R&D Notities 2002-2

# Correlations Between Latent Variables
# The Programs CORDIM and FMERGE

N.D Verhelst

N.H. Veldhuijzen

Cito Arnhem

# Correlations Between Latent Variables
# The Programs CORDIM and FMERGE

N.D. Verhelst
N.H. Veldhuijzen

Citogroep
Arnhem, oktober 2002

# 1 Introduction

In the analysis of test data, the most popular IRT models are the ones that assume a unidimensional latent variable. Sometimes, however, the correlations between two or more latent dimensions are needed. The present report discusses a possible estimation procedure.

One of the programs that does estimate correlations is the program MULTI by Frans Kamphuis. The procedure is a two-step procedure. In the first step the item parameters are estimated using CML-estimation for two or more scales. In the second step the item parameters are fixed, and the vector of means and the covariance matrix is estimated, using a Bayesian approach. Although the program MULTI seems to do well in practice, there are a couple of problems associated with its use:

1. The overhead for the user is quite large if for each dimension an incomplete design has been used.

2. In the estimation procedure of MULTI the simplifying assumption is made that for each dimension the posterior distribution of the latent variable given the score on the test is normal. If long tests are used and extreme scores are not very frequent, it is believed that violation of the assumption of normality will not do much harm. In cases, however, where most of the test booklets contain very few items, MULTI could give quite biased results. To see if this is really the case, it is useful of course to estimate the covariances using software which guarantees consistent results. The program CORDIM to be introduced here guarantees consistent results.

The procedure followed for the program CORDIM is a three-step procedure. In the first step item parameters are estimated for each dimension separately using CML. This is the same as with MULTI. In the second step, mean and standard deviation for each dimension separately are estimated by the program SAUL. The third step is the estimation of the correlation matrix between the dimensions. The program CORDIM only performs the third step, and it does so for each pair of dimensions separately

In the next section some algebraic derivations will be given on how the correlation can be estimated. In section 3 the users manual for the program CORDIM is presented.

## 2 Estimation of a latent correlation

Suppose for some scale the item parameters have been estimated using OPCML. For some (binary) item $i$ the probability of a correct response is given by

$$P(X_i = 1|\theta) = \frac{\exp[a_i(\theta - \beta_i)]}{1 + \exp[a_i(\theta - \beta_i)]} \tag{1}$$

In the second step, using SAUL, mean and standard deviation of the latent variable are estimated, giving as results $\mu$ and $\sigma$ respectively. Now we transform the latent variable and the item parameters as follows:

$$\begin{aligned} \theta^* &= \frac{\theta - \mu}{\sigma} \\ \beta_i^* &= \frac{\beta_i - \mu}{\sigma} \\ a_i^* &= a_i\sigma \end{aligned} \tag{2}$$

It follows immediately that the OPLM is valid now for the transformed variable with transformed parameters, but the main result is that the transformed latent variable is standardized, i.e., it has a mean of zero and a standard deviation of one. Notice that in using SAUL, one uses the assumption that the latent variable is normally distributed, and in CORDIM it will be assumed that all latent variables follow the multivariate normal distribution.

With two variables, $\theta_1$ and $\theta_2$, say, inferred by response patterns $\mathbf{x}$ and $\mathbf{y}$ respectively, the likelihood of the joint occurrence of $\mathbf{x}$ and $\mathbf{y}$ is given by

$$P(\mathbf{x},\ \mathbf{y}) = \int\int f(\mathbf{x}|\theta_1)\ f(\mathbf{y}|\theta_2)\ g(\theta_1, \theta_2)\ d\theta_2\ d\theta_1 \tag{3}$$

where

$$f(\mathbf{x}|\theta_1) = c_x \exp(s_x\theta_1) P_{0x}(\theta_1) \tag{4}$$

where $c_x$ depends only on the item parameters in the first scale, $s_x$ is the (weighted) score on the first scale, and $P_{0x}(\theta_1)$ is the probability of getting a score of zero on the first scale. $f(\mathbf{y}|\theta_2)$ is defined similarly, and $g(\theta_1, \theta_2)$ is the density function of the bivariate normal distribution, which has only one parameter since both variables are standardized.

Equation (3) can be written in an alternative way as

$$P(\mathbf{x},\ \mathbf{y}) = \int f(\mathbf{x}|\theta_1)\ g(\theta_1) \int f(\mathbf{y}|\theta_2)\ g(\theta_2|\theta_1)\ d\theta_2\ d\theta_1 \tag{5}$$

where

$$g(\theta_1) = \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{\theta_1^2}{2}\right] \tag{6}$$

and

$$g(\theta_2|\theta_1) = \frac{1}{\sqrt{2\pi}\sqrt{1-\rho^2}} \exp\left[-\frac{(\theta_2 - \rho\theta_1)^2}{2(1-\rho^2)}\right] \tag{7}$$

Now we transform variables in the integrals of (5):

$$z_1 = \frac{\theta_1}{\sqrt{2}} \tag{8}$$

and

$$z_2 = \frac{\theta_2 - \rho\theta_1}{\sqrt{2(1-\rho^2)}} \tag{9}$$

whence we can write

$$P(\mathbf{x},\ \mathbf{y}) = \frac{1}{\pi} \int f(\mathbf{x}|\theta_1(z_1))\ \exp(-z_1^2) \int f(\mathbf{y}|\theta_2(z_1, z_2))\ \exp(-z_2^2)\ dz_2\ dz_1 \tag{10}$$

where $\theta_1$ is now a function of $z_1$ and $\theta_2$ is a function of both $z_1$ and $z_2$. These functions can easily be written explicitly by solving (8) and (9):

$$\begin{aligned} \theta_1 &= \sqrt{2} z_1 \\ \theta_2 &= \sqrt{2}\left[\rho z_1 + z_2\sqrt{(1-\rho^2)}\right] \end{aligned} \tag{11}$$

Equation (10) is in a form which can easily be approximated by Gauss-Hermite quadrature. If one uses for both integrals the same number of quadrature points, $Q$ say,then we have as a practical result

$$P(\mathbf{x},\ \mathbf{y}) \approx \frac{1}{\pi} \sum_i^Q f(\mathbf{x}|\theta_1(z_i)) \sum_j^Q f(\mathbf{y}|\theta_2(z_i, z_j)) w_j w_i \tag{12}$$

where $z_i$ symbolizes the $i$-th quadrature point and $w_i$ its corresponding weight.

To find the derivative (with respect to $\rho$) of (12), notice that $\rho$ only appears in $f(\mathbf{y}|\theta_2(z_i, z_j))$. To find the derivative we apply the chain rule. First, it is easy to check that

$$A(\rho) \triangleq \frac{d}{d\rho}\theta_2 = \sqrt{2}\left[z_1 - z_2 \frac{\rho}{\sqrt{(1-\rho^2)}}\right] \tag{13}$$

Next, using (4), we have that

$$\frac{d}{d\theta_2} f(\mathbf{y}|\theta_2) = c_y \left(s_y \exp(s_y\theta_2) P_{0y}(\theta_2) + \exp(s_y\theta_2) \frac{d}{d\theta_2} P_{0y}(\theta_2)\right) \tag{14}$$

Notice that

$$P_{0y}(\theta_2) = \prod_i P(Y_i = 0|\theta_2) = \prod_i P_{0i}(\theta_2). \tag{15}$$

In order to have a result which applies to polytomous items as well, we take the derivative of $P_{0i}(\theta_2)$ with item $i$ a polytomous item:

$$P_{0i}(\theta_2) = \frac{1}{1 + \sum_{j=1} \exp\left[a_i \left(j\theta_2 - \eta_{ij}\right)\right]} \tag{16}$$

The derivative w.r.t. $\theta_2$ is easily found to be

$$\begin{aligned} \frac{d}{d\theta_2} P_{0i}(\theta_2) &= \frac{-\sum_{j=1} j a_i \exp\left[a_i \left(j\theta_2 - \eta_{ij}\right)\right]}{\left\{1 + \sum_{j=1} \exp\left[a_i \left(j\theta_2 - \eta_{ij}\right)\right]\right\}^2} \\ &= -P_{0i}(\theta_2) \times E(Y_i|\theta_2) \end{aligned} \tag{17}$$

So we find that

$$\begin{aligned} \frac{d}{d\theta_2} P_{0y}(\theta_2) &= \frac{d}{d\theta_2} \prod_i P_{0i}(\theta_2) \\ &= \sum_i E(Y_i|\theta_2) P_{0i}(\theta_2) \prod_{j \neq i} P_{0j}(\theta_2) \\ &= E(s_y|\theta_2) P_{0y}(\theta_2) \end{aligned} \tag{18}$$

Substituting (18) into (14) we find that

$$\frac{d}{d\theta_2} f(\mathbf{y}|\theta_2) = c_y \left[\left(s_y - E(s_y|\theta_2)\right) f(\mathbf{y}|\theta_2)\right] \tag{19}$$

and as a final result we find that

$$\frac{d}{d\rho} P(\mathbf{x},\ \mathbf{y}) \approx \frac{1}{\pi} \sum_i^Q f(\mathbf{x}|\theta_1(z_i)) \sum_j^Q A(\rho) B(\mathbf{y}, \theta_2) f(\mathbf{y}|\theta_2(z_i, z_j)) w_j w_i \tag{20}$$

where

$$B(\mathbf{y}, \theta_2) \triangleq s_y - E(s_y|\theta_2) \tag{21}$$

Using maximum likelihood estimation of $\rho$, the contribution of the response pattern $(\mathbf{x},\ \mathbf{y})$ to the gradient is

$$\frac{d}{d\rho} \ln P(\mathbf{x},\ \mathbf{y}) = \frac{\frac{d}{d\rho} P(\mathbf{x},\ \mathbf{y})}{P(\mathbf{x},\ \mathbf{y})} \tag{22}$$

so that it is easily seen that $\pi$, $c_x$ and $c_y$ cancel.

# 3 The programs CORDIM and FMERGE

The program CORDIM computes the correlations for all pairs of dimensions specified by the user. Since the data (the response patterns) can be collected in (complicated) incomplete designs, the major part of the code of CORDIM is bookkeeping. In the next subsection the input for CORDIM is discussed. One of the input files for CORDIM is a special data file where for each student the score is given for all of the dimensions. The construction of this data file is a serious problem on its own. To help the user, a special utility, FMERGE, has been constructed to build this data file. A special subsection is devoted to this utility.

## 3.1 The program CORDIM

The program CORDIM needs four kinds of information: (1) the item parameter estimates for all items involved in each dimension; (2) design information on the original data collection, also for each dimension; (3) marginal mean and standard deviation for each dimension and (4) a special data file which contains essentially the weighted score of each student on each of the dimensions.

Since most of the needed information resides in files which exist already when the program is run, the primary input for CORDIM consists of a 'job definition file' (JDF), where the names of the existing files are specified.

### 3.1.1 Parameter estimates and design information

It is assumed that the scales representing the dimensions are separately calibrated using OPCML. The item parameter estimates are stored in the *.PAR file for each dimension. We will use the name DIM$i$.PAR for the PAR file for dimension $i$. The total number of dimensions will be denoted by NDIM.

The design information is stored in the *.SCR file for each dimension. As a generic name we will use DIM$i$.SCR. Notice that CORDIM requires that the DIM$i$.SCR is exactly the one with which the DIM$i$.PAR file has been created. This means that it is not possible to create a PAR file, then modify the corresponding SCR file (for example by setting some items off), and than feed the PAR file and the modified SCR file to CORDIM. The reason for this is the following: the joint information in the PAR and SCR files is redundant in some respects (e.g. total number of items, discrimination indices), and this redundancy is used to check if PAR and SCR belong together.

### 3.1.2 Means and standard deviations

For each dimension an estimate of the mean and standard deviation of the latent variable must be supplied by the user. Although one can get these estimates by running OPMML, it is advisable not to do so, because OPMML usually gives other estimates of the item parameters than OPCML. An easy way to get estimates based on the CML-estimates of the item parameters is to run SAUL with an empty model. The output of SAUL is written on a file with extension SLF, where the estimate of the mean (labeled as 'additive constant') and the standard deviation are given.

With an empty model, the SLF file contains two estimates of the mean and the standard deviation: one on the 'original scale', i.e., the scale where the unit is defined by the discrimination indices specified by the user, and one on a transformed scale, where the geometric mean of the transformed discrimination indices equals one. The estimates needed for CORDIM are the ones on the original scale.

### 3.1.3 The data

CORDIM does not use the data file which was used for the calibration. The user must prepare a special data file (generic name: DATAFILE) which contains for each student NDIM pairs of numbers, each pair corresponding to a dimension. The first number of pair $i$ is the (absolute) booklet number the student got when answering items of dimension $i$. If the student has not been exposed to such items, the number must be zero. The second number of the pair is the weighted score for dimension $i$. If the first number is zero, the second number is arbitrary (but should be a number).

Each record in the file corresponds to a student. The numbers to be read are treated as integers (and hence must not contain decimal points). A record may contain additional information, such as an identification key for the student. If the NDIM pairs of numbers appear as the first information in the record (and each number separated by the previous one by a comma or a number of blanks), the data file can be read using free format; otherwise a format (applicable to all records) has to be specified (in the job definition file; see below). If zeros are not written, but blanks are used instead, formatted reading is compulsory (in free format a series of consecutive blanks is equivalent to a single blank; in formatted reading blank fields are read as zero).

In estimating the correlation between dimension $i$ and $j$, all records having booklet zero on dimension $i$ or $j$ are skipped. Notice that this is perhaps the trickiest point in the use of CORDIM: if students are not randomly assigned

to booklets (and to dimensions) the estimate of the correlation may be biased.

### 3.1.4 The job definition file (JDF)

The job definition file contains a number of records which are described in turn. The records of the job definition file are read in free format. Records contain either numeric or alphanumeric information (a file name or a format). Records containing alphanumeric information must not contain extra information; numeric records may contain trailing extra information, separated by at least one blank from the numeric input.

1. First record: NDIM, the number of dimensions.
2. Second record: an integer number. If zero the data from DATAFILE will be read in free format, otherwise the format to be used in reading DATAFILE is specified in the next record.
3. Third record (depending on value of previous record): the format for reading DATAFILE. Notice that the format must be included in parentheses. Do not use the word FORMAT. If record 2 contains a zero, omit this record (do not supply an empty record.) The format should be a valid Fortran style format expression.
4. fourth (or third) record: the name of DATAFILE (inclusive path if necessary). (max. 255 characters)
5. fifth (or fourth) record: the name of the output file (inclusive path if necessary).(max. 255 characters)
6. NDIM blocks of four records, holding respectively
   (a) a label for the dimension. First 16 characters are used
   (b) the name of the SCR file (DIM*i*.SCR) for this dimension, inclusive path if necessary (max. 255 characters)
   (c) the name of the PAR file (DIM*i*.PAR) for this dimension, inclusive path if necessary (max. 255 characters)
   (d) mean and standard deviation for this dimension (decimal numbers)

More than NDIM blocks of four records may be specified; only the first NDIM blocks are read. Notice that the order of the NDIM blocks must

correspond to the order of the dimensions in the DATAFILE. If more than NDIM blocks are present, only the first NDIM are read.

The program CORDIM is a console application (to be run in a DOS box). The command to run the program is

CORDIM JDF

where JDF is the name of the job definition file.

### 3.1.5 Simulation

The program offers a limited possibility to carry out simulation studies. If NDIM equals 2, the user will be asked if simulation runs are wanted. Entering zero will make the program estimate the correlation from the observed data; entering one will prompt the user to specify the correlation $\rho_t$ between the two dimensions and the number of replications (NREP) to be carried out. If simulations are wanted, the following procedure is carried out for each of the NREP replications

1. The design as specified in the *.SCR files is completely followed. The number of students simulated is exactly the same as found in the DATAFILE.

2. For each student who answered to items pertaining to both dimensions, a bivariate latent observation $(\theta_1, \theta_2)$ is drawn from the standardized bivariate normal distribution with correlation $\rho_t$.(Within the program CORDIM the item parameters are transformed in such a way that the marginal distributions of the latent variables are the standard normal distributions).

3. Two response patterns are simulated for this artificial student . Notice that

   - the booklets (for both dimensions) are the same as the booklets for the actual real student.
   - the item parameters used are the item parameter estimates resulting from the OPLM analyses on the real data.

4. The two weighted scores resulting from the generated responses patterns are substituted for the observed scores.

The simulated scores are used to estimate the correlation. Notice that the item parameters are not re-estimated from the simulated data. This implies that the simulation results are not influenced by the estimation errors in the item parameter estimates.

### 3.1.6 The output of CORDIM

CORDIM writes its output on a file specified in the job definition file. Apart from some echo of the input, the output consists of a square matrix (NDIM×NDIM) with the following information:

1. on the main diagonal (cell $(i, i)$): the total number of students having given information on the $i$-th dimension.
2. above the main diagonal (cell $(i, j)$): the number of students having given information on dimensions $i$ and $j$.
3. below the main diagonal (cell $(i, j)$): the estimate of the correlation $\rho_{ij}$.

In case simulations are done, the output file contains NREP+1 records. In the first record the value of NREP and $\rho_t$ is given; the subsequent records contain the replication number and the value of the estimated correlation.

## 3.2 The utility FMERGE

The utility FMERGE, a Windows application, is meant to be of help in constructing the file DATAFILE discussed in section 3.1.3. To make use of this utility, however, a certain amount of manual work has to be done by the user. In the first subsection to follow, a procedure is described how to prepare the input files for the utility FMERGE; in the next subsection the utility FMERGE itself is described.

### 3.2.1 Preparation of the input files for FMERGE

The usual situation in estimating the correlations between a number of latent variables will imply that an OPLM analysis has been carried out for each of the dimensions separately. For each of the dimensions, one needs the booklet number and the weighted score for each student. An easy way to construct the input files needed for FMERGE is to run OPLAT for each of the dimensions. The steps to be taken are described in turn

1. Run OPLAT for each dimension. Make sure that part of the data file is copied in the output file (default extension: .LAT). The minimal part that must be copied is the (unique) student identification. In the screen of OPLAT, choose DATA, enter 'Y' for copy, and select the appropriate columns.
2. Edit the *.LAT files by

(a) deleting the header records before the table of results (one record per student) and all the trailing records after that table;

(b) deleting all information in the table, except the booklet number, the weighted score and the student identification. Make sure that the student identification (ID) occupies the same number of positions for each dimension. It is not necessary that all files have the same format, as long as the identification does not precede booklet and score information. If the *.LAT files have different formats, free format must be specified in FMERGE. Notice that one cannot use free format reading with FMERGE if the identification field contains blanks.

(c) sorting the resulting table on ID (ascending.) This step is essential for a valid result of FMERGE. (If sorting is omitted, FMERGE will detect this and stop.)

(d) saving the edited and sorted table. The saved files (one per dimension) are the input files for FMERGE, and can be read with free format (see below).

### 3.2.2 Using FMERGE

It is easy to use, but the user should be meticulous in specifying the data for the utility, as there will be just the minimum amount of error checking. Essentially, only three errors will be trapped. The first is the incorrectness of the rank numbers supplied in Step 4 (see below); this error can be repaired on the spot. The second is the detection of duplicate studenIDs in an input file. This error will abort the utility as soon as the first of such errors obtains, as the user has to correct the input file in question. The third error is the detection of a not appropriately sorted input file The error message shown on the screen will also be written into the log file. The utility solicits information in four steps, which will be discussed now.

1. Step 1: After the welcome message, the user is asked whether the input files are to be read in free format or under format control. The choice is made by picking one of the two radio buttons marked "free format" and "fixed Fortran format" respectively. "Free format" is the default setting. "Free format" means that: - in each input record, the data must be in the order *booklet for dimension i, score on this booklet, studentID*; - if it is desired to have any other data in the record, these data should follow the studentID and be separated from it by at least one blank; these extra data will be ignored; - all numbers and strings

should be separated from each other by at least one blank. If the users opts for format control, an entry field comes up containing two parentheses. Within these parentheses, a valid Fortran-style format expression should be entered. This format expression: - should describe one physical record only (that is, the use of slashes is not allowed); - should be such that the data are read in the order *booklet number, score, ID*; - the format descriptor for the studentID should be "A$n$", where "$n$" should be the length of the string containing the studentID. No format validation is performed. Move on by pushing the OK button.

2. Step 2: A standard Windows file selector pops up, in which the input files (the edited and sorted *.LAT files) can be selected; the input files should all be in the same folder (directory). The number of files selected determine the number of dimensions to be used in the program CORDIM. Move on by pushing the OPEN button.

3. Step 3: As Windows may jeopardize the order in which the user selects the input files, a dialogue will appear containing two columns. The left column contains the names of the files selected, in lexicographical order, the right column displays the default rank numbers of the dimensions. The user can change these rank numbers, specifying the order in which the files are to be read. Use only the *Cursor-Up* and *Cursor-Down* keys to navigate between the cells in the right column. The rank numbers entered determine the order of the information in the output file. The chosen order is written on the FMERGE.LOG file. Move on by pushing the OK button.

4. Step 4: a standard Windows file selector pops up, asking to supply the name of the output file DATAFILE (see section 3.1.3). Enter a file name and move on by pushing the SAVE button. After this, the program immediately starts the processing of the input files. The DATAFILE will be saved in the same folder (directory) were the input files reside. As soon as all input files have been merged into the DATAFILE, the program will be closed. The DATAFILE will reside in the same folder (directory) as the input files. Note that the utility will store zeroes for both the booklet number and the test score in case a student did not obtain a score on items belonging to a dimension (see section 3.1.3).

The file FMERGE.LOG will reside in the same folder (directory) as the input files. The log file contains the names of the input files in the order they have been opened, the name of the output file (DATAFILE), and the Fortran-style format expression you can use for the program CORDIM. The

data file produced by FMERGE, however, is suited to be read by CORDIM in free format. Any error messages issued during the execution of the utility will be written to the log file, too.

# 4 Some test results.

Two tests have been carried out with CORDIM, one with artificial data and one with real data. These are described in turn

For the artificial data, 1000 students randomly drawn from a standard normal distribution responded to 30 items simulating the Rasch model. The item parameters of the first 15 items were (4*-1, 7*0, 4*1). The last 15 items had the same item parameters as the first 15. The test was split in two (strictly) parallel tests of 15 items, and each test was analyzed (using OPCML). For the two tests the correlation between the latent variables (which is one) was estimated. The result was equal to one. (The program CORDIM stops if the current value of the correlation is larger than 0.9995, and the gradient indicates that the estimate should be larger than this.).

In the national assessment (PPON) of 2002, data were collected on physics and technical insight in an incomplete design with ten booklets. Each booklet was administered to about 210 students, but only two of the then booklets had items in both domains. The scales were calibrated for physics and technical insight separately (each of the two dimensions yielding data of over 1200 students). The correlation was estimated on the data of the two booklets having items on physics as well as on technical insight. In all, these two booklets were administered to 422 students. The first booklet contained 15 physics' and 14 technical items; for the second booklet the number of items were 14 and 13 respectively; the two booklets did not have overlapping items. The correlation between the Warm-estimates of these 422 students on both dimensions was 0.42, the estimated correlation between the latent variables was 0.80.

As a partial check of the program CORDIM, two short simulations has been carried out. A special version of the program allows (in the case of two dimensions) the user to overwrite the observed scores and replace them by a simulated score based on a bivariate latent variable $(\theta_1, \theta_2)$ drawn from a bivariate (standardized) normal distribution with a specified value of the correlation. These simulated scores are then used to reestimate the correlation. Notice that data are generated using the same design as the observed data, but item parameters, and mean and standard deviation for the dimensions are not reestimated from the simulated data; the values resulting from the unidimensional analysis of the real data are kept fixed. Repeating this pro-

cedure a number of times gives the opportunity to estimate expected value and standard error of the estimator of the correlation coefficient.

The results of this simulation are summarized in Table 1. The number of replications for each simulation study was 100.

Table 1. Results of the simulation (NREP = 100)

| | artificial ($n = 1000$) | | | real ($n = 422$) | | |
|---|---|---|---|---|---|---|
| $\rho$ | 0.0 | 0.4 | 0.8 | 0.0 | 0.4 | 0.8 |
| mean | −0.002 | 0.402 | 0.801 | −0.012 | 0.410 | 0.785 |
| SD | 0.048 | 0.039 | 0.025 | 0.104 | 0.095 | 0.071 |

A few comments on Table 1.

1. On the average the theoretical correlation is nicely reproduced by the means of the estimates.

2. The standard deviations may be considered as disappointingly large. The standard error of the product-moment correlation for the case $\rho = 0$ is of the order $1/\sqrt{n}$, $(1/\sqrt{1000} = 0.0316)$. Here the standard errors are larger, but one should realize that the variables themselves are not observed.

3. The standard deviations decrease with increasing value of $\rho$

Of course, the value of the standard error will depend on many features of the data, like the balance in the design, the total number of observations, the number of items per booklet, the information per booklet and the information (on $\rho$) per pair of booklets that are observed jointly. All these interesting things can be investigated with the program CORDIM.