Technical Guide for Latent GOLD Choice 4.0: Basic and Advanced[1]

Jeroen K. Vermunt and Jay Magidson

Statistical Innovations Inc.

(617) 489-4490

http://www.statisticalinnovations.com

[1] This document should be cited as "J.K. Vermunt and J. Magidson (2005). Technical Guide for Latent GOLD Choice 4.0: Basic and Advanced. Belmont Massachusetts: Statistical Innovations Inc."

For more information about Statistical Innovations Inc., please visit our website at
http://www.statisticalinnovations.com or contact us at

Statistical Innovations, Inc.
375 Concord Avenue, Suite 007
Belmont, MA 02478
e-mail: will@statisticalinnovations.com

Technical Guide for Latent GOLD Choice 4.0: Basic and Advanced

01/30/06

Contents

Part I: Basic Model Options, Technical Settings, and Output Sections

1 Introduction to Part I (Basic Models)

Latent GOLD Choice is a program for the analysis of various types of preference data; that is, data containing (partial) information on respondents' preferences concerning one or more sets of alternatives, objects, options, or products. Such data can be obtained from different response formats, the most important of which are:

- first choice out of a set of M alternatives, possibly including a none option,
- paired comparison,
- full ranking of a set of M alternatives,
- partial ranking of a set of M alternatives,
- best and worst choices out of a set of M alternatives; also referred to as maximum-difference scaling,
- binary rating of a single alternative (positive-negative, yes-no, like-dislike),
- polytomous rating of a single alternative, e.g., on a 5-point scale,
- assigning a probability to a set of M alternatives; also referred to as constant-sum data,
- distribution of a fixed number of points (votes, chips, dollars, etc.) among a set of M alternatives; also referred to as an allocation format,
- pick any out of a set of M alternatives,
- pick k (a prespecified number) out of a set of M alternatives, which is an example of what can be called a joint choice.

Latent GOLD Choice will accept each of these formats, including any combination of these.

The purpose of a discrete choice analysis is to predict stated or revealed preferences from characteristics of alternatives, choice situations, and respondents. The regression model that is used for this purpose is the conditional logit model developed by McFadden (1974). This is an extended multinomial logit model that allows the inclusion of characteristics of the alternatives – attributes such as price – as explanatory variables. Although the conditional logit model was originally developed for analyzing *first* choices, each of the other response formats can also be handled by adapting the basic model to the format concerned. For example, a ranking task is treated as a sequence of first choices, where the alternatives selected previously are eliminated; a rating task is modelled by an adjacent-category ordinal logit model, which is a special type of conditional logit model for ordinal outcome variables.

Latent GOLD Choice is not only a program for modeling choices or preferences, but also a program for latent class (LC) analysis. A latent class or finite mixture structure is used to capture preference heterogeneity in the population of interested. More precisely, each latent class corresponds to a population segment that differs with respect to the importance (or weight) given to the attributes of the alternatives when expressing that segment's preferences. Such a discrete characterization of unobserved heterogeneity is sometimes referred to as a nonparametric random-coefficients approach (Aitkin, 1999; Laird, 1978; Vermunt, 1997; Vermunt and Van Dijk, 2001; Vermunt and Magidson, 2003). Latent GOLD Choice implements a nonparametric variant of the random-coefficient or mixed conditional logit model (Andrews et al., 2002; Louviere et al., 2000; McFadden and Train, 2000). The LC choice model can also be seen as a variant of the LC or mixture regression model (Vermunt and Magidson, 2000; Wedel and DeSarbo, 1994, 2002).

Most studies will contain multiple observations or multiple replications per respondent: e.g., respondents indicate their first choice for several sets of products or provide ratings for various products. This introduces dependence between observations. It is this dependence caused by the repeated measures that makes it possible to obtain stable estimates of the class-specific regression parameters.

A third aspect of the model implemented in Latent GOLD Choice is that class membership can be predicted from individual characteristics (covariates). In other words, one can not only identify latent classes, clusters, or segments that differ with respect to their preferences, but it is also possible to predict to which (unobserved) subgroup an individual belongs using covariates. Such a profiling of the latent classes substantially increases the practical usefulness of the results and improves out-of-study prediction of choices (Magidson, Eagle, and Vermunt, 2003; Natter and Feurstein, 2002; Vermunt and Magidson, 2002).

The next section describes the LC models implemented in Latent GOLD Choice. Then, attention is paid to estimation procedures and the corresponding technical options of the program. The output provided by the program is described in the last section.

Several tutorials are available to get you up and running quickly. These include:

- cbcRESP.sav - a simulated choice experiment
  - Tutorial 1: Using LG Choice to Estimate Discrete Choice Models
  - Tutorial 2: Using LG Choice to Predict Future Choices
- brandABresp.sav - a simulated brand-price choice experiment
  - Tutorial 3: Estimating Brand and Price Effects
  - Tutorial 4: Using the 1-file Format
- bank45.sav & bank9-1-file.sav - real data from a bank segmentation study
  - Tutorial 5: Analyzing Ranking Data
  - Tutorial 6: Using LG Choice to Estimate max-diff (best-worst) and Other Partial Ranking Models
- conjoint.sav, ratingRSP.sav, ratingALT.sav, ratingSET.sav: simulated data utilizing a 5-point ratings scale
  - Tutorial 7: LC Segmentation with Ratings-based Conjoint Data
  - Tutorial 7A: LC Segmentation with Ratings-based Conjoint Data

All of the above tutorials are available on our website at
http://www.statisticalinnovations.com/products/choice.html#tutorialslink

2 The Latent Class Model for Choice Data

In order to be able to describe the models of interest, we first must clarify some concepts and introduce some notation. The data file contains information on I cases or subjects, where a particular case is denoted by i. For each case, there are T_i replications, and a particular replication is denoted by t. Note that in the Latent GOLD Choice interface, the T_i observations belonging to the same case are linked by means of a common *Case ID*.

Let y_{it} denote the value of the dependent (or response) variable for case i at replication t, which can take on values $1 \leq m \leq M$. In other words, M is number of alternatives and m a particular alternative. Three types of explanatory variables can be used in a LC choice model: attributes or characteristics of alternatives (z^{att}_{itmp}), predictors or characteristics of replications (z^{pre}_{itq}), and covariates or characteristics of individuals (z^{cov}_{ir}). Here, the indices p, q, and r are used to refer to a particular attribute, predictor, and covariate. The total number of attributes, predictors, and covariates is denoted by P, Q, and R, respectively. Below, we will sometimes use vector notation $\mathbf{y}_i$, $\mathbf{z}_i$, and $\mathbf{z}^{cov}_i$ to refer to all responses, all explanatory variables, and all covariate values of case i, and $\mathbf{z}^{att}_{it}$ and $\mathbf{z}^{pre}_{it}$ to refer to the attribute and predictor values corresponding to replication t for case i. Another variable that plays an important role in the models discussed below is the latent class variable denoted by x, which can take on values $1 \leq x \leq K$. In other words, the total number of latent classes is denoted by K.

Two other variables that may be used in the model specification are a replication-specific scale factor s_{it} and a replication-specific weight v_{it}. Their default values are one, in which case they do not affect the model structure.

2.1 First Choices

We start with the description of the regression model for the simplest and most general response format, first choice. For simplicity of exposition, we assume that each replication or choice set has the same number of alternatives. Later on it will be shown how to generalize the model to other formats, including choice sets with unequal numbers of alternatives.

A conditional logit model is a regression model for the probability that case i selects alternative m at replication t given attribute values $\mathbf{z}^{att}_{it}$ and predictor values $\mathbf{z}^{pre}_{it}$. This probability is denoted by $P(y_{it} = m|\mathbf{z}^{att}_{it}, \mathbf{z}^{pre}_{it})$. Attributes are characteristics of the alternatives; that is, alternative m will have different attribute values than alternative m'. Predictors on the other hand, are characteristics of the replication or the person, and take on the same value across alternatives. For the moment, we assume that attributes and predictors are numeric variables. In the subsection " Coding of Nominal Variables", we explain in detail how nominal explanatory variables are dealt with by the program.

The conditional logit model for the response probabilities has the form

$$P(y_{it} = m|\mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pre}) = \frac{\exp(\eta_{m|\mathbf{z}_{it}})}{\sum_{m'=1}^{M} \exp(\eta_{m'|\mathbf{z}_{it}})},$$

where $\eta_{m|\mathbf{z}_{it}}$ is the systematic component in the utility of alternative m for case i at replication t. The term $\eta_{m|\mathbf{z}_{it}}$ is a linear function of an alternative-specific constant β_m^{con}, attribute effects β_p^{att}, and predictor effects β_{mq}^{pre} (McFadden, 1974). That is,

$$\eta_{m|\mathbf{z}_{it}} = \beta_m^{con} + \sum_{p=1}^{P} \beta_p^{att}\, z_{itmp}^{att} + \sum_{q=1}^{Q} \beta_{mq}^{pre}\, z_{itq}^{pre},$$

where for identification purposes $\sum_{m=1}^{M} \beta_m^{con} = 0$, and $\sum_{m=1}^{M} \beta_{mq}^{pre} = 0$ for $1 \leq q \leq Q$, a restriction that is known as effect coding. It is also possible to use dummy coding using either the first or last category as reference category (see subsection 2.8). Note that the regression parameters corresponding to the predictor effects contain a subscript m, indicating that their values are alternative specific.

The inclusion of the alternative-specific constant β_m^{con} is optional, and models will often not include predictor effects. Without alternative-specific constants and without predictors, the linear model for $\eta_{m|\mathbf{z}_{it}}$ simplifies to

$$\eta_{m|\mathbf{z}_{it}} = \sum_{p=1}^{P} \beta_p^{att}\, z_{itmp}^{att}.$$

In a latent class or finite mixture variant of the conditional model, it is assumed that individuals belong to different latent classes that differ with respect to (some of) the β parameters appearing in the linear model for η (Kamakura and Russell, 1989). In order to indicate that the choice probabilities depend on class membership x, the logistic model is now of the form

$$P(y_{it} = m|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pre}) = \frac{\exp(\eta_{m|x,\mathbf{z}_{it}})}{\sum_{m'=1}^{M} \exp(\eta_{m'|x,\mathbf{z}_{it}})}. \tag{1}$$

Here, $\eta_{m|x,\mathbf{z}_{it}}$ is the systematic component in the utility of alternative m at replication t given that case i belongs to latent class x. The linear model for $\eta_{m|x,\mathbf{z}_{it}}$ is

$$\eta_{m|x,\mathbf{z}_{it}} = \beta_{xm}^{con} + \sum_{p=1}^{P} \beta_{xp}^{att}\, z_{itmp}^{att} + \sum_{q=1}^{Q} \beta_{xmq}^{pre}\, z_{itq}^{pre}. \tag{2}$$

As can be seen, the only difference with the aggregate model is that the logit regression coefficients are allowed to be Class specific.

In the LC choice model, the probability density associated with the responses of case i has the form

$$P(\mathbf{y}_i|\mathbf{z}_i) = \sum_{x=1}^{K} P(x) \prod_{t=1}^{T_i} P(y_{it}|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred}). \tag{3}$$

Here, $P(x)$ is the unconditional probability of belonging to Class x or, equivalently, the size of latent class x. Below, it will be shown that this probability can be allowed to depend an individual's covariate values $\mathbf{z}_i^{cov}$, in which case $P(x)$ is replaced by $P(x|\mathbf{z}_i^{cov})$.

As can be seen from the probability structure described in equation (3), the T_i repeated choices of case i are assumed to be independent of each other given class membership. This is equivalent to the assumption of local independence that is common in latent variable models, including in the traditional latent class model (Bartholomew and Knott, 1999; Goodman, 1974a, 1974b; Magidson and Vermunt, 2004). Also in random-coefficients models, it is common to assume responses to be independent conditional on the value of the random coefficients.

2.2 Rankings and Other Situations with Impossible Alternatives

In addition to models for (repeated) first choices, it is possible to specify models for rankings. One difference between first-choice and ranking data is that in the former there is a one-to-one correspondence between replications and choice sets while this is no longer the case with ranking data. In a ranking task, the number of replications generated by a choice sets equals the number of choices that is provided. A full ranking of a choice set consisting of five alternatives yields four replications; that is, the first, second, third, and fourth choice. Thus, a set consisting of M alternatives, generates $M-1$

replications. This is also the manner in which the information appears in the response data file. With partial rankings, such as first and second choice, the number of replications per set will be smaller.

The LC model for ranking data implemented in Latent GOLD Choice treats the ranking task as sequential choice process (Böckenholt, 2002; Croon, 1989; Kamakura et al., 1994). More precisely, each subsequent choice is treated as if it were a first choice out of a set from which alternatives that were already selected are eliminated. For example, if a person's first choice out of a set of 5 alternatives is alternative 2, the second choice is equivalent to a (first) choice from the 4 remaining alternatives 1, 3, 4, and 5. Say that the second choice is 4. The third choice will then be equivalent to a (first) choice from alternatives 1, 3, and 5.

The only adaptation that is needed for rankings is that it should be possible to have a different number of alternatives per set or, in our terminology, that certain alternatives are "impossible". More precisely, M is still the (maximum) number of alternatives, but certain alternatives cannot be selected in some replications. In order to express this, we need to generalize our notation slightly. Let A_{it} denote the set of "possible" alternatives at replication t for case i. Thus, if $m \in A_{it}$, $P(y_{it} = m|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pre})$ is a function of the unknown regression coefficients, and if $m \notin A_{it}$, $P(y_{it} = m|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pre}) = 0$. An easy way to accomplish this without changing the model structure is by setting $\eta_{m|x,\mathbf{z}_{it}} = -\infty$ for $m \notin A_{it}$. Since $\exp(-\infty) = 0$, the choice probability appearing in equation (1) becomes:

$$P(y_{it} = m|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pre}) = \frac{\exp(\eta_{m|x,\mathbf{z}_{it}})}{\sum_{m' \in A_{it}} \exp(\eta_{m'|x,\mathbf{z}_{it}})}$$

if $m \in A_{it}$ and $P(y_{it} = m|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pre}) = 0$ if $m \notin A_{it}$. As can be seen, the sum in the denominator is over the possible alternatives only.

When the dependent variable is specified to be a ranking variable, the specification of those alternatives previously selected as impossible alternatives is handled automatically by the program. The user can use a missing value in the sets file to specify alternatives as impossible. This makes it possible to analyze choice sets with different numbers of alternatives per set, as well as combinations of different choice formats. In the one-file data format, choice sets need not have the same numbers of alternatives. In this case, the program treats "unused" alternatives as "impossible" alternatives.

2.3 Ratings

A third type of dependent variable that can be dealt with are preferences in the form of ratings. Contrary to a first choice or ranking task, a rating task concerns the evaluation of a single alternative instead of the comparison of a set of alternatives. Attributes will, therefore, have the same value across the categories of the response variable. Thus for rating data, it is no longer necessary to make a distinction between attributes and predictors.

Another important difference with first choices and rankings is that ratings outcome variables should be treated as ordinal instead of nominal. For this reason, we use an adjacent-category ordinal logit model as the regression model for ratings (Agresti, 2002; Goodman, 1979; Magidson, 1996). This is a restricted multinomial/conditional logit model in which the category scores for the dependent variable play an important role. Let y^*_m be the score for category m. In most cases, this will be equally-spaced scores with mutual distances of one – e.g., 1, 2, 3, ... M, or 0, 1, 2, ... $M-1$ – but it is also possible to use scores that are not equally spaced or non integers. Note that M is no longer the number of alternatives in a set but the number of categories of the response variable.

Using the same notation as above, the adjacent-category ordinal logit model can be formulated as follows

$$\eta_{m|x,\mathbf{z}_{it}} = \beta^{con}_{xm} + y^*_m \cdot \left(\sum_{p=1}^{P} \beta^{att}_{xp}\, z^{att}_{itp} + \sum_{q=1}^{Q} \beta^{pre}_{xq}\, z^{pre}_{itq} \right).$$

The attribute and predictor effects are multiplied by the fixed category score y^*_m to obtain the systematic part of the "utility" of rating m . As can be seen, there is no longer a fundamental difference between attributes and predictors since attribute values and predictor effects no longer depend on m. For ratings, $\eta_{m|x,\mathbf{z}_{it}}$ is defined by substituting $y^*_m \cdot z^{att}_{itp}$ in place of the category-specific attribute values z^{att}_{itmp} in equation (2), and the category-specific predictor effects β^{pre}_{xmq} are replaced by $y^8_m \cdot \beta^{pre}_{xq}$. The relationship between the category-specific utilities $\eta_{m|x,\mathbf{z}_{it}}$ and the response probabilities is the same as in the model for first choices (see equation 1).

As mentioned above, in most situations the category scores y^*_m are equally spaced with a mutual distance of one. In such cases, $y^*_m - y^*_{m-1} = 1$, and as result

$$\log \frac{P(y_{it} = m|x, \mathbf{z}^{att}_{it}, \mathbf{z}^{pre}_{it})}{P(y_{it} = m-1|x, \mathbf{z}^{att}_{it}, \mathbf{z}^{pre}_{it})} \quad = \quad \eta_{m|x,\mathbf{z}_{it}} - \eta_{m-1|x,\mathbf{z}_{it}}$$

$$\begin{aligned} = & \left(\beta^{con}_{xm} - \beta^{con}_{x(m-1)}\right) \\ & + \sum_{p=1}^{P} \beta^{att}_{xp}\, z^{att}_{itp} + \sum_{q=1}^{Q} \beta^{pre}_{xq}\, z^{pre}_{itq}. \end{aligned}$$

This equation clearly shows the underlying idea behind the adjacent-category logit model. The logit in favor of rating m instead of $m-1$ has the form of a standard binary logit model, with an intercept equal to $\beta^{con}_{xm} - \beta^{con}_{x(m-1)}$ and slopes equal to β^{att}_{xq} and β^{pre}_{xq}. The constraint implied by the adjacent-category ordinal logit model is that the slopes are the same for each pair of adjacent categories. In other words, the attribute and predictor effects are the same for the choice between ratings 2 and 1 and the choice between ratings 5 and 4.

2.4 Replication Scale and Best-Worst Choices

A component of the LC choice model implemented in Latent GOLD Choice that has not been introduced thus far is the replication-specific scale factor s_{it}. The scale factor allows the utilities to be scaled differently for certain replications. Specifically, the scale factor enters into the conditional logit model in the following manner:

$$P(y_{it} = m | x, \mathbf{z}^{att}_{it}, \mathbf{z}^{pre}_{it}, s_{it}) = \frac{\exp(s_{it} \cdot \eta_{m|x,\mathbf{z}_{it}})}{\sum_{m'=1}^{M} \exp(s_{it} \cdot \eta_{m'|x,\mathbf{z}_{it}})}.$$

Thus, it is seen that while the scale factor is assumed to be constant across alternatives within a replication, it can take on different values between replications. The form of the linear model for $\eta_{m|x,\mathbf{z}_{it}}$ is not influenced by the scale factors and remains as described in equation (2). Thus, the scale factor allows for a different scaling of the utilities across replications. The default setting for the scale factor is $s_{it} = 1$, in which case it cancels from the model for the choice probabilities.

Two applications of this type of scale factor are of particular importance in LC Choice modeling. The first is in the analysis of best-worst choices or maximum-difference scales (Cohen, 2003). Similar to a partial ranking task, the selection of the best and worst alternatives can be treated as a sequential choice process. The selection of the best option is equivalent to a first choice. The selection of the worst alternative is a (first) choice out of the remaining alternatives, where the choice probabilities are negatively related to the utilities of these alternatives. By declaring the dependent variable to be a ranking, the program automatically eliminates the best alternative from the set available for the second choice. The fact that the second choice is not the second best but the worst can be indicated by means of a replication scale factor of -1, which will reverse the choice probabilities. More precisely, for the worst choice,

$$P(y_{it} = m|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pre}, s_{it}) = \frac{\exp(-1 \cdot \eta_{m|x,\mathbf{z}_{it}})}{\sum_{m' \in A_{it}} \exp(-1 \cdot \eta_{m'|x,\mathbf{z}_{it}})}$$

if $m \in A_{it}$ and 0 if $m \notin A_{it}$.

The second noteworthy application of the scale factor occurs in the simultaneous analysis of stated and revealed preferences. Note that use of a scale factor larger than 0 but smaller than 1 causes $s_{it} \cdot \eta_{m|x,\mathbf{z}_{it}}$ to be shrunk compared to $\eta_{m|x,\mathbf{z}_{it}}$ and as a result, the choice probabilities become more similar across alternatives. A well-documented phenomenon is that stated preferences collected via questionnaires yield more extreme choice probabilities than revealed preferences (actual choices) even if these utilities are the same (Louviere et al., 2000). A method to transform the utilities for these two data types to the same scale is to use a somewhat smaller scale factor for the revealed preferences than for the stated preferences. Assuming that the scale factor for the stated preferences is 1.0, values between 0.5 and 1.0 could be tried out for the revealed preferences; for example,

$$P(y_{it} = m|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pre}, s_{it}) = \frac{\exp(0.75 \cdot \eta_{m|x,\mathbf{z}_{it}})}{\sum_{m'=1}^{M} \exp(0.75 \cdot \eta_{m'|x,\mathbf{z}_{it}})}.$$

A limitation of the scale factor implemented in Latent GOLD Choice is that it cannot vary across alternatives. However, a scale factor is nothing more than a number by which the attributes (and predictors) are multiplied, which is something that users can also do themselves when preparing the data files for the analysis. More precisely, the numeric attributes of the alternatives may be multiplied by the desired scale factor.

2.5 Replication Weight and Constant-sum Data

The replication weights v_{it} modify the probability structure defined in equation (3) as follows:

$$P(\mathbf{y}_i|\mathbf{z}_i) = \sum_{x=1}^{K} P(x) \prod_{t=1}^{T_i} \left[P(y_{it}|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred})\right]^{v_{it}}.$$

The interpretation of a weight is that choice y_{it} is made v_{it} times.

One of the applications of the replication weight is in the analysis of constant-sum or allocation data. Instead of choosing a single alternative out of set, the choice task may be to attach a probability to each of the alternatives. These probabilities serve as replication weights. Note that with such a response format, the number of replications corresponding to a choice set will be equal to the number of alternatives. A similar task is the distribution of say 100 chips or coins among the alternatives, or a task with the instruction to indicate how many out of 10 visits of a store one would purchase each of several products presented.

Other applications of the replication weights include grouping and differential weighting of choices. Grouping may be relevant if the same choice sets are offered several times to each observational unit. Differential weighting may be desirable when analyzing ranking data. In this case, the first choice may be given a larger weight in the estimation of the utilities than subsequent choices. It is even possible to ask respondents to provide weights – say between 0 and 1 – to indicate how certain they are about their choices. In the simultaneous analysis of stated and revealed preference data, it is quite common that several stated preferences are combined with a single revealed preference. In such a case, one may consider assigning a higher weight to the single revealed preference replication to make sure that both preference types have similar effects on the parameter estimates.

2.6 Other Choice/Preference Formats

In the previous sections, we showed how to deal with most of the response formats mentioned in the introduction. To summarize, first choice is the basic format, rankings are dealt with as sequences of first choices with impossible alternatives, ratings are modelled by an ordinal logit model, best-worst choices can be treated as partial rankings with negative scale factors for the second (worst) choice, and the analysis of constant-sum data involves the use of replications weights.

A format that has not been discussed explicitly is paired comparisons (Dillon and Kumar, 1994). Paired comparisons are, however, just first choices out of sets consisting of two alternatives, and can therefore be analyzed in the same way as first choices. Another format mentioned in the introduction is binary rating. Such a binary outcome variable concerning the evaluation of a single alternative (yes/no, like/dislike) can simply be treat as a rating. The most natural scoring of the categories would be to use score 1 for the positive response and 0 for the negative response, which yields a standard binary logit model. The pick any out of M format can be treated in the same manner as binary ratings; that is, as a set of binary variables indicating whether the various alternatives are picked or not.

Another format, called joint choices, occurs if a combination of two or more outcome variables are modelled jointly. Suppose the task is to give the two best alternatives out of a set of M, which is a pick 2 out of M format. This can be seen as a single choice with $M \cdot (M-1)/2$ joint alternatives. The attribute values of these joint alternatives are obtained by summing the attribute values of the original pair of alternatives. Other examples of joint choices are non-sequential models for rankings (Böckenholt, 2002; Croon, 1989) and best-worst choices (Cohen, 2003). For example, a pair of best and worst choices can also be seen as a joint choice out of $M \cdot (M-1)$ joint alternatives. The attribute values of these joint alternatives are equal to the attribute values of the best minus the attributes of the worst. What is clear from these examples is that setting up a model for a joint choice can be quite complicated.

Another example of a situation in which one has to set up a model for a joint response variable is in capture-recapture studies (Agresti, 2002). For the subjects that are captured at least ones, one has information on capture at the various occasions. The total number of categories of the joint dependent variable is $2^T - 1$, where T is the number of time point or replications.

Note that these examples of joint choice models all share the fact that the number of possible joint alternatives is smaller than the product of the number of alternatives of the separate choices. That is, in each case, certain combinations of choices are impossible, and hence the model of interest cannot be set up as a series of independent choices. Instead, these situations should be specified as a single joint choice.

The last "choice format" we would like to mention is the combination of different response formats. The most general model is the model for first choices. Models for rankings and ratings are special cases that are obtained by adapting the model for first choices to the response format concerned. This is done internally (automatically) by the program. It is, however, also possible to specify ranking or rating models as if they were models for first choices. In the ranking case, this involves specifying additional choice sets in which earlier selected alternatives are defined as "impossible". A rating model can be specified as a first choice model by defining the categories of the dependent variable as alternatives with attribute values z_{itpm}^{att} equal to $y_m^* \cdot z_{itp}^{att}$.

Given the fact that each of the response formats can be treated as a first choice, it is possible to make any combination of the formats that were discussed. Of course, setting up the right alternatives and sets files may be quite complicated. An issue that should be taken into account when using combinations of response formats is the scaling of the replications. For example, it might be that the utilities should be scaled in a different manner for ratings than for first choices.

2.7 Covariates

In addition to the explanatory variables that we called attributes and predictors, it is also possible to include another type of explanatory variable – called covariates – in the LC model. While attributes and predictors enter in the regression model for the choices, covariates are used to predict class membership. In the context of LC analysis, covariates are sometimes referred to as concomitant or external variables (Clogg, 1981; Dayton and McReady, 1988; Kamakura et al., 1994; Van der Heijden et al., 1996).

When covariates are included in the model, the probability structure changes slightly compared to equation (3). It becomes

$$P(\mathbf{y}_i|\mathbf{z}_i) = \sum_{x=1}^{K} P(x|\mathbf{z}_i^{cov}) \prod_{t=1}^{T_i} P(y_{it}|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred}). \tag{4}$$

As can be seen, class membership of individual i is now assumed to depend on a set of covariates denoted by $\mathbf{z}_i^{cov}$. A multinomial logit is specified in which class membership is regressed on covariates; that is,

$$P(x|\mathbf{z}_i^{cov}) = \frac{\exp(\eta_{x|\mathbf{z}_i})}{\sum_{x'=1}^{K} \exp(\eta_{x'|\mathbf{z}_i})},$$

with linear term

$$\eta_{x|\mathbf{z}_i} = \gamma_{0x} + \sum_{r=1}^{R} \gamma_{rx} z_{ir}^{cov}. \tag{5}$$

Here, γ_{0x} denotes the intercept or constant corresponding to latent class x and γ_{rx} is the effect of the rth covariate for Class x. Similarly to the model for the choices, for identification, we either set $\sum_{x=1}^{K} \gamma_{rx} = 0$, $\gamma_{r1} = 0$, or

$\gamma_{rK} = 0$ for $0 \leq r \leq R$, which amounts to using either effect or dummy coding. Although in equation (5) the covariates are assumed to be numeric, the program can also deal with nominal covariates (see subsection 2.8).

We call this procedure for including covariates in a model the "active covariates method": Covariates are active in the sense that the LC choice solution with covariates can be somewhat different from the solution without covariates. An alternative method, called "inactive covariates method", involves computing descriptive measures for the association between covariates and the latent variable after estimating a model without covariate effects. More detail on the latter method is given in the subsection explaining the Profile and ProbMeans output.

Another approach that can be used to explore the relationship between covariates and the latent variable is through the use of the CHAID option. This option may be especially valuable when the goal is to profile the latent classes using many inactive covariates. This option requires the SI-CHAID 4.0 add-on program, which assesses the statistical significance between each covariate and the latent variable. For further information about the CHAID option see Section 4.9.

2.8 Coding of Nominal Variables

In the description of the LC choice models of interest, we assumed that attributes, predictors, and covariates were all numeric. This limitation is not necessary however, as Latent GOLD Choice allows one or more of these explanatory variables to be specified to be nominal. For nominal variables, Latent GOLD Choice sets up the design vectors using either effect (ANOVA-type) coding or dummy coding with the first or last category as reference category for identification. Effect coding means that the parameters will sum to zero over the categories of the nominal variable concerned, In dummy coding, the parameters corresponding to the reference category are fixed to zero.

Suppose we have a nominal attribute with 4 categories in a model for first choices. The effect coding constraint implies that the corresponding 4 effects should sum to 0. This is accomplished by defining a design matrix with 3 numeric attributes z^{att}_{it1m}, z^{att}_{it2m}, and z^{att}_{it3m}. The design matrix that is set up for the 3 non-redundant terms (β^{att}_{x1}, β^{att}_{x2}, β^{att}_{x3}) is as follows:

$$\begin{array}{lrrr} \text{category 1} & 1 & 0 & 0 \\ \text{category 2} & 0 & 1 & 0 \\ \text{category 3} & 0 & 0 & 1 \\ \text{category 4} & -1 & -1 & -1 \end{array},$$

where each row corresponds to a category of the attribute concerned and each column to one of the three parameters. Although the parameter for the last category is omitted from model, you do not notice that because it is computed by the program after the model is estimated. The parameter for the fourth category equals $-\sum_{p=1}^{3}\beta^{att}_{xp}$; that is, minus the sum of the parameters of the three other categories. This guarantees that the parameters sum to zero since $\sum_{p=1}^{3}\beta^{att}_{xp} - \sum_{p=1}^{3}\beta^{att}_{xp} = 0$.

Instead of using effect coding, it is also possible to use dummy coding. Depending on whether one uses the first or the last category as reference category, the design matrix will look like this

$$\begin{array}{lccc} \text{category 1} & 0 & 0 & 0 \\ \text{category 2} & 1 & 0 & 0 \\ \text{category 3} & 0 & 1 & 0 \\ \text{category 4} & 0 & 0 & 1 \end{array}.$$

or this

$$\begin{array}{lccc} \text{category 1} & 1 & 0 & 0 \\ \text{category 2} & 0 & 1 & 0 \\ \text{category 3} & 0 & 0 & 1 \\ \text{category 4} & 0 & 0 & 0 \end{array}.$$

Whereas in effect coding the category-specific effects should be interpreted in terms of deviation from the average, in dummy coding their interpretation is in terms of difference from the reference category. Note that the parameter for the reference category is omitted, which implies that it is equated to 0.

It also possible to work with user specified coding schemes. An example is

$$\begin{array}{lccc} \text{category 1} & 0 & 0 & 0 \\ \text{category 2} & 1 & 0 & 0 \\ \text{category 3} & 1 & 1 & 0 \\ \text{category 4} & 1 & 1 & 1 \end{array},$$

which yields parameters that can be interpreted as differences between adjacent categories. More precisely, β_{x1}^{att} is the difference between categories 2 and 1, β_{x2}^{att} between categories 3 and 2, and β_{x3}^{att} between categories 4 and 3.

As explained in the previous sections, the effect and dummy coding constraints are not only imposed on the attribute effects, but also on the constants and the predictor effects in the regression model for first choices and rankings, on the constants in the regression model for ratings, and on the intercepts and covariate effects in the regression model for the latent classes.

2.9 Known-Class Indicator

Sometimes, one has a priori information – for instance, from an external source – on the class membership of some individuals. For example, in a four-class situation, one may know that case 5 belongs to latent class 2 and case 11 to latent class 3. Similarly, one may have a priori information on which class cases do not belong to. For example, again in a four-class situation, one may know that case 19 does not belong to latent class 2 and that case 41 does not belong to latent classes 3 or 4. In Latent GOLD, there is an option – called "*Known Class*" – for indicating to which latent classes cases do *not* belong to.

Let $\boldsymbol{\tau}_i$ be a vector of 0-1 variables containing the "*Known Class*" information for case i, where $\tau_{ix} = 0$ if it is known that case i does not belong to class x, and $\tau_{ix} = 1$ otherwise. The vector $\boldsymbol{\tau}_i$ modifies the model with covariates defined in equation (4) as follows:

$$P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\tau}_i) = \sum_{x=1}^{K} \tau_{ix}\, P(x|\mathbf{z}_i^{cov}) \prod_{t=1}^{T_i} P(y_{it}|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred}).$$

As a result of this modification, the posterior probability of belonging to class x will be equal to 0 if $\tau_{ix} = 0$.

The known-class option has three important applications.

1. It can be used to estimate models with training cases; that is, cases for which class membership has been determined using a gold standard method. Depending on how this training information is obtained, the missing data mechanism will be MCAR (Missing Completely At Random, where the known-class group is a random sample from all cases), MAR (Missing At Random, where the known-class group is a random sample given observed responses and covariate values), or NMAR (Not

Missing At Random, where the known-class group is a non-random sample and thus may depend on class membership itself). MAR occurs, for example, in clinical applications in which cases with more than a certain number of symptoms are subjected to further examination to obtain a perfect classification (diagnosis). NMAR may, for example, occur if training cases that do not belong to the original sample under investigation are added to the data file.

Both in the MAR and MCAR situation, parameter estimates will be unbiased. In the NMAR situation, however, unbiased estimation requires that separate class sizes are estimated for training and non-training cases (McLachlan and Peel, 2000). This can easily be accomplished by expanding the model of interest with a dichotomous covariate that takes on the value 0 for training cases and 1 for non-training cases.

2. Another application is specifying models with a partially missing discrete variable that affects one or more response variables. An important example is the *complier average causal effect* (CACE) model proposed by Imbens and Rubin (1997), which can be used to determine the effect of a treatment conditional on compliance with the treatment. Compliance is, however, only observed in the treatment group, and is missing in the control group. This CACE model can be specified as a model in which class membership (compliance) is known for the treatment group, and which a treatment effect is specified only for the compliance class.

3. The known-class indicator can also be used to specify *multiple-group LC models*. Suppose we have a three-class model and two groups, say males and females. A multiple-group LC model is obtained by indicating that there are six latent classes, were males may belong to classes 1–3 and females to classes 4–6. To get the correct output, the grouping variable should not only be used as the known-class indicator, but also as a nominal covariate.

2.10 Zero-Inflated Models

When the zero-inflated option is used, the model is expanded with M latent classes that are assumed to respond with probability one to a certain category; that is, $P(y_{it} = m|x,\mathbf{z}_{it}^{pred}) = 1$ for $x = K + m$. Such latent classes are sometimes referred to as stayer classes (in mover-stayer models) or brand-loyal classes (in brand-loyalty models).

2.11 Restrictions on the Regression Coefficients

Various types of restrictions can be imposed on the Class-specific regression coefficients: attribute and predictor effects can be fixed to zero, restricted to be equal across certain or all Classes, and constrained to be ordered. Moreover, the effects of numeric attributes can be fixed to one. These constraints can either be used as a priori restrictions derived from theory or as post hoc restrictions on estimated models.

Certain restrictions apply to parameters within each Class, while others apply across Classes. The within-Class restrictions are:

- No Effect: the specified effect(s) are set to zero;
- Offset: the selected effect(s) are set to one, thus serving as an offset.[2] The offset effect applies to numeric attributes only.

Between-Class restrictions are:

- Merge Effects: the effects of a selected attribute/predictor are equated across 2 or more specified Classes;
- Class Independent: the effects of a selected attribute/predictor are equated across all Classes;
- Order (ascending or descending): in each Class, the effect of a selected numeric attribute/predictor is assumed to have the same sign or the effects corresponding to a selected nominal attribute/predictor are assumed to be ordered (either ascending or descending). That is, for numeric attributes/predictors, the ascending restriction implies that the Class-specific coefficients should be at least zero ($\beta \geq 0$) and the descending restriction that they are at most zero ($\beta \leq 0$). For nominal attributes/predictors, ascending implies that the coefficient of category

[2] The term offset stems from the generalized linear modeling framework. It refers to a regression coefficient that is fixed to 1, or equivalently, to a component that offsets the linear part of the regression model by a fixed amount. An offset provides the same role as a cell weight in log-linear analysis. An offset is in fact, the log of a cell weight.

$p+1$ is larger than or equal to the one of category p ($\beta_p \leq \beta_{p+1}$, for each p) and descending that the coefficient of category $p+1$ is smaller than or equal to the one of category p ($\beta_p \geq \beta_{p+1}$, for each p).

The "Class Independent" option can be used to specify models in which some attribute and predictor effects differ across Classes while others do not. This can either be on a priori grounds or can be based on the test statistics from previously estimated models. More specifically, if the Wald(=) test is not significant, it makes sense to check whether an effect can be assumed to be Class independent.

There is a special variant of the Class-independent option called "No Simple" that can be used in conjunction with the constants in a rating model. With this option, the constants are modeled as $\beta^{con}_{xm} = \beta^{con}_{\cdot m} + \beta^{con}_{x\cdot} \cdot y^*_m$, where $\beta^{con}_{x\cdot}$ is subjected to an effect or dummy coding constraint. This specification of Class-specific constants is much more parsimonious and is, in fact, equivalent to how x-y relationships with ordinal y's are modeled in LC Cluster models. Rather that estimating $K \cdot M$ intercept terms, one now estimates only $M+K-1$ coefficients; that is, one extra coefficient per extra latent class.

"Order" constraints are important if one has a priori knowledge about the sign of an effect. For example, the effect of price on persons' preferences is usually assumed to be negative – or better, non-positive – for each latent class (segment). If the price effect is specified to be "Descending", the resulting parameter estimate(s) will be constrained to be in agreement with this assumption.

The "No Effect" option makes it possible to specify a different regression equation for each latent class. More specifically, each latent class may have different sets of attributes and predictors affecting the choices. Post hoc constraints can be based on the reported z value for each of the coefficients. An example of an a priori use of this constraint is the inclusion of a random-responder class, a class in which all coefficient are zero. This is specified as follows:

	Class 1	Class 2	Class 3	Class 4	
Constants	–	2	3	4	
Attribute1	–	2	3	4	,
Attribute2	–	2	3	4	

where "–" indicates that the effect is equal to 0. In this example, Class 1 is the random-responder class.

"Merge Effects" is a much more flexible variant of "Class Independent". It can be used to equate the parameters for any set of latent classes. Besides post hoc constraints, very sophisticated a priori constraints can be imposed with this option. An important application is the specification of LC DFactor structures in which each latent class corresponds to the categories of two or more latent variables. For example, consider a set of constraints of the form:

	Class 1	Class 2	Class 3	Class 4
Constants	1	1	3	3
Attribute1	1	2	1	2
Attribute2	1	2	1	2

,

where the same numbers in a given row indicate that the associated class parameters are equal. This restricted 4-Class model is a 2-dimensional DFactor model: the categories of DFactor 1 differ with respect to the constants and the categories of DFactor 2 with respect to the two attribute effects. Specifically, level 1 of DFactor 1 is formed by Classes 1 and 2 and level 2 by Classes 3 and 4; level 1 of DFactor 2 is formed by Classes 1 and 3 and level 2 by Classes 2 and 4.

The option "Offset" can be used to specify any nonzero fixed-value constraint on the Class-specific effect of a numeric attribute. This means that it is possible to refine the definition of any Class (segment) by enhancing or reducing the estimated part-worth utility of any numeric attribute for that Class. Recall that numeric attribute p enters as $\beta_{xp}^{att}\, z_{itmp}^{att}$ in the linear part of the conditional logit model. Suppose, after estimating the model, the estimate for β_{xp}^{att} turned out to be 1.5 for Class 1. If z_{itmp}^{att} is specified to be an offset, the importance of this attribute to be reduced (1.5 would be reduced to 1) for this Class. But suppose that you wish to enhance the importance of this attribute for Class 1; say, you wish to restrict β_{xp}^{att} to be equal to 2. The trick is to recode the attribute, replacing each code by twice the value. Thus, the recoded attribute is defined as $2 \cdot z_{itmp}^{att}$. If we restrict the effect of this recoded attribute to 1, we obtain $1 \cdot 2 \cdot z_{itmp}^{att}$, which shows that the effect of z_{itmp}^{att} is equated to 2. Such recoding can be done easily within Latent GOLD Choice, using the Replace option.

In addition to post hoc refinements to customize the definition of the resulting latent classes, the offset restriction can also be used to make the Classes conform to various theoretical structures. Probably the most important a priori application of "Offset" is that of defining stayer- or brand-loyal classes. A brand-loyal class selects one of the brands with a probability equal to 1 and is not affected by the other attributes. An example of a restrictions table corresponding to such a structure is:

	Class 1	Class 2	Class 3	Class 4
Brand1(100)	$*$	−	−	−
Brand2(100)	−	$*$	−	−
Brand3(100)	−	−	$*$	−
Constants	−	−	−	4
Attribute1	−	−	−	4
Attribute2	−	−	−	4

.

Here, "−" means no effect and "$*$" means offset. As can be seen, Classes 1, 2, and 3 are only affected by an offset, and Class 4 – the brand-switching or mover class – is affected by the constants and the two attributes. The numeric "attributes" Brand1(100), Brand2(100), and Brand3(100) are brand "dummies" that take on the value 100 for the brand concerned and are 0 otherwise.[3] As a result of the fixed effect of 100, the probability of selecting the corresponding brand will be equal to one. To illustrate this, suppose that a choice set consists of three alternatives and that (only) alternative 1 is associated with brand 1. The probability that someone belonging to Class 1 selects alternative 1 equals $\exp(100)/[\exp(100) + \exp(0) + \exp(0)] = 1.0000$.[4] Although this model is similar to a zero-inflated model, the offset-based specification is much more flexible in the sense that the number of brand-loyal classes does not need to coincide with the number of alternatives per set. In the above example, the sets could consist of four instead of three alternatives, say three different brands and a "none" alternative.

Now, we will discusses several more advanced applications of the restriction options. The first is a model for ratings with a parsimonious specification of the Class dependence of the constants. Let "One" be an attribute with the constant value l. The model of interest is obtained with the restrictions

[3] It is not necessary to assume that the 100 appears in all alternatives for the brand concerned. The value could also be 100 if a particular condition is fulfilled – for example, if the price of the evaluated product is larger than a certain amount – and 0 otherwise. This shows that the offset option provides a much more flexible way of specifying classes with zero response probabilities than the zero-inflated option.

[4] While a value of 100 for the offset can be used to fix a probability to 1.0000, a value of -100 can be use to fix a probability to 0.0000. For example, $\exp(-100)/[\exp(-100) + \exp(0) + \exp(0)] = 0.0000$.

table

	Class 1	Class 2	Class 3	Class 4	
Constants	1	1	1	1	
One	–	2	3	4	.
Attribute1	1	2	3	4	
Attribute2	1	2	3	4	

Instead of having a separate set of constants for each latent class, the restricted constant for category m in Class x equals $\beta_{xm}^{con} = \beta_{m}^{con} + y_m^* \cdot \beta_{x1}^{att}$, where for identification, $\beta_{11}^{att} = 0$. Note that this is equivalent to using the "no-simple" setting for the constants and similar to the treatment of ordinal indicators in the LC Cluster and DFactor Modules of Latent GOLD.

Suppose you assume that the effect of price is negative (descending) for Classes 1-3 and unrestricted for Class 4. This can be accomplished by having two copies of the price variable in the model, say Price1 and Price2. The effect of Price1 is specified as ordered and is fixed to zero in Class 4. The effect of Price2 is fixed to zero in Classes 1-3.

Suppose your assumption is that the effect of a particular attribute is at least 2. This can be accomplished by combining a fixed value constraint with an order constraint. More precisely, an additional attribute defined as $2 \cdot z_{itmp}^{att}$ is specified to be an offset and the effect of the original attribute z_{itmp}^{att} defined to be ascending.

Our final example is an exploratory variant of the DFactor structure described above. Suppose you want a two-DFactor model without assumptions on which discrete factor influences which attribute effects. This can be accomplished having 3 copies of all attributes in the attributes file. With two attributes (brand and price), the restriction table is of the form

	Class 1	Class 2	Class 3	Class 4	
Brand1	1	1	1	1	
Brand2	–	–	3	3	
Brand3	–	2	–	2	.
Price1	1	1	1	1	
Price2	–	–	3	3	
Price3	–	2	–	2	

The first copy (Brand1 and Price1) defines a main effect for each attribute. The second copy (Brand2 and Price2) is used to define the first DFactor, a contrast between Classes 3/4 and 1/2. The third copy (Brand3 and Price3) specifies DFactor 2 by means of a contrast between Classes 2/4 and 1/3.

2.12 General Latent Class Choice Model

In the previous subsections, we described the various elements of the LC model implemented in Latent GOLD Choice. Now, we will combine all these elements and provide the structure of the general LC choice model. The general probability density function associated with case i is

$$\begin{aligned} P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\tau}_i) &= \sum_{x=1}^{K} \tau_{ix}\, P(x|\mathbf{z}_i^{cov}) P(\mathbf{y}_i|x, \mathbf{z}_i^{att}, \mathbf{z}_i^{pred}) \\ &= \sum_{x=1}^{K} \tau_{ix}\, P(x|\mathbf{z}_i^{cov}) \prod_{t=1}^{T_i} \left[P(y_{it}|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred}) \right]^{v_{it}}, \end{aligned} \tag{6}$$

where $P(x|\mathbf{z}_i^{cov})$ and $P(y_{it} = m|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pre})$ are parameterized by logit models; that is,

$$\begin{aligned} P(x|\mathbf{z}_i^{cov}) &= \frac{\exp(\eta_{x|\mathbf{z}_i})}{\sum_{x'=1}^{K} \exp(\eta_{x'|\mathbf{z}_i})} \\ P(y_{it} = m|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pre}, s_{it}) &= \frac{\exp(s_{it} \cdot \eta_{m|x,\mathbf{z}_{it}})}{\sum_{m' \in A_{it}} \exp(s_{it} \cdot \eta_{m'|x,\mathbf{z}_{it}})}. \end{aligned}$$

The linear model for $\eta_{x|\mathbf{z}_i}$ is

$$\eta_{x|\mathbf{z}_i} = \gamma_{0x} + \sum_{r=1}^{R} \gamma_{rx} z_{ir}^{cov}.$$

For first choices and rankings, $\eta_{m|x,\mathbf{z}_{it}}$ equals

$$\eta_{m|x,\mathbf{z}_{it}} = \beta_{xm}^{con} + \sum_{p=1}^{P} \beta_{xp}^{att}\, z_{itmp}^{att} + \sum_{q=1}^{Q} \beta_{xmq}^{pre}\, z_{itq}^{pre},$$

if $m \in A_{it}$ and $-\infty$ otherwise. For ratings,

$$\eta_{m|x,\mathbf{z}_{it}} = \beta_{xm}^{con} + y_m^* \cdot \left(\sum_{p=1}^{P} \beta_{xp}^{att}\, z_{itp}^{att} + \sum_{q=1}^{Q} \beta_{xq}^{pre}\, z_{itq}^{pre} \right).$$

3 Estimation and Other Technical Issues

3.1 Log-likelihood and Log-posterior Function

The parameters of the LC choice model are estimated by means of Maximum Likelihood (ML) or Posterior Mode (PM) methods. The likelihood function is derived from the probability density function defined in equation (6). Let $\boldsymbol{\vartheta}$ denote the vector containing the γ and β parameters. As before, $\mathbf{y}_i$ and $\mathbf{z}_i$ denote the vectors of dependent and explanatory variables for case i , and I denotes the total number of cases.

ML estimation involves finding the estimates for $\boldsymbol{\vartheta}$ that maximize the log-likelihood function

$$\log \mathcal{L} = \sum_{i=1}^{I} w_i \log P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\vartheta}).$$

Here, $P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\vartheta})$ is the probability density associated with case i given parameter values $\boldsymbol{\vartheta}$ and w_i is the *Case Weight* corresponding to case i.[5] This case weight w_i can be used to group identical response patterns or to specify (complex survey) sampling weights. In the former case, w_i will serve as a frequency count, and in the latter case, Latent GOLD Choice will provide pseudo ML estimates (Patterson, Dayton, and Graubard, 2002).[6] The other type of weight – *Replication Weight* v_{it} – that was introduced in the previous section modifies the definition of the relevant probability density $P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\vartheta})$. The exact form of $P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\vartheta})$ is described in equation (6).

In order to prevent boundary solutions or, equivalently, to circumvent the problem of non-existence of ML estimates, we implemented some ideas from Bayesian statistics in Latent GOLD Choice. The boundary problem that may occur is that the (multinomial) probabilities of the model for the latent classes or the model for the choices, rankings, or ratings may converge to zero. This occurs if a β or γ parameter becomes very extreme, tends to go to (minus) infinity. The boundary problem is circumvented by using Dirichlet priors for the latent and the response probabilities (Clogg et al., 1991; Galindo-Garre, Vermunt, and Bergsma, 2004; Gelman et. al., 1996; Schafer, 1997). These are so-called conjugate priors since they have same form as the corresponding multinomial probability densities. The implication of using priors is that the estimation method is no longer ML but PM (Posterior Mode).

Denoting the assumed priors for $\boldsymbol{\vartheta}$ by $p(\ \boldsymbol{\vartheta})$ and the posterior by $\mathcal{P}$, PM estimation involves finding the estimates for $\boldsymbol{\vartheta}$ that maximize the log-

[5] In order to simplify the discussion, in this section we discuss only on the situation without known-class indicators.

[6] In Latent GOLD Choice Advanced, there is a more elegant option for dealing with sampling weights, as well as with other complex survey sampling features.

posterior function

$$\begin{aligned}\log \mathcal{P} &= \log \mathcal{L} + \log p(\boldsymbol{\vartheta}) \\ &= \sum_{i=1}^{I} w_i \log P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\vartheta}) + \log p(\boldsymbol{\vartheta}),\end{aligned}$$

or, in other words, finding the point where $\frac{\partial \log \mathcal{P}}{\partial \boldsymbol{\vartheta}} = 0$. Algorithms that are used to solve this problem – EM and Newton-Raphson – are described below.

The user-defined parameters in the priors $p(\boldsymbol{\vartheta})$ can be chosen in such a way that $\log p(\boldsymbol{\vartheta}) = 0$, which makes PM estimation turn into ML estimation. PM estimation can also be seen as a form of penalized ML estimation, in which $p(\boldsymbol{\vartheta})$ serves as a function penalizing solutions that are too near to the boundary of the parameter space and, therefore, smoothing the estimates away from the boundary.

3.2 Missing Data

3.2.1 Dependent variable

If the value of the dependent variable is missing for one or more of the replications of case i, the replications concerned are omitted from the analysis. The remaining replications will, however, be used in the analysis. Thus, instead of using list-wise deletion of cases, Latent GOLD Choice provides ML or PM estimates based on all available information. The assumption that is made is that the missing data are missing at random (MAR) or, equivalently, that the missing data mechanism is ignorable (Little and Rubin, 1987; Schafer, 1997; Skrondal and Rabe-Hesketh, 2004; Vermunt, 1997).

In the case of missing data, it is important to clarify the interpretation of the chi-squared goodness-of-fit statistics. Although parameter estimation with missing data is based on the MAR assumption, the chi-squared statistics not only test whether the model of interest holds, but also the much more restrictive MCAR (missing completely at random) assumption (see Vermunt, 1997). Thus, caution should be used when interpreting the overall goodness-of-fit tests in situations involving missing data.

3.2.2 Attributes, predictors, and covariates

Missing values on attributes will never lead to exclusion of cases or replications from the analysis. If the technical option for including missing values on covariates and predictors is off, cases with missing covariate values and replications with missing predictor values are excluded from the analysis. When this technical option is on, such cases and replications are retained by imputing the missing values using the method described below.

Missing values on numeric predictors and covariates are replaced by the sample mean. This is the mean over all cases without a missing value for covariates and the mean of all replications without a missing value for predictors. Missing values on numeric attributes are not imputed with a mean, but with a 0, which implies that a missing value in the alternatives file is, in fact, equivalent to using a 0.

Missing values on nominal attributes, predictors, and covariates is dealt with via the design matrix. In fact, the effect is equated to zero for the missing value category. Recall the effect and dummy coding schemes illustrated in subsection 2.8 for the case of a nominal attribute with 4 categories. Suppose there is also a missing category. In the case of effects coding, the design matrix that is set up for the 3 non-redundant terms is then

$$\begin{array}{lrrr}
\text{category 1} & 1 & 0 & 0 \\
\text{category 2} & 0 & 1 & 0 \\
\text{category 3} & 0 & 0 & 1 \\
\text{category 4} & -1 & -1 & -1 \\
\text{missing} & 0 & 0 & 0
\end{array}.$$

As can be seen, the entries corresponding to the missing category are all equal to 0, which amounts to setting its coefficient equal to zero. Since in effect coding the unweighted mean of the coefficients equals zero, equating the effect of the missing value category to zero implies that it is equated to the unweighted average of the effects of the other four categories. This imputation method for nominal variables is therefore similar to mean imputation with numeric variables.

In the case of dummy coding with the first category as the reference category, the design matrix that is set up for the 3 non-redundant terms is

$$\begin{array}{lrrr}
\text{category 1} & 0 & 0 & 0 \\
\text{category 2} & 1 & 0 & 0 \\
\text{category 3} & 0 & 1 & 0 \\
\text{category 4} & 0 & 0 & 1 \\
\text{missing} & 1/4 & 1/4 & 1/4
\end{array}.$$

The number 1/4 (one divided by the number of categories of the nominal attribute concerned) implies that the parameter of the missing value category is equated to the unweighted mean of the parameters of the other four categories. Note that the coefficient for the reference category is fixed to 0. Also with “dummy last”, we would get a row with 1/4s for the missing value category.

3.3 Prior Distributions

The different types of priors have in common that their user-defined parameters (*Bayes Constants*) denoted by α can be interpreted as adding α observations – for instance, the program default of one – generated from a conservative null model (as is described below) to the data. All priors are defined in such a way that if the corresponding α's are set equal to zero, $\log\, p\,(\boldsymbol{\vartheta}) = 0$, in which case we will obtain ML estimates. We could label such priors as “non-informative”. Below we present the $\log\, p\,(\boldsymbol{\vartheta})$ terms for the various types of distributions without their normalizing constants. The symbols U^{cov} and $U^{att,pre}$ are used to denote the number of different covariate and attribute/predictor patterns. A particular pattern is referred to by the index u.

The Dirichlet prior for the latent probabilities equals

$$\log p\left[P(x|\mathbf{z}_u^{cov})\right] = \frac{\alpha_1}{K \cdot U^{cov}} \log P(x|\mathbf{z}_u^{cov}).$$

Here, K denotes the number of latent classes and α_1 the *Bayes Constant* to be specified by the user. As can be seen, the influence of the prior is equivalent to adding $\frac{\alpha_1}{K}$ cases to each latent class. These cases are distributed evenly over the various covariate patterns. This prior makes the sizes of the latent classes slightly more equal and the covariate effects somewhat smaller.

For the dependent variable, we use the following Dirichlet prior:

$$\log p\left[P(y=m|x,\mathbf{z}_u^{att},\mathbf{z}_u^{pred})\right] = \frac{\widehat{\pi}_m\,\alpha_2}{K \cdot U^{att,pred}} \log P(y|x,\mathbf{z}_u^{att},\mathbf{z}_u^{pred}),$$

where $\widehat{\pi}_m$ is the observed marginal distribution of the dependent variable y. This prior can be interpreted as adding $\frac{\alpha_2}{K}$ observations to each latent class with preservation of the observed distribution of y, where α_2 is a parameter to be specified by the user. The $\frac{\alpha_2}{K}$ observations are distributed evenly over the observed attribute/predictor patterns. This prior makes the class-specific response probabilities slightly more similar to each other and smooths the β parameters somewhat towards zero.

The influence of the priors on the final parameter estimates depends on the values chosen for the α's, as well as on the sample size. The default settings are $\alpha_1 = \alpha_2 = 1.0$. This means that with moderate sample sizes the influence of the priors on the parameter estimates is negligible. Setting $\alpha_1 = \alpha_2 = 0$ yields ML estimates.

3.4 Algorithms

To find the ML or PM estimates for the model parameters $\boldsymbol{\vartheta}$, Latent GOLD Choice uses both the EM and the Newton-Raphson algorithm. In practice, the estimation process starts with a number of EM iterations. When close enough to the final solution, the program switches to Newton-Raphson. This is a way to exploit the advantages of both algorithms; that is, the stability of EM when it is far away from the optimum and the speed of Newton-Raphson when it is close to the optimum.

The task to be performed for obtaining PM estimates for $\boldsymbol{\vartheta}$ is finding the parameter values for which

$$\frac{\partial \log \mathcal{P}}{\partial \boldsymbol{\vartheta}} = \frac{\partial \log \mathcal{L}}{\partial \boldsymbol{\vartheta}} + \frac{\partial \log p(\boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}} = 0. \tag{7}$$

Here,

$$\begin{aligned}
\frac{\partial \log \mathcal{L}}{\partial \boldsymbol{\vartheta}} &= \sum_{i=1}^{I} w_i \frac{\partial \log P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}} \\
&= \sum_{i=1}^{I} w_i \frac{\partial \log \sum_{x=1}^{K} P(x|\mathbf{z}_i^{cov}, \boldsymbol{\vartheta}) P(\mathbf{y}_i|x, \mathbf{z}_i^{att}, \mathbf{z}_i^{pred}, \boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}} \\
&= \sum_{i=1}^{I} \sum_{x=1}^{K} w_{xi} \frac{\partial \log P(x|\mathbf{z}_i^{cov}, \boldsymbol{\vartheta}) P(\mathbf{y}_i|x, \mathbf{z}_i^{att}, \mathbf{z}_i^{pred}, \boldsymbol{\vartheta})}{\partial \boldsymbol{\vartheta}},
\end{aligned} \tag{8}$$

where

$$w_{xi} = w_i \, P(x|\mathbf{z}_i, \mathbf{y}_i, \boldsymbol{\vartheta}) = w_i \, \frac{P(x|\mathbf{z}_i^{cov}, \boldsymbol{\vartheta}) P(\mathbf{y}_i|x, \mathbf{z}_i^{att}, \mathbf{z}_i^{pred}, \boldsymbol{\vartheta})}{P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\vartheta})}. \tag{9}$$

The *EM algorithm* is a general method for dealing with ML estimation with missing data (Dempster, Laird, and Rubin, 1977; McLachlan and Krishnan, 1997). This method exploits the fact that the first derivatives of the incomplete data log-likelihood ($\log \mathcal{L}$) equal the first derivatives of the complete data log-likelihood ($\log \mathcal{L}^c$). The complete data is the log-likelihood that we would have if we knew to which latent class each case belongs:

$$\begin{aligned}\log \mathcal{L}^c &= \sum_{i=1}^{I}\sum_{x=1}^{K} w_{xi} \log P(x|\mathbf{z}_i^{cov}, \boldsymbol{\vartheta}) P(\mathbf{y}_i|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred}\boldsymbol{,}\, \boldsymbol{\vartheta}) \\ &= \sum_{i=1}^{I}\sum_{x=1}^{K} w_{xi} \log P(x|\mathbf{z}_i^{cov},\ \boldsymbol{\vartheta}) \qquad (10) \\ &\quad + \sum_{i=1}^{I}\sum_{x=1}^{K} w_{xi} \sum_{t=1}^{T_i} v_{it} \log P(y_{it}|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred}, \boldsymbol{\vartheta}).\end{aligned}$$

Each νth cycle of the EM algorithm consist of two steps. In the Expectation (E) step, estimates $\widehat{w}_{xi}^{\nu}$ are obtained for w_{xi} via equation (9) filling in $\widehat{\boldsymbol{\vartheta}}^{\nu-1}$ as parameter values. The Maximization (M) step, involves finding new $\widehat{\boldsymbol{\vartheta}}^{\nu}$ improving $\log \mathcal{L}^c$. Note that, actually, we use PM rather than ML estimation, which means that in the M step we update the parameters in such a way that

$$\log \mathcal{P}^c = \log \mathcal{L}^c + \log p(\boldsymbol{\vartheta}) \qquad (11)$$

increases rather than (10). Sometimes closed-form solutions are available in the M step. In other cases, standard iterative methods can be used to improve the complete data log-posterior defined in equation (11). Latent GOLD Choice uses iterative proportional fitting (IPF) and unidimensional Newton in the M step (see Vermunt 1997, Appendices).

Besides the EM algorithm, we also use a *Newton-Raphson* (NR) method. [7] In this general optimization algorithm, the parameters are updated as follows:

$$\widehat{\boldsymbol{\vartheta}}^{\nu} = \widehat{\boldsymbol{\vartheta}}^{\nu-1} - \varepsilon\ \boldsymbol{H}^{-1}\mathbf{g}.$$

The gradient vector $\mathbf{g}$ contains the first-order derivatives of the log-posterior to all parameters evaluated at $\widehat{\boldsymbol{\vartheta}}^{\nu-1}$, $\mathbf{H}$ is the Hessian matrix containing the second-order derivatives to all parameters, and ε is a scalar denoting the step size. Element g_k of $\mathbf{g}$ equals

$$g_k = \sum_{i=1}^{I} w_i \frac{\partial \log P(\mathbf{y}_i|\mathbf{z}_i\boldsymbol{,}\, \boldsymbol{\vartheta})}{\partial \vartheta_k} + \frac{\partial \log p(\boldsymbol{\vartheta})}{\partial \vartheta_k},$$

[7] Haberman (1988) proposed estimating standard LC models by Newton Raphson.

and element $H_{kk'}$ of $\mathbf{H}$ equals

$$H_{kk'} = \sum_{i=1}^{I} w_i \frac{\partial^2 \log P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\vartheta})}{\partial\vartheta_k \partial\vartheta_{k'}} + \frac{\partial^2 \log p(\boldsymbol{\vartheta})}{\partial\vartheta_k \partial\vartheta_{k'}}.$$

Latent GOLD Choice computes these derivatives analytically. The step size ε $(0 < \varepsilon \leq 1)$ is needed to prevent decreases of the log-posterior to occur. More precisely, when a standard NR update $-\mathbf{H}^{-1}\mathbf{g}$ yields a decrease of the log-likelihood, the step size is reduced till this no longer occurs.

The matrix $-\mathbf{H}^{-1}$ evaluated at the final $\widehat{\boldsymbol{\vartheta}}$ yields the standard estimate for the asymptotic variance-covariance matrix of the model parameters: $\widehat{\Sigma}_{standard}(\boldsymbol{\vartheta}) = -\,\widehat{\mathbf{H}}^{-1}$.[8] Latent GOLD Choice also implements two alternative estimates for $\Sigma(\boldsymbol{\vartheta})$. The first alternative is based on the outer-product of the cases' contributions to the gradient vectors; that is, $\widehat{\Sigma}_{outer}(\boldsymbol{\vartheta}) = \widehat{\mathbf{B}}^{-1}$, where element $B_{kk'}$ of $\mathbf{B}$ is defined as

$$B_{kk'} = \frac{N}{N-1} \sum_{i=1}^{I} w_i \frac{\partial \log P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\vartheta})}{\partial\vartheta_k} \frac{\partial \log P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\vartheta})}{\partial\vartheta_{k'}}.$$

Note that $\mathbf{B}$ is the sample covariance matrix of the case-specific contributions to the elements of the gradient vector.

The third estimator for $\Sigma(\boldsymbol{\vartheta})$ is the so-called robust, sandwich, or Huber-White estimator, which is defined as

$$\widehat{\Sigma}_{robust}(\boldsymbol{\vartheta}) = \widehat{\mathbf{H}}^{-1}\, \widehat{\mathbf{B}}\ \ \widehat{\mathbf{H}}^{-1}.$$

The advantage of $\widehat{\Sigma}_{outer}(\boldsymbol{\vartheta})$ compared to the other two is that is much faster to compute because it uses only first derivatives. It may thus be an alternative for $\widehat{\Sigma}_{standard}(\boldsymbol{\vartheta})$ in large models. The advantage of the robust method is that contrary to the other two methods, it does not rely on the assumption that the model is correct.

Note that $\widehat{\Sigma}(\boldsymbol{\vartheta})$ can be used to obtain the standard error for any function $h(\widehat{\boldsymbol{\vartheta}})$ of $\widehat{\boldsymbol{\vartheta}}$ by the delta method:

$$\widehat{se}\left(h(\widehat{\boldsymbol{\vartheta}})\right) = \sqrt{\left(\frac{h(\widehat{\boldsymbol{\vartheta}})}{\partial\widehat{\boldsymbol{\vartheta}}}\right)' \widehat{\Sigma}(\boldsymbol{\vartheta}) \left(\frac{h(\widehat{\boldsymbol{\vartheta}})}{\partial\widehat{\boldsymbol{\vartheta}}}\right)}. \qquad (12)$$

[8] The matrix $-\mathbf{H}$ is usually referred to as the observed information matrix, which serves as an approximation of the expected information matrix.

Latent GOLD Choice uses the delta method, for example, to obtain standard errors of probabilities and redundant parameters.

Inequality restrictions – needed for ordered clusters, order-restricted predictor effects, and positive variances – are dealt with using an active-set variant of the Newton-Raphson method described above (Galindo-Garre, Vermunt, Croon, 2001; Gill, Murray, and Wright, 1981). For that purpose, the effects involved in the order constraints are reparameterized so that they can be imposed using simple nonnegativity constraints of the form $\vartheta \geq 0$. In an active-set method, the equality constraint associated with an inequality constraint becomes activate if it is violated (here, parameter is equated to 0 if it would otherwise become negative), but remains inactive if its update yields an admissible value (here, a positive update).

3.5 Convergence

The exact algorithm implemented in Latent GOLD Choice works as follows. The program starts with EM until either the maximum number of EM iterations (*Iteration Limits EM*) or the EM convergence criterion (*EM Tolerance*) is reached. Then, the program switches to NR iterations which stop when the maximum number of NR iterations (*Iteration Limits Newton-Raphson*) or the overall converge criterion (*Tolerance*) is reached. The convergence criterion that is used is

$$\sum_{u=1}^{npar} \left| \frac{\widehat{\vartheta}_u^{\nu} - \widehat{\vartheta}_u^{\nu-1}}{\widehat{\vartheta}_u^{\nu-1}} \right|,$$

which is the sum of the absolute relative changes in the parameters. The program also stops its iterations when the change in the log-posterior is negligible, i.e., smaller than 10^{-12}.

The program reports the iteration process in Iteration Detail. Thus, it can easily be checked whether the maximum number of iterations is reached without convergence. In addition, a warning is given if one of the elements of the gradient is larger than 10^{-3}.

It should be noted that sometimes it is more efficient to use only the EM algorithm, which is accomplished by setting *Iteration Limits Newton-Raphson* = 0 in the Technical Tab. This is, for instance, the case in models with many parameters. With very large models, one may also consider suppressing the computation of standard errors and Wald statistics, or to Pause the model estimation to examine preliminary output.

3.6 Start Values

Latent GOLD Choice generates random start values. So long as the technical option *Seed* equals 0 (the default option), these differ every time that a model is estimated because the seed of the random number generator is then obtained from the system time. The seed used by the program is reported in the output. A run can be replicated by specifying the reported best start seed as *Seed* in the Technical Tab and setting the number of *Random Sets* to zero.

Since the EM algorithm is extremely stable, the use of random starting values is generally good enough to obtain a converged solution. However, there is no guarantee that such a solution is also the global PM or ML solution. A well-known problem in LC analysis is the occurrence of local maxima which also satisfy the likelihood equations given in (7).

The best way to prevent ending up with a local solution is to use multiple sets of starting values which may yield solutions with different log-posterior values. In Latent GOLD Choice, the use of such multiple sets of random starting values is automated. The user can specify how many sets of starting values the program should use by changing the *Random Sets* option in the Technical Tab. Another relevant parameter is *Iterations* specifying the number of iterations to be performed per start set. More precisely, within each of the random sets, Latent GOLD Choice performs the specified number of EM iterations. Subsequently, within the best 10 percent in terms of log-posterior, the program performs an extra 2 times *Iterations* EM iterations. Finally, it continues with the best solution until convergence. It should be noted that while such a procedure increases considerably the probability of finding the global PM or ML solution, especially if both parameters are set large enough, there is no guarantee that it will be found in a single run.

When a model contains two or more latent classes or one or more DFactors, the starting values procedure will generate the specified number of starting sets and perform the specified number of iterations per set. In one-class models in which local maxima may occur – for example, in models with continuous factors (see Advanced option) – both the specified number of starting sets and iterations per set are reduced by a factor of three. In one-class models in which local maxima cannot occur, the number of starting sets is automatically equated to 1.

With the option *Tolerance*, one can specify the EM convergence criterion to be used within the random start values procedure. Thus, start values iterations stop if either this tolerance or the maximum number of iterations is reached.

3.7 Bootstrapping the P Value of L^2 or -2LL Difference

Rather than relying on the asymptotic p value, it also possible to estimate the p value associated with the L^2 statistic by means of a parametric bootstrap. This option is especially useful with sparse tables (Langeheine, Pannekoek, and Van de Pol, 1996) and with models containing order restrictions (Galindo and Vermunt, 2005; Vermunt, 1999, 2001). The model of interest is then not only estimated for the sample under investigation, but also for B replication samples. These are generated from the probability distribution defined by the ML estimates. The estimated bootstrap p value, $\widehat{p}_{boot}$, is defined as the proportion of bootstrap samples with a larger L^2 than the original sample. The standard error of $\widehat{p}_{boot}$ equals $\sqrt{\frac{\widehat{p}_{boot}(1-\widehat{p}_{boot})}{B}}$. The precision of $\widehat{p}_{boot}$ can be increased by increasing the number of replications B. The number of replications is specified by the parameter *Replications*.

A similar procedure is used to obtain a bootstrap estimate of the p value corresponding to the difference in log-likelihood value between two nested models, such as two models with different numbers of latent classes. The -2LL-difference statistic is defined as $-2 \cdot (LL_{H_0} - LL_{H_1})$, where H_0 refers to the more restricted hypothesized model (say a K–class model) and H_1 to the more general model (say a model with $K+1$ classes). Replication samples are generated from the probability distribution defined by the ML estimates under H_0. The estimated bootstrap p value, $\widehat{p}_{boot}$, is defined as the proportion of bootstrap samples with a larger -2LL-difference value than the original sample.

The bootstrap of the -2LL-difference statistic comparing models with different numbers of latent classes was used by McLachlan and Peel (2000) in the context of mixture of normals. Vermunt (2001) used bootstrap p values for both the L^2 and the -2LL-difference statistic in the context of order-restricted latent class models, where the L^2 measured the goodness-of-fit for an ordinal latent class model and the -2LL difference concerned the difference between an order-restricted and an unrestricted latent class model.

The other parameter is *Seed*, which can be used to replicate a bootstrap. The seed used by the bootstrap to generate the data sets is reported in the output.

Two technical details about the implementation of the bootstrap should be mentioned. For each bootstrap replication, the maximum likelihood estimates serve as start values. Thus, no random sets are used for the replications. To gain efficiency in term of computation time, the iterations within a bootstrap replication terminate when the replicated L^2 is smaller (-2LL-diff value is larger) than the original one, even if the convergence criterion or the maximum number of iterations is not reached.

3.8 Identification Issues

Sometimes LC models are not identified; that is, it may not be possible to obtain unique estimates for some parameters. Non-identification implies that different parameter estimates yield the same log-posterior or log-likelihood value. When a model is not identified, the observed information matrix, $-\mathbf{H}$, is not full rank, which is reported by the program. Another method to check whether a model is identified is to run the model again with different starting values. Certain model parameters are not identified if two sets of starting values yield the same $\log \mathcal{P}$ or $\log \mathcal{L}$ values with different parameter estimates.

With respect to possible non-identification, it should be noted that the use of priors may make models identified that would otherwise not be identified. In such situations, the prior information is just enough to uniquely determine the parameter values.

A related problem is "weak identification", which means that even though the parameters are uniquely determined, sometimes the data is not informative enough to obtain stable parameter estimates. Weak identification can be detected from the occurrence of large asymptotic standard errors. Local solutions may also result from weak identification.

Other "identification issues" are related to the order of the Classes and the uniqueness of parameters for nominal variables. For unrestricted Choice models, the Classes are reordered according to their sizes: the first Class is always the largest Class. Parameters (γ's and β's) involving nominal variables are identified by using either effect or dummy coding, which means that parameters sum to zero over the relevant indices or that parameters corresponding to the first or last category are fixed to zero. Note that the Parameters output also contains the redundant γ and β parameters, and in the case of effect coding also their standard errors.

3.9 Selecting and Holding out Choices or Cases

The replication and case weights can be used to omit certain choices or cases (records with a common case ID) from the analysis. With a weight equal to zero, one can remove a choice/case from the analysis, and no output is provided for this choice/case. Alternatively, a very small weight (1.0e-100) can be used to exclude choices/cases for parameter estimation, while retaining the relevant prediction and classification output.

3.9.1 Replication and case weights equal to zero

Setting case weights equal to zero will eliminate the corresponding cases from the analysis. This feature can be used to select a subset of cases for the analysis. For example, by specifying a variable with the value 1 for males and 0 for females as a case weight, one will perform an analysis for males only. This "zero case weight" option makes it straightforward to perform separate analyses for different subgroups that are in the same data file. It should be noted that no output is provided for the cases with zero weights.

Similarly, with a replication weight equal to zero, one removes the corresponding replication from the analysis. This option can, therefore, be used to select choices to be used for parameter estimation; for example, one may wish to select the first and last choice from a full ranking for a maximum-difference analysis.

3.9.2 Replication weights equal to a very small number

An important feature of the Latent GOLD Choice program is that it allows specifying hold-out choices. These are choices that are not used for parameter estimation, but for which one obtains prediction information. Hold-out choices are defined by means of replications weights equal to a very small number; i.e., 1.0e-100. These replications will be excluded when estimating the specified model. Their predicted values, however, may be written to the output file. This "very small replication weight" option can be used for validation purposes; that is, to determine the prediction performance of the estimated model for hold-out choices.

3.9.3 Case weights equal to a very small number

In some situations, one may desire removing certain cases from the analysis, but nevertheless obtaining classification and prediction output for all cases. This can be accomplished by using case weights equal to a very small number – i.e., 1.0e-100 – for the cases that should not be used for parameter estimation. The program treats such a weight as if it were a zero, which means that results are not influenced by the presence of these cases and that computation time is comparable to the analysis of a data set without these cases. An important difference with the "zero case weight" option is that this "very small case weight" option yields classification and prediction information for the cases concerned.

One possible application is the analysis of very large data sets. With this option one can use a subset of cases for parameter estimation, but obtain class membership information for all cases. Another application is predicting class membership for new cases based on parameter values obtained with another sample. By appending the new cases to the original data file and giving them a weight equal to 1.0e-100, one obtains the relevant output for these cases after restoring and re-estimating the original model.

4 The Latent Gold Choice Output

Below, we provide technical details on the quantities presented in the various Latent GOLD Choice output sections (Model Summary, Parameters, Importance, Profile, ProbMeans, Set Profile, Set Probmeans, Iteration Detail, Frequencies, Standard Classification, and Covariate Classification), as well as on the output that can be written to files (Standard Classification, Covariate Classification, Predicted values, Individual Coefficients, Cook's D, and Variance-Covariance Matrix).

4.1 Model Summary

This first part of the output section reports the number of cases ($N = \sum_{i=1}^{I} w_i$), the total number of replications ($N_{rep} = \sum_{i=1}^{I} w_i \sum_{t=1}^{T_i} v_{it}$), the number of estimated parameters ($npar$), the number of activated constraints (in models with order restrictions), the seed used by the pseudo random number generator, the seed of the best start set, and the seed used by the bootstrap procedure.

The last part (Variable Detail) contains information on the variables that are used in the analysis. The other four parts - Chi-squared Statistics, Log-likelihood Statistics, Classification Statistics, Covariate Classification Statistics, and Prediction Statistics - are described in more detail below.

4.1.1 Chi-squared statistics

The program reports chi-squared and related statistics, except when the data file contains replication weights other than 0 or 1. The three reported chi-squared measures are the likelihood-ratio chi-squared statistic L^2, the Pearson chi-squared statistic X^2, and the Cressie-Read chi-squared statistic CR^2. Before giving the definitions of the chi-squared statistics, we need to explain two types of groupings that have to be performed with the original cases.

The first is the grouping of identical cases; that is, cases that have the same covariate, known-class, predictor, and attribute values, and give the same responses. This yields I^* unique data patterns with observed frequency counts denoted by n_{i^*}, where i^* denotes a particular data pattern. These frequency counts are obtained by summing the case weights w_i of the cases with data pattern i^*; that is, $n_{i^*} = \sum_{i \in i^*} w_i$.[9] In order to obtain the chi-squared statistics, we also need to group cases with identical covariate, known-class, predictor, and attribute values, which amounts to grouping cases without taking into account their responses.[10] This yields the sample sizes N_u for the U relevant multinomials, where u denotes a particular multinomial or "covariate" pattern. These sample sizes are obtained by $N_u = \sum_{i \in u} w_i$ or $N_u = \sum_{i^* \in u} n_{i^*}$.[11] Note that $N = \sum_{u=1}^{U} N_u$.

Let $\widehat{m}_{i^*}$ denote the estimated cell count for data pattern i^*, which is obtained by:

$$\widehat{m}_{i^*} = N_{u_{i^*}} \widehat{P}(\mathbf{y}_{i^*} | \mathbf{z}_{i^*}), \tag{13}$$

i.e., by the product of the total number of cases with the same "covariate" pattern as data pattern i^* ($N_{u_{i^*}}$) and the estimated multinomial probability

[9] With the somewhat loose but rather simple notation $i \in i^*$ we mean "all the cases with data pattern i^*".

[10] With missing values on some replications, also the missing data pattern is used as a grouping criterion. That is, cases belonging to the same "covariate" pattern should also have observed values on the same set of replications or.

[11] With $i \in u$ we mean "all the cases with covariate pattern u", and with $i^* \in u$ "all the data patterns with covariate pattern u".

corresponding to data pattern i^*.[12]

Using these definitions of $\widehat{m}_{i^*}$, n_{i^*}, and N, the chi-squared statistics are calculated as follows:[13]

$$
\begin{aligned}
L^2 &= 2\sum_{i^*=1}^{I^*} n_{i^*} \log \frac{n_{i^*}}{\widehat{m}_{i^*}}, \\
X^2 &= \sum_{i^*=1}^{I^*} \frac{(n_{i^*})^2}{\widehat{m}_{i^*}} - N, \\
CR^2 &= 1.8\sum_{i^*=1}^{I^*} n_{i^*} \left[\left(\frac{n_{i^*}}{\widehat{m}_{i^*}}\right)^{2/3} - 1\right].
\end{aligned}
$$

The number of degrees of freedom is defined by

$$
df = \min\left\{\sum_{u=1}^{U}\left(\prod_{t=1}^{T_u^*} M_{ut}^* - 1\right), N\right\} - npar.
$$

Here, T_u^* is the total number of replications in "covariate" pattern u, and M_{ut}^* denotes the number of alternatives of the tth observed replication corresponding to "covariate" pattern u. The term $\min\{\cdot\}$ indicates that df is based on the sample size N when the number of independent cells in the hypothetical frequency table is larger than the sample size. The chi-squared values with the corresponding df yield the asymptotic p -values, which can be used to determine whether the specified model fits the data.

If the *Bootstrap* L^2 option is used, the program also provides the estimated bootstrap p-value corresponding to the L^2 statistic, as well as its standard error. This option is especially useful with sparse tables, in which case the asymptotic p-values cannot be trusted. Note that sparseness almost always is a problem in LC choice models. The best indication of sparseness is when df is (much) larger than the total sample size N.

The program reports the Bayesian Information Criterion (BIC), the Akaike Information Criterion (AIC), Akaike Information Criterion 3 ($AIC3$), and the Consistent Akaike Information Criterion ($CAIC$) based on the L^2 and df, which is the more common formulation in the analysis of frequency

[12] In order to get meaningful chi-squared statistics, in models with a known-class indicator we, in addition, divide by $\sum_{x=1}^{K} \tau_{i^*x} P(x|\mathbf{z}_{i^*})$.

[13] Note that we are using a somewhat unconventional formula for X^2. The reason for this is that the sum $\sum_{i^*=1}^{I^*}$ is over the nonzero observed cells only.

tables. They are defined as

$$\begin{aligned}
BIC_{L^2} &= L^2 - \log(N)\, df, \\
AIC_{L^2} &= L^2 - 2\ df, \\
AIC3_{L^2} &= L^2 - 3\ df, \\
CAIC_{L^2} &= L^2 - [\log(N) + 1]\ \ df.
\end{aligned}$$

These information criteria weight the fit and the parsimony of a model: the lower BIC, AIC, $AIC3$, or $CAIC$, the better the model.

Use of information criteria based on L^2 or $\log \mathcal{L}$ (see below) should yield the same result. The differences between BIC, AIC, $AIC3$, and $CAIC$ values across models are the same with both methods. However, with extremely large df, the L^2 based information measures may become more highly negative than the maximum precision can indicate, which makes their rounded values meaningless. In such cases, one has to use the (equivalent) $\log \mathcal{L}$ based measures.

The last statistic that is provided in the chi-squared statistics section is the Dissimilarity Index (DI), which is a descriptive measure that is defined as follows:

$$DI = \frac{\left\{ \left(\sum_{i^*=1}^{I^*} |n_{i^*} - \widehat{m}_{i^*}| \right) + \left(N - \sum_{i^*=1}^{I^*} \widehat{m}_{i^*} \right) \right\}}{2\,N}.$$

It should be noted that the term $(N - \sum_{i^*=1}^{I^*} \widehat{m}_{i^*})$ captures the contribution of the zero observed cells to DI. This term is added to the formula because $\sum_{i^*=1}^{I^*} |n_{i^*} - \widehat{m}_{i^*}|$ is a sum over the non-zero observed cell counts only. DI is a descriptive measure indicating how much observed and estimated cell frequencies differ from one another. It indicates which proportion of the sample should be moved to another cell to get a perfect fit.

4.1.2 Log-likelihood statistics

The program also reports the values of the log-likelihood ($\log \mathcal{L}$), the log-prior ($\log p(\boldsymbol{\vartheta})$), and log-posterior ($\log \mathcal{P}$) . Recall that

$$\begin{aligned}
\log \mathcal{L} &= \sum_{i=1}^{I} w_i \log \widehat{P}(\mathbf{y}_i|\mathbf{z}_i), \\
\log \mathcal{P} &= \log \mathcal{L} + \log p(\widehat{\boldsymbol{\vartheta}}).
\end{aligned}$$

In addition, the Bayesian Information Criterion (BIC), the Akaike Information Criterion (AIC), the Akaike Information Criterion 3 ($AIC3$),[14] and the Consistent Akaike Information Criterion ($CAIC$) based on the log-likelihood are reported. These are defined as

$$\begin{aligned}
BIC_{\log\mathcal{L}} &= -2\log\mathcal{L} + (\log N)\ npar, \\
AIC_{\log\mathcal{L}} &= -2\log\mathcal{L} + 2\ npar, \\
AIC3_{\log\mathcal{L}} &= -2\log\mathcal{L} + 3\ npar, \\
CAIC_{\log\mathcal{L}} &= -2\log\mathcal{L} + [(\log N)\ + 1]\ npar.
\end{aligned}$$

If the *Bootstrap -2LL diff* option is used, the program also provides the estimated bootstrap p-value (and the standard error) for the -2LL difference test between a restricted and an unrestricted model.

4.1.3 Classification statistics

This set of statistics contains information on how well the observed y and z values predict the latent class, or, in other words, how well the latent classes are separated. Classification is based on the latent classification or posterior class membership probabilities. For response pattern i, these are calculated as follows:

$$\widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i) = \frac{\widehat{P}(x|\mathbf{z}_i^{cov})\widehat{P}(\mathbf{y}_i|x, \mathbf{z}_i^{att}\ , \mathbf{z}_i^{pred})}{\widehat{P}(\mathbf{y}_i|\mathbf{z}_i)}. \tag{14}$$

These quantities are used to compute the estimated proportion of classifications errors (E), as well as three R^2-type measures for nominal variables: the proportional reduction of classification errors $R^2_{x,errors}$, a measure based on entropy labelled $R^2_{x,entropy}$, and a measure based on qualitative variance labelled $R^2_{x,variance}$. The latter is similar to the Goodman and Kruskal tau-b association coefficient for nominal dependent variables (Magidson, 1981).

The proportion of classification errors is defined as:

$$E = \frac{\sum_{i=1}^{I} w_i \left[1 - \max \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)\right]}{N}.$$

[14] New results by Andrews and Currim (2003) and Dias (2004) suggest that AIC3 is a better criterion than BIC and AIC in determining the number of latent classes in choice models.

Each of the three R^2-type measures is based on the same type of reduction of error structure; namely,

$$R_x^2 = \frac{\text{Error}(x) - \text{Error}(x|\mathbf{z}, \mathbf{y})}{\text{Error}(x)}, \tag{15}$$

where Error(x) is the total error when predicting x without using information on $\mathbf{z}$ and $\mathbf{y}$, and Error$(x|\mathbf{z}, \mathbf{y})$ is the prediction error if we use all observed information from the cases. Error$(x|\mathbf{z}, \mathbf{y})$ is defined as the (weighted) average of the case-specific errors Error$(x|\mathbf{z}_i, \mathbf{y}_i)$,

$$\text{Error}(x|\mathbf{z}, \mathbf{y}) = \frac{\sum_{i=1}^{I} w_i \text{Error}(x|\mathbf{z}_i, \mathbf{y}_i)}{N}.$$

The three R^2 measures differ in the definition of Error$(x|\mathbf{z}_i, \mathbf{y}_i)$. In $R^2_{x,errors}$, it equals $1 - \max \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)$, in $R^2_{x,entropy}$, $\sum_{x=1}^{K} - \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i) \log \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)$, and in $R^2_{x,variance}$, $1 - \sum_{x=1}^{K} [\widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)]^2$. In the computation of the total error Error(x), the $\widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)$ are replaced by the estimated marginal latent probabilities $\widehat{P}(x)$, which are defined as

$$\widehat{P}(x) = \frac{\sum_{i=1}^{I} w_i \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)}{N} = \frac{\sum_{i=1}^{I} \widehat{w}_{xi}}{N}. \tag{16}$$

The Average Weight of Evidence (AWE) criterion adds a third dimension to the information criteria described above. It weights fit, parsimony, and the performance of the classification (Banfield and Raftery, 1993). This measure uses the so-called classification log-likelihood, which is equivalent to the complete data log-likelihood $\log \mathcal{L}^c$, i.e.,

$$\log \mathcal{L}^c = \sum_{i=1}^{I} \sum_{x=1}^{K} \widehat{w}_{xi} \log \widehat{P}(x|\mathbf{z}_i^{cov}) \widehat{P}(\mathbf{y}_i|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred}).$$

AWE can now be defined as

$$AWE = -2 \log \mathcal{L}^c + 2 \left(\frac{3}{2} + \log N \right) npar.$$

The lower AWE, the better a model.

The Classification Table cross-tabulates modal and probabilistic class assignments. More precisely, the entry (x, x') contains the sum of the class x

posterior membership probabilities for the cases allocated to modal class x'. Hence, the diagonal elements ($x = x'$) are the numbers of correct classifications per class and the off-diagonal elements ($x \neq x'$) the corresponding numbers of misclassifications. From the classification table, one can not only see how many cases are misclassified (as indicated by the proportion of classification errors E), but also detect which are the most common types of misclassifications. If a particular entry (x, x') with $x \neq x'$ is large, this means that classes x and x' are not well separated.

The marginals of the Classification Table provides the distribution of cases across classes under modal (column totals) and probabilistic (row totals) classification. Except for very rare situations, these marginal distributions will *not* be equal to one another. This illustrates the phenomenon that modal class assignments do not reproduce the estimated latent class distribution. Whereas the row totals are in agreement with the estimated classes sizes,[15] the column totals provide the latent class distribution that is obtained when writing the class assignments to a file using the Latent GOLD Choice output-to-file option.

4.1.4 Covariate classification statistics

These statistics indicate how well one can predict class membership from an individual's covariate values, and are therefore only of interest if the estimated model contains active covariates. The measures are similar to the ones that are reported in the section "Classification Statistics"; that is, the estimated proportion of classification errors, the proportional reduction of classification errors, an entropy-based R^2 measure, and a qualitative variance-based R^2 measure. The difference is that now the predictions (and computations) are based on the model probabilities $\widehat{P}(x|\mathbf{z}_i)$ instead of the posterior probabilities $\widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)$. Whereas the total error can still be denoted as Error(x), the model prediction error in equation (15) should now be denoted as Error$(x|\mathbf{z})$ instead of Error$(x|\mathbf{z}, \mathbf{y})$.

4.1.5 Prediction statistics

Prediction statistics indicate how well the observed choices, rankings, or ratings are predicted by the specified model. For rankings, the prediction

[15] There may be a very small difference, which is caused by the Bayes constant for the latent classes.

statistics are based on first choices only. For choice and rating variables, all replications are used for obtaining the prediction measures.

The predicted values used in the computation of the prediction statistics are based on the estimated individual-specific response probabilities, which are denoted by $\widehat{P}_{m|it}$. For ratings, we also make use of the estimated expected values $\widehat{y}_{it} = \sum_{m=1}^{M} y_m^* \widehat{P}_{m|it}$, where y_m^* is the score for response category m. As is shown in detail below, $\widehat{P}_{m|it}$ is computed by weighting Class-specific estimates by the posterior membership probabilities $\widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)$. This means that our procedure can be called *posterior-mean/mode or expected/modal a posteriori prediction* .

The individual-specific response probabilities $\widehat{P}_{m|it}$ can be obtained as follows:

$$\widehat{P}_{m|it} = \sum_{x=1}^{K} \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i) \widehat{P}(y_{it} = m|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred}). \tag{17}$$

As can be seen, these are weighted averages of the Class-specific response probabilities, where the posterior class-membership probabilities serve as weights.

There are two other prediction methods – HB-like and marginal mean prediction. In the first, one obtains $\widehat{P}_{m|it}$ with the individual-specific utilities $\widehat{\eta}_{m|it}$,

$$\widehat{P}_{m|it} = \frac{\exp(\widehat{\eta}_{m|it})}{\sum_{m'=1}^{M} \exp(\widehat{\eta}_{m'|it})} \tag{18}$$

The $\widehat{V}_{itm}$ are weighted averages of the Class-specific utilities defined in equation (2), where the posterior class-membership probabilities serve as weights; that is,

$$\widehat{\eta}_{m|it} = \sum_{x=1}^{K} \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)\, \widehat{\eta}_{m|x,\mathbf{z}_{it}}.$$

Because of the similarity with prediction in Hierarchical Bayes (HB) procedures, we call this alternative method *HB-like prediction*. Note that the way we compute $\widehat{\eta}_{m|it}$ is equivalent to computing $\widehat{\eta}_{m|it}$ with the individual-specific $\widehat{\beta}_{ip}$ parameters defined in equation (20).

Marginal mean (mode) prediction differs from posterior mean prediction in that the prior class membership probabilities $\widehat{P}(x|\mathbf{z}_i)$ are used in the formula for $\widehat{P}_{m|it}$ given in equation (17) instead of the posterior membership probabilities $\widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)$. Whereas posterior mean and HB-like prediction provide a good indication of the within-sample prediction performance, marginal mean prediction gives a good indication of the out–of-sample prediction performance.

The most natural predicted value for a categorical dependent variable is the mode; that is, the m with the largest $\widehat{P}_{m|it}$. The *Prediction Table* cross-classifies observed and predicted values based on such a modal assignment. For ratings, which are ordinal dependent variables, we make use of the mean $(\widehat{y}_{it})$ in some of the error measures. Error measures may also be based on the estimated probabilities instead of a single predicted value.

The error measures reported in prediction statistics are obtained as follows:

$$\text{Error} = \frac{\sum_{i=1}^{I} w_i \sum_{t=1}^{T_i} v_{it}\, \text{Error}_{it}}{\sum_{i=1}^{I} w_i \sum_{t=1}^{T_i} v_{it}} \tag{19}$$

As can be seen, "Error" is a weighted average of the replication-specific errors "Error_{it}". Latent GOLD Choice uses four types of error measures (*Squared Error, Absolute Error, Minus Log-likelihood, and Prediction Error)*, which differ in the definition of Error_{it}. For ratings, the Error_{it} for *Squared Error* and *Absolute Error* equal $(y_{it} - \widehat{y}_{it})^2$ and $|y_{it} - \widehat{y}_{it}|$, respectively. For choices and ranking, these errors equal $\sum_{m=1}^{M}[I_m(y_{it}) - \widehat{P}_{m|it}]^2$ and $\sum_{m=1}^{M} |I_m(y_{it}) - \widehat{P}_{m|it}|$, where indicator variable $I_m(y_{it})$ equals 1 if $y_{it} = m$ and 0 otherwise. The Error_{it} for *Minus Log-likelihood* equals $\sum_{m=1}^{M} - I_m(y_{it}) \ln \widehat{P}_{m|it}$. In the computation of *Prediction Error,* Error_{it} equals 0 if the modal prediction is correct and 1 otherwise.

The general definition of the (pseudo) R^2 of an estimated model is the reduction of errors compared to the errors of a baseline model. More precisely,

$$R_y^2 = \frac{\text{Error(baseline)} - \text{Error(model)}}{\text{Error(baseline)}}.$$

Latent GOLD Choice uses two different baseline models, called *Baseline* and *Baseline(0)*, yielding two R^2 measures, called R_y^2 and $R_y^2(0)$. In *Baseline,* the Error is computed with response probabilities equal to the average $\widehat{P}_{m|it}$,

$$\widehat{P}_m = \frac{\sum_{i=1}^{I} w_i \sum_{t=1}^{T_i} v_{it}\, \widehat{P}_{m|it}}{\sum_{i=1}^{I} w_i \sum_{t=1}^{T_i} v_{it}},$$

In models with an unrestricted set of constants, $\widehat{P}_m$ equals the observed distribution of y. In that case, *Baseline* can be interpreted as the constants only model. The response probabilities under *Baseline(0)* are $\widehat{P}_m(0) = 1/M$, which means that *Baseline(0)* is the equal-probability model.

4.2 Parameters

The first part of the Parameters output contains *Class-specific* and *overall* R^2_y and $R^2_y(0)$ values based on *Squared Error.* The overall measures are the same as the ones appearing in *Prediction Statistics*. The logic behind the computation of the Class-specific $R^2_{y|x}$ measures is the same as for the overall measures (see description of *Prediction Statistics*). The Class-specific errors are obtained by

$$\text{Error}_{y|x} = \frac{\sum_{i=1}^{I} w_i \sum_{t=1}^{T_i} v_{it}\, \text{Error}_{xit}}{\sum_{i=1}^{I} w_i \sum_{t=1}^{T_i} v_{it}},$$

with $\widehat{w}_{xi} = w_i\ \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)$, as in equation (9). The definition of Error_{xit} is based on the Class-specific response probabilities $\widehat{P}(y_{it} = m|x, \mathbf{z}^{att}_{it}, \mathbf{z}^{pred}_{it})$ or, shortly, $\widehat{P}_{m|xit}$. For ratings, the predicted value equals $\widehat{y}_{xit} = \sum_{m=1}^{M} y^*_m P_{m|xit}$ and the corresponding error is $\text{Error}_{xit} = (y_{it} - \widehat{y}_{xit})^2$. For choice and ranking variables, Error_{xit} equals $\sum_{m=1}^{M} [I_m(y_{it}) - \widehat{P}_{m|xit}]^2$. Similar to the overall R^2_y measures, the *Baseline* error is based on the average $\widehat{P}_{m|xit}$ and *Baseline(0)* on $1/M$.

In the second part of the Parameters output, the program reports the estimates obtained for the β and γ parameters appearing in the linear predictors η, the estimates for error variances and covariances σ, as well as the corresponding estimated asymptotic standard errors, $\widehat{se}(\beta)$, $\widehat{se}(\gamma)$, and $\widehat{se}(\sigma)$. These standard errors are the squared roots of the diagonal elements of the estimated variance-covariance matrix $\widehat{\Sigma}(\boldsymbol{\vartheta})$. As described earlier, one of three methods can be used to obtain $\widehat{\Sigma}(\boldsymbol{\vartheta})$, yielding either a standard, outer-product based, or robust standard errors and Wald statistics.

The significance of sets of parameters can be tested by means of the reported Wald statistic labeled *Wald.* We also report a Wald statistic labeled *Wald(=)*, which tests whether regression coefficients are equal between Classes (Class Independent). The general formula for a Wald statistic (W^2) is

$$W^2 = \left(\mathbf{C}'\boldsymbol{\vartheta}\right)' \left(\mathbf{C}'\,\widehat{\Sigma}(\boldsymbol{\vartheta})\mathbf{C}\right)^{-1} \left(\mathbf{C}'\boldsymbol{\vartheta}\right),$$

where the tested set of linear constraints is: $\mathbf{C}'\boldsymbol{\vartheta} = \mathbf{0}$. The Wald test is a chi-squared test. Its number of degrees of freedom equals the number of constraints. Computation of standard errors and Wald statistics can be suppressed, which may be useful in models with many parameters.

The Parameters output also contains the means and standard deviations of the conditional logit coefficients (last two columns in model for choices/rankings/ratings). These are the typical fixed and random effects in multilevel, mixed, or random-coefficient logit models. Let $\widehat{\beta}_{xp}$ denote the estimated value of one of the conditional logit parameters, which can be a constant, an attribute effect, or a predictor effect. Using basic statistics calculus, the *Mean* of $\widehat{\beta}_{xp}$ can be defined as $\sum_{x=1}^{K} \widehat{P}(x) \widehat{\beta}_{xp}$ and the *Std.Dev.* of $\widehat{\beta}_{xp}$ as $\sqrt{\sum_{x=1}^{K} \widehat{P}(x)(\widehat{\beta}_{xp})^2 - \left[\sum_{x=1}^{K} \widehat{P}(x) \widehat{\beta}_{xp}\right]^2}$.

4.3 Importance

The Importance output reports the maximum effect for each of the attributes, including the constants, as well as re-scaled maximum effects that add up to one within latent classes.

Let a denote a level of attribute p, A_p its total number of levels, and $\widehat{\eta}_{a|xp}$ the utility associated with level a for latent class x. For numeric attributes, $\widehat{\eta}_{a|xp}$ equals the attribute effect times the numeric score of category a $(z_{ap}\beta_{xp}^{att})$; for nominal attributes, it is simply the effect for category a (β_{xap}^{att}). The maximum effect of attribute p for latent class x is defined as

$$\text{maxeff}_{xp} = \max(\widehat{\eta}_{a|xp}) - \min(\widehat{\eta}_{a|xp}).$$

These maximum effects can be compared both across attributes and across latent classes.

Often it is relevant to compare the relative importances of the attributes across latent classes. These relative importances or relative maximum effects are obtained as follows:

$$\text{releff}_{xp} = \frac{\text{maxeff}_{xp}}{\sum_p \text{maxeff}_{xp}}.$$

As can be seen, releff_{xp} is a maximum effect that is re-scaled to sum to 1 across attributes within a latent class. The relative importances are depicted in a plot. Attributes can be deleted from the Importance output using the plot control. The relative effects are then rescaled to sum to one for the remaining attributes. This feature can, for example, be useful if one is interested in relative effects without considering the constants or the effect corresponding to a none option.

4.4 Profile and ProbMeans

The content of the Profile and ProbMeans output will be explained together because these two output sections are strongly related. Both sections contain 1) marginal latent probabilities, 2) transformed Class-specific attribute effects, and 3) information on the relation between (active and inactive) covariates and class membership.

The first row of each output section contains the estimated marginal latent class probabilities $\widehat{P}(x)$ (see equation 16). In Profile these are called *Class Size* and in ProbMeans *Overall Probability.*

The Profile output contains transformed attributes effects (β parameters), including the constants. As above, let a denote a level of the attribute p, A_p its total number of levels, and $\widehat{\eta}_{a|xp}$ the utility associated with level a for latent class x. The reported "choice probabilities" for attribute p are obtained as follows:

$$\widehat{P}_p(a|x) = \frac{\exp(\widehat{\eta}_{a|xp})}{\sum_{a'=1}^{A} \exp(\widehat{\eta}_{a'|xp})}.$$

The $\widehat{P}_p(a|x)$ can be interpreted as the estimated choice probabilities in a set of A_p alternatives that differ only with respect to the attribute concerned. For numeric attributes, we also report the means $\sum_{a=1}^{A_p} z_{ap}\widehat{P}_p(a|x)$.

In ProbMeans, the choice probabilities $\widehat{P}_p(a|x)$ are re-scaled to sum to one over latent classes. That is,

$$\widehat{P}_p(x|a) = \frac{\widehat{P}(x)\widehat{P}_p(a|x)}{\sum_{x'=1}^{K} \widehat{P}(x')\widehat{P}_p(a|x')}.$$

This number can be interpreted as the probability of being in latent class x given choice a on "set" p.

The third part of the Profile and ProbMeans output sections provides information for covariates. This is information obtained by aggregating and re-scaling posterior membership probabilities (Magidson and Vermunt, 2001). Let b denote a particular level of covariate r,and B_r the number of categories of the covariate concerned, and let the frequency count $\widehat{n}_r(x,b)$ be defined as follows:

$$\widehat{n}_r(x,b) = \sum_{i:z_{ir}=b} w_i \, \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i),$$

where $i : z_{ir} = b$ denotes that the sum is over the cases with value b on the covariate concerned. In Profile, we report the probability of being in covariate level b given that one belongs to latent class x,

$$\widehat{P}_r(b|x) = \frac{\widehat{n}_r(x,b)}{\sum_{b'=1}^{B_r}\widehat{n}_r(x,b')},$$

and for numeric covariates also the means $\sum_{b=1}^{B} z_{br}\widehat{P}_r(b|x)$, where z_{br} is the score of covariate category b. ProbMeans contains the probability of being in latent class x given covariate level b:

$$\widehat{P}_r(x|b) = \frac{\widehat{n}_r(x,b)}{\sum_{x'=1}^{K}\widehat{n}_r(x',b)}$$

For nominal attributes/covariates, the Profile plot depicts the choice probabilities $\widehat{P}_p(a|x)$ and covariate probabilities $\widehat{P}_r(b|x)$. For numeric attributes and covariates, the Profile plot contains 0-1 means, which are means that are re-scaled to be in the 0-1 interval. In ProbMeans, the quantities $\widehat{P}_p(x|a)$ and $\widehat{P}_r(b|x)$ are plotted in Uni- and Tri-plots (Magidson and Vermunt, 2001; Vermunt and Magidson 2000). Similar plots have been proposed by Van der Ark and Van der Heijden (1998) and Van der Heijden, Gilula, and Van der Ark (1999) for standard LC and latent budget models.

A nice feature of the Profile and ProbMeans output is that it describes the relationships between the latent variable and all variables selected as attributes or covariates. This means that even if a certain covariate effect is fixed to zero, one still obtains its ProbMeans information. This feature is exploited in the "inactive covariates method". Advantages of working with inactive instead of active covariates are that the estimation time is not increased and that the obtained solution is the same as without covariates.

4.5 Set Profile and Set ProbMeans

The Set Profile and Set ProbMeans output sections contain information on the estimated choice probabilities per choice set. For rankings, these are based on the first choice replications only. For choices and ratings, all replications are used.

Let ℓ denote a particular choice set number as indicated by the *Set ID* variable. The Class-specific and the overall choice probabilities for *Set* ℓ are obtain as follows:

$$\widehat{P}_\ell(m|x) \quad = \quad \frac{\sum_{i=1}^{I}\widehat{w}_{xi}\sum_{t\in\ell}v_{it}\widehat{P}(y_{it}=m|x,\mathbf{z}_{it}^{att},\mathbf{z}_{it}^{pred})}{\sum_{i=1}^{I}\widehat{w}_{xi}\sum_{t\in\ell}v_{it}},$$

$$\widehat{P}_\ell(m) \quad = \quad \frac{\sum_{i=1}^{I} w_i \sum_{t\in\ell} v_{it}\widehat{P}_{m|it}}{\sum_{i=1}^{I} w_i \sum_{t\in\ell} v_{it}}.$$

Here, $\widehat{w}_{xi}$ is the case weight times the posterior membership probability (see equation 9), and $\widehat{P}_{m|it}$ is the individual-specific choice probability which, depending on the type of prediction, is defined by equation (17) or (18). The computation of the *Set Average* is the same, except that the summations are over all t instead of $t \in \ell$.

Set Profile also contains information on the observed choice probabilities $p_\ell(m)$, as well as residuals per alternative and per set that compare observed with overall estimated choice probabilities. The standardized residual (*StdResid*) for alternative m of set ℓ is obtained as follows:

$$\frac{p_\ell(m) - \widehat{P}_\ell(m)}{\sqrt{\widehat{P}_\ell(m)}}\sqrt{N_\ell},$$

where $N_\ell = \sum_i w_i \sum_{t\in\ell} v_{it}$. The univariate residual (*UniResid*) for set ℓ is defined as:

$$\frac{\sum_{m=1}^{M_\ell} \frac{\left[p_\ell(m)-\widehat{P}_\ell(m)\right]^2}{\widehat{P}_\ell(m)} N_\ell}{M_\ell - 1.}.$$

Note that this is just a Pearson chi-squared divided by the number of "degrees of freedom", or the number of possible alternatives in set ℓ minus 1.

The Set Probmeans is obtained by re-scaling the $\widehat{P}_\ell(m|x)$; that is,

$$\widehat{P}_\ell(x|m) = \frac{\widehat{P}(x)\widehat{P}_\ell(m|x)}{\sum_{m=1}^{M_\ell} \widehat{P}(x)\widehat{P}_\ell(m|x)}.$$

These quantities, which can be plotted with the Probmeans in the Uni-plot and Tri-plot, indicate the probability of being in latent class x given that alternative m was selected in set ℓ.

The file in which the choice sets are defined may contain choice sets that are not presented to respondents. For such "simulation sets", the Set Profile output reports $\widehat{P}(y = m|x, \mathbf{z}_\ell^{att}, \overline{\mathbf{z}}^{pred})$, the estimated Class-specific choice probabilities given their attribute values and the mean of the predictor values.[16] The overall choice probabilities for simulation sets are weighted averages of the Class-specific choice probabilities, where $P(x)$ serves as weight.

[16] Note that the predictor values are missing for simulation sets.

4.6 Frequencies / Residuals

Latent GOLD Choice reports estimated and observed cell frequencies ($\widehat{m}_{i^*}$ and n_{i^*}), as well as standardized residuals ($\widehat{r}_{i^*}$). The computation of estimated cell entries was described in equation (13). The standardized residuals are defined as

$$\widehat{r}_{i^*} = \frac{\widehat{m}_{i^*} - n_{i^*}}{\sqrt{\widehat{m}_{i^*}}}.$$

Note that $(\widehat{r}_{i^*})^2$ is cell i^*'s contribution to the X^2 statistic.

This output section also contains a column *Cook's D (Cook's Distance).* This measure can be used to detect influential cases or, more precisely, cases having a larger influence on the parameter estimates than others. The exact formula that is used in Latent GOLD Choice 4.0 is given in equation (21). A typical cut-point for Cook's D is four times the number of parameters divided by the number of cases (Skrondal and Rabe-Hesketh, 2004). Note that the reported value in a particular row corresponds to the Cook's D for each of the cases with that data pattern.

4.7 Classification Information

The Classification output section contains the classification information for each data pattern i^*. We report the posterior class membership probabilities $\widehat{P}(x|\mathbf{z}_{i^*}, \mathbf{y}_{i^*})$, as well as the modal Class (the latent class with largest probability). This method of class assignment is sometimes referred to as posterior mode, empirical Bayes modal (EBM), or modal a posteriori (MAP) estimation (Skrondal and Rabe-Hesketh, 2004).

Classification can also be based on covariates only. This involves using the model probabilities $\widehat{P}(x|\mathbf{z}_u)$ – sometimes referred to as prior probabilities – as classification probabilities for each covariate pattern u. The same modal classification rule can be applied as with the posterior class membership probabilities.

4.8 Output-to-file Options

Five types of items can be written to output files: classification, classification based on covariates, predicted values, individual-specific coefficients, and the estimated variance-covariance matrix of the model parameters.

With *Standard Classification* and *Covariate Classification*, the output file will contain the posterior class-membership probabilities $\widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)$ and the model probabilities $\widehat{P}(x|\mathbf{z}_i)$, respectively, as well as the modal Class assignment based on these probabilities. With the option *Predicted Values* to a file, one obtains the estimated individual-specific choice probabilities $\widehat{P}_{m|it}$ which, depending on the type of prediction, are defined by equation (17) or (18), as well as the predicted value, which is a mode with choices and rankings and a mean with ratings. In addition, a CHAID (.chd) input file can be created for further profiling of the latent classes. (see Section 4.9).

With *Individual Coefficients*, one obtains the estimated individual-specific regression coefficients. Let $\widehat{\beta}_{xp}$ denote the estimated value of one of the conditional logit parameters, which can be a constant, an attribute effect, or a predictor effect. The posterior-mean or expected a posteriori estimate of a particular regression coefficient for case i is defined as follows:

$$\widehat{\beta}_{ip} = \sum_{x=1}^{K} \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)\, \widehat{\beta}_{xp} \tag{20}$$

that is, as a weighted average of the Class-specific coefficients. These estimates are similar to the individual coefficients obtained in multilevel, mixed, random-effects, or hierarchical Bayes (HB) models. The person-specific coefficients can be used to predict person i's new choices.The person-specific coefficients can be used to predict person i's new responses. The posterior standard deviations are defined as

$$\widehat{\sigma}_{\widehat{\beta}_{iq}} = \sqrt{\sum_{x=1}^{K} \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i)\, \left(\widehat{\beta}_{xq} - \widehat{\beta}_{iq}\right)^2}$$

Another output-to-file item is *Cook's D (Cook's Distance)*. It can be used to detect influential cases or, more precisely, cases with a large influence on the parameter estimates. The formula that is used the following:

$$C_i = -2\, \mathbf{g}_i'\, \mathbf{H}^{-1}\, \mathbf{g}_i, \tag{21}$$

where $\mathbf{H}$ is the Hessian matrix and $\mathbf{g}_i$ the vector with the gradient contributions of case i. A typical cut-point for Cook's D is four times the number of parameters divided by the number of cases (Skrondal and Rabe-Hesketh, 2004).

The last output-to-file item is the *Variance-Covariance Matrix* of the model parameters. Dependent of the type of variance estimator that is requested this will be $\widehat{\boldsymbol{\Sigma}}_{standard}(\boldsymbol{\vartheta})$, $\widehat{\boldsymbol{\Sigma}}_{outer}(\boldsymbol{\vartheta})$, or $\widehat{\boldsymbol{\Sigma}}_{robust}(\boldsymbol{\vartheta})$. Note that also the variances and covariances involving the omitted categories of the effect coded nominal variables are reported.

4.9 The CHAID Output Option

(This option requires the SI-CHAID 4.0 program)

The CHAID (CHi-squared Automatic Interaction Detector) analysis option can be used to assess the statistical significance of each Covariate in its relationship to the latent classes, as well as to develop detailed profiles of these classes, based on the relationships in 3- and higher-way tables. For example, in tutorial #6A, a CHAID analysis is used to explore the relationship between an individual's banking usage during some period (number of checks written, ATM usage, average balance) and the latent classes obtained in tutorial #6. If this option is selected, at the conclusion of the Latent GOLD Choice run, a CHAID (.chd) file is created which can be used as input to the SI-CHAID 4.0 program.

Part II: Advanced Model Options, Technical Settings, and Output Sections

5 Introduction to Part II (Advanced Models)

This part of the manual describes the three Advanced options of Latent GOLD Choice 4.0. These are:

1. An option for specifying models with continuous latent variables, which are referred to as continuous factors (CFactors).

2. A multilevel extension of the LC Choice model, which is a model containing group-level continuous latent variables (GCFactors) and/or a group-level nominal latent variable (GClasses).

3. An option to deal with the sampling design, which yields correct statistical tests for complex survey sampling designs that deviate from simple random sampling.

The *Continuous Factors* (CFactors) option makes it possible to specify random-coefficients conditional logit models. One may, however, also combine CFactors and latent classes in a single model, yielding LC Choice models in which the alternative-specific constants, predictor effects, and/or attribute effects may vary within latent classes.

The *Multilevel Model* option can be used to define LC Choice models for nested data, such as employees nested within firms, pupils nested within schools, clients nested within stores, patients nested within hospitals, citizens nested within regions, and repeated measurements nested within individuals. Note that a LC Choice model is itself a model for two-level data; that is, a model for multiple responses per case. The multilevel LC Choice model is thus, in fact, a model for three-level data; that is, for multiple responses nested within cases and cases nested within groups. As in any multilevel analysis, the basic idea of a multilevel LC Choice analysis is that one or more parameters of the model of interest is allowed to vary across groups using a random-effects modeling approach. In Latent GOLD Choice, the group-level random effects can either be specified to be continuous (group-level continuous factors: GCFactors) or discrete (group-level latent classes: GClasses), yielding either a parametric or a nonparametric approach, respectively.

One variant of the multilevel LC model involves including group-level random effects in the model for the latent classes, which is a way to take into account that groups differ with respect to the distribution of their members across latent classes (Vermunt, 2003, 2005; Vermunt and Magidson, 2005). Not only the intercept, but also the covariate effects may have a random part. Another variant involves including GCFactors and GClasses in the model for the choices. By combining group-level with case-level latent classes, one obtains a three-level conditional logit model with nonparametric random effects, and by combining group-level continuous factors with case-level continuous factors one obtains a standard three-level random-coefficients conditional logit model. The latter is a special case of the three-level generalized linear model (Vermunt, 2002c, 2004).

The *Survey* option makes it possible to get correct statistical tests with stratified and clustered samples, as well as with sampling weights and samples from finite populations. The design-corrected variance-covariance matrix of the model parameters is obtained by the well-known linearization estimator. Sampling weights can also be dealt with using a two-step procedure that involves estimating the model without sampling weights, and subsequently correcting the latent class distribution and covariate effects using the sampling weights.

The next three sections describe the three Advanced options in more detail. Attention is paid to model components, estimation issues, and application types. The last section discusses the output obtained with the Latent GOLD Choice Advanced options.

6 Continuous Factors

6.1 Model Components and Estimation Issues

Let F_{di} denote the score of case i on continuous latent variable, factor, or random effect number d. The total number of CFactors is denoted by D – thus, $1 \leq d \leq D$ – and the full vector of CFactor scores by $\mathbf{F}_i$. The maximum number of CFactors that can be included in a Latent GOLD Choice model is three, thus $0 \leq D \leq 3$.

Recall that without CFactors the most general Latent GOLD Choice structure for $P(\mathbf{y}_i|\mathbf{z}_i)$ equals

$$P(\mathbf{y}_i|\mathbf{z}_i) = \sum_{x=1}^{K} P(x|\mathbf{z}_i)P(\mathbf{y}_i|x, \mathbf{z}_i)$$

where

$$P(\mathbf{y}_i|x, \mathbf{z}_i) = \prod_{t=1}^{T_i} P(y_{it}|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred}).$$

If we include CFactors in a model, the assumed structure for $P(\mathbf{y}_i|\mathbf{z}_i)$ becomes

$$P(\mathbf{y}_i|\mathbf{z}_i) = \sum_{x=1}^{K} \int_{\mathbf{F}_i} f(\mathbf{F}_i)\, P(x|\mathbf{z}_i)P(\mathbf{y}_i|x, \mathbf{z}_i, \mathbf{F}_i)\, \mathrm{d}\mathbf{F}_i, \tag{22}$$

where

$$P(\mathbf{y}_i|x, \mathbf{z}_i, \mathbf{F}_i) = \prod_{t=1}^{T_i} P(y_{it}|x, \mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred}, \mathbf{F}_i)$$

The F_{di} are assumed to be standard normally distributed and mutually independent. In other words, $f(\mathbf{F}_i) = N(\mathbf{0}, \mathbf{I})$, where $\mathbf{I}$ is the identity matrix. As will be shown below, this specification is much less restrictive than one may initially think.

It is also possible to define models – standard random-effects conditional logit models – containing CFactors but no latent classes x. That simplifies the structure for $P(\mathbf{y}_i|\mathbf{z}_i)$ to

$$P(\mathbf{y}_i|\mathbf{z}_i) = \int_{\mathbf{F}_i} f(\mathbf{F}_i)\, P(\mathbf{y}_i|\mathbf{z}_i, \mathbf{F}_i)\, \mathrm{d}\mathbf{F}_i$$

with

$$P(\mathbf{y}_i|\mathbf{z}_i, \mathbf{F}_i) = \prod_{t=1}^{T_i} P(y_{it}|\mathbf{z}_{it}^{att}, \mathbf{z}_{it}^{pred}, \mathbf{F}_i)$$

Equation (22) shows that the F_{di} may appear in the model for the choices, but not in the model for the latent classes.[17] Compared to models without CFactors, the linear predictor in the model for the choices is the expanded with the following additional term:

$$\sum_{d=1}^{D} \lambda_{xmd}^{con} \cdot F_{di} + \sum_{d=1}^{D}\sum_{p=1}^{P} \lambda_{xpd}^{att} \cdot F_{di} \cdot z_{mitp}^{att} + \sum_{d=1}^{D}\sum_{q=1}^{Q} \lambda_{xmqd}^{pred} \cdot F_{di} \cdot z_{itq}^{pred}$$

[17] There is a trick for including CFactor effects in the model for the latent classes using the multilevel option.

In the first term, the F_{di} define random effects for the alternative-specific constants, and the $F_{di} \cdot z^{att}_{mitp}$ and $F_{di} \cdot z^{pred}_{itq}$ product terms define random coefficients for the attributes and predictors. An important difference with the more standard specification of random-effects models is that here each F_{di} can serve as a random effect for each of the model effects, which, as will be shown below, yields parsimonious random-effects covariance structures. Another important difference is that the size of the parameters associated with the random effects may differ across latent classes.

Model restrictions One can use the parameter constraints "Class Independent", "No Effect", and "Merge Effects", which imply equal λ's among all Classes, zero λ's in selected Classes, and equal λ's in selected Classes, respectively.

ML (PM) estimation and technical settings The main complication in the ML (PM) estimation of models with CFactors is that we have to deal with the multidimensional integral appearing in the definition of the marginal density $P(\mathbf{y}_i|\mathbf{z}_i)$ (see equation 22). Because a closed form expression for this integral is not available, it must be solved using approximation methods. Latent GOLD Choice approximates the conditional density $P(\mathbf{y}_i|\mathbf{z}_i)$ by means of Gauss-Hermite numerical integration, implying that the multidimensional integral is replaced by multiple sums (Bock and Aitkin, 1981). With three CFactors and B quadrature nodes per dimension, the approximate density equals

$$P(\mathbf{y}_i|\mathbf{z}_i) \approx \sum_{x=1}^{K} \sum_{b_1=1}^{B} \sum_{b_2=1}^{B} \sum_{b_3=1}^{B} P(x|\mathbf{z}_i)\, P(\mathbf{y}_i|x, \mathbf{z}_i, F_{b_1}, F_{b_2}, F_{b_3})\, P_{b_1}\, P_{b_2}\, P_{b_3}.$$

Here, F_{b_d} is the location and P_{b_d} the weight corresponding to quadrature node b_d for CFactor d. These nodes and weights are obtained from published quadrature tables (Stroud and Secrest, 1966). As can be seen, because of the multiple sums, this approximate density is very similar to the density of a LC model with multiple latent variables. The above approximation also shows that – given the fact that one will usually use at least 10 quadrature points per dimension (Lessafre and Spiessens, 2001) – because of computation burden, it does not make sense to have models with more than three CFactors.

Similar to what Latent GOLD Choice does for standard LC Choice models, the ML (PM) estimation problem for models with CFactors is solved using a combination of EM and Newton-Raphson with analytic first- and second-order derivatives.

The only new technical setting in models with CFactors is the parameter specifying the number of quadrature nodes to be used in the numerical integration. The default value is 10, the minimum 2, and the maximum 50.

6.2 Application Types

6.2.1 Random-effects conditional logit models

An important application of the CFactor option involves random-effects discrete-choice modeling (McFadden and Train, 2000; Skrondal and Rabe-Hesketh, 2004).[18] Let us first look at the *random intercept* case in a model for first choices containing $M-1$ alternative-specific constants and P attributes.[19] Such a model has the following form:

$$\eta_{m|\mathbf{z}_{it},F_{1i}} = \beta^{con}_{m} + \sum_{p=1}^{P} \beta^{att}_{p}\, z^{att}_{itmp} + \lambda^{con}_{m1} \cdot F_{1i} \tag{23}$$

Note that a single CFactor is used to capture the variation in each of the $M-1$ constants, a specification that is also used in the random-effects multinomial logistic regression model proposed by Hedeker (2003). The random part of the alternative-specific constant corresponding to category m is denoted as Ψ^{con}_{m}. Its variance equals $\sigma^2_{\Psi^{con}_m} = (\lambda^{con}_{m1})^2$ and the covariance between Ψ^{con}_{m} and $\Psi^{con}_{m'}$ equals $\sigma_{\Psi^{con}_m,\Psi^{con}_{m'}} = \lambda^{con}_{m1} \cdot \lambda^{con}_{m'1}$.

The model can be expanded to include *random slopes* or *random coefficients* in addition to random intercept terms. However, a slight complication is that one has to decide whether the various random effects should be uncorrelated or not. For uncorrelated random effects, expanding the model of equation (23) with a random slope for the first attribute yields

$$\eta_{m|\mathbf{z}_{it},\mathbf{F}_{i}} = \beta^{con}_{m} + \sum_{p=1}^{P} \beta^{att}_{p}\, z^{att}_{itmp} + \lambda^{con}_{m1} \cdot F_{1i} + \lambda^{att}_{12} \cdot F_{2i} \cdot z^{att}_{mit1}.$$

The variances of the random intercept terms and for the random slope of z^{att}_{mit1} equal $\sigma^2_{\Psi^{con}_m} = (\lambda^{con}_{xm1})^2$ and $\sigma^2_{\Psi^{att}_1} = (\lambda^{att}_{12})^2$, respectively.

[18] Random-effects models are also referred to as multilevel, hierarchical, mixed-effects, mixed, and random-coefficients models.

[19] Note that in Choice the intercept terms are referred to as constants.

The same model, but now with correlated random effects can be defined as follows:

$$\eta_{m|\mathbf{z}_{it},\mathbf{F}_i} = \beta^{con}_m + \sum_{p=1}^{P} \beta^{att}_p \, z^{att}_{itmp} + \lambda^{con}_{m1} \cdot F_{1i} + \lambda^{att}_{11} \cdot F_{1i} \cdot z^{att}_{mit1} + \lambda^{att}_{12} \cdot F_{2i} \cdot z^{att}_{mit1},$$

As can be seen, here F_{1i} does not only affect the constants but also the effect of z^{att}_{mit1}. The variance-covariance matrix of the random effects ($\mathbf{\Sigma_\Psi}$), can be obtained by $\mathbf{\Sigma_\Psi} = \mathbf{\Lambda}\,\mathbf{\Lambda}'$, where $\mathbf{\Lambda}$ is a matrix collecting the λ parameters. More specifically, in our example, $\sigma^2_{\Psi^{con}_m} = (\lambda^{con}_{m1})^2$, $\sigma^2_{\Psi^{att}_1} = (\lambda^{att}_{11})^2 + (\lambda^{att}_{12})^2$, $\sigma_{\Psi^{con}_m,\Psi^{con}_{m'}} = \lambda^{con}_{m1} \cdot \lambda^{con}_{m'1}$, and $\sigma_{\Psi^{con}_m,\Psi^{att}_1} = \lambda^{con}_{m1} \cdot \lambda^{att}_{11}$.

Whereas the random-effects models presented thus far contained as many CFactors as random terms, this is not necessary in general. In fact, with three CFactors – the Latent GOLD Choice maximum – one can define models with any number of random effects. This is accomplished with the following "factor-analytic" specification:

$$\eta_{m|\mathbf{z}_{it},\mathbf{F}_i} = \beta^{con}_m + \sum_{p=1}^{P} \beta^{att}_p \, z^{att}_{itmp} + \sum_{d=1}^{D} \lambda^{con}_{xmd} \cdot F_{di} + \sum_{d=1}^{D} \sum_{p=1}^{P} \lambda^{att}_{xpd} \cdot F_{di} \cdot z^{att}_{mitp}. \quad (24)$$

where again $\mathbf{\Sigma_\Psi} = \mathbf{\Lambda}\,\mathbf{\Lambda}'$. This "factor-analytic" specification in which each CFactor may be associated with multiple random effects is equivalent to the generalized random coefficient (GRC) formulation proposed by Skrondal and Rabe-Hesketh (2004, p. 101). In fact, it is assumed that the unobserved heterogeneity in the regression coefficients can be summarized by at most three underlying CFactors.

6.2.2 LC (FM) regression models with random effects

A unique feature of Latent GOLD Choice is that it allows you to combine random effects with latent classes. More specifically, it is possible to specify LC Choice models in which the intercept and/or some of the regression coefficients vary within latent classes. Lenk and DeSarbo (2000) proposed using random effects in FMs of generalized linear models and Böckenholt (2001) proposed using random effects in LC models for ranking data.

It has been observed that the solution of a LC Choice analysis may be strongly affected by heterogeneity in the constants. In *choice-based conjoint studies*, for example, it is almost always the case that respondents differ with respect to their brand preferences, irrespective of the attributes of the offered products. A LC Choice model captures this brand heterogeneity phenomenon via Classes with different constants. However, the analyst often likes to find relatively small number of latent classes that differ in more meaningful ways with respect to attribute effects on the choices. By including random alternative-specific constants (intercepts) in the LC Choice model, for example,

$$\eta_{m|x,\mathbf{z}_{it},F_{1i}} = \beta_{xm}^{con} + \sum_{p=1}^{P} \beta_{xp}^{att} \, z_{mitp}^{att} + \lambda_{xm}^{con} \cdot F_{1i},$$

it is much more likely that one will succeed in finding such meaningful Classes (segments). The random intercept term, which may have a different effect in each latent class, will filter out (most of) the "artificial" variation in the constants.

7 Multilevel LC Choice Model

7.1 Model Components and Estimation Issues

To explain the multilevel LC model implemented in Latent GOLD Choice, we need to introduce and some new terminology. Higher-level observations will be referred to as groups and lower-level observations as cases. The records of cases belonging to the same group are connected by the *Group ID* variable. It should be noted that higher-level observations can also be individuals, for example, in longitudinal applications. "Cases" would then be the multiple time points within individuals and replications the multiple choices of an individual at the various time points.

The index j is used to refer to a particular group and I_j to denote the number of cases in group j. With y_{jit} we denote the response at replication t of case i belonging to group j, with $\mathbf{y}_{ji}$ the full vector of responses of case i in group j, and with $\mathbf{y}_j$ the responses of all cases in group j. Rather than expanding the notation with new symbols, group-level quantities will be referred to using a superscript g: Group-level classes (GClasses), group-level continuous factors (GCFactors), and group-level covariates (GCovariates) are denoted by x^g, $\mathbf{F}_j^g$, and $\mathbf{z}_j^g$, and group-level parameters by γ^g, β^g, and λ^g.

The most general probability structure for a multilevel LC Choice model is

$$P(\mathbf{y}_j|\mathbf{z}_j, \mathbf{z}_j^g) = \sum_{x^g=1}^{K^g} \int_{\mathbf{F}_j^g} f(\mathbf{F}_j^g)\, P(x^g|\mathbf{z}_j^g)\, P(\mathbf{y}_j|\mathbf{z}_j, x^g, \mathbf{F}_j^g)\, \mathrm{d}\mathbf{F}_j^g, \qquad (25)$$

where

$$P(\mathbf{y}_j|\mathbf{z}_j, x^g, \mathbf{F}_j^g) = \prod_{i=1}^{I_j} P(\mathbf{y}_{ji}|\mathbf{z}_{ji}, x^g, \mathbf{F}_j^g).$$

Assuming that the model of interest may also contain CFactors, for each case i, $P(\mathbf{y}_{ji}|\mathbf{z}_{ji}, x^g, \mathbf{F}_j^g)$ has a structure similar to the one described in equation (22); that is,

$$P(\mathbf{y}_{ji}|\mathbf{z}_{ji}, x^g, \mathbf{F}_j^g) = \sum_{x=1}^{K} \int_{\mathbf{F}_{ji}} f(\mathbf{F}_{ji})\, P(x|\mathbf{z}_{ji}, x^g, \mathbf{F}_j^g)\, P(\mathbf{y}_{ji}|x, \mathbf{z}_{ji}, \mathbf{F}_{ji}, x^g, \mathbf{F}_j^g)\, \mathrm{d}\mathbf{F}_{ji}.$$

where

$$P(\mathbf{y}_{ji}|x, \mathbf{z}_{ji}, \mathbf{F}_{ji}, x^g, \mathbf{F}_j^g) = \prod_{t=1}^{T_i} P(\mathbf{y}_{jih}|x, \mathbf{z}_{jit}^{att}, \mathbf{z}_{jit}^{pred}, \mathbf{F}_{ji}, x^g, \mathbf{F}_j^g)$$

These four equations show that a multilevel LC Choice model is a model

- for $P(\mathbf{y}_j|\mathbf{z}_j, \mathbf{z}_j^g)$, which is the marginal density of all responses in group j given all exogenous variable information in group j,
- containing GClasses (x^g) and/or (at most three mutually independent) GCFactors ($\mathbf{F}_j^g$),
- containing GCovariates $\mathbf{z}_j^g$ affecting the group classes x^g,
- assuming that the I_j observations for the cases belonging to group j are independent of one another given the GClasses and GCFactors,
- allowing the GClasses and GCFactors to affect the case-level latent classes x and/or the responses $\mathbf{y}_{ji}$.

GCFactors enter in exactly the same manner in the linear term of the conditional logit model as case-level CFactors. We refer to their coefficients as $\lambda_{mxd}^{con,g}$, $\lambda_{xpd}^{att,g}$, and $\lambda_{mxqd}^{pred,g}$. GCFactors can also be used in the model for the Classes. We will denote a GCFactor effect on the latent classes as $\lambda_{xrd}^{0,g}$, $0 \leq 1 \leq R$, where the superscript 0 refers to the model for the latent classes.

GClasses enter in the conditional logit model for the choices as $\beta^{con,g}_{xm,x^g} + \sum_{p=1}^{P} \beta^{att,g}_{xp,x^g} \cdot z^{att}_{mjitp} + \sum_{q=1}^{Q} \beta^{pred,g}_{xmq,x^g} \cdot z^{pred}_{jitq}$. Inclusion of GClasses in the model for the Classes implies that the γ parameters become GClass dependent; that is $\eta_{x|\mathbf{z}_{ji},x^g} = \gamma_{x^g,x0} + \sum_{r=1}^{R} \gamma_{x^g,xr} \cdot z^{cov}_{jir}$. Note that this is similar to a LC Regression analysis, where x^g now plays the role of x, and x the role of a nominal y variable.

The remaining linear predictor is the one appearing in the multinomial logistic regression model for the GClasses. It has the form $\eta_{x^g|\mathbf{z}^g_i} = \gamma^g_{x^g,0} + \sum_{r=1}^{R^g} \gamma^g_{x^g,r} \cdot z^{g,cov}_{jr}$. This linear predictor is similar to the one for the Classes (in a standard LC model), showing that GCovariates may be allowed to affect GClasses in the same way that covariates may affect Classes.

Below we will describe the most relevant special cases of this very general latent variable model,[20] most of which were described in Vermunt (2002b, 2003, 2004, and 2005) and Vermunt and Magidson (2005). We then provide some expressions for the exact forms of the various linear predictors in models with GClasses, GCFactors, and GCovariates.

Model restrictions One can use the parameter constraints "Class Independent", "No Effect", and "Merge Effects", implying equal λ's (β's) among all Classes, zero λ's (β's) in selected Classes, and equal λ's (β's) in selected Classes.

ML (PM) estimation and technical settings Similar to what was discussed in the context of CFactors, with GCFactors, the marginal density $P(\mathbf{y}_j|\mathbf{z}_j)$ described in equation (25) is approximated using Gauss-Hermite quadrature. With three GCFactors and B quadrature nodes per dimension, the approximate density equals

$$P(\mathbf{y}_j|\mathbf{z}_j, \mathbf{z}^g_j) \approx \sum_{x^g=1}^{K^g} \sum_{b_1=1}^{B} \sum_{b_2=1}^{B} \sum_{b_3=1}^{B} P(x^g|\mathbf{z}^g_j)\, P(\mathbf{y}_j|\mathbf{z}_j, x^g, F^g_{b_1}, F^g_{b_2}, F^g_{b_3})\, P^g_{b_1}\, P^g_{b_2}\, P^g_{b_3}.$$

ML (PM) estimates are found by a combination of the upward-downward variant of the EM algorithm developed by Vermunt (2003, 2004) and Newton-

[20] In fact, the multilevel LC model implemented in Latent GOLD Choice is so general that many possibilities remain unexplored as of this date. It is up to Latent GOLD Choice Advanced users to further explore its possibilities.

Raphson with analytic first-order derivatives.[21]

The only new technical setting in multilevel LC Choice models is the same as in models with CFactors; that is, the number of quadrature nodes to be used in the numerical integration concerning the GCFactors. As explained earlier in the context of models with CFactors, the default value is 10, the minimum 2, and the maximum 50.

7.2 Application Types

7.2.1 Two-level LC Choice model

The original multilevel LC model described by Vermunt (2003) and Vermunt and Magidson (2005b) was meant as a tool for multiple-group LC analysis in situations in which the number of groups is large. The basic idea was to formulate a model in which the latent class distribution (class sizes) is allowed to differ between groups by using a random-effects approach rather than by estimating a separate set of class sizes for each group – as is done in a traditional multiple-group analysis.

When adopting a nonparametric random-effects approach (using GClasses), one obtains the following multilevel LC Choice model:

$$P(\mathbf{y}_j|\mathbf{z}_j) = \sum_{x^g=1}^{K^g} P(x^g) \left[\prod_{i=1}^{I_j} \sum_{x=1}^{K} P(x|x^g) \prod_{t=1}^{T_i} P(y_{jit}|x, \mathbf{z}_{jit}^{att}, \mathbf{z}_{jit}^{pred}) \right],$$

in which the linear predictor in the logistic model for $P(x|x^g)$ equals $\eta_{x|x^g} = \gamma_{x^g,x0}$. Here, we are in fact assuming that the intercept of the model for the latent classes differs across GClasses.

When adopting a parametric random-effects approach (GCFactors), one obtains

$$P(\mathbf{y}_j|\mathbf{z}_j) = \int_{-\infty}^{\infty} f(F_{1j}^g) \left[\prod_{i=1}^{I_j} \sum_{x=1}^{K} P(x|F_{1j}^g) \prod_{t=1}^{T} P(y_{jit}|x, \mathbf{z}_{jit}^{att}, \mathbf{z}_{jit}^{pred}) \right] \mathrm{d}F_{1j}^g,$$

where the linear term in the model for $P(x|F_{1j}^g)$ equals $\eta_{x|F_{1j}^g} = \gamma_{x0} + \lambda_{x01}^{0,g} \cdot F_{1j}^g$. Note that this specification is the same as in a random-intercept model for a nominal dependent variable.

[21] Numeric second-order derivatives are computed using the analytical first-order derivatives.

Vermunt (2005) expanded the above parametric approach with covariates and random slopes, yielding a standard random-effects multinomial logistic regression model, but now for a *latent* categorical outcome variable. With covariates and multiple random effects, we obtain

$$P(\mathbf{y}_j|\mathbf{z}_j) = \int_{\mathbf{F}_j^g} f(\mathbf{F}_j^g) \left[\prod_{i=1}^{I_j} \sum_{x=1}^{K} P(x|\mathbf{z}_{ji}^{cov}, \mathbf{F}_j^g) \prod_{t=1}^{T} P(y_{jit}|x, \mathbf{z}_{jit}^{att}, \mathbf{z}_{jit}^{pred}) \right] \mathrm{d}\mathbf{F}_j^g,$$

where the linear predictor for x equals

$$\eta_{x|\mathbf{z}_{ji},\mathbf{F}_j^g} = \gamma_{x0} + \sum_{r=1}^{R} \gamma_{xr} \cdot z_{jir}^{cov} + \sum_{d=1}^{D^g} \lambda_{x0d}^{0,g} \cdot F_{dj}^g + \sum_{d=1}^{D^g} \sum_{r=1}^{R} \lambda_{xrd}^{0,g} \cdot F_{dj}^g \cdot z_{jir}^{cov}.$$

Also when adopting a nonparametric random-effects approach, one may include covariates in the multilevel LC model; that is,

$$\eta_{x|\mathbf{z}_{ji},x^g} = \gamma_{x^g,x0} + \sum_{r=1}^{R} \gamma_{x^g,xr} \cdot z_{jir}^{cov}.$$

This yields a model for the latent classes in which the intercept and the covariate effects may differ across GClasses. In fact, we have a kind of LC Regression structure in which the latent classes serve as a nominal dependent variable and the GClasses as latent classes.

An important extension of the above nonparametric multilevel LC models is the possibility to regress the GClasses on group-level covariates. This part of the model has the same form as the multinomial logistic regression model for the Classes in a standard LC or FM model.

7.2.2 LC discrete-choice models for three-level data

Another application type of the Latent GOLD Choice multilevel option is three-level regression modeling (Vermunt, 2004). A three-level LC conditional logit model would be of the form

$$P(\mathbf{y}_j|\mathbf{z}_j) = \sum_{x^g=1}^{K^g} P(x^g) \left[\prod_{i=1}^{I_j} \sum_{x=1}^{K} P(x) \prod_{t=1}^{T_i} P(y_{jit}|x, \mathbf{z}_{jit}^{pred}, \mathbf{z}_{jit}^{att}, x^g) \right].$$

Suppose we have a model for first choices with alternative-specific constants and P attributes. The simplest linear predictor in a model that includes

GClasses would then be

$$\eta_{m|\mathbf{z}_{jit},x,x^g} = \beta^{con}_{xm} + \sum_{p=1}^{P} \beta^{att}_{xp}\, z^{att}_{mitp} + \beta^{con,g}_{m,x^g},$$

which is a model in which (only) the constants are affected by the GClasses. A more extended model is obtained by assuming that also the attribute effects vary across GClasses; that is,

$$\eta_{m|\mathbf{z}_{jit},x,x^g} = \beta^{con}_{xm} + \sum_{p=1}^{P} \beta_{xp} \cdot z^{att}_{mitp} + \beta^{con,g}_{m,x^g} + \sum_{p=1}^{P} \beta^{att,g}_{p,x^g} \cdot z^{att}_{mitp}.$$

In practice, it seems to be most natural to allow effects of attributes and predictors that change values across replications to be Class dependent and effects of predictors that change values across cases to depend on the GClasses.

The most extended specification is obtained if all the effects are assumed to be Class dependent, which implies including Classes-GClasses (x-x^g) interactions. Such a model is defined as

$$\eta_{m|\mathbf{z}_{jit},x,x^g} = \beta^{con}_{xm} + \sum_{p=1}^{P} \beta_{xp} \cdot z^{att}_{mitp} + \beta^{con,g}_{xm,x^g} + \sum_{p=1}^{P} \beta^{att,g}_{xp,x^g} \cdot z^{att}_{mitp}.$$

It should be noted that in each of the above three models, identifying constraints have to be imposed on the parameters involving the GClasses. The attribute effects for the GClasses, for example, are restricted by $\sum_{x^g=1}^{K^g} \beta^{att,g}_{xp,x^g} = 0$, $\beta^{att,g}_{xp,1} = 0$, or $\beta^{att,g}_{xp,K^g} = 0$, for $1 \leq p \leq P$ and $1 \leq x \leq K$. In other words, the parameters in the model for the dependent variable either sum to zero across GClasses, are equal to zero for the first GClass, or are equal to zero for the last GClass.

7.2.3 Three-level random-coefficients conditional logit models

Combining the GCFactors from the multilevel model with the CFactors option makes it possible to specify three-level random-coefficient conditional logit models. These are similar to other types of three-level GLM regression models with parametric random effects (Im and Gionala, 1988; Skrondal and Rabe-Hesketh, 2004; Rodriguez and Goldman, 2001; Vermunt, 2002c, 2004). In terms of probability structure, this yields:

$$P(\mathbf{y}_j|\mathbf{z}_j) = \int_{\mathbf{F}^g_j} f(\mathbf{F}^g_j) \left[\prod_{i=1}^{I_j} \int_{\mathbf{F}_{ji}} f(\mathbf{F}_{ji}) \prod_{t=1}^{T_i} P(y_{jit}|\mathbf{z}^{pred}_{jit}, \mathbf{z}^{att}_{jit}, \mathbf{F}_{ji}, \mathbf{F}^g_j) \mathrm{d}\mathbf{F}_{ji} \right] \mathrm{d}\mathbf{F}^g_j.$$

The simplest special case is obtained by assuming that the conditional logit model contains random intercepts at both the case and the group level. The corresponding linear predictor in a model with P attributes equals

$$\eta_{m|\mathbf{z}_{jit},F_{1ji},F_{1j}^{g}} = \beta_m^{con} + \sum_{p=1}^{P} \beta_{xp} \cdot z_{mjitp}^{att} + \lambda_{m1}^{con} \cdot F_{1ji} + \lambda_{m1}^{con,g} \cdot F_{1j}^{g}.$$

Such a model containing a single CFactor and a single GCFactor will suffice in most three-level random-effects applications. However, similar to the random effects models discussed in the context of the CFactors option, this model can be expanded with random slopes at both levels using the factor-analytic or generalized random-effects specification illustrated in equation (24).

7.2.4 LC growth models for multiple response

Suppose one has a longitudinal data set containing multiple responses for each time point. The multiple responses could be used to build a time-specific latent classification, while the pattern of (latent) change over time could be described using a (LC) growth model. Specification of such a model would involve using the index i for the time points and the index j for the cases (time points are nested within cases). The LC model for the time points would be a LC Choice model. The multinomial logistic regression model for the (time-specific) latent classes will have the form of a LC growth model: class membership depends on time, where the intercept and possibly also the time slope is allowed to vary across individuals. This variation can be modelled using continuous random effects (GCFactors) and/or discrete random effects (GClasses).

7.2.5 Two-step IRT applications

Another application of the Latent GOLD Choice multilevel option is in IRT models for educational testing that assume a two-stage response process (Bechger et al., 2005; Westers and Kelderman, 1991, 1993). These models associate a discrete (usually binary) latent response to each observed item response, where a standard IRT model is specified for the discrete latent responses. A specific mechanism is assumed for the relationships between the latent and observed item responses. In Westers and Kelderman's SERE model, for example, the first latent class knows and the second latent class does not know the correct answer on a multiple choice item, implying that the first class gives the correct answer with probability one and the other class guesses with probabilities that depend on the attractiveness of the alternatives. Using the Latent GOLD Choice notation, the SERE model would be defined as a Rasch-like model for the latent classes

$$\eta_{x|\mathbf{z}_i,F_j^g} = \gamma_{x0} + \sum_{r=1}^{R} \gamma_{xr} \cdot z_{jir}^{cov} + \lambda_x^{0,g} \cdot \, F_j^g,$$

where the covariates are item dummies. The constraint that $P(y_{ji} = 1|x = 1) = 1.00$ can be imposed by using the offset option, and $P(y_{ji} = m|x = 2)$ is left unrestricted.

7.2.6 Non multilevel models

The final use of the multilevel option we describe here does not yield a multilevel model, but is a trick for estimating models that cannot be estimated any other way. The trick consists of using a Group ID variable that is identical to the Case ID or, equivalently, to have groups that consist of no more than one case each. GCFactors can then be used as CFactors. This makes it possible to define models in which CFactors affect the latent classes. Another possibility is to use the GClasses as an additional case-level nominal latent variable, yielding a model in which one nominal latent variable may affect another nominal latent variable.

8 Complex Survey Sampling

The Survey option makes it possible to obtain consistent parameter estimates and correct standard errors with complex sampling designs. This option can be used in combination with any model that can be estimated with Latent GOLD Choice. Parameter estimation is based on the so-called pseudo-ML estimator that uses the sampling weights as if they were case weights. Correct statistical tests with stratified and clustered samples, as well as with sampling weights and samples from finite populations are obtained using the linearization variance estimator.

Latent GOLD Choice also implement an alternative method to deal with sampling weights. This is a two-step procedure in which the model is first estimated without making use of the sampling weights, and in which subsequently the latent class sizes and covariate effects are corrected using the sampling weights.

8.1 Pseudo-ML Estimation and Linearization Estimator

The survey option can be used to take into account the fact that cases may

1. belong to the same stratum,
2. belong to the same primary sampling unit (PSU), often referred to as a sampling cluster,
3. contain a sampling weight,
4. be sampled from a finite population.

Let o denote a particular stratum, c a particular PSU in stratum o, and i a particular case in PSU c of stratum o. Moreover, let O be the number of strata, C_o the number of PSUs in stratum o, and I_{oc} the number of cases in PSU c of stratum o. The sampling weight corresponding to case i belonging to PSU c of stratum o is denoted by sw_{oci}, and the population size (total number of PSUs) of stratum o by N_o.[22]

From this notation, it can be seen that PSUs are nested within strata, and that cases are nested within PSUs. In other words, records with the same Case ID should belong to the same PSU, and all records with the same PSU identifier should belong to the same stratum. The population size N_o indicates the population number of PSUs in stratum o, and should thus have the same value across records belonging to the same stratum. Another thing that should be noted is that in *multilevel models*, the strata, PSUs, and sampling weights concern groups rather than cases; that is, one has strata and PSUs formed by groups and sampling weights for groups.

For parameter estimation, only the sampling weights need to be taken into account. When sampling weights are specified, Latent GOLD Choice will estimate the model parameters by means of pseudo-ML (PM) estimation (Skinner, Holt, and Smith, 1989). Recall that ML estimation involves maximizing

$$\log \mathcal{L} = \sum_{i=1}^{I} w_i \log P(\mathbf{y}_i|\mathbf{z}_i, \boldsymbol{\vartheta}),$$

[22] In Latent GOLD Choice, one can either specify the fraction $\frac{C_o}{N_o}$ or the population size N_o. If the specified number in "Population Size" is smaller than 1 it is interpreted as a fraction, otherwise as a population size.

where w_i is a case weight. In pseudo-ML estimation, one maximizes

$$\log \mathcal{L}_{pseudo} = \sum_{o=1}^{O} \sum_{c=1}^{C_c} \sum_{i=1}^{I_{oc}} sw_{oci} \log P(\mathbf{y}_{oic}|\mathbf{z}_{oic}, \boldsymbol{\vartheta}),$$

which is equivalent to maximizing $\log \mathcal{L}$ using the sampling weights as if they were case weights. In Latent GOLD Choice one may also have both case and sampling weights, in which case we get

$$\log \mathcal{L}_{pseudo} = \sum_{o=1}^{O} \sum_{c=1}^{C_c} \sum_{i=1}^{I_{oc}} w_{oci} \cdot \left[sw_{oci} \log P(\mathbf{y}_{oic}|\mathbf{z}_{oic}, \boldsymbol{\vartheta}) \right],$$

which is equivalent to performing ML estimation using the $sw_{oci} \cdot w_{oci}$ as "case" weights.

Each of the four complex sampling characteristics is taken into account by the so-called linearization estimator of variance-covariance matrix of the parameter estimates (Skinner, Holt, and Smith, 1989). Application of this method in the context of FM and LC models was proposed by Vermunt (2002b) and Wedel, Ter Hofstede, and Steenkamp (1998). The overall structure of $\widehat{\Sigma}_{survey}(\boldsymbol{\vartheta})$ is similar to the robust or sandwich estimator $\widehat{\Sigma}_{robust}(\boldsymbol{\vartheta})$ discussed earlier; that is,

$$\widehat{\Sigma}_{survey}(\boldsymbol{\vartheta}) = \widehat{\mathbf{H}}^{-1} \, \widehat{\mathbf{B}} \; \widehat{\mathbf{H}}^{-1}.$$

As can be seen, a matrix $\mathbf{B}$ is "sandwiched" between the inverse of the Hessian matrix. For the computation of $\mathbf{B}$, one needs two components: the contribution of PSU c in stratum o to the gradient of parameter k, denoted by g_{ock}, and its sample mean in stratum o, denoted by $\overline{g}_{ok}$. These are obtained as follows:

$$g_{ock} = \sum_{i=1}^{I_{oc}} sw_{oci} \frac{\partial \log P(\mathbf{y}_{oci}|\mathbf{z}_{oci}, \boldsymbol{\vartheta})}{\partial \vartheta_k}$$

and

$$\overline{g}_{ok} = \frac{\sum_{c=1}^{C_o} g_{ock}}{C_o}$$

Using these two components, element $B_{kk'}$ of $\mathbf{B}$ can be defined as

$$B_{kk'} = \sum_{o=1}^{O} \frac{C_o}{C_o - 1} (1 - \frac{C_o}{N_o}) \sum_{c=1}^{C_o} (g_{ock} - \overline{g}_{ok})(g_{ock'} - \overline{g}_{ok'}).$$

Note that if we neglect the finite population correction factor $(1-\frac{C_o}{N_o})$, $\mathbf{B}$ is the sample covariance matrix of the PSU-specific contributions to the gradient vector.

Various observations can be made from the formula for $B_{kk'}$. The first is that without complex sampling features (one stratum, single case per PSU, no sampling weights, and $\frac{C_o}{N_o} \approx 0$), the above procedure yields $\widehat{\boldsymbol{\Sigma}}_{robust}(\boldsymbol{\vartheta})$, which shows that $\widehat{\boldsymbol{\Sigma}}_{survey}(\boldsymbol{\vartheta})$ not only takes into account the sampling design, but is also a robust estimator of $\boldsymbol{\Sigma}(\boldsymbol{\vartheta})$. Second, the fact that gradient contributions are aggregated for cases belonging to the same PSU shows that the PSUs are treated as the independent observational units, which is exactly what we want. Third, the term $\frac{C_o}{C_o-1}$ is only defined if each stratum contains at least two PSUs: Latent GOLD Choice "solves" this problem by skipping strata for which $C_o = 1$ and by giving a warning that this happens. A common solution to this problem is to merge strata.

The design effect corresponding to a single parameter equals the ratio of its design corrected variance and its variance assuming simple random sampling. A multivariate generalization is obtained as follows (Skinner, Holt, and Smith, 1989):

$$\begin{aligned} deff &= \operatorname{tr}\left[\widehat{\boldsymbol{\Sigma}}_{standard}(\boldsymbol{\vartheta})^{-1}\widehat{\boldsymbol{\Sigma}}_{survey}(\boldsymbol{\vartheta})\right] / \, npar \\ &= \operatorname{tr}\left[(-\widehat{\mathbf{H}})\,\widehat{\mathbf{H}}^{-1}\,\widehat{\mathbf{B}}\;\widehat{\mathbf{H}}^{-1}\right] / \, npar = \operatorname{tr}\left[-\widehat{\mathbf{B}}\;\widehat{\mathbf{H}}^{-1}\right] / \, npar, \end{aligned}$$

where "tr" is the trace operator. The generalized design effect is thus the average of the diagonal elements of $-\widehat{\mathbf{B}}\,\widehat{\mathbf{H}}^{-1}$. Note that this number equals the average of the eigenvalues of this matrix.

8.2 A Two-step Method

Latent GOLD Choice also implements an alternative two-step method for dealing with sampling weights in LC analysis, which was described in Vermunt (2002b) and Vermunt and Magidson (2001). The procedure involves performing an unweighted analysis followed by a weighted analysis in which the parameters in the model part for the response variables are fixed to their unweighted ML (PM) estimates. More specifically, in step two, the class sizes and the covariates effects are adjusted for the sampling weights. The adjusted log-likelihood function that is maximized equals

$$\log \mathcal{L}_{adjusted} = \sum_{i=1}^{I} sw_i \, \log \sum_{x=1}^{K} P(x|\mathbf{z}_i, \boldsymbol{\vartheta}_{adjusted}) \, P(\mathbf{y}_i|x, \mathbf{z}_i, \widehat{\boldsymbol{\vartheta}}_{ML}),$$

where $\boldsymbol{\vartheta}_{adjusted}$ are the unknown parameters to be estimated.

The rationale of this procedure is that an unweighted analysis may yield more stable (more efficient) estimates for the parameters defining the latent classes, but yields biased class sizes and covariate effects. The latter are corrected in the second step of the procedure.

9 Latent Gold Choice's Advanced Output

This section describes the changes and additional items in the Latent GOLD Choice output sections when the Advanced options are used.

9.1 Model Summary

For multilevel models, the first part of the Model Summary output reports the number of groups (J) in addition to the number cases and replications. When the Survey option is used, the program reports the generalized design effect ($deff$), which is an overall measure indicating how many times larger the design corrected variances are compared to the asymptotic variances.

For multilevel models, *Chi-squared Statistics* are not reported and the bootstrap L^2 and $-2LL$-difference options are not available. When the Survey option is used, the bootstrap-based L^2 and LL-difference tests are corrected for the complex sampling design by multiplying the bootstrap replications' L^2 and $-2LL$-difference values by the generalized design effect $deff$. Note that the bootstrap replication samples themselves are obtained by simple random sampling.

In multilevel models, as in all other Latent GOLD Choice models, the number of cases serves as N (sample size) in the computation of the BIC and CAIC values that appear in the *Log-likelihood Statistics.* An alternative would have been to assume N to be equal to the number of groups instead of the number of cases. Users who prefer this alternative definition of BIC and CAIC may compute these statistics themselves.

The *Classification Statistics* contain information on how well one can predict an individual's CFactor scores and a group's GClass membership and GCFactor scores. For GClasses, one obtains the same information as for the latent classes (proportion of classification errors and three R^2 measures). For CFactors and GCFactors, one obtains only the standard R^2, which can be interpreted as a reliability measure. In multilevel models with covariates, *Covariate Classification Statistics* will contain information for the GClasses.

The *Prediction Statistics* are the same as in models without CFactors, GClasses, and GCFactors. The R^2_y measures indicates how well a model predicts the choices given all predictors, covariates, and latent variables.

9.2 Parameters

This section reports the parameters corresponding to CFactors, GClasses, and GCFactors. CFactors, GClasses, and GCFactors effects may appear in the *Model for Choices/Rankings/Ratings*. In multilevel models, GClasses and GCFactors may be used in the *Model for Classes*. When GClasses affect a particular term (the intercept or a covariate effect), one obtains a separate set of coefficients for each GClass. GCFactors enter as random effects in the regression model for the discrete latent variable(s). In models with GClasses, the parameters output contains the coefficients of the multinomial logistic regression *Model for GClasses*.

The reported Class-specific $R^2_{y|x}$ measures are obtained by averaging the predicted values over the other latent variables included in the model. This is the reason that in a one-Class model, the "Class-specific" $R^2_{y|1}$ may be lower that the overall R^2_y.

When the Survey option is used, one obtains design corrected standard errors and Wald statistics.

In models with CFactors, one obtains an output subsection called *Random Effects*. This subsection provides the CFactor effects $\mathbf{\Lambda}$ and the variance-covariance matrix of the random effects, $\mathbf{\Sigma_\Psi} = \mathbf{\Lambda}\,\mathbf{\Lambda}'$.

9.3 GProfile

The first part of this output section reports the sizes of the GClasses $[P(x^g)]$ and the probability of being in a certain latent class for each GClass $[P(x|x^g)]$. The second part of the GProfile section reports the GClass-specific probabilities for the choice variable. The computation of this part of the GProfile output is similar to the computation of the same kinds of numbers in the Profile output.

9.4 ProbMeans

In models with CFactors, the Probmeans output reports the average CFactor posterior mean for each covariate category.

9.5 Frequencies

Frequencies are not reported in multilevel LC models.

9.6 Classification

The *Standard Classification* output provides information on the CFactor and GCFactor posterior means $\widehat{E}(F_{di}|\mathbf{z}_i,\mathbf{y}_i)$ and $\widehat{E}(F^g_{dj}|\mathbf{z}_j,\mathbf{y}_j)$, the GClass posterior probabilities $\widehat{P}(x^g|\mathbf{z}_j,\mathbf{y}_j)$, and the modal GClass for each data pattern. The posterior means are obtained using Gauss-Hermite quadrature; for example,

$$\begin{aligned}\widehat{E}(F_{di}|\mathbf{z}_i,\mathbf{y}_i) &= \frac{\int_{-\infty}^{\infty} F_{di}\ P(\mathbf{y}_i|\mathbf{z}_i,F_{di})\,\mathrm{d}\,F_{di}}{\int_{-\infty}^{\infty} P(\mathbf{y}_i|\mathbf{z}_i,F_{di})\,\mathrm{d}\,F_{di}}\\ &\approx \frac{\sum_{b_d=1}^{B} F_{b_d}\ P(\mathbf{y}_i|\mathbf{z}_i,F_{b_d})\,P_{b_d}}{\sum_{b_d=1}^{B} P(\mathbf{y}_i|\mathbf{z}_i,F_{b_d})\,P_{b_d}}.\end{aligned}$$

In multilevel models with covariates, the *Covariate Classification* output section reports the GClass membership probabilities given group-level covariates $[\widehat{P}(x^g|\mathbf{z}^g_j)]$.

9.7 Output-to-file Options

The *Standard Classification* option can be used to write the CFactors and GCFactors posterior means, the GClasses posterior probabilities, and the modal GClass to an output file. In models with GClasses, *Covariate Classification* saves the classification of groups into GClasses based on group covariates to the output file.

The *Individual Coefficients* corresponding to CFactor effects are computed in a special way:

$$\widehat{\lambda}_{iqd} = \sum_{x=1}^{K} \widehat{P}(x|\mathbf{z}_i,\mathbf{y}_i)\ \left[\widehat{E}(F_{di}|\mathbf{z}_i,\mathbf{y}_i,x)\,\widehat{\lambda}_{xqd}\right],$$

where $\widehat{E}(F_{di}|\mathbf{z}_i, \mathbf{y}_i, x)$ is the posterior mean of F_{di} given that i belongs to latent class x. The $\widehat{\lambda}_{iqd}$ can be used together with the $\widehat{\beta}_{iq}$ to obtain HB-like predicted values for case i. The posterior standard deviation of $\widehat{\lambda}_{iqd}$ equals

$$\widehat{\sigma}_{\widehat{\lambda}_{iqd}} = \sqrt{\sum_{x=1}^{K} \widehat{P}(x|\mathbf{z}_i, \mathbf{y}_i) \left[\widehat{E}(F_{di}|\mathbf{z}_i, \mathbf{y}_i, x)\, \widehat{\lambda}_{xqd} - \widehat{\lambda}_{iqd}\right]^2},$$

HB-like individual coefficients for a "full" intercept or predictor term may also be obtained by summing the various individual coefficient components for that term. For example, for a random-intercept model such as given in equation (23), the HB-like individual coefficient for a “full” alternative-specific constant is computed by summing $\widehat{\beta}_{im}^{con}$ and $\widehat{\lambda}_{im1}^{con}$.

In multilevel models, the *Cook's D* value is computed per group rather than per case. Thus, rather than for detecting influential cases, it can be used for detecting influential groups.

10 Bibliography

Agresti, A. (2002). *Categorical data analysis.* Second Edition, New York: Wiley.

Aitkin (1999). A general maximum likelihood analysis of variance components in generalized linear models. Biometrics, 55, 218-234.

Andrews, R.L., Ainslie, A., and Currim, I.S. (2002). An empirical comparison of logit choice models with discrete versus continuous representations of heterogeneity, *Journal of Marketing Research*, 39, 479-487.

Andrews, R.L., and Currim, I.S. (2003). A Comparison of Segment Retention Criteria for Finite Mixture Logit Models, *Journal of Marketing Research*, 40, 235-243.

Banfield, J.D., and Raftery, A.E. (1993). Model-based Gaussian and non-Gaussian clustering. *Biometrics*, 49, 803-821.

Bartholomew, D.J., and Knott, M. (1999). *Latent variable models and factor analysis.* London: Arnold.

Bechger, T.M., Maris, G., Verstralen, H.H.F.M., and Verhelst, N.D. (2005). The Nedelsky model for multiple-choice items. In A. Van der Ark, M.A. Croon and K. Sijtsma (eds), *New Developments in Categorical Data Analysis for the Social and Behavioral Sciences*, 187-206, Mahwah: Erlbaum.

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

Böckenholt, U. (2001). Mixed-effects analyses of rank-ordered data. *Psychometrika*, 66, 45-62.

Böckenholt, U. (2002). Comparison and choice: analyzing discrete preference data by latent class scaling models. J.A. Hagenaars and A.L. McCutcheon (eds.), *Applied latent class analysis*, 163-182. Cambridge: Cambridge University Press.

Buse, A. (1982). The likelihood ratio, Wald, and Lagrange multiplier tests: An expository note. *The American Statistician*, 36, 153-157.

Clogg, C.C. (1981). New developments in latent structure analysis. D.J. Jackson and E.F. Borgotta (eds.), *Factor analysis and measurement in sociological research*, 215-246. Beverly Hills: Sage Publications.

Clogg, C.C., Rubin, D.R., Schenker, N., Schultz, B., Weidman, L. (1991). Multiple imputation of industry and occupation codes in census public-use samples using Bayesian logit regression. *Journal of the American Statistical Association*, 86, 68-78.

Cohen, S. (2003). Maximum difference scaling: improved measures of importance and preference for segmentation. *Proceedings Sawtooth Software Conference 2003.*

Collins, L.M., Fidler, P.F., Wugalter, S.E., and Long, L.D. (1993). Goodness-of-fit testing for latent class models. *Multivariate Behavioral Research*, 28, 375-389.

Croon, M.A. (1989). Latent class models for the analysis of rankings. G. De Soete, H. Feger, and K.C. Klauer, *New developments in psychological choice modeling,* 99-121. Elsevier Science Publishers.

Dayton, C.M., and Macready, G.B. (1988). Concomitant-variable latent-class models. *Journal of the American Statistical Association* , 83, 173-178.

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

Dias, J.G. (2004). *Finite Mixture Models: Review, Applications, and Computer-intensive Methods.* Phd. Dissertation. Research School Systems, Organisation and Management (SOM), Groningen of University, The Netherlands.

Dillon, W.R., and Kumar, A. (1994). Latent structure and other mixture models in marketing: An integrative survey and overview. R.P. Bagozzi (ed.), *Advanced methods of Marketing Research*, 352-388, Cambridge: Blackwell Publishers.

Galindo-Garre, F., Vermunt, J.K., and Croon M.A. (2002). Likelihood-ratio tests for order-restricted log-linear models: A comparison of asymptotic and bootstrap methods. *Metodología de las Ciencias del Comportamiento*, 4, 325-337.

Galindo-Garre, F., Vermunt, J.K., and W. Bergsma (2004). Bayesian posterior estimation of logit parameters with small samples. *Sociological Methods and Research*, 33, 88-117.

Gill, P.E., Murray, W., and Wright, M.H. (1981). *Practical optimization.* London: Academic Press.

Gelman, Andrew, Carlin, John B., Stern, Hal .S., and Robin, Donald B. (1995). *Bayesian data analysis.* London: Chapman & Hall.

Goodman, L.A. (1974a). The analysis of systems of qualitative variables when some of the variables are unobservable: Part I - A modified latent structure approach. *American Journal of Sociology*, 79, 1179-1259.

Goodman, L.A. (1974b). Exploratory latent structure analysis using both identifiable and unidentifiable models. *Biometrika*, 61, 215-231.

Goodman, L.A. (1979). Simple models for the analysis of association in cross-classifications having ordered categories. Journal of the American Statistical Association, 74, 537-552.

Haberman, S.J. (1988). A stabilized Newton-Raphson algorithm for log-linear models for frequency tables derived by indirect observations. C.Clogg (ed.), *Sociological Methodology 1988*, 193-211. San Francisco: Jossey-Bass.

Hedeker, D. (2003). A mixed-effects multinomial logistic regression model. *Statistics in Medicine*, 22, 1433-1446.

Im, S., and Gionala, D. (1988). Mixed models for binomial data with an application to lamb mortality, *Applied Statistics*, 37, 196-204.

Imbens, G.W., and Rubin, D.B. (1997). Estimating outcome distributions for compliers in instrumental variable models, *Review of Economic Studies*, 64, 555-574.

Kamakura, W.A., and Russell, G.J. (1989). A probabilistic choice model for market segmentation and elasticity structuring. *Journal of Marketing Research*, 26, 379-390.

Kamakura, W.A., Wedel, M., and Agrawal, J. (1994). Concomitant variable latent class models for the external analysis of choice data. *International Journal of Research in Marketing*, 11, 451-464.

Laird, N. (1978). Nonparametric maximum likelihood estimation of a mixture distribution. *Journal of the American Statistical Association* , 73, 805-811.

Langeheine, R., Pannekoek, J., and Van de Pol, F. (1996). Bootstrapping goodness-of-fit measures in categorical data analysis. *Sociological Methods and Research*, 24, 492-516.

Lenk, P.J., and DeSarbo, W.S. (2000). Bayesian inference for finite mixture models of generalized linear models with random effects, *Psychometrika*, 65, 93-119.

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.

Little, R.J., and Rubin, D.B. (1987). *Statistical analysis with missing data.* New York: Wiley.

Louviere, J.J., Hensker D.A., and Swait, J.D. (2000). *Stated choice methods: analysis and application.* Cambridge: Cambridge University Press.

Magidson, J. (1981). Qualitative variance, entropy, and correlation ratios for nominal dependent variables. *Social Sciences Research*, 10, 177-194.

Magidson, J. (1996). Maximum likelihood assessment of clinical trials based on an ordered categorical response. *Drug Information Journal*, 30 (1), 143-170.

Magidson, J., Eagle, T., and Vermunt, J.K. (2003). New developments in latent class choice modeling. *Proceedings Sawtooth Software Conference 2003.*

Magidson, J. and Vermunt, J.K. (2001). Latent class factor and cluster models, bi-plots and related graphical displays, *Sociological Methodology*, 31, 223-264.

Magidson, J., and Vermunt, J.K, (2004) Latent class analysis. D. Kaplan (ed.), *The Sage Handbook of Quantitative Methodology for the Social Sciences*, Chapter 10, 175-198. Thousand Oakes: Sage Publications.

McFadden (1974). Conditional logit analysis of qualitative choice behaviour. I. Zarembka (ed.), Frontiers in econometrics, 105-142. New York: Academic Press.

McFadden, D. and Train, D. (2000). Mixed MNL models for discrete response. *Journal of Applied Econometrics*, 15, 447-470.

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

Natter, M. and Feurstein, M. (2002). Real world performance of choice-based conjoint models, *European Journal of Operational Research*, 137, 448-458.

Patterson, B.H., Dayton,C.M.; Graubard, B.I. (2002). Latent class analysis of complex sample survey data: application to dietary data. *Journal of the American Statistical Association*, 97, 721-728.

Rodriguez, G. and Goldman, N. (2001). Improved estimation procedures for multilevel models for binary response: a case study. *Journal of the Royal Statistical Society*, Series A, 164, 339-355.

Schafer, J.L. (1997). *Analysis of incomplete multivariate data* . London: Chapman & Hall.

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

Skinner, C.J., Holt, D., and Smith, T.M.F. (eds.) (1989). *Analysis of Complex Surveys*, New York: Wiley.

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

Van der Ark, L.A., and Van der Heijden, P.G.M. (1998). Graphical display of latent budget and latent class analysis, with special reference to correspondence analysis. J. Blasius and M.Greenacre (eds.) *Visualization of categorical data*. Boston: Academic Press.

Van der Heijden, P.G.M., Dessens, J., and Böckenholt, U. (1996). Estimating the concomitant-variable latent class model with the EM algorithm. *Journal of Educational and Behavioral Statistics*, 5, 215-229.

Van der Heijden P.G.M., Gilula, Z., and Van der Ark, L.A. (1999). On a relationship between joint correspondence analysis and latent class analysis. M.Sobel and M.Becker (eds.), *Sociological Methodology 1999* , 81-111. Boston: Blackwell Publishers.

Vermunt, J.K. (1997). *Log-linear models for event histories.* Thousand Oakes: Series QASS, vol 8. Sage Publications.

Vermunt, J.K. (2002a). A general latent class approach for dealing with unobserved heterogeneity in the analysis of event history data. J.A. Hagenaars and A.L. McCutcheon (eds.), *Applied latent class analysis* , 383-407. Cambridge: Cambridge University Press.

Vermunt, J.K. (2002b). Comments on Latent class analysis of complex sample survey data. *Journal of the American Statistical Association*, 97, 736-737.

Vermunt, J.K. (2002c). An Expectation-Maximization algorithm for generalised linear three-level models. *Multilevel Modelling Newsletter*, 14, 3-10.

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. (2005). Mixed-effects logistic regression models for indirectly observed outcome variables. *Multivariate Behavioral Research*, in press.

Vermunt, J.K., and Magidson, J. (2000). *Latent GOLD User's Manual.* Boston: Statistical Innovations.

Vermunt, J.K., and Magidson, J. (2001): *Latent Class Analysis with Sampling Weights*, Paper presented at the 6th annual meeting of the Methodology Section of the American Sociological Association, University of Minnesota, May 4-5, 2001.

Vermunt, J.K., and Magidson, J. (2002). Latent Class Models for Classification, *Computational Statistics and Data Analysis*, 41, 531-537.

Vermunt, J.K. and Magidson, J. (2003). Nonparametric random-coefficients models. M. Lewis-Beck, A. Bryman, and T.F. Liao (eds.), *Encyclopedia of Research Methods for the Social Sciences.* NewBury Park: Sage Publications, Inc.

Vermunt, J.K, and Magidson, J. (2005). Hierarchical mixture models for nested data structures. C. Weihs and W. Gaul (eds.), *Classification: The Ubiquitous Challenge,* in press. Heidelberg: Springer.

Vermunt, J.K. and Van Dijk. L. (2001). A nonparametric random-coefficients approach: the latent class regression model. *Multilevel Modelling Newsletter*, 13, 6-13.

Wedel, M., and DeSarbo, W.S (1994). A review of recent developments in latent class regression models. R.P. Bagozzi (ed.), *Advanced methods of Marketing Research*, 352-388, Cambridge: Blackwell Publishers.

Wedel, M., and DeSarbo, W.S (2002). J.A. Hagenaars and A.L. McCutcheon (eds.), *Applied latent class analysis*, 366-382. Cambridge: Cambridge University Press.

Wedel, M., Ter Hofstede, F., and Steenkamp, J.-B.E.M. (1998). Mixture model analysis of complex samples, *Journal of Classification*, 15, 225-244.

Westers, P., and H. Kelderman, (1991). Examining differential item functioning due to item difficulty and alternative attractiveness. *Psychometrika,* 57, 107-118.

Westers, P., and H. Kelderman, (1993). *Generalizations of the Solution- error Response-error Model.* Research Report 93-1, Faculty of Educational Science and Technology, University of Twente.

11 Notation

11.1 Basic Models

$P(\cdot)$	probability
i, I	case index, # of cases
t, T_i	replication index, # of replications for case i
y_{it}	response of case i at replication t
m	category of the response variable
y^*_m,	score assigned to category m of a rating variable
x	nominal latent variable, a particular latent class
K	# of latent classes
r, R	covariate index, # of covariates
p, P	attribute index, # of attributes
q, Q	predictor index, # of predictors
z^{cov}_{ir}	covariate
z^{att}_{mitq}	attribute
z^{pred}_{itq}	predictor
η	linear predictor
β, γ	parameter in model for y_{it}, parameter in model for x
τ_{ix}	known-class indicator
w_i	case weight
v_{it}	replications weight
u, U	"covariate" pattern index, # of "covariate" patterns
i^*, I^*	unique data pattern index, # of unique data patterns
N	total sample size (after weighting)

11.2 Advanced Models

d, D	CFactor index, # of CFactors
F_{id}	scores of case i CFactor d
λ_d	an effect of CFactor d
j, J	group index, # of groups
I_j	# of cases in group j
y_{jit}	response of case i of group j at replication t
$\mathbf{y}_j$	vector of responses of group j
g	group-level quantity
x^g	group-level nominal latent variable, a particular group GClass
$z_{jr}^{g,cov}$	group-level covariate
F_{jd}^g	score of group j of group-level continuous factor (GCFactor) d
$\gamma^g, \beta^g, \lambda^g$	group-level parameters
o, O	stratum, # of strata
c, C_o	PSU, # of PSU's in stratum o
sw_{oci}	sampling weight
I_{oc}	# of cases in PSU c of stratum o
N_o	total # of PSUs in population in stratum o