Multilevel linear models in Stata: a simulation approach

Isabel Cañette

Senior Statistician

StataCorp LP

2012 Spanish Stata Users Group meeting
Barcelona, September 12, 2012

Simulating data for our models

Simulating data is a powerful tool to understand the model we want to fit, and also to spot identification issues.
Let's start by fitting a linear model on the homework dataset[1]

```
use homework
regress math homework
```

The same coefficients can be obtained by using `xtmixed`

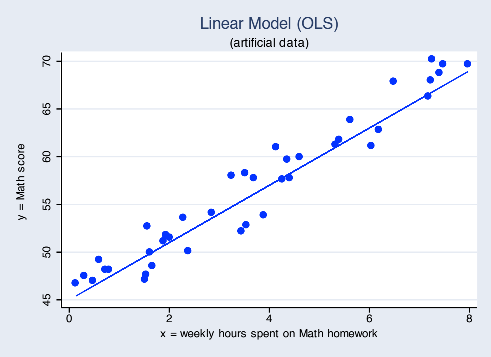
```
. xtmixed math homework, nolog noheader

------------------------------------------------------------------------------
        math |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
    homework |   3.126375   .2860801    10.93   0.000     2.565668    3.687081
       _cons |   45.56015   .7055719    64.57   0.000     44.17726    46.94305
------------------------------------------------------------------------------

------------------------------------------------------------------------------
  Random-effects Parameters  |   Estimate   Std. Err.     [95% Conf. Interval]
-----------------------------+------------------------------------------------
                sd(Residual) |   9.661575   .2998812       9.09134    10.26758
------------------------------------------------------------------------------
```

[1] Kreft, I.G.G and de J. Leeuw. 1998. Introducing Multilevel Modeling. Sage. Rabe-Hesketh, S. and A. Skrondal. 2008. Multilevel and Longitudinal Modeling Using Stata, Second Edition. Stata Press

Simulating data for this model is very simple

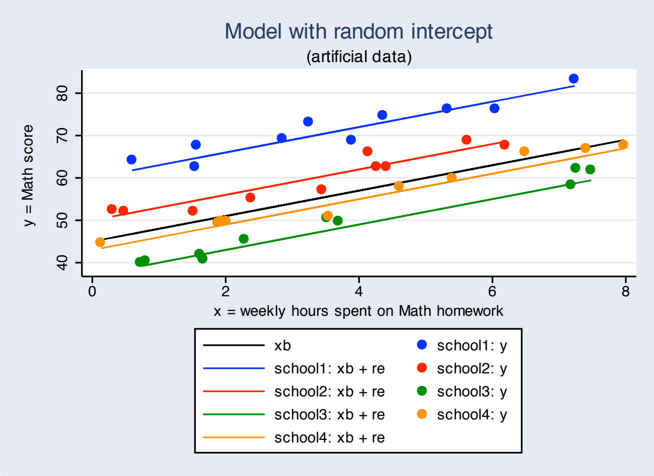


```
. gen x = 8*runiform()
. gen y1 = 3.13*x + 45.56 + 9.66*rnormal()
```

(Notice that I should use the saved results instead of copying them from the screen; I'm just doing this for didactic purposes)

Random-effect models

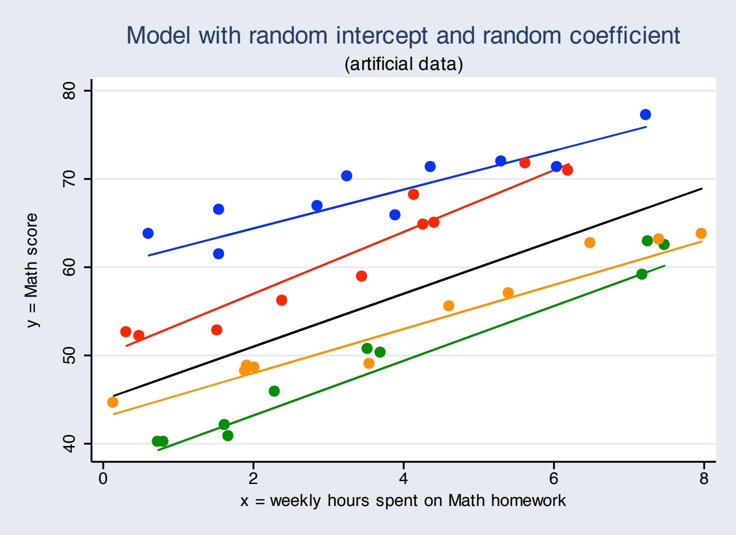
Random intercept only: we are assuming that the intercept varies randomly across schools



The syntax to fit this model would be:

```
xtmixed math homework || schid:
```

Random intercept and random slope: we are assuming that both, intercept and slope, vary randomly across schools)



```
xtmixed math homework || schid: homework
```

```
. xtmixed math homework || schid: homework, nolog  noheader nolrtest

------------------------------------------------------------------------------
        math |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
    homework |   1.974516   .8314652     2.37   0.018     .3448746    3.604158
       _cons |   46.46441   1.608962    28.88   0.000      43.3109    49.61792
------------------------------------------------------------------------------

------------------------------------------------------------------------------
  Random-effects Parameters  |   Estimate   Std. Err.     [95% Conf. Interval]
-----------------------------+------------------------------------------------
schid: Independent           |
               sd(homework)  |   3.709275   .6847578       2.58316    5.326314
                  sd(_cons)  |    7.12292   1.255007      5.042925    10.06082
-----------------------------+------------------------------------------------
               sd(Residual)  |    7.34461   .2419451       6.88539    7.834457
------------------------------------------------------------------------------

. est store original1
```

Simulating data for one-level random-effects models

math	coef
homework	1.974516
_cons	46.46441
schid	Estimate
sd(homework)	3.709275
sd(_cons)	7.12292
sd(Residual)	7.34461

```
set seed 1357
set sortseed 159
set obs 100 // 100 schools
generate schid = _n  // school identifier
generate nu0 = 7.12*rnormal() // random intercept per school
generate nu1 = 3.709*rnormal() // random slope per school
expand 200  // 200 students per school
generate stud_id = _n // student identifier
generate homework = 8*runiform() // indep. variable
generate residual = 7.34*rnormal() // residuals
generate  math = 1.97*homework + 46.46 + nu0 + nu1*homework + residual
xtmixed math homework || schid: homework, nolog noheader nolrtest
est store simulated1
```

```
. estimates table  original1 simulated1

-----------------------------------------
    Variable | original1    simulated1
-------------+---------------------------
math         |
    homework |  1.9745165    1.8530287
       _cons |  46.464411    46.569009
-------------+---------------------------
lns1_1_1     |
       _cons |  1.3108365    1.3818598
-------------+---------------------------
lns1_1_2     |
       _cons |  1.9633177    1.8942815
-------------+---------------------------
lnsig_e      |
       _cons |  1.9939667    1.9986072
-----------------------------------------
```

We have assumed that the slope and the intercept are independent. We could have assumed that there was a correlation among them.

```
. xtmixed math homew || schid: homew, cov(unstructured) var nolo nolr nohead

------------------------------------------------------------------------------
        math |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
    homework |   1.980164   .9284486     2.13   0.033      .160438    3.799889
       _cons |   46.32561   1.758934    26.34   0.000     42.87816    49.77305
------------------------------------------------------------------------------

------------------------------------------------------------------------------
  Random-effects Parameters  |   Estimate   Std. Err.     [95% Conf. Interval]
-----------------------------+------------------------------------------------
schid: Unstructured          |
               var(homework) |   17.72652   6.260285      8.871839    35.41875
                  var(_cons) |   62.42455   21.38154      31.90093    122.1539
         cov(homework,_cons) |  -27.59391   10.56626      -48.3034   -6.884412
-----------------------------+------------------------------------------------
               var(Residual) |   53.29462   3.465962      46.91658    60.53972
------------------------------------------------------------------------------

. est store original2
```

Simulating data for one-level models with correlated random effects

math	coef
homework	1.980164
_cons	46.32561
schid	Estimate
var(homework)	17.72652
var(_cons)	62.42455
cov(homework,_cons)	-27.59391
var(Residual)	53.29462

```
clear
set seed 1357
set sortseed 159
set obs 100 // 100 schools
generate schid = _n  // school identifier
matrix a = (17.73, -27.59 \ -27.59, 62.42)
drawnorm nu1 nu0, cov(a) // random slope and intercept
expand 200  // 200 students per school
generate stud_id = _n // student identifier
generate homework = 8*runiform() // indep. variable
generate residual = sqrt(53.29)*rnormal() // residuals
generate  math = 1.98*homework + 46.33 + nu0 + nu1*homework + residual
xtmixed math homework || schid: homework,  ///
         cov(unstructured) var nolog noheader nolrtest
est store original2
```

```
. xtmixed math homework || schid: homework, cov(unstructured) var
(output omitted)
. est store simulated2

. est table original2 simulated2

----------------------------------------
    Variable | original2    simulated2
-------------+--------------------------
math         |
    homework |  1.9801637    2.1013484
       _cons |  46.325606    45.970628
-------------+--------------------------
lns1_1_1     |
       _cons |  1.4375308    1.4200276
-------------+--------------------------
lns1_1_2     |
       _cons |  2.0669793    2.0222833
-------------+--------------------------
atr1_1_1_2   |
       _cons | -1.1865765   -1.1093948
-------------+--------------------------
lnsig_e      |
       _cons |  1.9879177    1.9931474
----------------------------------------
```

Multilevel nested models

Often, researchers tend to model the "natural" nesting structure. For example, schools are naturally nested within regions, because a school can't be in two regions.
xtmixed assumes, by default, that consecutive levels are nested.

```
. xtmixed math homework || region: ||schid:
```

This specification assumes that I have a random intercept for each region, and also one random intercept for each school.

Meaning of "nested"

xtmixed assumed that schools on different regions are different, no matter if we repeat the identificators across regions. If we code:

region	schid
1	1
1	2
1	3
2	1
2	2
2	3

xtmixed will interpret that (the effect of) school 1 from region 1 and (the effect of) school 1 from region 2 are different.

Simulating data for nested random-effects models

```
set seed 1357
set sortseed 713
scalar sd_int_region = 5
scalar sd_int_school = 7
scalar sd_res = 1
qui set obs 20  // number of region
gen region = _n // region identifier
gen int_region = sd_int_region*rnormal()
expand 100 // number of schools per region
sort region
gen schoolid = _n // school identifier
gen int_school = sd_int_school*rnormal()
qui expand 100 // number of students per school
gen res = rnormal() // residuals
gen homework = 8*runiform() // indep. variable
gen y = 2*homework +46 + int_region + int_school + res
```

```
. xtmixed y homework || region: ||school:, nolog nolr nohead

------------------------------------------------------------------------------
           y |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
    homework |   2.000976   .0009745  2053.38   0.000     1.999067    2.002886
       _cons |   46.19403   .8541039    54.08   0.000     44.52002    47.86805
------------------------------------------------------------------------------

------------------------------------------------------------------------------
  Random-effects Parameters  |   Estimate   Std. Err.     [95% Conf. Interval]
-----------------------------+------------------------------------------------
region: Identity             |
                   sd(_cons) |   3.753788   .6304813      2.700866    5.217188
-----------------------------+------------------------------------------------
schoolid: Identity           |
                   sd(_cons) |   7.060727   .1122247      6.844161    7.284145
-----------------------------+------------------------------------------------
                sd(Residual) |    .998948   .0015874      .9958415    1.002064
------------------------------------------------------------------------------
```

Crossed effects

Sometimes we don't want to consider nested-effect models, but crossed-effect models, i.e., models where levels that are not nested. For example, in the pig dataset, we have the dependent variable weight and information on the week and the id.
We may think that each individual pig has some random departure from the line:

```
xtmixed weight week ||id:
```

or instead, that each week determines some departure from this line:

```
xtmixed weigh week || week:
```

What if we want both? We don't want to consider these effects as "nested" How do we simulate data for this model?

Simulating data for crossed-effects models

```
set seed 1357
set sortseed 793
scalar sd_re_week = 1
scalar sd_re_id = 3.5
scalar sd_res = 2
set obs 50 //number of pigs
gen id = _n // pig identifier
gen re_id = sd_re_id*rnormal() // random intercept, pig level
expand 20 // number of weeks
bysort id: gen week = _n // week identifier; these repeat across pigs
gen re_week = sd_re_week*rnormal() // random effect, week
bysort week: replace re_week = re_week[1] // needs to be unique per week
gen res = sd_res*rnormal()
gen weight = 6*week + 19 + re_id + re_week  + res
```

We can estimate the model with the following syntax:

```
. xtmixed weigh week || _all:R.week || id:, nolog nolr nohead

------------------------------------------------------------------------------
      weight |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
        week |   6.003322   .0415515   144.48   0.000     5.921882    6.084761
       _cons |   19.41274   .6880104    28.22   0.000     18.06426    20.76121
------------------------------------------------------------------------------

------------------------------------------------------------------------------
  Random-effects Parameters  |   Estimate   Std. Err.     [95% Conf. Interval]
-----------------------------+------------------------------------------------
_all: Identity               |
                 sd(R.week)  |   1.033334   .1851922      .7272604    1.468221
-----------------------------+------------------------------------------------
id: Identity                 |
                  sd(_cons)  |   3.358588   .3453138      2.745619    4.108404
-----------------------------+------------------------------------------------
               sd(Residual)  |   2.004485   .0464529      1.915476    2.097631
------------------------------------------------------------------------------
```

Stata tip: always use the R. notation for the level with less categories.

What does exactly, the _all:R.var notation do?

It creates a level "_all" containing all the observations in one category; At this level, a set of covariates is included, consisting of dummies for the categories of var, while constraining the variances to be the same.
That is:

```
xtmixed weight week || _all:R.week
```

Is the same as

```
generate one = 1
tab id, gen(week_dummy)
xtmixed weight week || one: week_dummy*, cov(identity) nocons
```

Which is just an inefficient way to fit the model:

```
xtmixed weight week || week:
```

Naturally-nested vs model-nested models

Let's assume that we have data on return on assets for a set of firms, which belong to different industries and different countries. Industries and countries are naturally crossed. We can model them as they are:

```
. xtmixed asset || _all: R.country ||industry:
```

We might think, instead, that each industry behaves differently for each country, i.e., we can create a "virtual" level, country-industry.

```
. use asset2, clear
. xtmixed asset || country: || industry:
```

Application: Dyadic data analysis

This area is becoming increasingly popular among social scientists, and consists of statistical techniques to study data comprised by pairs of correlated individuals.

Some examples are member of a couple, parent and child, individuals matched in an experimental design, etc.

The main tools used for these problems are multilevel models and structural equation models (implemented in the `sem` command).

Kenny et al.[2] used an hypothetical dyadic study predicting likelihood of marriage.

The variables of interest were

- the dependent variable: likelihood of marriage within 5 years, as perceived by each member of the couple.
- the main predictor: a composite score measuring the contribution made by each member to the household
- two more covariates:
  - gender (women = -1, men = 1)
  - culture (Asian = -1, American = 1).

Notice the particular coding used for binary variables. It is done to interpret the coefficients as differences from the grand mean.

[2] Kenny, D., D. Kashi and W. Cook. 2006. Dyadic Data Analysis. The Guilford Press

The dataset looks like this:

```
     +-------------------------------------------------------------+
     | dyad   person   future   contribution   gender   culture |
     |-------------------------------------------------------------|
  1. |    1        1       75            -10       -1         1 |
  2. |    1        2       90             -5        1         1 |
     |-------------------------------------------------------------|
  3. |    2        1       55              0       -1         1 |
  4. |    2        2       75             10        1         1 |
     |-------------------------------------------------------------|
  5. |    3        1       45            -10       -1         1 |
  6. |    3        2       33            -15        1         1 |
     |-------------------------------------------------------------|
  7. |    4        1       70              5       -1         1 |
  8. |    4        2       75             15        1         1 |
     |-------------------------------------------------------------|
  9. |    5        1       50              0       -1         1 |
 10. |    5        2       40             -5        1         1 |
   (...)
```

The authors fit the following model:

```
 gen contrib_cult = contribution*culture

. xtmixed future contribution culture contrib_cult  || dyad: , ///
>      reml var nolog noheader

------------------------------------------------------------------------------
      future |      Coef.   Std. Err.      z    P>|z|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
contribution |   .8447885    .255653     3.30   0.001     .3437178    1.345859
     culture |  -9.032817   4.469649    -2.02   0.043    -17.79317    -.272466
contrib_cult |   .4872612    .255653     1.91   0.057    -.0138095     .988332
       _cons |   71.83089   4.469649    16.07   0.000     63.07054    80.59124
------------------------------------------------------------------------------

------------------------------------------------------------------------------
  Random-effects Parameters  |   Estimate   Std. Err.     [95% Conf. Interval]
-----------------------------+------------------------------------------------
dyad: Identity               |
                  var(_cons) |   176.4304   102.5106      56.49412    550.9898
-----------------------------+------------------------------------------------
               var(Residual) |   43.75333   21.43787      16.74738    114.3077
------------------------------------------------------------------------------
LR test vs. linear regression: chibar2(01) =     8.44 Prob >= chibar2 = 0.0018
```

The postestimation command `estat icc` computes the intraclass correlation, which in this case is the proportion of the total variance due to variation between couples.

```
. estat icc

Residual intraclass correlation

------------------------------------------------------------------------------
                       Level |        ICC   Std. Err.     [95% Conf. Interval]
-----------------------------+------------------------------------------------
                        dyad |   .8012872   .1274283      .4565605    .9508703
------------------------------------------------------------------------------
```

Categorical variables in this dataset are manually coded to obtain the difference from the grand mean. This kind of coding can be tricky when there are more than two categories, or with unbalanced data.

In Stata you don't need to do that; you can use the factor variable notation, and there is a battery of post estimation commands that will compute all the effects that you need (see `contrast`, `margins`, `marginsplot`).

Here is how to fit the previous model using the factor variable notation.

```
. gen cult2 = cult == 1
. xtmixed future  i.cult2##c.contrib  || dyad: , reml var nolog noheader nolr
------------------------------------------------------------------------------
              future |      Coef.   Std. Err.      z    P>|z|     [95% Conf. In
---------------------+--------------------------------------------------------
             1.cult2 |  -18.06563   8.939298    -2.02   0.043    -35.58634    -
        contribution |   .3575272   .2322398     1.54   0.124    -.0976544    .
                     |
cult2#c.contribution |
                  1  |   .9745225   .5113061     1.91   0.057     -.027619    1
                     |
               _cons |   80.86371   6.307358    12.82   0.000     68.50151
------------------------------------------------------------------------------


------------------------------------------------------------------------------
  Random-effects Parameters  |   Estimate   Std. Err.     [95% Conf. Interval]
-----------------------------+------------------------------------------------
dyad: Identity               |
                  var(_cons) |   176.4304   102.5106      56.49412    550.9898
-----------------------------+------------------------------------------------
               var(Residual) |   43.75333   21.43787      16.74738    114.3077
------------------------------------------------------------------------------
```

The differences from the grand mean can be computed and tested with contrast.

```
. contrast g.cult2 g.cult2#c.contrib

Contrasts of marginal linear predictions

Margins      : asbalanced

(output omitted)
----------------------------------------------------------------------
                     |   Contrast   Std. Err.     [95% Conf. Interval]
---------------------+------------------------------------------------
future               |
               cult2 |
        (0 vs mean)  |   9.032817   4.469649       .272466    17.79317
        (1 vs mean)  |  -9.032817   4.469649     -17.79317    -.272466
                     |
cult2#c.contribution |
        (0 vs mean)  |  -.4872612    .255653     -.988332     .0138095
        (1 vs mean)  |   .4872612    .255653    -.0138095      .988332
----------------------------------------------------------------------
```

Final remarks

- `xtmixed` is a versatile command that allows us to fit a variety of models.
- Understanding the mechanics of each piece in the syntax allows us to fit very sophisticated models.
- Simulating data allows us to get a deeper insight on multilevel models, to understand the particular specification we want to use, and eventually spot identification problems.
- `xtmixed` also allows us to specify different structures for the errors, feature not covered in this talk. This feature opens a new array of models, including more sophisticated models with multivariate response.
- The `sem` command can also be used to fit multilevel models. The choice of the command will depend on convenience (data setting) and on the particular model.