The Stata Journal

The *Stata Journal* publishes reviewed papers together with shorter notes or comments, regular columns, book reviews, and other material of interest to Stata users. Examples of the types of papers include 1) expository papers that link the use of Stata commands or programs to associated principles, such as those that will serve as tutorials for users first encountering a new field of statistics or a major new technique; 2) papers that go "beyond the Stata manual" in explaining key features or uses of Stata that are of interest to intermediate or advanced users of Stata; 3) papers that discuss new commands or Stata programs of interest either to a wide spectrum of users (e.g., in data management or graphics) or to some large segment of Stata users (e.g., in survey statistics, survival analysis, panel analysis, or limited dependent variable modeling); 4) papers analyzing the statistical properties of new or existing estimators and tests in Stata; 5) papers that could be of interest or usefulness to researchers, especially in fields that are of practical importance but are not often included in texts or other journals, such as the use of Stata in managing datasets, especially large datasets, with advice from hard-won experience; and 6) papers of interest to those who teach, including Stata with topics such as extended examples of techniques and interpretation of results, simulations of statistical concepts, and overviews of subject areas.

The *Stata Journal* is indexed and abstracted by *CompuMath Citation Index*, *Current Contents/Social and Behavioral Sciences*, *RePEc: Research Papers in Economics*, *Science Citation Index Expanded* (also known as *SciSearch*), *Scopus*, and *Social Sciences Citation Index*.

For more information on the *Stata Journal*, including information for authors, see the webpage

http://www.stata-journal.com

Subscriptions are available from StataCorp, 4905 Lakeway Drive, College Station, Texas 77845, telephone 979-696-4600 or 800-STATA-PC, fax 979-696-4601, or online at

http://www.stata.com/bookstore/sj.html

Subscription rates listed below include both a printed and an electronic copy unless otherwise mentioned.

U.S. and Canada		Elsewhere	
Printed & electronic		**Printed & electronic**	
1-year subscription	$ 98	1-year subscription	$138
2-year subscription	$165	2-year subscription	$245
3-year subscription	$225	3-year subscription	$345
1-year student subscription	$ 75	1-year student subscription	$ 99
1-year institutional subscription	$245	1-year institutional subscription	$285
2-year institutional subscription	$445	2-year institutional subscription	$525
3-year institutional subscription	$645	3-year institutional subscription	$765
Electronic only		**Electronic only**	
1-year subscription	$ 75	1-year subscription	$ 75
2-year subscription	$125	2-year subscription	$125
3-year subscription	$165	3-year subscription	$165
1-year student subscription	$ 45	1-year student subscription	$ 45

Back issues of the *Stata Journal* may be ordered online at

http://www.stata.com/bookstore/sjj.html

Individual articles three or more years old may be accessed online without charge. More recent articles may be ordered online.

http://www.stata-journal.com/archives.html

The *Stata Journal* is published quarterly by the Stata Press, College Station, Texas, USA.

Address changes should be sent to the *Stata Journal*, StataCorp, 4905 Lakeway Drive, College Station, TX 77845, USA, or emailed to sj@stata.com.

The Stata Journal (2014)
14, Number 1, pp. 119–140

Estimating adjusted associations between random effects from multilevel models: The reffadjust package

Tom M. Palmer
Division of Health Sciences
Warwick Medical School
University of Warwick
Coventry, UK
t.m.palmer@warwick.ac.uk

Corrie M. Macdonald-Wallis
MRC and University of Bristol Integrative Epidemiology Unit
School of Social and Community Medicine
University of Bristol
Bristol, UK

Debbie A. Lawlor
MRC and University of Bristol Integrative Epidemiology Unit
School of Social and Community Medicine
University of Bristol
Bristol, UK

Kate Tilling
School of Social and Community Medicine
University of Bristol
Bristol, UK

Abstract. We describe a method to estimate associations between random effects from multilevel models. We provide two new postestimation commands, `reffadjustsim` and `reffadjust4nlcom`, which are distributed as the `reffadjust` package. These commands produce the estimates and their associated confidence intervals. The commands are used after official Stata multilevel model estimation commands `mixed`, `meqrlogit`, and `meqrpoisson` (formerly named `xtmixed`, `xtmelogit`, and `xtmepoisson`, respectively, before Stata 13) and with models fit in the MLwiN statistical software package via the `runmlwin` command. We demonstrate our commands with several simulated datasets and for a bivariate outcome model investigating the relationship between weight and mean arterial pressure in pregnant women using data from the Avon Longitudinal Study of Parents and Children. Our method and commands help to improve the interpretability of estimated random-effects variance components from multilevel models.

Keywords: st0327, reffadjust, reffadjustsim, reffadjust4nlcom, meqrlogit, meqrpoisson, mixed, multilevel models, runmlwin, xtmelogit, xtmepoisson, xtmixed, ALSPAC

1 Introduction

In this article, we describe how to estimate the association between two random effects from a multilevel model while possibly adjusting for other random effects in the model (Tilling, Sterne, and Wolfe 2001; Macdonald-Wallis et al. 2012). We implement our method in two commands, `reffadjust4nlcom` and `reffadjustsim`, which make up the random-effects adjustment (`reffadjust`) package (Palmer and Macdonald-Wallis 2012). Our commands are postestimation commands used after Stata's multilevel modeling commands `mixed`, `meqrlogit`, and `meqrpoisson` and after multilevel models fit using the MLwiN software in Stata via `runmlwin` (Rasbash et al. 2009; Leckie and Charlton 2011, 2013). Note that the official Stata commands `mixed`, `meqrlogit`, and `meqrpoisson` have been renamed in Stata 13 and were formerly called `xtmixed`, `xtmelogit`, and `xtmepoisson`, respectively. Our `reffadjust` commands work with the former `xt` prefixed command names in Stata 11 and 12.

In section 2, we introduce multilevel models with a simulated example. We describe how to estimate the association between random effects from a multilevel model and demonstrate this using our `reffadjust` commands. In section 3, we provide an example estimating the associations between random effects from a bivariate outcome multilevel model investigating change in weight and mean arterial pressure (MAP) in pregnant women using data from the Avon Longitudinal Study of Parents and Children (ALSPAC). This example demonstrates our commands when the multilevel model is fit by the MLwiN software package (Rasbash et al. 2009) run from Stata by `runmlwin` (Leckie and Charlton 2011). In section 4, we describe the methodology behind our commands and their syntax and options. In section 5, we demonstrate the commands after a multilevel logistic regression fit with `meqrlogit`. Section 6 is a discussion.

2 Estimating associations between random effects from multilevel models

2.1 Simulating and fitting a random-intercept model

We start by simulating a simple two-level, random-intercept model to clearly demonstrate how our method of estimating associations between random effects works with known parameter values. We simulate $j = 1, \ldots, J$ individuals, each with $i = 1, \ldots, n$ repeated measurements: j denotes level 2 of the model and i denotes level 1 of the model. We denote the outcome variable Y, the fixed-effects intercept or constant β_0 (`cons`), random effects u_{0j} (`u0`), and level 1 residuals ϵ_{ij} (`e`). So y_{ij} denotes the value of the outcome for the ith measurement of the jth individual. The model is written

$$y_{ij} = \beta_0 + u_{0j} + \epsilon_{ij}, \quad u_{0j} \sim N(0,\ \sigma^2_{u0}), \quad \epsilon_{ij} \sim N(0,\ \sigma^2_{\epsilon}) \tag{1}$$

We simulate data from this model by setting $J = 1000$ individuals, each with $n = 10$ repeated measures, for 10,000 observations in total. We set the other parameters as follows: $\beta_0 = 1$, $\sigma^2_{u0} = 0.25$, and $\sigma^2_{\epsilon} = 1$.

```
. set seed 12345
. set obs 1000
obs was 0, now 1000
. generate int j = _n
. generate double u0 = rnormal(0, sqrt(.25))
. expand 10
(9000 observations created)
. generate byte cons = 1
. by j, sort: generate int i = _n
. generate double e = rnormal()
. generate double y = cons + u0 + e
```

We then fit the model using restricted maximum-likelihood estimation with `mixed`.

```
. mixed y || j: , reml nolog nolrtest noheader
```

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
_cons	1.011687	.0187194	54.04	0.000	.9749982	1.048377

Random-effects Parameters	Estimate	Std. Err.	[95% Conf.	Interval]
j: Identity				
var(_cons)	.2500984	.01575	.221058	.2829539
var(Residual)	1.003164	.0149543	.9742783	1.032906

The parameters are estimated at approximately their true values. We see that `mixed` does not report z statistics and p-values for estimated variance components of the random effects. This is because symmetric Wald-type confidence intervals (CIs) may not be appropriate for estimated variance components, because they may have skewed distributions and because variances must be positive (Gutierrez, Carter, and Drukker 2001).

Our `reffadjust` commands use the estimated random-effects variances and covariances; there are two ways to estimate these from a multilevel model. The first method is to use the estimates from the model (that is, the `var(_cons)` estimate from the `mixed` results above). The second method is to use the model estimates to predict the random effects (also known as best linear unbiased predictions, or BLUPs) and then to calculate their variance. We generate the BLUPs and calculate their variance as follows:

```
. predict double u0hat, reffects
. quietly summarize u0hat
. display %8.4f `r(Var)´
  0.1783
```

We see that the variance of the BLUPs of 0.1783 is smaller than the estimate from the model above of 0.25 [95% CI: 0.22, 0.28]. The variance of the BLUPs is downwardly biased compared with the true value of 0.25. This downward bias in the variance of the BLUPs is well known and is given by (Morris 2002)

$$\text{var}(\widehat{u}_0) = \left(\frac{n\sigma^2_{u0}}{n\sigma^2_{u0} + \sigma^2_\epsilon}\right)(1 - 1/J)\sigma^2_{u0}$$

This is biased because it differs from simply σ^2_{u0}. For this simulated dataset, we see that our calculated variance of the BLUPs agrees with the formula $[\{10 \times 0.25/(10 \times 0.25 + 1)\} \times (1 - 1/1000) \times 0.25]$.

Variances of BLUPs are downwardly biased: the individual BLUPs are shrunken toward the population mean (Robinson 1991; Morris 2002; Carpenter, Goldstein, and Rasbash 2003). This bias is more pronounced for small sample sizes at either level of the model (that is, in this example, for a small number of individuals J or small numbers of repeated measurements n) and for cases where the residual variance σ^2_ϵ is not small compared with the random-effects variances σ^2_{u0}. This bias in the variance of the BLUPs is perhaps surprising given their name; however, the "unbiased" in BLUP refers to the property that $E(\widehat{u}_0) = E(u_0)$ as opposed to $E(\widehat{u}_0|u_0) = u_0$ (Robinson 1991). Hence, in our `reffadjust` commands, we use the estimated variance components of the random effects from the model, rather than the estimated variance of the BLUPs, to avoid this downward bias in the variance of the BLUPs.

2.2 Simulating and fitting a mixed-effects model

We now consider a slightly more complex model in which we include a covariate X as both a fixed and a random effect. Our new model is given by

$$y_{ij} = \beta_0 + \beta_1 x_{ij} + u_{0j} + u_{1j}x_{ij} + \epsilon_{ij}$$
$$\begin{pmatrix} u_{0j} \\ u_{1j} \end{pmatrix} \sim \text{MVN}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{bmatrix} \sigma^2_{u0} & \sigma_{u01} \\ \sigma_{u01} & \sigma^2_{u1} \end{bmatrix}\right), \quad \epsilon_{ij} \sim N(0, \sigma^2_\epsilon)$$

So u_{1j} are the random effects for the covariate X, σ^2_{u1} is their variance, and σ_{u01} is the covariance between the two random effects. We simulate data from this model as follows, where $X_{ij} \sim N(i,\ 1)$, $\beta_0 = 1$, $\beta_1 = 1$, $\sigma^2_\epsilon = 1$, $\sigma^2_{u0} = \sigma^2_{u1} = 1$, and $\sigma_{u01} = 0.25$.

```
. clear
. set seed 12345
. matrix V = (1, .25 \ .25, 1)
. set obs 1000
obs was 0, now 1000
. generate int j = _n
. corr2data u0 u1, double cov(V)
. expand 10
(9000 observations created)
. generate byte cons = 1
```

```
. by j, sort: generate int i = _n
. by j, sort: generate double x = rnormal(i, 1)
. generate double e = rnormal()
. generate double y = cons + x + u0 + u1*x + e
```

Figure 1 shows the simulated outcome Y for a sample of 10 of our 1,000 level 2 individuals.

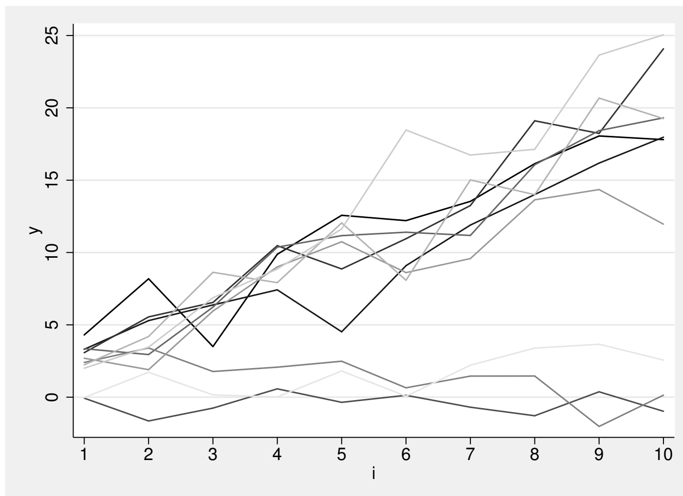

Figure 1. Plot of outcome y versus the level 1 index i for 10 level 2 individuals from the second simulated dataset. Lines on the plot are shaded using the approach of Kohler and Eckman (2011).

We then fit the model using restricted maximum-likelihood estimation with `mixed`.

```
. mixed y x || j: x, reml covariance(uns) nolog nolrtest noheader
```

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
x	.997819	.0316633	31.51	0.000	.9357601	1.059878
_cons	1.00568	.0380484	26.43	0.000	.9311064	1.080253

Random-effects Parameters	Estimate	Std. Err.	[95% Conf.	Interval]
j: Unstructured				
var(x)	.9915986	.0448656	.9074499	1.083551
var(_cons)	1.018231	.065386	.897813	1.154799
cov(x,_cons)	.2555368	.0386803	.1797248	.3313488
var(Residual)	.9827593	.0155486	.9527523	1.013711

The model parameters are again estimated close to their true values, with all CIs including the true values.

Estimating associations between random effects

So far, we have estimated the covariance, 0.26 [95% CI: 0.18, 0.33], between the two random effects in the model. Our approach implemented in `reffadjust` estimates the mean difference in one random effect associated with a unit change in the other. To estimate the association between u_0 and u_1, we propose a linear model of the form

$$u_{0j} = \gamma u_{1j} + \nu_j, \quad \nu_j \sim N(0,\ \tau^2) \tag{2}$$

γ is then given by the analytic form of the maximum likelihood estimate of a regression coefficient from a simple linear model with a single covariate,

$$\gamma = \frac{\text{cov}(u_0, u_1)}{\text{var}(u_1)} = \frac{\sigma_{u01}}{\sigma_{u1}^2} \tag{3}$$

Hence, we can obtain an estimate for γ by substituting in our estimates of σ_{u01} and σ_{u1} from the model. Note that (2) does not require an intercept, because the random effects have mean 0. In our simulated data, γ should be approximately equal to 0.25 (0.25/1). To check this, we can fit the model in the simulated data by using the actual random effects (that is, the actual simulated values and not the BLUP estimates derived from fitting the model). In the following code, we first use `egen` to select a single observation per individual.

```
. egen pickone = tag(j)
. regress u0 u1 if pickone==1, nocons noheader
```

u0	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
u1	.25	.0306339	8.16	0.000	.1898857	.3101143

As we expected, the association between u_0 and u_1 is 0.25 [95% CI: 0.19, 0.31] when we use the actual random effects. Of course, in a real dataset, these actual values of the random effects are not available, but we can estimate this association with our `reffadjust` commands.

```
. quietly mixed y x || j: x, reml covariance(uns)
. reffadjust4nlcom _cons x, eqn(j)
. nlcom `r(beta_x)´
       _nl_1:  tanh([atr1_1_1_2]_cons)*exp([lns1_1_1]_cons +
> [lns1_1_2]_cons)/exp(2*[lns1_1_1]_cons)
```

y	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
_nl_1	.2577018	.0381171	6.76	0.000	.1829936	.33241

```
. reffadjustsim _cons x, eqn(j) seed(101112)
```

_cons	Median	2.5 Percentile	97.5 Percentile
x	.2575872	.183246	.3311031

Both commands estimate the association between u_0 and u_1 as 0.26 [95% CI: 0.18, 0.33], which is very close to the true value of 0.25. This means that for a 1-unit increase in u_1, u_0 increases by an average of 0.26 units.

The `reffadjust4nlcom` command works by returning an expression for the association, as given by (3), to use in `nlcom`, which obtains a delta-method CI. The expression in the output above looks complex because we have to back-transform the `mixed` random-effects variance-component estimates from their estimation metric. Random-effects variances are estimated on the log standard-deviation scale, and random-effects covariances are estimated as the inverse hyperbolic tangent of the correlation. Our `reffadjustsim` command employs a simulation approach to produce a delta-method CI for the estimated association between the random effects. The commands are described in more detail in section 4.

To demonstrate our warning about the downward bias in the variance of BLUPs, let's use the BLUPs to estimate γ.

```
. predict double u1hat double u0hat, reffects
. corr u0hat u1hat if pickone==1, covariance
(obs=1000)
```

	u0hat	u1hat
u0hat	.730469	
u1hat	.29556	.983507

```
. regress u0hat u1hat if pickone==1, nocons noheader
```

u0hat	Coef.	Std. Err.	t	P>|t|	[95% Conf.	Interval]
u1hat	.3005166	.0255551	11.76	0.000	.2503688	.3506644

Using the BLUPs only, we see that the estimate of σ^2_{u1} of 0.98 is close to its true value (which was set to 1). The estimate of σ^2_{u0} of 0.73 is, as we described earlier, downwardly biased with respect to its true value of 1. As a result, the estimated covariance between the BLUPs of 0.296 is also biased with respect to its true value of 0.25. This causes the estimate of γ using the BLUPs of 0.30 [95% CI: 0.25, 0.35] to be upwardly biased. This is especially troubling in this case because the lower CI limit just excludes the true value of 0.25 (at the third decimal place). Figure 2 illustrates this point further: the right-hand side shows the narrower spread of the BLUPs that gives rise to the steeper line of fitted values. As explained previously, this is the reason that our `reffadjust` commands use the estimated random-effects variances and covariances from the model results to estimate the associations between random effects rather than the variances and covariances of the BLUPs.

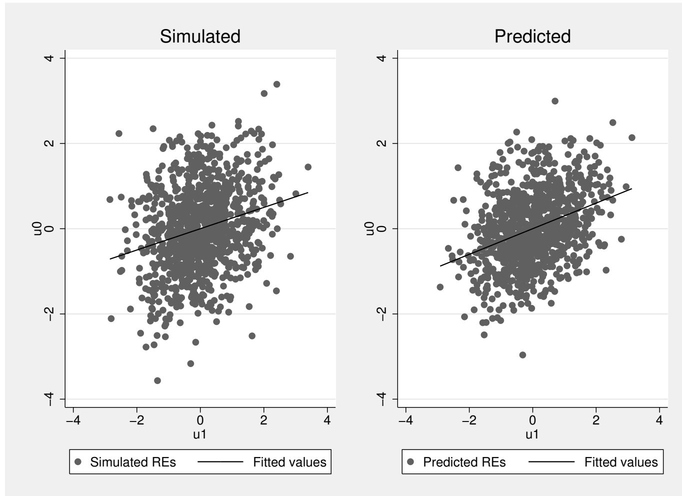

Figure 2. Plot of regression of u_0 on u_1 using simulated (left) and predicted (right) random effects

3 Estimating associations between random effects from multivariate outcome models fit using MLwiN via the runmlwin command

In this section, we describe fitting a bivariate multilevel model for weight and MAP (the average pressure in an artery over a complete cycle of one heartbeat) in women during pregnancy using data from ALSPAC (http://www.bris.ac.uk/alspac).

3.1 Introduction to ALSPAC data example

The data are from ALSPAC, which is described elsewhere (Golding, Pembrey, and Jones 2001; Fraser et al. 2013). The two outcomes are weight and MAP during pregnancy. MAP was calculated as follows (note that the weighting allows for the lower pressure during the diastolic phase of the cardiac cycle):

$$\text{MAP} = \frac{1}{3}(\text{systolic blood pressure}) + \frac{2}{3}(\text{diastolic blood pressure})$$

Multivariate multilevel models are often used to model longitudinal measurements of health-related outcomes, notably, growth curves (Royston 1995; Pan and Goldstein 1997, 1998; Goldstein and Kounali 2009). Hence, we use a bivariate multilevel model with linear splines for gestational age to assess whether an increase in weight precedes a rise in MAP and to determine the periods of pregnancy during which associations are strongest.

We restrict our analysis to singleton full-term births ($\geq$ 37 weeks' gestation) with no evidence of preeclampsia or a previous diagnosis of hypertension ($N = 11650$). All weight and blood-pressure measurements, which were taken routinely as part of antenatal care by midwives or obstetricians, were abstracted from obstetric records by six trained research midwives. This resulted in a median of 14 (interquartile range 11 to 16) blood-pressure measurements and median of 12 (interquartile range 10 to 14) weight measurements per woman. Our original analysis presented in Macdonald-Wallis et al. (2012) adjusted for the confounding variables maternal height, age, parity, education, smoking, and offspring sex; however, in this example, for simplicity, we use unadjusted models.

3.2 Multivariate outcome multilevel model

We denote the two outcomes $y^{(1)}$ (`map`) and $y^{(2)}$ (`weight`) measured on occasions $i = 1, \ldots, n_j$ (level 1) for individuals $j = 1, \ldots, J$ (level 2). Hence, $y_{ij}^{(1)}$ is the value of the response $y^{(1)}$ at the ith measurement of the jth individual; t_{ij} denotes the time of this observation. To simplify our model, we chose to represent the time variable t by a set of splines for each outcome. The splines for MAP were chosen to be between 0–18 weeks, 18–29 weeks, 29–36 weeks, and $\geq$ 36 weeks; the splines for weight were the same except that the last two periods were combined into a single period. More details about how these knot positions were chosen are given in Macdonald-Wallis et al. (2012).

Our bivariate multilevel model is

$$\texttt{map}_{ij} = (\beta_{\texttt{cons\_1}} + u_{\texttt{cons\_1},j}) + \sum_{k=1}^{4} (\beta_{\texttt{sp}k} + u_{\texttt{sp}k,j})\texttt{sp}k + \epsilon_{ij}^{(1)}$$

$$\texttt{weight}_{ij} = (\beta_{\texttt{cons\_2}} + u_{\texttt{cons\_2},j}) + \sum_{l=1}^{3} (\beta_{\texttt{spw}l} + u_{\texttt{spw}l,j})\texttt{spw}l + \epsilon_{ij}^{(2)}$$

$$u_j \sim \text{MVN}_9\left(\mathbf{0}, \Sigma_u\right), \qquad \begin{bmatrix} \epsilon_{ij}^{(1)} \\ \epsilon_{ij}^{(2)} \end{bmatrix} \sim \text{MVN}\left(\mathbf{0},\ \Sigma_\epsilon\right)$$

where Σ_u and Σ_ϵ are unrestricted variance–covariance matrices. After reading in the data and generating the splines with the `mkspline` command, we fit the model using the MLwiN statistical software via the `runmlwin` command as follows (Rasbash et al. 2009; Leckie and Charlton 2011):

```
runmlwin (map cons sp1 sp2 sp3 sp4, eq(1))         ///
    (weight cons spw1 spw2 spw3, eq(2)),           ///
    level2(id: (cons sp1 sp2 sp3 sp4, eq(1))       ///
        (cons spw1 spw2 spw3, eq(2)), reset(none)) ///
    level1(visit_unique: cons)                     ///
    maxiter(5000) nopause
```

The fixed-effects estimates are shown below (note that `runmlwin` includes the suffix _# for each parameter estimate, where # is the equation number).

```
. runmlwin, noheader noretable
```

	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
map						
cons_1	83.07113	.2021721	410.89	0.000	82.67488	83.46738
sp1_1	-.1795707	.012932	-13.89	0.000	-.2049169	-.1542245
sp2_1	.111918	.0083765	13.36	0.000	.0955004	.1283356
sp3_1	.3123525	.0129339	24.15	0.000	.2870025	.3377025
sp4_1	1.158547	.0250621	46.23	0.000	1.109426	1.207668
weight						
cons_2	60.44829	.1326018	455.86	0.000	60.18839	60.70818
spw1_2	.3157002	.0033848	93.27	0.000	.3090661	.3223343
spw2_2	.53217	.0022896	232.43	0.000	.5276824	.5366576
spw3_2	.4595202	.002689	170.89	0.000	.4542499	.4647905

We see that the mean value of MAP at 0 weeks' gestation is 83.07 mmHg [95% CI: 82.67, 83.47] and that the mean weight is 60.45 kg [95% CI: 60.19, 60.71]. Because of the large sample size, all the CIs for the fixed-effects estimates exclude the null of 0. We also see that for MAP, the first spline period is associated with a decrease in the mean level of MAP, whereas the subsequent periods are associated with increasing average levels of MAP. Hence, these estimates imply a U shape for the population mean of MAP across gestation (see figure 2(b) of Macdonald-Wallis et al. [2012] for further information). The lower diagonal estimates of the elements of the random-effects variance–covariance matrix $\widehat{\Sigma}_u$ can be shown conveniently (without their CIs) as follows:

```
. mat b = e(b)
. mat brp2 = b[1,"RP2:"]
. mat Su = J(9,9,.)
. local k 1
. forvalues i=1/9 {
  2.         forvalues j=1/9 {
  3.                 if `j´ <= `i´ {
  4.                         mat Su[`i´,`j´] = brp2[1,`k++´]
  5.                 }
  6.         }
  7. }
. mat bfp = b[1,1..9]
. local names : colnames bfp
. mat rownames Su = `names´
. mat colnames Su = `names´
```

```
. mat list Su, format(%3.2f) noheader noblank
        cons_1   sp1_1   sp2_1   sp3_1   sp4_1  cons_2  spw1_2  spw2_2  spw3_2
cons_1   67.09       .       .       .       .       .       .       .       .
 sp1_1   -2.55    0.18       .       .       .       .       .       .       .
 sp2_1   -0.02   -0.03    0.12       .       .       .       .       .       .
 sp3_1    0.34   -0.03   -0.04    0.36       .       .       .       .       .
 sp4_1   -0.27    0.02   -0.01   -0.11    1.34       .       .       .       .
cons_2   25.21    0.45   -0.65    0.21   -0.47  150.21       .       .       .
spw1_2   -0.09   -0.00    0.01   -0.01   -0.00   -0.98    0.05       .       .
spw2_2   -0.16    0.01    0.01   -0.00    0.01   -0.31    0.01    0.03       .
spw3_2   -0.10    0.01    0.01    0.02    0.03    0.03    0.00    0.02    0.05
```

And the estimates of Σ_ϵ are

$$\widehat{\Sigma}_\epsilon = \begin{bmatrix} 36.55\ [95\%\ \text{CI}:\ 36.12, 36.98] & \\ 0.06\ [95\%\ \text{CI}:\ 0.01, 0.11] & 0.79\ [95\%\ \text{CI}:\ 0.78, 0.80] \end{bmatrix}$$

As an example, we estimate the association between the random effects for the third and second spline periods for MAP ($u_{\mathtt{sp3}}$ and $u_{\mathtt{sp2}}$) by using our **reffadjust4nlcom** command. We also show this association adjusted for the first spline-period random effect ($u_{\mathtt{sp1}}$) and the baseline random effect ($u_{\mathtt{cons\_1}}$).

```
. reffadjust4nlcom sp3_1 sp2_1, eqn(RP2)
. nlcom `r(beta_sp2_1)´
       _nl_1:  [RP2]cov(sp2_1\sp3_1)/[RP2]var(sp2_1)
```

	Coef.	Std. Err.	z	P>|z|	[95% Conf. Interval]	
_nl_1	-.3659381	.0847744	-4.32	0.000	-.5320929	-.1997833

```
. reffadjust4nlcom sp3_1 sp2_1 sp1_1 cons_1, eqn(RP2)
. nlcom `r(beta_sp2_1)´, noheader
```

	Coef.	Std. Err.	z	P>|z|	[95% Conf. Interval]	
_nl_1	-.4790297	.0835663	-5.73	0.000	-.6428166	-.3152429

Hence, a 1 mmHg per week increase in the second MAP spline-period random effect is on average associated with a −0.37 mmHg per week [95% CI: −0.53, −0.20] increase in the third spline-period random effect. This association increases in magnitude to −0.48 mmHg per week [95% CI: −0.64, −0.32] on adjustment for the earlier MAP spline-period random effects. We argue that these associations are easier to interpret than the estimated covariance between these two random effects in the matrix above of −0.04 [95% CI: −0.07, −0.02], which equated to a correlation of −0.21 [95% CI: −0.30, −0.11] (obtained by running `runmlwin, correlations` after the initial `runmlwin` command).

We also investigate the association between the third MAP spline-period random effect and previous spline-period random effects for both outcomes. Because this involves adjusting for more than four random effects, we use our `reffadjustsim` command.

```
. reffadjustsim sp3_1 sp2_1 sp1_1 cons_1 spw2_2 spw1_2 cons_2, eqn(RP2)
```

sp3_1	Median	2.5 Percentile	97.5 Percentile
sp2_1	-.4823532	-.6405699	-.2808418
sp1_1	-.4113592	-.5789848	-.2575228
cons_1	-.01079	-.016345	-.0050133
spw2_2	.2870492	.108691	.4641665
spw1_2	-.2271708	-.4063453	-.0531179
cons_2	.0014225	-.000989	.0038398

While the weight spline-period random effects are associated with the MAP `sp3` random effect, they do not materially alter the association between the `sp3` and `sp2` MAP random effects.

MLwiN can also perform Bayesian estimation, after which the Markov chains of the posterior distributions of the parameters can be read into Stata using `runmlwin`'s `mcmcsum, getchains` postestimation command (Leckie and Charlton 2011). Both of our `reffadjust` commands can be used with these Markov chains from Bayesian estimation (see the examples in the `reffadjust4nlcom` and `reffadjustsim` help files for more information).

4 The reffadjust package

This package includes two programs, `reffadjust4nlcom` and `reffadjustsim`, which are postestimation commands used after the following multilevel modeling commands: `mixed` (formerly `xtmixed`), `meqrlogit` (formerly `xtmelogit`), `meqrpoisson` (formerly `xtmepoisson`), and `runmlwin`.

4.1 reffadjust4nlcom: Delta-method CIs using analytic expressions for regression coefficients

The `reffadjust4nlcom` command produces a local macro of the form of (3) or (6) for the appropriate number of included covariates. As the name suggests, this expression is then passed to `nlcom` to obtain a delta-method CI for the estimated association.

In our example using the second simulated dataset above, we used (3), the analytic expression of a regression coefficient, for a model with a single covariate. If there were three random effects in the model, for example,

$$\begin{pmatrix} u_{0j} \\ u_{1j} \\ u_{2j} \end{pmatrix} \sim \mathrm{MVN}\left(\begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}, \begin{bmatrix} \sigma^2_{u0} & \sigma_{u01} & \sigma_{u02} \\ \sigma_{u01} & \sigma^2_{u1} & \sigma_{u12} \\ \sigma_{u02} & \sigma_{u12} & \sigma^2_{u2} \end{bmatrix} \right) \tag{4}$$

and we wished to fit the model

$$u_{0j} = \gamma_1 u_{1j} + \gamma_2 u_{2j} + \nu_j, \quad \nu_j \sim N(0, \tau^2) \tag{5}$$

we would use the analytic expression for the coefficients from regression models, including two covariates. For example, γ_1, the mean difference in u_0 associated with a unit change in u_1, adjusted for u_2, is given by

$$\gamma_1 = \frac{\sigma_{u2}^2 \sigma_{u01} - \sigma_{u12}\sigma_{u02}}{\sigma_{u1}^2 \sigma_{u2}^2 - (\sigma_{u12})^2} \tag{6}$$

Formulas such as (3) and (6) are derived from the well-known matrix form of the maximum likelihood estimates for linear regression. Formulas for coefficients from models with three and four covariates are given in the supplementary material of Macdonald-Wallis et al. (2012). As far as we know, it is not possible to derive analytic formulas for coefficients adjusting for larger numbers of covariates, because there is no convenient analytic expression for the inverse of a matrix of order greater than 4. We also note that the formulas for the coefficients from models without an intercept are slightly different. However, this is not problematic for our approach, because the random effects have mean 0, in which case coefficients for models with and without an intercept are equivalent.

4.2 reffadjustsim: Simulating from a multivariate normal distribution for the estimated random effects

The `reffadjustsim` command repeatedly simulates from a multivariate normal distribution by using the estimated random-effects variances and covariances as the mean vector and their estimated variance–covariance matrix ($\widehat{\mathbf{V}}$) as the variance–covariance matrix. Hence, for a model with three random effects, as in (4), for each iteration r we simulate

$$\begin{pmatrix} \sigma_{u0r}^2 \\ \sigma_{u1r}^2 \\ \sigma_{u2r}^2 \\ \sigma_{u01r} \\ \sigma_{u02r} \\ \sigma_{u12r} \end{pmatrix} \sim \text{MVN} \left(\begin{pmatrix} \widehat{\sigma}_{u0}^2 \\ \widehat{\sigma}_{u1}^2 \\ \widehat{\sigma}_{u2}^2 \\ \widehat{\sigma}_{u01} \\ \widehat{\sigma}_{u02} \\ \widehat{\sigma}_{u12} \end{pmatrix}, \widehat{\mathbf{V}} \right)$$

Therefore for (5), we denote for each iteration r

$$\boldsymbol{\gamma}_r = \begin{pmatrix} \gamma_{1r} \\ \gamma_{2r} \end{pmatrix}, \quad \boldsymbol{\Sigma}_r = \begin{bmatrix} \sigma_{u1r}^2 & \sigma_{u12r} \\ \sigma_{u12r} & \sigma_{u2r}^2 \end{bmatrix}, \quad \boldsymbol{\sigma}_{0r} = \begin{pmatrix} \sigma_{u01r} \\ \sigma_{u02r} \end{pmatrix}$$

and for each iteration, the estimates $\widehat{\boldsymbol{\gamma}}_r$ are found as the solution to (Fisher 1925, chap. 5)

$$\widehat{\boldsymbol{\gamma}}_r = \boldsymbol{\Sigma}_r^{-1} \boldsymbol{\sigma}_{0r}$$

The `reffadjustsim` command reports the median of the r estimates as the estimate of the association between the random effects, and the 2.5 and 97.5 centiles of the r

realizations are reported as the 95% CI limits. The advantage of `reffadjustsim` over `reffadjust4nlcom` is that it can include more than four random effects as covariates.

The methods are described in more detail in Tilling, Sterne, and Wolfe (2001) and Macdonald-Wallis et al. (2012). One limitation is that both commands assume the estimated random-effects variance components follow a multivariate normal distribution. Multivariate normality of these variance components may not hold for small samples: the random-effects variances will most likely have a skewed distribution because they must be positive. However, we suggest that the delta method is a reasonable first approximation to obtain CIs for these parameters. Wald-type CIs for maximum likelihood estimates of variance components from multilevel models become more accurate as the sample sizes at the different levels of the model become larger (Raudenbush and Bryk 2002); hence, our delta-method CIs for the estimated associations between the random effects should also become more accurate.

4.3 Syntax

`reffadjust4nlcom` *depvar indepvars*, `eqn(`*string*`)` [`mcmcsum sf(`*numlist*`)` `sublevel(`*#*`)`]

`reffadjustsim` *depvar indepvars*, `eqn(`*string*`)` [`mcmcsum sf(`*numlist*`)` `sublevel(`*#*`) centileopts(`*string*`) level(`*#*`) n(`*#*`) post replace` `saving(`*filename* [`, replace`]`) seed(`*#*`) statadrawnorm waldtype`]

4.4 Options

`eqn(`*string*`)` specifies the name of the equation from which the coefficients are to be extracted. For example, a bivariate two-level, random-effects model from `runmlwin` will typically return four equations (FP1, FP2, RP1, RP2). `eqn()` is required.

`mcmcsum` for `reffadjust4nlcom` specifies that the returned local use variable names of chains that are returned from MLwiN Bayesian Markov chain Monte Carlo estimation by `mcmcsum, getchains`. `mcmcsum` for `reffadjustsim` calculates centiles from the Bayesian posterior distribution of the coefficients using chains imported by `mcmcsum, getchains`. Note that your `runmlwin` model must have been fit by Markov chain Monte Carlo. The options `seed()`, `n()`, `statadrawnorm`, `waldtype`, and `post` are not allowed with `mcmcsum`. For both commands, `mcmcsum` is allowed only with `runmlwin` estimates.

`sf(`*numlist*`)` specifies the scaling factors. If specified, each number corresponds to the respective covariate (*indepvar*); that is, the first number is the scaling factor for the first coefficient and so on. If `sf(`*numlist*`)` is specified, the *numlist* must be the same length as the number of covariates. For example, to scale the coefficient by 2 times the dependent variable (Y), then with one covariate (X), specify `sf(2)`. To scale the coefficient by 2 times the covariate, specify `sf(.5)`; in this case, you scale by $2/2^2$ because a regression coefficient is given by $\text{cov}(X, Y)/\text{var}(X)$.

`sublevel(#)` specifies the sublevel of a repeated group variable. For example, in the model

```
. mixed y || school: z1 z2, nocons cov(id) ///
    || school: z3 z4, nocons cov(un)
```

`z1` and `z2` are at sublevel 1, and `z3` and `z4` are at sublevel 2 of the `school` group variable. `sublevel()` is valid only with `mixed` or `xtmixed`, `meqrlogit` or `xtmelogit`, and `meqrpoisson` or `xtmepoisson`.

`centileopts(`*string*`)` specifies the options passed to `centile`; note that you may not specify the `centile(#)` option here. The reported percentiles can be changed through the `level(#)` option.

`level(#)`; see [R] **estimation options**.

`n(#)` specifies the number of observations to be simulated. The default is `n(10000)`, and # is not allowed to be less than 10. `n()` is not allowed with `mcmcsum`, where `n()` is taken as the number of observations in the dataset imported by `mcmcsum, getchains`.

`post` causes `reffadjustsim` to behave like a Stata estimation (e-class) command. `post` may be specified only with `waldtype`. When `post` is specified, `reffadjustsim` will post the vector of adjusted estimates and its estimated variance–covariance matrix to `e()`. Thus after posting, you could treat the estimation results in the same way that you would treat results from other Stata estimation commands. For example, after posting, you could redisplay the results by typing `reffadjustsim` without any arguments, or you could use `test` to perform simultaneous tests of hypotheses on linear combinations of the estimates. Specifying `post` clears out the previous estimation results, which can be recovered only by refitting the original model or by storing the estimation results before running `reffadjustsim` and then restoring them; see [R] **estimates store**.

`replace` overwrites variables named `beta_`*indepvar* if they exist in the dataset. It is valid only with `mcmcsum`.

`saving(`*filename*[`, replace`]`)` saves the simulated realizations of the random-effects variances and covariances to *filename*, optionally replacing *filename* if it exists.

`seed(#)` specifies the initial value of the random-number seed. The default is the current random-number seed. Specifying `seed()` is the same as typing `set seed` # before issuing the command; see `help set_seed`. `seed()` is not allowed with `mcmcsum`.

`statadrawnorm` uses Stata's `drawnorm` to simulate the adjusted coefficients. For speed, by default, `reffadjustsim` uses its own Mata implementation; see `help drawnorm`. `statadrawnorm` is not allowed with `mcmcsum`.

`waldtype` reports coefficients as means with Wald-type CIs. By default, `reffadjustsim` reports coefficients as medians and centiles of the simulated coefficients. This option can produce inaccurate results (as indicated in the warning below, please compare with the default output). `waldtype` is not allowed with `mcmcsum`.

4.5 Warnings

Note on multivariate response models

Covariates (*indepvars*) in `runmlwin` estimates from multivariate response models have suffix _#, where # is the corresponding equation number. For example, from (1), `cons` would be referred to as `cons_1`.

Note on shrinkage estimates

`reffadjust4nlcom` and `reffadjustsim` use the estimated random-effects variances and covariances from the model. They do not use the shrinkage estimates of these parameters, that is, the variances and covariances of the BLUPs (Rasbash et al. 2009, chap. 3).

Warning about p-values for these estimates

The *p*-values associated with these estimates from `nlcom` may be affected by boundary value issues in the estimation of the random-effects variances and covariances; see *Distribution theory for likelihood-ratio test* in [ME] **me** (StataCorp 2013) and see Gutierrez, Carter, and Drukker (2001).

Warning about the waldtype option

By default, `reffadjustsim` reports coefficients as medians with 2.5 and 97.5 percentiles. Coefficients can be reported as means with Wald-type CIs with the `waldtype` option. Means and Wald-type CIs may not be accurate. We recommend that you compare results with the default output and, if possible, also with the delta-method CI via `reffadjust4nlcom` and `nlcom`.

Interpretation of coefficients

The coefficients estimated by `reffadjust4nlcom` and `reffadjustsim` represent the mean difference in the random effects entered as dependent variables, which is associated with a unit increase in the random effects entered as independent variables, while adjusting for the other random effects included in the model as independent variables.

Parameters estimated with missing variance

A multilevel model can occasionally be declared as converged by an estimation command, but some parameters (especially random-effects variances and covariances) may not have a standard error. A warning is issued that resulting CIs may not be valid in this case.

Random-effects covariance structures

The `reffadjust` commands work only with exchangeable or unstructured random-effects variance–covariance structures from `mixed`, `meqrlogit`, and `meqrpoisson`.

5 Estimating associations between random effects from random-effects logistic regression models

The `reffadjust` commands can also be used after the official Stata `meqrpoisson` and `meqrlogit` commands for fitting multilevel Poisson and logistic regression models, respectively. To demonstrate this, we define the two-level logistic regression model

$$
\begin{aligned}
y_{ij} &\sim \text{Bernoulli}(p_{ij}) \\
\text{logit}(p_{ij}) &= \beta_0 + \beta_1 X_{ij} + u_{0j} + u_{1j} X_{ij} \\
\begin{pmatrix} u_{0j} \\ u_{1j} \end{pmatrix} &\sim \text{MVN}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{bmatrix} \sigma^2_{u0} & \sigma_{u01} \\ \sigma_{u01} & \sigma^2_{u1} \end{bmatrix} \right)
\end{aligned}
$$

where $\text{logit}(p) = \log\{p/(1-p)\}$. We simulate data from this model by setting $\beta_0(\texttt{cons}) = -4$, $\beta_1 = 1$, $X_{ij} \sim N(0.2i,\ 1)$, $\sigma^2_{u0} = \sigma^2_{u1} = 1$, and $\sigma_{u01} = 0.25$. We tabulate the outcome y by using the following code:

```
. clear
. set seed 12345
. matrix V = (1, .25 \ .25, 1)
. set obs 1000
obs was 0, now 1000
. generate int j = _n
. corr2data u0 u1, double cov(V)
. expand 10
(9000 observations created)
. generate byte cons = -4
. by j, sort: generate int i = _n
. by j, sort: generate double x = rnormal(0.2*i, 1)
. generate double p = invlogit(cons + x + u0 + u1*x)
. generate byte y = rbinomial(1, p)
```

```
. tab y
```

y	Freq.	Percent	Cum.
0	8,481	84.81	84.81
1	1,519	15.19	100.00
Total	10,000	100.00	

We can see that 15% of the 10,000 observations in the simulated dataset have the outcome.

We then fit the model with `meqrlogit`.

```
. meqrlogit y x || j: x, covariance(uns) nolog noheader nogroup nolrtest
```

y	Coef.	Std. Err.	z	P>|z|	[95% Conf. Interval]	
x	1.011929	.0664607	15.23	0.000	.8816683	1.14219
_cons	-3.935916	.1302443	-30.22	0.000	-4.19119	-3.680642

Random-effects Parameters	Estimate	Std. Err.	[95% Conf. Interval]	
j: Unstructured				
var(x)	.7440409	.1277764	.5313949	1.04178
var(_cons)	.9644602	.2959304	.5285738	1.759799
cov(x,_cons)	.2869809	.1197563	.0522628	.5216989

The fixed-effects estimates (the first table of results above) are on the log scale, and we can see they are approximately at their true values. However, there is some downward bias in the estimated random-effects variances in the second table of results. For example, the estimate of σ^2_{u1} (denoted `var(x)`) is 0.74 [95% CI: 0.53, 1.04], which is less than its true value of 1, although the true value is just within the CI.

We then estimate the association between u_0 and u_1 with our **`reffadjust`** commands.

```
. reffadjust4nlcom _cons x, eqn(j)
. nlcom `r(beta_x)´
       _nl_1:  tanh([atr1_1_1_2]_cons)*exp([lns1_1_1]_cons +
> [lns1_1_2]_cons)/exp(2*[lns1_1_1]_cons)
```

y	Coef.	Std. Err.	z	P>|z|	[95% Conf. Interval]	
_nl_1	.3857058	.1994699	1.93	0.053	-.0052481	.7766596

```
. reffadjustsim _cons x, eqn(j)
```

_cons	Median	2.5 Percentile	97.5 Percentile
x	.3771738	-.0715286	.7283991

Both the estimated associations of 0.39 [95% CI: −0.01, 0.78] and 0.38 [95% CI: −0.07, 0.73] are upwardly biased with respect to the true value of 0.25 because of the bias in the estimated variance components. However, both CIs are wide and contain the true value. Carlin et al. (2001) discuss some of the problems of fitting binary outcome multilevel models: in particular, when the proportion of observations with the outcome is small at a particular level of the model, then estimation of random-effects variances and covariances can be problematic.

6 Discussion

We have described `reffadjustsim` and `reffadjust4nlcom`, new postestimation commands that estimate associations between random effects used after multilevel-model estimation commands (`mixed` and `xtmixed`, `meqrlogit` and `xtmelogit`, `meqrpoisson` and `xtmepoisson`, and `runmlwin`). We have demonstrated the use of the commands after `mixed`, a bivariate outcome model fit with MLwiN via `runmlwin`, and `meqrlogit`.

To estimate the association between random effects, we have shown that it is preferable to use estimated variance components from the model rather than from the BLUPs because of the well-known downward bias in the variance of the BLUPs, which occurs because BLUPs are shrinkage estimates. A limitation of our commands is that they assume the estimated random-effects variance components follow a multivariate normal distribution. Multivariate normality of these variance components may not hold: the random-effects variances will most likely have a skewed distribution because they must be positive. Multivariate normality of the estimated random-effects variance components will improve as the numbers in each group of the multilevel structure increase (Raudenbush and Bryk 2002); therefore, we suggest using the delta method as a reasonable first approximation to obtain CIs for these parameters. One way to overcome this limitation when fitting models in MLwiN using `runmlwin` is to use Bayesian estimation, and our commands can produce Bayesian credible intervals for the associations between random effects from such models. Our commands can also use random-effects structures, including the `_all` group identifier available in `mixed`, `meqrlogit`, and `meqrpoisson`.

In applied research, our `reffadjust` commands have been used to derive associations between changes in weight with changes in blood pressure from bivariate multilevel spline models (Macdonald-Wallis et al. 2013) and to estimate associations between energy intake trajectories and offspring body-mass index (Anderson et al. 2013).

In summary, our `reffadjust` commands estimate associations between random effects, with delta-method CIs, for use after several multilevel modeling commands. We hope that our method and commands help to improve the interpretability of estimated random-effects variance components from multilevel models.

7 Acknowledgments

We thank George Leckie and Chris Charlton from the Centre for Multilevel Modelling at the University of Bristol for helpful comments on our programs. This work was supported by the Wellcome Trust (grant number WT087997MA) and UK Medical Research Council (grant number G1000726). Tom Palmer, Corrie Macdonald-Wallis, and Debbie Lawlor were supported by the MRC Centre for Causal Analyses in Translational Epidemiology (grant number G0600705) before it became the MRC Integrative Epidemiology Unit at the University of Bristol. Corrie Macdonald-Wallis is supported by an MRC fellowship (grant number MR/J011932/1). The UK Medical Research Council and the Wellcome Trust (grant number 092731) and the University of Bristol provide core support for ALSPAC. We are extremely grateful to all the families who took part in this study, the midwives for their help in recruiting those families, and the whole ALSPAC team, which includes interviewers, computer and laboratory technicians, clerical workers, research scientists, volunteers, managers, receptionists, and nurses.

8 References

Anderson, E. L., K. Tilling, A. Fraser, C. Macdonald-Wallis, P. Emmett, V. Cribb, K. Northstone, D. A. Lawlor, and L. D. Howe. 2013. Estimating trajectories of energy intake through childhood and adolescence using linear-spline multilevel models. *Epidemiology* 24: 507–515.

Carlin, J. B., R. Wolfe, C. H. Brown, and A. Gelman. 2001. A case study on the choice, interpretation and checking of multilevel models for longitudinal binary outcomes. *Biostatistics* 2: 397–416.

Carpenter, J. R., H. Goldstein, and J. Rasbash. 2003. A novel bootstrap procedure for assessing the relationship between class size and achievement. *Journal of the Royal Statistical Society, Series C* 52: 431–443.

Fisher, R. A. 1925. *Statistical Methods for Research Workers*. Edinburgh: Oliver & Boyd.

Fraser, A., C. Macdonald-Wallis, K. Tilling, A. Boyd, J. Golding, G. Davey Smith, J. Henderson, J. Macleod, L. Molloy, A. Ness, S. Ring, S. M. Nelson, and D. A. Lawlor. 2013. Cohort profile: The Avon longitudinal study of parents and children: ALSPAC mothers cohort. *International Journal of Epidemiology* 42: 97–110.

Golding, J., M. Pembrey, and R. Jones. 2001. ALSPAC—the Avon longitudinal study of parents and children. *Paediatric and Perinatal Epidemiology* 15: 74–87.

Goldstein, H., and D. Kounali. 2009. Multilevel multivariate modelling of childhood growth, numbers of growth measurements and adult characteristics. *Journal of the Royal Statistical Society, Series A* 172: 599–613.

Gutierrez, R. G., S. Carter, and D. M. Drukker. 2001. sg160: On boundary-value likelihood-ratio tests. *Stata Technical Bulletin* 60: 15–18. Reprinted in *Stata Technical Bulletin Reprints*, vol. 10, pp. 269–273. College Station, TX: Stata Press.

Kohler, U., and S. Eckman. 2011. Stata tip 103: Expressing confidence with gradations. *Stata Journal* 11: 627–631.

Leckie, G., and C. Charlton. 2011. runmlwin: Stata module for fitting multilevel models in MLwiN software package. Centre for Multilevel Modelling, University of Bristol. http://www.bristol.ac.uk/cmm/software/runmlwin/.

———. 2013. runmlwin—a program to run the MLwiN multilevel modelling software from within Stata. *Journal of Statistical Software* 52: 1–40.

Macdonald-Wallis, C., K. Tilling, A. Fraser, S. M. Nelson, and D. A. Lawlor. 2013. Gestational weight gain as a risk factor for hypertensive disorders of pregnancy. *American Journal of Obstetrics and Gynecology* 209: 327.e1–327.e17.

Macdonald-Wallis, C. M., D. A. Lawlor, T. M. Palmer, and K. Tilling. 2012. Multivariate multilevel spline models for parallel growth processes: Application to weight and mean arterial pressure in pregnancy. *Statistics in Medicine* 31: 3147–3164.

Morris, J. S. 2002. The BLUPs are not "best" when it comes to bootstrapping. *Statistics & Probability Letters* 56: 425–430.

Palmer, T., and C. Macdonald-Wallis. 2012. reffadjust: Stata module to estimate adjusted regression coefficients for the association between two random effects variables. Statistical Software Components S457403, Department of Economics, Boston College. http://ideas.repec.org/c/boc/bocode/s457403.html.

Pan, H., and H. Goldstein. 1997. Multi-level models for longitudinal growth norms. *Statistics in Medicine* 16: 2665–2678.

———. 1998. Multi-level repeated measures growth modelling using extended spline functions. *Statistics in Medicine* 17: 2755–2770.

Rasbash, J., C. Charlton, W. J. Browne, M. Healy, and B. Cameron. 2009. MLwiN version 2.1. Centre for Multilevel Modelling, University of Bristol. http://www.bristol.ac.uk/cmm/software/mlwin.

Raudenbush, S. W., and A. S. Bryk. 2002. *Hierarchical Linear Models: Applications and Data Analysis Methods.* 2nd ed. Thousand Oaks, CA: Sage.

Robinson, G. K. 1991. That BLUP is a good thing: The estimation of random effects. *Statistical Science* 6: 15–32.

Royston, P. 1995. Calculation of unconditional and conditional reference intervals for foetal size and growth from longitudinal measurements. *Statistics in Medicine* 14: 1417–1436.

StataCorp. 2013. *Stata 13 Multilevel Mixed-Effects Reference Manual.* College Station, TX: Stata Press.

Tilling, K., J. A. C. Sterne, and C. D. A. Wolfe. 2001. Multilevel growth curve models with covariate effects: Application to recovery after stroke. *Statistics in Medicine* 20: 685–704.

About the authors

Tom Palmer is an assistant professor in medical statistics. He is the author of the `confunnel` and `bpbounds` commands and coauthor of the `winbugsfromstata` package.

Corrie Macdonald-Wallis is a research associate in biostatistics. Her PhD thesis is titled "Maternal blood pressure change in pregnancy".

Debbie Lawlor is a professor of epidemiology with research interests in life course and genetic epidemiology of women's reproductive health, in diabetes and cardiovascular disease, and in methods for improving causal inference in observational epidemiology.

Kate Tilling is a professor of medical statistics with research interests in longitudinal and multilevel modeling, survival analysis, and causal inference.