

Module 7: Multilevel Models for Binary Responses

R Practical

Camille Szmaragd and George Leckie¹
Centre for Multilevel Modelling

Pre-requisites

- Modules 1-6

Contents

Introduction to the Bangladesh Demographic and Health Survey 2004 Dataset	2
P7.1 Two-Level Random Intercept Model	4
P7.1.1 Specifying and estimating a two-level model	5
P7.1.2 Interpretation of the null two-level model	7
P7.1.3 Adding an explanatory variable.....	9
P7.2 Latent Variable Representation of a Random Intercept Model.....	14
P7.2.1 Comparison of a single-level and multilevel threshold model.....	14
P7.2.2 Variance partition coefficient	18
P7.3 Population-Averaged and Cluster-Specific Effects	19
P7.4 Predicted Probabilities from a Multilevel Model	20
P7.5 Two-Level Random Slope Model	27
P7.5.1 Allowing the effect of wealth to vary across communities	28
P7.5.2 Interpretation of a random slope model	30
P7.5.3 Fitting random coefficients to categorical wealth	35
P7.6 Adding Level 2 Explanatory Variables: Contextual Effects.....	44
P7.6.1 Contextual effects	45
P7.6.2 Cross-level interactions	50
P7.7 Estimation of Binary Response Models: MCMC Methods	55
References	55

¹ This R practical is adapted from the corresponding MLwiN practical: Steele, F. (2008) Module 7: Multilevel Models for Binary Responses. LEMMA VLE, Centre for Multilevel Modelling. Accessed at <http://www.cmm.bris.ac.uk/lemma/course/view.php?id=13>.

Most of the sections within this module have online quizzes for you to test your understanding. To find the quizzes:

From within the LEMMA learning environment

- Go down to the section for **Module 7: Multilevel Models for Binary Responses**
- Click "[7.1 Two-Level Random Intercept Model](#)" to open Lesson 7.1
- Click  to open the first question

Introduction to the Bangladesh Demographic and Health Survey 2004 Dataset

You will be analysing data from the Bangladesh Demographic and Health Survey (BDHS),² a nationally representative cross-sectional survey of women of reproductive age (13-49 years).

Our response variable is a binary indicator of whether a woman received antenatal care from a medically-trained provider (a doctor, nurse or midwife) at least once before her most recent live birth. To minimise recall errors, the question was asked only about children born within five years of the survey. For this reason, our analysis sample is restricted to women who had a live birth in the five-year period before the survey. Note that if a woman had more than one live birth during the reference period, we consider only the most recent.

These data were analysed in Module 6 using single-level models. In this module, we consider multilevel models to allow for and to explore between-community variance in antenatal care. The data have a two-level hierarchical structure with 5366 women at level 1, nested within 361 communities at level 2. In rural areas a community corresponds to a village, while an urban community is a neighbourhood based on census definitions.

We consider a range of predictors. At level 1, we consider variables such as a woman's age at the time of the birth and education. Level 2 variables include an indicator of whether the region of residence is classified as urban or rural. We will also derive community-level measures by aggregating woman-level variables, for example the proportion of respondents in the community who are in the top quintile of a wealth index.

² We thank MEASURE DHS for their permission to make these data available for training purposes. Additional information about the 2004 BDHS and other Demographic and Health Surveys, including details of how to register for a DHS Download Account, is available from www.measuredhs.com.

The file contains the following variables:

Variable name	Description and codes
comm	Community identifier
womid	Woman identifier
antemed	Received antenatal care at least once from a medically-trained provider, e.g. doctor, nurse or midwife (1 = yes, 0 = no)
bord	Birth order of child (ranges from 1 to 13)
mage	Mother's age at the child's birth (in years)
urban	Type of region of residence at survey (1 = urban, 0 = rural)
meduc	Mother's level of education at survey (1 = none, 2 = primary, 3 = secondary or higher)
islam	Mother's religion (1 = Islam, 0 = other)
wealth	Household wealth index in quintiles (1 = poorest to 5 = richest)

The dataset also contains a number of extra variables derived from those above (see the practical for Module 6).

P7.1 Two-Level Random Intercept Model

Download the R dataset for this lesson:

From within the LEMMA Learning Environment

- Go to **Module 7: Multilevel Models for Binary Responses**, and scroll down to **R Datasets and R files**
- Right click "7.1.txt" and select **Save Link As...** to save the dataset to your computer

Read the dataset into R and create a dataframe object named `mydata`³:

```
> mydata <- read.table("7.1.txt", header = TRUE, sep = ",")
```

and use the `str` command to produce a summary of the dataset:

```
> str(mydata)
'data.frame': 5366 obs. of 17 variables:
 $ comm : int 1 1 1 1 1 1 1 1 1 1 ...
 $ womid : int 1 2 3 4 5 6 7 8 9 10 ...
 $ antemed: int 0 1 1 0 0 1 0 0 0 1 ...
 $ bord : int 4 2 3 6 6 4 2 3 1 1 ...
 $ mage : int 33 21 26 28 37 29 20 29 19 19 ...
 $ urban : int 0 0 0 0 0 0 0 0 0 0 ...
 $ meduc : int 2 3 2 1 2 2 3 3 3 3 ...
 $ islam : int 1 1 1 1 1 1 1 1 1 1 ...
 $ wealth : int 3 4 2 2 4 4 2 3 3 4 ...
 $ magec : num 9.37 -2.63 2.37 4.37 13.37 ...
 $ magecsq: num 87.72 6.94 5.6 19.06 178.64 ...
 $ meduc2 : int 1 0 1 0 1 1 0 0 0 0 ...
 $ meduc3 : int 0 1 0 0 0 0 1 1 1 1 ...
 $ wealth2: int 0 0 1 1 0 0 1 0 0 0 ...
 $ wealth3: int 1 0 0 0 0 0 0 1 1 0 ...
 $ wealth4: int 0 1 0 0 1 1 0 0 0 1 ...
 $ wealth5: int 0 0 0 0 0 0 0 0 0 0 ...
```

³ At the beginning of your R session, you will need to set R's working directory to the file location where you saved the dataset. This can be done using the command line and the `setwd` function:

```
> setwd("C:\\userdirectory\\")
```

Or through selecting Change Dir... on the File menu.

P7.1.1 Specifying and estimating a two-level model

We will begin by fitting a null or empty two-level model, that is a model with only an intercept and community effects.

$$\log\left(\frac{\pi_{ij}}{1-\pi_{ij}}\right) = \beta_0 + u_{0j}$$

The intercept β_0 is shared by all communities while the random effect u_{0j} is specific to community j . The random effect is assumed to follow a normal distribution with variance σ_{u0}^2 .

R's main command for fitting multilevel models for binary and other discrete response variables is the `glmer` command which is part of an additional `lme4` library⁴, which we used already in Module 5. This library can be installed through the R Packages menu; select Install Package(s) and then select the correct Mirror and package from the scroll-down menus. As you will see, there is a variety of additional packages that can be installed with R. You only need to install a package once to your own computer. If you then want to use the package, you simply need to call it from within R prior to using the command for the first time in each R session⁵:

```
> library(lme4)
Loading required package: Matrix
Loading required package: lattice

Attaching package: 'lme4'
```

The following object(s) are masked from package:stats :

AIC

The syntax for `glmer` is similar to that for the `lmer` command which we introduced in Module 5. To fit the above model using the `glmer` command and to create a model object `fit`, we type:

```
> fit <- glmer(antemed ~ (1 | comm), family = binomial("logit"), data = mydata)
```

The binary response variable (`antemed`) follows the command which is then followed by a `~` and then by a list of fixed part explanatory variables (excluding the constant as this is included by default⁶). The above model contains only an intercept and so no fixed part explanatory variables are specified. The level 2 random part of the model is specified in brackets by the list of random part explanatory variables (the constant has to be explicitly specified by `1`, followed by

⁴ `lme4` is a package developed by Douglas Bates and Martin Maechler for fitting linear and generalized linear mixed-effect models. For more details about this library, see Module 5.

⁵ You may get slightly different messages when calling `library(lme4)` as it will depend on your system and the version of R you have installed.

⁶ Note, to omit the constant you need to add `-1` to the right-hand side of the `"~"` sign.

a single vertical bar `|` and then by the level 2 identifier (`comm`). The family option is used to specify a binomial distribution for the response with a `logit` link function. The `data` option specifies the dataframe being used to fit the model.

We then display the results using the `summary` command, which gives the following output:

```
> summary(fit)

Generalized linear mixed model fit by the Laplace approximation
Formula: antemed ~ (1 | comm)
Data: mydata
AIC BIC logLik deviance
6640 6653 -3318 6636
Random effects:
Groups Name Variance Std.Dev.
comm (Intercept) 1.4644 1.2101
Number of obs: 5366, groups: comm, 361

Fixed effects:
Estimate Std. Error z value Pr(>|z|)
(Intercept) 0.14811 0.07136 2.075 0.0379 *
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Before interpreting the model, we will discuss the estimation procedure that `glmer` uses⁷. The estimation procedure optimizes a function of the log likelihood using penalized iteratively re-weighted least squares. The log-likelihood is evaluated using the Laplacian approximation⁸. This approximation method may lead to unsatisfactory estimates, so we suggest that the reader uses the parameter estimates provided by R as a starting point and check their results using MCMC (for example with `MLwiN` or `WinBUGS`).

⁷ For further details see the PDF vignettes available on the `lme4` website <http://cran.r-project.org/web/packages/lme4>, in particular the vignette entitled "Computational Methods" which deals with the statistical theory.

⁸ In the `glmer` help file, it is noted that by specifying the option `nAGQ=n`, with `n` greater than 1, changes the approximation method to adaptive Gauss-Hermite approximation, with a greater value of `n` leading to more accurate evaluation of the log-likelihood. However, when we tried this in this and other examples we found that specifying different values of `n` did not lead to any change in our estimates. This apparent bug may be fixed in later versions of `lme4`. We refer the reader to the following blog for a discussion of the different approximation methods available in R (`glmer`) and Stata (`xtmelogit` command): http://www.stat.columbia.edu/~cook/movabletype/archives/2010/09/r_vs_stata_or_d.html.

P7.1.2 Interpretation of the null two-level model

From the model estimates (using Laplacian approximation), we can say that the log-odds of receiving antenatal care from a medically-trained provider in an 'average' community (one with $u_{0j} = 0$) is estimated as $\beta_0 = 0.148$. The intercept for community j is $0.148 + u_{0j}$, where the variance of u_{0j} is estimated as $\sigma_{u_0}^2 = 1.464$.

The likelihood ratio statistic for testing the null hypothesis, that $\sigma_{u_0}^2 = 0$, can be calculated by comparing the two-level model, with the corresponding single-level model without the level 2 random effects.

```
> fita <- glm(antemed ~ 1, data = mydata, family = binomial("logit"))
> logLik(fita) - logLik(fit)
'log Lik.' -399.8392 (df=1)
```

The test statistic is 799.8 ($-2 \times (-399.83)$) with 1 degree of freedom, so there is strong evidence that the between-community variance is non-zero.⁹

We will now examine estimates of the community effects or residuals, \hat{u}_{0j} , obtained from the null model. To calculate the residuals and produce a 'caterpillar plot' with the community effects shown in rank order together with 95% confidence intervals we can use the same commands as we used P5.1.2 for the continuous response two-level random intercepts model:

```
> u0 <- ranef(fit, postVar = TRUE)
> u0se <- sqrt(attr(u0[[1]], "postVar")[1, , ])
> commid <- as.numeric(rownames(u0[[1]]))
> u0tab <- cbind("commid" = commid, "u0" = u0[[1]], "u0se" = u0se)
> colnames(u0tab)[2] <- "u0"
> u0tab <- u0tab[order(u0tab$u0), ]
> u0tab <- cbind(u0tab, c(1:dim(u0tab)[1]))
> u0tab <- u0tab[order(u0tab$commid), ]
> colnames(u0tab)[4] <- "u0rank"
```

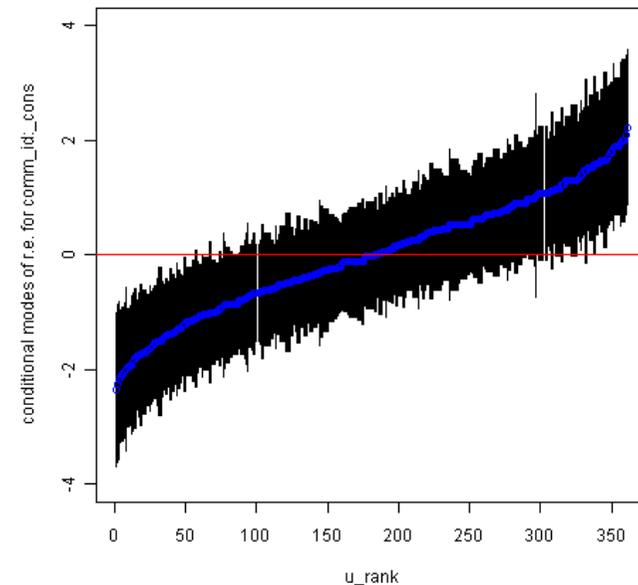
⁹ Note that the test statistic has a non-standard sampling distribution as the null hypothesis of a zero variance is on the boundary of the parameter space; we do not envisage a negative variance. In this case the correct p-value is half the one obtained from the tables of chi-squared distribution with 1 degree of freedom.

```
> plot(u0tab$u0rank, u0tab$u0, type = "n", xlab = "u_rank", ylab = "conditional
modes of r.e. for comm_id:_cons", ylim = c(-4, 4))

> segments(u0tab$u0rank, u0tab$u0 - 1.96*u0tab$u0se, u0tab$u0rank, u0tab$u0 +
1.96*u0tab$u0se)

> points(u0tab$u0rank, u0tab$u0, col = "blue")

> abline(h = 0, col = "red")
```



The plot shows the estimated residuals for all 361 communities in the sample. For a substantial number of communities, the 95% confidence interval does not overlap the horizontal line at zero, indicating that uptake of antenatal care in these communities is significantly above average (above the zero line) or below average (below the zero line). Compared to the plot for the US election data (C7.2), the confidence intervals are quite wide. This is because the sample size within a community is much smaller than the sample size within a state, leading to larger standard errors for the estimated community residuals \hat{u}_{0j} .

P7.1.3 Adding an explanatory variable

Next we will include maternal age as an explanatory variable in the model. Although we know from our single-level analysis (P6.1 and P6.6) that there is a curvilinear relationship between the log-odds of antenatal care and age, we will start by fitting a linear age effect.

$$\log\left(\frac{\pi_{ij}}{1-\pi_{ij}}\right) = \beta_0 + \beta_1 \text{magec}_{ij} + u_{0j}$$

```
> (fit2 <- glmer(antemed ~ magec + (1 | comm), family = binomial("logit"), data = mydata))

Generalized linear mixed model fit by the Laplace approximation
Formula: antemed ~ magec + (1 | comm)
Data: mydata
   AIC   BIC logLik deviance
6603 6623  -3299    6597
Random effects:
Groups Name      Variance Std.Dev.
comm  (Intercept) 1.4622   1.2092
Number of obs: 5366, groups: comm, 361

Fixed effects:
              Estimate Std. Error z value Pr(>|z|)
(Intercept)  0.144680    0.071365   2.027  0.0426 *
magec        -0.032394    0.005163  -6.275 3.51e-10 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Correlation of Fixed Effects:
      (Intr)
magec 0.009
```

Note that there is little change in the estimate of the between-community variance, suggesting that the distribution of maternal age is similar across communities.

The equation of the average fitted regression line, expressing the relationship between the log-odds of receiving antenatal care and maternal age, is:

$$\log\left(\frac{\hat{\pi}_{ij}}{1-\hat{\pi}_{ij}}\right) = 0.144 - 0.032 \text{magec}_{ij}$$

The fitted line for a given community will differ from the average line in its intercept, by an amount \hat{u}_j for community j . A plot of the predicted community lines will therefore show a set of parallel lines. To produce this plot, we first need to calculate the predicted log-odds of antenatal care for each woman, based on her age at survey and community of residence. To do this we compute the predicted probability of antenatal care for each woman using the `fitted` command. This command extracts the predicted (or fitted) values and is an alternative to the `predict` command which is not available for the models fitted using `lmer` and `glmer`.

```
> predprob <- fitted(fit2)
```

We then transform these predicted probabilities to predicted log odds using the `logit()` function from the R additional `VGAM` library.

```
> library(VGAM)
Loading required package: splines
Loading required package: stats4

Attaching package: 'VGAM'
```

```
The following object(s) are masked from package:splines :
```

```
bs,
ns
```

```
The following object(s) are masked from package:lme4 :
```

```
fitted,
formula,
residuals
```

```
The following object(s) are masked from package:Matrix :
```

```
print
```

```
The following object(s) are masked from package:stats :
```

```
biplot,
case.names,
coefficients,
df.residual,
fitted,
fitted.values,
formula,
poly,
residuals,
variable.names,
weights
```

```
The following object(s) are masked from package:base :
```

```
identity,
print,
scale.default
```

```
> predlogit <- logit(predprob)
```

Loading the `VGAM` library will highlight that some commands from the `lme4` library, including `fitted`, are now masked. This means that the next time you need to use the `fitted` command you will need to specify that you want to use the `fitted` command from the `lme4` library and not the `VGAM` library.

This document is only the first few pages of the full version.

To see the complete document please go to learning materials and register:

<http://www.cmm.bris.ac.uk/lemma>

The course is completely free. We ask for a few details about yourself for our research purposes only. We will not give any details to any other organisation unless it is with your express permission.