The MLA Program for Two-Level Analysis with Resampling Options

Frank M. T. A. Busing

Erik Meijer

Rien van der Leeden

Abstract

Estimation in multilevel models is ususally based on maximum likelihood (ML) theory. In practice, however, the assumptions underlying ML theory, such as a sufficiently large sample size and normally distributed error terms, are often violated. Resampling methods, in particular the bootstrap and jackknife, provide a way to study the influence of the violation of assumptions upon the estimation results. The computer program MLA for two-level analysis has been especially developed to study the implementation of resampling methods for multilevel models. Some ideas of resampling regression models can be used but adaptations had to be made for hierarchically structured data. In MLA common estimates for multilevel models are obtained by a fast Quasi-Newton method that estimates all parameters jointly. Additionally, MLA provides several simple non-iterative estimators which may be considered as alternatives. The program offers a user-friendly interface. In this paper MLA is introduced. Results of a limited simulation study in which resampling-estimators and ML estimators are compared will be discussed as well. Although the main reason for developing the program was to use it for our own research purposes, for other researchers the MLA program may also be a useful addition to the multilevel analysis software currently available.

1 Introduction

In the social and behavioral sciences, multilevel analysis has by now been established as a powerful technique for the analysis of hierarchically structured data. Usually, the technique amounts to fitting hierarchically formulated linear regression models, often referred to as multilevel models. The parameters of these models are typically obtained by means of a maximum likelihood procedure. Today, several computer programs are available especially designed for fitting multilevel models. The MLA program contributes to this already existing software by providing, among other things, several resampling options. In this paper, the MLA program is introduced. We will start with a brief discussion of some topics relevant to multilevel analysis in general.

1.1 *Hierarchical data*

Hierarchically structured data arise in a variety of research areas. Such data are characterized by so-called "nested" membership relations among the units of observation. Classical examples of hierarchically structured data are found in educational research where, for instance, students are nested within classes and classes are nested within schools. Other examples are, for instance in (clinical) psychology, where clients are nested within therapy groups, people nested within families or employees nested within organisations.

With hierarchical data, it is common to have information obtained from the different *levels* in the hierarchy. For instance, one has variables describing the individual students, but also variables describing their schools.

Nested structures naturally arise where explicit hierarchical sampling schemes are used. This is often the case in large scale educational research where, for instance, a set of schools is sampled first, followed by the sampling of a set of students within these schools. However, in many other cases where data are not explicitly sampled that way, it appears to be a fruitful approach to treat them as having a hierarchical structure. For instance, in a medical study it may be considered important to view patients as nested within general practitioners. Apart from that, there are several types of data for which it proves to be very useful to apply the concept of hierarchy, because it makes their analysis more easy and transparent. One example is the hierarchical treatment of repeated measures data, where measurements at different points in time are considered nested within individuals. Another example is the analysis of data from meta analysis, where, say, p-values can be treated as being nested within studies, providing a (partial) solution for the problem of comparing apples with oranges.

In nested data the strength of the hierarchical structure is reflected in the *intra-class correlation*, defined as the proportion of the total variance that is between the groups.

1.2 *Multilevel models*

Multilevel models can be conceived as linear regression models, specified separately for each level of the hierarchy, but statistically connected. For comprehensive discussions on the theory and application of these models, mainly within the context of social and behavioral science research, we refer to the textbooks by Goldstein (1987, 1995), Bryk and Raudenbush (1992), and Longford (1993). Throughout the literature multilevel models are referred to under various names. Although there are minor differences, these models are all basically the same.

For each specific level in the hierarchy, predictor variables may be included in the appropriate level model. Suppose we have data of students nested within schools, and we want to predict the score on a math test from the amount of time spent on doing math

homework. Furthermore, we expect smaller schools to be more effective than larger ones, so we collect the school size as another variable. Clearly, at the student level, 'math' is the dependent variable, and 'homework' is the predictor variable. At the school level, 'size' may be considered as a predictor variable for the regression parameters estimated within the schools.

At first glance, the situation presented above seems to lead to a hierarchically structured regression procedure, which proceeds in two steps: First, the models for all schools are estimated, and then the intercept and slope estimates are used as the dependent variables in the Level-2 model, which is estimated subsequently. Although such procedures have been proposed in the past, this is not what will be discussed here under the heading of multilevel models, because there is no statistical connection between the Level-1 and Level-2 models. In multilevel models, separate regression equations for each level are only formulated because they facilitate insight and understanding. The statistical linkage of both levels is established by the Level-2 model which states that Level-1 regression coefficients—intercepts and slopes—are treated as *random variables* at the second level. The Level-2 model models intercept and slope estimates as a mean value over all schools plus a school-specific deviation or residual. It follows that we are not primarily looking for intercept and slope estimates for each separate school, but for their means and variances (and their covariance) over all schools. In this way, just as students are considered a sample from a population of students, schools are considered a sample from a population of schools.

Theoretically, we can model as many levels as we want. In practice, however, most applications of multilevel analysis concern problems with two or three levels. In fact, a majority of applications just concerns two-level data and can be viewed as "within-and-between-analysis" problems. Models with more than three levels are increasingly complex to interpret and should be limited to rather simple cases.

1.3 Random regression coefficients

In multilevel modeling, we are usually not looking for estimates of the regression coefficients within each separate group, but for their variances and covariances. However, there can be circumstances in which we still want to obtain the "best" estimates for these coefficients (Bryk and Raudenbush (1992) call these coefficients *random Level-1 coefficients*). The first thing that comes to mind is to simply estimate them by a separate (OLS) regression for each group. However, this procedure has the serious disadvantage that the coefficients will not be estimated with the same precision for each group, caused by, for instance, differences in the number of units per group.

Within the framework of multilevel analysis the way to obtain best estimates for these

coefficients is a method called *shrinkage* estimation. The underlying idea of this method is that there are basically two sources of information: the estimates from each group separately and the estimates that could be obtained from the total sample, ignoring any grouping. Shrinkage estimation consists of a weigted combination of these two sources. The more reliable the estimates are within the separate groups, the more weight is put upon them. Vice versa, the less reliable these estimates are, that is, the less precise, the more weight is put upon the estimates obtained from the total sample. The result is that estimates are "shrunken" towards the mean of the estimates over all groups. The amount of shrinkage depends on the reliability of the estimates from the separate groups. The less precise the estimates are, the more they are "shrunken" towards the mean over all groups.

1.4 Estimation and software

Fitting a multilevel model actually amounts to fitting one, combined model instead of separate models for different levels. Multilevel models can be viewed as special cases of the general mixed linear model (cf. Harville, 1977). Such models are characterized by a set of fixed parameters and a set of random parameters or variance components, associated with the fixed and random parts of the model, respectively.

To obtain estimates for the parameters, several estimation procedures have been proposed. These procedures are all versions, in one way or another, of full information (FIML) or restricted maximum likelihood (REML). FIML and REML estimators have several attractive properties, such as consistency and efficiency. A drawback of both approaches, however, is their relative complexity. Generally, parameter estimates must be obtained iteratively and serious computational difficulties may arise during such procedures.

Three major packages for multilevel analysis are MLn (Rasbash & Woodhouse, 1995), VARCL (Longford, 1990) and HLM (Bryk, Raudenbush, Seltzer, & Congdon, 1988), although multilevel models can also be estimated with BMDP (BMDP-5V procedure, Schluchter, 1988), SAS (MIXED procedure, SAS Institute, 1992), and GENMOD (a program based on the work of Mason, Wong, & Entwistle, 1983). The three major packages use different methods for maximizing the likelihood. In MLn an *Iterative Generalized Least Squares* (IGLS) procedure is implemented (Goldstein, 1986), and a restricted version of IGLS (RIGLS; Goldstein, 1989). VARCL uses Fisher scoring (Longford, 1987) and HLM uses the EM algorithm (Dempster, Laird, & Rubin, 1977; Bryk & Raudenbush, 1992). A comparative study of several of these programs is given by Kreft, De Leeuw, and Van der Leeden (1994).

1.5 *Why another program for multilevel analysis?*

In this paper we introduce a new program for multilevel analysis, called MLA. This program is designed to analyze data with a two-level hierarchical structure. It has been especially designed for our own research purposes, in particular to study estimation using several resampling methods. As such, the program reflects our research interests. Nevertheless, MLA contains some specific features that could make it a useful addition to the available software for multilevel analysis, including MLn, VARCL and HLM.

Simulation options

The MLA program provides several options for simulation, in particular four different methods for bootstrap estimation. In our opinion such features can be very useful analyzing empirical data sets because in practice the assumptions underlying maximum likelihood theory will often be violated. One problem applying maximum likelihood theory to multilevel modelling, is that estimation of parameters and standard errors, as well as hypothesis testing, rely on large sample theory, and, unfortunately, little is known about the behavior of the estimates when sample size is small (Raudenbush, 1988). Another problem is that it is usually assumed that the error terms are normally distributed. If this is not the case, it may have other undesirable consequences for using standard errors for hypothesis testing and construction of confidence intervals. Bootstrap estimation can be a valuable approach to assess and improve the quality of estimates of parameters and standard errors. It may give information supplementary to the insights already provided by an increasing number of simulation studies based on artificially generated data (Busing, 1993; Van der Leeden & Busing, 1994; Kreft, 1994). The bootstrap methods will be discussed in section 3. The jackknife methods are described in Busing, Meijer, and Van der Leeden (1994, section 2.8.1).

Alternative simple estimation methods

Usually, complicated iterative estimation procedures are used to estimate the parameters of multilevel models. From a theoretical and technical point of view, these procedures provide the best estimates that can be obtained. However, in practice, some of the algorithms used may be rather slow under certain conditions. In other cases serious computational difficulties may arise that are not easy to overcome. De Leeuw and Kreft (1994) discuss alternative estimation procedures for both fixed and random parameters in multilevel models that are non-iterative and relatively easy to implement. Moreover, in certain cases the quality of the parameter estimates is rather good. Hence, one could study the real gain of the complicated iterative procedures over these simpler alternatives (Van der Leeden & Busing, 1994). Therefore, in MLA, we have implemented a one-step

and a two-step OLS procedure, and we are working on a non-iterative WLS procedure.

Simple procedures can always be used additionally to complex ones, and vice versa. It depends on the structure of the data which estimation procedures are to be preferred (De Leeuw & Kreft, 1994; Kreft, 1994).

Fast maximum likelihood algorithm

To optimize the likelihood function, the MLA program uses the *Broyden-Fletcher-Goldfarb-Shanno* (BFGS) variant of the Davidon-Fletcher-Powell method (Press, Flannery, Teukolsky, & Vetterling, 1986). This is a fast and stable Quasi-Newton method to optimize arbitrary functions. It requires programming of the function and the gradient (the vector of first derivatives) of the function with respect to the parameters. It optimizes the function with respect to both fixed and random parameters simultaneously. As such, it resembles most the algorithm used by VARCL, although the BFGS method does not compute the inverse of the information matrix at each iteration. The algorithms of MLn and HLM alternately update the fixed and the random parameters.

2 The model and its estimation

2.1 The model

In MLA, the following general two-level model is implemented. Suppose data are obtained from N individuals nested within J groups, with group j containing N_j individuals. Now, for group j ($j = 1, \ldots, J$), y_j is a vector containing values on an outcome variable, Z_j is an $N_j \times q$ matrix with explanatory variables (including the constant), β_j is a vector of regression coefficients, and ε_j is a vector with random error terms (vectors and matrices of appropriate dimensions). Then, for each group j, the Level-1 or within-group model can be written as

$$y_j = Z_j\beta_j + \varepsilon_j. \tag{1}$$

It is assumed that β_j is a vector of *random* regression coefficients and that group level variables W_j may exist that explain part of the variation in the random coefficients. Then, the Level-2 or between-group model can be written as

$$\beta_j = W_j\gamma + u_j, \tag{2}$$

where W_j is a $q \times p$ matrix with explanatory variables (including the constant) obtained at the group level, γ is a vector containing fixed coefficients and u_j is a vector with random error terms.

Substitution of Equation (2) into Equation (1) gives the total,"combined" model

$$y_j = X_j\gamma + Z_ju_j + \varepsilon_j. \tag{3}$$

in which $X_j = Z_j W_j$. This is a *mixed linear model* (Harville, 1977).

Generally, it is assumed that $\varepsilon_j \sim N(0, \sigma_\varepsilon^2 I_{N_j})$ and $u_j \sim N(0, \Theta)$, where σ_ε^2, the variance of the Level-1 error term, is an unknown (scalar) parameter, and Θ, the covariance matrix of the Level-2 error terms, is a (symmetric) matrix of unknown parameters. The expectation of y_j conditional on X_j and Z_j is $E(y_j|X_j, Z_j) = X_j\gamma$. The covariance matrix V_j of y_j conditional on X_j and Z_j, that is, the matrix containing the variances and covariances of the random part $Z_j u_j + \varepsilon_j$ in Equation (3) conditional on Z_j, is expressed as

$$V_j = E(y_j - X_j\gamma)(y_j - X_j\gamma)' = Z_j\Theta Z_j' + \sigma_\varepsilon^2 I_{N_j}. \tag{4}$$

The parameters of the model that have to be estimated are the fixed coefficients (elements of the vector γ), the covariance matrix Θ of the random components u, and the variance σ_ε^2 of the errors ε. The elements of γ are called the *fixed parameters*, and σ_ε^2 and the elements of Θ are called the *random parameters*.

2.2 *Estimation*

The model parameters are usually estimated by *full information maximum likelihood* (FIML). This estimation method has desirable properties, such as consistency and efficiency. Given the model assumptions (such as normality of the random error components), the density of y_j, given X_j and Z_j, is

$$f(y_j|X_j, Z_j) = \frac{1}{(2\pi)^{N_j/2}(\det V_j)^{1/2}} e^{-\frac{1}{2}(y_j - X_j\gamma)'V_j^{-1}(y_j - X_j\gamma)},$$

so that the contribution of Level-2 unit j to the minus-log-likelihood function is

$$\begin{aligned} L_j &= -\log f(y_j|X_j, Z_j) \\ &= \frac{N_j}{2}\log(2\pi) + \frac{1}{2}\log\det V_j + \frac{1}{2}(y_j - X_j\gamma)'V_j^{-1}(y_j - X_j\gamma) \end{aligned}$$

and the minus-log-likelihood function for the whole sample is simply the sum of all Level-2 units j, $L = \sum_{j=1}^{J} L_j$. This is the function that is minimized with respect to the parameters to obtain maximum likelihood estimators. Specifically, it will produce a set of fixed parameter estimates, $\hat{\gamma}$, and a set of random parameter estimates, $\hat{\Theta}$ for the second level and $\hat{\sigma}_\varepsilon^2$ for the first level. Details can be found in Busing et al. (1994).

The asymptotic covariance matrix of the estimators is obtained from the minus-log-likelihood function in the same way as with other maximum likelihood estimators (see Busing et al., 1994, pp. 60–61). This covariance matrix is used to obtain the standard errors for both fixed and random parameters.

A detailed description of all derivations used here and an extensive discussion of the computational formulas used in the program can be found in Appendix A in Busing et al. (1994).

Van der Leeden and Busing (1994) showed that, especially with small sample sizes, simple noniterative estimators of the parameters may be good alternatives for maximum likelihood estimators (see also De Leeuw & Kreft, 1994). Therefore, besides maximum likelihood, the MLA program has several options for least squares estimation. The *within-group* OLS estimators are simply the OLS estimators of β_j within each group j,

$$\hat{\beta}_j^{\text{OLS}} = (Z_j'Z_j)^{-1}Z_j'y_j. \tag{5}$$

The *one-step* OLS estimator of γ is the OLS estimator where the dependencies among subjects from the same group are ignored,

$$\hat{\gamma}^{\text{OLS}} = (X'X)^{-1}X'y, \tag{6}$$

where $X = (X_1', \ldots, X_J')'$ and $y = (y_1', \ldots, y_J')'$.

Using estimates (8) and (7) (see section 2.3 below) for the residuals, *two-step* OLS estimators of the random parameters are obtained by $\hat{\Theta} = \sum_{j=1}^{J} \hat{u}_j\hat{u}_j'/J$ and $\hat{\sigma}_\varepsilon^2 = \sum_{i=1}^{N} \hat{\varepsilon}_i^2/N$.

Formulas for standard errors of the OLS parameter estimators can be found in Busing et al. (1994, pp. 14–15).

2.3 *Residuals*

The residuals ε_j and u_j in (1) and (2) can be estimated for diagnostic purposes. They can be used for two-step OLS estimators (see section 2.2) and they are required for bootstrap estimation (see section 3). There are (at least) two ways in which the residuals can be estimated: by simple OLS decomposition (*raw residuals*), and by the method of *shrinkage*.

The Level-2 *raw residuals* are

$$\hat{u}_j^{\text{OLS}} = (Z_j'Z_j)^{-1}Z_j'\hat{r}_j \tag{7}$$

where $\hat{r}_j = y_j - X_j\hat{\gamma}$, and $\hat{\gamma}$ is the FIML estimator of γ. The Level-1 raw residuals are

$$\hat{\varepsilon}_j^{\text{OLS}} = \hat{r}_j - Z_j\hat{u}_j^{\text{OLS}} \tag{8}$$

The raw residuals are unbiased, though inefficient, estimators.

The Level-2 *shrunken residuals* are given by

$$\hat{u}_j^{\text{SHR}} = (Z_j\hat{\Theta})'[Z_j\hat{\Theta}Z_j' + \hat{\sigma}_\varepsilon^2 I_{N_j}]^{-1}\hat{r}_j. \tag{9}$$

where $\hat{r}_j$ as above, and $\hat{\Theta}$ and $\hat{\sigma}_\varepsilon^2$ are the FIML estimators of Θ and σ_ε^2, respectively. The Level-1 shrunken residuals are

$$\hat{\varepsilon}_j^{\text{SHR}} = \hat{r}_j - Z_j\hat{u}_j^{\text{SHR}}. \tag{10}$$

The shrunken residuals are the expected values of the residuals given the data of all groups if the estimates of the parameters are the true values. The shrunken residuals are (asymptotically) more efficient than the raw residuals, but they may not reflect the true variation in the residuals. Formula (9) is computationally rather inefficient. In the program a more efficient formula is used.

The shrunken residuals can also be used to obtain estimates of the (random) regression coefficients β_j for each Level-2 unit. These estimators are called the *posterior means*, because they are the expectations of the coefficients given the maximum likelihood estimates and the data of all Level-2 units. Their formula is

$$\widehat{\beta}_j^{\mathtt{SHR}} = W_j\widehat{\gamma} + \widehat{u}_j^{\mathtt{SHR}},$$

where $\widehat{\gamma}$ is the FIML estimator of γ and $\widehat{u}_j^{\mathtt{SHR}}$ contains the shrunken residuals.

3 *Simulation*

Maximum likelihood estimation theory is based on several assumptions. Two important ones are (1) The Level-1 (ε) and Level-2 (u) random errors are normally distributed. The likelihood function is derived under this assumption, and therefore, the FIML estimators and the estimators of their standard errors depend on it. (2) The sample size is (sufficiently) large. More specifically, the properties of the maximum likelihood estimators, such as their consistency, their (asymptotic) efficiency, and their (asymptotic) normal distribution, as well as the formulas for their standard errors are derived under the assumption that the sample size goes to infinity ($N \longrightarrow \infty$).

In practice, these assumptions will not be completely satisfied. One can only hope that they are met approximately. To be able to get an indication of how severe the finite sample size and possible nonnormality influence the results, the MLA program offers simulation options. In this section, these simulation options will be described. The focus will be on the possible bias of the estimates and on the possibly incorrect standard errors. More subtle information can, however, be extracted from the program by using an output file for the simulation results.

In many models and situations maximum likelihood estimators are biased in finite samples. For a general class of regression models including multilevel models, however, Magnus (1978) proved that the maximum likelihood estimators of the *fixed* regression coefficients are unbiased. On the other hand, Busing (1993) showed in a Monte Carlo simulation study that the maximum likelihood estimators of the *random* parameters in multilevel models are biased (see also Breslow and LinBreslow and Lin (1995)).

The standard errors of the maximum likelihood estimators that are reported by MLA are derived from asymptotic theory. This means that they are based on the idea that

as the sample size goes to infinity, the distribution of the estimators will converge to a (multivariate) normal distribution with a certain covariance matrix (see Appendix A of Busing et al., 1994). The reported standard errors are the square roots of the diagonal elements of this matrix. The exceedance probabilities of the according t-values that are reported are based on the approximation of the distribution of the estimators by the normal distribution. In finite samples, this approximation may not be very good. The true standard errors may be quite different from the reported ones based on asymptotic theory, and the distributions of the estimators may not be normal. In fact, Busing (1993) showed in his simulation study that the distributions of the random parameters can be severely skewed. As mentioned above, however, the focus is on the bias and the standard errors and not on the specific distribution.

3.1 The bootstrap

The idea of the *bootstrap* is that the empirical distribution function $\widehat{F}_N$ is a consistent estimator of the distribution function F in the population. The sampling and estimation process is simulated by drawing samples with replacement from $\widehat{F}_N$, which is completely known once the original sample is obtained. In the simulation, the distribution $\widehat{F}_N$ plays the role of F, and, if θ is the parameter of interest, $\widehat{\theta}$ plays the role of θ. Simulation samples are drawn from $\widehat{F}_N$ and $\widehat{\theta}$ is estimated by θ^* in the same way θ was estimated by $\widehat{\theta}$. Because $\widehat{F}_N \longrightarrow F$, it is assumed that the properties of the estimator θ^* based on the distribution $\widehat{F}_N$ give information about the properties of $\widehat{\theta}$ based on the distribution F.

The bootstrap is implemented as follows: B bootstrap samples $\{z^*_{b1}, z^*_{b2}, \ldots, z^*_{bN}\}$, $b = 1, \ldots, B$, are drawn from $\widehat{F}_N$, that is, these samples are drawn with replacement from $\{z_1, z_2, \ldots, z_N\}$, being a random sample of size N from F. From each of the B samples, the parameter $\widehat{\theta}$ is estimated, thereby obtaining B estimators θ^*_b, $b = 1, \ldots, B$. Now the expectation of θ^* (given $\widehat{F}_N$) is estimated by the mean of the estimators θ^*_b, that is, $\theta^*_{(.)} = \sum_{b=1}^{B} \theta^*_b / B$. The variance of θ^* (given $\widehat{F}_N$) is estimated by the variance of the estimators θ^*_b, that is, $\widehat{\mathrm{var}}(\theta^*) = \sum_{b=1}^{B} (\theta^*_b - \theta^*_{(.)})^2/(B-1)$.

The bias of $\widehat{\theta}$ is estimated by the (estimated) bias of θ^*:

$$\widehat{\mathrm{bias}}_B = \widehat{\mathrm{bias}}(\theta^*) = \theta^*_{(.)} - \widehat{\theta}, \tag{11}$$

and the bias-corrected estimator of θ is therefore

$$\begin{aligned} \widehat{\theta}_B &= \widehat{\theta} - \widehat{\mathrm{bias}}_B \\ &= 2\widehat{\theta} - \theta^*_{(.)}. \end{aligned} \tag{12}$$

The standard error of $\widehat{\theta}$ is simply estimated by the square root of the variance of θ^*_b:

$$\widehat{\mathrm{se}}_B = \sqrt{\widehat{\mathrm{var}}(\theta^*)} = \sqrt{\frac{1}{B-1} \sum_{b=1}^{B} \left(\theta^*_b - \theta^*_{(.)}\right)^2}. \tag{13}$$

The bootstrap is essentially a *nonparametric* method because samples are drawn from the nonparametric empirical distribution function $\widehat{F}_N$. However, if we are willing to make a distributional assumption about F, we can define a *parametric bootstrap*. If this stronger assumption is met, we benefit from the fact that the *parametric* empirical distribution function $\tilde{F}_N$ will generally be a more efficient estimator of F. The parametric bootstrap is defined exactly analogous to the nonparametric bootstrap, except that bootstrap samples are drawn from $\tilde{F}_N$ instead of $\widehat{F}_N$. This means that samples with replacement are drawn from a generally more smooth distribution function, instead of from the original data. Hence, the values of the z_{bi}^* in the bootstrap sample will usually not be values also encountered in the original sample.

3.2 Resampling multilevel models

Bootstrap methods can not be straightforwardly applied to multilevel models, because they require the observations to be independently distributed. Clearly this is not the case with hierarchical data, where the observations are subject to intraclass dependency. Therefore, special resampling schemes have to be devised.

The *parametric bootstrap* can easily be implemented in multilevel analysis. If the Z_j and W_j variables are considered fixed in (1) and (2), bootstrap samples $\{y_{b1}^*, \ldots, y_{bJ}^*\}$ can be obtained in the following way. First, for each $j = 1, \ldots, J$, draw a bootstrap Level-2 error vector u_j^* from a normal distribution with mean zero and covariance matrix $\widehat{\Theta}$. Then, draw a bootstrap Level-1 error vector ε_j^* from a normal distribution with mean zero and covariance matrix $\widehat{\sigma}_\varepsilon^2 I_{N_j}$. Finally, the bootstrap sample of y is obtained from

$$\beta_j^* = W_j\widehat{\gamma} + u_j^* \text{ and } y_j^* = Z_j\beta_j^* + \varepsilon_j^*. \tag{14}$$

Then, bias-corrected bootstrap estimators and bootstrap estimators of the covariance matrix of the parameters are obtained in the usual way.

For the *nonparametric bootstrap*, several situations can be studied. If the explanatory variables can be considered *fixed* design variables, then, analogously to regression analysis, the *errors* have to be estimated and subsequently resampled (see, e. g. Efron, 1982, pp. 35–36). As explained in section 2.3, the shrunken residuals (9) and (10) can be used as estimators of the Level-2 and Level-1 errors, respectively, or, alternatively, the *raw residuals* (7) and (8) can be used instead of the shrunken residuals.

Unlike in regression analysis, the estimated residuals in multilevel analysis do not necessarily have a zero mean. Therefore, the residuals are centered first. Otherwise, the possibly nonzero mean of the errors would necessarily lead to biased estimators of the constant. Once (centered) estimates $\{\widehat{u}_j\}$, $j = 1, \ldots, J$, and $\{\widehat{\varepsilon}_{ij}\}$, $j = 1, \ldots, J$, $i = 1, \ldots, N_j$, of the errors are obtained, nonparametric bootstrap samples $\{u_j^*\}$, $j = 1, \ldots, J$, and $\{\varepsilon_{ij}^*\}$,

$j = 1, \ldots, J$. $i = 1, \ldots, N_j$ are drawn, and nonparametric bootstrap samples of y are obtained from (14). Then, estimators can be obtained in the usual way, and bootstrap bias-corrected estimators and standard errors can be obtained straightforwardly. This bootstrap procedure in which estimated errors are resampled, is called the *error bootstrap*. In section 5 we discuss results of a limited study to assess the performance of both versions of the error bootstrap.

If the Z and W variables are considered *random*, nonparametric bootstrap samples can be drawn by resampling complete *cases*. The bootstrap samples can be drawn in the following way. First, a sample of size J is drawn with replacement from the *Level-2 units*. This gives a sample j_k^*, $k = 1, \ldots, J$ of Level-2 unit numbers and accompanying Level-2 variables $W_{j_k^*}$. Then for each k, a nonparametric bootstrap sample of complete cases from the (original) unit $j = j_k^*$ is drawn, giving $\{(y_{ik}^*, Z_{ik}^*), k = 1, \ldots, J, i = 1, \ldots, N_{j_k^*}\}$. This is called the *cases bootstrap* for both levels. It depends on the nature of the data if it makes sense to resample units from both levels, or only from Level-2 or Level-1. For instance, when we have repeated measurements nested within individuals, the best approach will be to resample Level-2 units (individuals) only.

Besides bootstrap methods, two experimental *jackknife* (Efron, 1982) methods are implemented in MLA as well (see Busing et al., 1994).

4 *The interface*

To give an impression of the interface of the MLA program, in this section a small example of an input file and the corresponding output file will be given. An extensive discussion of all options is given by Busing et al. (1994).

The MLA program uses an *input-file-name* and an *output-file-name* as parameters. Both files are simple ascii files. The first parameter is the input file and the second the output file. The program can be started by the command

MLA *input-file-name output-file-name*

The input file consists of case insensitive statements, substatements and keywords. Every statement consists of a keyword preceded by a slash (e. g., /TITLE). The substatements must appear on the line(s) thereafter and should precede the next statement. Statements within the input file and substatements within statements may appear in any order, except for the /END statement. This is the last statement to be read. In the following section, an input file will be annotated. A similar file was used in a simulation study of which results are discussed in the next section.

4.1 *Input*

For the simulation study, an input file was used to perform error bootstrap analysis with shrunken residuals.

The first statement is the /TITLE statement. Although this statement is optional and in our simulation study rather superfluous, in 'single analysis' research it is highly recommended. In contrast to comments, the title is repeated on top of every part of the output and covers the first non-empty line following the /TITLE statement. Comments, preceded by a percent sign (%), may appear throughout the input file. All text on a line, after and including the percent sign is considered comment and will be ignored as program input.

```
/TITLE
  simulation study 04-95; shrunken residuals bootstrap
```

The /DATA statement contains information about the data file. The name of the data file is given following the keyword `file`. The name is written after an equals sign and must satisfy the usual DOS conventions on filenames. The file itself is a free-field formatted numbers-only `ascii` file. In this file, one case may consist of more than one line and cases must be sorted by the Level-2 identifier variable. This must be one of the variables in the data file and must contain a code (number) that identifies the Level-2 units. Level-1 units are interchangeable within a Level-2 unit. The number of the identifier variable has to follow the keyword `id2` and it must indicate the position of this variable in the data file. The `variables` substatement specifies the total number of variables in the data file. Here, the data are read from `mls.out` with a total of 8 variables, the first variable serving as Level-2 identifier.

```
/DATA
  file = mls.out
  vars = 8
  id2 = 1
```

The /MODEL statement is followed by a set of equations that specifies the model to be estimated. Here, the model specified is a multilevel model with one predictor at each level. The Level-1 equation is given by

$$Y_{ij} = \beta_{1j} + \beta_{2j} Z_{ij} + \varepsilon_{ij},$$

and the Level-2 equations are given by

$$\beta_{1j} = \gamma_1 + \gamma_2 W_j + u_{1j},$$
$$\beta_{2j} = \gamma_3 + \gamma_4 W_j + u_{2j}.$$

In the input file every equation must appear on a single line. The above equations are simply copied and written by `ascii` characters.

```
/MODEL
  v4 = b1 + b2*v6 + e
  b1 = g1 + g2*v7 + u1
  b2 = g3 + g4*v7 + u2
```

The term v*n* denotes the *n*-th variable in the data file, v*n* either indicating a predictor variable or the response variable (Z_{ij}, W_j or Y_{ij} respectively). b*k* is beta component *k* (β_{kj}, the *k*th element of a typical β_j) and e is the Level-1 random term (ε_{ij}). The variance of this term is to be estimated from the data. g*l* is element *l* of γ (γ_l), the vector containing the fixed parameters to be estimated in the multilevel model. Finally, u*k* is Level-2 random term *k* (u_{kj}, the *k*th element of a typical u_j). The variances and the covariances of these terms are to be estimated from the data. In the equations, each term is followed by a number (except for the Level-1 random term e). For the v*n* term, this number is the variable number, the position of the variable in the data file (e. g., the response variable, v4, the fourth variable in the data file). The other terms only use a number for identification, without any additional meaning.

With the substatements provided with the `/SIMULATION` statement, one can choose between the different kinds of simulation (using the keyword `kind`), and specify special simulation features (using the keywords `method`, `type` and `resample`). Additional features are the number of replications and the initial seed for the random number generator (`replications` and `seed`). Finally, one can specify a separate output file for intermediate results of the simulation (`file`). The input file described here was used to perform a bootstrap analysis with resampling of the shrunken residuals. The number of replications was set to 100.

```
/SIMULATION
  kind = bootstrap
  method = error
  type = shrunken
  replications = 100
```

The substatement `kind` sets the kind of simulation to "bootstrap". The user may also choose jackknife simulation. Choosing bootstrap simulation commits the user to choose between three bootstrap methods (see section 3). In this case, error bootstrap was chosen and not cases bootstrap or parametric bootstrap. The choice of error bootstrap forces the user to make another choice. As explained in section 3, MLA performs two types of error bootstrap, using either raw or shrunken residuals. For this example, the shrunken residuals were chosen.

As indicated before, the last statement read is the `/END` statement.

```
/END
```

Information following this statement is not even read. After reading the `/END` statement, MLA will read the data and start the analysis.

In the following we discuss the output file resulting from running the program using the input file described above.

4.2 Output

The output of MLA consists of a single text file, the second parameter in the statement that starts program execution. The output file described here does not contain all possible output MLA is able to produce. For a complete reflection on the output of MLA see Busing et al. (1994).

The data used for this example were generated with MLS, a mixed linear two level model sampling utility written by Busing (1994). The input for MLS gathers all necessary parameters to specify a two level model, including sample sizes and distributions for the predictor variables. The output of MLS was written to `mls.out`.

All MLA output contains the MLA title page. It is the first part of the output. It only supplies information about the name and origin of the program.

Except for the title page, every part contains a header. The header is always the same and includes two lines of standard text and the title of the analysis, supplied by the user.

The second part of the output contains an echo of the input file statements. This part is always included in the output file. A digested reflection of the input file is supplied whenever the keyword `input` is used in the `/OUTPUT` statement.

The next part contains the FIML estimates, with associated standard errors, t-values, and exceedance probabilities (p-values) of the t-values. This part is default and appears in all output.

```
FULL INFORMATION MAXIMUM LIKELIHOOD ESTIMATES

FIXED PARAMETERS

   LABEL        ESTIMATE          SE             T          PROB(T)

      G1        1.237551    0.163401          7.57          0.0000
      G2        0.935004    0.173520          5.39          0.0000
      G3        1.454278    0.161045          9.03          0.0000
      G4        0.920952    0.170684          5.40          0.0000

RANDOM PARAMETERS

   LABEL        ESTIMATE          SE             T          PROB(T)

   U1*U1        0.616896    0.187506          3.29          0.0010
   U2*U1        0.410167    0.154530          2.65          0.0079
   U2*U2        0.594809    0.182160          3.27          0.0011

      E        1.089241    0.064290         16.94          0.0000

INTRA-CLASS CORRELATION = 0.6169 / ( 1.0892 + 0.6169 ) = 0.3616
```

```
CONVERGENCE CRITERION REACHED

# ITERATIONS = 8
-2*LOG(L)    = 1944.180943
```

Whenever there are residuals associated with the grand mean, the intra-class correlation is computed and given just below the FIML estimates. A short description of the final iteration results is given in the FIML part. Here, convergence is reached in 8 iterations. The analysis yields a -2*LOG(L) value of 1944.18.

If the /SIMULATION statement is used, the FIML output is followed by the simulation output. In our example, a bootstrap simulation was performed with 100 bootstrap replications. No replications were incorrect (i. e., with inadmissible parameter values or non-convergence). The final bootstrap estimates that were computed are given below.

```
BOOTSTRAP ESTIMATES

  # REPLICATIONS         = 100
  # CORRECT REPLICATIONS = 100

FIXED PARAMETERS

   LABEL       ESTIMATE            SE

      G1       1.210736       0.157299
      G2       0.949960       0.136275
      G3       1.435671       0.152851
      G4       0.926623       0.138406

RANDOM PARAMETERS

   LABEL       ESTIMATE            SE

   U1*U1       0.681441       0.196592
   U2*U1       0.441227       0.116087
   U2*U2       0.683241       0.147255

       E       1.160796       0.057853
```

The final part of the output contains system information and is always shown. This information is the start and finish date and time, the elapsed time and the announcement that the program terminated correctly.

5 *Results of error bootstrap evaluation*

The MLA program was primarily designed for research purposes. The different methods for bootstrap estimation are still being further developed. In order to investigate the properties of the two versions of the nonparametric error bootstrap a limited Monte Carlo study was carried out of which the results are reported here.

Data were generated for a two-level model containing one predictor variable at both levels using MLS (Busing, 1994). The fixed parameters (γ) were set to the neutral value of 1.0, the Level-2 variance components Θ_{11}, Θ_{12}, and Θ_{22} were set to 0.5, 0.25 and 0.5, respectively, and the Level-1 variance component σ_ϵ^2 was set to the value of 1.0. These values correspond to an intraclass correlation of 0.33 and an intercept-slope correlation of 0.5. Predictor variables were drawn from a standard normal distribution. The number of Monte Carlo replications was set to 500. Each replication was followed by a multilevel analysis step using MLA. FIML results for all parameter estimates and their corresponding standard error estimates were saved, as well as error bootstrap results, using either raw or shrunken residuals for resampling. The number of bootstrap replications was set to 100. More details concerning the data generation procedure used here can be found in Busing (1993) and Van der Leeden and Busing (1994).

Conditions that are thought to influence the performance of the bootstrap estimators were varied, that is, small versus large sample results were compared and data were generated from either normal distributions or standardized lognormal distributions with a skewness of 1. In hierarchical data, sample size is hierarchically defined: sample size at each level has to be considered, as well as total sample size. An appropriate variation of J, the number of Level-2 units was chosen, that is, $J = 10$, 25, or 65. For each Level-2 unit, the number of Level-1 units, N_j, was randomly varied with an average value of 10, 25, or 65, where each J was combined with all average N_j's. Thus, a completely balanced design was obtained.

In the presentation of results we focus on the Level-2 variance component estimates. We may expect the quality of estimators corresponding to a particular level to depend upon the sample size at that level (cf. Bryk & Raudenbush, 1992, p. 223). Hence, estimation of the Level-2 variance components will generally be more problematic than the estimation of the Level-1 variance component. Therefore, we will discuss results for these estimators (the elements of Θ) here only.

Table 1 below shows the relative bias (which is defined as $(\overline{\theta}-\theta)/\theta$, where θ is the true value of a parameter, and $\overline{\theta}$ is the mean of its estimates) of the intercept variance estimator $\hat{\Theta}_{11}$ and the slope variance estimator $\hat{\Theta}_{22}$, respectively, for a subset of combinations in the simulation design. Of the total of 9 possible combinations of J and N_j, 6 are considered here. Results were further accumulated for the combinations $J \times N_j = 10 \times 10$, 10×25, and 25×10, which were labeled "small" sample sizes, and for the combinations $J \times N_j = 25 \times 65$, 65×25, and 65×65, which were labeled "large". Results for $\hat{\Theta}_{12}$ were roughly similar and will be omitted here.

Table 1: Relative bias for intercept and slope variance estimators $\hat{\theta}_{11}$ and $\hat{\theta}_{22}$: FIML, raw residual and shrunken residual bootstrap results

Sample size	$\hat{\theta}_{11}$			$\hat{\theta}_{22}$		
Distribution	FIML	Raw	Shrunken	FIML	Raw	Shrunken
Small						
Normal	−0.18	−0.21	0.05	−0.17	−0.23	0.08
Skewed	−0.17	−0.20	0.06	−0.16	−0.24	0.08
Large						
Normal	−0.04	−0.05	−0.01	−0.05	−0.07	−0.02
Skewed	−0.05	−0.06	−0.02	−0.05	−0.07	−0.02

Table 1 shows that the shrunken residual bootstrap estimator is generally superior compared to its raw residual couterpart. For small sample sizes both variance components are slightly over-estimated, whereas large sample sizes show a minimal under-estimation. For small sample sizes the shrunken residual bootstrap estimator is obviously performing better than the FIML estimator, which is subject to serious negative bias. The raw residual bootstrap estimator is even performing worse than the FIML estimator. No substantial differences in results are shown for samples drawn from normal versus skewed distributions. On the whole it can be concluded that bootstrapping is only useful for small sample sizes. For large sample sizes the bootstrap estimators show only marginal improvement over the FIML estimators.

Much more has to be said about the results of this simulation study. We are currently in the process of a thorough analysis of all data obtained. Results of this examination will be presented in future.

6 *Concluding remarks*

In this paper, the MLA program for two-level analysis was introduced. The main features of this program, currently not implemented in other multilevel software, are the extensive resampling options. Although the program was written primarily for our own research purposes, in particular to explore the properties of these resampling options, we think that resampling may be interesting for other researchers as well, and therefore the program is available for other users.

The program is by no means a serious competitor for other multilevel software, because MLA lacks several important possibilities. For example, the number of levels is limited to 2, the program does not provide a possibility to estimate multilevel logistic re-

gression models and other non-standard models, and, unlike MLn, it is not possible to take heteroskedasticity into account. On the other hand, because of the resampling options, it may be a useful addition to the existing software.

The results of the simulation study of the bootstrap estimators are still quite limited in scope. We are currently analyzing the data obtained and we will report more results about it in future. Furthermore, we will perform a similar simulation study to assess the properties of the cases bootstrap as well. Many refinements and extensions of the bootstrap estimators may be needed to fine-tune the estimators to obtain optimal results.

As always, much research has still to be done. We hope that the MLA program, and especially the resampling theory underlying the program, may contribute to theoretical and applied research in the field of multilevel analysis.

Additional information on MLA. The MLA program is developed by Frank Busing, Erik Meijer and Rien van der Leeden and can be obtained from the authors (for addresses see the list of contributors). The distribution disk, including executables and examples, costs $20. Features not described above include summary statistics, possibilities for imposing constraints and several technical setting options. Special output consists of technical information, residuals, posterior means and simple diagnostics. The program can handle up to 16 equations with 32 term each. Other limitations are 16000 Level-2 units, 8000 Level-1 units per Level-2 unit, 16000 variables, 16000 simulation replications and 64 constraints. These limitations are the absolute maxima and can be somewhat lower depending on the amount of memory available. MLA runs on any IBM-PC/AT, PS/2 or compatible under MS-DOS, PC-DOS, DR-DOS or OS/2. A minimum of 256Kb of free RAM is necessary. MLA also runs in a DOS environment under WINDOWS or OS/2.

Acknowledgement

Bram van der Burg is gratefully acknowledged for performing the simulation study.

References

Breslow, N. E., & Lin, X. (1995). Bias correction in generalised linear mixed models with a single component of dispersion. *Biometrika, 82*, 81–91.

Bryk, A. S., & Raudenbush, S. W. (1992). *Hierarchical linear models: Applications and data analysis methods.* Newbury Park, CA: Sage.

Bryk, A. S., Raudenbush, S. W., Seltzer, M., & Congdon, R. T. (1988). *An introduction to HLM: Computer program and user's guide.* University of Chicago.

Busing, F. M. T. A. (1993). *Distribution characteristics of variance estimates in two-level models; A Monte Carlo study* (Tech. Rep. No. PRM 93-04). Leiden, The Nether-

lands: Leiden University, Department of Psychometrics and Research Methodology.

Busing, F. M. T. A. (1994). *MLS* [computer program].

Busing, F. M. T. A., Meijer, E., & Van der Leeden, R. (1994). *MLA. Software for multilevel analysis of data with two levels. User's guide for version 1.0b* (Tech. Rep. No. PRM 94-01). Leiden, The Netherlands: Leiden University, Department of Psychometrics and Research Methodology.

De Leeuw, J., & Kreft, I. G. G. (1994). *Questioning multilevel models* (UCLA Statistics Series No. 143). Los Angeles: University of California.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum likelihood for incomplete data via the EM algorithm. *Journal of the Royal Statistical Society B, 39*, 1–38.

Efron, B. (1982). *The jackknife, the bootstrap and other resampling plans.* Philadelphia: SIAM.

Goldstein, H. (1986). Multilevel mixed linear model analysis using iterative generalized least squares. *Biometrika, 73*, 43–56.

Goldstein, H. (1987). *Multilevel models in educational and social research.* London: Oxford University Press.

Goldstein, H. (1989). Restricted (unbiased) iterative generalized least squares estimation. *Biometrika, 76*, 622–623.

Goldstein, H. (1995). *Multilevel statistical models* (2nd ed.). London: Edward Arnold.

Harville, D. A. (1977). Maximum likelihood approaches to variance component estimation and to related problems. *Journal of the American Statistical Association, 72*, 320–340.

Kreft, I. G. G. (1994). *Are multilevel techniques necessary? An attempt at demystification.* Unpublished manuscript, California State University, School of Education, Los Angeles.

Kreft, I. G. G., De Leeuw, J., & Van der Leeden, R. (1994). A review of five multilevel analysis programs: BMDP-5V, GENMOD, HLM, ML3, VARCL. *American Statistician, 48*, 324–335.

Longford, N. T. (1987). A fast scoring algorithm for maximum likelihood estimation in unbalanced mixed models with nested random effects. *Biometrika, 74*, 817–827.

Longford, N. T. (1990). *VARCL. Software for variance component analysis of data with nested random effects (maximum likelihood).* Princeton, NJ: Educational Testing Service.

Longford, N. T. (1993). *Random coefficient models.* Oxford: Clarendon Press.

Magnus, J. R. (1978). Maximum likelihood estimation of the GLS model with unknown parameters in the disturbance covariance matrix. *Journal of Econometrics, 7*, 281–312.

Mason, W. M., Wong, G. M., & Entwistle, B. (1983). Contextual analysis through

the multilevel linear model. In S. Leinhardt (Ed.), *Sociological methodology* (pp. 72–103). San Francisco: Jossey-Bass.

Press, W. H., Flannery, B. P., Teukolsky, S. A., & Vetterling, W. T. (1986). *Numerical recipes: The art of scientific computing.* Cambridge, UK: Cambridge University Press.

Rasbash, J., & Woodhouse, G. (1995). *MLn. Command Reference, Version 1.0.* London: University of London, Institute of Education.

Raudenbush, S. W. (1988). Educational applications of hierarchical linear models: A review. *Journal of Educational Statistics, 13*, 85–116.

SAS Institute. (1992). *SAS/STAT software: Changes and enhancements, Release 6.07* (SAS Technical Report No. P-229). Cary, NC: Author.

Schluchter, M. D. (1988). *BMDP5V—unbalanced repeated measures models with structured covariance matrices* (Tech. Rep. No. 86). Los Angeles: BMDP Statistical Software.

Van der Leeden, R., & Busing, F. M. T. A. (1994). *First iteration versus final IGLS/RIGLS estimates in two-level models: A Monte Carlo study with ML3* (Tech. Rep. No. PRM 94-02). Leiden, The Netherlands: Leiden University, Department of Psychometrics and Research Methodology.

SB

SSS '95

SYMPOSIUM STATISTISCHE SOFTWARE 1995

Toeval zit Overal:

Programmatuur
voor
Random Coëfficient Modellen

Bundel bij het zevende Symposium Statistische Software
gehouden op 9 november 1995 in de Jaarbeurs te Utrecht

interuniversitair expertisecentrum ProGAMMA
Groningen, november 1995

iec ProGAMMA
Postbus 841
9700 AV Groningen

telefoon: 050 - 363 6900
fax: 050 - 363 6687
e-mail: gamma.post@gamma.rug.nl
www: http://www.gamma.rug.nl

CIP-GEGEVENS KONINKLIJKE BIBLIOTHEEK DEN HAAG

Symposium Statistische Software, nr. 7 /
onder redactie van T.A.B. Snijders ... [et al.]
Groningen, iec ProGAMMA
ISBN: 90-5144-043-X
Trefwoord: statistische software

ORGANI

Het Symp
Programm
Research

Organisa
K.P. Kon
D.M. van

A.M.J.G.

Programm
T.A.B. Sn
B. Engel
J.C. van
A. Keen
G.J. Sten
M. Verbe

Contacta
SSS '95
iec ProG
Postbus 8
9700 AV
telefoon:
fax:
e-mail:
www:

ORGANISATIE SYMPOSIUM STATISTISCHE SOFTWARE 1995

Het Symposium Statistische Software 1995 is georganiseerd door de Sectie Statistische Programmatuur (SSP) van de Nederlandse Vereniging voor Statistiek en Operationele Research (VVS) in samenwerking met het interuniversitair expertisecentrum ProGAMMA.

Organisatiecommissie

K.P. Koning	iec ProGAMMA, secretaris SSP
D.M. van der Sluis	Rijksuniversiteit Groningen, Faculteit PPSW, penningmeester SSP
A.M.J.G. de Wit	iec ProGAMMA

Programmacommisie & redactie bundel

T.A.B. Snijders	Rijksuniversiteit Groningen, Statistiek en Meettheorie
B. Engel	DLO, Groep Landbouwwiskunde
J.C. van Houwelingen	Rijksuniversiteit Leiden, Medische Statistiek
A. Keen	DLO, Groep Landbouwwiskunde
G.J. Stemerdink	voorzitter SSP
M. Verbeek	Katholieke Universiteit Leuven, Center for Economic Studies

Contactadres

SSS '95 Organisatiecommissie
iec ProGAMMA
Postbus 841
9700 AV Groningen
telefoon: 050 - 363 6900
fax: 050 - 363 6687
e-mail: gamma.post@gamma.rug.nl
www: http://www.gamma.rug.nl