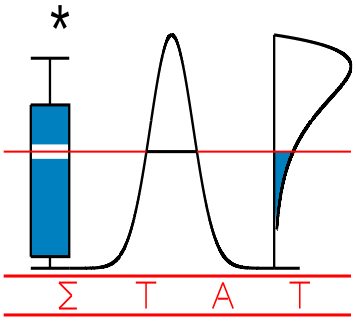
T E C H N I C A L

R E P O R T

0253

CLASSIFICATION OF CLUSTERED DATA USING A SAS-MACRO: AN APPLICATION TO LATENT CLASS MODELS

SPIESSENS, B., VERBEKE, G., FIEUWS, S. and F. RIJMEN



I A P S T A T I S T I C S

N E T W O R K

INTERUNIVERSITY ATTRACTION POLE

http://www.stat.ucl.ac.be/IAP

Classification of clustered data using a SAS-macro: an application to latent class models

Bart Spiessens Geert Verbeke* Steffen Fieuws Frank Rijmen

*Corresponding author: Geert Verbeke, Biostatistical Center, Catholic University of Leuven, Kapucijnenvoer 35, B-3000 Leuven, Belgium. Email: Geert.Verbeke@med.kuleuven.ac.be

Abstract

In item-response theory, subjects often give answers to several questions, leading to correlated measurements. The correlation that arises in this way between the measurements of a single subject has to be taken into account in an appropriate way. If responses are binary, logistic random-effects models could be used to model the correlation through subject-specific effects. Often, one is interested in finding latent classes or clusters in the whole population. Recently, a SAS-macro has been developed to allow for finite mixtures of random-effects models, such that subpopulations can be identified. The SAS-macro actually covers a broader range of random-effects models, but we will show how the macro can be used to analyse latent class models in the case of repeated binary measurements. We will use a dataset on complex reasoning problems where the hypothesis is tested that there are three latent classes of reasoners.

KEY WORDS: Conditional reasoning, Heterogeneous model, IRT models, Normality assumption, Random effects, Repeated measurements

1 Introduction

In studies where subjects contribute more than one measurement, the correlation between these measurements has to be taken into account. One way is to use mixed models in which subject-specific effects (random effects) model the correlation. For continuous data, linear and nonlinear mixed models could be used (Verbeke and Molenberghs, 2000; Davidian and Giltinan, 1995). For binary, count and categorical data, generalised linear mixed models are adopted (Breslow and Clayton, 1993). E.g., in item-response theory (IRT), subjects often give binary responses on several items and IRT models could be analysed using logistic random-effects models (Rijmen, Tuerlinckx and De Boeck, 2002). When the interest lies in finding clusters in the whole population, these models can be extended to incorporate discrete mixture models (Mislevy and Verhelst, 1990). In this case it is assumed that there exist unknown latent classes where each class can be represented by an IRT model.

In this paper, we will present a SAS-macro (Spiessens, Verbeke and Komárek, 2002) for fitting finite mixtures of nonlinear and generalised linear mixed models and apply it to a latent class analysis involving IRT models. It will be shown how the model can be fitted using the macro and the results will be discussed extensively.

Spiessens, Verbeke and Komárek (2002) extend the normality assumption for the random effects to incorporate mixtures of normal components and use this to classify clustered data into the different mixture components. Verbeke and Lesaffre (1996) already did this for linear mixed models (see also Muthén and Shedden, 1999).

In the next section, the example dataset is described and several models will be intro-

duced. Afterwards in Section 3, the methodology introduced by Spiessens, Verbeke and Komárek (2002) for fitting finite mixtures of nonlinear and generalised linear mixed models is given. Section 4 shows how the macro can be used in our example. Finally, Section 5 gives a discussion about the obtained results.

2 Latent class models for complex reasoning problems

2.1 Classical logistic random-effects models

The dataset has been taken from Rijmen and De Boeck (2001). They set up an experiment with 214 high school students between 16 and 19 years of age. The participants had to solve 30 reasoning problems and for each problem, the possible outcomes were 'necessarily true' if they thought that the conclusion must be true, 'necessarily false' if they thought that the conclusion must be false, 'undecidable' if they thought that the conclusion could be true or false and 'I don't know' if they couldn't solve the problem. The latter option was given to avoid guessing behaviour. For 12 problems, the correct answer was 'necessarily true'. Twelve other questions had 'necessarily false' as correct answer and for 6 problems the conclusion was 'undecidable'. The last 6 problems were used as 'filler items' to make sure that the correct conclusion was sometimes something else as 'necessarily false' or 'necessarily true' and were not included in the analysis.

Rijmen and De Boeck (2001) wanted to test whether it is necessary to assume that basic rules differ in their contribution to the difficulty of complex reasoning problems. They

perform a regression analysis on the 24 proportions which were transformed into their logits (logit$(p) = \log[p/(1-p)]$).

Several factors were included in the design and the following three factors were withheld in the final analysis. The first factor was modus ponens (MP: 'if p, then q' and 'p', hence 'q') versus modus tollens (MT: 'if p, then q' and 'not q', hence 'not p'). The second factor refers to the inferences that were combined with MP or MT: MP + conjunction (C: 'p' and 'q' yields 'p and q'), disjunctive syllogism (DS: 'p or q' and 'not p' yields 'q') and disjunctive modus ponens (DMP: 'if p or q, then r' and 'p' yields 'r'). Combinations of these two factors yield 6 problem types (see also Table 1). The 24 questions that were used in the analyses were equally divided over the 6 problem types. The last factor that was included in the final analysis was whether the conclusion was 'true' or 'false'. Factors that were left out of the model were 'content of the problems' and 'the order of presentation of the premises'. Both were not significant at the final analysis.

[Table 1 about here.]

From this experiment, Rijmen and De Boeck (2001) conclude that rule theories can account for the difficulty of the complex reasoning problems, if different weights are allowed for different basic rules. A reasoning problem with a MT is more difficult to solve than a problem involving a MP. For the second factor, they find that DS is more difficult than DMP and DMP is more difficult than MP+C. Finally, a problem with a true conclusion is easier to solve than a problem with a false conclusion.

Instead of transforming the 24 proportions to their logit values, one could also analyse

the data using a logistic random-effects model as follows. Denote y_{ij} as the jth binary measurement of the ith subject and the vector $\boldsymbol{y}_i = (y_{i1}, \ldots, y_{in_i})'$ contains all the measurements of the ith subject $(i = 1, ..., N)$. In the logistic random-effects model, one models the probability of success, denoted by p_{ij}, conditional on the random effects $\boldsymbol{b}_i$, as

$$\begin{aligned} \text{logit}(P(y_{ij} = 1|\boldsymbol{b}_i)) &= \log\left(\frac{p_{ij}}{1-p_{ij}}\right) = \boldsymbol{x}'_{ij}\boldsymbol{\beta} + \boldsymbol{z}'_{ij}\boldsymbol{b}_i, \\ \boldsymbol{b}_i &\sim \mathcal{N}(\boldsymbol{\mu}, R) \\ y_{ij}|\boldsymbol{b}_i &\sim \text{Bernoulli}(p_{ij}) \end{aligned} \tag{1}$$

where the p-dimensional vector $\boldsymbol{\beta}$ are the fixed-effects parameters (population-level), the q-dimensional vector $\boldsymbol{b}_i$ are the random effects of the ith individual, and the vectors $\boldsymbol{x}_{ij}$ and $\boldsymbol{z}_{ij}$ contain the covariates of the ith subject at the jth measurement, corresponding to the fixed and random effects, respectively. The random effects $\boldsymbol{b}_i$ are assumed to follow a normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix R and the distribution of y_{ij} given $\boldsymbol{b}_i$ is assumed to be a Bernoulli distribution with mean p_{ij}. If all the subjects have the same number of measurements, we can write $n_i = n, i = 1, \ldots, N$. This is the case in our example dataset where $n = 24$, the number of items per subject, and the total number of subjects is $N = 214$.

In this model, the correlation between measurements of the same individual is modelled through the random effects $\boldsymbol{b}_i$. Parameter estimation can be done using maximum likelihood, e.g., through the SAS procedure NLMIXED (SAS Version 8.0, 1999). In this procedure, the marginal likelihood, in which the random effects have been integrated out, is maximised and the integration can be performed, e.g., using (adaptive) Gaussian quadrature (Lesaffre and

Spiessens, 2001).

Special cases of model (1) are the Rasch model and the linear logistic test model (LLTM).

In the Rasch model, only random-intercepts are used ($\boldsymbol{z}_{ij} = 1$ and $\boldsymbol{b}_i = b_i \sim \mathcal{N}(0, \sigma^2)$) and for each item, a binary covariate is included in the model, that is

$$\boldsymbol{x}'_{ij} = (0 \dots 0 \underbrace{1}_{j} 0 \dots 0), \quad j = 1, \dots, n,$$

and $\boldsymbol{\beta}$ is a n-dimensional vector.

In the LLTM, $\boldsymbol{x}'_{ij}$ can contain several item-specific covariates. In our example, $\boldsymbol{x}'_{ij} = (x_{ij1}, x_{ij2}, x_{ij3}, x_{ij4}, x_{ij5})$ is a five-dimensional design vector for item j, coding the six problem types and whether the conclusion of the item was 'true' or 'false', i.e.,

$\boldsymbol{x}_{\mathrm{ij1}}$**:** intercept, 1 for all i

$\boldsymbol{x}_{\mathrm{ij2}}$**:** 1 for the MP problems, 0 for the MT problems

$\boldsymbol{x}_{\mathrm{ij3}}$ **and** $\boldsymbol{x}_{\mathrm{ij4}}$**:** two dummy variables encoding the MP+C, DS and DMP problems:

1	0 for MP+C
0	1 for DS
0	0 for DMP

$\boldsymbol{x}_{\mathrm{ij5}}$**:** 1 for the 'false' problems and 0 for the 'true' problems.

In Section 4, the LLTM will be analysed and discussed.

2.2 Finite mixtures of logistic random-effects models

Rijmen and De Boeck (2002) investigate the hypothesis that there are three levels of performance associated with conditional reasoning. This can be done by extending model (1) to incorporate finite mixtures of normal distributions, i.e., we will assume that

$$\begin{aligned} \text{logit}(P(y_{ij}=1|\boldsymbol{b}_i)) &= \log\left(\frac{p_{ij}}{1-p_{ij}}\right) = \boldsymbol{x}'_{ij}\boldsymbol{\beta} + \boldsymbol{z}'_{ij}\boldsymbol{b}_i, \\ \boldsymbol{b}_i &\sim \sum_{g=1}^{G} \pi_g \mathcal{N}(\boldsymbol{\mu}_g, R) \\ y_{ij}|\boldsymbol{b}_i &\sim \text{Bernoulli}(p_{ij}) \end{aligned} \tag{2}$$

where G is the number of mixture components. The probability of belonging to component g is π_g, such that $\sum_{g=1}^{G} \pi_g = 1$. Further, $\boldsymbol{\mu}_g$ is the mean of the gth component and it is assumed that each component has the same covariance matrix R. This constraint is needed to avoid infinite likelihoods (Böhning, 1999, p. 199). Let the density of a multivariate normal distribution with mean $\boldsymbol{\mu}_g$ and covariance matrix R be denoted by $\phi_{\boldsymbol{\mu}_g,R}(.)$. The joint density function of $\boldsymbol{y}_i$ can then be written as

$$\begin{aligned} f_i(\boldsymbol{y}_i) &= \int f_i(\boldsymbol{y}_i|\boldsymbol{b}_i) g(\boldsymbol{b}_i)\, d\boldsymbol{b}_i \\ &= \int f_i(\boldsymbol{y}_i|\boldsymbol{b}_i) \sum_{g=1}^{G} \pi_g \phi_{\boldsymbol{\mu}_g,R}(\boldsymbol{b}_i)\, d\boldsymbol{b}_i \\ &= \sum_{g=1}^{G} \pi_g \int f_i(\boldsymbol{y}_i|\boldsymbol{b}_i) \phi_{\boldsymbol{\mu}_g,R}(\boldsymbol{b}_i)\, d\boldsymbol{b}_i \\ &= \sum_{g=1}^{G} \pi_g f_{ig}(\boldsymbol{y}_i). \end{aligned}$$

It is seen that the density function of $\boldsymbol{y}_i$ comes from a mixture of densities with component probabilities $\pi_1, \ldots, \pi_G$ and component densities $f_{ig}(\boldsymbol{y}_i)$.

Let us now write $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_G)'$, the vector of all component probabilities and $\boldsymbol{\mu} = (\boldsymbol{\mu}_1', \ldots, \boldsymbol{\mu}_G')'$, the vector of all component means. Further, $\boldsymbol{\psi} = (\boldsymbol{\beta}', \boldsymbol{\mu}', \text{vec}(R)')'$ where vec(R) is a vector with the upper-triangular elements of R stacked on top of each other and $\boldsymbol{\theta} = (\boldsymbol{\psi}', \boldsymbol{\pi}')'$, the vector of all parameters in the model. We will now have to maximise the following loglikelihood

$$l(\boldsymbol{\theta}; \boldsymbol{y}) = \sum_{i=1}^{N} \log \left\{ \sum_{g=1}^{G} \pi_g f_{ig}(\boldsymbol{y}_i; \boldsymbol{\psi}) \right\}. \tag{3}$$

This loglikelihood will be maximised using the Expectation-Maximisation-algorithm (EM), introduced by Dempster, Laird and Rubin (1977), which is typically used for mixture problems. This will be described in the next section.

A further extension of model (2) is to allow the parameter vector $\boldsymbol{\beta}$ to vary between components. E.g., Rijmen and De Boeck (2002) use a random intercept (with different means in the latent classes) and allow the effect of MP versus MT (x_{ij2}) to be different in the latent classes. Their model can be written down as:

$$\log \left(\frac{p_{ij|g}}{1 - p_{ij|g}} \right) = x_{ij1}\beta_{1g} + x_{ij2}\beta_{2g} + x_{ij3}\beta_3 + x_{ij4}\beta_4 + x_{ij5}\beta_5 + b_{ig} \tag{4}$$

where $p_{ij|g}$ is the probability that the ith person gives a correct answer on the jth question, given that he belongs to class g. Further, $\boldsymbol{x}_{ij} = (x_{ij1}, x_{ij2}, x_{ij3}, x_{ij4}, x_{ij5})'$ is defined as before. The vector $(\beta_{1g}, \beta_{2g}, \beta_3, \beta_4, \beta_5)'$ is the five-dimensional parameter vector corresponding to the design vector $\boldsymbol{x}_{ij}$ in class g. Rijmen and De Boeck (2002) only allow the intercept and the effect of MP versus MT to vary between the different components. Finally, b_{ig} is a random intercept, which corresponds to a persons individual level of reasoning ability. It is

assumed that these random effects follow a normal distribution with mean 0 and variance σ^2. It will be shown in Section 4 how this model can be fitted using our SAS-macro.

3 Fitting mixtures in logistic random-effects models

3.1 The EM-algorithm

Define the indicator variables $w_{ig}, i = 1, \ldots, N; g = 1, \ldots, G$ as follows:

$$w_{ig} = \begin{cases} 1 & \text{if the } i\text{th subject belongs to the } g\text{th component} \\ 0 & \text{otherwise.} \end{cases}$$

It follows from this definition that $P(w_{ig} = 1) = \pi_g$. The joint loglikelihood for the observed measurements $\boldsymbol{y}$ and for the vector $\boldsymbol{w}$ of all unobserved w_{ig} equals:

$$l(\boldsymbol{\theta}; \boldsymbol{y}, \boldsymbol{w}) = \sum_{i=1}^{N} l_i(\boldsymbol{\theta}; \boldsymbol{y}_i, \boldsymbol{w}_i) = \sum_{i=1}^{N} \sum_{g=1}^{G} w_{ig}[\log(\pi_g) + \log\{f_{ig}(\boldsymbol{y}; \boldsymbol{\psi})\}]. \tag{5}$$

Maximisation of $l(\boldsymbol{\theta}; \boldsymbol{y}, \boldsymbol{w})$ depends on the unknown indicator variables w_{ig}. Therefore, the expected value of $l(\boldsymbol{\theta}; \boldsymbol{y}, \boldsymbol{w})$, conditional on $\boldsymbol{y}$ will be maximised. In the first step, the Expectation-step, the conditional expected value of the loglikelihood, referred to as the objective function Q, will be calculated. Suppose that $\boldsymbol{\theta}^{(t)}$ is the current estimate of $\boldsymbol{\theta}$. Then,

$$\begin{aligned} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) &= E[l(\boldsymbol{\theta}; \boldsymbol{y}, \boldsymbol{w}) | \boldsymbol{y}, \boldsymbol{\theta}^{(t)}] \\ &= \sum_{i=1}^{N} \sum_{g=1}^{G} E[w_{ig} | \boldsymbol{y}_i, \boldsymbol{\theta}^{(t)}][\log(\pi_g) + \log\{f_{ig}(\boldsymbol{y}_i; \boldsymbol{\psi})\}] \end{aligned} \tag{6}$$

where

$$E[w_{ig}|\boldsymbol{y}_i, \boldsymbol{\theta}^{(t)}] = \left. \frac{\pi_g^{(t)} f_{ig}(\boldsymbol{y}_i; \boldsymbol{\psi}^{(t)})}{\sum_{g=1}^{G} \pi_g^{(t)} f_{ig}(\boldsymbol{y}_i; \boldsymbol{\psi}^{(t)})} \right|_{\boldsymbol{\theta}^{(t)}} \equiv \pi_{ig}(\boldsymbol{\theta}^{(t)}).$$

Here, $\pi_{ig}(\boldsymbol{\theta}^{(t)})$ is the posterior probability of the ith subject to belong to the gth population. It is seen that the E-step actually reduces to calculating the posterior probabilities.

In the second step of the algorithm, the M-step, we have to maximise the objective function $Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)})$ with respect to $\boldsymbol{\theta}$ in order to get the updated parameter vector $\boldsymbol{\theta}^{(t+1)}$. It is seen from (6) that the objective function can be written as a sum of two parts:

$$\begin{aligned} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) &= Q_1(\boldsymbol{\pi}; \boldsymbol{\theta}^{(t)}) + Q_2(\boldsymbol{\psi}; \boldsymbol{\theta}^{(t)}) \\ &= \sum_{i=1}^{N} \sum_{g=1}^{G} \pi_{ig} \log(\pi_g) + \sum_{i=1}^{N} \sum_{g=1}^{G} \pi_{ig} \log\{f_{ig}(\boldsymbol{y}_i; \boldsymbol{\psi})\}. \end{aligned} \quad (7)$$

Maximising the first part of (7) results in updating the component probabilities in the following way:

$$\pi_g^{(t+1)} = \frac{1}{N} \sum_{i=1}^{N} \pi_{ig}(\boldsymbol{\theta}^{(t)}),$$

i.e., the updated estimate of the gth component probability is the average of the posterior probabilities in this component.

The second part of (7) is more difficult to maximise, since it requires numerical procedures such as Newton-Raphson. We will now explain how standard software, such as the SAS PROC NLMIXED, can be used to maximise this part of the objective function. We want to maximise

$$Q_2(\boldsymbol{\psi}; \boldsymbol{\theta}^{(t)}) = \sum_{i=1}^{N} \sum_{g=1}^{G} \pi_{ig}(\boldsymbol{\theta}^{(t)}) \log\{f_{ig}(\boldsymbol{y}_i; \boldsymbol{\psi})\} \quad (8)$$

with respect to $\boldsymbol{\psi}$. If the posterior probabilities $\pi_{ig}(\boldsymbol{\theta}^{(t)})$ were integers, this would be the loglikelihood for the homogeneous model based on observations from $\sum_{i=1}^{N}\sum_{g=1}^{G}\pi_{ig}(\boldsymbol{\theta}^{(t)})$ individuals.

Maximisation of (8) is equivalent to maximising

$$\begin{aligned} A.Q_2(\boldsymbol{\psi};\boldsymbol{\theta}^{(t)}) &= \sum_{i=1}^{N}\sum_{g=1}^{G} A.\pi_{ig}(\boldsymbol{\theta}^{(t)})\log\{f_{ig}(\boldsymbol{y}_i;\boldsymbol{\psi})\} \\ &= \sum_{i=1}^{N}\sum_{g=1}^{G} a_{ig}(\boldsymbol{\theta}^{(t)})\log\{f_{ig}(\boldsymbol{y}_i;\boldsymbol{\psi})\} \end{aligned} \tag{9}$$

where A is an arbitrary constant and $a_{ig}(\boldsymbol{\theta}^{(t)}) = A.\pi_{ig}(\boldsymbol{\theta}^{(t)})$. By taking A sufficiently large, and by rounding off the numbers a_{ig} to integers, the loglikelihood in (9) can be maximised as if it was a loglikelihood coming from an homogeneous model with $\sum_{i=1}^{N}\sum_{g=1}^{G} a_{ig}$ subjects. This will be an approximation to maximising the loglikelihood in (8). The larger A, the better this approximation will be. However, in practice, we would have to multiply our dataset A times, which would increase the computation time substantially. Fortunately, the SAS procedure NLMIXED provides a REPLICATE statement which can be used when modelling datasets where different subjects have identical data. The use of this REPLICATE statement allows us to increase A without affecting the computation time, such that the loglikelihood in (8) can be approximated arbitrarily close.

The algorithm will iterate between the E-step and M-step until the difference between two successive loglikelihood evaluations in (3) is smaller than some small value ϵ, that is until

$$|l(\boldsymbol{\theta}^{(t)};\boldsymbol{y}) - l(\boldsymbol{\theta}^{(t+1)};\boldsymbol{y})| < \epsilon.$$

In practice one often uses $\epsilon = 1e-08$ as stopping criterion. The maximum likelihood estimate of $\boldsymbol{\theta}$ is denoted by $\widehat{\boldsymbol{\theta}}$. In the SAS-macro it is possible to specify other stopping criteria such as the maximal absolute difference between two successive parameter estimates or the derivative of the loglikelihood in (3).

3.2 Standard errors

One of the drawbacks of the EM-algorithm is that it doesn't provide standard errors automatically. For this, one would have to calculate the observed information matrix, i.e., the negative of the second derivative of the loglikelihood in (3). One of the reasons for using the EM-algorithm was to avoid this calculation.

Louis (1982) provides a procedure to approximate this observed information matrix $\mathcal{I}(\boldsymbol{\theta}, \boldsymbol{y})$. In the case of finite mixture problems, it can be shown (McLachlan and Krishnan, 1997) that $\mathcal{I}(\boldsymbol{\theta}, \boldsymbol{y})$ can be approximated by the empirical observed information matrix

$$\mathcal{I}_e = \sum_{i=1}^{N} \boldsymbol{s}(\boldsymbol{y}_i, \widehat{\boldsymbol{\theta}}) \boldsymbol{s}'(\boldsymbol{y}_i, \widehat{\boldsymbol{\theta}})$$

where

$$\boldsymbol{s}(\boldsymbol{y}_i, \widehat{\boldsymbol{\theta}}) = E\left\{ \left. \frac{\partial l_i(\boldsymbol{\theta}; \boldsymbol{y}_i, \boldsymbol{w}_i)}{\partial \boldsymbol{\theta}} \right| \boldsymbol{y}_i \right\}.$$

It is seen that this approximation can be expressed in terms of the conditional expectation of the gradient of the loglikelihood in (5), evaluated in $\widehat{\boldsymbol{\theta}}$. The computation of the second order derivative of this loglikelihood is not needed.

3.3 Empirical Bayes estimation

The EB estimates in the SAS PROC NLMIXED are defined as the mode of $f_i(\boldsymbol{b}_i|\boldsymbol{y}_i, \boldsymbol{\theta}) \propto f_i(\boldsymbol{y}_i|\boldsymbol{b}_i, \boldsymbol{\theta})g(\boldsymbol{b}_i)$, the posterior distribution of the random effects, conditional on $\boldsymbol{y}_i$. However, because of the possible multimodality of the random-effects distribution under the heterogeneous model, this definition is not suitable anymore. The posterior distribution of $\boldsymbol{b}_i$ can also be written as

$$f_i(\boldsymbol{b}_i|\boldsymbol{y}_i, \boldsymbol{\theta}) = \sum_{g=1}^{G} \pi_{ig}(\boldsymbol{\theta}) f_{ig}(\boldsymbol{b}_i|\boldsymbol{y}_i, \boldsymbol{\psi}),$$

where $f_{ig}(\boldsymbol{b}_i|\boldsymbol{y}_i, \boldsymbol{\psi})$ is the posterior density function of $\boldsymbol{b}_i$, conditional on the fact that $\boldsymbol{b}_i$ was sampled from component g in the mixture. Therefore, it is natural to calculate the EB estimates under the heterogeneous model as

$$\widehat{\boldsymbol{b}}_i \;=\; \sum_{g=1}^{G} \pi_{ig}(\widehat{\boldsymbol{\theta}}) \widehat{\boldsymbol{b}}_i^{(g)}$$

where $\widehat{\boldsymbol{b}}_i^{(g)}$ is the EB estimate of the random effect for the ith subject in the gth component.

3.4 Classification of clustered data

Classifying clustered data based on the EB estimates can be potentially dangerous, because these EB estimates have different distributions for all subjects and are shrunk toward the mean. It is common practice, when working with mixtures, to classify these clusters based on the posterior probabilities. A subject will be classified into the component for which it has the highest posterior probability to belong to.

3.5 Accelerating the EM-algorithm

The EM-algorithm often converges very slowly because of flat loglikelihoods. Therefore, we have implemented the accelerator of Jamshidian and Jennrich (1997). They argue that the accelerator needs good starting values and recommend to use a few EM-steps until the difference between two successive values of the loglikelihood is smaller than 0.5 before switching on the accelerator.

4 An application of the SAS-macro

4.1 The linear logistic test model

Let us now first have a look at the LLTM of Section 2.1. This can be fitted using the SAS PROC NLMIXED (SAS Version 8.0, 1999) in the following way:

```
proc nlmixed data=reasoning;
parms beta1=1 beta2=0 beta3=0 beta4=0 beta5=0 s2=1;
eta = b + beta1*X1 + beta2*X2 + beta3*X3 + beta4*X4 + beta5*X5;
expeta = exp(eta); p = expeta/(1+expeta);
model y ~ binary(p);
random b ~ normal(0,s2) subject=idnr;
estimate 'MP+C versus DS' beta3 - beta4;
run;
```

In this case the random intercept, b, follows a normal distribution with mean 0 and variance σ^2. The PARMS statement gives starting values for the parameters. The loglikelihood for this model was -2642.29 and the maximum likelihood estimates can be found in Table 2. Since $\widehat{\beta}_2$ is positive, it is seen that questions with a MP are easier to answer than questions with a MT. For the second factor in the model, the parameter β_3 expresses the difference between MP+C and DMP and β_4 expresses the difference between DS and DMP. The difference between MP+C and DS can be estimated using an estimate statement and is the difference between β_3 and β_4. From Table 2 we see that MP+C is easier than DMP and DMP is easier to answer than DS. Finally, questions with a 'false' answer are more difficult to answer than questions with a true answer. This can be seen from $\widehat{\beta}_5$ which is negative.

[Table 2 about here.]

A figure of the estimated random-effects distribution in the one-component model together with the EB-estimates of the random effects is shown in Figure 1(a). Only 15 different EB-estimates can be distinguished because each person gave at least 10 out of 24 correct answers. Since we only fit a random-intercepts model, the sufficient statistic is the sum of the correct answers, leaving only 15 different possibilities.

[Figure 1 about here.]

4.2 Latent class analyses

Rijmen and De Boeck (2002) tried to find subclasses of individuals in this dataset and came up with a 3-component model, as shown in Section 2.2. The following syntax can be used in

our macro to fit model (4). First, we have to create a dataset containing starting values for the different parameters in the model. Starting values were taken from the one-component analysis (see Table 2), except for σ^2 which was taken a little bit smaller than 0.65 because it represents the variance within a component and not the total variance.

```
data startv;
input parameter$ estimate;
cards;
beta11 0.5
beta12 1
beta13 1.5
beta21 0.8
beta22 0.9
beta23 1
beta3 0.86
beta4 -0.81
beta5 -0.55
s2 0.5
;
run;


options nonotes;
```

```
run;


%HetNlmixed(DATA = reasoning, OPTIONS = %str(qpoints=5 noad),
PARMS = startv, SUBJECT = idnr, RESPONSE = y, COMPSPEC = X1 X2,
PROGRAMSTAT =
%str(eta = b*X1 + beta11*X11 + beta12*X12 + beta13*X13 + beta21*X21 +
beta22*X22 + beta23*X23 + beta3*X3 + beta4*X4 + beta5*X5;
expeta = exp(eta); p = expeta/(1+expeta);), MODELSTAT =
%str(y ~ binary(p)), RANDOMSTAT =
%str(b ~ normal(0,s2)), ESTIMSTAT = %str(
estimate 'overall intercept 1' beta11+beta21/2+(beta3+beta4)/3+beta5/2;
estimate 'overall intercept 2' beta12+beta22/2+(beta3+beta4)/3+beta5/2;
estimate 'overall intercept 3' beta13+beta23/2+(beta3+beta4)/3+beta5/2;
estimate 'deviation intercept 3-1' beta13+beta23/2-beta11-beta21/2;
estimate 'deviation intercept 2-1' beta12+beta22/2-beta11-beta21/2;
estimate 'overall effect X2' beta21*PI1+beta22*PI2+beta23*(1-PI1-PI2);),
G = 3, DECISION = 2, STOPRULE = 1e-08, A = 1e06, ACCEL = yes,
ACCSTART = 0.5, MAXITER = 1000, ENDPOST = poster, EB = ebest,
EBmean = %str('beta11','beta12','beta13'));
```

We will now briefly describe the syntax of the macro. Details can be found in Spiessens, Verbeke and Komárek (2002).

In the DATA statement, the name of the dataset is specified. The OPTIONS statement can be used to specify options for the SAS PROC NLMIXED. Here, we have used non-adaptive Gaussian quadrature with 5 quadrature points to speed up the algorithm. As Lesaffre and Spiessens (2001) show, one has to be very careful when non-adaptive Gaussian quadrature is used, but in this case, satisfactory results were obtained with this small number of quadrature points. The dataset containing the starting values is given in the PARMS statement. The variable containing the identification numbers of the subjects ($idnr$) is put in the SUBJECT statement. The RESPONSE statement contains the name of the response variable (y). The component specific effects that are used in the model are specified in the COMPSPEC statement. In this case, X1, which is also a random effect, and X2 are specified. The PROGRAMSTAT and MODELSTAT statements contain similar programming and modelling statements as used in the SAS PROC NLMIXED, except that now we specify three different means $\beta_{11}, \beta_{12}, \beta_{13}$ for X1 and $\beta_{21}, \beta_{22}, \beta_{23}$ for X2 corresponding to the three latent classes. The RANDOMSTAT statement is also similar to the RANDOM statement of SAS PROC NLMIXED. Here, we only have one random effect b, which corresponds to the intercept X1. In the ESTIMSTAT statement, different estimate statements can be specified. This way, we can reproduce the estimates of Rijmen and De Boeck (2002), who use a different parametrisation, or calculate the overall effect of X2 over the different classes. The number of components is specified in the G statement. The DECISION and STOPRULE statements allow to change the convergence criterion. The default (DECISION $= 2$) is that the algorithm will stop as soon as two successive loglikelihood evaluations are smaller than

STOPRULE. One can change the stopping criterion to the maximum absolute difference between two successive parameter estimates (DECISION = 1) or to the derivative of the loglikelihood in (3) (DECISION = 3). By default, STOPRULE = 1e-08. The multiplication factor used in (9) is specified in the A statement. The statement 'ACCEL = yes' specifies that the acceleration of Jamshidian and Jennrich (1997) has to be used. In this case the acceleration will start when the difference between two successive loglikelihoods is smaller than ACCSTART = 0.5. Finally, the posterior probabilities and empirical Bayes estimates are written to the SAS datasets 'poster' and 'ebest'. This is specified in the ENDPOST and EB statements. The EBMEAN statement is necessary to calculate the EB-estimates. More details about the macro can be found in Spiessens, Verbeke and Komárek (2002).

The loglikelihood of this model was equal to -2595.69 and the maximum likelihood estimates can be found in Table 3. Note that likelihood ratio tests are not allowed to compare the one-component with the three-component model, because of boundary problems (Böhning, 1999). Model comparison could be based on other criteria such as the Akaike's information criterion (AIC, Akaike, 1974) or the Bayesian information criterion (BIC, Schwarz, 1978). Based in these criteria, the three-component model is better than the one-component model.

[Table 3 about here.]

From Table 3 it is seen that there is a large component with a component probability of 0.70, a smaller component (0.26) and a very tiny component (0.04). From the intercepts (β_{11},β_{12} and β_{13}) and from the overall intercepts in the different classes (averaged over the other covariates in the model as was done by Rijmen and De Boeck (2002)) it is seen that

the reasoning ability is highest for the second class, intermediate for the first class and lowest for the third class. There is a significant difference between the second and first class, but the difference between the third and first class is not significant. In all three groups, a MP question was easier than a MT question. The difference was largest in the third component and smallest in the first component. The parameters β_3, β_4 and β_5 do not differ much from the one-component analysis and the same conclusions can be drawn. Rijmen and De Boeck (2002) interpret the first class as corresponding to unsophisticated reasoners and the second class to intermediate reasoners. There is no class of sophisticated reasoners, however, Rijmen and De Boeck (2002) note that this class might show up if the study was conducted among adults.

In Figure 1(b) the estimated random-effects distribution of the three-component model is shown, together with the EB-estimates, calculated under this model. Comparing this figure to Figure 1(a), it is clearly seen that clusters could not have been found based on the EB-estimates of the one-component model due to the shrinkage effect. Also, in Figure 1(b) the shrinkage effect is visible. It is seen that the EB-estimates are shrunk toward the overall mean, compared to the modes of the different components.

To show the effect of speed of convergence, we performed the analysis with and without the acceleration. Without the acceleration, convergence was reached after 107 iterations. The analysis took 4:34:14 hours on a Windows NT workstation 4.0, Pentium III 800 Mhz processor and 256 Mb RAM. With the acceleration, convergence was reached after 35 iterations and 1:49:25 hours. The acceleration started after 28 iterations. Another way of

accelerating the EM-algorithm which is often adopted is to perform only 1 iteration in the M-step. This can be done by specifying `%str(maxiter=1)` in the OPTIONS statement of the macro. However, in this case the algorithm converged only after 301 iterations and took 4:27:22 hours. It is seen that there is only a small gain in time compared to performing a full M-step, because of the high number of iterations.

5 Discussion

A method for fitting finite mixtures of nonlinear and generalised linear mixed models was described. It was shown how latent class analyses involving random effects can be performed using a SAS-macro which was previously introduced by Spiessens, Verbeke and Komárek (2002). Also mixtures of IRT models (Rasch, LLTM model, ...) can be fitted using the macro. We have applied the macro to a study on complex reasoning ability which was conducted by Rijmen and De Boeck (2001, 2002). The macro can also be used for latent class analyses were no random effects are involved. In that case, no RANDOMSTAT statement should be specified and the variables which have a different effect in the components are given in the COMPSPEC statement.

The method is based on extending the assumption that the random effects come from a normal distribution. This assumption is often made for computational reasons, or software restrictions. However, as shown by Verbeke and Lesaffre (1996), this assumption is difficult to check, because the random effects are latent, and never observed. Often, empirical Bayes (EB) estimates are used to study the distribution of the random effects, but these estimates

have different distributions for all subjects, unless all subjects have the same covariates. The EB estimates are also shrunk toward the mean, which makes them unsuitable for model diagnostics, as was shown in our example. Further, if this assumption fails to hold, the estimates of the parameters in nonlinear and in generalised linear mixed models are biased (Spiessens *et al.* 2002; Hartford and Davidian, 2000). Verbeke and Lesaffre (1997) show that the parameters in linear mixed models do not suffer from this problem. In practice, misspecifying the random-effects distributions regularly happens, e.g., when an important categorical variable has been left out of the model.

A drawback of the macro is that convergence can be very slow. However, an acceleration can be used to speed up the rate of convergence. In our example, using the acceleration, the algorithm converged 60% faster compared to the algorithm without the acceleration.

No formal test for the number of components has been implemented yet. Usual likelihood ratio tests are not valid, because of boundary problems (Böhning, 1999). Often one uses other criteria like AIC or BIC to compare different models. These have also been implemented in the macro.

Finally, the macro can be downloaded from the homepage of the Biostatistical Centre: *http://www.kuleuven.ac.be/biostat*.

Acknowledgement

This work was supported by Onderzoeksfonds Katholieke Universiteit Leuven PDM/01/157. Frank Rijmen was supported by the Fund for Scientific Research Flanders (FWO).

References

Akaike, H. (1974), "A new look at the statistical model identification", *IEEE Transactions on automatic control*, **19**, 716–723.

Böhning, D. (1999), *Computer-Assisted Analysis of Mixtures and Applications : meta-analysis, disease mapping and others*, Number 81 in Monographs on Statistics and Applied Probability, Chapman & Hall/CRC.

Breslow, N.E. and Clayton, D.G. (1993), "Approximate inference in generalized linear mixed models", *Journal of the American Statistical Association*, **88**, 9–25.

Davidian, M. and Giltinan, D.M. (1995), *Nonlinear models for repeated measurement data*, Chapman & Hall.

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

Hartford, A. and Davidian, M. (2000), "Consequences of misspecifying assumptions in non-linear mixed effects models", *Computational Statistics and Data Analysis*, **34**, 139–164.

Jamshidian, M. and Jennrich, R.I. (1997), "Acceleration of the EM algorithm by using quasi-Newton methods", *Journal of the Royal Statistical Society, Series B*, **59**, 569–587.

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.

Louis, T.A. (1982), "Finding the observed information matrix when using the EM algorithm", *Journal of the Royal Statistical Society, Series B*, **44**, 226–233.

McLachlan, G.J. and Krishnan, T. (1997), *The EM algorithm and extensions*, John Wiley & Sons, Inc.

Mislevy, R.J. and Verhelst, N. (1990), "Modeling item responses when different subjects employ different solution strategies", *Psychometrika*, **55**, 195–215.

Muthén, B. and Shedden, K. (1999), "Finite mixture modeling with mixture outcomes using the EM-algorithm", *Biometrics*, **55**, 463–469.

Rijmen, F. and De Boeck, P. (2001), "Propositional reasoning: The differential contribution of "rules" to the difficulty of complex reasoning problems", *Memory & Cognition*, **29**, 165–175.

Rijmen, F. and De Boeck, P. (2002), "A latent class model for individual differences in the interpretation of conditionals", *Submitted*.

Rijmen, F., Tuerlinckx, F., and De Boeck, P. (2002), "A nonlinear mixed model framework for IRT models", *Submitted*.

SAS Institute Inc., Cary, NC: SAS Institute Inc. (1999), *SAS OnlineDoc, Version 8*.

Schwarz, G. (1978), "Estimating the dimension of a model", *The Annals of Statistics*, **6**, 461–464.

Spiessens, B., Lesaffre, E., Verbeke, G., and Kim, K. (2002), "Group sequential methods for an ordinal logistic random-effects model under misspecification", *Accepted in Biometrics*.

Spiessens, B., Verbeke, G., and Komárek, A. (2002), "The use of mixed models for longitudinal count data when the random-effects distribution is misspecified", *Submitted*.

Verbeke, G. and Lesaffre, E. (1996), "A linear mixed-effects model with heterogeneity in the random-effects population", *Journal of the American Statistical Association*, **91**, 217–221.

Verbeke, G. and Lesaffre, E. (1997), "The effect of misspecifying the random effects distribution in linear mixed models for longitudinal data", *Computational Statistics and Data Analysis*, **23**, 541–556.

Verbeke, G. and Molenberghs, G. (2000), *Linear mixed models for longitudinal data*, Springer Series in Statistics, Springer-Verlag, New-York.

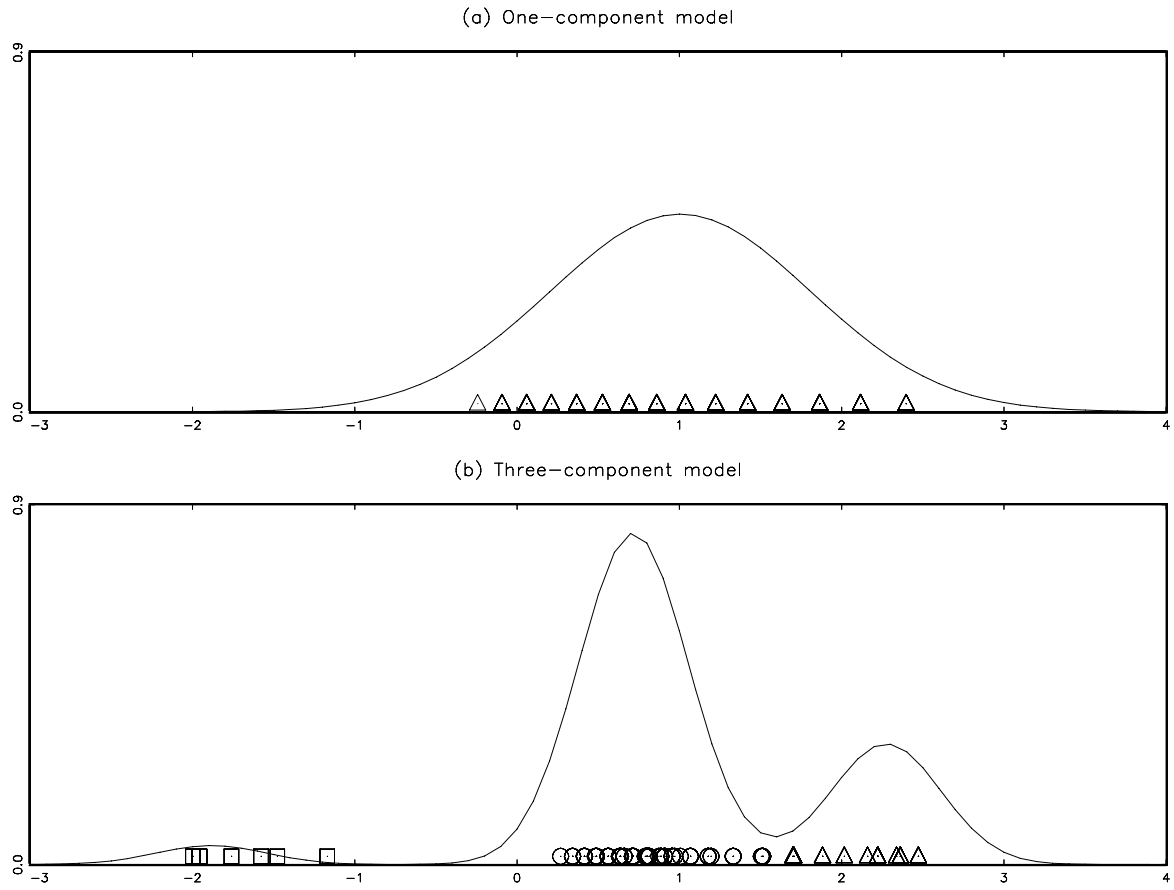


Figure 1*: (a) Estimated random-effects distribution and EB-estimates (△) under the one-component model. (b) Estimated random-effects distribution and EB-estimates under the three-component model (◯ the first component; △ the second component; □ the third component).*

Table 1: *The six problem types for the experiment of Rijmen and De Boeck (2001)*

	Modus ponens	*Modus tollens*
Modus ponens + *Conjunction*	IF p, THEN q p IF r, THEN s r	IF p, THEN q p IF r, THEN s NOT s
	q AND s	q AND NOT r
Disjunctive syllogism	p OR q IF r, THEN NOT q r	p OR q IF q, THEN r NOT r
	p	p
Disjunctive modus ponens	IF p OR q, THEN r IF s, THEN q s	IF p OR NOT q, THEN r IF q, THEN s NOT s
	r	r

Table 2: *Maximum likelihood estimates for the one-component model* ($ll = -2642.29, AIC = 5296.58, BIC = 5316.78$)

Parameter	Estimate	SE	p-value
$\widehat{\beta}_1$	1.01	0.09	<0.0001
$\widehat{\beta}_2$	0.97	0.07	<0.0001
$\widehat{\beta}_3$	0.86	0.09	<0.0001
$\widehat{\beta}_4$	-0.81	0.08	<0.0001
$\widehat{\beta}_3 - \widehat{\beta}_4$	1.67	0.09	<0.0001
$\widehat{\beta}_5$	-0.55	0.07	<0.0001
$\widehat{\sigma}^2$	0.65	0.10	<0.0001

Table 3: *Maximum likelihood estimates for the three-component model (*$ll = -2595.69, AIC = 5215.38, BIC = 5255.77$*)*

Parameter	Estimate	SE	p-value
$\widehat{\pi}_1$	0.70	0.05	<0.0001
$\widehat{\pi}_2$	0.26	0.05	<0.0001
$\widehat{\pi}_3$	0.04	0.01	0.0045
$\widehat{\beta}_{11}$	0.72	0.10	<0.0001
$\widehat{\beta}_{12}$	2.27	0.19	<0.0001
$\widehat{\beta}_{13}$	-1.88	0.39	<0.0001
$\widehat{\beta}_{21}$	0.73	0.09	<0.0001
$\widehat{\beta}_{22}$	1.46	0.35	<0.0001
$\widehat{\beta}_{23}$	5.13	0.52	<0.0001
$\widehat{\beta}_3$	0.88	0.09	<0.0001
$\widehat{\beta}_4$	-0.83	0.07	<0.0001
$\widehat{\beta}_5$	-0.56	0.08	<0.0001
$\widehat{\sigma}^2$	0.11	0.07	0.1008
Estimate statements			
overall intercept 1	0.82	0.07	<0.0001
overall intercept 2	2.74	0.25	<0.0001
overall intercept 3	0.42	0.31	0.1755
deviation intercept 2-1	1.92	0.21	<0.0001
deviation intercept 3-1	-0.40	0.32	0.2113
overall effect X2	1.10	0.14	<0.0001