The University of Texas at Austin
Educational Psychology
College of Education

Writing HLM Equations Correctly (And Easily)

Reza Norouzian © 2020-present

The University of Texas at Austin

Table of Contents

HLM Model Equations

You may have noticed that as HLM models get more complex, it becomes a non-trivial issue to write down their equations. I always ask students try NOT to memorize certain equations as the number of possible models are countless, and additionally such a strategy limits their modeling choices in the future.

It is, however, becoming increasingly common these days for researchers to include the model equations in their manuscripts (especially those intended for highly ranked journals). Fairly soon, my colleagues and I will be releasing a *development version* of an R package that alleviates this issue to some degree.

`equatiomatic` is a new R package that is still a *work in progress* and thus can be used only for calls to the `lmer()` function from the `lme4` package. Let's try to get this new package installed and apply it to some of the models we fit together during the course so far (and more). If you find any particular errors with this package (I have already found several), *I highly recommend that you send a quick email to me at:* reza.norouzian@austin.utexas.edu or my colleague, *Dr. Daniel Anderson* at: daniela@uoregon.edu.

However, before learning how to use the R package, a brief discussion on *illegitimate/misspecified* HLM models is waranted. So, let's see how such models, at this stage of the course, can come about.

Illegitimate/Invalid HLM Models

Fitting random slopes for predictors that don't vary within clusters

By now, we know that the formula defined in the *random part* of the model has to do with the variation at the lower level of our data. That is, variation within each level or cluster of our grouping variable. As such, to define the random part of the model syntax, it is very important to know that:

This is a place for predictors that can vary within each level of the grouping variable.

For example, in a 2-level model, the isolated use of a level-2 predictor in the random-part would not be legitimate because a level-2 predictor doesn't vary within each level or cluster of our grouping variable.

For instance, the following model is NOT a legitimate model:

```
hsb <-
read.csv('https://raw.githubusercontent.com/rnorouzian/e/master/hsb.csv')

mn <- lmer(math ~ ses + sector + (sector | sch.id), data = hsb)
```

You can see how `sector` doesn't vary within each school (`sch.id`) in the data extracted from the `hsb` dataset below:

```
##    sch.id    ses sector
## 1    1296  0.322      0
## 2    1296  0.052      0
## 3    1308 -0.568      1
## 4    1308  0.522      1
```

In other words, if we want to see how a relationship (e.g., `math ~ ses`) varies across various levels of our grouping variable (`sch.id`), that is how the slope for a predictor varies across various levels of our grouping variable (`sch.id`), then the predictor in that relationship must be a variable within each school (like `ses`):

ses: Level-1 Predictor
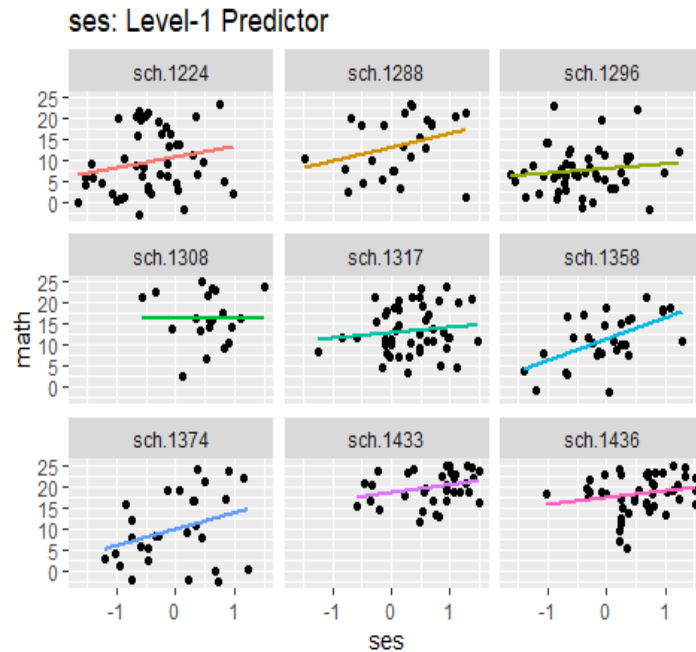
not a constant (like `sector`):

sector: Level-2 Predictor
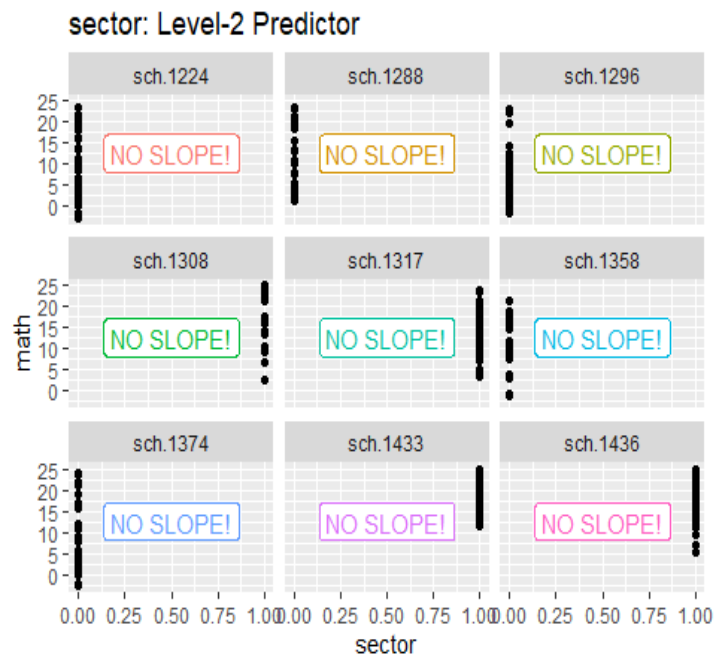
We can immediately see that in each school `sector` is a virtical accumulation of points. This is because each school can belong to only one type of `sector` (*public* or *private*) but not both. So, there are no slopes for `sector` in each school and asking the model to take a non-existing slope across the schools as random is illegitimate.

Therefore, the use of `sector`, a (level-2) predictor that doesn't vary within various levels of our grouping variable (`sch.id`), in the random part of the model: `(1 + sector|sch.id)` in not legitimate.

Note that sometimes a meaningless model can run without any warning but it is not a legitimate HLM model.

For example:

```
mn2 <- lmer(math ~ ses + (sector | sch.id), data = hsb)
```

The above model really runs because we have *removed* `sector` from the fixed part of the model!

We can extend this logic of HLM modeling to trickier models like:

```
mm <- lmer(math ~ ses*sector + (ses:sector | sch.id), data = hsb)
```

What do you think about the legitimacy of the above model, is it a legitimate model?

Let's use the small data extracted from the `hsb` dataset I showed above again:

```
##   sch.id    ses sector
## 1   1296  0.322      0
## 2   1296  0.052      0
## 3   1308 -0.568      1
## 4   1308  0.522      1
```

As we saw earlier `sector` doesn't vary within each cluster (`sch.id`) but how about its interaction with `ses` (i.e., `ses:sector`)?

Remember that an interaction is really the product of the two variables. If so, do you think `ses:sector` like `sector` doesn't vary within each cluster (`sch.id`)?

Let's take a look at `ses:sector`:

```
##   sch.id    ses sector ses:sector
## 1   1296  0.322      0      0.000
## 2   1296  0.052      0      0.000
## 3   1308 -0.568      1     -0.568
## 4   1308  0.522      1      0.522
```

`ses:sector` DOES vary (e.g., rows 3 and 4) within each cluster (`sch.id`)! Therefore, model `mm` is a legitimate model while `mn` is not!

Now, what do you think about the legitimacy of the following model?

```r
mm2 <- lmer(math ~ ses*sector + (ses*sector | sch.id), data = hsb)
```

Ignoring the "mixed" nature of HLM models

In the previous section, we discussed problems with the incorrect use of level-2 predictors in the random part of our models. However, another group of illegitimate models come about when we use level-1 predictors only. In this case, the problem arises when we forget that anything random needs to have a fixed representation in the model so that the model can be called a **"mixed-model"**.

For example, suppose we specify a model like so:

```r
mp <-  lmer(math ~ female*minority + (ses | sch.id), data = hsb)
```

In `mp`, what is the role of `ses` in the random part? What ***effect*** (i.e., slope/difference) is going to be measured across the levels of the grouping variable that would cause random variation across them?

Model `mp` is showing that that effect is the intercept and slope of `ses` by the way that its random part is set up (recall the random part can also be written as `(1 + ses | sch.id)`).

If so, then, where is `ses` in the fixed part to denote that we have measured the effect of `ses` across the levels of the grouping variable?!

Model `mp` is, then, illegitimate regardless of throwing any warning!

IF we had measured the intercept and slope of `ses` in the fixed part of the model, that is:

```r
mp1 <-  lmer(math ~ ses+female*minority + (ses | sch.id), data = hsb)
```

then, the model would have been a legitimate HLM model.

Note that the above case could happen in a trickier way as well. Take, for example, the following model:

```r
mq <- lmer(math ~ female*minority + (ses:female | sch.id), data = hsb)
```

Model `mq` is also illegitimate. Because the random part of the model is saying that the slope for each unique combination of `ses` and `female` in each `sch.id` varies across `sch.id`, but no `ses` effect is included in the fixed part of the model.

Now, what do you think about the legitimacy of the following model?

```r
mq2 <- lmer(math ~ ses+female*minority + (ses:female | sch.id), data = hsb)
```

Package equatiomatic Installation

The previous discussion on the model legitimacy is important because our package only works with legitimate HLM models (You'll get either an error or nonsensical (scrambled

HTML) output if you input an illegitimate model to `equatiomatic` main function: `extract_eq`)!

You'll need to install the development version of the package from github, and to do so you will need to have installed the R package remotes. Then, to install `equatiomatic` use the following command:

```r
remotes::install_github("datalorax/equatiomatic@merMod-refactor")
```

Note: *In the near future, this method of installation may change. Please contact us via email (see above) to inquire.*

Writing the Model Equations in `equatiomatic`

Model 1:

Let's start with a random-intercept (simplest model):

```r
library(equatiomatic)
library(lme4)

hsb <-
read.csv('https://raw.githubusercontent.com/rnorouzian/e/master/hsb.csv')

m1 <- lmer (math ~ 1 + (1 | sch.id), data = hsb)
extract_eq(m1)
```

$$\text{math} \sim N\left(\alpha_{j[i]}, \sigma^2\right)$$
$$\alpha_j \sim N\left(\mu_{\alpha_j}, \sigma^2_{\alpha_j}\right), \text{for sch.id j} = 1, \dots, J$$

As you can see, the notations maybe a bit different from what we used in the course but that is not an issue as long as you understand them and can describe them to your audience, they help you save time. At the moment, `extract_eq()` uses α to denote the intercept and β to denote the slopes. Elements that are taken as random, are subscripted by an $[i]$ notation, meaning that they are supposed to vary across the levels of the *grouping variable*. At the higher level, the subscripted element (here $\alpha_{j[i]}$) is opened up to show its distributional specification. Notice that the part $for sch.id j$ should be $for\ sch.id\ j$ (again remember the package is still being developed). If you want to manually correct problems like that you can render your `.Rmd` file as a `.docx` (MS Word) file and then correct the space problems in the rendered file.

Let's go over a few more models just to make sure we have grasped the idea behind the automated equation writing.

Model 2:

Here is a means-as-outcomes model:

```
m2 <- lmer(math ~ meanses + (1|sch.id), data = hsb)
```

```
extract_eq(m2)
```

$$\text{math} \quad \sim N\left(\alpha_{j[i]}, \sigma^2\right)$$
$$\alpha_j \quad \sim N\left(\gamma_0^{\alpha} + \gamma_1^{\alpha}(\text{meanses}), \sigma_{\alpha_j}^2\right), \text{for sch.id } j = 1, \dots, J$$

Model 22:

And then:

```
hsb <- transform(hsb, sector = factor(ifelse(sector==0, "pub", "cath")))
hsb <- transform(hsb, sector = relevel(sector, ref = "pub"),
gender=ifelse(female==0, "M", "F"))
```

```
m22 <- lmer(math ~ sector + (1|sch.id), data = hsb)
```

```
extract_eq(m22)
```

$$\text{math} \quad \sim N\left(\alpha_{j[i]}, \sigma^2\right)$$
$$\alpha_j \quad \sim N\left(\gamma_0^{\alpha} + \gamma_1^{\alpha}(\text{sector}_{\text{cath}}), \sigma_{\alpha_j}^2\right), \text{for sch.id } j = 1, \dots, J$$

Model 3:

Now let's use a model with correlated (denoted as off-diagonal ρs) random slopes and intercepts. Note that this also means we have a G matrix defining the covariance matrix of our $MVnorm$ distribution:

```
m3 <- lmer(math ~ ses + (ses | sch.id), data = hsb)
```

```
extract_eq(m3)
```

$$\text{math} \quad \sim N\left(\alpha_{j[i]} + \beta_{1j[i]}(\text{ses}), \sigma^2\right)$$
$$\begin{pmatrix} \alpha_j \\ \beta_{1j} \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_{\alpha_j} \\ \mu_{\beta_{1j}} \end{pmatrix}, \begin{pmatrix} \sigma_{\alpha_j}^2 & \rho\alpha_j\beta_{1j} \\ \rho\beta_{1j}\alpha_j & \sigma_{\beta_{1j}}^2 \end{pmatrix}\right), \text{for sch.id } j = 1, \dots, J$$

Model 4:

Also:

```
m4 <- lmer(math ~ ses*sector + (ses | sch.id), data = hsb)
```

```
extract_eq(m4)
```

$$\text{math} \sim N\big(\alpha_{j[i]} + \beta_{1j[i]}(\text{ses}), \sigma^2\big)$$

$$\begin{pmatrix} \alpha_j \\ \beta_{1j} \end{pmatrix} \sim N\left(\begin{pmatrix} \gamma_0^{\alpha} + \gamma_1^{\alpha}(\text{sector}_{\text{cath}}) \\ \gamma_0^{\beta_1} + \gamma_1^{\beta_1}(\text{sector}_{\text{cath}}) \end{pmatrix}, \begin{pmatrix} \sigma_{\alpha_j}^2 & \rho\alpha_j\beta_{1j} \\ \rho\beta_{1j}\alpha_j & \sigma_{\beta_{1j}}^2 \end{pmatrix}\right), \text{ for sch.id } j = 1, \dots, J$$

Model 41:

The converging alternative of `m4` was:

```
m41 <- lmer(math ~ ses*sector + (1 | sch.id), data = hsb)
```

```
extract_eq(m41)
```

$$\text{math} \sim N\big(\alpha_{j[i]} + \beta_1(\text{ses}), \sigma^2\big)$$

$$\alpha_j \sim N\left(\gamma_0^{\alpha} + \gamma_1^{\alpha}(\text{sector}_{\text{cath}}) + \gamma_2^{\alpha}(\text{ses} \times \text{sector}_{\text{cath}}), \sigma_{\alpha_j}^2\right), \text{ for sch.id } j = 1, \dots, J$$

Model 42:

Also, suppose you want to know the equations for a model we didn't run in the class:

```
m42 <- lmer(math ~ ses+minority + (ses | sch.id), data = hsb)
```

```
extract_eq(m42)
```

$$\text{math} \sim N\big(\alpha_{j[i]} + \beta_{1j[i]}(\text{ses}) + \beta_2(\text{minority}), \sigma^2\big)$$

$$\begin{pmatrix} \alpha_j \\ \beta_{1j} \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_{\alpha_j} \\ \mu_{\beta_{1j}} \end{pmatrix}, \begin{pmatrix} \sigma_{\alpha_j}^2 & \rho\alpha_j\beta_{1j} \\ \rho\beta_{1j}\alpha_j & \sigma_{\beta_{1j}}^2 \end{pmatrix}\right), \text{ for sch.id } j = 1, \dots, J$$

Model 43:

Here is an example of an **"illegitimate"** model (**scrambled output not shown**):

```
m43 <- lmer(math ~ gender*minority + (ses | sch.id), data = hsb)
```

```
extract_eq(m43)
```

Model 5:

Also, for a very busy singular G matrix model we had:

```
sng <-
read.csv('https://raw.githubusercontent.com/rnorouzian/e/master/sng.csv')

m5 <- lmer(y ~ A * B * C + (A * B * C | group), data = sng)
extract_eq(m5)
```

$$y \sim N(\mu, \sigma^2)$$
$$\mu = \alpha_{j[i]} + \beta_{1j[i]}(A) + \beta_{2j[i]}(B) + \beta_{3j[i]}(C) + \beta_{4j[i]}(A \times B) + \beta_{5j[i]}(A \times C) + \beta_{6j[i]}(B \times C) + \beta_{7j[i]}(A \times B \times C)$$

$$
\begin{pmatrix} \alpha_j \\ \beta_{1j} \\ \beta_{2j} \\ \beta_{3j} \\ \beta_{4j} \\ \beta_{5j} \\ \beta_{6j} \\ \beta_{7j} \end{pmatrix} \sim N \left(\begin{pmatrix} \mu_{\alpha_j} \\ \mu_{\beta_{1j}} \\ \mu_{\beta_{2j}} \\ \mu_{\beta_{3j}} \\ \mu_{\beta_{4j}} \\ \mu_{\beta_{5j}} \\ \mu_{\beta_{6j}} \\ \mu_{\beta_{7j}} \end{pmatrix}, \begin{pmatrix} \sigma^2_{\alpha_j} & \rho\alpha_j\beta_{1j} & \rho\alpha_j\beta_{2j} & \rho\alpha_j\beta_{3j} & \rho\alpha_j\beta_{4j} & \rho\alpha_j\beta_{5j} & \rho\alpha_j\beta_{6j} & \rho\alpha_j\beta_{7j} \\ \rho\beta_{1j}\alpha_j & \sigma^2_{\beta_{1j}} & \rho\beta_{1j}\beta_{2j} & \rho\beta_{1j}\beta_{3j} & \rho\beta_{1j}\beta_{4j} & \rho\beta_{1j}\beta_{5j} & \rho\beta_{1j}\beta_{6j} & \rho\beta_{1j}\beta_{7j} \\ \rho\beta_{2j}\alpha_j & \rho\beta_{2j}\beta_{1j} & \sigma^2_{\beta_{2j}} & \rho\beta_{2j}\beta_{3j} & \rho\beta_{2j}\beta_{4j} & \rho\beta_{2j}\beta_{5j} & \rho\beta_{2j}\beta_{6j} & \rho\beta_{2j}\beta_{7j} \\ \rho\beta_{3j}\alpha_j & \rho\beta_{3j}\beta_{1j} & \rho\beta_{3j}\beta_{2j} & \sigma^2_{\beta_{3j}} & \rho\beta_{3j}\beta_{4j} & \rho\beta_{3j}\beta_{5j} & \rho\beta_{3j}\beta_{6j} & \rho\beta_{3j}\beta_{7j} \\ \rho\beta_{4j}\alpha_j & \rho\beta_{4j}\beta_{1j} & \rho\beta_{4j}\beta_{2j} & \rho\beta_{4j}\beta_{3j} & \sigma^2_{\beta_{4j}} & \rho\beta_{4j}\beta_{5j} & \rho\beta_{4j}\beta_{6j} & \rho\beta_{4j}\beta_{7j} \\ \rho\beta_{5j}\alpha_j & \rho\beta_{5j}\beta_{1j} & \rho\beta_{5j}\beta_{2j} & \rho\beta_{5j}\beta_{3j} & \rho\beta_{5j}\beta_{4j} & \sigma^2_{\beta_{5j}} & \rho\beta_{5j}\beta_{6j} & \rho\beta_{5j}\beta_{7j} \\ \rho\beta_{6j}\alpha_j & \rho\beta_{6j}\beta_{1j} & \rho\beta_{6j}\beta_{2j} & \rho\beta_{6j}\beta_{3j} & \rho\beta_{6j}\beta_{4j} & \rho\beta_{6j}\beta_{5j} & \sigma^2_{\beta_{6j}} & \rho\beta_{6j}\beta_{7j} \\ \rho\beta_{7j}\alpha_j & \rho\beta_{7j}\beta_{1j} & \rho\beta_{7j}\beta_{2j} & \rho\beta_{7j}\beta_{3j} & \rho\beta_{7j}\beta_{4j} & \rho\beta_{7j}\beta_{5j} & \rho\beta_{7j}\beta_{6j} & \sigma^2_{\beta_{7j}} \end{pmatrix} \right), \text{ for group j = 1, ...,J}
$$

Model 51:

Which we then, following Bates et al. (2015), simplified to:

```
m51 <- lmer(y ~ A * B * C + (A + B + C || group), data = sng)
extract_eq(m51)
```

$$y \sim N(\mu, \sigma^2)$$
$$\mu = \alpha_{j[i]} + \beta_{1j[i]}(A) + \beta_{2j[i]}(B) + \beta_{3j[i]}(C) + \beta_4(A \times B) + \beta_5(A \times C) + \beta_6(B \times C) + \beta_7(A \times B \times C)$$

$$
\begin{pmatrix} \alpha_j \\ \beta_{1j} \\ \beta_{2j} \\ \beta_{3j} \end{pmatrix} \sim N \left(\begin{pmatrix} \mu_{\alpha_j} \\ \mu_{\beta_{1j}} \\ \mu_{\beta_{2j}} \\ \mu_{\beta_{3j}} \end{pmatrix}, \begin{pmatrix} \sigma^2_{\alpha_j} & 0 & 0 & 0 \\ 0 & \sigma^2_{\beta_{1j}} & 0 & 0 \\ 0 & 0 & \sigma^2_{\beta_{2j}} & 0 \\ 0 & 0 & 0 & \sigma^2_{\beta_{3j}} \end{pmatrix} \right), \text{ for group j = 1, ...,J}
$$

Model 6:

Here is a model with the correlation(s) between random-effects dropped (set to 0). Here is another one:

```
m6 <- lmer(math ~ ses + (ses || sch.id), data = hsb)
extract_eq(m6)
```

$$\text{math} \quad \sim N\big(\alpha_{j[i]} + \beta_{1j[i]}(\text{ses}), \sigma^2\big)$$

$$\begin{pmatrix} \alpha_j \\ \beta_{1j} \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_{\alpha_j} \\ \mu_{\beta_{1j}} \end{pmatrix}, \begin{pmatrix} \sigma^2_{\alpha_j} & 0 \\ 0 & \sigma^2_{\beta_{1j}} \end{pmatrix} \right), \text{ for sch.id } j = 1, \dots, J$$

Model 7:

Finally, `extract_eq` is a model with a split grouping variable (i.e., use of `group` in two separate (`|`) expressions):

```
m7 <- lmer(y ~ A * B * C + (A * C | group) + (B|group) , data = sng)
extract_eq(m7)
```

$$y \sim N(\mu, \sigma^2)$$
$$\mu = \alpha_{j[i],k[i]} + \beta_{1j[i]}(A) + \beta_{2k[i]}(B) + \beta_{3j[i]}(C) + \beta_4(A \times B) + \beta_{5j[i]}(A \times C) + \beta_6(B \times C) + \beta_7(A \times B \times C)$$

$$\begin{pmatrix} \alpha_j \\ \beta_{1j} \\ \beta_{3j} \\ \beta_{5j} \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_{\alpha_j} \\ \mu_{\beta_{1j}} \\ \mu_{\beta_{3j}} \\ \mu_{\beta_{5j}} \end{pmatrix}, \begin{pmatrix} \sigma^2_{\alpha_j} & \rho\alpha_j\beta_{1j} & \rho\alpha_j\beta_{3j} & \rho\alpha_j\beta_{5j} \\ \rho\beta_{1j}\alpha_j & \sigma^2_{\beta_{1j}} & \rho\beta_{1j}\beta_{3j} & \rho\beta_{1j}\beta_{5j} \\ \rho\beta_{3j}\alpha_j & \rho\beta_{3j}\beta_{1j} & \sigma^2_{\beta_{3j}} & \rho\beta_{3j}\beta_{5j} \\ \rho\beta_{5j}\alpha_j & \rho\beta_{5j}\beta_{1j} & \rho\beta_{5j}\beta_{3j} & \sigma^2_{\beta_{5j}} \end{pmatrix} \right), \text{ for group } j = 1, \dots, J$$

$$\begin{pmatrix} \alpha_k \\ \beta_{2k} \end{pmatrix} \sim N\left(\begin{pmatrix} \mu_{\alpha_k} \\ \mu_{\beta_{2k}} \end{pmatrix}, \begin{pmatrix} \sigma^2_{\alpha_k} & \rho\alpha_k\beta_{2k} \\ \rho\beta_{2k}\alpha_k & \sigma^2_{\beta_{2k}} \end{pmatrix} \right), \text{ for group.1 } k = 1, \dots, K$$

In the above model, we have *uncorrelated* `B` from `A` and `C` in the random part of the model.