Applied Multilevel Regression Modeling

Day 5: Three-level Models & Recap

Constantin Manuel Bosancianu

WZB Berlin Social Science Center
Institutions and Political Inequality
bosancianu@icloud.com

August 2, 2019

Yesterday

Unlike OLS-based regression, in MLM we have multiple indicators of model fit: *logLik*, deviance, AIC, BIC. They are mostly used for relative comparisons of models.

A statistical test of improvement in the goodness-of-fit exists: the LRT. It is restricted to nested models, though.

Multilevel models have to abide by some of the same assumptions as OLS-based regression; other assumptions, though, are unique to the MLM setting.

Today

Three-level models: notation and differences from the two-level setup.

Practical advice: data management in MLM.

Applied task, based on data on tobacco lobby contributions to politicians in the US: writing up 3–4 models, estimating them, assessing model fit and choosing best-fitting model, and presenting quantities of interest from it.

3-level models

3-level models

The underlying mechanics stay exactly the same as in the 2-level instance.

Things get complicated with sample size, and with notation.

In terms of the first, the same guidelines about minimum sample size apply $\Rightarrow$ you'll need at least 30 units at the third level.[1]

[1] If each of these needs at least 1–3 observations, then the level 2 minimum size is 60–90.

3-level models

Thing get difficult very quickly with 3 levels.

$$
\begin{cases}
EFF_{ijk} = \beta_{0jk} + \beta_{1jk} * EDU_{ijk} + e_{ijk} \\
\\
\beta_{0jk} = \gamma_{00k} + \gamma_{01k} * STS_{jk} + v_{0jk} \\
\beta_{1jk} = \gamma_{10k} + \gamma_{11k} * STS_{jk} + v_{1jk} \\
\\
\gamma_{00k} = \lambda_{000} + \lambda_{001} \times COR_k + \omega_{00k} \\
\gamma_{01k} = \lambda_{010} + \lambda_{011} \times COR_k + \omega_{01k} \\
\gamma_{10k} = \lambda_{100} + \lambda_{101} \times COR_k + \omega_{10k} \\
\gamma_{11k} = \lambda_{110} + \lambda_{111} \times COR_k + \omega_{11k}
\end{cases}
\tag{1}
$$

3-level models

In terms of cross-level interactions, we can have:

- ✓ $L_3 \times L_2 \times L_1$: λ_{111} $(COR \times STS \times EDU)$
- ✓ $L_3 \times L_2$: λ_{011} $(COR \times STS)$
- ✓ $L_3 \times L_1$: λ_{101} $(COR \times EDU)$
- ✓ $L_2 \times L_1$: λ_{110} $(STS \times EDU)$

How many random effects in the model?

Keep in mind that n does not help you in any way when estimating L2 and L3 parameters.

ICC

One minor complication is the existence of multiple ICCs. Let's assume that the L_1, L_2 and L_3 residual variances are e_{ijk}, v_{0jk}, and ω_{00k} respectively.

$$ICC^{(3)} = \frac{s^2_{\omega_{00k}}}{s^2_{e_{ijk}} + s^2_{v_{0jk}} + s^2_{\omega_{00k}}} \tag{2}$$

$$ICC^{(2+3)} = \frac{s^2_{\omega_{00k}} + s^2_{v_{0jk}}}{s^2_{e_{ijk}} + s^2_{v_{0jk}} + s^2_{\omega_{00k}}} \tag{3}$$

$ICC^{(3)}$ reveals the share of level 3 variance out of total variance. $ICC^{(2+3)}$ displays similar information, for the second and third levels together.

Data management for MLM

Data management

Time for a "lighter" topic: how to manage data for an MLM analysis.

If you already have a data set which was built from the ground up with MLM in mind (e.g. CSES, WVS, ESS), half the work is already done.

You'll want to add some more group-level data to it (e.g. income inequality, unemployment rate), which means you'll have to build an additional data set yourself.

Data management

Merging will be done with the `join` family of function from `dplyr`.

```
merged_data <- left_join(l1data, l2data, by = "ID_var")
```

The two data sets have to have a variable called the same, with the same categories (if we continue with the country example, these categories can be the country names).

Biggest danger: `merged_data` ends up having duplicated rows, because there is more than one observation in `l2data` that can be matched with an observation from `l1data`.

Data management

country	X_1	X_2
Albania	1	10000
Algeria	1	8000
Belgium	0	22000

Data 1

country	X_3	X_4
Albania	25	1
Algeria	40	0
Argentina	35	5
Belgium	120	9
Bulgaria	25	6

Data 2

Here, the `country` variable is the same, and we can merge the data without a problem, since all the categories in the first data set are also present in the second (Albania, Algeria, Belgium).[2]

[2] Had this not been the case, the merging procedure wouldn't have worked.

Data management

If, however, you have to construct the data by yourselves, a good practice is to pay very close attention to the ID variable (in the past example, this was `country`).

As the data sets become more and more complex (pupils in classrooms, in schools, in countries), you have to make sure this variable is as clear and clean as possible.

Also, try to keep the level 1 and level 2 data sets separate, up until the actual moment of running the analyses—it makes it easier to graphically examine the variables.

Wide format

Religion	<$10k	$10-20k	$20-30k	$30-40k	$40-50k	$50-75k	$75-100k
Agnostic	27	34	60	81	76	137	122
Atheist	12	27	37	52	35	70	73
Buddhist	27	21	30	34	33	58	62
Catholic	418	617	732	670	638	1116	949
Evangelical Protestant	575	869	1064	982	881	1486	949
Hindu	1	9	7	9	11	34	47
Hist. Black Protestant	228	244	236	238	197	223	131
Jehovah's Witness	20	27	24	24	21	30	15
Jewish	19	19	25	25	30	95	69
DK/ref	15	14	15	11	10	35	21

Easy to look at breakdowns and examine associations, but not easy to feed into R's functions.

Long format

Religion	Income	Frequency
Agnostic	<$10k	27
Agnostic	$10-20k	34
Agnostic	$20-30k	60
Agnostic	$30-40k	81
Agnostic	$40-50k	76
Agnostic	$50-75k	137
Agnostic	$75-100k	122
Agnostic	$100-150k	109
Agnostic	>150k	84
Agnostic	Don't know/refused	96

Tidy data: (1) every observation is a row; (2) every variable is a column; (3) every data set contains a single type of observation.

dplyr & tidyr

Worth investing a lot of time in mastering these 2 packages:

- ✓ `mutate`: creating new variables;
- ✓ `rename`: renaming variables;
- ✓ `case_when`: recoding;
- ✓ `left_join`: data merging (only one of the functions in the family);
- ✓ `filter`: data subsetting;
- ✓ `select`: selecting columns (or excluding them).

`gather()` and `spread()` from `tidyr` do data reshaping in a very flexible way.

Applied task

Campaign contributions and voting

Data on 527 Members of Congress from 50 US states (data collected in late 1990s and early 2000s).

Outcome is `pct100`—the percentage of the time that legislator votes for pro-tobacco legislation (on a scale from 0 to 100).

Level-1 predictors:

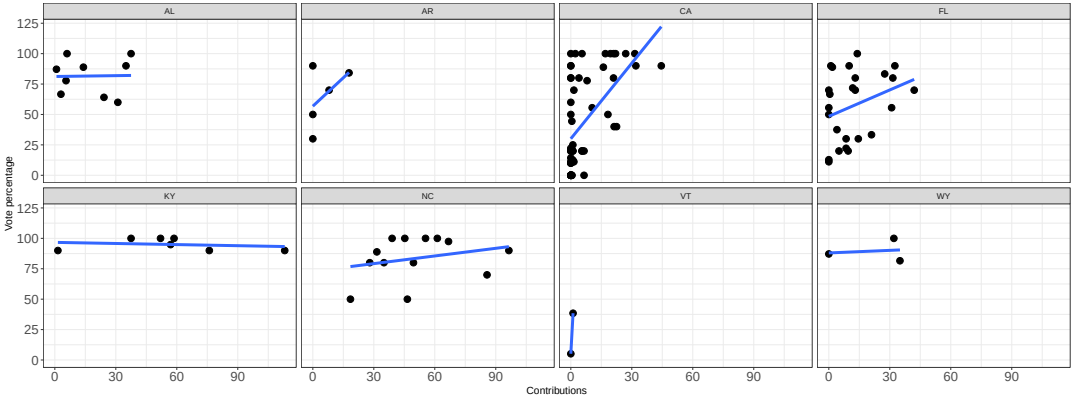
- ✓ `money1000`: the amount of contributions (in 1000s of USD) received by the legislator from tobacco-industry lobby groups;
- ✓ `party`: legislator is Republican (1 = yes; 0 = no);
- ✓ `house`: legislator is a member of the House (1 = yes; 0 = no).

Quick look at the data

house	state	sid	lastname	votepct	party	money	acres	pct100	money1000
0	AK	1	Murkowski	0.8421053	1	9166	0	84.21053	9.166
0	AK	1	Stevens	0.8461538	1	0	0	84.61538	0.000
1	AK	1	Young	0.5714286	1	23500	0	57.14286	23.500
0	AL	2	Sessions	0.8717949	1	-800	0	87.17949	0.800
0	AL	2	Shelby	0.6410256	1	24166	0	64.10256	24.166
1	AL	2	Aderholt	0.9000000	1	35000	0	90.00000	35.000

Observations are nested in states, with state-level information on number of acres of tobacco cultivated in the state (a likely proxy for how many workers the industry employs).

Quick look at the data



Congratulations! You're done with the first part.

References