

WINMIRA 2001

(c) 2000, 2001 by Matthias von Davier

Table of Contents

WINMIRA user manual.....	1
The WINMIRA THANK YOU PAGE:.....	3
Preface.....	5
How to use WINMIRA?.....	7
How to do a Latent Class Analysis.....	8
How to analyse data with the Rasch Model.....	10
How to do a Mixed Rasch Model.....	12
How to analyse data with the Hybrid Model.....	14
Open data.....	17
Open SPSS Data Files.....	17
Import ASCII Data.....	18
Recode data with WINMIRA.....	20
How to edit variables with WINMIRA?.....	21
Select variables.....	22
Choose Number of Classes:.....	25

Table of Contents

Choosing Output Options:.....	26
Appending variables to the data:.....	28
Edit filenames:.....	29
Edit default values:.....	31
Testing the fit of a model with the Bootstrap:.....	33
Start Run:.....	35
Parameter constraints.....	37
Probability constraints.....	38
Logistic parameter constraints:.....	40
Class size constraints.....	41
The rationale of Mixture Distribution Models.....	43
Parameter estimation in MDM.....	45
The LCA for ordinal variables.....	47
The threshold approach in the ordinal LCA.....	48
The threshold approach in the LCA (cont.):.....	51

Table of Contents

What does the LCA–part of the program?.....	55
What can LCA be used for?.....	56
The Mixed Rasch Model.....	61
The dichotomous Mixed Rasch model.....	62
The polytomous Mixed Rasch model.....	63
The Rating Scale Model.....	65
The Equidistance Model.....	66
The Dispersion Model	68
The Ordinal (Partial Credit) Model	69
Person Parameters.....	70
Latent Score Distributions.....	72
Category Characteristic Curves.....	74
The Q–Index.....	75
The Hybrid Model.....	79
Using the output of WINMIRA.....	81

Table of Contents

Printing the output.....	81
Display Graphical Output.....	82
Example Output file.....	85
References:.....	109

WINMIRA user manual

– by Matthias von Davier – e-mail: winmira@von-davier.de

WINMIRA is a software for analyses with a variety of discrete mixture distribution models for dichotomous and polytomous categorical data. This software can be used for

the Rasch Model,

the Latent Class Analysis,

the Mixed Rasch Model,

and the Hybrid Model.

The WINMIRA THANK YOU PAGE:

I have maintained this software for almost 10 years now and I would like to thank the following people for their support not only during the revision of the current version of the software and the manual:

Jürgen Rost & Kentaro Yamamoto

and

Rolf Langeheine, Ivo Molenaar, Thorsten Meiser, Knud Sievers, Olaf Köller, Claus Carstensen...

and all other colleagues and users of the software that came up will valuable hints and recommendations.

Many thanks also to my family *Alina & Thomas von Davier* for their patience and support.

Thanks also to the component writers of TRegisterApp, TxyGraph, HugeArray

and to Borland Inprise and SPSS

Preface

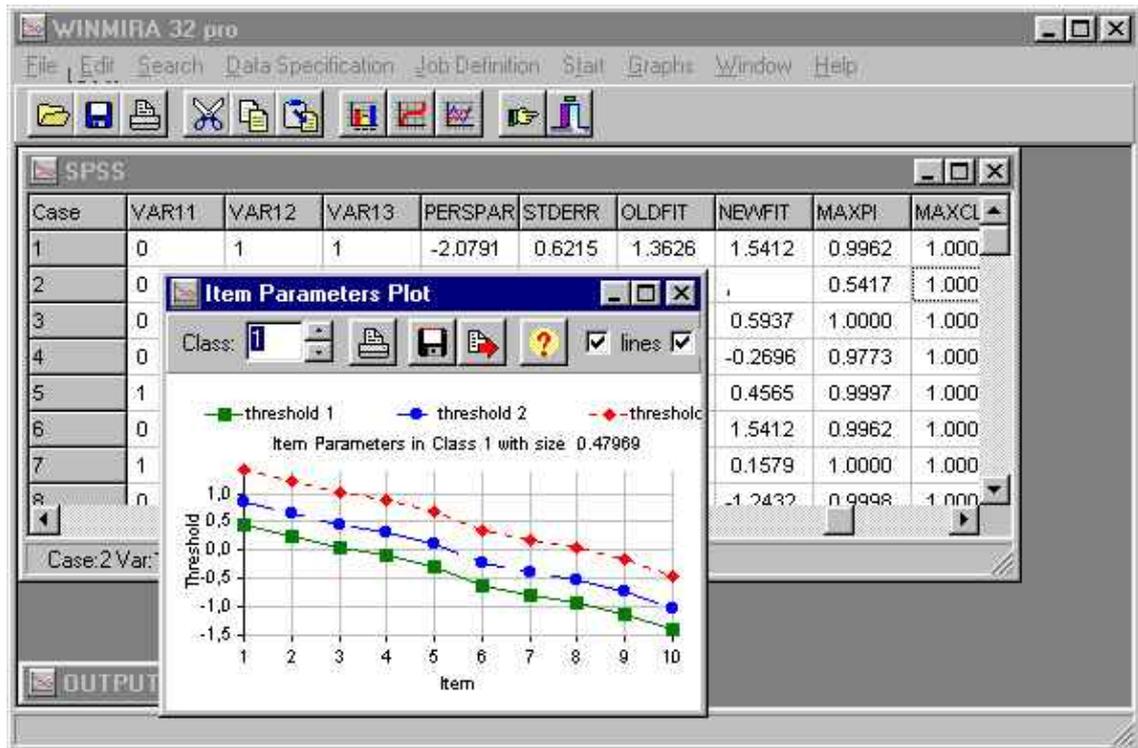
The development of WINMIRA was aimed at producing an easy to use software tool for categorical data analysis with a variety of models including the Latent Class Analysis and the Mixed Rasch Model. Even though the user interface is more or less self explaining, there will still remain difficulties in using this software. Some of the models which can be estimated with WINMIRA are comparably complex, and therefore, quite a few selections have to be made. As this software is thought to be a scientific tool for data analysis, I tried not to restrict the use of the software in order to make it even more easy to use. This lack of control puts full responsibility in the hand of the user! It is especially important to be sure about which model has been specified and estimated, as the lack of input restrictions makes it even more important to examine the results of the analyses very carefully. For example, please consider a model with too many classes and only a few items. The algorithm may still run, though the final solution is not identified and there may be some classes with a class size near to zero. In addition, some parameters may diverge to minus or plus infinity so that some categories in the respective latent class have expected frequencies close to zero. Perhaps, most of these cases could be prevented by the software, but that would mean to have a program with many warnings in the output and nasty message boxes during runtime.

Therefore, it can not be guaranteed that the software fulfills all requirements of the users (see the license agreement), and responsibility has to be taken by the users to interpret and examine the output carefully. Support in using the software can be given, of course, e.g. by supplying references or helping to interpret some pieces of output. Problems with the software and ideas for improving it should be reported directly to my e-mail address:

winmira@von-davier.de

Of course, not everything can be taken into account, but in the past I've been able to improve the software mainly because there have been users of previous versions sending very helpful comments.

How to use WINMIRA?



This HTML coursework provides cookbook examples of how to use the software WINMIRA 32. It will not tell much about the mathematical background of the models implemented in WINMIRA. Please refer to the references part of this hypertext in order to find out more about the math. These pages are intended to provide a quick reference for enabling novice users to start using the software without a human tutor.

Contents:

How to analyse data with...

- the Latent Class Analysis
- the Rasch Model
- the Mixed Rasch Model
- the Hybrid Model

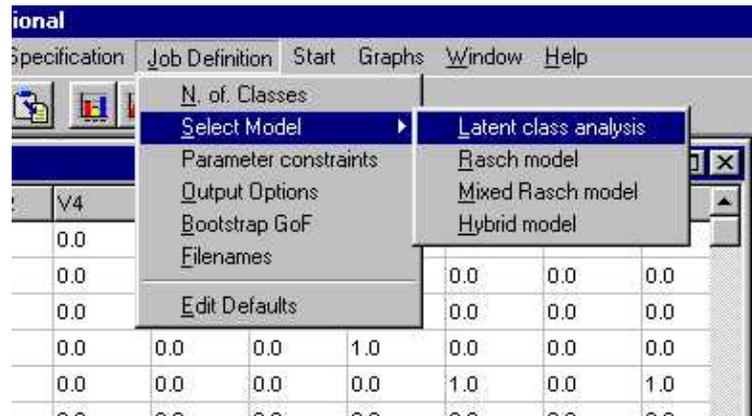
WINMIRA related WWW Links

- Winmira Homepage
 - ProGAMMA
 - ASC
-

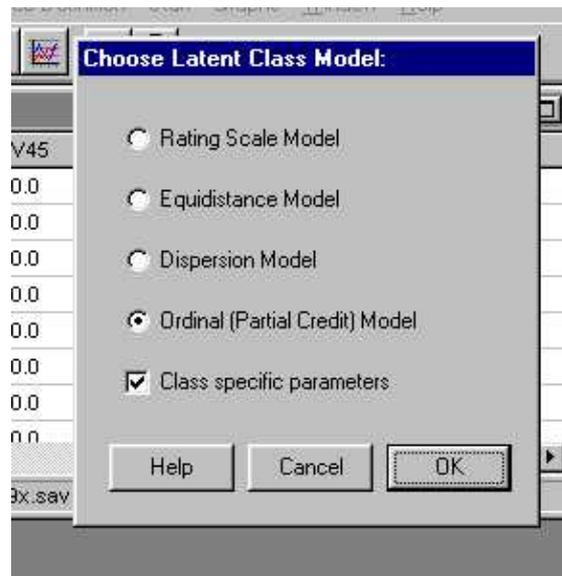
How to do a Latent Class Analysis

- Open Ascii or SPSS Data Files
- Recode Data if necessary
- Select Variables
- Choose the number of latent classes

Choose the Latent class analysis from the Job Definition > Select Model submenu as shown in the following screenshot.



This will open the model selection dialog for the Latent Class Analysis, see below.



If you use WINMIRA for the first time, please do not change anything.

Clicking OK with the default settings will choose the most general LCA model which is suitable both for dichotomous and polytomous data. The remaining options impose restrictions on the model parameters of the LCA. More information on restricted models can also be found in the references.

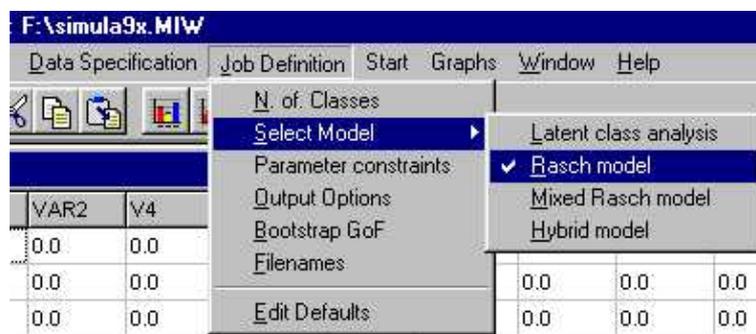
- Change Output Options if necessary,
 - choose Bootstrap Fit Statistics if your data are sparse,
 - Start Run,
 - Display Graphical Output
-

How to analyse data with the Rasch Model

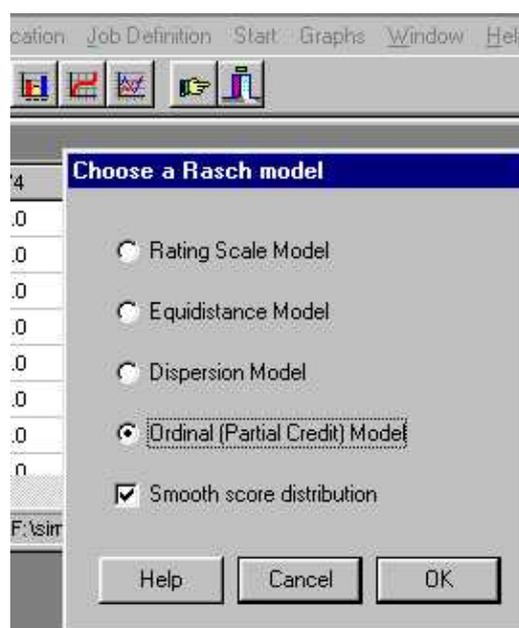
- Open Ascii or SPSS Data Files
- Recode Data if necessary
- Select Variables
- **Do not** choose the number of latent classes! (not necessary for the ordinary Rasch model, this model assumes that the same parameters hold for the whole population.).

Choose the Rasch model from the Job Definition>Select Model submenu

as shown in the following screenshot.



This will open the model selection dialog for the Rasch Model and the Mixed Rasch Model, see below.



If you use WINMIRA for the first time, please do not change anything. Clicking OK with the default settings will choose the most common Rasch model which is suitable both for dichotomous and polytomous data. The

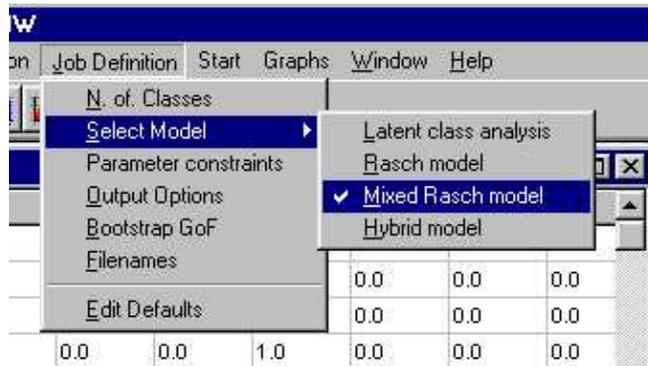
remaining options impose restrictions on the model parameters of the model. More information on restricted Rasch models can also be found in the references.

- Change Output Options if necessary,
 - choose Bootstrap Fit Statistics if your data are sparse,
 - Start Run,
 - Display Graphical Output
-

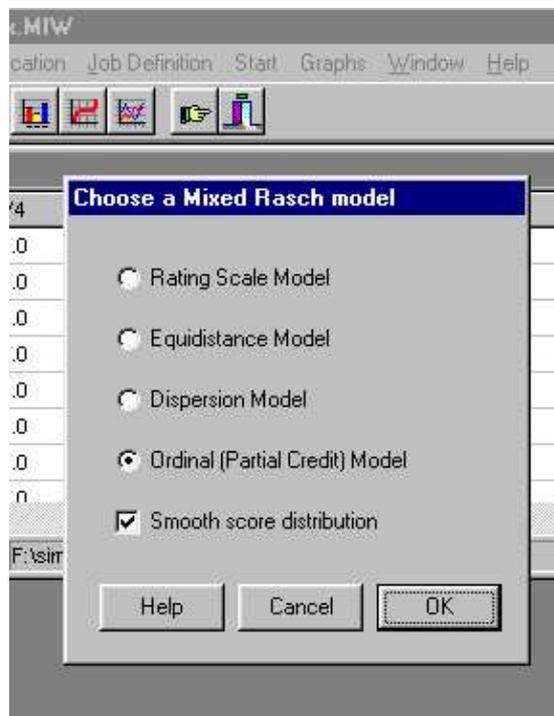
How to do a Mixed Rasch Model

- Open Ascii or SPSS Data Files
- Recode Data if necessary
- Select Variables
- Choose the number of latent classes

Choose the Mixed Rasch Model from the Job Definition>Select Model submenu as shown in the following screenshot.



This will open the model selection dialog for the Mixed Rasch Model, see below.



If you use WINMIRA for the first time, please do not change anything. Clicking OK with the default settings will choose the most common Mixed Rasch Model which is suitable both for dichotomous and polytomous data.

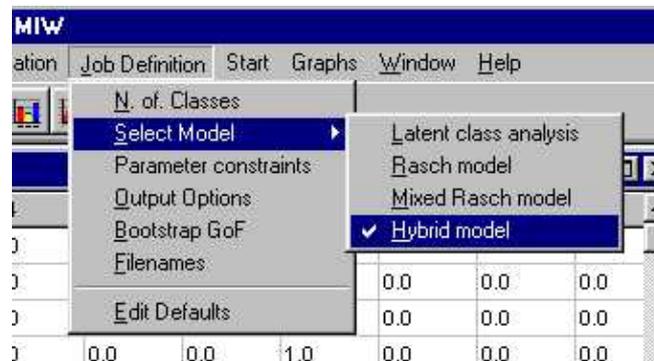
The remaining options impose restrictions on the model parameters of the Mixed Rasch Model. More information on restricted Rasch models can also be found in the references.

- Change Output Options if necessary,
 - choose Bootstrap Fit Statistics if your data are sparse,
 - Start Run,
 - Display Graphical Output
-

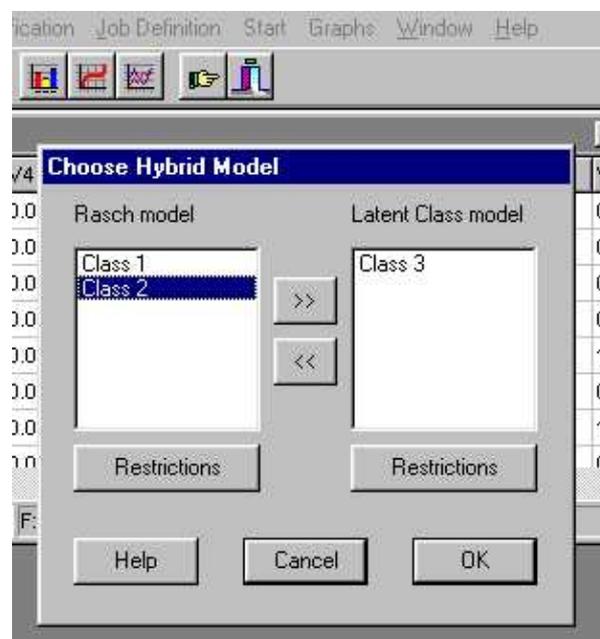
How to analyse data with the Hybrid Model

- Open Ascii or SPSS Data Files
- Recode Data if necessary
- Select Variables
- Choose the number of latent classes

Choose the Hybrid model from the Job Definition > Select Model submenu as shown in the following screenshot.

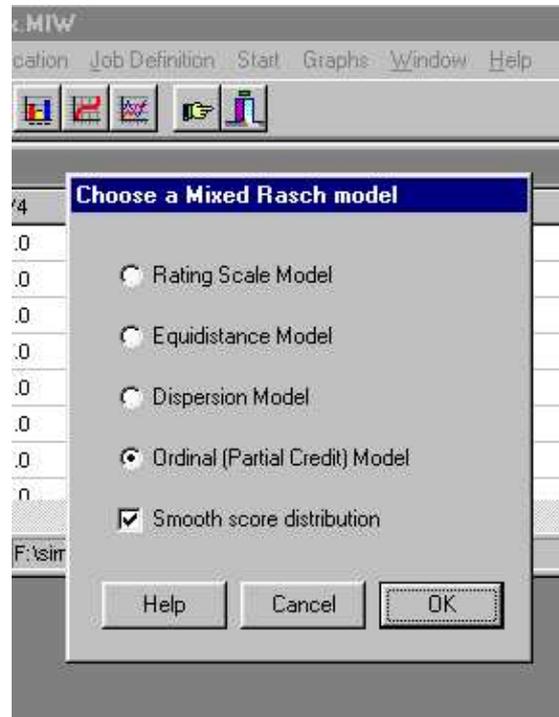


This will open the model specification dialog for the Hybrid Model as depicted below:



Please click on the Class 'n' lines for choosing one of the available Rasch or Latent Class Submodels. A doubleclick on the line 'Class 2' as shown above will open the following dialog and lets you choose between any of the available models, like it can be done for the Rasch Model or the Mixed

Rasch Model.



In contrast to the Mixed Rasch Model or the Latent Class Analysis, you can specify a different model within each of the classes, so that there are no similarity restrictions between the classes despite the fact that all models assume a logistic distributional function as their basis.

If you use WINMIRA for the first time, please do not change anything. Clicking OK with the default settings will choose the most common Mixed Rasch Model which is suitable both for dichotomous and polytomous data. The remaining options impose restrictions on the model parameters of the Mixed Rasch Model. More information on restricted Rasch models can

also be found in the references.

- Change Output Options if necessary,
 - choose Bootstrap Fit Statistics if your data are sparse,
 - Start Run,
 - Display Graphical Output
-

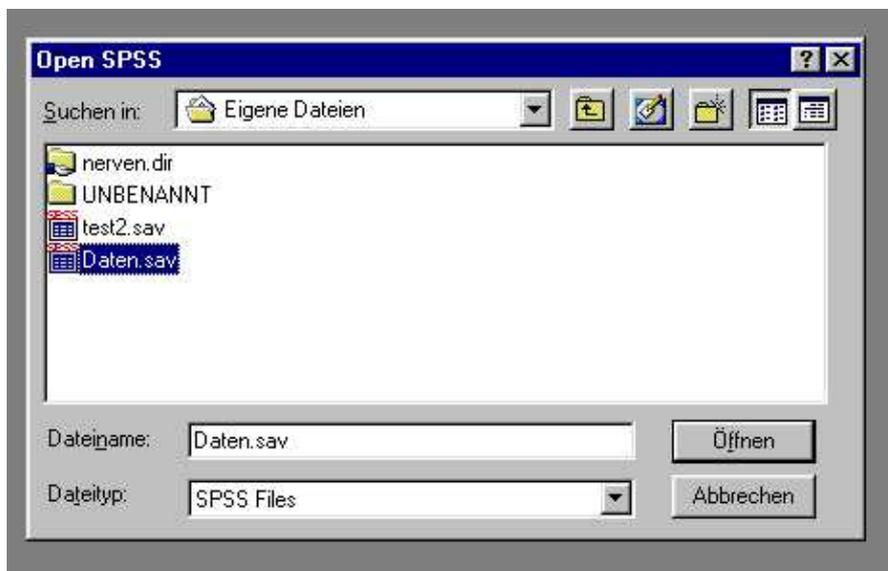
Open data

Open SPSS Data Files

Please select the menu entry **File > Open > Open Spss Data** as shown in the screenshot below:



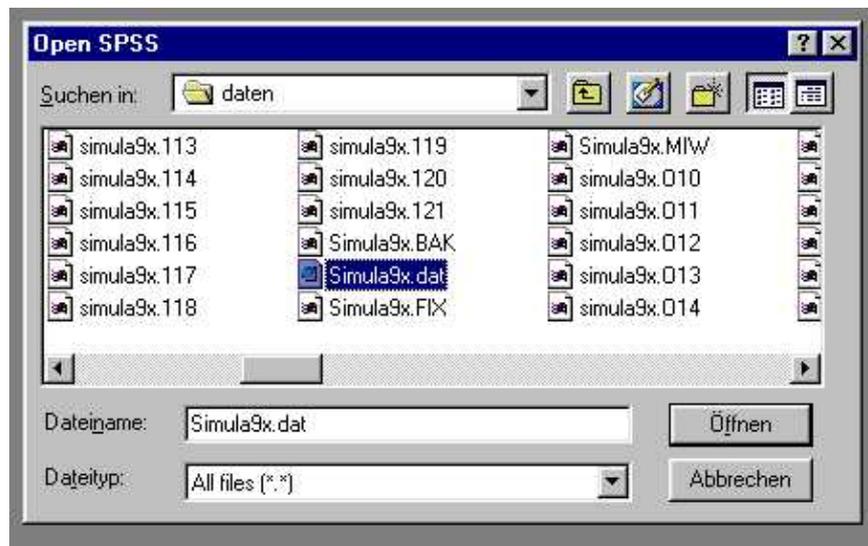
Select the data file in the file selection dialog:



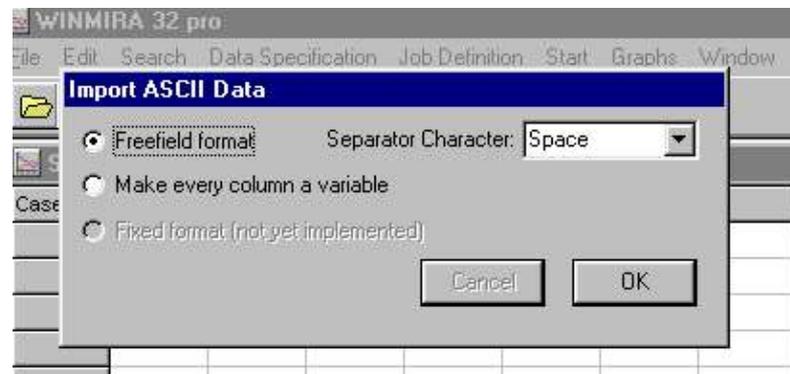
In this example, the file named "Daten.sav" was chosen. Click the OK button in order to confirm the selection. This will load the datafile.

Import ASCII Data

If you choose a file with an extension different from ".SAV", it is assumed that ASCII data are imported. Alternatively, you may also choose 'Load ASCII Data' from the submenu of the **F**ile > **O**pen menu entry.



Clicking the OK button will open another window which asks for information needed to import ASCII data.



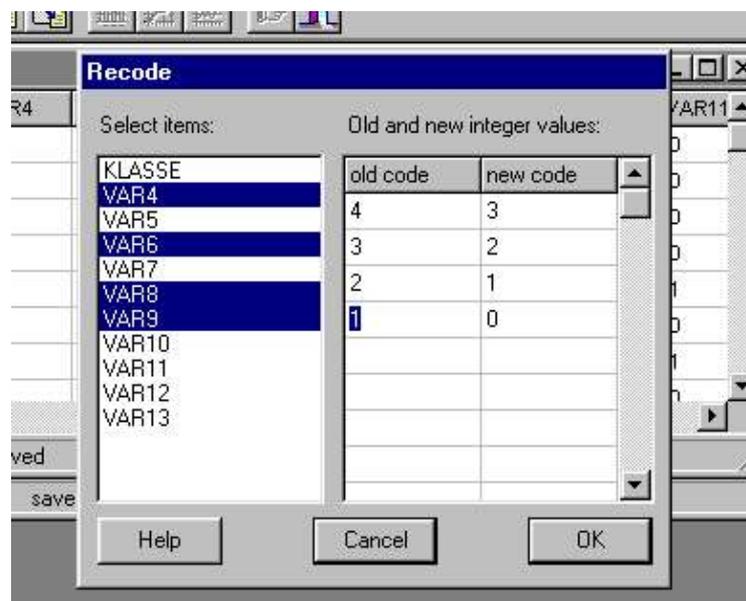
Please select the separator character (in most cases, this will be the space or the tab character) and click OK in order to import ASCII data.

Recode data with WINMIRA

please choose the Data Specification > Recodings menu entry as depicted in the following figure:



This will open the Recode dialog, which contains a list of all variables in the dataset.



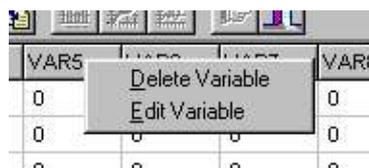
In this example the variables VAR4, VAR6, VAR8 and VAR9 are chosen

to be recoded. Please enter all values to be recoded together with the new codes in the grid on the right hand side of the dialog. In order to start recoding, click on the OK button. The data will be then recoded into the same variables.

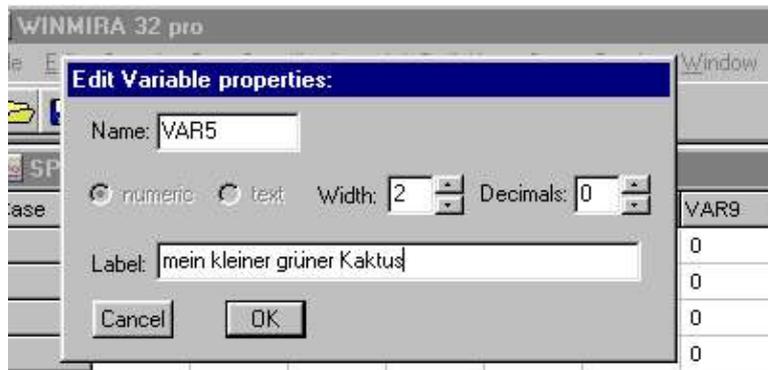
A good idea is to rename the variables that have been recoded. This can be done with the edit variable dialog.

How to edit variables with WINMIRA?

by doubleclicking on the first row in the data file (the fixed row where the variable labels are), the variable menu is activated.



Please choose Ddelete Variable if you need to remove variables from the dataset or click on Edit Variable in order to open the following dialog:



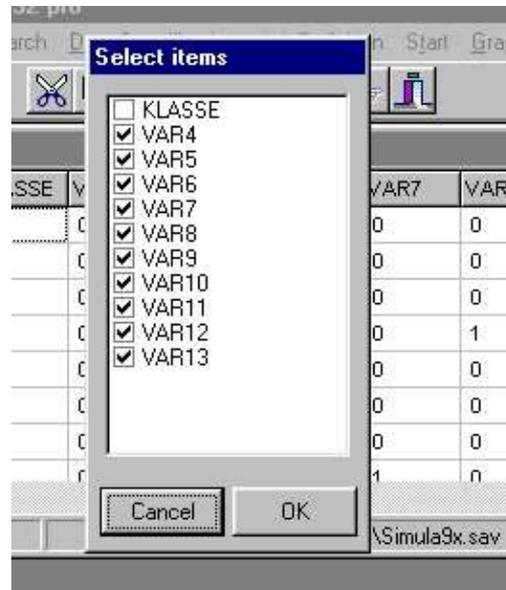
In the example dialog above, the variables name is 'VAR5' and the label is 'mein kleiner grüner Kaktus' (my small green succulent? plant). You may change the variable's name, label and output format, i.e., it's printed width and decimals.

Select variables

Click on the menu entry Data specification > Select Variables as seen below:

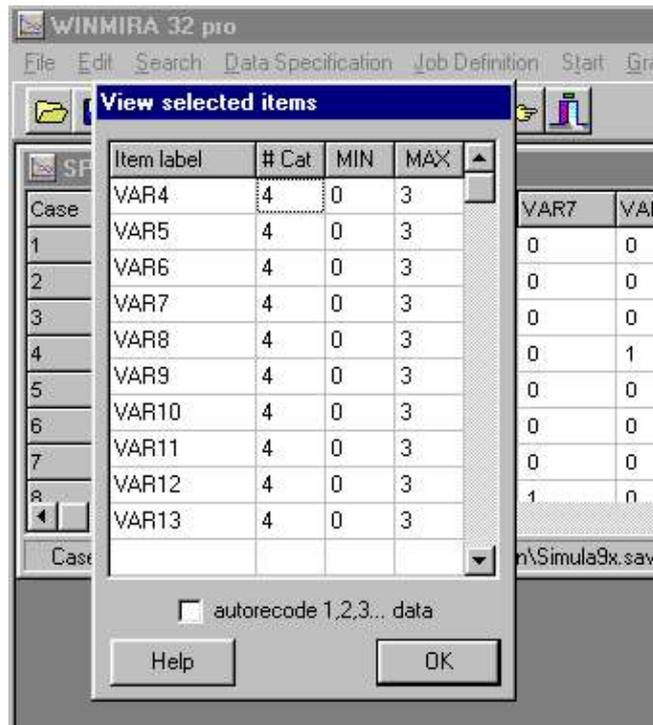


Indicate which items of the current dataset are to be included in the analysis. Please do not choose all items, think of which items represent the construct you want to measure and select the items accordingly.



After clicking the ok button, the selected items will be analysed in order to obtain the number of categories and the missing values for each item, and in addition, whether all categories have at least been chosen once. If the category codes do not start from 0 [zero], as it is required by the algorithms of WINMIRA, the items will be 'autorecoded', i.e., the minimum code (say 1 [one] instead of 0 [zero]) will be subtracted from all observed values. Below, the 'view selected items' dialog is depicted. This dialog shows the selected variables together with their number of categories as extracted from the data by the testing algorithm, together

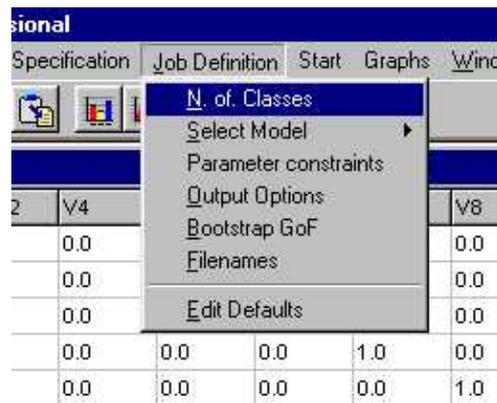
with the minimum and maximum code. If the minimum code is larger than 0 (zero), autorecode will be enabled in the dialog automatically.



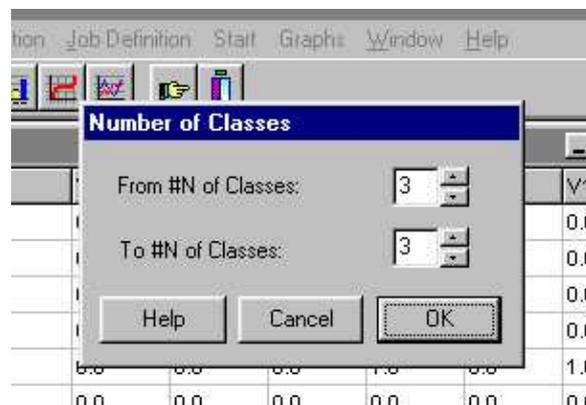
Please click the OK button in order to confirm the selection. This will enable WINMIRA 32's other menus that have been disabled before. Now the model selection and other more detailed specifications of the model to be estimated are made available, as the data specification is completed by accepting the selection in this dialog.

Choose Number of Classes:

Please choose the Job Definition > N. of Classes menu entry like shown in the figure below:



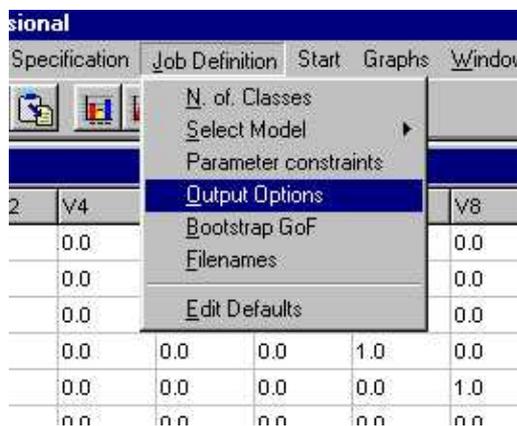
This will open the Number of Classes Dialog, which allows to specify a upper and lower bounds for the number of classes to be computed during the computation with WINMIRA.



In this example, only one model with three latent classes will be computed. Increasing the number in the 'To #N of Classes' field to 5 will make WINMIRA 32 to compute three models with three, four and five latent classes. Decreasing the 'From #N of Classes' field to 2 will make WINMIRA to compute both the two and the three class solution for the specified model.

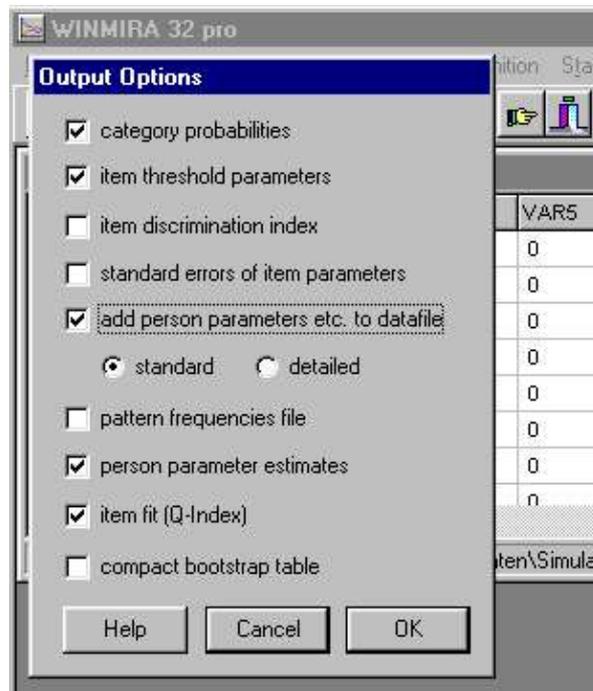
Choosing Output Options:

Please choose the Job Definition > Output Options menu entry like shown in the figure below:



This will open the Output Options Dialog, which allows to specify a number of tables to be generated by WINMIRA and additional information

to be written in additional files or to be appended to the dataset.



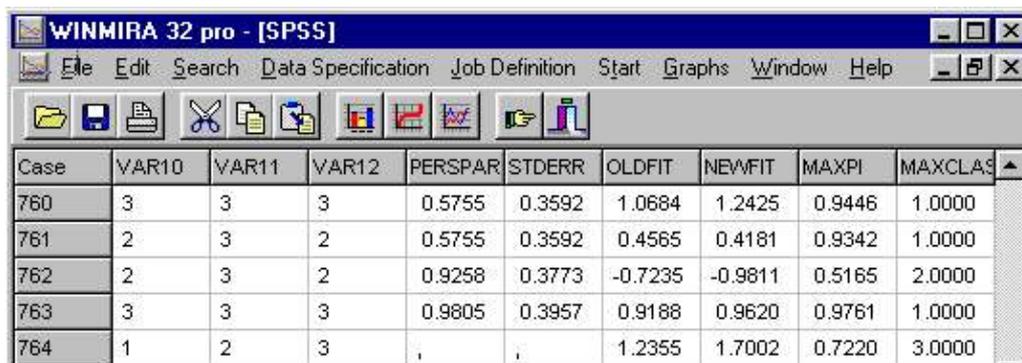
As a default, both the category probabilities and the item parameter estimates for each latent class are written in tabular form to the output file. For Rasch Models and Mixed Rasch Models (and also for the Rasch Classes in Hybrid models), person parameter estimates and the item fit measure Q–Index are printed out in addition to the other two default tables.

In this example, 'add person parameters etc. to datafile' was chosen in addition to the default output options. This will generate additional variables for each person (case) in the dataset. These additional

variables contain information about the most probable class, the personparameter estimate and person fit measures.

Appending variables to the data:

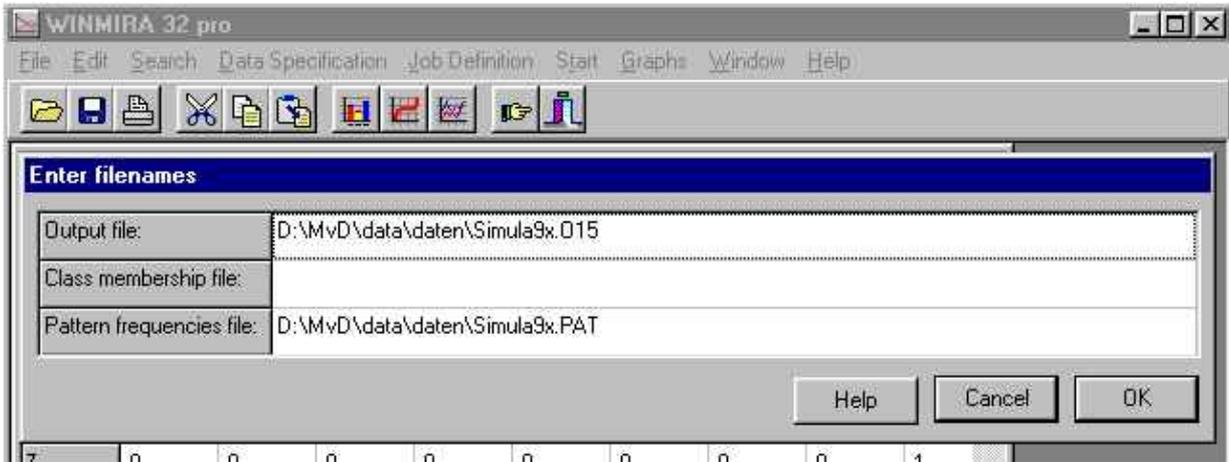
When 'append person parameters etc. to datafile' was chosen in the Output Options menu, a number of variables will be added to the datafile that was used for estimating the model. The picture below shows the realisations of these variables for the cases 760 to 764 in our example data.



Case	VAR10	VAR11	VAR12	PERSPAR	STDERR	OLDFIT	NEWFIT	MAXPI	MAXCLASS
760	3	3	3	0.5755	0.3592	1.0684	1.2425	0.9446	1.0000
761	2	3	2	0.5755	0.3592	0.4565	0.4181	0.9342	1.0000
762	2	3	2	0.9258	0.3773	-0.7235	-0.9811	0.5165	2.0000
763	3	3	3	0.9805	0.3957	0.9188	0.9620	0.9761	1.0000
764	1	2	3	,	,	1.2355	1.7002	0.7220	3.0000

The variable 'PERSPAR' contains the trait estimate, 'STDERR' contains the standard error of estimation for the sufficient statistic. 'MAXPI' contains the maximum of the posterior probabilities of being member of one of the latent classes given the observed response vector of the respective person. 'MAXCLASS' is the latent class carrying the maximum

of the "new" output file will be modified in the following way:



If <filename.OUT> is already existing, the "new" output will be renamed to <filename.OU1>. If <filename.OU1> exists, then the current will be in <filename.OU2> and so on.

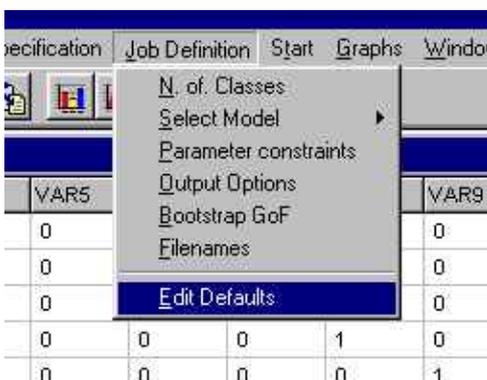
The pattern–frequency–file's name is not modified automatically, as usually only one pattern–frequency (the one according to the finally selected model) is required. Please modify the pattern–frequency–file name manually if more than one pattern–frequency file is required for additional analyses.

The class membership filename is now obsolete, as all person–related statistics can be appended directly to the datafile in the WINMIRA 32 pro version of the software.

If WINMIRA 32 refuses to start the analysis it is good advice to check whether the destination path is existing, especially if data and definition files for running WINMIRA 32 are moved from one computer to another.

Edit default values:

Please select the menu entry Job Definition > Edit Defaults as shown in the screenshot below:



This will open the edit defaults dialog box as shown below. Please be careful when modifying any of the values in this dialog box, as some very important basic parameters of WINMIRA 32 can be modified here. Please examine your output-files carefully if you have changed any of these values.



The *maximum number of iterations* limits the process of parameter estimation by default to a maximum of 500 iterations. Please examine the output–file to find out whether this number was reached. Typically this happens when the *accuracy criterion* was not met, i.e. the difference of the log–likelihood between two subsequent iterations was greater than the criterion. In these cases, the maximum number of iteration should be increased.

The *random start value* is used in some of the random number generators for the initial response pattern splitting in the EM algorithm. Please change this value whenever you think the results obtained by WINMIRA could be a local maximum (mainly due to small sample size).

The *number of start values* is used for the initial search for the (relatively)

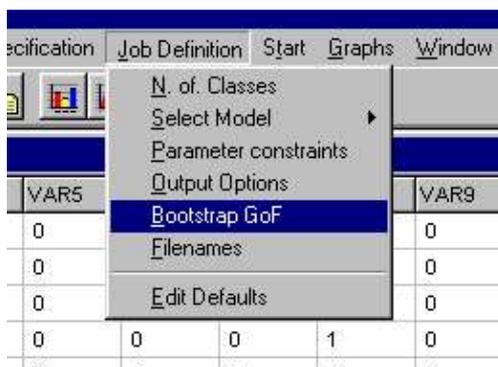
best starting point in the parameter space. Increase this number if you suspect that WINMIRA 32 might get stuck in local maxima.

The *sort output by class size checkbox* allows to choose between an unsorted output file, where classes are assigned by chance or by the within class restrictions (like in Hybrid models, or if models with unique parameter constraints imply the order of the classes), and an output sorted by class sizes (whenever all mixture components carry the same structure like in the ordinary LCA or the ordinary mixed Rasch model).

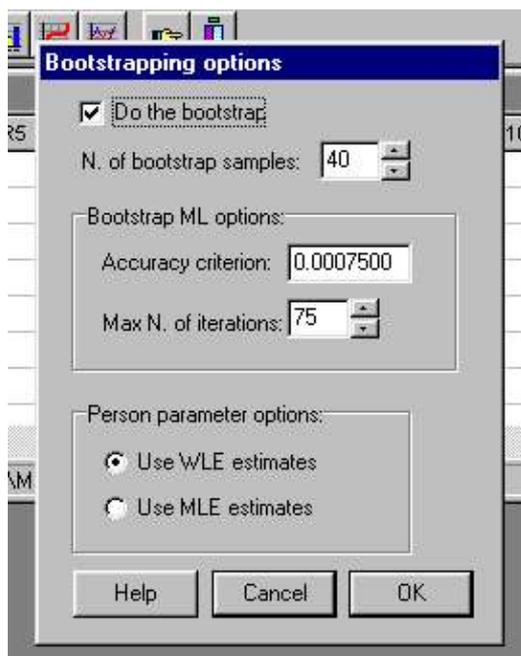
The step–width for minimization is used in some of the estimation procedures, please do not change this value, because it is the result of a lot of numerical fine tuning experiments.

Testing the fit of a model with the Bootstrap:

Please choose the Job Definition > Bootstrap GoF menu entry like shown in the figure below:



This will open the Bootstrap Dialog, which allows to specify the number of simulations and other parameters in order to perform the parametric bootstrap test for Chisquared Goodness of Fit statistics.



The bootstrap will be performed for four different Chisquared Goodness-of-Fit statistics, namely the Pearson X^2 , the Cressie-Read statistic, the Likelihood-Ratio and the Freeman-Tukey statistic. PLEASE

NOTE that only the Pearson X^2 and the Cressie–Read statistic work well when the data are extremely sparse!

In this example, 40 bootstrap samples are generated. The number of iterations starting from the parameter estimates of the real data are only 75 in this example, as the original data's estimates are extremely good starting values for the simulated dataset, which was generated using exactly these parameters. The data are generated based on CML estimates (Conditional–Maximum–Likelihood), so that the WLE – person parameters are only used for the extreme rawscore groups.

Start Run:

Choose Start > Start Job (or function key F9) in order to make WINMIRA start computations with the current settings.



This will minimize WINMIRA's main window and will open the run dialog, which is used to display some information about the computational process.

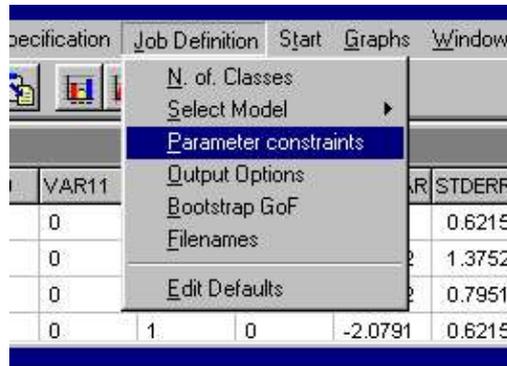


Finally, the main window of WINMIRA will reappear and the programs output file will be shown in a second child window within the main form.

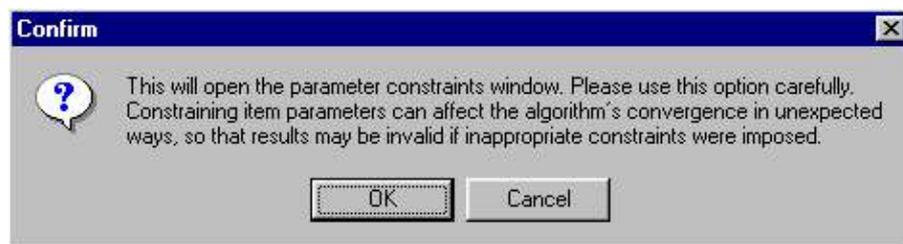
The output file shown in the active child window contains descriptive statistics of the dataset, model parameters estimates and information about the model data fit. This file is already saved, usually in the same directory where the dataset is found, unless some other directory and filename was specified in the filename dialog.

Parameter constraints

The Job Definition > Parameter constraints menu entry



opens a confirmation message window in order to make sure that parameter constraints should be used by expert users only and all computations carried out with parameter constraints should be examined carefully:



Winmira offers three types of parameter constraints:

- either category probability constraints,
 - or constrain item parameter and threshold distances,
 - where both can have simultaneous constraints imposed on class sizes
-

Probability constraints

WINMIRA 32 pro provides the means to impose both parameter fixations and equality constraints on the class specific response probabilities. The picture below shows the different ways to enter fixations or equality constraints in WINMIRA 32 (even though the constraints seen below might not make too much sense, the primary reason for entering the numbers and letters in this way is to show what can be done in one single screen shot).

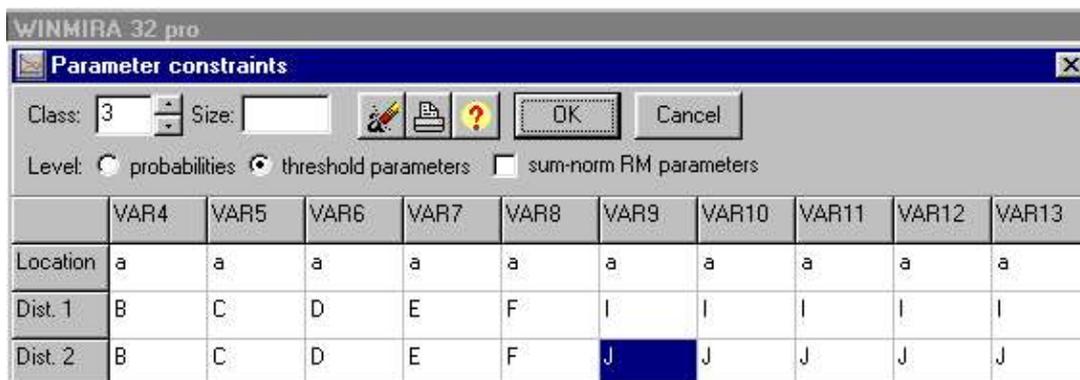
	VAR4	VAR5	VAR6	VAR7	VAR8	VAR9	VAR10	VAR11	VAR12	VAR13
Option 0	0.25000	a	B	C	a	0.25000	0.25000	B	a	C
Option 1	0.25000	a	B	C	a	a	0.25000	B	a	C
Option 2	0.25000	a	B	C	a	a	0.25000	B	a	B
Option 3	0.25000	a	B	C	a	a	a	a	C	B

Parameter fixations are entered as numerical values between 0.000 and 1.000 in the constraints dialog. Equality constraints are one letter entries in the constraints dialog. All cells, or better the corresponding class specific response probabilities, carrying the same letter (CaSE SenSitiVE) will be set to the same numerical value in each iteration of the estimation algorithm. Nevertheless, certain regularity conditions are imposed afterwards. If you try to constrain two items of extremely different difficulty to carry the same response probabilities, the algorithm will nevertheless assume that the overall response probabilities are like given by the dataset and will adapt the equality constraints to match the overall probabilities.

Please click through the classes by means of the Class up down arrow on the top left of the window panel to see all constraints. Constraints can be made within and across classes.

Logistic parameter constraints:

WINMIRA 32 pro provides the means to impose both parameter fixations and equality constraints on item difficulties as well as on threshold distances when estimating polytomous models. The picture below shows the constraint dialog when using it for these logistic constraints.



Parameter fixations are entered as numerical values (both negative and positive, in contrast to probability constraints) in the constraints dialog. Equality constraints are one letter entries in the constraints dialog. All cells, or better the corresponding class specific response probabilities, carrying the same letter (CaSE SenSitiVE) will be set to the same numerical value in each iteration of the estimation algorithm. The example shows that all item locations are equal (all cells show an 'a') in class 3. In

addition, the threshold distances are also carrying constraints. Items VAR9 to VAR13 are constraints analog to the Rating scale model, whereas items VAR4 to VAR8 carry constraints like in the equidistance model. This shows that constraining parameters can be more flexible than using the models 'hard-wired' in WINMIRA. Nevertheless, be aware that using constraints can also mean that one specifies a model that cannot be estimated, or that at least will slow down or disturb the convergence of the algorithm .

Please click through the classes by means of the Class up down arrow on the top left of the window panel to see all constraints. Constraints can be made within and across classes.

Class size constraints

The types of parameter constraints, probability constraints and logistic item and threshold parameter constraints, cannot be mixed. Nevertheless, both types allow constraining the class sizes to constants like it is shown in the following picture.



In this example, the size of class 3 was fixed at 20%, or 0.200. Equality constraints are also possible. Please make sure that class sizes have to add up to 1.000. Please click through the classes by means of the Class up down arrow on the top left of the window panel to see all constraints. Constraints can be made within and across classes.

The rationale of Mixture Distribution Models

Mixture distribution models (MDM) relax the assumption that the observed data were drawn from a homogeneous population. It is rather assumed that the sample is drawn from an unknown mixture of distributions (Everitt & Hand, 1981), which are referred to as latent classes in this context. Mixture distribution models are more flexible as compared to classical statistical modeling, where we usually apply a statistical model to a set of data and assume that the model is valid for the data and that all model parameters (e.g. factor loadings, path coefficients, item parameters) are the same for all individuals of the population. Discrete MDM, in contrast, are based on the idea that different sets of model parameters are valid for different subpopulations. In the case of latent subpopulations, their number is not known but must be identified when the model is applied. These subpopulations are solely defined by their property of being homogeneous in the sense that a particular model holds for this latent class. In particular, latent classes are not defined by manifest variables like gender, age or socio-economic status, where the partition is done by a manifest observable moderator variable and parameter estimation can be performed for each manifest group. Hence the aim of MDM is twofold: to 'unmix' the data into homogeneous subpopulations and to estimate the parameters for each subpopulation separately. The general structure of

discrete MDM is

$$P(X = x) = \sum_{g=1}^G \pi_g P(X = x | \theta_g) \quad (1)$$

where $x = (x^1, \dots, x^k)$ is a vector valued observation on k variables (items), and θ_g is the group (class) specific vector valued parameter of the conditional distribution in class g . It is assumed that the overall probability of an observation x is a weighted sum of conditional probabilities within these subpopulations. The weights π_g are the mixing proportions (which are often referred to as class sizes) and represent the relative sizes of the subpopulations. The family of discrete MDM can be divided into at least two groups of models. Firstly, MDM which assume the same type of model in all subpopulations but with different sets of model parameters (these models are commonly referred to as MDM). Secondly, discrete MDM which can be defined with a different type of model in each subpopulation. WINMIRA 32 can handle both types of MDM, namely any possible combination of the class specific models available in the program can be specified and analyzed with WINMIRA 32, as long as there are no identification problems. The section on What can LCA be used for gives examples illustrating the capabilities of the mixture distribution models.

Parameter estimation in MDM

Parameter estimation in discrete Mixture Distribution Models is quite complicated and time consuming. An iterative algorithm, the EM–algorithm (E–stimation / M–aximization or *iterative proportional fitting*), has to be employed, because the latent classes are not known beforehand. The EM–algorithm works as follows:

- Within each E–step, the expected frequencies of the sufficient statistics for the model parameters are computed for each subpopulation. This is usually done by computing posterior probabilities given the current parameter estimates.
- Within each M–step, Maximum Likelihood estimates in each subpopulation are computed (by means of some standard procedure like Newton–Raphson) given the sufficient statistics from the previous E–step.

The iteration procedure is based on the first and second order partial derivatives of the likelihood function L of the "complete" crosstable, i.e.

the observed crosstable extended by the latent class variable. From the matrix of the second order derivatives, only the diagonal elements are used so that the iteration rule for a model parameter μ is:

$$\mu^{t+1} = \mu^t - [(L'(\mu))/(L''(\mu))],$$

where $L'(\mu)$ is the first partial derivative, $L''(\mu)$ the second order derivative with respect to the parameter μ , and μ^t the estimate of μ in iteration step t . The number of iterations is restricted to 10 within each M -step for the Latent Class models and 1 for the Mixed Rasch models. The parameter estimates of the previous M -step serve as start values for succeeding M -steps.

The LCA for ordinal variables

WINMIRA 32 is capable of estimating the parameters of 8 different latent class models for manifest variables with ordered categories. Four of these models assume class specific and four models assume class independent distances between response categories. The models in each of the two groups result from the assumption of :

- ordered categories only,
- equidistant categories for each variable,
- equal distances between categories for all variables,
- scaled distances between categories but different dispersions for each variable.

WINMIRA 32 provides

- parameter estimates for all models,
- response probabilities,
- various goodness of fit statistics (Likelihood Ratio, Cressie–Read, Pearson X^2 and Freeman–Tukey)
- capabilities for performing bootstrap (or monte–carlo) tests for these statistics,

- information criteria (AIC, BIC, CAIC),
- output of most likely class membership for each person.

The threshold approach in the ordinal LCA

The 8 LCA models covered by LACORD (LAtent Class analysis for ORDinal variables) which are included in WINMIRA 32 can be derived from a very simple assumption regarding the ratio of response probabilities of each two adjacent categories,

$$(p^{vix}) / (p^{vi(x-1)}), \text{ for } x=1, \dots, m,$$

where p^{vix} is the probability of person v for scoring in category x on item i . As in log-linear models, it is assumed that the logarithm of this ratio is a linear function of some parameters depending on the variable i , the category x and the latent class g to which a person v belongs. Without further specification of that linear function f_{gix} , the assumption may be written as

$$\ln([(p^{vix}) / (p^{vi(x-1)})]) = f_{gix}, \text{ for } x \text{ in } (1, \dots, m) \quad (10)$$

This simple approach leads to a very handy model structure, because it can be derived without further assumptions that the category or response probabilities in a latent class g are:

$$p^{gix} = [(\exp(\sum_{s=0}^{x_s} f_{gis})) / (\sum_{t=0}^m \exp(\sum_{s=0}^t f_{gis}))] \quad (11)$$

Although this is a very straightforward model derivation, the parameter interpretation remains somewhat diffuse. All that is known about the linear components of f_{gix} (which have not been specified yet) is that the response probability of a category x in relation to its preceding category increases monotonously with f_{gix} . A very convincing interpretation of the model parameters is found with the transformation of model assumption (equation 10 above) into its equivalent form

$$[(p^{vix}) / (p^{vix} + p^{vi(x-1)})] = [(\exp(f_{gix})) / (1 + \exp(f_{gix}))] \quad (12)$$

The ratio defined above is referred to as a threshold probability, because it is the conditional probability of choosing x , if only x or $x-1$ are considered. This can be interpreted as passing the threshold from the lower category to reach the higher category.

The term on the left hand side varies between 0 and 1, and denotes a

probability, namely the conditional probability of responding in category x , if the response is either in x or in $x-1$. This conditional probability is usually referred to as threshold probability, because it can be interpreted as the probability of passing a threshold between two response categories. It equals 0.5 if both categories have the same probability, 0.0 if nobody passes and 1.0 if all people pass the threshold to x . The latter form of equation 11 shows that the threshold probabilities depend on the model parameters f_{gix} and their dependency is defined by the smooth logistic curve known from the Rasch model. Hence, the model parameters f_{gix} can be interpreted as defining the location of threshold x on a latent continuum.

This was outlined in some detail in order to stress the point that a threshold assumption is not necessary to derive the model, but it helps to interpret the parameters. The model itself only requires knowledge of which categories are adjacent and, in that sense, requires ordered categories.

[more]

The threshold approach in the LCA (cont.):

If the class specific response probabilities (equation 11) are substituted for the category probabilities in the general latent class model,

$$P_{vix} = \sum_{g=1}^G \pi_g P_{gix}$$

where π_g is the relative size of class g , the model structure underlying all 8 latent class models of WINMIRA 32 is obtained. They only differ in their specification of f_{gix} , which is outlined in the following.

If there should be no restriction at all, f_{gix} itself may be considered a model parameter and nothing more than a reparameterization of the polytomous LC-model is obtained. In the ordinal model of the Latent Class module of WINMIRA 32, however, another reparameterization is used in order to make the results comparable with the other models:

$$(8) f_{gix} = \mu_{ig} + \lambda_{ixg} \text{ and } \sum_x \lambda_{ixg} = 0.$$

In this specification, μ_{ig} can be interpreted as the *mean location* of all thresholds of variable i in class g and, because of the normalizing condition, the λ_{ixg} parameterize the deviation of threshold x from that

mean. Model (4) follows from equating the threshold distances (not their mean location) over classes:

$$(4) f_{gix} = \mu_{ig} + \lambda_{ix} \text{ and } \sum_x \lambda_{ix} = 0$$

Both models (8) and (4) assume an order of the categories insofar as for model derivation it must be known which categories are adjacent (see the first formula in this chapter). Neither of the models require ordered categories in the sense that the thresholds are ordered, i.e. have decreasing (easiness-) parameters f_{gix} . Whether the latter is the case can be seen from the results of a data analysis. Models (1) and (5) result from models (4) and (8) by equating the threshold distances over the variables:

$$(5) f_{gix} = \mu_{ig} + \lambda_{xg} \text{ and } \sum_x \lambda_{xg} = 0,$$

$$(1) f_{gix} = \mu_{ig} + \lambda_x \text{ and } \sum_x \lambda_x = 0.$$

Models (2) and (6), however, are not so easily obtained, since simple equating over categories would make the index x disappear from the model equation. An appropriate coefficient is required, generating the individual threshold location by means of the mean location μ_{ig} and a distance parameter δ_{ig} . This coefficient is $(m+1-2x)$, in order to avoid non-integer

coefficients, only half the distance is parameterized, so that the models are

$$(2) fg_{ix} = \mu_{ig} + (m+1-2x)\delta_i$$

and for class specific distances

$$(6) fg_{ix} = \mu_{ig} + (m+1-2x)\delta_{ig}.$$

Models (3) and (7), finally, are a combination of these last models and will not be derived in detail here:

$$(3) fg_{ix} = \mu_{ig} + \lambda^x + (m+1-2x)\delta_i \text{ with } \sum_i \delta_i = 0$$

$$(7) fg_{ix} = \mu_{ig} + \lambda_{xg} + (m+1-2x)\delta_{ig} \text{ with } \sum_i \delta_{ig} = 0$$

The normalizing condition for the δ_{ig} -parameters is necessary, because the basic-threshold-distances for all variables are already defined through the λ_{xg} -parameters. The δ_{ig} 's parameterize the deviation of all distances of a particular variable from the mean distance. The latter parameters may be interpreted as dispersion parameters, since the dispersion of the probability

distribution of a variable depends on the widths of its threshold intervals: the smaller δ_{ig} is, the closer the thresholds and the greater the dispersion of that variable. This relationship is more easily understood if the opposite case is imagined: a large distance between the left and the right threshold of a category means that it is easy to get in but hard to get out" of this category. Hence, the probability distribution has a peak over this category and, therefore, a smaller dispersion in general.

So far all models of the Latent Class Module of WINMIRA 32 have been specified. It must be noted, however, that they are usually written in a slightly different way. This different notation is obtained if the f_{gix} -terms are cumulated, as indicated in equation 1. Since

$$\sum_{s=1}^x \mu_{ig} = x\mu_{ig} \text{ and } \sum_{s=1}^x (m+1-2s) = x(m-x)$$

model (2) may be written as

$$(2') p^{vix} = \sum_g \pi_g [(\exp(x\mu_{ig} + x(m-x)\delta_i)) / (\sum_x \exp(x\mu_{ig} + x(m-x)\delta_i))] .$$

In order to get rid of the summation symbol when the f_{gix} -parameters have to be summed up, cumulative threshold parameters

$$a_{gix} = \sum_{s=1}^K f_{gis}$$

may be defined. This notation is helpful for deriving equations for parameter estimation, but they have no direct interpretation. The parameters provided in the output file and in the graphical display of the item parameters are f_{gix} -parameters as introduced in the formulae above.

What does the LCA-part of the program?

Parameter estimation and related computations can only be made for a fixed number of latent classes, since the number of classes in LCA is not a model parameter but an a-priori model assumption.

Starting values for parameter estimation are generated by a random number generator. The program can estimate the parameters for the 8 different latent class models for ordinal variables by using an extended EM-algorithm. The extension consists of a short Newton-algorithm within each M-step for maximizing the likelihood function.

Some models cannot be applied if either the number of categories is too small (less than 4) or if the manifest variables have different numbers of

categories.

In the latter case only models (4) and (8) are applied assuming ordered categories without further restrictions. In case of only two categories for all variables, the first model is sufficient to reproduce the parameters of the unrestricted, dichotomous LCA, because two categories have only one threshold and no threshold distance. In case of 3 categories, models (1), (2), (5), and (6) can be estimated, because 3 categories have 2 thresholds and one distance. Model (6) is equivalent to the unrestricted LCA in this case.

What can LCA be used for?

The Latent Class Analysis may be applied whenever a latent, i.e. non-observable typology or classification is to be identified for a set of persons or objects which are characterized by several categorical variables.

Examples are:

- 400 persons responses to a 9-item questionnaire aimed at assessing their attitudes towards ways of environmental protection. Each item is rated on a 5-point scale. Two groups of persons are assumed,

which attribute responsibility for environmental protection to external or to internal factors respectively.

- 1000 individuals rated their proximity to four political parties on a 5–point scale. The question is whether the four parties can be located on a latent continuum or whether latent types of persons with different patterns of proximity have to be assumed.
- All patients of a psychiatric clinic were rated according to a list of clinical symptoms. The aim is to analyze whether the classical psychiatric categories of mental disease can be reproduced by identifying latent types of persons with corresponding symptom patterns on the basis of these data.

These examples share

- a relatively large number of persons (observations),
- a relatively small number of manifest variables, which can be responses in a test or questionnaire, expert ratings, standardized behavior observations, or all kinds of observable variables like hair color, sex or social status,
- that these variables are discrete, i.e. they have a limited number of categories and each individual has one (and only one) value or category on each variable, and

- that a latent classification or typology of the individuals is aimed at, which is 'latent' because no observable or manifest variable can produce this classification. It is probabilistic because probabilities of membership in the latent classes instead of deterministic assignments are obtained for each individual.

The least restrictive model in the program is the unrestricted latent class model, which can be applied to unordered categories (nominal scale variables) or be used to check if categories are ordered. If the categories of all variables are defined in the same way and, hence, all variables have the same number of categories, eight different models can be computed and checked for their fit to the data (see below). The program's use lies in the identification of a latent classification for the individuals.

So far it is only a general description of latent class analysis and its results. One property of the Latent Class module of WINMIRA 32 is its applicability to ordered categories. However, it is not known whether the distances between the categories are the same between all categories, whether the distances have the same size for all variables, nor whether they are the same for all groups of persons. A systematic combination of these three types of restrictive assumptions leads to a system of 8 models for ordinal variables.

Beginning with the latter, the distinction of category distances which hold for all persons and class-specific distances divides the system in two groups of four models each., i.e.

class independent distances

and

class specific distances.

The assumption that all manifest variables have the same distances between categories gives models (1) and (5) (called *rating scale LCA* in the model selection menu). The so-called equidistance assumption, i.e. all categories of a manifest variable have the same distance, yields models (2) and (6). (called *equidistance LCA*). A scaling concept which goes back to Thurstone's method of successive intervals assumes that the categories have their own distances as a characteristic of the response format, but the variables have their own dispersions. If no assumption about category distances at all is made models (4) and (8) are obtained, where (8) is the unrestricted latent class model and (4) makes the assumption that the same distances hold for all persons, whatever they are for a particular variable and a particular category.

It can be decided empirically which of these assumptions is most

appropriate to the data. Please refer to section on the Bootstrap or to the section describing an Example Output file for further information on model selection.

The Mixed Rasch Model

The Mixed–Rasch Model extends the Rasch model to a discrete mixture model. The main goal of applying this model is to classify a possible inhomogeneous sample into Rasch–homogenous subsamples.

The Mixed Rasch model can be used for very different tasks, e. g.

- for testing model fit of the Rasch Model (by comparing the one–class and the two–class solution),
- for identifying a Rasch scaleable subpopulation (or separating a class of unscaleables, respectively),
- for analyzing rating data, when different subsamples have different response sets,
- for measuring a latent ability, when different people apply different solution strategies for solving the items, or
- for profile analysis of questionnaire items with ordinal response formats.

WINMIRA 32 can be applied to dichotomous and polytomous data. All characteristics of the Rasch Model are preserved within the latent classes, so that the program can also be used for ordinary Rasch Analyses (by

computing the one-class solution).

The dichotomous Mixed Rasch model

The model equation of the MRM for dichotomous item responses $x^{vi} \in \{0,1\}$ is

$$P(x^{vi}) = \sum_{g=1}^G \pi_g \left(\frac{\exp(x^{vi}(\theta_{vg} - \sigma_{ig}))}{1 + \exp(\theta_{vg} - \sigma_{ig})} \right) \quad (3)$$

where θ_{vg} is the person parameter of subject v in latent class g and σ_{ig} is the item difficulty of item i in latent class g (Rost 1990). The latent classes are identified by means of an EM-Algorithm and the item – or threshold parameters are computed by means of conditional maximum Likelihood (CML) estimation within each M-step. The CML estimation requires the latent score distributions, i.e., the distributions of test scores in each latent class, to be estimated in order 'to condition out' the person parameters in the CML-procedure. Then, the probability of a response pattern \underline{x} can be written as:

$$P(\underline{x}) = \sum_{g=1}^G \pi_g \pi_{rg} \left[\exp(\sum_{i=1}^k x_i \sigma_{ig}) / (\gamma^g) \right] \quad (4)$$

with response pattern $\underline{x}=(x^1,\dots,x^k)$, latent score distribution parameters π_{rg} , the probability of score r in class g , and symmetric function γ_{rg} of order r in class g .

The polytomous Mixed Rasch model

WINMIRA can be used to estimate model parameters for different polytomous (ordinal) Rasch models the 'rating scale model' (Andrich 1978), the 'equidistance model' (Andrich 1982), the 'dispersion model' (Rost 1988) and the 'partial credit model' (Masters 1982) as well as their mixture generalizations (compare Rost 1991, von Davier & Rost, 1995). Generalized to mixture distribution models, the polytomous Rasch models can be written as follows:

$$P(\underline{x}) = \sum_{g=1}^G \pi_g \pi_{rg} [\exp(\sum_{k_i=1}^k \sum_{x_s=1}^m \alpha_{ixg}) / (\gamma_{rg})] \quad (5)$$

with response pattern $\underline{x} = (x^1,\dots,x^k)$, $x^i \in \{0,\dots,m\}$, π_{rg} probability of score r in class g , and symmetric function γ_{rg} of order r in class g . In case of the most restrictive model, i.e. the mixed Rating Scale Model, α_{ixg} is decomposed as follows:

$$\alpha_{ixg} = \mu_{ig} + \lambda_{xg}$$

for all g and $x \in \{1, \dots, m\}$, with the condition $\sum_x \lambda_{xg} = 0$ to avoid indeterminacies. The category parameters λ_{xg} parameterizes the easiness of threshold x in class g . In the program output, the combined parameters α_{ixg} are printed for each model, so that different models can be compared more easily. The second model, i.e. the mixed Equidistance Model, has the decomposition:

$$\alpha_{ixg} = \mu_{ig} + (m+1-2x)\delta_{ig}$$

for all g and $x \in \{1, \dots, m\}$. The third model, i.e. the mixed Dispersion Model, has both equidistance and threshold parameters and is decomposed as follows:

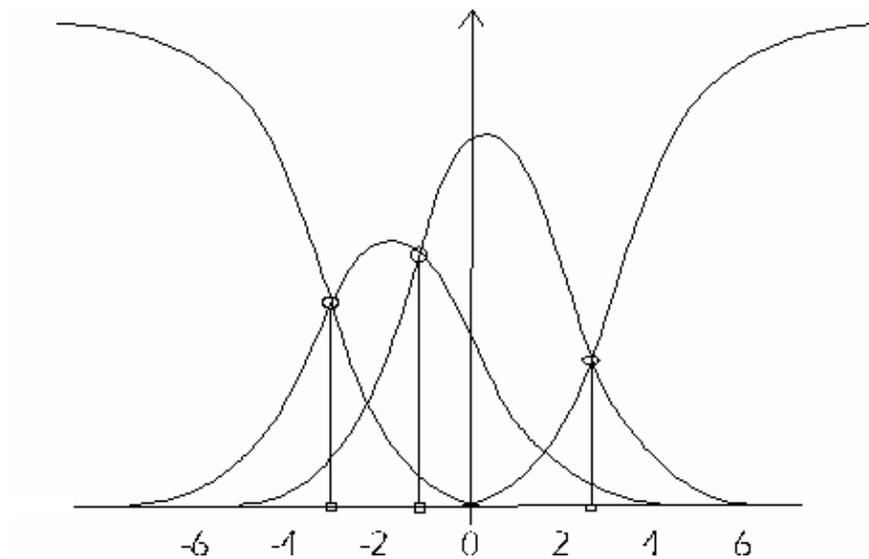
$$\alpha_{ixg} = \mu_{ig} + \lambda_{xg} + (m+1-2x)\delta_{ig}$$

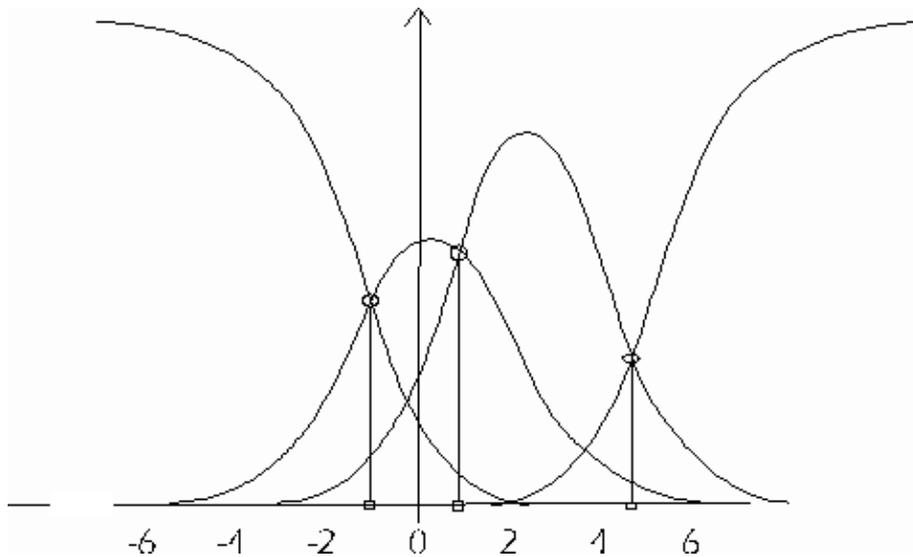
for all g , $x \in \{1, \dots, m\}$. In the fourth model, i.e., the mixed Ordinal (or Partial Credit) Model, all α_{ixg} are estimated separately, there is no restriction except the normalizing condition

The Rating Scale Model

$$(\alpha_{ixg} = \mu_{ig} + \lambda_{xg})$$

The Rating Scale Model assumes that all threshold distances are constant across the items, which is indicated by the missing item index i for the threshold parameters λ_{xg} . The figure below shows the category characteristic curves (CCCs) for two items as an example, where the difference between threshold 1 and 2 is 2.0 for both items, the distance between threshold 2 and 3 is approx. 3.5 for both items.



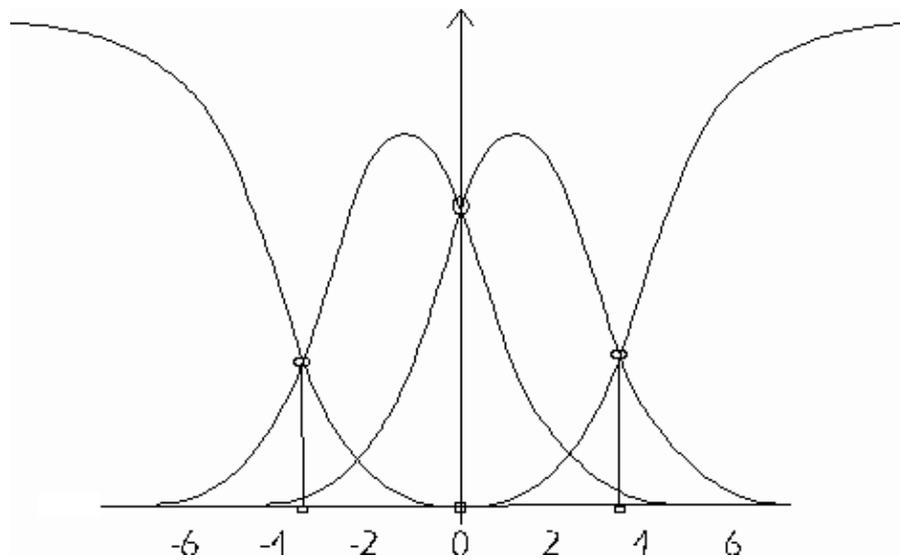
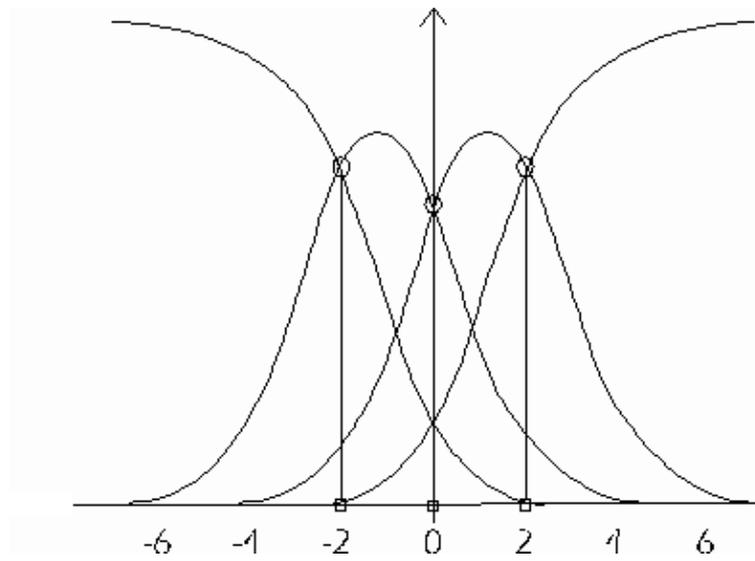


[more]

The Equidistance Model

$$(\alpha_{ixg} = \mu_{ig} + (m+1-2x)\delta_{ig})$$

In the Equidistance Model (formerly called Dispersion Model), all threshold distances are assumed to be constant within each item, as indicated by the missing threshold index x for the dispersion parameters δ_{ig} . The figure below shows an example where the threshold distance is 2.0 units for the first and 3.5 units for the second item.



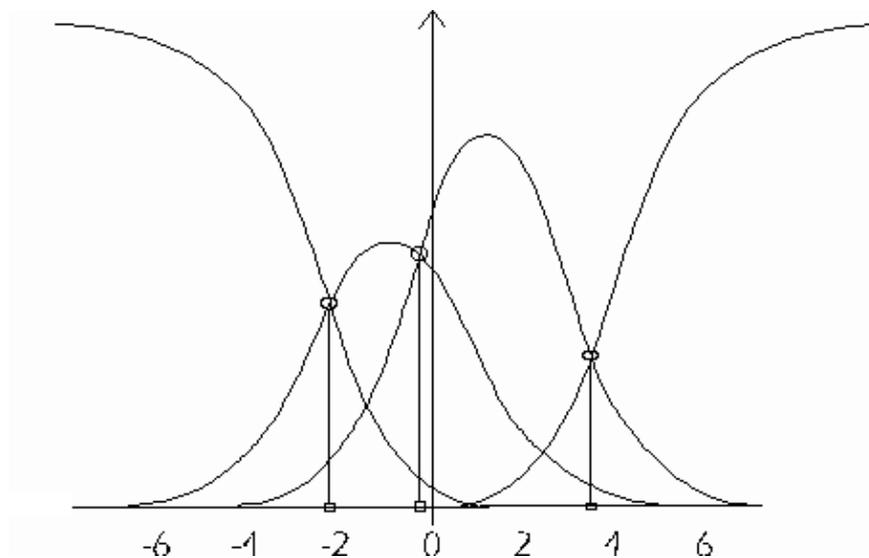
[more]

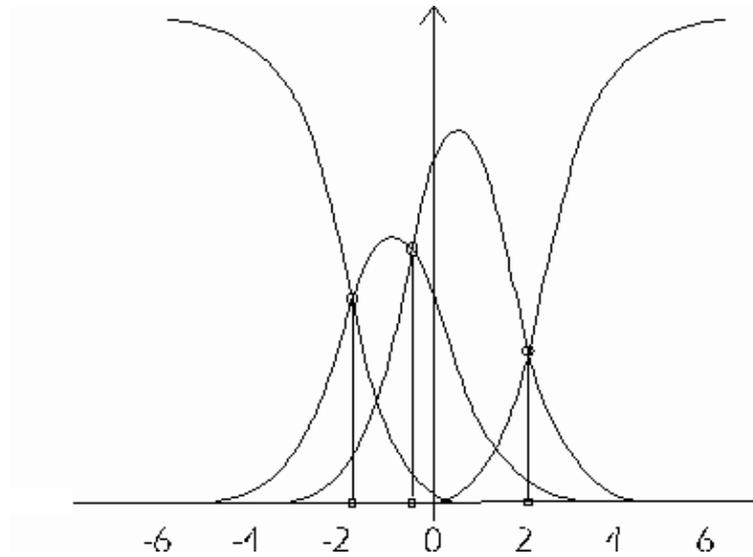
The Dispersion Model

$$\left(\alpha_{ixg} = \mu_{ig} + \lambda_{xg} + (m+1-2x)\delta_{ig} \right)$$

In the Dispersion Model (formerly called Successive Interval Model), there are both equidistance parameters δ_{ig} and threshold parameters λ_{xg} .

The λ -parameters define basic distances for the thresholds, which can be increased or decreased by the δ -parameters: negative values $\delta < 0$ decrease the distances, positive values $\delta > 0$ increase the distances. In this example, the second threshold distance is larger than the first one within both items.



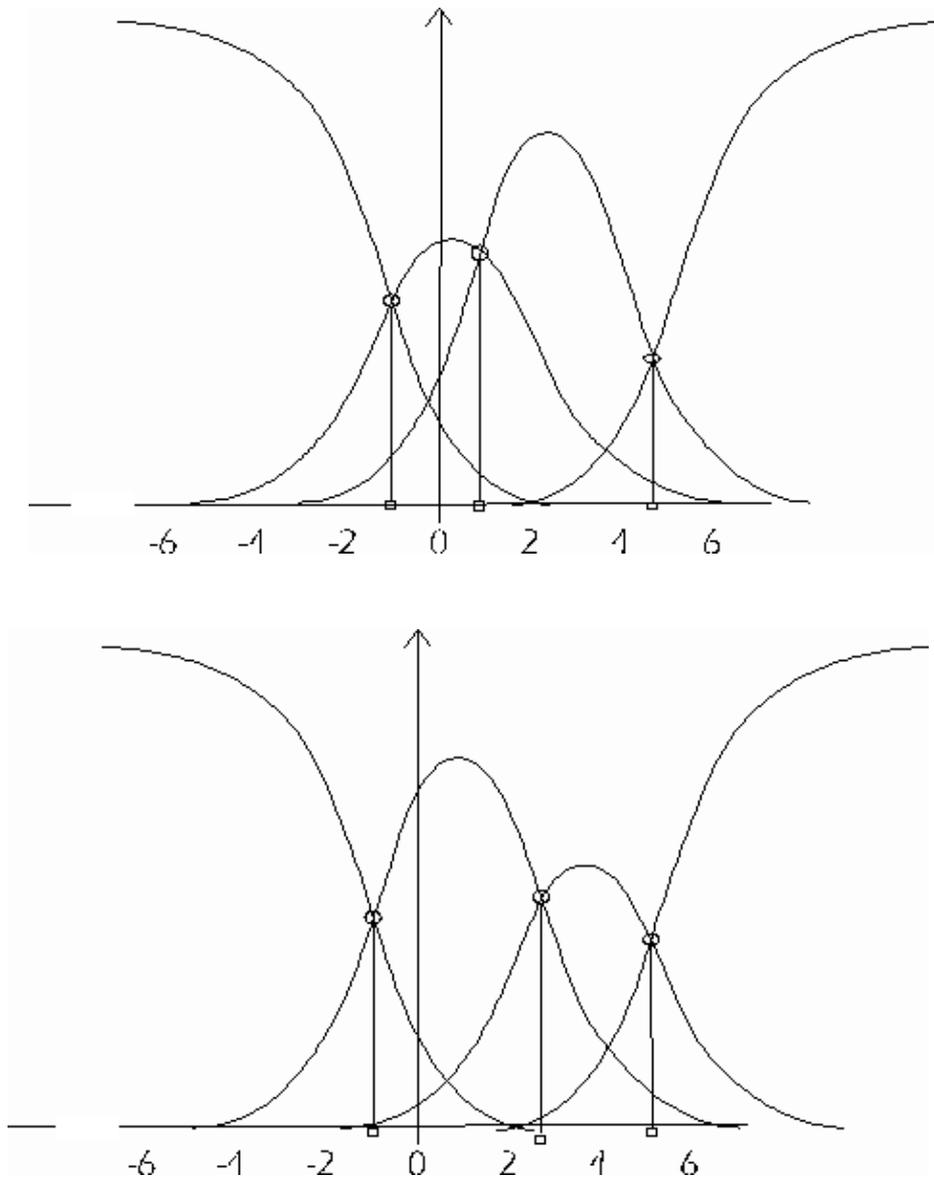


[more]

The Ordinal (Partial Credit) Model

(α_{ixg} without restrictions)

In the Ordinal or Partial Credit Model, there is one parameter α_{ixg} for each threshold of each item, without any restriction except the inevitable normalizing conditions.



Person Parameters

The previous subsection presented the model equation of the mixed Rasch model in the conditional notation. Conditional maximum likelihood

estimation (CML) is used in WINMIRA in order to avoid estimating biased person and item parameters simultaneously. This ensures that consistent item parameters are obtained which can be used to obtain person parameter estimates.

Estimation of person parameters is carried out by an UML-procedure after the CML-estimation of the item parameters is completed. Maximum likelihood estimates (MLE) as well as weighted likelihood estimates (WLE, see Warm, 1989) can be computed with WINMIRA. Warm's WLE estimates have, as compared to the MLE estimates, two main advantages: First, their bias is smaller (Warm 1989, Hoijsink & Boomsma, 1995), and second, they produce reasonable estimates even for the two extreme response patterns, i.e., for the patterns with zero and maximum score.

The estimation of person parameters is optional and can be chosen with the corresponding option in the Output Options menu of WINMIRA. If "compute person parameters" was chosen, a table with MLE and WLE estimates along with the corresponding standard errors will be included in the output file for each Rasch model class. The person parameter estimates are also printed in a separate file if append person parameters to datafile was selected in the options menu. This file can be used in subsequent analyses.

Latent Score Distributions

The latent score distributions determine the score probabilities in each Rasch type latent class. These score probabilities are necessary in the conditional maximum likelihood estimation for conditioning out the person parameters in order to obtain unbiased item parameter estimates. The latent score distributions can be fully parameterized (i.e. one parameter for each score in each latent class) or estimated by assuming a two-parameter model. According to this restricted model, the score distribution within each class g is parameterized by a location parameter τ_g and a dispersion parameter δ_g , assuming that the following relationship holds:

$$P_{rg} = \frac{e^{\left[\frac{\gamma}{L} \tau_g + g(r, mk) \delta_g \right]}}{\sum_{s=0}^{mk} e^{\left[\frac{\gamma}{L} \tau_g + g(s, mk) \delta_g \right]}} \quad (9)$$

where $(m+1)$ is the number of response categories, k is the number of items, $g(r, mk) = 4r(mr-r)/(mk)^2$ normalizing coefficient.

The parameters of this distribution have the mathematical property of a

location parameter (τ) and a dispersion parameter (δ). The following figure shows the shape of the score distribution as a function of 4 different sets of parameters. It can be seen that the model is capable of approximating very different shapes like symmetric unimodal, extremely "dislocated" and u-shaped distributions.

Figure 1a:
 Tau = 0, Delta = 3.

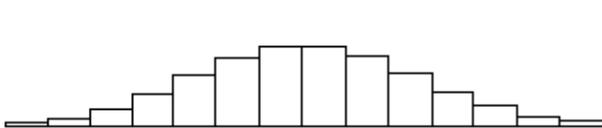


Figure 1b:
 Tau = -6, Delta = 6.

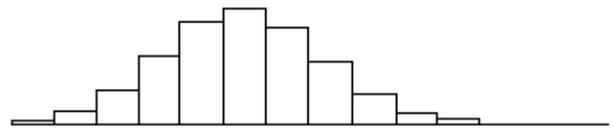


Figure 1c:
 Tau = 9, Delta = 2.

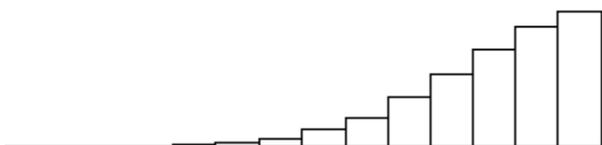
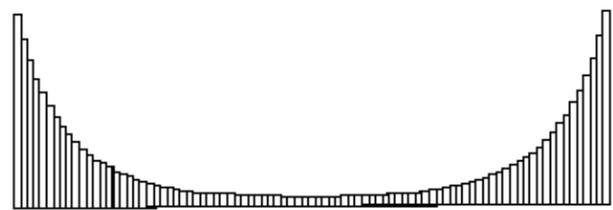


Figure 1d:
 Tau = 0, Delta = -3.

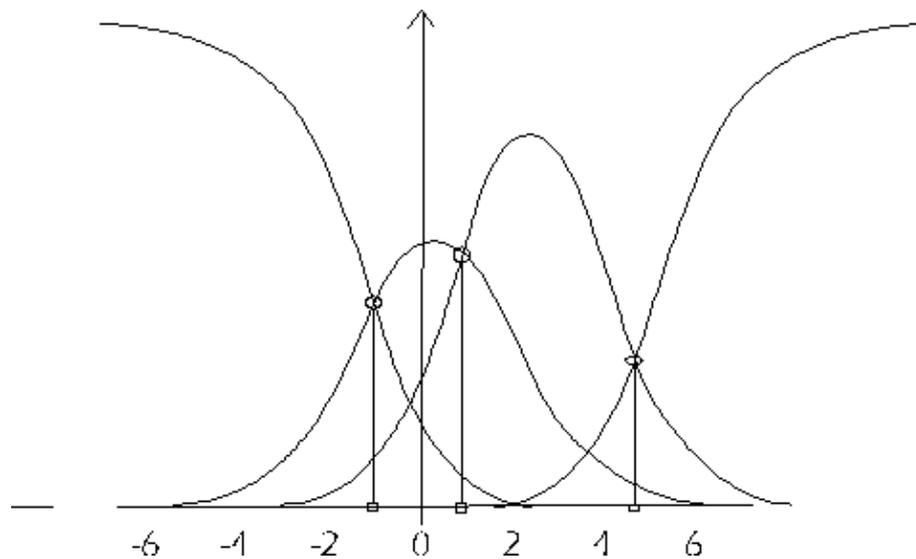


A special feature of this parameterization is the relative invariance of shape

of the score distribution w.r.t. how many score frequencies have to be fitted. In the figure above, the same parameters were used to smooth 14 and 88 raw score frequencies in each of the four diagrams.

Category Characteristic Curves

The so-called category-characteristic curve (CCC) can be used to visualize the relationship between item parameters and response probabilities. The x-axis represents the latent dimension and the y-axis shows the response probability for each category $x \in \{0, \dots, m\}$. The intersections of the response probability curves are given by the item parameters in the de-cumulated notation (i.e. the threshold parameters). In the example below, these thresholds take the values $-1, 1$ and 4.5 respectively.



Given a person with parameter θ , the response probability for this person can be directly taken from the CCC, as both the item (threshold) parameters and person parameters are located on the same latent dimension.

[proceed]

The Q-Index

The item Q-index (Rost and von Davier, 1994) is an item fit index which makes use of the statistical properties of Rasch models, i.e., parameter separability and conditional inference. The item-Q includes no assumption about the scale level of the response variable. Instead, it is based on the

log-likelihood of the observed item-pattern. It can be applied to any unidimensional Rasch model like the dichotomous model, the rating scale model (Andrich 1978), the equidistance model (Andrich 1982), the partial credit model (Masters 1982) or dispersion model (Rost 1988). The fit of an item i is evaluated with regard to the conditional probability of its observed item response vector, i.e.

$$p(x^i, n^{i0}, \dots, n^{im}) = \exp(\sum_x x^{vi} \beta^{iv}) / \tau(\beta; n^{i0}, \dots, n^{im})$$

where the denominator is given by the symmetric functions of order $(n^{i0}..n^{im})$ of the person parameters. This conditional pattern-probability is standardized twice. First, it is divided by the maximum probability a pattern with a particular score distribution can reach, i.e. the probability of the optimum-pattern, \underline{x}^{OPT} or Guttman-pattern. The Likelihood ratio

$$LR_{i,OPT} = p(\underline{x}^i | n^{ix}) / p(\underline{x}^{OPT} | n^{ix})$$

then, approximates 1 (one) for an increasing pattern probability. The logarithm of this Likelihood ratio is standardized again, i.e. it is divided by the smallest possible Likelihood ratio

$$LR_{PESS,OPT} = p(\underline{x}^{PESS} | n^{ix}) / p(\underline{x}^{OPT} | n^{ix})$$

where $\underline{x}^{\text{PESS}}$ denotes the pessimum-pattern or the anti-Guttman-pattern which is the pattern with lowest probability. The ratio of both log-likelihood-ratios is

$$Q_i = \ln[P(X_i | n_{ix}) / P(X^{\text{OPT}} | n_{ix})] / \ln[P(X^{\text{PESS}} | n_{ix}) / P(X^{\text{OPT}} | n_{ix})]$$

$$= [\sum^V (x^{vi} - x^{v,\text{OPT}}) \beta^v] / [\sum^V (x^{v,\text{PESS}} - x^{v,\text{OPT}}) \beta^v]$$

which is a very simple function of the ability parameters. Q varies between 0 and 1, where 0 indicates perfect (Guttman pattern) fit and 1 indicates perfect misfit (anti-Guttman pattern) or deviance from the model. A value $Q = 0.5$ indicates independence of the trait and the item, i.e., random response behavior. Rost & von Davier (1994) presented another standardization of the item Q -index with zero mean and unit variance and which can be assumed to be asymptotically normal. In WINMIRA, both the Q -index as well as its asymptotically normal standardization are listed in a table in the output file if the respective output option is activated.

The Hybrid Model

The Hybrid Model (Yamamoto, 1989) assumes that the data can be described by a mixture of an IRT model (like the Rasch model) and the Latent Class Analysis. This implies that each observed response pattern either stems from a latent subpopulation where the IRT model holds or that the response pattern can be fitted by one of the latent classes in the LCA part of the model.

Hybrid mixtures of IRT models and Latent Class models can be written as

$$P(\mathbf{X} = \mathbf{x}^v) = \sum_{g=1}^G \pi_g P^{\text{IRT}}(\mathbf{X} = \mathbf{x} | \theta_{vg}) + \sum_{c=1}^C \pi_c P^{\text{LCA}}(\mathbf{X} = \mathbf{x} | c)$$

where $\sum_{g=1}^G \pi_g + \sum_{c=1}^C \pi_c = 1$, i.e., the sum of all class sizes is one. The first mixture sum stands for the IRT mixture components and the second sum stands for the Latent Class type mixture components.

In WINMIRA, Hybrid models can be estimated for polytomous data, and additionally, models with more than one IRT class can be specified (v. Davier, 1994). This type of Hybrid model integrates a finite number of mixed Rasch models and a finite number of Latent Class models, so that each latent class of this new model family can have its own structure.

Using the output of WINMIRA

The output files generated by WINMIRA are organized as follows: An ASCII text file contains tables with summaries of the sufficient statistics, the final parameter estimates and Goodness-of-Fit statistics. In addition, graphs of item and person parameters can be produced and saved as bitmaps or metafiles. Finally, person specific information can be appended to the data file.

- printing output files
 - graphical output
 - append variables to the datafile
 - example output file (a little outdated)
-

Printing the output

Please print output files by choosing the File > Print Output menu entry.



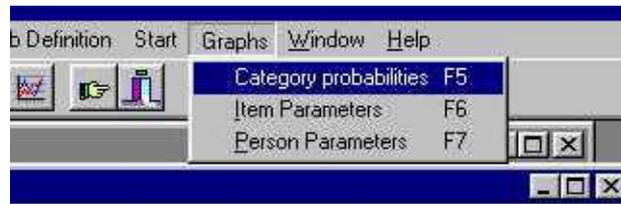
Alternatively, you can use the printer speed button in the panel.



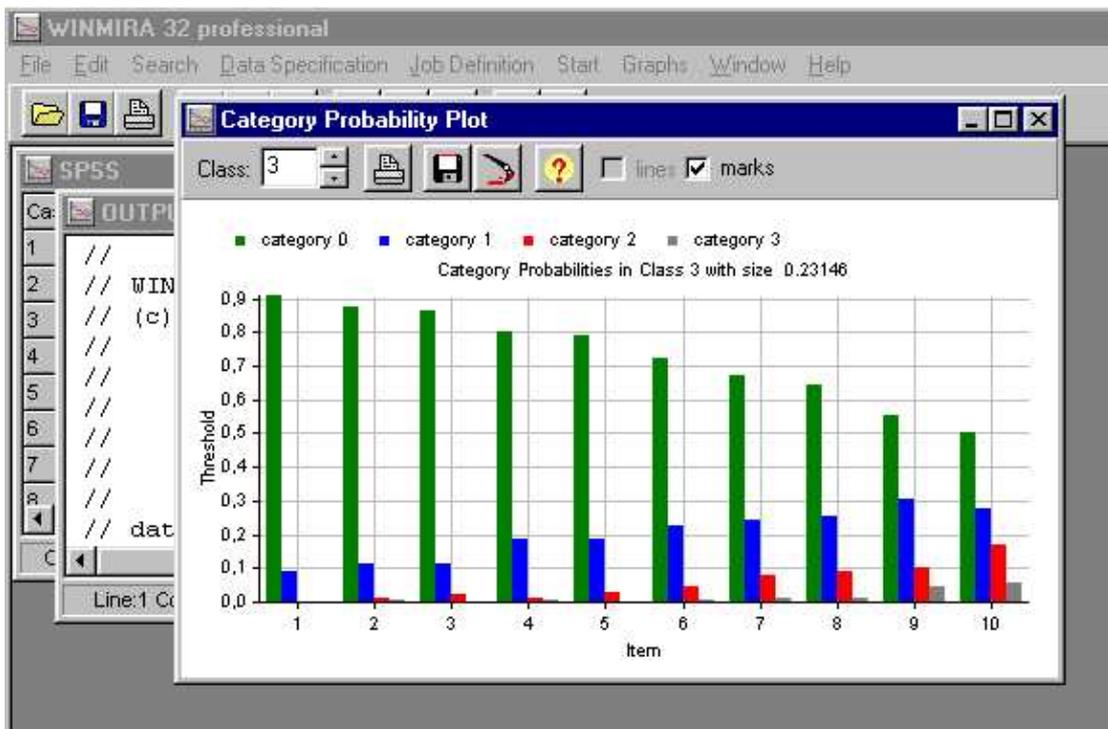
Printing the output will produce a number of pages with all tables contained in the output file. Please use the print option in the graph windows if hardcopies of the graphical output is also required.

Display Graphical Output

By choosing any of the function keys F5, F6 or F7 or the corresponding entries in the graphs menu, plots of class specific model parameters are displayed. The plots are resizeable and can be printed and/or saved to a file (see below).

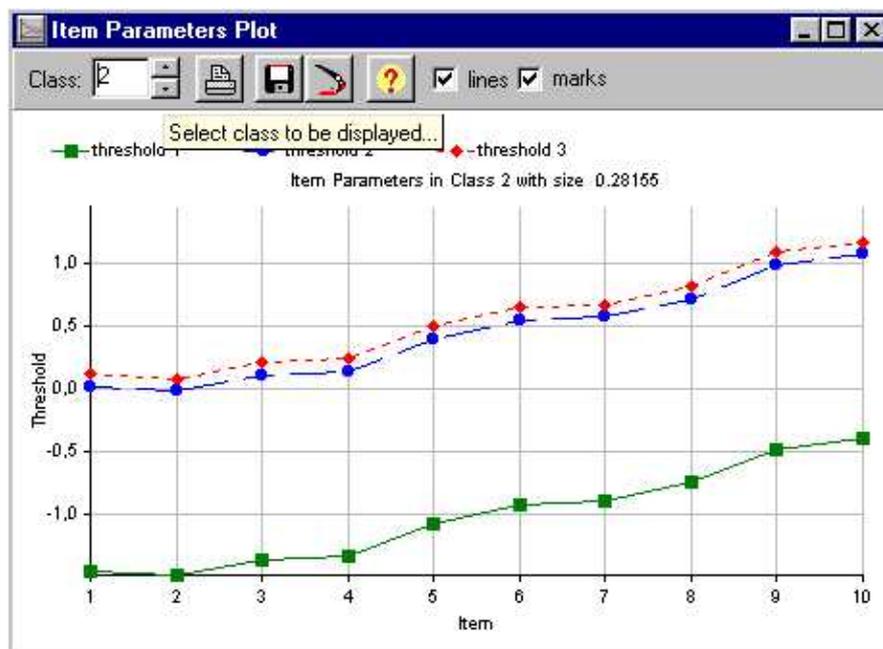


"Category probabilities" or the function key "F5" displays a histogram of the class specific response probabilities for all items.



"Item parameters" or "F6" displays class specific threshold parameters.

The spin button on the left side of the graphic control panel can be used to select the latent class to be displayed.



Each graph can be printed or saved directly from within the graphs control panel. The person parameter graph shows the absolute raw score frequencies for each class and, if the class was assumed to be Rasch homogenous, a simultaneous person parameter plot for the Maximum likelihood and the Warm person parameter.



Example Output file

A Hybrid model combining ordinal Rasch models and Latent Class models:

Below, an output file as produced by WINMIRA 32 is commented in order to give a guideline for reading the results of an analysis. To learn more about discrete mixture distribution models, a book like "Lehrbuch der Testtheorie" by J. Rost is highly recommended. To learn more about using the program WINMIRA 32, please refer to the on-line manual or the user manual.

```
//  
// WINMIRA 32 beta v0.97  
// (c) 1998,1999 by Matthias von Davier  
// IPN - institute for science education  
// Olshausenstrasse 62  
// 24098 Kiel  
// Germany  
// email: vdavier@ipn.uni-kiel.de or rost@ipn.uni-kiel.de  
//  
// date of analysis: 24.11.98 time : 12:05:01  
//
```

Filenames:

```
data: simula9x.dat  
output: simula9x.out  
member: simula9x.mem  
patterns: simula9x.pat
```

```
number of persons : 2001  
number of items : 10  
number of categories : 4  
number of classes : 3  
max. number of iterations : 350  
accuracy criterion : 0.0010
```

random start value : 4321

The output file starts with a summary of the data specification and a description of the selected model. In this example, three latent classes are assumed with different models holding in each class.

variable labels, positions and sample frequencies:

	n of	rec-	start	end	categories					
no.	label	cats	ord	col.	col.	0	1	2	3	N
01	item01	4	1	3	3	781	509	350	361	2001
02	item02	4	1	4	4	798	476	379	348	2001
03	item03	4	1	5	5	742	497	402	360	2001
04	item04	4	1	6	6	691	551	388	371	2001
05	item05	4	1	7	7	697	545	394	365	2001
06	item06	4	1	8	8	616	566	451	368	2001
07	item07	4	1	9	9	601	557	436	407	2001
08	item08	4	1	10	10	590	539	434	438	2001
09	item09	4	1	11	11	571	566	427	437	2001
10	item10	4	1	12	12	534	546	453	468	2001

number of cases with invalid data: 0

The table above shows the label and the position in the datafile for each

variable in the scale as defined by the user. The data file may have more than one row per observation, so that both the record (row) and the start and end columns of the variables have to be specified.

```
saturated likelihood : -14815.1431
```

The saturated likelihood is the theoretical maximum of the likelihood function that can be reached. This maximum can only be met by the saturated model by assuming one parameter for each observed response pattern. The saturated likelihood is used in the likelihood ratio Goodness-of-Fit test.

```
number of different patterns : 1834
```

```
number of possible patterns : 1048576
```

The ratio of observed to possible patterns indicates that many of the possible pattern have zero frequencies, i.e. they haven't been observed. This implies that traditional Goodness-of-Fit statistics (see below, at the end of this example output) can not be used for testing a model for this dataset.

```
Number of iterations needed: 117
```

117 iterations were needed to reach the default accuracy criterion and terminating estimation.

fitted model: Hybrid model in 3 latent classes.

A discrete mixture of different models for each class was fitted in this example. The model of class one and two is a polytomous Rasch model (the Rating Scale model in this example). In the third class, local independence according to the ordinary Latent Class model is assumed.

Final Estimates in CLASS 1 of 3 with size 0.49311

=====

(MIRA) Mixed Rasch Model:
according to the rating scale model

The class size indicates that about 50 percent of the population can be fitted by a polytomous Rasch model which was assumed to hold in this class.

Expected Score Frequencies and Personparameters:

Raw-	Expected	MLE-	SE(MLE)	WLE-	SE(WLE)
score	freq.	estimate		estimate	

_____	_____	_____	_____	_____	_____
-------	-------	-------	-------	-------	-------

0		28.92		*****		*****		-3.597		1.375
1		67.99		-2.957		0.984		-2.550		0.795
2		68.12		-2.288		0.694		-2.072		0.621
3		85.94		-1.897		0.569		-1.755		0.532
4		54.58		-1.614		0.499		-1.512		0.477
5		62.72		-1.388		0.453		-1.312		0.440
6		37.86		-1.198		0.422		-1.139		0.413
7		34.08		-1.030		0.399		-0.983		0.393
8		27.82		-0.878		0.382		-0.841		0.378
9		34.41		-0.737		0.369		-0.708		0.367
10		31.17		-0.605		0.359		-0.582		0.358
11		38.97		-0.478		0.352		-0.462		0.351
12		28.80		-0.357		0.347		-0.344		0.346
13		32.78		-0.238		0.343		-0.230		0.343
14		28.91		-0.121		0.341		-0.117		0.341
15		32.26		-0.005		0.340		-0.005		0.340
16		34.43		0.111		0.341		0.107		0.341
17		30.11		0.228		0.343		0.220		0.343
18		23.74		0.347		0.347		0.335		0.346
19		33.34		0.469		0.352		0.452		0.351
20		27.10		0.596		0.360		0.573		0.358
21		33.97		0.729		0.370		0.699		0.367
22		23.65		0.870		0.383		0.833		0.379
23		31.81		1.024		0.401		0.976		0.395
24		19.25		1.193		0.424		1.133		0.415
25		20.26		1.387		0.457		1.308		0.443
26		8.06		1.616		0.504		1.512		0.481

27		0.82		1.904		0.576		1.761		0.538
28		2.87		2.305		0.701		2.087		0.629
29		2.00		2.987		0.993		2.581		0.807
30		0.72		*****		*****		3.656		1.397

The table above shows the expected rawscore frequencies in class 1, the person parameter estimate and the standard error of estimation of the individual parameter for all rawscores in class one. In this table, both maximum likelihood (MLE) and bayes weighted likelihood estimates (WLE) are shown. The WLE should usually be preferred as they are less biased and give reasonable estimates even for the to extreme score groups. In the case of mixture distribution models, the class specific expected frequencies can not be compared to the observed frequencies, as only the overall observed frequencies are known. Nevertheless, it can be seen from the expected frequencies for example, in which class most of the higher scoring persons belong.

WLE estimates : Mean = -0.726 Stdev = 1.203

Reliability = 0.818

Raw-score : Mean = 10.515 Stdev = 7.672

This is the mean and the standard deviation of the WLE person parameter and the raw score.

expected category frequencies and item scores:

Item .	Item`s	relative	category			
label .	Score	Stdev	frequencies			
_____	_____	_____	0	1	2	3
item01	0.54	0.84	0.646	0.216	0.095	0.043
item02	0.65	0.92	0.602	0.205	0.135	0.058
item03	0.75	0.99	0.561	0.213	0.140	0.087
item04	0.82	1.02	0.525	0.233	0.140	0.102
item05	0.93	1.07	0.478	0.238	0.157	0.127
item06	1.13	1.12	0.404	0.229	0.197	0.170
item07	1.24	1.15	0.367	0.224	0.210	0.199
item08	1.33	1.18	0.348	0.212	0.205	0.235
item09	1.47	1.17	0.290	0.222	0.216	0.272
item10	1.64	1.16	0.239	0.198	0.244	0.319
Sum: :	10.50					

The expected category frequencies show the descriptive characteristics of the items in each class. Item 1 and item 2, for example, are very difficult in this class, because about 80 to 90 percent of the individuals choose only the lowest two categories 0 and 1. These relative frequencies are overall or mean values, because in Rasch–Model–Classes these relative frequencies depend on the distribution of the individual parameter in the class.

In the ordinal Rasch model, there is a strictly positive relationship between

the probability of choice of the higher of two adjacent categories and the individual parameter. Subjects with a high individual parameter have higher probabilities for the upper categories than in the table above. Accordingly, a subject with a low parameter will have higher probabilities for the lower categories.

threshold parameters: rating scale model

item	threshold parameters			
label	1	2	3	location
item01	0.468	0.861	1.419	0.916
item02	0.229	0.622	1.180	0.677
item03	0.035	0.428	0.985	0.483
item04	-0.086	0.307	0.865	0.362
item05	-0.280	0.113	0.670	0.167
item06	-0.609	-0.216	0.341	-0.162
item07	-0.779	-0.386	0.171	-0.332
item08	-0.917	-0.524	0.034	-0.469
item09	-1.134	-0.741	-0.184	-0.687
item10	-1.405	-1.012	-0.455	-0.957

mean threshold distances: 0.393 0.558

This table shows the item parameter estimates for the Rating Scale model

in class one. These threshold parameters should be ordered, i.e. the parameters should decrease from threshold to threshold whenever an ordered response format is assumed. A decreasing easiness of the threshold parameters indicates that every response category is representative for an interval of the individual parameter dimension. The last column is an overall easiness parameter (computed by summing up all threshold parameters). Again item1 and item2 seem to be very difficult, that is, they have a high difficulty parameter.

item fit assessed by the Q-index

itemlabel	Q-index	Z _q	p(X>Z _q)	
item01	0.1099	-0.7154	0.76281	-.....!.Q..+
item02	0.1241	0.5296	0.29818	-.Q...!.+
item03	0.1113	-0.4274	0.66545	-.....!Q...+
item04	0.1157	0.2674	0.39459	-..Q!.+
item05	0.1026	-0.5451	0.70715	-.....!.Q..+
item06	0.1014	0.0200	0.49201	-...Q!.+
item07	0.0995	0.1398	0.44439	-...Q!.+
item08	0.0927	-0.7459	0.77213	-.....!.Q..+
item09	0.1052	0.9727	0.16534	-Q...!.+
item10	0.0886	0.1952	0.42261	-...Q!.+

-?:p<0.05, +?:p>0.95

-!:p<0.01, +!:p>0.99

The table above shows the Q-Index, a class-specific item-fit measure for Rasch models. The Q-index lies between zero (indicating perfect discrimination, i.e., a Guttman – pattern) and one (indicating perfect "anti-discrimination"). A value of 0.5 indicates no relationship between the individual parameter and the reaction to the item. The Z_q column is a transformation of the Q-index that is approximative normally distributed if the Rasch model holds for the respective item. High positive values indicate that the item discrimination is lower than assumed by the Rasch model (under-fit), negative values indicate higher discrimination than assumed (over-fit).

In this example, all items seem to fit fairly well. Misfitting items can be detected by examining the Z_q value in the table above. High positive values indicate lower discrimination than expected. The Z_q transform of Q-index for items detects very small deviations of the item characteristic with increasing power (i.e. sample size). Therefore, an item should be removed from the scale only after examining the items content and additional information from the estimated model (e.g. "strange" category frequencies or non-monotone threshold parameters).

Final Estimates in CLASS 2 of 3 with size 0.28155

=====

(MIRA) Mixed Rasch Model:

according to the rating scale model

Following now is the output for latent class 2. This class is expected to include about 30% of the sample. The model assumed in this class is the rating scale model again. Therefore, a detailed comment is omitted in the subsequent output for class 2 and continued for latent class 3, as that class is assumed to follow a different model, namely the traditional latent class model.

Expected Score Frequencies and Personparameters:

Raw-	Expected	MLE-	SE(MLE)	WLE-	SE(WLE)
score	freq.	estimate		estimate	
0	2.08	*****	*****	-4.101	1.474
1	0.01	-3.321	1.029	-2.940	0.871
2	2.87	-2.570	0.746	-2.368	0.688
3	0.01	-2.110	0.623	-1.974	0.592
4	12.25	-1.769	0.549	-1.667	0.530
5	18.72	-1.495	0.500	-1.415	0.486
6	25.30	-1.264	0.463	-1.198	0.453

7		29.29		-1.063		0.435		-1.008		0.428
8		37.70		-0.884		0.413		-0.838		0.408
9		24.28		-0.720		0.395		-0.682		0.392
10		32.38		-0.570		0.381		-0.538		0.379
11		27.25		-0.429		0.370		-0.403		0.368
12		44.25		-0.295		0.361		-0.275		0.360
13		21.60		-0.167		0.355		-0.152		0.354
14		17.67		-0.043		0.350		-0.033		0.350
15		24.73		0.078		0.347		0.084		0.347
16		21.58		0.198		0.346		0.199		0.346
17		14.06		0.318		0.346		0.314		0.346
18		25.60		0.438		0.349		0.429		0.348
19		15.99		0.561		0.353		0.547		0.352
20		30.30		0.688		0.359		0.668		0.358
21		29.71		0.820		0.368		0.793		0.366
22		42.05		0.960		0.380		0.925		0.377
23		15.51		1.111		0.397		1.066		0.392
24		19.22		1.277		0.419		1.219		0.411
25		9.10		1.465		0.450		1.390		0.437
26		13.68		1.687		0.495		1.587		0.473
27		6.08		1.964		0.564		1.824		0.527
28		0.11		2.347		0.686		2.133		0.614
29		0.01		3.001		0.974		2.595		0.782
30		0.28		*****		*****		3.605		1.347

WLE estimates : Mean = -0.026 Stdev = 0.890

Reliability = 0.794

Raw-score : Mean = 14.476 Stdev = 6.398

expected category frequencies and item scores:

Item	Item`s	relative category				
label	Score	Stdev	frequencies			
_____	_____	_____	0	1	2	3
item01	1.75	1.05	0.137	0.297	0.242	0.324
item02	1.78	1.03	0.128	0.284	0.267	0.320
item03	1.69	1.04	0.142	0.323	0.240	0.295
item04	1.67	1.02	0.134	0.339	0.250	0.277
item05	1.49	1.07	0.202	0.351	0.205	0.242
item06	1.38	1.01	0.216	0.371	0.231	0.181
item07	1.36	1.02	0.225	0.371	0.224	0.180
item08	1.26	1.00	0.259	0.369	0.228	0.144
item09	1.07	0.97	0.326	0.387	0.175	0.112
item10	1.02	0.94	0.342	0.385	0.183	0.090
Sum:	14.47					

threshold parameters: rating scale model

item	threshold parameters			
label	1	2	3	location
_____	_____	_____	_____	_____
item01	-1.447	0.019	0.122	-0.435

```

item02 | -1.484 | -0.018 | 0.085 | -0.472
item03 | -1.356 | 0.110 | 0.213 | -0.345
item04 | -1.331 | 0.135 | 0.238 | -0.319
item05 | -1.073 | 0.393 | 0.496 | -0.061
item06 | -0.918 | 0.548 | 0.651 | 0.094
item07 | -0.893 | 0.573 | 0.676 | 0.119
item08 | -0.747 | 0.719 | 0.822 | 0.265
item09 | -0.476 | 0.990 | 1.093 | 0.536
item10 | -0.392 | 1.074 | 1.177 | 0.619
mean threshold
distances: 1.466 0.103

```

item fit assessed by the Q-index

```

itemlabel | Q-index | Zq | p(X>Zq)
-----|-----|-----|-----
item01 | 0.1483 | -0.7426 | 0.77115 | -.....!.Q..+ |
item02 | 0.1807 | 0.6670 | 0.25238 | -.Q...!....+ |
item03 | 0.1528 | -0.2980 | 0.61714 | -.....!Q...+ |
item04 | 0.1631 | 0.4262 | 0.33497 | -..Q...!....+ |
item05 | 0.1550 | -0.7569 | 0.77545 | -.....!.Q..+ |
item06 | 0.1641 | 0.3487 | 0.36367 | -..Q...!....+ |
item07 | 0.1969 | 1.4552 | 0.07281 | Q.....!....+ |
item08 | 0.1543 | -0.3350 | 0.63118 | -.....!Q...+ |

```

```

item09 | 0.1499 | -0.7769 | 0.78138 | -.....!.Q..+ |
item10 | 0.1625 | -0.0382 | 0.51525 | -.....Q.....+ |

```

```
-?:p<0.05, +?:p>0.95
```

```
-!:p<0.01, +!:p>0.99
```

According to the Q-index, there is no item with a significant deviation from the expected characteristic as predicted by the Rating Scale model in this latent class. Similarly to the results in class 1, there is no indication to assume that the model does not fit the item responses in this class.

Nevertheless, a decision regarding model fit should only be based on the goodness-of-fit statistics for the whole model, which are given at the end of the output file.

The third latent class is a LCA-type class, i.e., it is assumed there are no systematic differences between the members of this class. In our example dataset, about 20 percent of the observed patterns can be fitted by this class.

```
Final Estimates in CLASS 3 of 3 with size 0.22534
```

```
=====
```

```
(LCA) Latent Class Analysis: class-specific thresholds:
according to the ordinal (partial credit) model
```

The partial credit model reserves one parameter for each threshold, i.e., in

contrast to the two Rasch model classes before, no restrictions are imposed on the threshold parameters.

expected category frequencies and item scores:

Item	Item`s	relative category	frequencies			
label	Score	Stdev	0	1	2	3
item01	1.72	1.05	0.147	0.286	0.266	0.301
item02	1.41	1.15	0.293	0.252	0.210	0.245
item03	1.52	1.10	0.242	0.233	0.286	0.240
item04	1.53	1.09	0.216	0.290	0.242	0.253
item05	1.49	1.10	0.246	0.249	0.274	0.230
item06	1.50	1.05	0.211	0.291	0.281	0.217
item07	1.46	1.11	0.248	0.283	0.227	0.242
item08	1.56	1.12	0.224	0.271	0.229	0.276
item09	1.50	1.08	0.224	0.286	0.256	0.234
item10	1.46	1.08	0.234	0.297	0.241	0.228
Sum	:	:	15.16			

The expected category frequencies are sufficient statistics of the item parameters in the latent classes. These conditional frequencies have to be estimated in each E–step of the EM algorithm in WINMIRA 32 for both, Latent Class models and Rasch models.

These category frequencies hold for all subjects in the case of a LCA class, because subjects do not differ systematically in the latent classes of LCA. Because of the local independence assumption in the LCA, the probability of any pattern can be computed by means of these relative category frequencies by simple multiplication.

threshold parameters: ordinal (partial credit) model

item	threshold parameters			
label	1	2	3	location
item1	-0.663	0.073	-0.126	-0.239
item2	0.151	0.180	-0.151	0.060
item3	0.038	-0.207	0.177	0.003
item4	-0.294	0.181	-0.044	-0.052
item5	-0.011	-0.096	0.177	0.023
item6	-0.320	0.034	0.257	-0.009
item7	-0.133	0.219	-0.064	0.007
item8	-0.187	0.167	-0.187	-0.069
item9	-0.244	0.110	0.089	-0.015
item10	-0.236	0.207	0.058	0.010

The threshold parameters in class 3 are listed in the table above. As in classes 1 and 2 before, the last column is an overall difficulty parameter. It can be seen that, as compared to the class 1, these parameters do not vary a

lot. This holds also for the threshold parameters, as the category frequencies are more or less equally distributed for all items.

The class specific output ends here. The following part contains some general information on whether and where the class membership information has been saved and overall goodness-of-fit measures.

statistics of expected class membership:

	exp.	mean			
class	size	prob.	1	2	3
1	0.521	0.865	0.865	0.051	0.084
2	0.289	0.770	0.066	0.770	0.164
3	0.190	0.707	0.126	0.167	0.707

The table above shows some descriptives of this assignment procedure. The expected class size is printed in the second column, the mean of the assignment probability (maximum posterior probability) is listed in the third column. The table is completed by a list of mean posterior probabilities for all classes, given that the current class (row number) has maximum posterior probability.

Goodness of fit statistics:

estimated saturated

model model

Log-Likelihood : -24568.35 -14823.74

Number of parameters : 112 1048575

geom. mean likelihood : 0.29293527 0.47672437

Information Criteria:

AIC-Index : 49360.70 2126797.49

BIC-Index : 49988.06 8000287.94

CAIC-Index : 50100.06 9048862.94

Power Divergence GoF statistics:

emp. value chi-square p-value

Likelihood ratio : 19489.21 p= 1.0000

Cressie Read : 190988.37 p= 1.0000

Pearson Chisquare : 1049539.31 p= 0.2287

Freeman-Tukey Chi² : 12989.60 p= 1.0000

Degrees of freedom : 1048463

To evaluate the fit of a specified model, the goodness of fit table has to be examined carefully. In the case of many items with more than 2 response categories, there are a lot of possible response patterns, most of which are not observed. Data with many zero frequencies are referred to as sparse

data. In case of sparse data, the traditional goodness-of-fit significance tests (in the table Likelihood ratio up to Neyman Chi-square) cannot be used (compare v.Davier, 1997).

```
WARNING: Number of cells is larger than number of different patterns!!!
```

```
obs.patterns/cells = 0.001749038696289060
```

```
number of zero cells = 1046742
```

```
WARNING: Number of cells is larger than number of subjects!!!
```

```
subjects/cells = 0.001908302307128910
```

```
The data might be very sparse, please do not use the  
chi square p-value approximation for the Power Divergence  
Goodness of Fit Statistics.
```

```
Consider to use the parametric bootstrap procedure instead.
```

```
In addition, several start values should be used (see defaults menu) in  
order to examine the occurrence of local likelihood maxima.
```

If the data table is sparse, some researchers rely on so-called information criteria (see above) to compare different models. Information criteria (IC) are based on the log-likelihood and the number of estimated model parameters. The number of parameters is included as a penalty term, so that more parsimonious models are preferred. These IC's, for instance the BIC, are compared for different models which were estimated for the same data and the model with the smallest IC is chosen.

A better way than evaluating the models by means of information criteria is to use the parametric bootstrap procedure for the evaluation of Goodness-of-Fit Statistics, which is implemented in WINMIRA 32 (see section 2.9). In the example above, 20 bootstrap samples have been simulated. At the bottom of the table, the empirical p-values for these 20 samples are listed. It can be seen that between 2 and 17 bootstrap samples showed a higher Goodness-of-Fit value than the real data. Therefore, the assumption that the data were generated by the specified HYBRID model is not falsified.

Parametric Bootstrap estimates for Goodness of Fit:

No.: Satlik LogLik LR CressieRead Pearson X² FT

1	-14836.781	-24541.457	19409.352	190450.757	1069253.060	12984.9860
2	-14883.365	-24563.595	19360.461	188222.734	1073851.379	13063.5891
3	-14898.684	-24566.254	19335.140	183807.288	1048087.972	13191.3068
4	-14940.153	-24682.554	19484.801	187564.883	1069896.114	13257.5533
5	-14927.206	-24750.710	19647.008	188408.323	1059873.202	13285.5702
6	-14891.243	-24556.653	19330.819	185447.350	1051618.126	13143.8871
7	-14888.905	-24668.821	19559.833	189667.564	1060352.931	13162.2576
8	-14869.646	-24694.736	19650.179	191346.730	1054225.383	13145.5455
9	-14864.434	-24710.431	19691.993	189267.848	1041317.125	13196.0404
10	-14885.997	-24752.450	19732.907	191276.684	1066221.749	13251.3002

```
11 -14909.753 -24742.399 19665.291 189308.678 1050411.606 13234.0416
12 -14903.741 -24674.261 19541.040 189206.466 1072625.593 13213.9483
13 -14870.321 -24462.530 19184.418 183339.043 1064587.444 13102.2003
14 -14868.631 -24076.854 18416.446 180624.497 1193808.517 12879.2104
15 -14817.184 -24093.470 18552.571 178027.561 1050349.133 12828.3274
16 -14850.098 -24197.824 18695.452 184627.623 1177492.758 12930.2590
17 -14857.015 -24029.929 18345.826 173529.495 1062309.443 12879.3370
18 -14865.694 -24052.424 18373.459 182297.386 1273164.429 12868.3238
19 -14846.662 -23975.028 18256.733 179065.513 1209296.487 12802.8427
20 -14832.243 -23909.898 18155.309 166976.243 1020880.850 12800.8061
```

```
Z: 0.668 1.018 -0.593 -0.4293
```

```
P(X>Z): 0.252 0.154 0.724 0.6661
```

```
Mean: 19119.452 184623.133 1088481.165 13061.0666
```

```
Stdev: 553.577 6255.476 65621.729 166.4833
```

```
p-values (emp. PDF): 0.350 0.100 0.850 0.6000
```

For very sparse data tables and small or moderate sample sizes, the bootstrap procedure should be used only for the Cressie–Read and the Pearson Chi–Square statistics (v. Davier, 1996). None of the four statistics in the table above rejects the model in this example. But nevertheless, both the FT and the Likelihood Ratio statistics can not be recommended, as a very large sample size seems to be necessary to make the bootstrap reliable for these statistics.

References:

Andrich, D.: A rating formulation for ordered response categories, *Psychometrika*, 1978, 43, 561–573.

Andrich, D.: Application of a psychometric rating model to ordered categories which are scored with successive integers. *Applied Psychological Measurement*, 1978, 2, 4, 581–594.

Andrich, D.: An extension of the Rasch model for ratings providing both location and dispersion parameters, *Psychometrika*, 1982, 47, 105–113.

Bozdogan, H.: Model Selection and Akaike's Information Criterion (AIC): The General Theory and its Analytical Extensions. *Psychometrika*, 1987, 52, 3, 345–370.

Cressie, T. R. C. & Read, N. A. C.: Multinomial Goodness-of-Fit Statistics. *Journal of the Royal Statistical Society Series B*, 46, 440–464, 1984.

Davier, von, M.: Neue Probabilistische Testmodelle und ihre Anwendungen. Poster präsentiert auf dem 39. Kongreß der Deutschen Gesellschaft für Psychologie, 1994.

Davier, von, M.: Methoden zur Prüfung probabilistischer Testmodelle. IPN Schriftenreihe, Band 157, 1997.

Davier, von M. & Rost, J.: Self Monitoring – A Class Variable? In: Rost, J. & Langeheine, R. (Eds.) Applications of Latent Trait and Latent Class Models. Proceedings of the IPN Symposium in Sankelmark 1994.

Davier, von M. & Rost, J.: Polytomous Mixed Rasch Models. Chapter 20 in: Fischer, G. & Molenaar, I. (Eds.): Rasch Models: Foundations, Recent Developments and Applications. New York: Springer, 1995.

Dempster, A. P., Laird, N. M. & Rubin, D. B.: Maximum Likelihood from Incomplete Data via the EM-Algorithm. Journal of the Royal Statistical Society Series B, 39, 1–38, 1977.

Drasgow, F., Levine, M., Williams, E.: Appropriateness measurement with polychotomous item response models and standardized residuals, British J. of Math. and Stat. Psychology, 1985, 38, 67–86.

Efron, B.: The Jackknife, the Bootstrap and other Resampling Plans. SIAM: Society for Industrial and Applied Mathematics, 1982.

Efron, B. & Tibshirani, R. J.: An Introduction to the Bootstrap. Monographs on Statistics and Applied Probability 57. New York: Chapman & Hall, 1993.

Everitt B.S. & Hand D.J.: Finite Mixture Distributions. London: Chapman & Hall, 1981.

Giegler, H. & Rost, J.: (1990) Ordinale manifeste Variablen — Nominale latente

Variablen — Latent Class Analyse für ordinale Variablen. In: Faulbaum, F., Haux, R. & Jöckel, K.-H. (Hrsg.) *SoftStat'89 Fortschritte der Statistik Software 2*. Stuttgart: Gustav Fischer.

Hojtink, H. & Boomsma, A.: On Person Parameter Estimation in the Dichotomous Rasch Model. Chapter 4 in: Fischer, G. H. & Molenaar, I. (eds.): *Rasch Models: Foundations, Recent Developments and Applications*, New York: Springer, 1995.

Langeheine, R. & Rost, J.: Latent Trait and Latent Class Models. New York: Plenum, 1988.

Lazarsfeld, P. F.: The Logical and Mathematical Foundations of Latent Structure Analysis. In: S.A.Stouffler et al.: *Measurement and Prediction* Princeton: Princeton University Press, 1950.

Lazarsfeld, P. F. & Henry, N. W.: Latent Structure Analysis. New York: Houghton Mifflin, 1968.

Masters, G.: A Rasch model for partial credit scoring, *Psychometrika*, 1982, 47, 147–174.

McLachlan, G. J. & Basford, K. E.: Mixture Models. New York: Marcel Dekker, 1988.

Meehl, P.E.: Factors and Taxa, Traits and Types, Differences of Degree and Differences in Kind. *Journal of Personality* , 60, 1, 117–174, 1992.

Mooney, C. Z. & Duval, R. D.: Bootstrapping — A Nonparametric Approach to Statistical Inference. Sage Series: Quantitative Applications in the Social Sciences, Vol. 95, 1993.

Rasch, G.: Probabilistic Models for some Intelligence and Attainment Tests. Copenhagen: Denmarks Paedagogiske Institut, 1960.

Read, N. A. C. & Cressie, T. R. C.: Goodness-of-Fit Statistics for Discrete Multivariate Data. Springer Series in Statistics. New York: Springer, 1988.

Rost, J.: Rating Scale Analysis with Latent Class Models. *Psychometrika*, 53, 327–348, 1988.

Rost, J. : Measuring attitudes with a threshold model drawing on a traditional scaling concept, *Applied Psych. Measurement*, 1988, 12, 397–409.

Rost, J. : Rasch models in latent classes: An integration of two approaches to item analysis, *Applied Psychol. Measurement*, 1990, 14, 271–282.

Rost, J.: A logistic mixture distribution model for polychotomous item responses. *The British Journal for Mathematical and Statistical Psychology*, 1991, 44, 75–92.

Rost, J., Carstensen, C. & Davier, von M.: An Application of the Mixed Rasch Model to Personality Questionnaires. In: Rost, J. & Langeheine, R. (Eds.) *Applications of Latent Trait and Latent Class Models. Proceedings of the IPN Symposium in Sankelmark 1994.*

Rost, J. & Davier, von, M.: A conditional Item Fit Index for Rasch Models, Applied Psychological Measurement, 1994.

Rost, J. & Davier, von, M.: Mixture Distribution Rasch Models. Chapter 14 in: Fischer, G. & Molenaar, I. (Eds.): Rasch Models: Foundations, Recent Developments and Applications. New York: Springer, 1995.

Rost, J. & Georg, W.: Alternative Skalierungsmöglichkeiten zur klassischen Testtheorie am Beispiel der Skala "Jugendzentrismus". Zentral Archiv Informationen, 28, 52–74, 1991.

Rost, J. & Langeheine, R.: A guide through latent structure models for categorical data. In: Rost, J. & Langeheine, R. (Eds.) Applications of Latent Trait and Latent Class Models. Proceedings of the IPN Symposium in Sankelmark 1994.

Tarnai, C., Rost, J.: Identifying aberrant response patterns in the Rasch model – The Q index. Soz.wiss. Forschungsdokumentationen, Munster, 1990.

Warm, T. A.: Weighted likelihood estimation of ability in item response models. *Psychometrika*, 1989, 54, 427–450.

Wright, B. D.: Afterword, in: Rasch, G., Probabilistic models for some intelligence and attainment test, Chicago, 1980.

Wright, B. D., Masters, G.: Rating scale analysis, Chicago MESA Press, 1982.

Wright, B. D., Stone, M.: Best test design, Chicago, MESA Press, 1979.

Yamamoto, K.: A Hybrid model of IRT and latent class models. *ETS Research Report (RR-89-41)*, Princeton, NJ: Educational Testing Service, 1989.

Yamamoto, K. & Everson, H.: Detecting Speededness using a "Hybrid" IRT Latent Class Model. *In: Rost, J. & Langeheine, R. (Eds.) Applications of Latent Trait and Latent Class Models.* Proceedings of the IPN Symposium in Sankelmark 1994.
