Contents

31

Latent GOLD

Jeroen K. Vermunt

Department of Methodology and Statistics, Tilburg University

CONTENTS

31.1 Introduction

Latent GOLD® is developed by Jeroen Vermunt and Jay Magidson, with Windows programming assistance from Alexander Ahlstrom. It was designed originally with an SPSS-like point-and-click GUI interface, implementing the most important types of latent class and mixture models. The initial release in 2000 (version 2.0) contained three modules: Cluster, Discrete Factor (DFactor), and Regression, representing latent class (LC) analysis with a single nominal latent variable, LC with multiple ordinal latent variables, and mixture regression analysis of two-level data sets. Each of these modules could accommodate response variables of different scale types (e.g., continuous, categorical). The DFactor module is in fact an IRT routine implementing multidimensional discretized IRT models with fixed nodes for the discretized traits.

The primary extension in *Latent GOLD 3.0*, released in 2003, was the Choice module, a special type of Regression module tailored to discrete choice data which utilizes alternative-specific predictors. This extension required special provisions for data handling and model specification unique to first choice, ranking and various partial ranking designs.

Latent GOLD 4.0 was released in 2005. Major new features included the ability to define models containing continuous (in addition to discrete) latent variables, to estimate multilevel latent variable models, and to deal with data sets obtained from complex sampling schemes. This version made it possible

to define IRT models with continuous latent traits, including mixture IRT models and certain types of multilevel IRT models.

Version 4.5, released in 2008, introduced the Syntax module. This module allows users to define their own latent variable models with intuitive syntax statements, which allows for the specification of parameter constraints and starting values, performing simulations and power computations, using models for multiple imputation, using saved model parameters to score new data sets, as well as other capabilities. More advanced functionalities of the computational code of *Latent GOLD* that were not accessible with the point-and-click user interface became available to the (more advanced) user via the syntax.

The newest versions 5.0 and 5.1 from 2013 and 2015, respectively, contain several technical improvements which may reduce computation time enormously. The first improvement is the use of multiple processors, which allow for computation distributed over multiple scores. The second is the replacement of numerical second-order derivatives, used by the Newton-Raphson estimation algorithm and for standard error computation in complex multilevel models and latent Markov models, by analytic expressions. The many other extensions include the possibility to specify a log-linear scale model for categorical response variables, to perform three-step latent class analysis, to specify user-defined Wald tests, and to estimate latent Markov models using a point-and-click user interface. Version 5.1 allows for 64-bit computing, making it possible to use all available RAM of the current computers.

Latent GOLD is written in C++. Its users are researchers from universities and commercial and government organizations (in approximately equal numbers). The academic users are from a wide variety of applied fields, while a large percentage of nonacademic users are from marketing research.

31.2 Functionality

Latent GOLD performs maximum-likelihood estimation of a very general class of latent variable models. A *Latent GOLD* model may contain multiple latent variables, which can be nominal, ordinal (classes with fixed locations), continuous (normally distributed or with a user-defined distribution), or any combination of these. The latent variables may be from a hierarchical structure; that is, one may define latent variables that vary across groups, across individuals (cases), and/or across time points (dynamic latent variables). The response variables can be binary, ordinal, nominal, counts, continuous, or any combination of these. Moreover, explanatory variables affecting the latent variables and/or the responses may be included in the model. Examples of models that can be estimated with *Latent GOLD* include:

- LC models for categorical response variables, including models with ex-

planatory variables, models with local dependencies, models for ordinal variables, models with order-restricted classes, models with (partially) known class memberships, and multilevel LC models;

- factor analysis models, multiple group factor models, factor models with explanatory variables, mixture factor models, multilevel (mixture) factor models;
- IRT models for binary, ordinal and nominal items, discretized IRT models, multidimensional IRT models, multiple group IRT models, IRT models with covariates, IRT models with local dependencies, mixture IRT models, multilevel IRT models, bi-factor models, IRT models for testlets;
- two-, three- and four-level regression models with random effects for response variables of any type, mixture regression models, multilevel mixture regression models, regression models with both continuous and discrete random effects;
- latent Markov models (also with many time points), including models with predictors, a discrete mixture, a multilevel structure, etc..

Latent GOLD implements marginal-maximum likelihood estimation using a combination of the EM and Newton-Raphson algorithms. It automatically generates multiple random start values for the parameter estimates to reduce the chance of obtaining a local solution.

A unique feature of *Latent GOLD* is the option to estimate extended IRT models consisting of one or more continuous latent variables in combination with one or more discrete latent variables. This makes it possible to use flexible (or even unspecified) distributions for the latent traits, as well as to define mixture IRT models with latent classes at the individual level, the group level, or both.

31.3 User Interface

The *Latent GOLD* Windows user interface allows opening multiple data files, including SPSS, text, and comma-delimited files. For each of these data sets, multiple models can be run. These models can be specified either by using the point-and-click modules (Cluster, DFactor, Regression, Choice, Step3, or Markov) or the Syntax module. Output has a tree-structure, with entries reporting information on the data set, statistics, parameter estimates and tests, predicted values for probabilities and means, diagnostics for residual local dependencies, estimates of the discrete and continuous latent variables scores, details on the iteration steps during estimation, and some graphs. Text output appears in a spread-sheet format, which can easily be copied and

pasted into a spreadsheet for further processing. Also, output can be changed interactively; that is, columns can be added or removed and graphs can be modified. Output can be stored in text and html format (per data set, per model, or per output section). It is also possible to write selected output, for example, latent class assignments, estimated ability scores, or simulated responses, to data files.

A unique feature of *Latent GOLD* sessions using the point-and-click modules is the option to label and store the specified models in a file with extension lgf, which can be saved for later restoration. Models defined with the Syntax module can similarly be saved to a file with extension lgs. Syntax models can also be saved with parameter estimates, for use as starting values when models are rerun. Syntax files can be read and modified with any text editor.

Latent GOLD can be run as a Windows program but also in batch mode (as a DOS-like console program). The latter option may be useful in research and production environments. Special provisions are available for storing *Latent GOLD* output when performing simulation studies.

31.4 Sample Input and Output

The examples will be for the Syntax module, which is the easiest and most flexible way to formulate IRT models. A Syntax model contains three sections: Options, Variables, and Equations. The Options section specifies the technical settings of the algorithm, choice of starting values, handling of missing values, and the number of quadrature nodes for numerical integration, as well as the type of output required. In the Variables section, the user selects (i) variables from the data file – which can be independent variables (covariates, grouping variables), dependent variables (items), id variables, weight variables, etc. – and (ii) the latent variables to be included in the model. For each of these variables the user should set the scale type; for nominal and ordinal latent variables, the user should also specify the number of categories (latent classes). The Equations section can be used to define regression equations for the latent and dependent variables, provide equations for the free variances, covariances, and associations between these variables, and to introduce constraints on the model parameters.

Following is an example of the Variables and Equations section for a simple IRT model, where the item scores (y1, y2, etc.) are treated as the dependent variables and are assumed to be controlled by a latent continuous variable named 'theta':

```
variables
  dependent y1, y2, y3, y4, y5, y6;
  latent theta continuous;
equations
```

```
  theta;
  y1  <- 1 + (1) theta;
  y2 - y6  <- 1 + theta;
```

In the Equations section, the first line defines the variance of theta, which is estimated as a free parameter. The second line contains the regression equation for the first item, which has a free intercept ("1" means intercept) and a theta slope (discrimination parameter) fixed to 1 (for identification, either the variance of theta or the slope of one of the items must be set to 1). The third line defines the unrestricted equations for the remaining items y2 to y6. In this specification, we use the default scale type for the dependent variables, which implies that they are modeled using an adjacent category logit model, that is, a 2PL model when the items are binary or a generalized partial credit model when the items are polytomous. Fixing the slope to 1 for all items would yield a Rasch or partial credit model. Other link functions can be used for ordinal items, such as "cumlogit" or "probit", yielding graded-response and normal ogive models, respectively.

The nominal response model is obtained through the following input specification:

```
variables
  dependent y1 nominal, y2 nominal, y3 nominal, y4 nominal,
    y5 nominal, y6 nominal;
  latent theta continuous;
equations
  (1) theta;
  y1 - y6 <- 1 + theta;
```

This time the items are defined to be nominal and the variance of theta rather than the slope for y1 is restricted to 1.

A unique feature of *Latent GOLD* is its option to relax local independence assumptions, which is achieved with multidimensional or multilevel modeling (see below) or by adding log-linear association terms for item pairs. For instance, suppose the responses to items y3 and y4 and items y5 and y6 are not assumed to be given independently. The assumption can be introduced by adding the lines "y3 `<->` y4;" and "y5 `<->` y6;" to the input code, which add residual correlations to the above IRT models.

Another unique option is the possibility to define discretized IRT models; that is, models with a latent trait distribution approximated using latent classes with ordered (or fixed) locations or unordered (or free) locations. A 3-class model with fixed locations is specified as follows:

```
variables
  dependent y1, y2, y3, y4, y5, y6;
  latent dtheta ordinal 3 scores=(-1 0 1);
equations
  dtheta <- 1;
```

```
  y1 - y5  <- 1 + dtheta;
```

We refer to this specification as a latent class factor or discrete factor model. The same model can also be specified as:

```
variables
  dependent y1, y2, y3, y4, y5, y6;
  latent theta continuous, dtheta ordinal 3 scores=(-1 0 1);
equations
  dtheta <- 1;
  theta <- dtheta;
  (0) theta;
  y1 <- 1 + (1) theta;
  y2 - y6 <- 1 + theta;
```

What happens here is that the continuous theta is regressed on the discrete 3-class dtheta variable and its residual variance is set to 0. This means that theta is replaced by dtheta times a slope parameter which rescales the three fixed nodes of -1, 0 and 1. The same model can easily be turned into one with free locations of the classes by changing the definition of the discrete latent variable into "dtheta nominal 3". It is also possible to relax the assumption of a residual variance of theta equal to 0, which yields an IRT model with a latent trait distributed as a mixture of normals.

A multiple-group IRT model can be defines as follows:

```
variables
  independent agecat nominal;
  dependent y1, y2, y3, y4, y5, y6;
  latent theta continuous;
equations
  theta | agecat;
  theta <- (m) 1 | agecat;
  y1 - y6 <- 1 + (1) theta;
  y2 <- 1 | agecat + theta;
  y3 - y6 <- 1 + theta;
  m[1] = 0;
```

In this specification, the mean and variance of theta are allowed to vary across age categories (indicated with "| agecat"). In addition, the intercept (difficulty) for the second item is allowed to vary across groups, which means that the item may show uniform DIF. The mean of age group 1 is fixed at 0 for identification purposes. It is straightforward to define models with multiple grouping variables and numeric covariates. The latter will be defined as numeric independent variables; they are used in the regression equations with a "+" instead of a "|". When the grouping variable is not an observed (independent) variable but a nominal latent variable, one obtains a mixture IRT model. Except for the variables definition, the model specification is the same as for a multiple-group IRT model.

The following example is for a two-dimensional IRT model:

```
variables
  dependent y1, y2, y3, y4, y5, y6;
  latent theta1 continuous, theta2 continuous;
equations
  theta1;
  theta2;
  theta1 <-> theta2;
  y1 <- 1 + (1) theta1
  y2 - y3 <- 1 + theta1;
  y4 <- 1 + theta1 + (1) theta2;
  y5 - y6 <- 1 + theta2;
```

Its two traits are allowed to correlate (indicated with theta1 `<->` theta2), while the fourth item is assumed to be affected by both traits.

A multilevel IRT model for individuals nested in groups (pupils in schools, say) can be specified using a group-level latent variable:

```
variables
  groupid schoolnr;
  dependent y1, y2, y3, y4, y5, y6;
  latent theta2 continuous group, theta1 continuous;
equations
  theta2;
  theta1;
  theta1 <- (1) theta2;
  y1 <- 1 + (1) theta1;
  y2 - y6  <- 1 + theta1;
```

In this specification, schools differ randomly with respect to the pupils' abilities, but not with respect to the item responses given the pupils' abilities. The latter implies that the school- and student-level measurement models are assumed to be identical. An alternative model is:

```
variables
  groupid schoolnr;
  dependent y1, y2, y3, y4, y5, y6;
  latent theta2 continuous group, theta1 continuous;
equations
  theta2;
  theta1;
  y1  <- 1 + (1) theta1 + (1) theta2;
  y2 - y6  <- 1 + theta1 + theta2;
```

In this model, the discrimination parameters are allowed to differ across levels. The multilevel feature can also be used to estimate bi-factor models, models

for testlets, and longitudinal models in a computationally efficient way; that is, as two-level IRT models.

In the examples discussed thus far, the data file was assumed to be in wide format, with a single record per person and each of the item responses in a separate column. Alternatively, a long data file format can be used, with multiple records per person and all responses in a single column. Identifiers are available for persons and items. Setting up an IRT model for a long file requires introduction of a case ID and specification of how the parameters vary across items.

The following syntax specifies the same simple IRT model as in the first example:

```
variables
  caseid childnr;
  independent itemnr nominal;
  dependent y;
  latent theta continuous;
equations
  theta;
  y  <- 1 | itemnr + (a) theta | itemnr;
  a[1] = 1;
```

The other examples can be translated similarly. Model definition using a long file is convenient for defining IRT models with item-specific predictors, usually referred to as linear logistic test models. An example of this type of model is:

```
variables
  caseid childnr;
  independent z1, z2, z3;
  dependent y;
  latent theta continuous;
equations
  theta;
  y  <- 1 + (1) theta + z1 + z2 + z3 + z1 z2;
```

The example is for a Rasch-like model with main effects for the three predictors and an interaction effect of z1 and z2.

31.5 Performance

Latent GOLD runs under Windows operating systems. Though no special versions for Apple or Unix computers exist, the program runs perfectly on these platform using the Windows emulator Wine. There are no limitations as to the maximum number of records or variables; the only limitation is

the computer's RAM. Starting from version 5.1, *Latent GOLD* is a 64-bit program, which implies that it can use all available RAM.

As already noted, the algorithms for the maximization of the log-likelihood function are a combination of EM and Newton-Raphson procedures, which are very fast. A 2PL model for 20 items and 2,000 observations runs in less than 1 second; a multilevel 2PL model for the same data in less than 10 seconds. These run times were obtained using a single start set, a switch from EM to Newton-Raphson after 50 iterations, numerical integration with 10 quadrature points, and the multiple-processors option. Generally, run time increases linearly with the number of persons, items, and quadrature points per dimension, and exponentially with the number of latent dimensions.

31.6 Support

The *Latent GOLD* program comes with a user guide explaining the interface. A technical manual provides details regarding the models and the estimation methods, and describes the various output sections. Separate versions of these documents are available for the Choice module, which can be obtained as a separate program. A third manual, the *Latent GOLD Syntax Guide*, documents and illustrates the Syntax.

The *Latent GOLD* Help menu offers the options "Syntax Examples" and "GUI Examples", which guide the user through a large number of examples of applications. Upon selection of a specific application, a corresponding lgs or lgf file is opened, restoring the relevant data file and all models associated with it.

Statistical Innovations provides technical support to *Latent GOLD* users. Extended support, typically requested to deal with issues not directly related to the program, such as tailoring models to specific research questions and data sets, can be obtained in the form of consulting. The company's website is www.statisticalinnovations.com.

31.7 Availability

The *Latent GOLD* program version 5.1 can be purchased via the internet from *Statistical Innovations* or *Science Direct*. A free trial version can be downloaded, which is fully operational, but with the limitation that it works only with the (many) example data sets accompanying the program. The manuals may be downloaded from the company's website or will be available

automatically, along with the Syntax Examples and GUI Examples menu entries, when installing the demo version.

The price depends on the modules one needs, for example, whether one also wishes the Choice module and/or the Advanced+Syntax module, as well as on the type of license one prefers (annual or perpetual). An annual license for the program which includes the Advanced+Syntax module used to illustrate the various types of IRT models costs US$695 for non-academic users, US$345 for academic users, US$100 for students, and US$25 for classroom use. A perpetual license for the same program costs US$1,595 for non-academic users and US$795 for academic users. Discounts are available for multiple licenses.