Running MLwiN from within Stata: the \texttt{runmlwin} command

Modern Modeling Methods (M3) Conference
University of Connecticut
26\textsuperscript{th} May 2011

George Leckie and Chris Charlton
Centre for Multilevel Modelling
University of Bristol
INTRODUCTION
Existing multilevel modelling commands in Stata

• Stata provide the `xtmixed`, `xtmelogit` and `xtmepoisson` commands to fit multilevel models
  - Limited range of models can be specified
  - Computationally quite slow to fit models

• Sophia Rabe-Hesketh and Anders Skrondal provide the `gllamm` command
  - Wide range of models can be specified
  - Computationally slow to fit models

• Other user-written multilevel modelling commands include: `hlm`, `realcomimpute`, `runmplus`, `sabre`, `winbugs`
Multilevel modelling in MLwiN

1. Estimation of multilevel models for continuous, binary, ordered categorical, unordered categorical and count data

2. Constraints allowing models such as the social relations models and behavioural genetics models to be formulated as multilevel models

3. Fast estimation via classical and Bayesian methods

4. Estimation of multilevel models for cross-classified and multiple membership non-hierarchical data structures

5. Estimation of multilevel multivariate response models, multilevel spatial models, multilevel measurement error models, multilevel multiple imputation models and multilevel factor models
Examples

1. Two-level multilevel model
2. Growth curve models
3. Multilevel models for binary responses
4. Simulation studies are easy
5. MCMC estimation
6. Cross-classified models
7. Spatial multilevel models
8. Export models to WinBUGS
9. Work efficiently
10. Resources to help you learn runmlwin
1. TWO-LEVEL MULTILEVEL MODELS
Two-level variance components model

- Inner-London schools exam scores data set
- Classic MLwiN User Manual example
- First analysed by Goldstein et al. (1993)
- Reanalysed by Goldstein (2010), Rabe-Hesketh and Skrondal (2008), Rasbash et al. (2009) and others
- 4059 students nested within 65 schools
2-user 2-core Stata network perpetual license:
Serial number:  50110514919
Licensed to:  Centre for Multilevel Modelling
             University of Bristol

Notes:
1. (/m# option or -set memory-)  500.00 MB allocated to data
2. (/v# option or -set maxvar-)  5000 maximum variables

running C:\Program Files (x86)\Stata11\sysprofile.do ...

running C:\Users\gl9158\profile.do ...

.
use "http://www.bristol.ac.uk/cmm/media/runmlwin/tutorial.dta", clear
use "http://www.bristol.ac.uk/cmm/media/runmlwin/tutorial.dta"

2-user 2-core Stata network perpetual license:
Serial number: 50110514919
Licensed to: Centre for Multilevel Modelling
University of Bristol

Notes:
1. (/m# option or -set memory-) 500.00 MB allocated to data
2. (/v# option or -set maxvar-) 5000 maximum variables

running C:\Program Files (x86)\Stata11\sysprofile.do ...
running C:\Users\gl9158\profile.do ...
.use "http://www.bristol.ac.uk/cmm/media/runmlwin/tutorial.dta", clear
The `runmlwin` command syntax

\[ \text{normexam}_{ij} = \beta_0 + u_j + e_{ij} \]

\[ u_j \sim \text{N}(0, \sigma_u^2) \]

\[ e_{ij} \sim \text{N}(0, \sigma_e^2) \]

. runmlwin normexam cons, ///

    level2(school: cons) ///

    level1(student: cons)
use "http://www.bristol.ac.uk/cmm/media/runmlwin/tutorial.dta"

2-user 2-core Stata network perpetual license:
Serial number: 50110514919
Licensed to: Centre for Multilevel Modelling
University of Bristol

Notes:
1. (/m# option or -set memory-) 500.00 MB allocated to data
2. (/v# option or -set maxvar-) 5000 maximum variables

running C:\Program Files (x86)\Stata11\sysprofile.do ...
running C:\Users\g19158\profile.do ...
.runmlwin normexam cons, level2[school: cons] level1[student: cons]
\[ \text{normexam}_{ij} \sim N(XB, \Omega) \]
\[ \text{normexam}_{ij} = \beta_{0ij}\text{cons} \]
\[ \beta_{0ij} = \beta_0 + u_{0j} + e_{0ij} \]

\[
\begin{bmatrix}
  u_{0j} \\
  e_{0ij}
\end{bmatrix} \sim N(0, \Omega_u) : \Omega_u = \begin{bmatrix} \sigma_{u0}^2 \\ \sigma_{u0}^2 \end{bmatrix}
\]
\[
\begin{bmatrix}
  u_{0j} \\
  e_{0ij}
\end{bmatrix} \sim N(0, \Omega_e) : \Omega_e = \begin{bmatrix} \sigma_{e0}^2 \\ \sigma_{e0}^2 \end{bmatrix}
\]
normexam$ij \sim N(XB, \Omega)$

$\text{normexam}_{ij} = \beta_{0ij} \text{cons}$

$\beta_{0ij} = -0.013(0.054) + u_{0j} + e_{0ij}$

\[
\begin{bmatrix}
u_{0j}
\end{bmatrix} \sim N(0, \Omega_u) : \Omega_u = \begin{bmatrix} 0.169(0.032) \end{bmatrix}
\]

\[
\begin{bmatrix}
e_{0ij}
\end{bmatrix} \sim N(0, \Omega_e) : \Omega_e = \begin{bmatrix} 0.848(0.019) \end{bmatrix}
\]

$-2\times\log\text{likelihood(IGLS Deviance)} = 11010.648 (4059 \text{ of } 4059 \text{ cases in use})$
. use "http://www.bristol.ac.uk/cmm/media/runmlwin/tutorial.dta", clear
. runmlwin normexam cons, level2(school: cons) level1(student: cons)

MLwiN 2.23 multilevel model
Normal response model
Estimation algorithm: IGLS

Level Variable No. of Groups Minimum Average Maximum
school 65 2 62.4 198

Run time (seconds) = 12.93
Number of iterations = 3
Log likelihood = -5505.3242
Deviance = 11010.648

<table>
<thead>
<tr>
<th>Name</th>
<th>Label</th>
<th>Type</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>school</td>
<td>School ID</td>
<td>byte</td>
<td>%9.0g</td>
</tr>
<tr>
<td>student</td>
<td>Student ID</td>
<td>int</td>
<td>%9.0g</td>
</tr>
<tr>
<td>normexam</td>
<td>Age 16 exam score...</td>
<td>float</td>
<td>%9.0g</td>
</tr>
<tr>
<td>cons</td>
<td>Constant</td>
<td>byte</td>
<td>%9.0g</td>
</tr>
<tr>
<td>standrt</td>
<td>Age 11 exam score...</td>
<td>float</td>
<td>%9.0g</td>
</tr>
<tr>
<td>girl</td>
<td>Girl</td>
<td>byte</td>
<td>%9.0g</td>
</tr>
<tr>
<td>schgend</td>
<td>School gender</td>
<td>byte</td>
<td>%9.0g</td>
</tr>
<tr>
<td>avsirt</td>
<td>School average LR...</td>
<td>float</td>
<td>%9.0g</td>
</tr>
<tr>
<td>schav</td>
<td>School average LR...</td>
<td>byte</td>
<td>%9.0g</td>
</tr>
<tr>
<td>vrband</td>
<td>Age 11 verbal reason...</td>
<td>byte</td>
<td>%9.0g</td>
</tr>
</tbody>
</table>

| normexam | Coef. | Std. Err. | z   | P>|z| | [95% Conf. Interval] |
|----------|-------|-----------|-----|------|----------------------|
| cons     | -0.131668 | 0.0536254 | -0.25 | 0.806 | -0.1182706 - 0.091937 |

Random-effects Parameters

<table>
<thead>
<tr>
<th>Level 2:</th>
<th>Estimate</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>var(cons)</td>
<td>0.1686251</td>
<td>0.0324466</td>
<td>0.1050309 - 0.2322194</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 1:</th>
<th>Estimate</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>var(cons)</td>
<td>0.8477613</td>
<td>0.0189712</td>
<td>0.8105786 - 0.8849441</td>
</tr>
</tbody>
</table>
. use "http://www.bristol.ac.uk/cmm/media/runmlwin/tutorial.dta", clear

. runmlwin normexam cons, level2(school: cons) level1(student: cons)

MLwiN 2.23 multilevel model
Normal response model
Estimation algorithm: IGLS

Number of obs = 4059

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
<th>Average</th>
</tr>
</thead>
</table>
| school         | 65           | 2                      | 62.4    | 198

Run time (seconds) = 12.93
Number of iterations = 3
Log likelihood = -5505.3242
Deviance = 11010.648

| normexam | Coef.    | Std. Err. | z       | P>|z|   | [95% Conf. Interval] |
|----------|----------|-----------|---------|-------|---------------------|
| cons     | -0.0131668 | 0.0536254 | -0.25   | 0.806 | -0.1182706 to 0.091937 |

<table>
<thead>
<tr>
<th>Random-effects Parameters</th>
<th>Estimate</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 2:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(cons)</td>
<td>0.1686251</td>
<td>0.0324466</td>
<td>0.1050309 to 0.2322194</td>
</tr>
<tr>
<td>Level 1:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(cons)</td>
<td>0.8477613</td>
<td>0.0189712</td>
<td>0.8105786 to 0.8849441</td>
</tr>
</tbody>
</table>
Retrieve the level 2 residuals

\[ \text{normexam}_{ij} = \beta_0 + u_j + e_{ij} \]

\[ u_j \sim N(0, \sigma_u^2) \]

\[ e_{ij} \sim N(0, \sigma_e^2) \]

. runmlwin normexam cons, ///

        level2(school: cons, residuals(u)) ///

        level1(student: cons)
Do not pause in MLwiN

\[ \text{normexam}_{ij} = \beta_0 + u_j + e_{ij} \]

\[ u_j \sim \text{N}(0, \sigma_u^2) \]

\[ e_{ij} \sim \text{N}(0, \sigma_e^2) \]

. runmlwin normexam cons, ///

    level2(school: cons, residuals(u)) ///

    level1(student: cons) nopause
```
.runmlwin normexam cons, level2(school: cons, residuals(u)) level1(student: cons) nopause

MLwiN 2.23 multilevel model Number of obs = 4059
Normal response model Estimation algorithm: IGLS

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>school</td>
<td>65</td>
<td>2</td>
</tr>
</tbody>
</table>

Run time (seconds) = 1.47
Number of iterations = 3
Log likelihood = -5505.3242
Deviance = 11010.648

| normexam   | Coef.    | Std. Err. | z      | P>|z| | [95% Conf. Interval] |
|------------|----------|-----------|--------|--------|----------------------|
| cons       | -0.0131668 | 0.0536254 | -0.25  | 0.806  | -0.1182706 0.091937  |

<table>
<thead>
<tr>
<th>Random-effects Parameters</th>
<th>Estimate</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 2:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(cons)</td>
<td>0.1686251</td>
<td>0.0324466</td>
<td>0.1050309 0.2322194</td>
</tr>
<tr>
<td>Level 1:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(cons)</td>
<td>0.8477613</td>
<td>0.0189712</td>
<td>0.8105786 0.8849441</td>
</tr>
</tbody>
</table>
```

. egen u0rank = rank(u0)

. serrbar u0 u0se u0rank, scale(1.96) yline(0)
. summarize u0
.
. generate u0std = (u0 - r(mean))/r(sd)
.
. generate u0uniform = (u0rank - 0.5)/_N
.
. generate u0nscore = invnorm(u0uniform)
.
. scatter u0std u0nscore, yline(0) xline(0) ///
    ylabel(-3(1)3) xlabel(-3(1)3) aspectratio(1)
Add covariates

\[ \text{normexam}_{ij} = \beta_0 + \beta_1 \text{standlrt}_{ij} + \beta_2 \text{girl}_{ij} + u_j + e_{ij} \]

\[ u_j \sim N(0, \sigma_u^2) \]

\[ e_{ij} \sim N(0, \sigma_e^2) \]

. runmlwin normexam cons standlrt girl, ///
  level2(school: cons) ///
  level1(student: cons) nopause
Include a random slope

\[ \text{normexam}_{ij} = \beta_0 + \beta_1 \text{standlrt}_{ij} + \beta_2 \text{girl}_{ij} + u_{0j} + u_{1j} \text{standlrt}_{ij} + e_{ij} \]

\[
\begin{pmatrix}
    u_{0j} \\
    u_{1j}
\end{pmatrix} \sim \mathcal{N}\left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma^2_{u0} & \sigma^2_{u1} \\ \sigma^2_{u01} & \sigma^2_{u1} \end{pmatrix} \right)
\]

\[ e_{ij} \sim \mathcal{N}(0, \sigma^2_e) \]

. runmlwin normexam cons standlrt girl, ///

    level2(school: cons standlrt) ///

    level1(student: cons) nopause
Allow for level 1 heteroskedasticity

$$normexam_{ij} = \beta_0 + \beta_1 standlrt_{ij} + \beta_2 girl_{ij} + u_{0j} + u_{1j} standlrt_{ij}$$

$$+ e_{2ij} \text{girl}_{ij} + e_{3ij} \text{boy}_{ij}$$

$$\begin{pmatrix} u_{0j} \\ u_{1j} \end{pmatrix} \sim N \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma^2_{u0} & \sigma_{u01} \\ \sigma_{u01} & \sigma^2_{u1} \end{pmatrix} \right)$$

$$\begin{pmatrix} e_{2ij} \\ e_{3ij} \end{pmatrix} \sim N \left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma^2_{e2} & 0 \\ 0 & \sigma^2_{e3} \end{pmatrix} \right)$$

```
. runmlwin normexam cons standlrt girl, ///
  level2(school: cons standlrt) ///
  level1(student: girl boy, diagonal) nopause
```
MLwiN 2.23 multilevel model
Normal response model
Estimation algorithm: IGLS

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>school</td>
<td>65</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>62.4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>198</td>
</tr>
</tbody>
</table>

Run time (seconds) = 1.61
Number of iterations = 4
Log likelihood = -4640.71
Deviance = 9281.4199

| normexam  | Coef.    | Std. Err. | z     | P>|z|  | [95% Conf. Interval] |
|-----------|----------|-----------|-------|------|---------------------|
| cons      | -.111534 | .0433072  | -2.58 | 0.010 | -.1964145 to -.0266536 |
| standlrt  | .5529361 | .0200758  | 27.54 | 0.000 | .5135882 to .5922841  |
| girl      | .1752785 | .0324156  | 5.41  | 0.000 | .1117451 to .2388126  |

Random-effects Parameters

<table>
<thead>
<tr>
<th>Level 2:</th>
<th>Estimate</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>var(cons)</td>
<td>.0862511</td>
<td>.017175</td>
<td>.0525887 to .1199135</td>
</tr>
<tr>
<td>cov(cons,standlrt)</td>
<td>.0190537</td>
<td>.0066789</td>
<td>.0059632 to .0321441</td>
</tr>
<tr>
<td>var(standlrt)</td>
<td>.0148919</td>
<td>.0044702</td>
<td>.0061304 to .0236534</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 1:</th>
<th>Estimate</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>var(girl)</td>
<td>.5251641</td>
<td>.0152836</td>
<td>.4952088 to .5551194</td>
</tr>
<tr>
<td>var(boy)</td>
<td>.5874345</td>
<td>.0209983</td>
<td>.5462786 to .6285904</td>
</tr>
</tbody>
</table>
. test [RP1]var(girl) = [RP1]var(boy)

( 1)  [RP1]var(girl) - [RP1]var(boy) = 0

      chi2(  1) =  5.74
  Prob > chi2 =  0.0166
2. GROWTH CURVE MODELS
Child weight data

• Weight gain of Asian children in a British community

• 68 children, one to five measurements per child

• First analysed by Goldstein (1986)

• Re-analysed by Rabe-Hesketh and Skrondal (2008) and others
use "http://www.stata-press.com/data/mlmus2/asian.dta"
• graph twoway ///
  (connect weight age, connect(ascending)), ///
ytitle("Weight in Kg") xtitle("Age in years")
Growth curve model

\[ \text{weight}_{ij} = \beta_0 + \beta_1 \text{age}_{ij} + \beta_2 \text{age}_{ij}^2 + u_{0j} + u_{1j} \text{age}_{ij} + e_{ij} \]

\[
\begin{pmatrix}
  u_{0j} \\
  u_{1j}
\end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_{u0}^2 & \sigma_{u01}^2 \\ \sigma_{u01}^2 & \sigma_{u1}^2 \end{pmatrix}\right)
\]

\[ e_{ij} \sim \mathcal{N}(0, \sigma_e^2) \]

. runmlwin weight cons age age2, ///

   level2(id: cons age, residuals(u)) ///

   level1(occ: cons) nopause
MLwiN 2.23 multilevel model
Normal response model
Estimation algorithm: IGLS

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>68</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2.9</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5</td>
</tr>
</tbody>
</table>

Run time (seconds) = 1.62
Number of iterations = 7
Log likelihood = -258.07785
Deviance = 516.1557

| weight | Coef.   | Std. Err. | z      | P>|z| | [95% Conf. Interval] |
|--------|---------|-----------|--------|------|----------------------|
| cons   | 3.494518| 0.1372489 | 25.46  | 0.000| 3.225515 3.76352    |
| age    | 7.704002| 0.2394275 | 32.18  | 0.000| 7.234733 8.173271   |
| age2   | -1.660475| 0.0885319 | -18.76 | 0.000| -1.833994 -1.486955 |

Random-effects Parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Estimate</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 2:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(cons)</td>
<td>0.4040045</td>
<td>0.1412488</td>
<td>0.1271619 0.6808471</td>
</tr>
<tr>
<td>cov(cons,age)</td>
<td>0.088273</td>
<td>0.0812774</td>
<td>-0.0710279 0.2475738</td>
</tr>
<tr>
<td>var(age)</td>
<td>0.2539857</td>
<td>0.0858503</td>
<td>0.0857222 0.4222493</td>
</tr>
<tr>
<td>Level 1:</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(cons)</td>
<td>0.331641</td>
<td>0.0532307</td>
<td>0.2273107 0.4359712</td>
</tr>
</tbody>
</table>
• generate prediction = ///
  _b[cons]*cons + _b[age]*age + _b[age2]*age2 ///
  + u0 + u1*age

• line prediction age, connect(a) ///
  ytitle("Predicted weight in Kg")
• `label define genderlabel 1 "Boy" 2 "Girl"`

• `label values gender genderlabel`

• `graph twoway (line weight age, connect(ascending)), ///
  by(gender) ///
  xtitle("Age in years") ytitle("Weight in Kg")`
Growth curve model by gender

\[ \text{weight}_{ij} = \beta_0 \text{boy}_j + \beta_1 \text{boy}_j \times \text{age}_{ij} + \beta_2 \text{girl}_j + \beta_3 \text{girl}_j \times \text{age}_{ij} + \beta_4 \text{age}_{ij}^2 \]
+ \( u_0 \text{boy}_j + u_1 \text{boy}_j \times \text{age}_{ij} + u_2 \text{girl}_j + u_3 \text{girl}_j \times \text{age}_{ij} + e_{0ij} \text{boy}_j + e_{2ij} \text{girl}_j \)

\[
\begin{pmatrix}
  u_{0j} \\
  u_{1j} \\
  u_{2j} \\
  u_{3j}
\end{pmatrix} \sim N\left( \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_{u0}^2 & 0 & \sigma_{u2}^2 & \sigma_{u3}^2 \\ 0 & \sigma_{u1}^2 & 0 & 0 \\ \sigma_{u2}^2 & 0 & \sigma_{u23}^2 & 0 \\ \sigma_{u3}^2 & 0 & 0 & \sigma_{u3}^2 \end{pmatrix} \right)
\]

\[
\begin{pmatrix}
  e_{0ij} \\
  e_{2ij}
\end{pmatrix} \sim N\left( \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} \sigma_{e0}^2 & 0 \\ 0 & \sigma_{e2}^2 \end{pmatrix} \right)
\]

\[\text{. matrix a} = (1,1,1,0,0,1,0,0,1,1)\]

\[\text{. runmlwin weight boy boyXage girl girlXage age2, ///}
\]

\[\text{level2(id: boy boyXage girl girlXage, elements(a)) ///}
\]

\[\text{level1(occ: boy girl, diagonal) nopause} \]
### Level Variable

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
<th>Group Minimum</th>
<th>Group Average</th>
<th>Group Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>68</td>
<td></td>
<td>1</td>
<td>2.9</td>
<td>5</td>
</tr>
</tbody>
</table>

Run time (seconds) = 1.58  
Number of iterations = 7  
Log likelihood = -247.49434  
Deviance = 494.98868

### weight

|       | Coef.     | Std. Err. | z       | P>|z|   | [95% Conf. Interval] |
|-------|-----------|-----------|---------|-------|-----------------------|
| boy   | 3.78267   | .1563113  | 24.20   | 0.000 | 3.476305 - 4.089034   |
| boyXage| 7.728288  | .2567359  | 30.10   | 0.000 | 7.225095 - 8.231481   |
| girl  | 3.266411  | .1796806  | 18.18   | 0.000 | 2.914244 - 3.618579   |
| girlXage| 7.502467  | .2341932  | 32.04   | 0.000 | 7.043457 - 7.961477   |
| age2  | -1.624745 | .0849193  | -19.13  | 0.000 | -1.791184 - 1.458306  |

### Random-effects Parameters

<table>
<thead>
<tr>
<th>Random-effects Parameters</th>
<th>Estimate</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Level 2:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(boy)</td>
<td>.1553577</td>
<td>.1659469</td>
<td>-.1698922 - .4806076</td>
</tr>
<tr>
<td>cov(boy,boyXage)</td>
<td>.102065</td>
<td>.1232655</td>
<td>-.1395309 - .3436609</td>
</tr>
<tr>
<td>var(boyXage)</td>
<td>.3869624</td>
<td>.1692804</td>
<td>.05179 - .7187458</td>
</tr>
<tr>
<td>var(girl)</td>
<td>.5685636</td>
<td>.2111509</td>
<td>.1547155 - .9824117</td>
</tr>
<tr>
<td>cov(girl,girlXage)</td>
<td>.0161196</td>
<td>.0864426</td>
<td>-.1533048 - .185544</td>
</tr>
<tr>
<td>var(girlXage)</td>
<td>.0799457</td>
<td>.0608557</td>
<td>-.0393292 - .1992206</td>
</tr>
<tr>
<td><strong>Level 1:</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(boy)</td>
<td>.4182827</td>
<td>.0929099</td>
<td>.2361826 - .6003828</td>
</tr>
<tr>
<td>var(girl)</td>
<td>.2429176</td>
<td>.0555108</td>
<td>.134183 - .3517168</td>
</tr>
</tbody>
</table>
3. MULTILEVEL MODELS FOR BINARY RESPONSES
Guatemalan immunization campaign

- Child immunization data

- 2159 children within 1595 mothers within 161 communities

- First analysed by Pebley, Goldman and Rodriguez (1996) and Rodriguez and Goldman (2001)

- Reanalysed by Rabe-Hesketh and Skrondal (2008) and others
Three-level binary response model

\[ \text{immun}_{ijk} \sim \text{Binomial}(1, \pi_{ijk}) \]

\[
\logit(\pi_{ijk}) = \beta_0 + \beta_1 \text{kid2p}_{ijk} + \beta_2 \text{rural}_k + \beta_3 \text{pcInd81}_k + v_k + u_{jk}
\]

\[ v_k \sim \text{N}(0, \sigma^2_u) \]

\[ u_{jk} \sim \text{N}(0, \sigma^2_u) \]

. runmlwin immun cons kid2p rural pcInd81, ///
  level3(cluster: cons) ///
  level2(mom: cons) ///
  level1(kid:) ///
  discrete(dist(binomial) link(logit) denom(cons)) ///
  nopause
MLwiN 2.23 multilevel model
Binomial logit response model
Estimation algorithm: IGLS, MQL1

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Minimum</td>
<td>Average</td>
<td>Maximum</td>
<td></td>
</tr>
<tr>
<td>cluster mom</td>
<td>161</td>
<td>1</td>
<td>13.4</td>
<td>55</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1595</td>
<td>1</td>
<td>1.4</td>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

Run time (seconds) = 2.89
Number of iterations = 5

| immun   | Coef.     | Std. Err. | z     | P>|z| | [95% Conf. Interval] |
|---------|-----------|-----------|-------|------|----------------------|
| cons    | -.1433676 | .1721252  | -0.83 | 0.405| -.4807268            |
|         |           |           |       |      | .1939915             |
| kid2p   | .9173057  | .1179051  | 7.78  | 0.000| .6862159             |
|         |           |           |       |      | 1.148395             |
| rural   | -.5668908 | .1480174  | 7.78  | 0.000| -.8569995            |
|         |           |           |       |      | -.276782             |
| pcInd81 | -.8460267 | .1797028  | -4.71 | 0.000| -1.198238            |
|         |           |           |       |      | -.4938157            |

Random-effects Parameters
<table>
<thead>
<tr>
<th>Estimate</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 3:</td>
<td>var(coeffs)</td>
<td>.2960818</td>
</tr>
<tr>
<td>Level 2:</td>
<td>var(coeffs)</td>
<td>.3674519</td>
</tr>
</tbody>
</table>
Refit the model using PQL2

\[
\text{immunized}_{ijk} \sim \text{Binomial}(1, \pi_{ijk})
\]

\[
\text{logit}(\pi_{ijk}) = \beta_0 + \beta_1 \text{kid2p}_{ijk} + \beta_2 \text{rural}_k + \beta_3 \text{pcInd81}_k + v_k + u_{jk}
\]

\[
v_k \sim N(0, \sigma_v^2)
\]

\[
u_{jk} \sim N(0, \sigma_u^2)
\]

. runmlwin immun cons kid2p rural pcInd81, ///
  level3(cluster: cons) ///
  level2(mom: cons) ///
  level1(kid:) ///
  discrete(d(binomial) l(logit) de(cons) pql2) ///
  initsprevious maxiterations(40) nopause
Model fitted using initial values specified as parameter estimates from previous model

MLwiN 2.23 multilevel model
Binomial logit response model
Estimation algorithm: IGLS, PQL2

Number of obs = 2159

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
<th>Group Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster</td>
<td>161</td>
<td>1</td>
<td>55</td>
</tr>
<tr>
<td>mom</td>
<td>1595</td>
<td>1</td>
<td>3</td>
</tr>
</tbody>
</table>

Run time (seconds) = 8.71
Number of iterations = 29

|        | Coef.     | Std. Err. | z      | P>|z| | [95% Conf. Interval] |
|--------|-----------|-----------|--------|------|---------------------|
| cons   | -0.189721 | 0.2437707 | -0.78  | 0.436 | -0.6675029 0.2880607 |
| kid2p  | 1.363391  | 0.1542231 | 8.84   | 0.000 | 1.061119  1.665663  |
| rural  | -0.850848 | 0.2157692 | -3.94  | 0.000 | -1.273585 -0.427785 |
| pcInd81| -1.313231 | 0.2638969 | -4.98  | 0.000 | -1.83046 -0.7960028 |

Random-effects Parameters

<table>
<thead>
<tr>
<th></th>
<th>Estimate</th>
<th>Std. Err.</th>
<th>[95% Conf. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 3:</td>
<td>var(cons)</td>
<td>0.6723186</td>
<td>0.1658299 0.3472979 0.9973393</td>
</tr>
<tr>
<td>Level 2:</td>
<td>var(cons)</td>
<td>2.352684</td>
<td>0.2589447 1.845162  2.860206</td>
</tr>
</tbody>
</table>
4. SIMULATION STUDIES ARE EASY
set seed 12345
postfile MQL1 ix fx cx sigmaf sigmac using "MQL1.dta", replace
set obs 2
generate cx = _n - 1
sort cx
expand 10
generate cid = _n
expand 2
bysort cid: gen fx = _n - 1
expand 10
bysort cid (fx): generate fid = _n
expand 2
bysort cid fid: gen ix = _n - 1
expand 10
bysort cid fid (ix): gen iid = _n
generate cons = 1
forvalues iteration = 1/10 {
    display _n(5) as txt "Iteration " as res " iteration'" as txt " of " as res " 100"
generate c = rnormal(0,1)
bysort cid (fid iid): replace c = c[1]
generate f = rnormal(0,1)
bysort cid fid (iid): replace f = f[1]
generate y = rbinomial(1, invlogit(0*cons + 1*ix + 1*fx + 1*cx + f + c))
runmlwin y cons ix fx cx, level3(cid: cons) level2(fid: cons) level1(iid:)
    discrete(distribution(binomial) link(logit) denominator(cons))
    nopause
post MQL1 ([FP1]ix) ([FP1]fx) ([FP1]cx) (sqrt([RP2]var(cons))) (sqrt([RP3]var(cons)))
    c f y
}
postclose MQL1
use "MQL1.dta", clear
tabstat ix fx cx sigmaf sigmac, format(%3.2f)
5. MCMC ESTIMATION
Refit the model using MCMC

\[ \text{immun}_{ijk} \sim \text{Binomial}(1, \pi_{ijk}) \]

\[ \logit(\pi_{ijk}) = \beta_0 + \beta_1 \text{kid2p}_{ijk} + \beta_2 \text{rural}_k + \beta_3 \text{pcInd81}_k + \nu_k + u_{jk} \]

\[ \nu_k \sim \text{N}(0, \sigma_u^2) \]

\[ u_{jk} \sim \text{N}(0, \sigma_u^2) \]

. runmlwin immun cons kid2p rural pcInd81, ///
  level3(cluster: cons) ///
  level2(mom: cons) ///
  level1(kid:) ///
  discrete(d(binomial) l(logit) de(cons) pql2) ///
  mcmc(on) initsprevious nopause
### Level Variable

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Minimum</td>
<td>Average</td>
</tr>
<tr>
<td>cluster</td>
<td>161</td>
<td>1</td>
</tr>
<tr>
<td>mom</td>
<td>1595</td>
<td>1</td>
</tr>
</tbody>
</table>

**Burnin** = 500  
**Chain** = 5000  
**Run time (seconds)** = 30  
**Deviance (dbar)** = 1619.22  
**Deviance (thetabar)** = 866.88  
**Effective no. of pars (pd)** = 752.34  
**Bayesian DIC** = 2371.56

### immun

<table>
<thead>
<tr>
<th>Variable</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>z</th>
<th>ESS</th>
<th>[95% Cred. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>cons</td>
<td>-.2200957</td>
<td>.3358961</td>
<td>-.66</td>
<td>42</td>
<td>-.889235 to .5050995</td>
</tr>
<tr>
<td>kid2p</td>
<td>1.754326</td>
<td>.230542</td>
<td>7.63</td>
<td>53</td>
<td>1.3257 to 2.208608</td>
</tr>
<tr>
<td>rural</td>
<td>-.1145384</td>
<td>.2759653</td>
<td>-4.15</td>
<td>56</td>
<td>-1.6628 to -.6193233</td>
</tr>
<tr>
<td>pcInd81</td>
<td>-1.709476</td>
<td>.3672927</td>
<td>-4.65</td>
<td>61</td>
<td>-2.520294 to -1.065675</td>
</tr>
</tbody>
</table>

### Random-effects Parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>ESS</th>
<th>[95% Cred. Int]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 3:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(cons)</td>
<td>1.161717</td>
<td>.3641234</td>
<td>81</td>
<td>.6004681 to 2.046271</td>
</tr>
<tr>
<td>Level 2:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(cons)</td>
<td>5.934905</td>
<td>1.221375</td>
<td>21</td>
<td>3.805253 to 8.552722</td>
</tr>
</tbody>
</table>
mcmcsum, trajectories
. mcmcsunm [RP2]var(cons), fiveplot
```
.mcmcsun [RP2]var(standlret)

<table>
<thead>
<tr>
<th>Percentiles</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>.0862781</td>
</tr>
<tr>
<td>MCSE of Mean</td>
<td>.0024099</td>
</tr>
<tr>
<td>Std. Dev.</td>
<td>.0467082</td>
</tr>
<tr>
<td>Mode</td>
<td>.0631075</td>
</tr>
<tr>
<td>0.5%</td>
<td>.0193246</td>
</tr>
<tr>
<td>2.5%</td>
<td>.0243268</td>
</tr>
<tr>
<td>5%</td>
<td>.0298808</td>
</tr>
<tr>
<td>25%</td>
<td>.0520173</td>
</tr>
<tr>
<td>50%</td>
<td>.0765091</td>
</tr>
<tr>
<td>75%</td>
<td>.1100566</td>
</tr>
<tr>
<td>95%</td>
<td>.179421</td>
</tr>
<tr>
<td>97.5%</td>
<td>.2023509</td>
</tr>
<tr>
<td>99.5%</td>
<td>.2549108</td>
</tr>
</tbody>
</table>

Thinned Chain Length: 5000
Effective Sample Size: 99
Raftery Lewis (2.5%): 25770
Raftery Lewis (97.5%): 23976
Brooks Draper (mean): 446390
```
Run the model for longer

\[ immunized_{ijk} \sim \text{Binomial}(1, \pi_{ijk}) \]

\[ \logit(\pi_{ijk}) = \beta_0 + \beta_1 kid2p_{ijk} + \beta_2 rural_k + \beta_3 pcInd81_k + v_k + u_{jk} \]

\[ v_k \sim \text{N}(0, \sigma_{v}^2) \]

\[ u_{jk} \sim \text{N}(0, \sigma_{u}^2) \]

. runmlwin immunized cons kid2p rural pcInd81, ///

  level3(cluster: cons) ///

  level2(mom: cons) ///

  level1(kid:) ///

  discrete(d(binomial) l(logit) de(cons) pql2) ///

  mcmc(burnin(5000) chain(50000) thinning(10)) ///

  initsprevious nopause
### Level Variable Summary

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
<th>Group Minimum</th>
<th>Group Average</th>
<th>Group Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>cluster</td>
<td>161</td>
<td>1</td>
<td>1</td>
<td>13.4</td>
<td>55</td>
</tr>
<tr>
<td>mom</td>
<td>1595</td>
<td>1</td>
<td>1</td>
<td>1.4</td>
<td>3</td>
</tr>
</tbody>
</table>

### Burnin and Chain Information

- **Burnin** = 5000
- **Chain** = 50000
- **Run time (seconds)** = 257
- **Deviance (d-bar)** = 1641.35
- **Deviance (theta-bar)** = 895.18
- **Effective no. of pars (pd)** = 746.17
- **Bayesian DIC** = 2387.51

### Immunization Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>z</th>
<th>ESS</th>
<th>[95% Cred. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>cons</td>
<td>-0.242</td>
<td>0.308</td>
<td>-0.79</td>
<td>424</td>
<td>-0.8276453, 0.3486197</td>
</tr>
<tr>
<td>kid2p</td>
<td>1.730</td>
<td>0.218</td>
<td>7.92</td>
<td>434</td>
<td>1.335295, 2.19006</td>
</tr>
<tr>
<td>rural</td>
<td>-1.089</td>
<td>0.295</td>
<td>-3.69</td>
<td>471</td>
<td>-1.687306, -0.5090104</td>
</tr>
<tr>
<td>pcInd81</td>
<td>-1.682</td>
<td>0.369</td>
<td>-4.56</td>
<td>587</td>
<td>-2.450633, -0.9707532</td>
</tr>
</tbody>
</table>

### Random-effects Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>ESS</th>
<th>[95% Cred. Int]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level 3:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(cons)</td>
<td>1.128</td>
<td>0.358</td>
<td>572</td>
<td>0.5575184, 1.94313</td>
</tr>
<tr>
<td>Level 2:</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>var(cons)</td>
<td>5.628</td>
<td>1.258</td>
<td>186</td>
<td>3.587712, 8.415654</td>
</tr>
</tbody>
</table>
6. CROSS-CLASSIFIED MODELS
Scottish neighbourhood study on child educational attainment

- Scottish neighbourhood study on child educational attainment
- 2310 students nested within 17 schools and 524 neighbourhoods
- First analysed by Garner and Raudenbush (1991)
- Re-analysed by Rabe-Hesketh and Skrondal (2008), Raudenbush (1993), Raudenbush and Bryk (2002) and others
. table neighid schid if inrange(neighid,26,38) | inrange(neighid,251,263) | inrange(neighid > ,793,800)

<table>
<thead>
<tr>
<th>neighid</th>
<th>2</th>
<th>8</th>
<th>10</th>
<th>15</th>
<th>16</th>
<th>17</th>
<th>18</th>
<th>19</th>
<th>20</th>
</tr>
</thead>
<tbody>
<tr>
<td>26</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>251</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>252</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>253</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>256</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>258</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>259</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>260</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>261</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>262</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>263</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>793</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>794</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>795</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>796</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>797</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>798</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>799</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Two-way cross-classified model

\[ \text{attain}_i = \beta_0 + u_{\text{schid}(i)}^{(3)} + u_{\text{neighid}(i)}^{(2)} + e_i \]

\[ u_{j}^{(3)} \sim \text{N}(0, \sigma_{u(3)}^2), \quad u_{j}^{(2)} \sim \text{N}(0, \sigma_{u(2)}^2), \quad e_i \sim \text{N}(0, \sigma_e^2) \]

. matrix b = (0,.33,.33,.33)

. runmlwin attain cons, ///

    level3(schid: cons) ///

    level2(neighid: cons) ///

    level1(studentid: cons) ///

    mcmc(cc) initsb(b)
. matrix b = (0,.33,.33,.33)

. runmlwin attain cons, level3(schid: cons) level2(neighid: cons) level1(student > id: cons) mcmc(cc) initb(b) nopause

MLwiN 2.23 multilevel model
Normal response model
Estimation algorithm: MCMC

<table>
<thead>
<tr>
<th>Level</th>
<th>No. of Groups</th>
<th>Minimum</th>
<th>Average</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>schid</td>
<td>17</td>
<td>22</td>
<td>135.9</td>
<td>286</td>
</tr>
<tr>
<td>neighid</td>
<td>524</td>
<td>1</td>
<td>4.4</td>
<td>16</td>
</tr>
</tbody>
</table>

Burnin = 500
Chain = 5000
Run time (seconds) = 6.88
Deviance (dbar) = 6039.42
Deviance (thetabar) = 5818.77
Effective no. of pars (pd) = 220.65
Bayesian DIC = 6260.07

<table>
<thead>
<tr>
<th>Variable</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>z</th>
<th>ESS</th>
<th>[95% Cred. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>attain</td>
<td>0.096258</td>
<td>0.0651659</td>
<td>1.48</td>
<td>228</td>
<td>-0.0349854 to 0.2170355</td>
</tr>
<tr>
<td>cons</td>
<td>0.0995634</td>
<td>0.0492998</td>
<td>2175</td>
<td>0.0387661 to 0.2261923</td>
<td></td>
</tr>
<tr>
<td>Level 3:</td>
<td>var(cons)</td>
<td>0.1422148</td>
<td>0.0217524</td>
<td>458</td>
<td>0.1033897 to 0.1892418</td>
</tr>
<tr>
<td>Level 2:</td>
<td>var(cons)</td>
<td>0.8002955</td>
<td>0.0260423</td>
<td>2583</td>
<td>0.7513126 to 0.8540478</td>
</tr>
</tbody>
</table>
Two-way cross-classified model

\[ \text{attain}_i = \beta_0 + \beta_1 p7vrq_i + \beta_2 p7read_i + \beta_3 \text{dadocc}_i + \beta_4 \text{daded}_i + \beta_5 \text{momed}_i + \beta_6 \text{dadunemp}_i + \beta_7 \text{male}_i + \beta_8 \text{deprive}_i + u^{(3)}_{\text{schid}(i)} + u^{(2)}_{\text{neighid}(i)} + e_i \]

\[ u^{(3)}_j \sim N(0, \sigma^2_u(3)), \quad u^{(2)}_j \sim N(0, \sigma^2_u(2)), \quad e_i \sim N(0, \sigma^2_e) \]

. matrix b = (0,0,0,0,0,0,0,0,0,1,1,1)

. runmlwin attain cons p7vrq p7read dadocc daded ///
   momed dadunemp male deprive, ///
   level3(schid: cons) ///
   level2(neighid: cons) ///
   level1(studentid: cons) mcmc(cc) initsb(b)
runmlwin attain cons p7vrq p7read ///
> dadocc daded momed dadunemp female deprive, ///
> level3(schid: cons) ///
> level2(neighid: cons) ///
> level1(studentid: cons) ///
> mcmc(cc) initsb(b) ///
> no pause

MLwiN 2.23 multilevel model
Normal response model
Estimation algorithm: MCMC

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>schid</td>
<td>17</td>
<td>22</td>
</tr>
<tr>
<td>neighid</td>
<td>524</td>
<td>1</td>
</tr>
</tbody>
</table>

Number of obs = 2310

Burnin = 500
Chain = 5000
Run time (seconds) = 26.6
Deviance (dbar) = 4744.77
Deviance (thetabar) = 4704.11
Effective no. of pars (pd) = 40.67
Bayesian DIC = 4785.44

<table>
<thead>
<tr>
<th>attain</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>z</th>
<th>ESS</th>
<th>[95% Cred. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>cons</td>
<td>0.0351255</td>
<td>0.0291963</td>
<td>1.20</td>
<td>1706</td>
<td>-0.0220368 - 0.0938575</td>
</tr>
<tr>
<td>p7vrq</td>
<td>0.0275779</td>
<td>0.0222758</td>
<td>12.12</td>
<td>4556</td>
<td>0.0231709 - 0.03204</td>
</tr>
<tr>
<td>p7read</td>
<td>0.0262253</td>
<td>0.017897</td>
<td>14.65</td>
<td>4689</td>
<td>0.0226729 - 0.0297626</td>
</tr>
<tr>
<td>dadocc</td>
<td>0.0080741</td>
<td>0.013761</td>
<td>5.87</td>
<td>6480</td>
<td>0.0053839 - 0.0107416</td>
</tr>
<tr>
<td>daded</td>
<td>0.142757</td>
<td>0.0411453</td>
<td>3.47</td>
<td>5452</td>
<td>0.0615814 - 0.2230719</td>
</tr>
<tr>
<td>momed</td>
<td>0.0605109</td>
<td>0.0379741</td>
<td>1.59</td>
<td>4703</td>
<td>-0.013253 - 0.1342922</td>
</tr>
<tr>
<td>dadunemp</td>
<td>-0.1224487</td>
<td>0.0468065</td>
<td>-2.62</td>
<td>4505</td>
<td>-0.2130983 - 0.028468</td>
</tr>
<tr>
<td>female</td>
<td>0.0558048</td>
<td>0.0280615</td>
<td>1.99</td>
<td>4699</td>
<td>0.0015048 - 0.1117451</td>
</tr>
<tr>
<td>deprive</td>
<td>-0.1562503</td>
<td>0.0260965</td>
<td>-5.99</td>
<td>3705</td>
<td>-0.207776 - 0.1055711</td>
</tr>
</tbody>
</table>
7. SPATIAL MULTILEVEL MODELS
Scottish lip cancer

- County level lip cancer counts between 1975 and 1980
- 56 Scottish counties
- First analysed by Clayton and Kaldor (1987)
- Re-analysed by Breslow and Clayton (1993), Leyland and Goldstein (2001), Rabe-Hesketh and Skrondal (2008) and others
use "http://www.bristol.ac.uk/cmm/media/runmlwin/lips1.dta"

Notes:
1. (/m# option or -set memory-) 500.00 MB allocated to data
2. (/v# option or -set maxvar-) 5000 maximum variables

running C:\Program Files (x86)\Stata11\sysprofile.do ...
running C:\Users\g9158\profile.do ...

. use "http://www.bristol.ac.uk/cmm/media/runmlwin/lips1.dta", clear
• use "http://www.bristol.ac.uk/cmm/media/runmlwin/lips1.dta", clear

• merge 1:1 area using "scotdb.dta"

• spmap obs using "scotcoord.dta", id(area) ///
  fcolor(Blues) legend(position(10)) ///
  clmethod(custom) clbreaks(0 5 10 15 20 40)
Over-dispersed Poisson model

\[ \text{obs}_i \sim \text{Poisson}(\pi_i) \]

\[ \log(\pi_i) = \text{offs}_i + \beta_0 + \beta_1 \text{perc}_aff_i + u_i \]

\[ u_i \sim \text{N}(0, \sigma_u^2) \]

runmlwin obs cons perc_aff, ///

level2(area: cons) ///

level1(area:) ///

discrete(dist(poisson) link(log) offset(offs)) ///

mcmc(chain(50000)) ///

initsprevious nopause
. run mlwin obs cons perc_aff, level2(area:cons) level1(area:) discrete(distribut
> ion(poisson) link(log) offset(offs)) mcmc(chain(50000) refresh(500)) init
> sprev ious nopause

MLwiN 2.23 multilevel model
Poisson response model
Estimation algorithm: MCMC, MQL1

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>area</td>
<td>56</td>
<td>1.0</td>
</tr>
</tbody>
</table>

Burnin = 500
Chain = 500000
Run time (seconds) = 10.3
Deviance (d) = 270.38
Deviance (theta) = 230.65
Effective no. of pars (pd) = 39.73
Bayesian DIC = 310.11

<table>
<thead>
<tr>
<th>obs</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>z</th>
<th>ESS</th>
<th>[95% Cred. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>cons</td>
<td>-.4835851</td>
<td>.1628447</td>
<td>-2.97</td>
<td>494</td>
<td>-.8143693 -.1711187</td>
</tr>
<tr>
<td>perc_aff</td>
<td>.0675449</td>
<td>.0143747</td>
<td>4.70</td>
<td>475</td>
<td>.0388835 .0956604</td>
</tr>
</tbody>
</table>

Random-effects Parameters

<table>
<thead>
<tr>
<th>Level 2:</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>ESS</th>
<th>[95% Cred. Int]</th>
</tr>
</thead>
<tbody>
<tr>
<td>var(cons)</td>
<td>.3852672</td>
<td>.1125743</td>
<td>6787</td>
<td>.2107703 .645173</td>
</tr>
</tbody>
</table>
CAR model

\[ \text{obs}_i \sim \text{Poisson}(\pi_i) \]

\[ \log(\pi_i) = \text{offs}_i + \beta_0 + \beta_1 \text{perc}_i + u_i \]

\[ u_i \sim N\left(\bar{u}_i, \frac{\sigma_u^2}{r_i}\right), \quad \bar{u}_i = \sum_{j \in \text{neigh}(i)} \frac{w_{i,j} u_i}{r_i} \]

. runmlwin obs perc_aff, ///

level2(area: cons, carids(neigh1-neigh11) ///
carweights(wcar1-wcar11)) ///

level1(cons:) ///

discrete(dist(poisson) link(log) offset(offs)) ///
mcmc(chain(50000)) initsp nopause
. runmlwin obs perc_aff, level2(area: cons, carids(neigh1-neigh11) carweights(wcar1-wcar11)) level1(cons:) discrete(distribution(poisson) link(log) offset(offs)) mcmc(chain(50000) re fresh(500)) inits previous nopause

MLwiN 2.23 multilevel model
Poisson response model
Estimation algorithm: MCMC, MQL1

<table>
<thead>
<tr>
<th>Level Variable</th>
<th>No. of Groups</th>
<th>Observations per Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>area</td>
<td>56</td>
<td>1</td>
</tr>
</tbody>
</table>

Burnin = 500
Chain = 50000
Run time (seconds) = 9.67
Deviance (dbar) = 268.77
Deviance (thetabar) = 240.42
Effective no. of pars (pd) = 28.35
Bayesian DIC = 297.13

<table>
<thead>
<tr>
<th>obs</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>z</th>
<th>ESS</th>
<th>[95% Cred. Interval]</th>
</tr>
</thead>
<tbody>
<tr>
<td>perc_aff</td>
<td>.035667</td>
<td>.0128288</td>
<td>2.78</td>
<td>354</td>
<td>.0090298 .0591634</td>
</tr>
</tbody>
</table>

Random-effects Parameters

<table>
<thead>
<tr>
<th>Level 2:</th>
<th>Mean</th>
<th>Std. Dev.</th>
<th>ESS</th>
<th>[95% Cred. Int]</th>
</tr>
</thead>
<tbody>
<tr>
<td>var(cons)</td>
<td>.5337886</td>
<td>.1900985</td>
<td>3602</td>
<td>.2512767 .9866003</td>
</tr>
</tbody>
</table>
Convolution model

\[ \text{obs}_i \sim \text{Poisson}(\pi_i) \]
\[ \log(\pi_i) = \text{offs}_i + \beta_0 + \beta_1 \text{perc}_\text{aff}_i + v_i + u_i \]
\[ v_i \sim \mathcal{N}\left(\bar{v}_i, \frac{\sigma_v^2}{r_i}\right), \quad \bar{v}_i = \sum_{j \in \text{neigh}(i)} \frac{w_{i,j}v_i}{r_i} \]
\[ u_i \sim \mathcal{N}(0, \sigma_u^2) \]

. runmlwin observed cons perc_aff, ///

level3(area: cons, carids(neigh1-neigh11) ///
carweights(wcar1-wcar11)) ///
level2(area: cons) level1(county:) ///
discrete(d(binomial) l(log) offset(offs)) ///
mcmc(on) initsprevius nopause
8. EXPORT MODELS TO WINBUGS
The runmlwin command syntax

\[ \text{normexam}_{ij} = \beta_0 + \beta_1 \text{standlrt}_{ij} + u_j + e_{ij} \]

\[ u_j \sim N(0, \sigma_u^2) \]

\[ e_{ij} \sim N(0, \sigma_e^2) \]

. runmlwin normexam cons standlrt, ///

  level2(school: cons) ///

  level1(student: cons) ///

  mcmc(savewinbugs(model(m.txt) inits(i.txt) ///

  data(d.txt) nofit)) ///

  initsprevious nopause
# WINBUGS 1.4 code generated from MLwiN program

#----MODEL Definition--------------

model
{
  # Level 1 definition
  for(i in 1:N) {
    normexam[i] ~ dnorm(mu[i], tau)
  }
  # Higher level definitions
  for (j in 1:n2) {
    u2[j] ~ dnorm(0, tau.u2)
  }
  # Priors for fixed effects
  for (k in 1:2) { beta[k] ~ dflat() }
  # Priors for random terms
  tau ~ dgamma(0.001000, 0.001000)
  sigma2 <- 1/tau
  tau.u2 ~ dgamma(0.001000, 0.001000)
  sigma2.u2 <- 1/tau.u2
}
t-distributed level 2 residuals

\[ \text{normexam}_{ij} = \beta_0 + \beta_1 \text{standlrt}_{ij} + u_j + e_{ij} \]

\[ u_j \sim t(0, \sigma_u^2, \nu) \]

\[ e_{ij} \sim N(0, \sigma_e^2) \]
# WINBUGS 1.4 code generated from MLwiN program

#----MODEL Definition---------------------

model
{
  # Level 1 definition
  for(i in 1:N)
  {normexam[i] ~ dnorm(mu[i], tau)
  }
  # Higher level definitions
  for (j in 1:n2)
  {u2[j] ~ dt(0, tau.u2, df)
  }
  # Priors for fixed effects
  for (k in 1:2)
  {beta[k] ~ dflat()
  }
  # Priors for random terms
  tau ~ dgamma(0.001000, 0.001000)
sigma2 <- 1/tau
tau.u2 ~ dgamma(0.001000, 0.001000)
sigma2.u2 <- 1/tau.u2
df ~ dunif(2, 200)
}
The **winbugs** suite of commands

```
. wbscript , ///
   model("\`c(pwd)\'\m.txt") inits("\`c(pwd)\'\i.txt") ///
   data("\`c(pwd)\'\d.txt") coda("\`c(pwd)\'\out") ///
   set(df) burn(500) update(5000) ///
   saving("\`c(pwd)\'\script.txt", replace) quit

. wbrun, script("\`c(pwd)\'\script.txt") ///
   winbugs("C:\Users\gl9158\WinBUGS14\winbugs14.exe")

. wbcoda, root("\`c(pwd)\'\out") clear
```
Licence Agreement

Introduction

This software and any associated documentation whether electronic or physical (hereinafter called "WinBUGS PACKAGE") is made available under a licence agreement and may be used only in accordance with the terms of that agreement. This is a legal agreement between you (the Licensee), and MRC and Imperial College of Science, Technology and Medicine (the Licensor). The terms are provided in the following pages.

Users are required to register and to pay a fee for the use of the WinBUGS PACKAGE. Details of fees and the procedure for registration and acceptance of the licence terms is provided here. There is no fee payable for the use of the demonstration (Internet) version of the WinBUGS package. Users of the demonstration version of the WinBUGS package can upgrade to the full version on payment of a fee.

The current fee is zero dollars ($0).

By completing and sending the registration you demonstrate your agreement to the terms of this licence and will become legally bound to the terms there.

It should be emphasised that the statistical tools provided in the WinBUGS PACKAGE are by their very nature partly subjective. The Licencee should offer advice on interpretation of results obtained using the WinBUGS PACKAGE. Any assistance will be strictly limited to attempting to help if there are problems with the software itself. The Licencee is not responsible for the results obtained using the licence.

```plaintext
display(log)
check(Q:/C-modelling/runmlwin/presentations/2011-05-26
Connecticut/m_modified.txt)
model is syntactically correct
data(Q:/C-modelling/runmlwin/presentations/2011-05-26
Connecticut/d.txt)
data loaded
compile(1)
model compiled
inits(1,Q:/C-modelling/runmlwin/presentations/2011-05-26
Connecticut/i_modified.txt)
model is initialized
gen.inits()
command #Bugs:gen.inits cannot be executed (is greyed out)
update(500)
set(df)
update(5000)
coda(*,Q:/C-modelling/runmlwin/presentations/2011-05-26
Connecticut/out)
```
. mcmcsun df, trajectories variables
```
.mcmcsum df, variables

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>Percentiles</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>df</td>
<td>Percentiles</td>
<td></td>
<td>Thinned Chain Length</td>
<td></td>
</tr>
<tr>
<td>Mean</td>
<td>100.388</td>
<td>0.5%</td>
<td>3.102855</td>
<td></td>
<td>5000</td>
</tr>
<tr>
<td>MCSE of Mean</td>
<td>4.172605</td>
<td>2.5%</td>
<td>7.934025</td>
<td>Effective Sample Size</td>
<td>107</td>
</tr>
<tr>
<td>Std. Dev.</td>
<td>57.77448</td>
<td>5%</td>
<td>11.977</td>
<td>Raftery Lewis (2.5%)</td>
<td>57207</td>
</tr>
<tr>
<td>Mode</td>
<td>21.98396</td>
<td>25%</td>
<td>49.3425</td>
<td>Raftery Lewis (97.5%)</td>
<td>5464</td>
</tr>
<tr>
<td></td>
<td></td>
<td>50%</td>
<td>102.25</td>
<td>Brooks Draper (mean)</td>
<td>13383</td>
</tr>
<tr>
<td></td>
<td></td>
<td>75%</td>
<td>149.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>95%</td>
<td>191.2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>97.5%</td>
<td>195.8</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>99.5%</td>
<td>199.3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
9. WORK EFFICIENTLY
(1) TWO-LEVEL MULTILEVEL MODELS

use "http://www.bristol.ac.uk/cmm/media/runmlwin/tutorial.dta", clear

runmlwin normexam cons, ///
   level2(school: cons) ///
   level1(student: cons)

runmlwin normexam cons, ///
   level2(school: cons, residuals(u)) ///
   level1(student: cons) nopause

egen pickone = tag(school)

preserve
   keep if pickone==1
   egen u0rank = rank(u0)
   serrbar u0 u0se u0rank, scale(1.96) yline(0)
   summarize u0
   generate u0std = (u0 - r(mean))/r(sd)
   generate u0uniform = (u0rank - 0.5)/_N
   generate u0nscore = invnorm(u0uniform)
   scatter u0std u0nscore, ///
      yline(0) xline(0) ylabel(-3(1)3) xlabel(-3(1)3) ///
      aspectratio(1)
10. RESOURCES TO HELP YOU LEARN RUNMLWIN
help runmlwin

Title
runmlwin - Run the MLwiN multilevel modelling software from within Stata

Syntax
runmlwin responses_and_fixed_part, random_part [discrete(discrete_options)] [mcmc(mcmc_options)]
[general_options]

where the syntax of responses_and_fixed_part is one of the following

for univariate continuous, binary, proportion and count response models

  depvar indepvars [if] [in]

for univariate ordered and unordered categorical response models

  depvar indepvars1 [(indepvars2, contrast(numlist)) ...] [if] [in]

where indepvars1 are those independent variables which appear with separate coefficients in each of every log-odds contrast, while indepvars2 are those independent variables which appear with common coefficients for those log-odds contrasts specified in contrast(numlist). Contrasts can be thought of as the separate "subequations" or "arms" of a multinomial response model. These contrasts are indexed 1,2,... up to the total number of contrasts included in the model. The total number of contrasts will be one less than the number of response categories.

for multivariate response models

  (depvar1 indepvars1, equation(numlist))
  (depvar2 indepvars2, equation(numlist))
  [(depvar3 indepvars3, equation(numlist))]
  [if] [in]

where equation(numlist) specifies equation numbers. Equation numbers are indexed 1,2,... up to the total number of equations (i.e. response variables) included in the model.

and the syntax of random_part is

  [ ... ] [level2(levelvar: [varlist] [, random_part_options])]
  level1(levelvar: [varlist] [, random_part_options])
Examples

IMPORTANT. The following examples will only work on your computer once you have installed MLwiN and once you have told runmlwin what the mlwin.exe file address is. See Remarks on installation instructions above for more information.

(a) Continuous response models

Two-level models

Setup

. use http://www.bristol.ac.uk/cmm/media/runmlwin/tutorial, clear

Two-level random-intercept model, analogous to xtreg (fitted using IGLS)
(See page 28 of the MLwiN User Manual)
(You will need to click the "Resume macro" button twice in MLwiN to fit the model.)
. runmlwin normexam cons standlrt, level2(school: cons) level1(student: cons)

Two-level random-intercept and random-slope (coefficient) model (fitted using IGLS)
(See page 59 of the MLwiN User Manual)
. runmlwin normexam cons standlrt, level2 (school: cons standlrt) level1 (student: cons)

Refit the model suppressing the two pauses in MLwiN (fitted using IGLS)
(See page 59 of the MLwiN User Manual)
. runmlwin normexam cons standlrt, level2 (school: cons standlrt) level1 (student: cons) nopause

Refit the model, where this time we additionally calculate the level 2 residuals (fitted using IGLS)
(See page 59 of the MLwiN User Manual)
. runmlwin normexam cons standlrt, level2 (school: cons standlrt, residuals(u)) level1 (student: cons)

Two-level random-intercept and random-slope (coefficient) model with a complex level 1 variance function
(fitted using IGLS)
(See page 99 of the MLwiN User Manual)
. matrix A = (1,1,0,0,0,1)
. runmlwin normexam cons standlrt girl, level2(school: cons standlrt) level1(girl, elements(A))

Two-level random-intercept and random-slope (coefficient) model using MCMC (where we first fit the model using IGLS to obtain initial values for the MCMC chains)
(See page 71 of the MLwiN MCMC Manual)
. runmlwin normexam cons standlrt, level2 (school: cons standlrt) level1 (student: cons)
. runmlwin normexam cons standlrt, level2 (school: cons standlrt) level1 (student: cons) mcmc(on) initprevious

Multivariate response models
runmlwin is a Stata command which allows Stata users to run the powerful MLwiN multilevel modelling software from within Stata.

The multilevel models fitted by runmlwin are often considerably faster than those fitted by the Stata’s xtmixed, xtmelogit and xtmepoisson commands. The range of models which can be fitted by runmlwin is also much wider than those commands. runmlwin also allows fast estimation on large data sets for many of the more complex multilevel models available through the user written gllamm command.

MLwiN has the following features:

1. Estimation of multilevel models for continuous, binary, count, ordered categorical and unordered categorical data
2. Fast estimation via classical and Bayesian methods
3. Estimation of multilevel models for cross-classified and multiple membership nonhierarchical data structures
4. Estimation of multilevel multivariate response models, multilevel spatial models, multilevel measurement error models and multilevel multiple imputation models

These details with a screen shot are available on our runmlwin leaflet (pdf, 0.1mb)

Presentations

We have provided a range of presentations showcasing runmlwin. These presentations provide a quick overview of how the command works and the range of models which can be fitted. More >>

Download
Presentations using runmlwin

- UK Stata Users' Group, 17th Meeting (16th September 2011)
  - Slides (PDF, 2.0mb)
  - Stata do-file (do, 0.1mb) to replicate all analyses presented in the slides.

- University of Bristol, Mplus/MlwiN User Group (MUGS) meeting (14th June 2011)
  - Slides (PDF, 2.3mb)
  - Stata do-file (do, 0.1mb) to replicate all analyses presented in the slides.

- Modern Modeling Methods (M3) Conference, University of Connecticut (26th May 2011)
  - Slides (PDF, 3.2mb)
  - Stata do-file (do, 0.1mb) to replicate all analyses presented in the slides.

- 2011 American Sociological Association Spring Methodology Conference, Tilburg University (20th May 2011)
  - Slides (PDF, 2.0mb)
  - Stata do-file (do, 0.1mb) to replicate all analyses presented in the slides.

- University of Bristol, e-Stat meeting (7th April 2011)
  - Slides (PDF, 1.7mb)
  - Stata do-file (do, 0.1mb) to replicate all analyses presented in the slides.

- 8th International Amsterdam Multilevel Conference (17th March 2011)
Examples using runmlwin

MLwiN User Manual

These do-files and log files replicate the analyses reported in the MLwiN User Manual (PDF, 4.6 mb) Rasbash, J., Steele, F., Browne, W.J. and Goldstein, H. (2009) Centre for Multilevel Modelling, University of Bristol.

Note that we have not created do-files for Chapters 1, 8 or 19 of the manual as no models are fitted in those chapters.

We have also not yet attempted to replicate the analysis in Chapter 17.

- 1 - Introducing Multilevel Models
- 2 - Introduction to Multilevel Modelling (do | log)
- 3 - Residuals (do | log)
- 4 - Random Intercept and Random Slope Models (do | log)
- 5 - Graphical Procedures for Exploring the Model (do | log)
- 6 - Contextual Effects (do | log)
- 7 - Modelling the Variance as a Function of Explanatory Variables (do | log)
- 8 - Getting Started with your Data
- 9 - Logistic Models for Binary and Binomial Responses (do | log)
- 10 - Multinomial Logistic Models for Unordered Categorical Responses (do | log)
- 11 - Fitting an Ordered Category Response Model (do | log)
- 12 - Modelling Count Data (do | log)
- 13 - Fitting Models to Repeated Measures Data (do | log)
- 14 - Multivariate Response Models (do | log)
For models fitted using MCMC estimation, we ask that you additionally cite:


**Papers using runmlwin**

Please let George Leckie (g.leckie@bristol.ac.uk) know of any further publications using runmlwin including forthcoming papers, books, PhD theses, etc.


**Books discussing runmlwin**

runmlwin user forum

Forum rules

NEWTOPIC★ Search this forum... Search

20 topics • Page 1 of 1

ANNOUNCEMENTS

Do-ffiles to replicate entire MLwiN User & MCMC Manuals
by GeorgeLeckie » Mon Apr 18, 2011 5:30 pm
0 123 by GeorgeLeckie Mon Apr 18, 2011 5:30 pm

Welcome to the runmlwin discussion forum
by GeorgeLeckie » Fri Apr 01, 2011 4:06 pm
0 130 by GeorgeLeckie Fri Apr 01, 2011 4:06 pm

TOPICS

MVs & error message 'line too long'
by julia1633 » Mon Aug 15, 2011 3:17 pm
17 125 by julia1633 Sun Aug 28, 2011 12:05 am

runmlwin in Batch mode - gui causing error?
by ash » Sat Aug 27, 2011 6:43 am
2 20 by ash Sat Aug 27, 2011 9:07 pm

Bug in residuals(u, savechains("u.dta", replace))?
by ash » Mon Aug 01, 2011 7:06 pm
4 76 by GeorgeLeckie Wed Aug 03, 2011 6:25 pm

Predictions via the runmlwin interface: a clarification
by ewancarr » Tue Jul 26, 2011 6:49 pm
6 84 by GeorgeLeckie Wed Jul 27, 2011 7:04 pm

Highly correlated multivariate dependents - numerical error
by ash » Sat Jul 23, 2011 10:48 am
1 67 by GeorgeLeckie Mon Jul 25, 2011 3:49 pm

Input dataset contains double precision data...
by ewancarr » Wed Jul 13, 2011 3:55 pm
2 81 by ewancarr Wed Jul 13, 2011 9:16 pm

Modelling Count Data (example do-file) - mismatch error
by leap » Tue Jul 12, 2011 10:18 am
1 48 by ChrisCharlton Wed Jul 13, 2011 3:32 pm

Error code: r(-1073740777);
by pd65 » Mon Jul 04, 2011 11:01 am
7 207 by GeorgeLeckie Thu Jul 07, 2011 3:15 pm

MCMC estimation
by isanne » Fri Apr 08, 2011 8:28 am
5 293 by ChrisCharlton Thu Jun 09, 2011 10:08 am
Citing runmlwin

• If you use runmlwin in your work, please cite runmlwin


• We can then add you to the list of papers using runmlwin on our website

• http://www.bristol.ac.uk/cmm/software/runmlwin/citations