AUSTRIAN JOURNAL OF STATISTICS
Volume 37 (2008), Number 3&4, 285–299

# Multilevel Latent Variable Modeling: An Application in Education Testing

Jeroen K. Vermunt
Department of Methodology and Statistics, Tilburg University, The Netherlands

**Abstract:** A framework for multilevel latent variable modeling is presented that includes many existing models as special cases. It is shown that parameters can be estimated by maximum likelihood using a special variant of the EM algorithm. An application is presented from the field of school effectiveness research. This application uses a novel multilevel mixture item response model which clusters schools based on the students' latent abilities and the item difficulties.

**Zusammenfassung:** Ein Rahmenmodell für multilevel latentes Variablenmodellieren wird präsentiert, welches viele existierende Modelle als Spezialfälle enthält. Es wird gezeigt, dass man durch eine spezielle Variante des EM-Algorithmus Maximum-Likelihoodschätzer der Parameter gewinnen kann. Eine Anwendung aus der Schuleffektivitätsforschung wird präsentiert. Diese Anwendung verwendet ein neuartiges multilevel Mischungsmodell der probabilistischen Testtheorie, welches Schulen auf der Basis von latenten Eigenschaften der Studierenden und Itemschwierigkeiten gruppiert.

**Keywords:** Factor Analysis, Mixture Models, Multilevel Analysis, Mixed Models, Generalized Linear Models, Item Response Theory, Maximum Likelihood Estimation, Item Bias Analysis.

## 1 Introduction

Skrondal and Rabe-Hesketh (2004) proposed a generalized latent variable modeling framework integrating

- factor analytic and random coefficient models,
- models with discrete and continuous unobserved variables, and
- hierarchical models with unobserved variables at different levels.

This framework, which they called "generalized linear latent and mixed models", is implemented in a software routine called GLLAMM Rabe-Hesketh et al. (2004). In this paper, I describe a strongly related framework that is implemented in the syntax version of the Latent GOLD software (Vermunt and Magidson, 2007). The most important extension compared to the GLLAMM approach is that it allows defining models with any combination of discrete and continuous latent variables at each level of the hierarchy. The modeling framework is illustrated with a multilevel application in educational testing; that is, using a set of mathematics test items taken from pupils nested within schools. An item response theory model—a logistic factor-analytic model—is constructed for the responses on the test items and the between-school differences in pupils' abilities and item difficulties is modeled using a discrete mixture distribution at the school level.

The next section introduces the generalized latent variable model of interest. Section 3 discusses maximum likelihood estimation using a special variant of the EM algorithm. Section 4 describes the illustrative example. I end with a short discussion.

# 2 The Multilevel Latent Variable Model

## 2.1 Elements of the Multilevel Latent Variable Model

The multilevel latent variable model (MLVM) contains four elements:

1. a set of response or *dependent variables* ($\mathbf{y}$), which may be binary, nominal, ordinal, continuous, counts, or any combination of these,
2. a set of *latent variables* ($\boldsymbol{\nu}$), which may be discrete (nominal or ordinal), continuous, or combinations of these,
3. a set of predictors or *independent variables* ($\mathbf{Z}$ and $\mathbf{W}$), and
4. nested or *multilevel observations* at $L$ levels.

Using the index $k$ to denote an independent observation corresponding to the highest level of the hierarchy, the regression equations defining a MLVM can be formulated with the following two equations:

$$g[\mathrm{E}(\mathbf{y}_k)] = \mathbf{Z}_k^{(1)}\boldsymbol{\beta} + \mathbf{W}_k^{(1)}\boldsymbol{\Lambda}^{(1)}\boldsymbol{\nu}_k \tag{1}$$

$$h[\mathrm{E}(\boldsymbol{\nu}_k^{(\ell)})] = \mathbf{Z}_k^{(\ell)}\boldsymbol{\gamma}^{\ell} + \mathbf{W}_k^{(\ell)}\boldsymbol{\Lambda}^{(\ell)}\boldsymbol{\nu}_k^{(\ell+)} \qquad \text{for } \ell = 2, \ldots, L\,. \tag{2}$$

Here, $g[\cdot]$ and $h[\cdot]$ are link functions (identity, logit, log, etc.) which may differ across dependent variables and across latent variables and which typically depend on the scale type of the left-hand variable. The free model parameters are the regression coefficients $\boldsymbol{\beta}$, $\boldsymbol{\Lambda}$, and $\boldsymbol{\gamma}$, as well as the residual (co)variances (or associations) between latent variables and between dependent variables. Note that $\boldsymbol{\nu}_k$ denotes the vector of latent variables of observation $k$ at all levels, whereas $\boldsymbol{\nu}_k^{(\ell)}$ and $\boldsymbol{\nu}_k^{(\ell+)}$ refer to the latent variables at level $\ell$ and $\ell$ and higher, respectively.

In one aspect, the MLVM framework implemented in Latent GOLD is slightly less general than suggested by the two model equations: the structural equation model for the latent variables at level $\ell$ is only partially implemented.[1] But in other aspects it is even more general than expressed in the above two equations, including that it allows specification of a Markovian structure for discrete latent variables at the lowest level (Frühwirth-Schatter, 2006; Paas et al., 2007; Vermunt et al., 1999), of interaction effects between latent variables, and of many different models for the residual (co)variances and associations.

It is important to note that the product term $\mathbf{W}_k^{(1)}\boldsymbol{\Lambda}^{(1)}$ in equation (1) is what yields the generalization and integration of the factor analytic and the random coefficient model. In

[1] Latent variables cannot be affected by other latent variables of the same scale type and the same level.

Table 1: Nine-fold classification of possible models with latent variables at two levels.

| | Higher-level $\nu$'s | | |
|---|---|---|---|
| Lower-level $\nu$'s | Continuous | Discrete | Combination |
| Continuous | A1 | A2 | A3 |
| Discrete | B1 | B2 | B3 |
| Combination | C1 | C2 | C3 |

fact, $\mathbf{\Lambda}^{(1)}$ is the factor loadings matrix of a factor analysis and $\mathbf{W}_k^{(1)}$ is the design matrix of a random coefficient model. This implies that by setting $\mathbf{W}_k^{(1)} = \mathbf{1} \otimes \mathbf{I}$ we obtain a factor analytic model and by setting $\mathbf{\Lambda}^{(1)} = \mathbf{I}$ we obtain a random coefficient model. The product $\mathbf{W}_k^{(1)}\mathbf{\Lambda}^{(1)}$—which Skrondal and Rabe-Hesketh (2004) refer to as the structure matrix $\mathbf{\Lambda}_k^{(1)}$—defines the generalized latent variable framework in which the effects of latent variables on responses may contain parameters, fixed terms, or products of these.

It should be noted that the latent variables $\boldsymbol{\nu}_k$ can be common factors in a factor analysis, random coefficients in a multilevel or mixed model, classes in a latent class model, or mixture components in a finite mixture model. In other words, the latent variables may be either discrete or continuous and may be used either to reveal structure (meaningful factors or clusters) or to correct for unobserved heterogeneity.

## 2.2 Some Special Cases

Assuming two levels of latent variables and taking into account that the latent variables at each level may be continuous, discrete, or a combination of these, we obtain the nine-fold classification provided in Table 1. One of the special cases, in which both the lower- and higher-level latent variables are discrete (B2), is the hierarchical variant of the latent class model proposed by Vermunt (2003, 2008). Here, lower-level units (cases) are clustered based on their observed responses as in a standard latent class model, whereas higher-level units (groups) are clustered based on the likelihood of their members to be in one of the case-level clusters. Vermunt (2003) also described a multilevel latent class model with continuous random effects at the group level which belongs to category B1.

A1 contains both three-level mixed models with continuous random effects (Hox, 2002; Snijders and Bosker, 1999) and two-level factor analytic and item response theory (IRT) models, such as the multilevel IRT models proposed by Fox and Glas (2001) and Raudenbush et al., 2003, as well as the multilevel factor analysis models proposed by Longford and Muthén (1992), Goldstein and Browne (2002), and Grilli and Rampichini (2007). In a recent paper, Palardy and Vermunt (2007) proposed a model of the form A3 for defining a multilevel extension of the mixture growth model (Vermunt, 2007). In the application described below, we use a type A2 model.

What is clear from the above table is that the presented MLVM framework yields a large number of options for its users. With latent variables at three instead of two levels, the number of possible specifications increases from 9 to 27. Of course, it depends on the specific application which of the specifications should be selected; that is, whether it is more meaningful from a substantive point of view and/or more practical to define the latent variables at a particular level to be continuous, discrete, or a combination of the two.

# 3 Parameter Estimation by Maximum Likelihood

## 3.1 Log-likelihood Function

The parameters of the MLVM can be estimated by maximum likelihood (ML). Based on the regression equations, the assumptions about the error distributions for the response and latent variables, and the hierarchical structure of the model, one can derive the density function associated with the response vector of an independent observation; that is, of an observation at the highest level of hierarchy. For simplicity of exposition, let us assume that we have a model with continuous latent variables at two levels, which typically will be a model for either univariate three-level data or multivariate two-level data. Here, we will use the terminology corresponding to a three-level model (Hox, 2002; Snijders and Bosker, 1999), where the three levels of the hierarchy are indexed by $i$, $j$, and $k$, respectively.

The likelihood function is based on the probability densities of the level-3 observations, denoted by $f(\mathbf{y}_k|\mathbf{Z}_k, \mathbf{W}_k)$. Here, $\mathbf{y}_k$, $\mathbf{Z}_k$, and $\mathbf{W}_k$ contain the responses and design vectors for all lower-level observations belonging to level-3 unit or group $k$. In order to simplify notation, the conditioning on the design vectors is replaced by an index corresponding to the unit concerned, yielding the short-hand notation $f_k(\mathbf{y}_k)$ for the probability density of level-3 unit $k$.

The log-likelihood to be maximized equals $\log L = \sum_{k=1}^{K} \log f_k(\mathbf{y}_k)$, where

$$\begin{aligned} f_k(\mathbf{y}_k) &= \int_{\boldsymbol{\nu}^{(3)}} f_k(\mathbf{y}_k|\boldsymbol{\nu}^{(3)}) f(\boldsymbol{\nu}^{(3)}) d\boldsymbol{\nu}^{(3)} \\ &= \int_{\boldsymbol{\nu}^{(3)}} \left\{ \prod_{j=1}^{n_k} f_{jk}(\mathbf{y}_{jk}|\boldsymbol{\nu}^{(3)}) \right\} f(\boldsymbol{\nu}^{(3)}) d\boldsymbol{\nu}^{(3)} \,, \end{aligned} \tag{3}$$

and

$$\begin{aligned} f_{jk}(\mathbf{y}_{jk}|\boldsymbol{\nu}^{(3)}) &= \int_{\boldsymbol{\nu}^{(2)}} f_{jk}(\mathbf{y}_{jk}|\boldsymbol{\nu}^{(2)}, \boldsymbol{\nu}^{(3)}) f(\boldsymbol{\nu}^{(2)}) d\boldsymbol{\nu}^{(2)} \\ &= \int_{\boldsymbol{\nu}^{(2)}} \left\{ \prod_{i=1}^{n_{jk}} f_{ijk}(y_{ijk}|\boldsymbol{\nu}^{(2)}, \boldsymbol{\nu}^{(3)}) \right\} f(\boldsymbol{\nu}^{(2)}) d\boldsymbol{\nu}^{(2)} \,. \end{aligned} \tag{4}$$

As can be seen, the responses of the $n_k$ level-2 units within level-3 unit $k$ are assumed to be independent of one another given the latent variables or random effects $\boldsymbol{\nu}^{(3)}$, and the responses of the $n_{jk}$ level-1 units within level-2 unit $jk$ are assumed to be independent of one another given the latent variables or random effects $\boldsymbol{\nu}^{(2)}$ and $\boldsymbol{\nu}^{(3)}$.

The integrals at the right-hand side of equations (3) and (4) can be evaluated by the Gauss-Hermite quadrature numerical integration method (Stroud and Secrest, 1966; Bock and Aikin, 1981; Rabe-Hesketh et al., 2004, Skrondal and Rabe-Hesketh, 2004), in which the multivariate normal mixing distribution is approximated by a limited number of discrete points. More precisely, the integrals are replaced by summations over $T^{(3)}$ and $T^{(2)}$

quadrature points, [2]

$$
\begin{aligned}
f_k(\mathbf{y}_k) &= \sum_{s=1}^{T^{(3)}} P_k(\mathbf{y}_k|\boldsymbol{\nu}_s^{(3)})\pi(\boldsymbol{\nu}_s^{(3)}) = \sum_{s=1}^{T^{(3)}} \left[\prod_{j=1}^{n_k} P_{jk}(\mathbf{y}_{jk}|\boldsymbol{\nu}_s^{(3)})\right] \pi(\boldsymbol{\nu}_s^{(3)}) \\
&= \sum_{s=1}^{T^{(3)}} \left[\prod_{j=1}^{n_k} \sum_{r=1}^{T^{(2)}} \left\{\prod_{i=1}^{n_{jk}} P_{ijk}(y_{ijk}|\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)})\right\} \pi(\boldsymbol{\nu}_r^{(2)})\right] \pi(\boldsymbol{\nu}_s^{(3)}) \,. \qquad (5)
\end{aligned}
$$

Here, $\boldsymbol{\nu}_r^{(2)}$ and $\boldsymbol{\nu}_s^{(3)}$ are quadrature nodes and $\pi(\boldsymbol{\nu}_r^{(2)})$ and $\pi(\boldsymbol{\nu}_s^{(3)})$ are quadrature weights corresponding to the (multivariate) normal densities of interest. Because the latent variables or random effects are orthogonalized, the nodes and weights of the separate dimensions equal the ones of the univariate normal density, which can be obtained from standard tables (see, e.g., Stroud and Secrest, 1966).[3] Suppose that each dimension is approximated with $Q$ quadrature nodes. The $T^{(2)} = Q^{R^{(2)}}$ and $T^{(3)} = Q^{R^{(3)}}$ weights are then obtained by multiplying the weights of the separate dimensions. The integral can be approximated to any practical degree of accuracy by setting $Q$ sufficiently large.[4]

## 3.2 The Upward-Downward Variant of the EM Algorithm

A natural way to solve the ML estimation problem for the MLVM is by means of the EM algorithm (Dempster et al., 1977). The E step of the EM algorithm involves computing the expectation of the complete data log-likelihood, which in the MLVM is of the form[5]

$$
\log L_c = \sum_{s=1}^{T^{(3)}} \sum_{r=1}^{T^{(2)}} \sum_{k=1}^{K} \sum_{j=1}^{n_k} \sum_{i=1}^{n_{jk}} P_{jk}(\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)}|\mathbf{y}_k) \log f_{ijk}(y_{ijk}|, \boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)}) \,. \qquad (6)
$$

This shows that, in fact, the E step involves obtaining the posterior probabilities $P_{jk}(\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)}|\mathbf{y}_k)$ given the current estimates for the unknown model parameters. In the M step of the algorithm, the model parameters are updated so that the expected complete data log-likelihood given in equation (6) is maximized (or improved). This can be accomplished using standard algorithms for the ML estimation of generalized linear models.

The problematic part in the implementation of EM for the MLVM is the E step in which one has to obtain the posterior probabilities $P_{jk}(\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)}|\mathbf{y}_k)$. A standard implementation of the E step would involve computing the joint conditional expectation of the

[2] Actually, we should use "$\approx$" instead of "$=$" sign in this expression because we are approximating the integral by a summation. However, for simplicity of notation in this and next formulas, we retain "$=$".

[3] Application of Gauss-Hermite in multiple correlated dimensions requires reparameterizing the model so that the "new" latent variables are orthogonal. This is achieved by means of a Cholesky decomposition of the variance-covariance matrix of the latent variables. For further details on this, see Skrondal and Rabe-Hesketh (2004) and Hedeker and Gibbons (1996).

[4] Lesaffre and Spiessens (2001) and Rabe-Hesketh et al. (2002) showed that the number of quadrature points needs to be very large in some situations. In such cases, it is better to use adaptive quadrature.

[5] The terms containing the priors $\pi(\boldsymbol{\nu}_r^{(2)})$ and $\pi(\boldsymbol{\nu}_s^{(3)})$ are omitted from $L_c$ because these do not contain parameters to be estimated.

$n_k \cdot R^{(2)} + R^{(3)}$ random effects for level-3 unit $k$; that is, the joint posterior distribution $P_k(\boldsymbol{\nu}_{r1}^{(2)}, \boldsymbol{\nu}_{r2}^{(2)}, \ldots, \boldsymbol{\nu}_{r_{n_k}}^{(2)}, \boldsymbol{\nu}_s^{(3)} | \mathbf{y}_k)$ with $Q^{\{n_k \cdot R^{(2)} + R^{(3)}\}}$ entries. Note that this amount to computing the expectation of all the "missing data" for a level-3 unit. These joint posteriors would subsequently be collapsed to obtain the marginal posterior probabilities for each level-2 unit $j$ within level-3 unit $k$, $P_{jk}(\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)} | \mathbf{y}_k)$. This yields a procedure in which computer storage and time increases exponentially with the number of level-2 units, which means that it can only be used with very small $n_k$.

However, it turns out that it is possible to compute the $n_k$ marginal posterior probability distributions $P_{jk}(\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)} | \mathbf{y}_k)$ without going through the full posterior distribution by making use of the conditional independence assumptions associated with the density function defined in equation (3). It that sense, our procedure is similar to the forward-backward algorithm for the estimation of hidden Markov models with large numbers of time points (Baum et al., 1970; Frühwirth-Schatter, 2006; Juang and Rabiner, 1991. Analogous to the forward-backward procedure, Vermunt (2004) called the algorithm described below an upward-downward procedure. In the graphical or Bayesian belief network modelling field, the MLVM would be recognized as a single-connected network or polytree, for which relevant marginal conditional probabilities can be obtained by propagation algorithms (Pearl, 1988). Both the forward-backward algorithm for hidden Markov models and the upward-downward algorithm discussed below are propagation algorithms.

In the upward-downward algorithm, latent variables are integrated out going from the lower to the higher levels. Subsequently, the relevant marginal posterior probabilities are computed going from the higher to the lower levels. This yields a procedure in which computer storage and time increases only linearly with the number of level-2 observations instead of exponentially, as would have been the case with a standard EM algorithm. This is the algorithm implemented in the Latent GOLD software package (Vermunt and Magidson, 2005, 2007).

The marginal posterior probabilities $P_{jk}(\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)} | \mathbf{y}_k)$ can be decomposed as follows:

$$P_{jk}(\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)} | \mathbf{y}_k) = P_k(\boldsymbol{\nu}_s^{(3)} | \mathbf{y}_k) P_{jk}(\boldsymbol{\nu}_r^{(2)} | \mathbf{y}_k, \boldsymbol{\nu}_m^{(3)}) \,.$$

Our procedure makes use of the fact that in the MLVM

$$P_{jk}(\boldsymbol{\nu}_r^{(2)} | \mathbf{y}_k, \boldsymbol{\nu}_s^{(3)}) = P_{jk}(\boldsymbol{\nu}_r^{(2)} | \mathbf{y}_{jk}, \boldsymbol{\nu}_s^{(3)}) \,;$$

i.e., $\boldsymbol{\nu}_r^{(2)}$ is independent of the observed responses of the other level-2 units within the same level-3 unit given $\boldsymbol{\nu}^{(3)}$. This is the result of the fact that level-2 observations are mutually independent given the level-3 random effects, as is expressed in the density function described in equation (3). Using this important result, we get the following slightly simplified decomposition:

$$P_{jk}(\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)} | \mathbf{y}_k) = P_k(\boldsymbol{\nu}_s^{(3)} | \mathbf{y}_k) P_{jk}(\boldsymbol{\nu}_r^{(2)} | \mathbf{y}_{jk}, \boldsymbol{\nu}_s^{(3)}) \,. \tag{7}$$

The computation of the marginal posterior probabilities therefore reduces to the computation of the two terms at the right-hand side of this equation. The term $P_{jk}(\boldsymbol{\nu}_r^{(2)} | \mathbf{y}_{jk}, \boldsymbol{\nu}_s^{(3)})$ is obtained as follows:

$$P_{jk}(\boldsymbol{\nu}_r^{(2)} | \mathbf{y}_{jk}, \boldsymbol{\nu}_s^{(3)}) = \frac{P_{jk}(\mathbf{y}_{jk}, \boldsymbol{\nu}_r^{(2)} | \boldsymbol{\nu}_s^{(3)})}{P_{jk}(\mathbf{y}_{jk} | \boldsymbol{\nu}_s^{(3)})} \,,$$

where

$$P_{jk}(\mathbf{y}_{jk}, \boldsymbol{\nu}_r^{(2)}|\boldsymbol{\nu}_s^{(3)}) = \pi(\boldsymbol{\nu}_r^{(2)}) \prod_{i=1}^{n_{jk}} P_{ijk}(y_{ijk}|\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)})$$

$$P_{jk}(\mathbf{y}_{jk}|\boldsymbol{\nu}_s^{(3)}) = \sum_{r=1}^{T^{(2)}} P_{jk}(\mathbf{y}_{jk}, \boldsymbol{\nu}_r^{(2)}|\boldsymbol{\nu}_s^{(3)}) \,.$$

The other term $P_k(\boldsymbol{\nu}_s^{(3)}|\mathbf{y}_k)$ is obtained by

$$P_k(\boldsymbol{\nu}_s^{(3)}|\mathbf{y}_k) = \frac{P_k(\mathbf{y}_k, \boldsymbol{\nu}_s^{(3)})}{P_k(\mathbf{y}_k)} \,, \tag{8}$$

where

$$P_k(\mathbf{y}_k, \boldsymbol{\nu}_s^{(3)}) = \pi(\boldsymbol{\nu}_s^{(3)}) \prod_{j=1}^{n_k} P_{jk}(\mathbf{y}_{jk}|\boldsymbol{\nu}_s^{(3)})$$

$$P_k(\mathbf{y}_k) = \sum_{s=1}^{T^{(3)}} P(\mathbf{y}_k, \boldsymbol{\nu}_s^{(3)}) \,.$$

Thus, first the level-2 posterior probabilities $P_{jk}(\boldsymbol{\nu}_r^{(2)}|\mathbf{y}_{jk}, \boldsymbol{\nu}_s^{(3)})$ are obtained from the level-1 information $P_{ijk}(y_{ijk}|\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)})$, and subsequently the level-3 posterior probabilities $P_k(\boldsymbol{\nu}_s^{(3)}|\mathbf{y}_k)$ are obtained from the level-2 information $P_{jk}(\mathbf{y}_{jk}|\boldsymbol{\nu}_s^{(3)})$. This is called the *upward* step of the algorithm because one goes up in the hierarchical structure. In the *downward* step, one computes $P_{jk}(\boldsymbol{\nu}_r^{(2)}, \boldsymbol{\nu}_s^{(3)}|\mathbf{y}_k)$ by means of equation (7).

The upward-downward method can easily be generalized to more than three levels. For example, with four levels, one would have to compute the three terms $P_m(\boldsymbol{\nu}_o^{(4)}|\mathbf{y}_m)$, $P_{km}(\boldsymbol{\nu}_r^{(3)}|\mathbf{y}_{km}, \boldsymbol{\nu}_o^{(4)})$, and $P_{jkm}(\boldsymbol{\nu}_r^{(2)}|\mathbf{y}_{jkm}, \boldsymbol{\nu}_s^{(3)}, \boldsymbol{\nu}_o^{(4)})$, where $m$ refers to a level-four unit and $o$ to a quadrature point for the level-four unit random effects. These three terms are obtained in the upward step and used to calculate the relevant marginal posteriors in the downward step.

Note that we described ML estimation for models with continuous latent variables and numerical integration. An almost equivalent procedure is, however, used for discrete latent variables. The only difference is that the "quadrature weights" are then not fixed but contain free parameters to be estimated (see, Vermunt, 2003, 2004).

## 3.3 Standard Errors and Identification Issues

Contrary to Newton-like methods, the EM algorithm does not provide standard errors of the model parameters as a by-product. Estimated asymptotic standard errors can be obtained by computing the observed information matrix, the matrix of second-order derivatives of the log-likelihood function toward all model parameters. The inverse of this matrix is the estimated variance-covariance matrix. Latent GOLD computes these second derivatives numerically using analytic first derivatives. Note that the first derivatives are provided by the proposed EM algorithm.

For checking identifiability, we use the Jacobian matrix, the matrix with the first derivatives of $f_k(\mathbf{y}_k)$ towards the model parameters, which can be obtained as a by-product of an EM iteration cycle. A sufficient condition for local identification is that the Jacobian is of full column rank (Rothenberg, 1971).

# 4 An Illustrative Application: A Multilevel Mixture IRT Model

The application uses a data set collected by Doolaard (1999), and which was also used by Fox and Glas (2001) to illustrate their multilevel IRT model. More specifically, information is available on a 18-item math test taken from 2156 pupils belonging to 97 schools in the Netherlands. The aim of the analysis is twofold: measuring pupils' math abilities and assessing differences between school. The first aim involves building a single factor or IRT model for the 18 math items, while the second aim involves introducing school-level random coefficients in the IRT model.

As far as the IRT model is concerned, two different models are considered: the two-parameter logistic (2-PL) model and the Rasch model, which is also referred to as the one-parameter logistic (1-PL) model.[6] As in Fox and Glas's multilevel IRT model, we are interested in school differences in ability. Unlike Fox and Glas, we also want to know whether the items' functioning is the same across schools; that is, whether equally able students from different schools are equally likely to answer each of the math items correctly. The latter is often referred to as item bias analysis. Such an analysis is feasible using a discrete finite mixture specification for the relevant school differences. The proposed multilevel mixture IRT model can, therefore, be seen as a practical method for detecting item bias in situations in which the number of groups is too large for a standard item bias analysis, in which group differences are modeled using fixed instead of random effects.

Let $y_{ijk}$ denote the binary response on item $i$ of pupil $j$ in school $k$. Note that $i$, $j$, and $k$ refer to a level-1, level-2, and level-3 unit, respectively. Denoting the latent ability of pupil $j$ in school $k$ by $\nu_{jk}^{(2)}$, we can define the 2-PL model as follows:

$$\text{logit}[P(y_{ijk}=1)] = \beta_i + \lambda_i^{(1)}\nu_{jk}^{(2)} \qquad \text{for } i = 1,\ldots,18 \tag{9}$$

$$\mathrm{E}(\nu_{jk}^{(2)}) = 0\,; \tag{10}$$

where $\lambda_i^{(1)}$ is the factor loading or discrimination for item $i$ and $-\beta_i/\lambda_i^{(1)}$ is what is usually referred to as the item difficult (the value of $\nu_{jk}^{(2)}$ at which one has a 50% percent likelihood to give a correct answer to the item concerned). For identification purposes, we will typically restrict one $\lambda_i^{(1)}$, say $\lambda_1^{(1)}$, to be equal to 1. The latent ability is assumed to come from a normal distribution with a mean equal to 0 and a free variance. With the restriction $\lambda_i^{(1)} = 1$ for all $i$, we obtain the Rasch model.

[6] For more information on IRT and the specific terminology used in IRT, see for example Van der Linden et al. (1997).

Fox and Glas (2001) proposed a multilevel extension of the standard IRT model in which a pupil's ability is affected by a normally distributed school-level latent variable or random effect $\nu_k^{(3)}$; that is,

$$\begin{aligned}\text{logit}[P(y_{ijk}=1)] &= \beta_i + \lambda_i^{(1)}\nu_{jk}^{(2)} \qquad \text{for } i=1,\ldots,18 &(11)\\ \text{E}(\nu_{jk}^{(2)}) &= \lambda^{(2)}\nu_k^{(3)} &(12)\\ \text{E}(\nu_k^{(3)}) &= 0\,. &(13)\end{aligned}$$

For identification, we fix either the variance of $\nu_k^{(3)}$ or the loading $\lambda^{(2)}$ to 1.

Suppose that rather than having a continuous school-level random effect, we wish to deal with the multilevel structure assuming that each school belongs to one of $M$ latent classes or mixture components with different mean abilities. Such a model can be formulated as follows:

$$\begin{aligned}\text{logit}[P(y_{ijk}=1)] &= \beta_i + \lambda_{i1}^{(1)}\nu_{jk}^{(2)} \qquad \text{for } i=1,\ldots,18 &(14)\\ \text{E}(\nu_{jk}^{(2)}) &= \sum_{m=1}^{M-1}\lambda_m^{(2)}\nu_{km}^{(3)} &(15)\\ \text{logit}[P(\nu_{km}^{(3)}=1)] &= \gamma_m^{(3)}\,, &(16)\end{aligned}$$

where $\nu_{km}^{(3)}$ represents one of $M-1$ indicator variables taking the value 1 if school $k$ belongs to latent class $m$ and otherwise 0 (with effect coding $\nu_{km}^{(3)}$ equals $-1$ if school $k$ belongs to class $M$). The $\lambda_m^{(2)}$ parameters capture differences between school-level classes in average abilities. The parameters $\gamma_m^{(3)}$ are the intercepts in the logit model for the latent classes.

Finally we could allow school-level classes to differ not only with respect to the students abilities, but also with respect to the item difficulties after controlling for the students abilities. This yields the model:

$$\begin{aligned}\text{logit}[P(y_{ijk}=1)] &= \beta_i + \lambda_{i1}^{(1)}\nu_{jk}^{(2)} + \sum_{m=1}^{M-1}\lambda_{i,m+1}^{(1)}\nu_{km}^{(3)} &(17)\\ \text{E}(\nu_{jk}^{(2)}) &= \sum_{m=1}^{M-1}\lambda_m^{(2)}\nu_{km}^{(3)} &(18)\\ \text{logit}[P(\nu_{km}^{(3)}=1)] &= \gamma_m^{(3)}\,. &(19)\end{aligned}$$

The $\lambda_{i,m+1}^{(1)}$ parameters capture differences between school-level classes in item difficulties. In this full model we have to impose identifying constraints on the $\lambda_{i,m+1}^{(1)}$ parameters; for example, $\lambda_{1,m+1}^{(1)}=0$ for $m=1,\ldots,M-1$.

The path diagram representing the multilevel IRT model is depicted in two different ways in Figures 1 and 2.[7] In Figure 1, the model is depicted as a two-level regression

[7] An extended discussion of path diagram depicting multilevel models can be found in Skrondal and Rabe-Hesketh (2004).

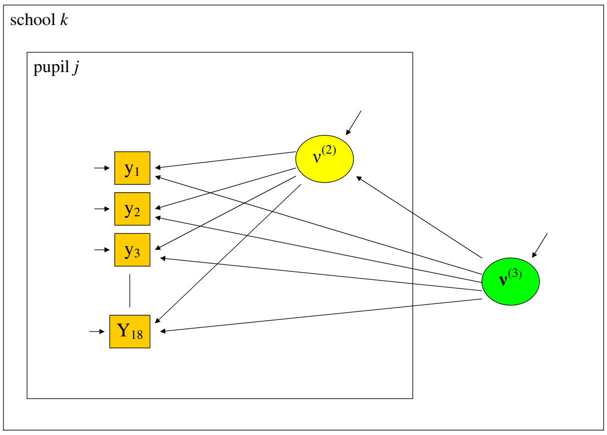

Figure 1: Multilevel IRT model as a two-level model for multivariate responses.

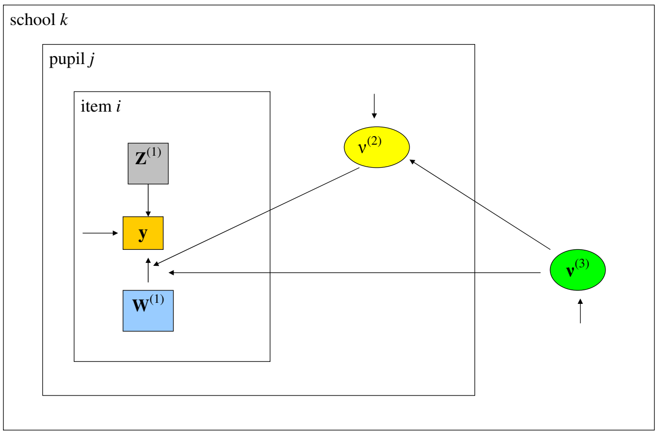

Figure 2: Multilevel IRT model as a three-level model for a univariate response.

model for multivariate responses. Figure 2 depicts the same model as a three-level regression model in which item dummies are used as item-level predictor variables.

This multilevel mixture IRT model can be extended in various ways. The most obvious and interesting extension is inclusion of pupil-level covariates in the regression model for the child's latent ability and school-level covariates in the model for the school-level class memberships.

Table 2 reports the fit measures obtained with the estimated 1- to 5-class models. As can be seen, the 2-PL models perform better than their Rasch counterparts, indicating that the Rasch assumption of equal discrimination across items is too strict for this data set. For the 2-PL specification, comparison of the models with and without item bias indicates that there is no evidence for item bias. In this specification the 3-class model without item bias is selected as the best according to the BIC criterion. In the Rasch specification, the 4-class model with item bias is the best model. This application shows that using

Table 2: BIC values obtained with the estimated multilevel mixture 2-PL and Rasch models ($N = 2156$).

| | 2-PL | | Rasch | |
|---|---|---|---|---|
| number of classes | without item bias | with item bias | without item bias | with item bias |
| 1 | 40701 | 40701 | 40750 | 40750 |
| 2 | 40502 | 40545 | 40562 | 40517 |
| 3 | **40449** | 40514 | 40515 | 40513 |
| 4 | 40455 | 40502 | 40524 | **40485** |
| 5 | 40469 | 40540 | 40538 | 40538 |

the too restricted Rasch model may lead to the erroneous conclusion that items function differentially across groups.

It should be noted that model selection is not as simple as could be concluded from the above example application. The first issues we would like to mention in this context is that the Rasch model may still be preferable because of theoretical reasons and that it may turn out to be the preferred model after eliminating some of the items from the test. An issue I would like to focus on in future research is the BIC measure itself, or, more specifically, the definition of the sample size in multilevel latent variable models. Should the sample size in the BIC formula be the number of individuals (pupils) or the number of groups (schools)? In the current application we used the number of pupils as $N$.

# 5 Discussion

I presented a general multilevel latent variable modeling framework and discussed some of its special cases. Moreover, attention was paid to parameter estimation by ML using a special variant of the EM algorithm. An application was presented from the field of school effectiveness research. The two questions addressed in the analysis were 1) whether the average latent ability differs across schools and 2) whether overall the performance on individual items differs across schools after controlling for a pupil's ability. The answer to the first question was yes and to the second no. In a more extended analysis, one would introduce covariates to explain why average abilities differ across schools.

A limitation of the approach presented in this article arises from the fact that, expect for models for continuous response variables (see Palardy and Vermunt, 2007), numerical integration is needed for parameter estimation using ML. This implies that the maximum number of latent dimensions that one can deal with in a single analysis is not very large. A possible way out to this problem is to switch to simulation based estimation algorithm within either a ML or a Bayesian framework. But even though the models described in this article can easily defined in Winbugs (Spiegelhalter et al., 2003), Bayesian MCMC estimation will typically be slower than Latent GOLD.

The models presented in this article are all hierarchical models; that is, models for data sets having an exactly nested group structure. Random effects and latent variable models are, however, also relevant in situations in which the grouping of observations is not exactly in agreement with a nested structure. A good example in context of school effectiveness research is the nesting of children within schools and neighborhoods: schools are not nested within neighborhoods because children from the same school can live in different neighborhoods. Except for models with continuous dependent and latent variables, parameter estimation using ML methods seems to be impossible with such crossed random effects, but again simulation based methods may provide a way out.

# References

Baum, L. E., Petrie, T., Soules, G., and Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *Annals of Mathematical Statistics*, *41*, 164-171.

Bock, R. D., and Aikin, M. (1981). Marginal maximum likelihood estimation of item parameters. *Psychometrika*, *46*, 443-459.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood estimation from incomplete data via the EM algorithm (with discussion). *Journal of the Royal Statistical Society, Series B*, *39*, 1-38.

Doolaard, S. (1999). *Schools in Change or School in Chain*. Unpublished doctoral dissertation, University of Twente, The Netherlands.

Fox, J. P., and Glas, C. A. W. (2001). Bayesian estimation of a multilevel IRT model using Gibbs sampling. *Psychometrika*, *66*, 269-286.

Frühwirth-Schatter, S. (2006). *Finite Mixture and Markov Switching Models*. New York, NY: Springer.

Goldstein, H., and Browne, W. (2002). Multilevel factor analysis modelling using Markov chain Monte Carlo estimation. In G. A. Marcoulides and I. Moustaki (Eds.), *Latent Variable and Latent Structure Models* (p. 225-243). Mahwah, NJ: Lawrence Erlbaum Associates.

Grilli, L., and Rampichini, C. (2007). Multilevel factor models for ordinal variables. *Structural Equation Modeling*, *14*, 1-25.

Hedeker, D., and Gibbons, R. D. (1996). MIXOR: A computer program for mixed effects ordinal regression analysis. *Computer Methods and Programs in Biomedicine*, *49*, 157-176.

Hox, J. (2002). *Multilevel Analysis: Techniques and Applications*. Mahwah, NJ.

Juang, B. H., and Rabiner, L. R. (1991). Hidden Markov models for speech recognition. *Technometrics*, *33*, 251-272.

Lesaffre, E., and Spiessens, B. (2001). On the effect of the number of quadrature points in a logistic random-effects model: an example. *Applied Statistics*, *50*, 325-335.

Longford, N., and Muthén, B. (1992). Factor analysis for clustered observations. *Psychometrika*, *57*, 581-597.

Paas, L. J., Vermunt, J. K., and Bijmolt, T. H. (2007). Discrete-time discrete-state latent Markov modelling for assessing and predicting household acquisitions of financial products. *Journal of the Royal Statistical Society, Series A*, *170*, 955-974.

Palardy, G., and Vermunt, J. K. (2007). *Multilevel growth mixture models for classifying group-level observations.* (submitted)

Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Fransisco, CA: Morgan Kaufmann Publishers Inc.

Rabe-Hesketh, S., Skrondal, A., and Pickles, A. (2002). Reliable estimation of generalised linear mixed models using adaptive quadrature. *The Stata Journal*, *2*, 1-21.

Rabe-Hesketh, S., Skrondal, A., and Pickles, A. (2004). Generalized multilevel structural equation modelling. *Psychometrika*, *69*, 183-206.

Raudenbush, S. W., Johnson, C., and Sampson, R. J. (2003). A multivariate, multilevel rasch model with application to self-reported criminal behavior. *Sociological Methodology*, *33*, 169-211.

Rothenberg, T. J. (1971). Identification of parametric models. *Econometrica*, *39*, 577-591.

Skrondal, A., and Rabe-Hesketh, S. (2004). *Generalized Latent Variable Modeling: Multilevel, Longitudinal and Structural Equation Models*. London: Chapman & Hall/CRC.

Snijders, T. A. B., and Bosker, R. J. (1999). *Multilevel Analysis*. London: Sage Publications.

Spiegelhalter, D., Thomas, A., Best, N., and Lunn, D. (2003). *Winbugs user manual version 1.4*. MRC Biostatistics Unit at Institute of Public Health at Cambridge University and Department of Epidemiology & Public Health at Imperial College School of Medicine.

Stroud, A. H., and Secrest, D. (1966). *Gaussian Quadrature Formulas*. Englewood Cliffs, NJ: Prentice Hall.

Van der Linden, W. J., Ronald, K., and Hambleton, R. K. (1997). *Handbook of Modern Item Response Theory*. New York, NY: Springer.

Vermunt, J. K. (2003). Multilevel latent class models. *Sociological Methodology*, *33*, 213-239.

Vermunt, J. K. (2004). An EM algorithm for the estimation of parametric and nonparametric hierarchical nonlinear models. *Statistica Neerlandica*, *58*, 220-233.

Vermunt, J. K. (2007). Growth models for categorical response variables: standard, latent-class, and hybrid approaches. In K. van Montfort, H. Oud, and A. Satorra (Eds.), *Longitudinal Models in the Behavioral and Related Sciences* (p. 139-158). Mahwah, NJ.

Vermunt, J. K. (2008). Latent class and finite mixture models for multilevel data sets. *Statistical Methods in Medical Research*. (in press)

Vermunt, J. K., Langeheine, R., and Böckenholt, U. (1999). Discrete-time discrete-state latent Markov models with time-constant and time-varying covariates. *Journal of Educational and Behavioral Statistics*, *24*, 178-205.

Vermunt, J. K., and Magidson, J. (2005). *Latent GOLD 4.0 user's guide*. Belmont, MA: Statistical Innovations Inc.

Vermunt, J. K., and Magidson, J. (2007). *LG-syntax user's guide: Manual for Latent GOLD 4.5 syntax module*. Belmont, MA: Statistical Innovations Inc.

# Appendix: Latent GOLD Syntax Files

The exact form of the Latent GOLD 4.5 syntax for a multilevel mixture IRT model depends on whether the data file is in a two-level multivariate or a three-level univariate response format. Let us first look at the set up for a two-level multivariate response file:[8]

```
variables
   groupid schoolid;
   dependent y1 binomial, y2 binomial, y3 binomial, ..., y18 binomial;
   latent nu2 continuous, nu3 nominal group 3 coding=last;
equations
   y1 <- 1 + (1) nu2 + (0) nu3 // equation for y1
   y2 <- 1 + nu2 + nu3         // equation for y2
   y3 <- 1 + nu2 + nu3         // equation for y3
   ...
   y18 <- 1 + nu2 + nu3        // equation for y18
   nu2 <- nu3;                 // equation for nu2
   nu3 <- 1;                   // equation for nu3
   nu2;                        // residual variance of nu2
```

The first part—the "variables" section—defines the dependent and latent variables which are in the model, as well as the "groupid" variable connecting the records of the children belonging to the same school. The 18 dependent variables are of the scale type "binomial", which means dichotomous variables modelled with a logit link; "nu2" is a "continuous" latent variable, which means normally distributed; and "nu3" is a "nominal" latent variable at the "group" level with "3" categories (its last category is used as the reference category).

The second part of the set up contains the regression equations which are rather similar to the equations presented in the text. We have a separate equation for each dependent and each latent variable. Note that in these equations, the constant is referred to as "1" (a predictor that has the value 1 for all records). The "(1)" and "(0)" in the equation for "y1" indicate that these parameters should be fixed to 1 and 0, respectively. The last equation is a "variance equation" (an equation without a "<-"), which is needed here to indicate that the residual variance of "nu2" is a free parameter (in the default setting, residual variances are fixed to 1).

In the case of a univariate three-level data file, the set up for the same model would be as follows:

```
variables
   groupid schoolid;
   caseid childid;
   dependent y binomial;
   independent itemnr nominal;
   latent nu2 continuous, nu3 nominal group 3 coding=last;
equations
   y <- 1 | itemnr + (lambda1) nu2 | itemnr + (lambda2) nu3 | itemnr;
   nu2 <- nu3;
   nu3 <- 1;
   nu2;
   lambda1[1] = 1;
   lambda2[1] = 0;
```

[8] To save space, we indicate with "..." that the (same kind of) information should be inserted at that place for y4 till y17.

Differences in the “`variables`” section compared to the previous set up are that we need a “`caseid`” to connect the 18 responses of a child, that we have only one response variable, and that we need to include an independent variable “`itemnr`” to allow the definition of models in which parameters differ across items. Note that “`itemnr`” is a variable in the data file which takes on the value 1 for the first item, 2 for the second, etc.. The regression equation for the dependent variable contains again an intercept, an effect of “`nu2`”, and an effect of “`nu3`”. With the appendix “`| itemnr`” one indicates that the value of the parameters concerned depend on the value of “`itemnr`”. In other words, both the intercept and the two slopes vary across items. The identifying restrictions on the model parameters are imposed by first defining labels for the parameters concerned – using “`(lambda1)`” and “`(lambda2)`” – and by subsequently adding two restrictions at the end of the “`equations` section. The restriction “`lambda1[1] = 1`”, for example, indicates the effect of “`nu2`” equals 1 for the first item (first category of the independent variable “`itemnr`”). The other equations are the same as above.

A Rasch model is obtained by fixing the effect of “`nu2`” to 1 for all items. In the first specification this requires putting “`(1)`” before the terms concerned. In the second specification, this could be done by removing “`| itemnr`” from the term for “`nu2`” or by replacing “`lambda1[1] = 1`” with “`lambda1 = 1`”. A model without item bias is obtained by eliminating “`nu3`” from the regression equation(s) for the items. Note that in a model without item bias “`nu3`” can also be specified to be “`continuous`” instead of “`nominal`”. If we would like to estimate the variance of “`nu3`” in such a model (rather than assuming that it is equal to 1) one should include a variance equation for “`nu3`” and fix its effect on “`nu2`” to be equal to 1.

Author’s Address:

Jeroen Vermunt
Faculty of Social and Behavioural Sciences
Department of Methodology and Statistics
PO Box 90153
5000 LE Tilburg
The Netherlands

E-mail: `j.k.vermunt@uvt.nl`