Generalized Multilevel Regression Example for a Binary Outcome

HLM

To estimate a multilevel logistic model, go to *Basic Settings* and choose *Bernoulli*. I used adaptive quadrature here (under Other Settings → Estimation Settings) but the sixth-order Laplace estimates (labeled Laplace-2 in the output) would also be acceptable. You must indicate the maximum number of iterations (100 was acceptable here) and the number of quadrature points. I used 7 for this example.[1]

Standard error of τ
INTRCPT1,β_0 0.21399

Final estimation of fixed effects (Unit-specific model)

Fixed Effect	Coefficient	Standard error	*t*-ratio	Approx. *d.f.*	*p*-value
For INTRCPT1, β_0					
INTRCPT2, γ_{00}	-3.177336	0.169536	-18.741	26	<0.001
DROPOUT, γ_{01}	0.021181	0.056373	0.376	26	0.710
For RACE slope, β_1					
INTRCPT2, γ_{10}	0.328923	0.254007	1.295	2154	0.195
For EROSION slope, β_2					
INTRCPT2, γ_{20}	0.810335	0.146449	5.533	2154	<0.001

Statistics for the current model
Deviance = 4803.854040
Number of estimated parameters = 5

Results for Population-Average Model
The value of the log-likelihood function at iteration 3 = -2.974930E+003

Final estimation of fixed effects: (Population-average model)

Fixed Effect	Coefficient	Standard error	*t*-ratio	Approx. *d.f.*	*p*-value
For INTRCPT1, β_0					
INTRCPT2, γ_{00}	-3.044702	0.161752	-18.823	26	<0.001
DROPOUT, γ_{01}	0.021796	0.055691	0.391	26	0.699
For RACE slope, β_1					
INTRCPT2, γ_{10}	0.314913	0.237768	1.324	2154	0.185
For EROSION slope, β_2					
INTRCPT2, γ_{20}	0.795626	0.138454	5.746	2154	<0.001

Fixed Effect	Coefficient	Odds Ratio	Confidence Interval
For INTRCPT1, β_0			
INTRCPT2, γ_{00}	-3.044702	0.047611	(0.034,0.066)
DROPOUT, γ_{01}	0.021796	1.022035	(0.911,1.146)
For RACE slope, β_1			
INTRCPT2, γ_{10}	0.314913	1.370140	(0.859,2.185)
For EROSION slope, β_2			
INTRCPT2, γ_{20}	0.795626	2.215827	(1.689,2.907)

Final estimation of fixed effects
(Population-average model with robust standard errors)

Fixed Effect	Coefficient	Standard error	*t*-ratio	Approx. *d.f.*	*p*-value
For INTRCPT1, β_0					
INTRCPT2, γ_{00}	-3.044702	0.144015	-21.142	26	<0.001
DROPOUT, γ_{01}	0.021796	0.075317	0.289	26	0.775
For RACE slope, β_1					
INTRCPT2, γ_{10}	0.314913	0.181398	1.736	2154	0.083
For EROSION slope, β_2					
INTRCPT2, γ_{20}	0.795626	0.130638	6.090	2154	<0.001

Fixed Effect	Coefficient	Odds Ratio	Confidence Interval
For INTRCPT1, β_0			

[1] Raudenbush, Yang, & Yousef (2000) results suggest 10 to 20. Capanu, Gönen, and Begg (2013) found no substantial differences between using 3, 5, or 7 quadrature points and that Laplace estimates did well (the PQL estimates of fixed effects performed the most poorly).

INTRCPT2, γ_{00}	-3.044702	0.047611	(0.035,0.064)
DROPOUT, γ_{01}	0.021796	1.022035	(0.875,1.193)
For RACE slope, β_1			
INTRCPT2, γ_{10}	0.314913	1.370140	(0.960,1.956)
For EROSION slope, β_2			
INTRCPT2, γ_{20}	0.795626	2.215827	(1.715,2.863)

The predicted probability can be computed from the results. I use the population average results to more accurately estimate the proportion of students who have carried a gun in the population. The following formula can be used assuming mean centering of the predictors (or if testing the intercept only model) and the proportion is desired for the case when all predictors equal their means:

$$\varphi = \frac{1}{\left(1 + e^{-\eta_{ij}}\right)}$$

Where η_{ij} is the predicted log odds given the regression, $\eta_{ij} = \beta_0 + \beta_1(DROPOUT) + \beta_2(RACE) + \beta_3(EROSION)$. η_{ij} is easy to calculate if all predictors are 0 (i.e., equal to their means when centered), because η_{ij} is simply equal to β_0:

$$\varphi = \frac{1}{\left(1 + e^{-(-3.045)}\right)} = .045$$

Thus, approximately 4.5% of students in the population are expected to report carrying a gun in the previous year.

R

Both adaptive quadrature and Laplace approximation (not higher-order Laplace) are available from R `lme4`, but random slopes are not allowed with adaptive quadrature. To save space, I only included the adaptive quadrature results. For Laplace estimates, just remove the `nAGC=` statement. Profile likelihood confidence intervals for the random effects (through the `lmerTest` `confint` function) are available whether Laplace or adaptive quadrature estimates are requested.

```
> library(lme4)
> mydata$gun <- as.numeric(mydata$gun)
> mydata$gun = mydata$gun - 1.
> mydata$race <- as.numeric(mydata$race)
```

NOTE: Use listwise deletion to make sure centering is based on same number of cases as used in the model

```
> mydata <-Subset(gun!='NA' & race!='NA'& erosion!='NA'& dropout!='NA')
>
> mydata$race <- mydata$race - mean(mydata$race)
> mydata$erosion <- mydata$erosion  - mean(mydata$erosion)
> mydata$dropout <- mydata$dropout - mean(mydata$dropout)
>
> rm(model2)
> #adaptive quadrature, 7 integration points (random slopes not possible with adapt quad in R)
> model2 <- glmer(gun ~ race + erosion + dropout + (1|schnum), family = binomial,
nAGQ=7,data=mydata)
> summary(model2)
Generalized linear mixed model fit by maximum likelihood (Adaptive Gauss-Hermite Quadrature, nAGQ
  = 7) [glmerMod]
 Family: binomial  ( logit )
Formula: gun ~ race + erosion + dropout + (1 | schnum)
   Data: mydata

     AIC      BIC   logLik deviance df.resid
   854.1    882.8   -422.1    844.1     2269

Scaled residuals:
    Min      1Q  Median      3Q     Max
-0.7677 -0.2359 -0.1872 -0.1584  6.9827
```

```
Random effects:
 Groups Name        Variance Std.Dev.
 schnum (Intercept) 0.291    0.5395
Number of obs: 2274, groups:  schnum, 28

Fixed effects:
            Estimate Std. Error z value             Pr(>|z|)
(Intercept) -3.12771    0.16124 -19.398 < 0.0000000000000002
race         0.38142    0.24082   1.584                0.113
erosion      0.81171    0.14077   5.766        0.00000000811
dropout      0.02875    0.05388   0.534                0.594

Correlation of Fixed Effects:
        (Intr) race   erosin
race    -0.073
erosion -0.290 -0.041
dropout  0.051 -0.054  0.025

> library(lmerTest)
> confint(model2)
Computing profile confidence intervals ...
                  2.5 %     97.5 %
.sig01       0.17025437  0.9813159
(Intercept) -3.48464102 -2.8146197
race        -0.10705263  0.8406401
erosion      0.53301192  1.0861359
dropout     -0.08676035  0.1371978
```

Note: Special code is needed (`0+slopevar`) to suppress multiple intercepts that are generated by default when there is more than one random effect (e.g., intercept and slope). This seems atypical to me so I modify the model to have only one intercept. For example,
`model1 <- glmer(gun ~ race + erosion + dropout + (race|schnum) + (0+erosion|schnum), family = binomial,data=mydata)`

SPSS

Because SPSS only provides PQL estimates, which I do not recommend using when other methods are available, I do not present a full example here. Below is syntax, however, for estimating a binary model.

```
GENLINMIXED
/DATA_STRUCTURE SUBJECTS=schnum
/FIELDS TARGET=gun
/TARGET_OPTIONS DISTRIBUTION=BINOMIAL LINK=LOGIT
/FIXED EFFECTS=race cerosion cdropout USE_INTERCEPT=TRUE
/RANDOM USE_INTERCEPT=TRUE SUBJECTS=schnum
COVARIANCE_TYPE=VARIANCE_COMPONENTS
/BUILD_OPTIONS TARGET_CATEGORY_ORDER=DESCENDING
INPUTS_CATEGORY_ORDER=DESCENDING
MAX_ITERATIONS=1500 CONFIDENCE_LEVEL=95 DF_METHOD=SATTERTHWAITE.
```

The "TARGET" is the outcome and the "INPUTS" are the predictors. The SUBJECTS variable is the group designation. ORDER=DESCENDING is used to specify that the 0 group is used as the comparison (typically what is desired) for the dependent or the independent variable. If omitted, the 1 group is used as the default.