)
```

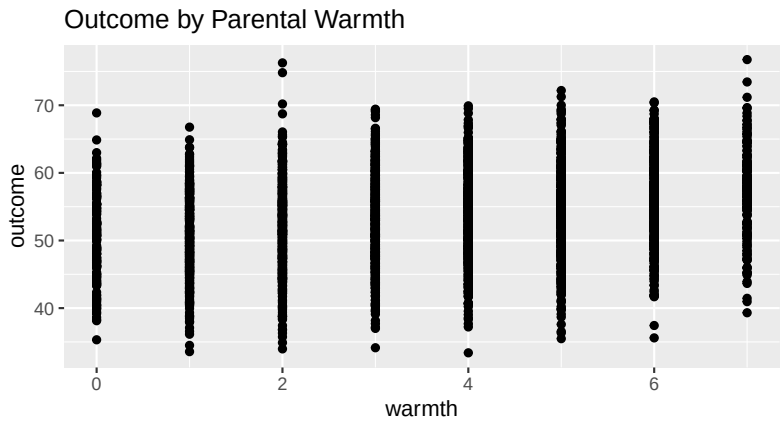
1.5.2.1 Get The Data

```r
library(haven)

df <- read_dta("simulated_multilevel_data.dta")
```

1.5.2.2 Graph

```r
library(ggplot2)

ggplot(df,
```

```
      aes(x = warmth,
          y = outcome)) +
  geom_point() +
  labs(title = "Outcome by Parental Warmth")
```



1.5.2.3 Run The Model

```
fit1 <- lmer(outcome ~ warmth + physical_punishment +
               group + HDI +
               (1 + warmth || country),
             data = df)

summary(fit1)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: outcome ~ warmth + physical_punishment + group + HDI + ((1 |
    country) + (0 + warmth | country))
   Data: df

REML criterion at convergence: 19350.3

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.4496 -0.6807  0.0016  0.6864  3.1792
```

```
Random effects:
 Groups    Name        Variance  Std.Dev.
 country   (Intercept)  3.611568 1.90041
 country.1 warmth       0.001876 0.04331
 Residual              36.049124 6.00409
Number of obs: 3000, groups:  country, 30

Fixed effects:
                    Estimate Std. Error t value
(Intercept)         49.88754    1.48203  33.662
warmth               0.96155    0.05875  16.367
physical_punishment -0.84556    0.07986 -10.588
group                1.08471    0.22017   4.927
HDI                  0.01044    0.02116   0.493

Correlation of Fixed Effects:
            (Intr) warmth physc_ group
warmth      -0.126
physcl_pnsh -0.135 -0.025
group       -0.218 -0.010 -0.019
HDI         -0.925 -0.006  0.008 -0.001
```

1.5.3 Julia

In Julia `MixedModels`, the general syntax for a multilevel model of the form described in Equation 1.2 is:

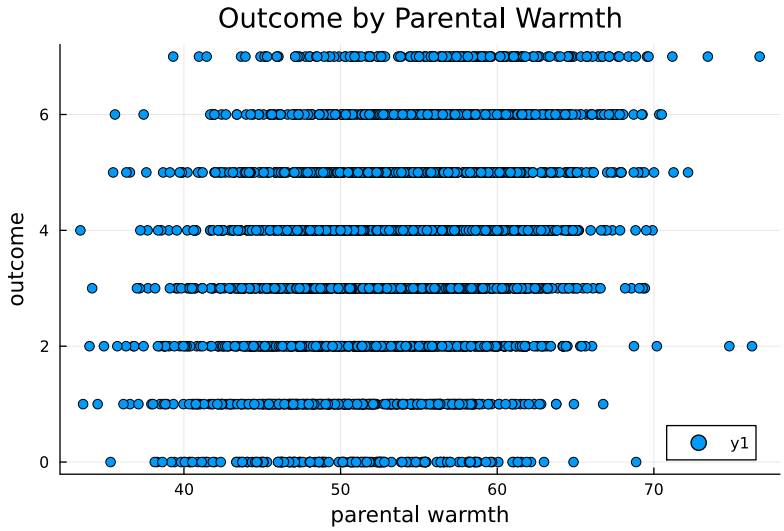
```
fit(MixedModel, @formula(y ~ x + z + (1 + x |
group)), data)
```

1.5.3.1 Load The Needed Packages And Load The Data

```
using Tables, MixedModels, StatFiles, DataFrames, CategoricalArrays, DataFramesMeta

df = DataFrame(load("simulated_multilevel_data.dta"))
```

1.5.3.2 Graph

```
using StatsPlots

@df df scatter(:outcome, :warmth,
               title = "Outcome by Parental Warmth",
               ylabel = "outcome",
               xlabel = "parental warmth")
```



1.5.3.3 Change Country To Categorical

```
@transform!(df, :country = categorical(:country))
```

1.5.3.4 Run The Model

```
m1 = fit(MixedModel, @formula(outcome ~ warmth + physical_punishment +
               group + HDI +
               (1 + warmth | country)), df)
```

```
Linear mixed model fit by maximum likelihood
```

```
 outcome ~ 1 + warmth + physical_punishment + group + HDI + (1 + warmth | country)
   logLik   -2 logLik     AIC       AICc        BIC
 -9667.9392 19335.8783 19353.8783 19353.9385 19407.9357

Variance components:
            Column    Variance   Std.Dev.   Corr.
country  (Intercept)   3.2369484 1.7991521
         warmth        0.0001080 0.0103903 +1.00
Residual               36.0187144 6.0015593
 Number of obs: 3000; levels of grouping factors: 30

  Fixed-effects parameters:

                          Coef.  Std. Error       z  Pr(>|z|)

(Intercept)           49.9018      1.43435     34.79    <1e-99
warmth                 0.961545    0.0582135   16.52    <1e-60
physical_punishment   -0.845389    0.0798149  -10.59    <1e-25
group                  1.08524     0.220055     4.93    <1e-06
HDI                    0.0101984   0.0204401    0.50    0.6178
```

2 Cross-Classified Models in Stata, R and Julia

2.1 Introduction

A two level multilevel model imagines that *Level 1* units are nested in *Level 2* units. A three level multilevel model imagines that *Level 1* units are nested in *Level 2* units, which are in turn nested in *Level 3*.

A cross-classified model imagines that the nesting is not hierarchical, but rather that there are two sets of clusters or nestings in which individuals may be nested.

In this data, *events* are nested inside *persons* which are in turn nested in *countries*, since in this data, individuals never change countries. However, the use of a cross-classified framework would allow for a situation in which *persons* moved from country to country, and experienced different *events* in different *countries*.

Below, I describe the use of Stata, R, and Julia to estimate cross-classified models. Because this document is built by `Quarto`, I describe calling these programs from within a `Quarto` environment. However, each piece of software could be used individually and separately.

2.2 The Data

The examples below use the `simulated_multilevel_longitudinal` file from *Multilevel Thinking*. Here is a direct link to download the data.

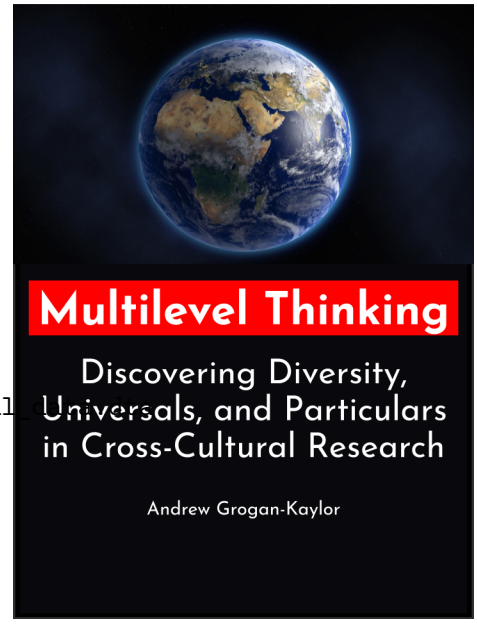


Figure 2.1: Book Cover For Multilevel Thinking

Table 2.1: Sample of Simulated Multilevel Longitudinal Data

country	HDI	family	id	group	t	physical_punishment	warmth	outcome
1	69	1	1.1	2	1	2	3	59.18
1	69	1	1.1	2	2	2	2	58.29
1	69	1	1.1	2	3	3	3	60.58
1	69	2	1.2	2	1	4	0	61.54
1	69	2	1.2	2	2	4	0	55.96
1	69	2	1.2	2	3	4	2	56.19

2.3 The Equation

$$\text{outcome}_{ijt} = \beta_0 + \beta_1 t_{ijt} + \beta_2 \text{warmth}_{ijt} + \beta_3 \text{physical punishment}_{ijt} + \beta_4 \text{group}_{ijt} + \beta_5 \text{HDI}_{ijt} + u_{0j} + v_{0i} + e_{ijt} \tag{2.1}$$

2.4 Setup

2.4.1 Stata

I need to use the library `Statamarkdown` to call Stata, or I could run Stata on its own

```
library(Statamarkdown)
```

2.4.2 R

In R, I use the library `lme4` to run multilevel models.

```
library(lme4)
```

2.4.3 Julia

I need to call Julia from R.

```
library(JuliaCall)

julia_setup(JULIA_HOME = "/Applications/Julia-1.8.app/Contents/Resources/julia/bin")
```

2.5 Get Data & Run Models

To explain statistical syntax for each software, I consider the more general case of a cross-classified model with dependent variable `y`, independent variables `x` and `z`, clustering variables `country` and `id`.

$$y = \beta_0 + \beta_1 x_{ijt} + \beta_2 z_{ijt} + u_{0j} + v_{0i} + e_{ijt} \tag{2.2}$$

2.5.1 Stata

In Stata `mixed`, the syntax for a multilevel model of the form described in Equation 2.2 is:

```
mixed y x || _all: R.group1 || group2:
```

2.5.1.1 Get The Data

> **Tip For Running Stata From Quarto**
>
> Because I am running Stata from inside a Quarto document, and running Stata in multiple chunks, I need to use the `collectcode=TRUE` option in the first Stata chunk. i.e. my Quarto chunk needs to begin with "'`{stata, collectcode=TRUE}`
> See Doug Hemken's excellent documentation on `Statamarkdown` here.

```
use simulated_multilevel_longitudinal_data.dta
```

2.5.1.2 Run The Model

```
mixed outcome t warmth physical_punishment group HDI || _all: R.country || id:
```

```
variable t not found
r(111);

end of do-file
r(111);
```

2.5.2 R

In R `lme4`, the general syntax for a multilevel model of the form described in Equation 2.2 is:

```
lmer(y ~ x + z + (1 | group1) + (1 | group2), data
= ...)
```

2.5.2.1 Get The Data

```
library(haven)

df <- read_dta("simulated_multilevel_longitudinal_data.dta")
```

2.5.2.2 Run The Model

```
fit1 <- lmer(outcome ~ t + warmth + physical_punishment +
               group + HDI +
               (1 | id) +
               (1 | country),
             data = df)

summary(fit1)
```

```
Linear mixed model fit by REML ['lmerMod']
Formula: outcome ~ t + warmth + physical_punishment + group + HDI + (1 |
    id) + (1 | country)
   Data: df

REML criterion at convergence: 57088.4

Scaled residuals:
    Min      1Q  Median      3Q     Max
-3.4471 -0.6226  0.0081  0.6153  3.1993

Random effects:
 Groups   Name        Variance Std.Dev.
 id       (Intercept)  8.864   2.977
 country  (Intercept)  3.924   1.981
 Residual             26.008   5.100
Number of obs: 9000, groups:  id, 3000; country, 30

Fixed effects:
                     Estimate Std. Error t value
(Intercept)         49.494782   1.471780  33.629
t                    0.987964   0.065840  15.005
warmth               0.946259   0.038200  24.771
physical_punishment -0.926880   0.049970 -18.549
group                0.985786   0.153550   6.420
HDI                  0.007543   0.021437   0.352

Correlation of Fixed Effects:
            (Intr) t      warmth physc_ group
t           -0.090
warmth      -0.085  0.008
physcl_pnsh -0.085  0.003 -0.019
group       -0.154  0.000 -0.013 -0.008
HDI         -0.943  0.000 -0.003  0.003  0.000
```

2.5.3 Julia

In Julia `MixedModels`, the general syntax for a multilevel model of the form described in Equation 2.2 is:

```
fit(MixedModel, @formula(y ~ x + z + (1 | group1) +
(1 | group2)), data)
```

2.5.3.1 Load The Needed Packages And Load The Data

```
using Tables, MixedModels, StatFiles, DataFrames, CategoricalArrays, DataFramesMeta

df = DataFrame(load("simulated_multilevel_longitudinal_data.dta"))
```

2.5.3.2 Change Country To Categorical

```
@transform!(df, :country = categorical(:country))
```

2.5.3.3 Run The Model

```
m1 = fit(MixedModel, @formula(outcome ~ t + warmth + physical_punishment +
                                group + HDI +
                                (1 | id) +
                                (1 | country)), df)
```

```
Linear mixed model fit by maximum likelihood
 outcome ~ 1 + t + warmth + physical_punishment + group + HDI + (1 | id) + (1 | country)
    logLik   -2 logLik      AIC         AICc        BIC
 -28533.9968  57067.9935  57085.9935  57086.0136  57149.9384

Variance components:
            Column   Variance Std.Dev.
id       (Intercept)   8.85264 2.97534
country  (Intercept)   3.65030 1.91058
Residual              26.00093 5.09911
 Number of obs: 9000; levels of grouping factors: 3000, 30

  Fixed-effects parameters:

                          Coef.  Std. Error       z  Pr(>|z|)
```

(Intercept)	49.4945	1.42422	34.75	<1e-99
t	0.987965	0.0658315	15.01	<1e-50
warmth	0.946255	0.0381869	24.78	<1e-99
physical_punishment	-0.926774	0.0499549	-18.55	<1e-76
group	0.985819	0.153487	6.42	<1e-09
HDI	0.00754357	0.0207101	0.36	0.7157