

# PT

## Psychometric Toolbox

### USER'S GUIDE

Prepared by:

Pere J. Ferrando

David Navarro-González

Urbano Lorenzo-Seva

Please reference this document as:

Ferrando, P.J., Navarro-González, D. & Lorenzo-Seva, U. (2016). *Psychometric Toolbox User's Guide. Technical Report*. Universitat Rovira i Virgili. Tarragona.

Available at: <http://psico.fcep.urv.cat/utilitats/PsychometricTools/index.html>

Last edit: July 2019

# Contents

1. Introduction and features .....	
2. Installation and setup .....	
3. Entering and importing data .....	
4. The data-preprocessor sub-program .....	
5. The GUTTMAN sub-program.....	
6. The Classical Item Analysis (CIA) sub-program .....	
7. The Item Factor Analysis (IFA) sub-program .....	
8. The Scoring and Norming sub-program .....	
9. The Item Response Theory-Basics (IRT-B) sub-program.....	
10. The Multiple-Choice Analysis (MCA) sub-program .....	

## 1. Introduction and features

The Psychometric Toolbox (PT) is a user-friendly, non-commercial package mainly intended to be used for instructional purposes in introductory courses of educational and psychological measurement, psychometrics and statistics. The PT package is organized in seven separate modules or sub-programs: (1) Data preprocessing, (2) Guttman scaling, (3) Classical item analysis, (4) Item factor analysis, (5) Scoring and Norming, (6) Item Response Theory analysis, and (7) Multiple-Choice analysis. Because they have been designed for instructional use, these modules are intended to (a) be very easy to use, (b) provide clear and well explained results, and (c) make use of graphical displays whenever possible. Apart from its wide scope, the PT is quite versatile and implements features that are not usually found in programs of this type. Thus, the preprocessing module allows the user to replace missing values, perform split-half test partitions, create sub-scales, randomly split the sample for cross-validation purposes, perform Jackknife re-sampling, and create automatically parallel forms. The item factor analysis computes the omega coefficient with its confidence intervals and displays one and two canonical solutions in order to detect violations of unidimensionality. The Scoring and Norming program provides confidence intervals for the transformed scores and creates the test normative table based on the user's specifications.

The program uses Excel modules (one for each sub-program) written in VBA Excel code. So, to run the PT you need to have VBA language installed on your computer. This language is included in Microsoft Office Professional.

## 2. Installation and setup

Psychometric Toolbox is a collection of Excel modules that does not require installation. They were developed in Microsoft Office Excel 2010 and can be downloaded from the official web site:

<http://psico.fcep.urv.cat/utilitats/PsychometricTools/index.html>

The program is distributed as ZIP file (.zip) in the main webpage. The user has to decompress the Excel macro inside the .zip file, and place it in any valid place of the drive. The datasets used in each example are available in the corresponding section of the website.



Figure 2. Display of the Scoring and norming module after the responses of 11 individuals to 7 items have been entered

To enter data, users can: (a) type the data in the “Data” sheet themselves; (b) paste in the data from some other Excel file; or (c) import a text file. Three functions can be used to import a dataset from text files. They are present in all the modules of the Psychometric Toolbox (except in Scoring multiple-choice tests module) and have an effect mainly on the Excel sheet labeled “Data”:



Clear  
Data

This function deletes all the data from the “Data” sheet. It is useful as a starting point to ensure that the data to be preprocessed does not get mixed up with previous data in the sheet.



Clear  
Results

This function deletes all the data in the “PreprocessedData” sheet. It is useful as a starting point to ensure that the outcomes of previous results do not get mixed up with the outcomes of newly read data.



Import  
Text File

This function imports data that is stored in a text file. The user must inform about the character (if any) that separates the columns in the text file. Typically, a space character or a tab character are used. Other common separators are a comma or a semicolon.



Select, Sort and  
Reverse Items

This function helps to select a particular set of items in a particular order. The user decides which items to include in the set and the order. The selected items are copied into the “PreprocessedData” sheet in the order defined. Optionally, the user can select which items should be reversed, if necessary. Finally, an extra column can be computed as the overall score (i.e., the addition of the responses to the selected items).

---

There are also two extra functions that allow the user to further analyze the data with Excel functions, and to save the Excel book as an Excel file. These two functions are also present in the modules of the Psychometric Toolbox:



Show  
Tabs

This function shows (or hides) the typical ribbons in Excel. When the typical ribbons are shown, the user can use any option in Excel to analyze the data.



Save

This function saves the current Excel book as an Excel file.

---

Finally, there are also three functions related to the Psychometric Toolbox itself:

---



Help

This function show some brief tips for guiding the user in the usage of the current module.



About

It displays a window containing the names and institution of the authors of Psychometric Toolbox.



Report  
bug

A function for reporting a bug that will redirect you to a web form where you can provide the authors some info about it. Your comments will help us improve the program so we encourage the users to use this button whenever their experience some problems with the Psychometric Toolbox.

---

## **4. The data preprocessor sub-program**

### 4.1 Summary

The aim of the DATA PREPROCESSOR sub-program is to explore and prepare the raw data to make it amenable to further analyses using the Psychometric Toolbox. In brief, the DATA PREPROCESSOR can be summarized as some kind of convenient data manipulation of a dataset that is stored in an Excel sheet labeled “Data”. After the data manipulation, the processed dataset is stored in a sheet labeled

“PreprocessedData”. Some basic item analyses are also provided. The functions can be classified as: (a) functions for processing data; (b) functions for exploring data; and (c) functions for saving data in a file so that it can be further analyzed using the Psychometric Toolbox. We shall now briefly review these three types of function.

Three functions can be used to import a dataset from text files. These functions have an effect mainly on the Excel sheet labeled “Data”, and have already been explained in “3. Entering and importing data” in this manual.

Nine functions can be used to process a dataset. These functions expect to find a dataset already available in the sheet labeled “Data”, and the outcome of the processing is stored in a new sheet labeled “PreprocessedData”. The functions are:



This function helps to reverse the responses of individuals to the selected set of items. The user decides which items to reverse, and defines their minimum and maximum scores. Please note that all the items are expected to have the same scale (i.e., to have the same minimum and maximum possible values). All the items (reversed and not reversed) are copied in the “PreprocessedData” sheet in the same order as in the “Data” sheet: the items that are now reversed are printed in red.



This function aims to split the items into two halves not necessarily of the same length. The user decides which items define the first half and, therefore, which items are sent to the second test half. The two resulting sub-tests are printed in the “PreprocessedData” sheet, and they are separated by a number of wait columns so that they can be easily identified. In addition, an extra column is computed as the overall sum score for each sub-test.



This function allows the user to define scales or sub-tests from the selected set of items. There are no restrictions on the assignment of items to the scales, and each item can also be repeatedly assigned to different scales. The resulting sub-test are printed in the “Sub-Tests” data sheet and are separated by a number of wait columns so that they can be easily

identified. Furthermore, a mouse over produces a pop-up comment displaying the items that compose the sub-test.



Random  
Sample Splitting

This function aims to divide the sample of individuals into two random samples. The first sample is printed in the “PreprocessedData” sheet, and the second sample is printed in the “PreprocessedDataBis” sheet.



Univariate

This function computes the univariate descriptive statistics of an item (or a test or sub-test score). The user must select the item to analyze. Please note that the item can be in any of the sheets available (i.e., the sheets labeled “Data”, “PreprocessedData”, or “PreprocessedDataBis”). The outcomes are printed in a sheet labeled “Univariate”. The descriptive statistics computed are: mean, 90% confidence interval of the mean, standard deviation, skewness, maximum, and minimum. In addition, the distribution of frequencies is printed, and a graphical representation of the distribution is drawn up in the form of a bar chart.



Bivariate

This function computes the bivariate descriptive statistics between two items (or scores). The user must select the items (columns) to analyze. Please note that the items can be in any of the sheets available (i.e., the sheets labeled “Data”, “PreprocessedData”, or “PreprocessedDataBis”). The outcomes are printed in a sheet labeled “Bivariate”. The descriptive statistics computed are: sample size, sum of squares, covariance, and correlation coefficient. In addition, the contingency table is printed, and a graphical representation of the bivariate distribution is drawn up in the form of a bivariate scatterplot.



Replace  
Missing Values

This function helps to fill in the missing values. The missing values must be cells with no value at all (i.e., blank cells). The user has to decide which value to use when a missing value is detected: (a) the mode of the item where the missing value is observed; (b) the mean of the item where the missing value is observed; or (c) an arbitrary value defined by



the user. All the items are copied in the “PreprocessedData” sheet in the same order as in the “Data” sheet: the cells to which a missing value has been added are printed in blue.



Jack  
Knife

This function helps to generate the corresponding N samples typically used in the Jackknife resampling technique. The outcomes are directly saved as text files in the computer. The user is asked for a filename: this filename plus an index (from 1 to N) is used to generate the names.



Parallel  
Forms

This function helps to generate two optimal parallel forms from an item pool which is calibrated using classical test theory. The parallel forms are obtained using two general procedures: a graphical procedure, and an objective procedure. The graphical procedure is Gulliksen’s matched random subtests method. This procedure is based on the criteria proposed by van der Linden and Boekkooi-Timminga, and uses zero-one programming. A detailed description of the procedures can be obtained from: Ferrando, P.J., Lorenzo-Seva, U. & Pallero, R. (2009). Implementación de procedimientos gráficos y analíticos para la construcción de formas paralelas. *Psicothema*, 21(2), 321-325. The outcomes are printed in three sheets: “Descriptives”, “SubTests”, and “Scatter plot”.

---

Two functions can be used to export the outcomes of preprocessing the data. These functions expect to find a preprocessed dataset already available in the sheet labeled “PreprocessedData”. The functions are:



Save data  
as text

The user can just copy the preprocessed data, and then paste it in any of the files in the Psychometric Toolbox (for example, in the [IRT](#) module). However, the “Save data as text” allows the data to be exported as a text file with specific separators for the columns. This option can be useful for further analyzing preprocessed data with other applications that do not accept Excel files (for example, FACTOR for computing exploratory factor analysis).



This option is useful when any of the two Split-Half functions are used: each sample is saved in a separate file. The user is asked to provide a file name for each half, and a character to be used as data separator. The outcome files are in text format.

---

## 4.2 Illustrative example

In this example, we use a database of 250 respondents and 20 items, which measures aggressive behavior (file “Preprocessor.dat”). This database contains missing values and negatively worded items, so some additional steps are required before the data is analysed. First of all, we have to clear all previous results and data by clicking on the respective buttons to ensure that no previous information remains in the macro’s memory. Now it is time to import the file containing the raw data by clicking on “Import Text File”. If the data contains missing values, like in this example, uncheck the option “Treat consecutive delimiters as one” so that the data can be read properly. As mentioned, there are some items which have to be reversed before analysis is begun. The best way to do this is to press the “Reverse items” button and select the items to be reversed. The items can be selected by pressing the Control button in the keyboard and clicking in the corresponding item numbers. In this example, the items to be reversed are the following: item numbers 1, 4, 8, 10, 11, 13, 14, 16, 18 and 20. As part of the reverse process, at this point the program needs to replace the missing values. Decide which value you want to replace the missing values with (mode, mean or arbitrary value). In this example, we have decided to use the mode.

Now, in the “PreprocessedData” tab we obtain the clean version of the questionnaire. No values are missing and all the items are properly defined. The reversed items are printed in red and the cell of the replaced missing values will turn blue (see Figure 3). Also, the average item means of direct and reversed items are shown.

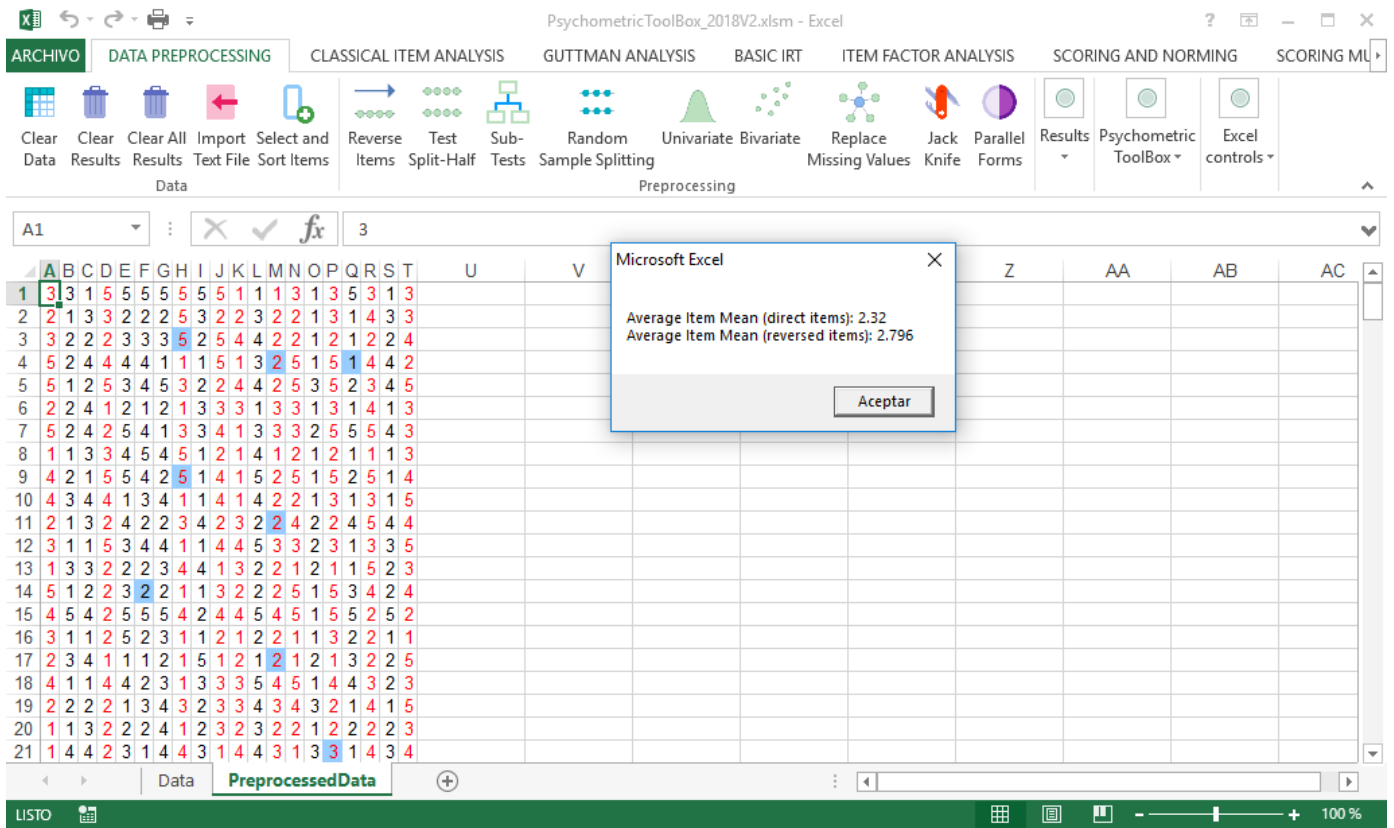
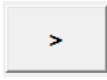
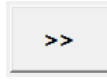
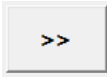


Figure 3. Display when missing values are replaced and some items are reversed

Now we need to export the new database containing the reversed items and import it again by saving this new database in a new file by clicking on “Save data as text” in the subsection Results. For creating the new text file, a column separator is required. We recommend to use a simple space for facilitating the visualization, but other separators could be used (e.g. “;”). Now erase the previous information and import the new database without the missing values and without the reversed items by clicking on the respective buttons (“Clear Data”, “Clear Results” and then “Import Text File” to select the new file).

Before computing the descriptive statistics, click on “Select and Sort Items” and select the items that will be used in the analysis. There is also the option to compute the total sum scores, which we are going to use for obtaining the total score of each participant. By default, all the items are excluded. You can select individual items by clicking on  or all items at once by clicking on . In this example, we need all 20 items, so clicking on  is the best option. A new column will be generated in the “PreprocessedData” tab containing the total score of each participant in bold type.

Now the data is ready to be analyzed, so click on “Univariate” to compute the univariate descriptive statistics and select the column containing the total score of all participants (in bold face). A new tab will be generated containing the output section of the analysis, including the mean and standard deviation of the participant’s scores, as well as the skewness and the minimum and maximum score. A graph will also be generated representing the distribution of all the participant’s scores. In this example, the mean score is 51.16 with a standard deviation of 10.5. Considering the length of the questionnaire (20 items) and the response options (from 1 to 5), the possible total scores range from 20 (answering 1 to all 20 items) to 100 (answering 5 to all 20 items), but the real range in this dataset is not that wide (the minimum score is 26 and the maximum 80). The skewness of the distribution is slightly positive but almost centered, and if we have a look at the graph, the distribution does not depart substantially from the normal.

Let’s assume that we want to create two parallel forms of this questionnaire, with similar means and distribution. The easiest way to perform this operation is to click on “Parallel Forms”, which creates a new window (see Figure 4).

Figure 4. The configuration options for the Parallel Forms function

The number of individuals and variables is automatically filled in with the present data. The Alpha value (by default 0.05) can also be modified. Three new tabs will be generated: the first one is “Descriptives”. This tab presents the difficulties (i.e. means) and discriminations (i.e. corrected item-total correlations) of all the items, as well as the distance between them. This distance is a representation of the similarity of the items, based on their mean and discrimination. The macro uses the distances to establish pairs of similar items and distribute them among the various subtests, starting from the shortest distance (the most similar items). For example, the first pair of items will be item number 1 and 7, because the distance between them

is the shortest of all the possible distances (0.035). This is purely descriptive and will be interpreted no further, so we will now focus on the next tab: “SubTests”, which contains the two resulting versions of the questionnaire and some statistics.

Ideally, the mean of the two versions of the questionnaire should be very similar, as should their standard deviation. In this example the mean of both versions is about 25.5 (25.8 and 25.3) and the standard deviations are 5.9 and 5.5. If there are non-significant differences between the means and/or the standard deviations, an “\*” symbol will appear next to the values, as in this case, for which the means and standard deviations are not significantly different between the two versions. Both the reliability of each version and the reliability of the whole questionnaire using the Spearman-Brown method can be seen. In this example, the reliability of each subtest is about .70 and the reliability of the full test with 20 items is .82 when Spearman-Brown prophecy is used. Finally, the items that make up each subtest are presented, as well as their difficulties and discriminations.

The next tab, “Scatter plot”, presents the bivariate Gulliksen’s plot in which each item is represented as a point in the space defined by the difficulty and discrimination values. This plot can be useful for understanding the process of selecting pairs of similar items and distributing them in each subtest. For example, as we mentioned above, the pair of items with the shortest distance between them are item numbers 1 and 7, followed by items 2 and 9. Figure 5 uses the Scatter plot to show this similarity in a more visual way.

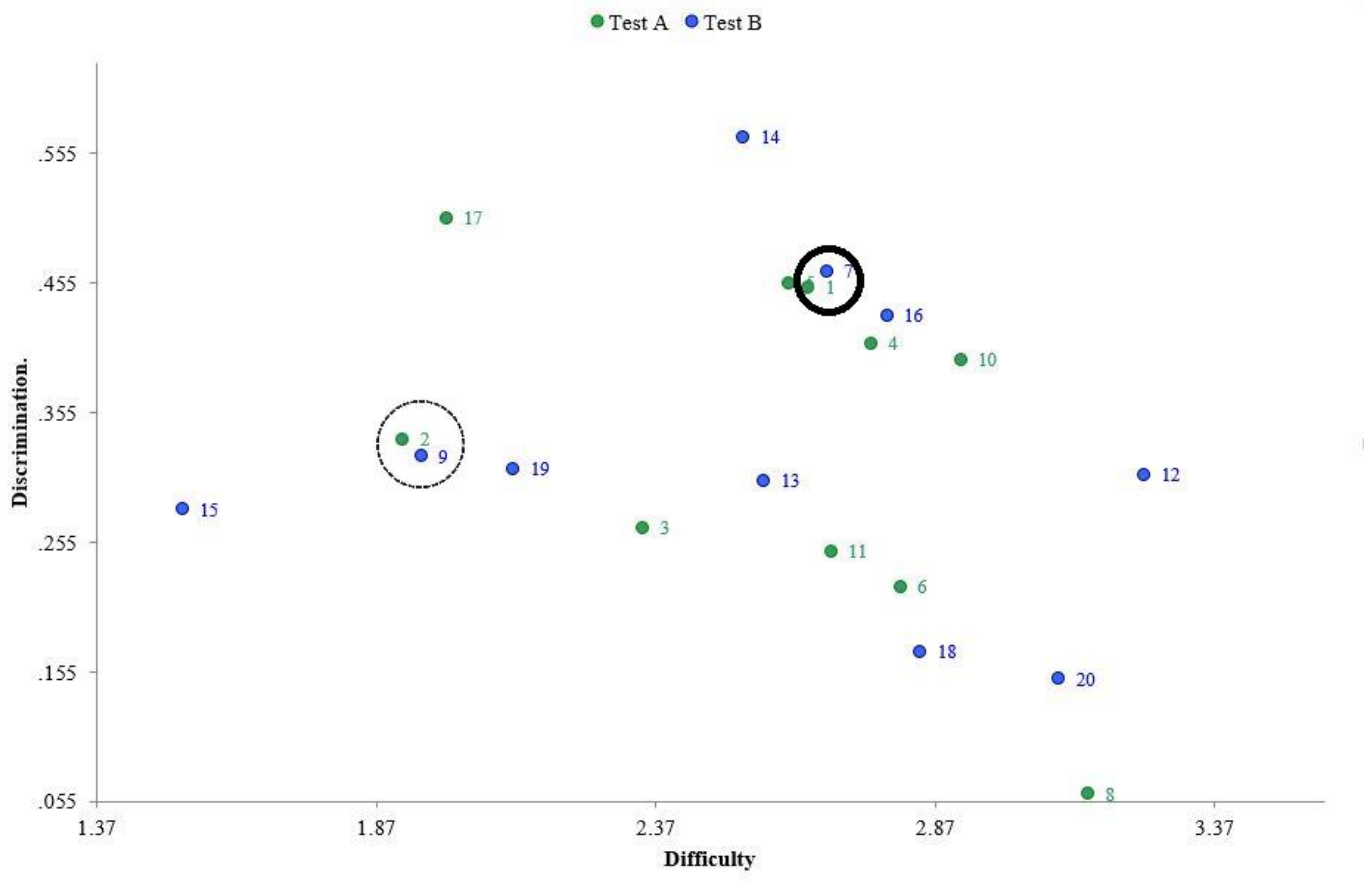


Figure 5. Graphic representation of the distance between the items. Circled with continuous line, the pair of items with the shortest distance (1 and 9). Circled with dotted line, the pair of items with the next shortest distance between them (items 2 and 9)

With these two versions of the questionnaire, bivariate statistics can be computed using item selection obtained by the Parallel Forms method. First, click on “Test Split-Half” and determine which items make up each subtest. In this example, the best solution was to use item numbers 1, 2, 4, 6, 8, 11, 5, 17, 10 and 3 for the first subtest and items number 7, 9, 16, 18, 20, 13, 14, 19, 12 and 15 for the second one. Next, Preprocessed Data presents the data and the total scores for each half separately. The next step is to click on “Bivariate” and select the two columns containing the total score for each subtest (in bold). A new tab will be generated containing the output of the analysis. The covariance and Pearson correlation coefficients are presented, with values of 22.48 for the covariance and .694 for the correlation coefficient. The contingency table is also presented, containing the number of participants with a specific score in the first half of the test and another specific score in the second half. For example, there are no participants with a total score of 15 in the first half and 20 in the second one, but there are 2 participants with a score of 16 in the first one and

11 in the second one. Finally, the bivariate Scatterplot is presented, representing the participant's scores in each half in the diagonal line.

It is noteworthy that if the user is interested in saving any of the obtained results, the best way to do it is copying the cells with the desired section and paste it in another file (another excel sheet, word file, etc.), because the "Save data as text" button only saves the "PreprocessedData" tab info.

Finally, there are other options in this module that are interesting to mention, but there are not connected with the current example.

The Sub-Tests option allows the user to generate multiple sub-tests with no restrictions in the number of items of each sub-tests and also allowing the user to use the same item in more than one sub-test. For clarification, an example of the creation of 3 sub-tests will be displayed.

Let's suppose that we want to create 3 sub-tests, the first one including items number 3, 4, 5, 9 and 10; the second one the items number 12, 13, 16 and 20 and the last one items number 1, 3, 4, 12, 18 and 19. The number of items in each sub-test is different, and there are some items included in two sub-tests. In order to allow some items to be re-used, the user has to select the option: "Display all the items" in the display box of the function. The selection should be the displayed in the Figure 6.

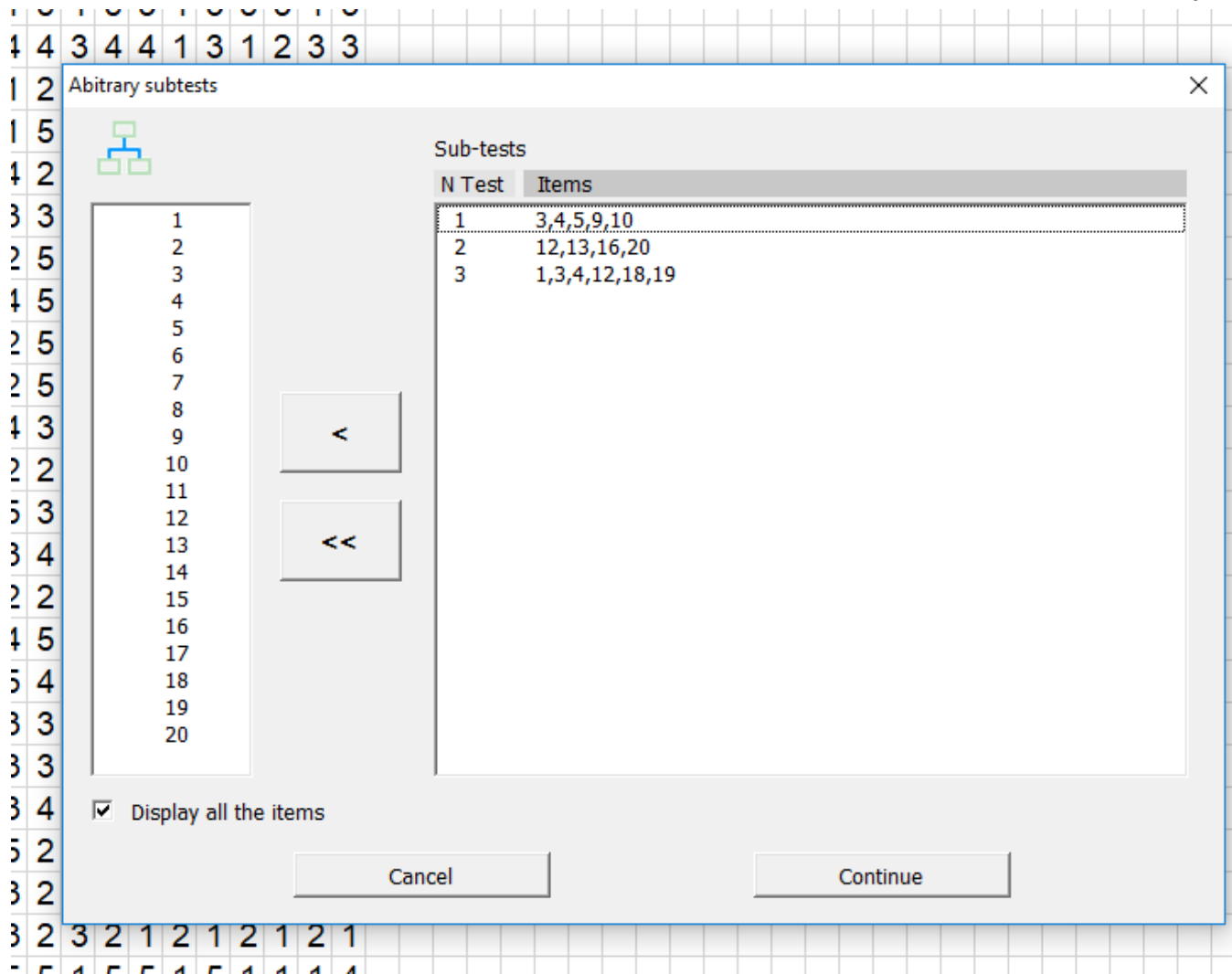


Figure 6. Selection of the items in the Sub-Tests function.

Once selected the sub-tests, a new “Sub-Tests” tab will be created, containing the item scores of each sub-tests. The items of each sub-tests can be displayed doing a mouse over the last column of each sub-test, producing a pop-up comment displaying the items that compose the sub-test, as presented in Figure 7.



PsychometricToolBox\_2018D

ARCHIVO DATA PREPROCESSING CLASSICAL ITEM ANALYSIS GUTTMAN ANALYSIS BASIC IRT ITEM

Clear Data Clear Results Clear All Results Import Text File Select and Sort Items Reverse Items Test Split-Half Sub-Tests Random Sample Splitting Univariate Bivariate Repla Missing V

Data Preprocessing

A1 : X ✓ fx 1

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	1	1	5	5	1		1	5	3	3	Items: 12,13,16,20							
2	3	3	2	3	4		3	4	3	3								
3	2	4	3	2	1		4	4	4	2								
4	4	2	4	1	1		3	4	1	4								
5	2	1	3	2	4		4	4	1	1								
6	4	5	2	3	3		1	3	3	3		4	4	5	1	2	1	
7	4	4	5	3	2		3	3	1	3		1	4	4	3	1	4	
8	3	3	4	1	4		4	5	4	3		5	3	3	4	5	1	

Figure 7. Displaying the sub-tests and which items compose the second one, displayed in the pop-up comment.

The “Random Sample Splitting” button splits the total sample into two samples with the same number of participant’s, and places the corresponding sample data in two different tabs: “PreprocessedData” and “PreprocessedDataBis”. The obtained subsamples can be saved by using the “Save Sub-Samples” button.

“Jack Knife” button performs the Jackknife resampling technique, generating N new samples. The outcomes are directly saved as text files in the computer. The user is asked for a filename: this filename plus an index (from 1 to N) is used to generate the names.

## 5. The “GUTTMAN” subprogram

### 5.1. Summary

The GUTTMAN module fits Guttman’s perfect scale model to a set of binary items. The program can fit the model in two ways: (a) the item ordering is provided by the user, or (b) the program automatically orders the items according to their difficulty indices. In both cases the ordering is assumed to be in the direction of increasing difficulty (i.e. lower  $p$ ).

In the computing section there are two functions for fitting the model in these two ways:



This function orders the items automatically according to their difficulty. The user does not have to provide any information. So it is recommended in calibration stages, or when the order of difficulty of the items is unknown for some reason.



This function requires the user to select the items in the right order of difficulty, and the model is assessed according this ordering. It is the recommended option when the user knows the order of difficulty of the items beforehand, or in cross-validation stages.

---

The output is organized in three parts. The first part displays the inter-item correlation matrix and prints the five highest correlations and the five lowest correlations together with the corresponding item pairs. The second part provides item and total-score descriptive statistics: item means or difficulty indices (i.e.  $p$  and  $q$  values) as well as the mean and standard deviation of the total test scores. The third part provides indices at the global and item level that allow the fit of the Guttman model to be assessed. The global indices are: the coefficient of reproducibility total (CRT), the marginal minimum reproducibility (MMR) the percentage of improvement (PI) and the coefficient of scalability (CS). The indices at the item level are the item coefficients of reproducibility (CR<sub>i</sub>). In all cases the indices are displayed together with their recommended cut-off values.

## 5.2. Foundations and Details

We agree with Edwards (1957) that Guttman analysis is more a procedure for assessing the degree of quality of an existing scale than a method for constructing one. So, for instructional purposes we generally use the GUTTMAN subprogram as an initial step for determining the extent to which an item set has some basic properties (good ordering of the items in terms of difficulty/extremeness, scalability of the response patterns, unidimensionality). We also introduce the Guttman model as a preliminary foundation for the most basic unidimensional cumulative IRT models (e.g. the Rasch model), which can be viewed as stochastic versions of it. The main advantage of this introduction is that the Guttman model is straightforward and easy to understand.

The GUTTMAN program in the PT is intended for binary items scored as 0 and 1, and implements the technique of analysis known as “deviation from perfect reproducibility” proposed by Goodenough and Edwards (e.g. Edwards, 1957). So, the number of Guttman errors is obtained by comparing each observed response pattern to the corresponding model-predicted pattern, and each deviation of an observed response from the predicted response is counted as an error. In the program, the predicted patterns can be obtained in two ways: (a) automatically, by ordering the items in terms of difficulty according to the observed proportions of endorsement, or (b), the ordering of difficulty is defined ‘a priori’ by the user.

The output is organized in three parts. The first and second part provide statistics used in conventional item analysis (see also the CIA subprogram) that describe the extremeness and internal consistency of the items. Thus, the first part of the output provides the item means or difficulty indices (i.e.  $p$  and  $q$  values) as well as the mean and standard deviation of the total test scores. The second part aims to assess the degree of inter-item consistency, and displays the inter-item correlation matrix as well as the five highest correlation values and the five lowest correlations together with the corresponding item pairs.

We turn now to the third part of the output: Goodness-of-fit assessment. In Guttman scaling, model-data fit mainly aims to assess the degree to which the observed responses deviate from the ideal deterministic type of responding. Cut-off values are then determined in order to establish whether the approximation is close enough to treat the observed data as if they were a perfect scale. There are several criteria for assessing this degree of deviation. Our program implements the following:

### **-Coefficient of reproducibility total (CRT)**

$$CRT = 1 - \frac{\text{Total } n^{\circ} \text{ errors}}{\text{Total } n^{\circ} \text{ responses}}$$

Conceptually the CRT assesses the extent to which the response patterns in the data set can be accurately reproduced from the total scores assigned to them.

In Guttman's original specifications, a set of items was considered scalable if the errors of reproducibility were 10% or less of the total responses (see e.g. McIver & Carmines, 1981). Following this specification, the amount of error would be considered tolerable and the data scalable if  $CRT \geq 0.90$ . This cut-off value is provided in our program together with the CRT value.

### **-Coefficient of reproducibility per item (CRi)**

$$CRi(j) = 1 - \frac{n^{\circ} \text{ errors in item } j}{n^{\circ} \text{ respondents}}$$

The CRi makes the same type of assessment as the CRT but at the level of each individual item not overall. For any item, the CRi value gives the proportion of responses to that item that can be correctly reproduced. The cut-off reference value is the same as the CRT: 0.90 (see Torgerson, 1958).

The key relation between both types of index is that the CRT is the arithmetic mean of all the CRi's. This property is particularly relevant in the process of item selection and test refinement: if the items with the lowest CRi's are discarded, then the CRT for the sub-set of retained items will necessarily increase.

The CRT has important shortcomings that have been repeatedly discussed in the literature (e.g. Edwards, 1957; Torgerson, 1958; McIver & Carmines, 1981)). One of its main limitations is that it depends on the marginal proportions of item endorsement, so that it might be grossly inflated if the data contains very extreme items (i.e. very "easy" or very "difficult"). To assess the extent to which the item modal values reproduce the observed response pattern, the following indexes are obtained:

### **-Minimum Marginal Reproducibility (MMR) and Percent Improvement (PI)**

Define SM as the sum of minimums (i.e. the least frequent scores) of the item marginals, then:

$$MMR = 1 - \frac{SM}{\text{Total } n^{\circ} \text{ responses}}$$

Conceptually, MMR is the CRT that would be obtained if the observed response patterns were corrected not on the basis of each total test score, but solely on the basis of the item modal values. The effective increase in reproducibility when the scores are used to correct the patterns is given by the PI:

$$PI = CRT - MMR.$$

### **-Coefficient of scalability (CS)**

The CS is obtained as:

$$CS = 1 - \frac{CRT - MMR}{1 - MMR} = \frac{PI}{PI(\max)}.$$

So, CS is the ratio between the effective increase in reproducibility defined above and the maximum possible increase given the marginal proportions of item endorsement. The standard recommended cut-off value for CS is 0.60 (see e.g. McIver & Carmines, 1981) and is printed in the GUTTMAN output.

To sum up, if the CRT is above 0.90, all the CRi's are above 0.90, and the CS is above 0.60, our item set can be regarded as behaving essentially as a perfect scale. If this is not the case, then a process of item selection must be undertaken (provided that enough items are available). In our practical sessions, in which an item selection process is always needed, we emphasize the need to use cross-validation to avoid over-fitting due to capitalization of change. The cross-validation process can be easily conducted by using the DP subprogram described in the chapter above.

### 5.3. Illustrative example

In this example, we use a database of 420 respondents and 10 items as a calibration sample (file "Guttman\_calibration.dat"). If any modifications are needed, we use a second sample with the same number of respondents and items (validation sample with the file name "Guttman\_validation.dat"). First of all, we clear all previous results and data by clicking on the respective buttons, to ensure that no previous information remains in the macro's memory. Now it is time to import the file containing the raw data by clicking on "Import Text File". Let's assume that we want to perform an exploratory analysis to make sure that the scale fits the Guttman model. Ideally, the 10 items should be sorted in ascending order of difficulty, so the expected pattern for a participant that answer correctly 5 items should look like the following:

1	1	1	1	1	0	0	0	0	0
---	---	---	---	---	---	---	---	---	---

When the participant could answer correctly the first five, but once he arrives to the sixth, he is unable to answer correctly any of the remaining items. When we expect that the items are sorted in ascending order of difficulty, we can use the “Automatic Selection” option.

The first output part that we will look at is the “CorrelationCoefficients” tab. The full correlation matrix is displayed, as well as the five highest correlations and the five lowest ones. For example, the highest correlation is between items 7 and 4 ( $r=0.418$ ), and the lowest is between items 10 and 1 ( $r=0.0024$ ).

The next tab (“Descriptive”) shows if the pattern of difficulty is the expected one. Figure 8 shows that the difficulty increases from item to item, going from  $p=0.9976$  in item 1 (very easy) to  $P=0.0024$  in item 10 (very difficult). The only exception is item 4, which is more difficult than expected. It can also be seen that items 1 and 10 have very extreme locations, which make them virtually useless for discriminating between subjects.

2 <b>Guttman</b>							
3							
4	Item	1	2	3	5	6	7
5	P	0.9976	0.9690	0.8071	0.5048	0.4929	0.4833
6	Q	0.0024	0.0310	0.1929	0.4952	0.5071	0.5167
7							
8	Item	4	8	9	10		
9	P	0.4643	0.1476	0.0167	0.0024		
10	Q	0.5357	0.8524	0.9833	0.9976		
11							
12							
13	Total Score						
14	Mean	4.8857					
15	Standard Deviation	1.9336					
16							

Figure 8. Displaying the pattern of difficulty. The item marked with a red square the only item that does not have the expected difficulty. The items circled in orange are the items with extreme difficulties.

For example, based on the obtained difficulty pattern, a participant’ scores that answer correctly 5 items should look like the following:

1	1	1	0	1	1	0	0	0	0
---	---	---	---	---	---	---	---	---	---

Where we can appreciate that the participant was unable to answer correctly the item number 4 despite that he actually could answer right items number 5 and 6, because item 4 has resulted to be more difficult than expected.

Let us now move on to the tab “Model Fit”, which shows whether the reproducibility values of the model are acceptable or not. Neither CRT (0.898) nor CS (0.562) have the recommended model indices (0.90 and 0.60, respectively). Also some items have better reproducibility coefficients than others. Specifically, items 4, 5, 6 and 7 have a CRI lower than 0.90.

In order to improve the model, we perform another analysis with the items that have (a) the best reproducibility coefficients and (b) locations that are not too extreme. We want to keep at least five items, so we discard the following items:

- Items 1 and 10 (their locations are too extreme and do not provide relevant information about the respondents).
- Items 4, 5 and 6 (their CRI coefficients are mediocre). Item 7 is preserved because we want to keep at least five items, and it had a better CRI than the other three items.

After these items have been discarded, the new selection in order of increasing difficulty is: Item 2, 3, 7, 8 and 9.

The next step in this example is to check if the five selected items in the order described above fit in with Guttman’s model. For this purpose, we use the second option of the “GUTTMAN” module: “User Selection”. We select the items in order of increasing difficulty (i.e. 2, 3, 7, 8, and 9). We confirm that this order is appropriate by checking the “descriptives” tab, which shows that all the coefficients have higher goodness of fit values than the recommended ones (CRT=0.977, CS=0.869), and all the items have good CRIs, even item 7. Also, Minimum Marginal Reproducibility (MMR) is 0.826, so the Percent Improvement (PI) = 0.151 (CRT-MMR).

The last step is to perform the cross-validation analysis. We use a second sample containing the same number of participants and the same 10 items (validation sample). In this case, we perform a confirmatory analysis, skipping the automatic selection and selecting the same five ordered items as in the previous analysis.

The results show that all the coefficients are even higher than in the calibration sample (CRT=0.985; MMR=0.838; Pi=0.147; Cs=0.906), and the order of difficulty remains the same. We thus conclude that the selected set of five items behaves like a Guttman scale.

#### References

- Edwards, A. L. (1957). *Techniques of Attitude Scale Construction*. New York: Appleton-Century-Crofts.
- McIver, J., & Carmines, E. G. (1981). *Unidimensional scaling* (No. 24). Newbury Park: Sage.
- Torgerson, W. S. (1958). *Theory and methods of scaling*. New York: Wiley.



## 6. The Classical Item Analysis (CIA) sub-program

### 6.1. Summary

The CIA module **(a)** performs classical test theory (CTT) item and scale analysis, and **(b)** estimates the reliability of the raw scores on the basis of Cronbach's alpha coefficient.

The CIA module contains two computing buttons:



Compute

This function performs a single-scale item analysis from the items contained in the "Pre-processed data" sheet. The only requirement is to determine which items will be used to perform the analysis and their order. The output will be generated in two tabs: "Descriptive" and "Alpha".



Compute  
Sub-Tests

In this case the analyses are performed on a scale-by-scale basis on the sub-tests or scales previously defined with the data preprocessor and contained in the "Sub-Tests" data sheet. As before, the output will be generated in two tabs labelled: "Descriptive" and "Alpha".

---

For both, single scale or multiple scales, the output is organized in two parts. The first part (Descriptive tab) provides the descriptive analysis of the item and test scores. At the item level, the reported statistics are: **(a)** the inter-item correlation matrix, **(b)** the item means and standard deviations, **(c)** the relative difficulty index and **(d)** the descriptive statistics for the test scores.

The second part of the output (Alpha tab) provides the item discrimination indices and the estimated reliability of the raw scores. The item indices are: **(a)** the corrected item-total correlation, **(b)** the squared multiple correlation, and **(c)** the alpha estimate if the item is deleted from the scale. The reliability estimates are: **(a)** the coefficient alpha based on the raw scores, **(b)** the corresponding 90% confidence interval, and **(c)** the standardized alpha.

### 6.2. Foundations and Details

The Classical Item Analysis (CIA) sub-program is intended for item selection and scale refinement.

Its main aim is to make a detailed assessment of (a) how each particular item functions as an element of the test, and (b) how the omission of each item would affect the properties of the total test score. As proposed by Nunally and Bernstein (1994) we intend to use CIA as a preliminary, quick, and efficient approach for item selection that can be used with small samples or pilot groups, and which allows the worst items to be discarded at this stage.

Conventional item analysis is based on two types of item characteristics (e.g. Mellenbergh, 2011): extremeness (difficulty in the case of ability items) and discriminating power. The most common statistic for measuring item extremeness is the arithmetic mean of the item scores (in the case of a binary item scored as 0-1, the mean is also the proportion of endorsement, usually known as the difficulty index). In the first part of the output, the CIA subprogram reports the item means and standard deviations. Because an item that has small standard deviation (or variance) does not contribute much to the variance of the total test scores, this item yields little information and can be considered superfluous.

For the graded-response case, the arithmetic mean of the item scores depends on the number of response categories. So, it seems useful to provide a ‘relative’ extremeness index in the 0-1 range for this type of scores. The chosen index, labelled as “relative item difficulty” is simply the item mean scaled in this range. For example, for an item  $j$  with possible scores 1,2,3,4, and 5, the relative difficulty is:

$$RD_j = \frac{\bar{X}_j - 1}{5 - 1}$$

Apart from the item descriptives, the first part of the CIA output also provides the mean and standard deviation of the total test scores. These two latter statistics aim to assess how the omission or inclusion of items will affect the shape and characteristics of the distribution of the scores at the test level.

We turn now to the assessment of the items’ discriminating power. Our theoretical basis is to consider the items as measures of a unidimensional continuum that can be modeled as a common factor (e.g. Henrysson, 1962). Within this framework, the most natural discrimination parameter is, possibly, the correlation between the scores on an item and the continuum it intends to measure. Conceptually, this

parameter provides two types of information: (a) the degree to which the item scores allow the respondents to be differentiated in terms of the attribute they measure (i.e. discriminating power), and (b) the degree of relation between the item scores and the measured attribute (i.e. item quality or item consistency). The ‘true’ levels of the attribute to be measured are indeed unknown, and the most common approach is to use as a proxy the scale scores computed from the other items in the set. This index is known as the corrected item-total correlation or item-rest correlation (Mellenbergh, 2011) and is the main discrimination index implemented in CIA.

In the approach we take here, the corrected item-total correlations can be viewed as proxies for the factor loadings that would be obtained by fitting Spearman’s factor analytic (FA) model to the item set (e.g. Henrysson, 1962). Indeed, item FA can be carried out directly by using the IFA subprogram described in the next chapter. However, we agree again with the recommendations given by Nunally and Bernstein (1994) that the type of analysis implemented in CIA is useful as a precursor for item FA. Finally, the program does not provide cut-off values for deciding whether the estimated item discrimination is acceptable or not (mainly because we believe that ‘acceptable’ values mostly depend on the type of test). However, generally speaking, reasonable minimally acceptable values are about 0.2 to 0.3.

Apart from the corrected item-total correlation, the CIA output provides the following auxiliary measures of discrimination: (a) the inter-item correlation matrix, (b) the squared multiple correlation between each item and the remaining items in the set, and (c), the estimated value of coefficient Alpha when each item is deleted from the set. The latter indicator is rather useful in the item selection process for deciding whether it is worth maintaining the item in the final version of the test or not.

The final part of the Alpha-tag output provides the reliability estimate for the test scores based on the selected items. The most natural estimate here is Cronbach’s alpha, which directly depends on the number of items and the average correlation between them. In effect, the standardized alpha estimate can be computed as:

$$\alpha_z = \frac{nr_{jk}}{1 + (n - 1)r_{jk}}$$

where  $\bar{r}_{jk}$  is the average inter-item correlation and  $n$  is the number of items. Next, consider the following approximate relation obtained from the domain-sampling theory (e.g. Nunnally & Bernstein, 1994)

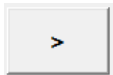
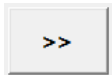
$$r_{jX} \cong \sqrt{\bar{r}_{jk}} .$$

In words: for a test of reasonable length, the correlation of item  $j$  with the total test score  $X$  (i.e. its discrimination index) approximately equals the average correlation between this item and the remaining items in the test. It then follows that the estimated value of the (standardized) coefficient alpha is essentially a direct function of the number of items and their discrimination values.

In recent years, coefficient alpha has come in for some criticism, and the trend at present is no longer to consider it as the ‘default’ reliability estimate. As is known, alpha is only an unbiased estimate of the reliability of the total scores if (a) the items are unidimensional in the factorial sense, (b) they are at least true-score equivalent, and (c) their measurement errors are uncorrelated (e.g. Raykov & Marcoulides, 2011). These conditions might be hard to fulfill in most practical applications and are seldom checked. For these reasons, a model-based reliability estimate that does not require tau-equivalence such as coefficient omega is theoretically superior (as discussed in the IFA subprogram). Even with its shortcomings, however, alpha is still widely used and we believe it must be implemented. Furthermore, for didactic purposes, it is very instructive to assess and check how the process of item selection and the fine trade-off between the number of retained items and their discriminating powers determines the reliability that the scores will have in the final version of the test.

The statistics reported in the reliability part of the output are: the alpha estimate based on the raw scores; the standardized alpha as discussed above; and the 90% confidence interval corresponding to the raw alpha estimate. Editorial guidelines and measurement standards require reporting confidence intervals around the point reliability estimates, and we agree that this is positive. Like any other statistic, a reliability estimate is impacted by sampling error variance, and a poor estimate obtained, for example, in a small sample can lead to misleading results when the properties of the test are interpreted.

### 6.3. Illustrative example

For this example, we use a database of 338 respondents and 35 items, which is the same as the one we use in the IFA and Scoring and norming module (file “data\_CIA\_IFA\_NORMS.dat”). Before getting to the computing part, we have to click on “Clear Results” and “Clear Data”, to ensure that any remaining information is erased. Now it is time to import the file containing the raw data by clicking on “Import Text File” and selecting the appropriate column separator. In this case, the module only has one computing option, so we click on “Compute” and choose the items to be included in the analysis. By default, all the items are excluded. You can select individual items by clicking on  or all items at once by clicking on .

. In this example we use all the items.

The first results that we look at are the ones on the descriptive analysis of the item and test scores. They consist of: the inter-item correlation matrix, the mean, standard deviation and relative difficulty of each item and finally, the descriptive statistics of the test scores. In our example, we can see that item 20 has the lowest mean score (2.80, 0.45 of relative difficulty) and item 19 has the highest (4.25, 0.81 of relative difficulty). With regards to the total test scores, the average is 122.43, the minimum score being 74 and the maximum 165.

The next part of the output (alpha tab) provides the item discrimination indices and the estimated reliability of the raw scores. In this example, Cronbach’s alpha was 0.889 (90% confidence interval is 0.872; 0.904). Regarding item indices, the following are reported: the scale mean and standard deviation if this item is deleted, the corrected item-total correlation, the square multiple correlation, and Cronbach’s Alpha obtained if item is deleted. As can be seen in Figure 9, the least discriminating item is item number 7, which has a corrected item-total correlation of 0.186 and a squared multiple correlation of 0.145.

12	Item-Total Statistics					
13						
14	Item	Scale Mean if Item Deleted	Scale Variance if Item Deleted	Corrected Item-Total Correlation	Square Multiple Correlation	Cronbach's Alpha if Item Deleted
15	1	119,2249	211,516	0,460	0,396	0,885
16	2	118,7071	217,134	0,396	0,347	0,887
17	3	118,4793	216,292	0,331	0,281	0,888
18	4	119,0000	206,160	0,664	0,597	0,881
19	5	119,2426	208,487	0,625	0,649	0,882
20	6	119,0562	213,347	0,385	0,451	0,887
21	7	118,6095	218,221	<u>0,186</u>	0,145	0,891
22	8	118,7515	213,897	0,477	0,411	0,885
23	9	118,9112	212,425	0,385	0,292	0,887
24	10	118,4645	215,306	0,405	0,489	0,886
25	11	118,5740	211,432	0,568	0,509	0,884

Figure 9. A view of some item-total statistics. The lowest corrected item-total correlation, corresponding to item number 7, is underlined.

As expected, if this item is deleted, we obtain the highest possible increase in the estimated reliability with a Cronbach's alpha of 0.891. There are other "bad" items (for example, item 15 or item 31) and if we decide to delete them, the Cronbach's alpha estimate will also increase. At the other extreme, the best items are 5, 22 and 29, and if they were to be deleted the decrease would be the largest possible, reducing alpha to 0.882-0.883. Obviously, these increases and decreases are very small because the number of items is relatively high, so deleting only one item at a time has little impact on the reliability estimate of the total test scores.

## References

- Nunnally J. C., & Bernstein I. H. (1994). *Psychometric Theory*. New York: McGraw-Hill.
- Mellenbergh, G. J. (2011). *A conceptual introduction to psychometrics: development, analysis and application of psychological and educational tests*. The Hage: Eleven International Publishing.
- Henrysson, S. (1962). The relation between factor loadings and biserial correlations in item analysis. *Psychometrika*, 27, 419-424.
- Raykov, T., & Marcoulides, G. A. (2011). *Introduction to psychometric theory*. London: Routledge.

## 7. The Item Factor Analysis (IFA) sub-program

### 7.1. Summary

The IFA module (a) assesses the dimensionality of a set of items according to the linear factor analysis (FA) model, (b) performs item and scale analysis on the basis of the unidimensional FA sub-model, and (c) estimates the reliability of the test scores with the omega coefficient. It should be emphasized that the purpose of the analysis is to obtain unidimensional scales. So, in truly multidimensional instruments, the analysis must be carried out on a scale-by-scale (or dimension-by dimension) basis.

The module contains two computing functions:



Item  
Appropriateness

This function assesses the appropriateness of the data and the unidimensionality assumption. For the set of items under study, IFA first obtains the descriptive item statistics and the inter-item correlation matrix, and assesses sampling adequacy with the KMO index. Next, IFA fits the one-factor and the two-factor models by using MINRES estimation and provides two goodness-of-fit indices for each solution: the gamma index (also known as GFI) and the root mean squared residual (see McDonald 1999). The two-factor solution is given in canonical or principal-axes form. The only information it needs are the items that will be used for the analysis.



Item Factor  
Analysis

This function performs unidimensional FA-based item analysis for the set of items judged to be essentially unidimensional and estimates the reliability of the test scores. The results displayed in the item-analysis output are the item means (difficulty indices), the standardized factor loadings (item discriminations) and the squared factor loadings (item reliabilities). Finally, IFA provides the omega reliability estimate for the test scores together with its 90% confidence interval. The only information it needs are the items that will be used for the analysis.

---

## 7.2. Foundations and Details

The CTT model used in item analysis and implemented in the previous (CIA) sub-program can be considered to be a particular case of the unidimensional factor analysis (FA) model or Spearman's model (e.g. Jöreskog, 1971). The main difference between the two models is that in the CTT model the error term is pure random error, whereas in the FA model the error term contains both random error and item specificity (e.g. Lord & Novick, 1968). So, the CIA procedure can be regarded as an approximate approach for fitting a unidimensional FA model in which there is no specific variance in any of the items.

The use of the linear FA model for item analysis and reliability estimation purposes has potentially important advantages which are discussed below. However, the procedure is also more complex than CTT analysis and can become unstable in some conditions (mainly small samples and a large number of items). For this reason, we recommend that a preliminary CTT-based item analysis is first carried out using CIA so that the worst items can be discarded. Analysis based on the FA model can then be carried out in a second stage for all the items that passed the CTT-based selection process. Note that, generally, the first stage analysis is based on a small pilot sample, while FA analysis requires a relatively large sample (say 300 or more) if results are to be stable.

IFA has two main advantages over CIA. First, it allows us to make a rigorous assessment of the unidimensionality assumption (which in CIA is taken for granted). Second, it provides a better assessment and interpretation of the item discriminations, and a better reliability estimate for the test scores (coefficient omega). With respect to the first point, a high coefficient alpha (or a strong inter-item consistency) by no means guarantees that the item set is unidimensional.

Before we discuss how the IFA model is implemented in the sub-program, a caveat is in order. The general FA model we implement is the linear model intended for continuous variables. In practice, this model generally works well for graded-response or more continuous items, especially if the marginal distributions are not too extreme (see Ferrando & Lorenzo-Seva, 2013 for a detailed discussion). For binary scores or extreme item distributions, non-linear FA models based on the underlying-variables approach are generally



more appropriate (but also more complex and less stable). We discuss these models in more advanced courses and fit them with our program FACTOR (Lorenzo-Seva & Ferrando, 2013).

The first part of the IFA module assesses the appropriateness of the data and the unidimensionality assumption. For the set of items under study, IFA (a) obtains the descriptive item statistics and the inter-item correlation matrix (in order to judge the internal consistency of the items), and assesses sampling adequacy with the KMO index. Next, IFA fits the one-factor and the two-factor models by using MINRES estimation and provides two goodness-of-fit indices for each solution: the gamma index (also known as GFI) and the root mean squared residual (see McDonald 1999). The two-factor solution is given in canonical or principal-axes form.

The rationale for the above is as follows. If the data is essentially unidimensional, then the loadings on the first factor should all be high and positive, while the loadings on the second factor should be negligible. Furthermore, the one-factor model should fit the data well, and the improvement in fit when going from 1 to 2 factors should be negligible. If this is not the case, inspection of the salient loadings on the second factor indicate which items must be discarded if unidimensionality is to be achieved.

The second part of the IFA implementation assumes that the item set obtained is (essentially) unidimensional. So, only the single-factor solution is provided for the set of selected items. The standardized loadings on this factor are labeled as “item discriminations”, which is justifiable. In effect, the magnitude of the loading indicates the sensitivity of the item as an indicator of the common dimension, and also its ability to discriminate between individuals with low and high values in the dimension (e.g. McDonald, 1999). When the model assumptions hold, the values of these loadings must be very similar to the corrected item-total correlations obtained with the CIA sub-program (e.g. Henrysson, 1962). In addition to the loadings, IFA also prints its squared values, which are labeled “item reliabilities”. Again, this labeling is justified: in the unidimensional model, the squared standardized loadings are the squared item-factor correlations, and so they measure the proportion of variance that the item scores share with the common factor or dimension. They are, then, appropriate reliability coefficients (e.g. Lord & Novick, 1968, section 9.11).

The final part of the IFA output provides the omega reliability estimate for the test scores based on the selected items (see McDonald, 1999) together with its 90% confidence interval. The omega coefficient is a

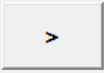
better reliability estimate than the alpha estimate provided in the CIA program for three main reasons.

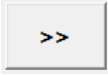
First, it is model based, so the basic unidimensionality assumption for the selected items is previously assessed via goodness-of-fit assessment. Second, it takes into account that the items do not generally measure the construct with the same degree of quality and precision. Finally, omega can be properly interpreted as the proportion of test variance due to the common trait or factor the test measure. So, it indicates how well the test scores measure the intended construct. If we denote by  $\lambda_j$  the standardized loading for item  $j$ , and by  $\sigma_j^2$  the residual variance, omega is given by:

$$\omega = \frac{\left( \sum_j^n \lambda_j \right)^2}{\left( \sum_j^n \lambda_j \right)^2 + \sum_j^n \sigma_j^2}$$

In the description of the CIA program we have already discussed the importance of setting a confidence interval around the reliability estimate. The 90% confidence interval for omega provided by IFA is obtained by using the delta linearization approach proposed by Raykov (Raykov & Marcoulides, 2011).

### 7.3. Illustrative example

For this example, we will use the same database of 338 respondents and 35 items that was used in the CIA and Scoring and norming module illustration (file “data\_CIA\_IFA\_NORMS.dat”). Before starting the analysis, we have to click on “Clear Results” and “Clear Data”, to ensure that any previous remaining information is erased. Now it is time to import the file containing the raw data by clicking on “Import Text File” and selecting the appropriate column separator. This module has two computing options. We start by clicking on “Item appropriateness” in order to assess the dimensionality of the items. By default, all the items are excluded. You can select individual items by clicking on  or all items at once by clicking on



. This operation can take a minute or two, depending on the number of respondents and items.

Once it has finished you can see the results on the “Item appropriateness” tab.

First, the descriptive item statistics are shown for the 35 items and the inter-item correlation matrix. Next, the KMO index is presented, in this case a commendable 0.87. If you scroll down, you will find the eigenvalues tab, which gives the percentage of variance explained by the number of factors and the communalities from the 35 items. Finally, it provides two goodness-of-fit indices for each solution (1-factor or 2-factor). The first is the gamma-Goodness-of-fit index (also known as GFI), with a value of 0.864 on the unidimensional solution and 0.925 on the 2-factor solution. The second index is the root mean squared residual, resulting in a value of 0.087 on the unidimensional solution and 0.064 on the 2-factor solution. In this example, we want to find an acceptable unidimensional solution, so we are going to check the communalities of the items and decide which items it is preferable to eliminate in order to improve model-data fit

Item number 7 has the lowest communality ( $h^2=.145$ ), followed by item 23 ( $h^2=.248$ ), item 32 ( $h^2=.251$ ), item 25 ( $h^2=.262$ ) and item 3 ( $h^2=.281$ ). If these five items are excluded, the GFI improves (0.881) and RMSR hardly increases (0.089), so we compute the unidimensional FA solution for the set of selected items by returning to “Compute” tab and selecting “Item Factor Analysis”.

Once again, this can take a little while. When the computation is over, the results display the item means (difficulty indices), the standardized factor loadings (item discriminations) and the squared factor loadings (item reliabilities). As far as the difficulty indices are concerned, item 20 has the lowest mean score (2.80) and item 19 has the highest (4.25). In this example, the item with the highest factor loading is number 29 (0.737), and the item with the lowest is number 31 (0.209). Item 5 has the highest reliability (0.638), and item 34 the lowest (0.269). Finally, IFA provides the omega reliability estimate for the test scores, and in the sample we used this was 0.921. The 90% confidence interval in absolute values and real values is 0.917-0.925.

If we compare the omega reliability estimate with that obtained by computing Cronbach’s Alpha (on CIA module), we can see that the Omega estimate is (0.921) is larger than alpha (0.891). This is the most usual result because Cronbach’s alpha tends to underestimate reliability.

## References

- Ferrando, P.J. & Lorenzo-Seva, U. (2013). *Unrestricted item factor analysis and some relations with item response theory*. Technical Report. Department of Psychology, Universitat Rovira i Virgili, Tarragona.
- Henrysson, S. (1962). The relation between factor loadings and biserial correlations in item analysis. *Psychometrika*, 27, 419-424.
- Jöreskog, K.G. (1971). Statistical analysis of sets of congeneric tests. *Psychometrika*, 36, 109-133.
- Lord, F.M., & Novick, M.R. (1968). *Statistical theories of mental test scores*. Reading (MA): Addison-Wesley.
- Lorenzo-Seva, U., & Ferrando, P. J. (2013). FACTOR 9.2: A Comprehensive Program for Fitting Exploratory and Semiconfirmatory Factor Analysis and IRT Models. *Applied Psychological Measurement*, 37, 497-498.
- McDonald, R. P. (1999). *Test theory: A unified approach*. Mahwah, NJ: Lawrence Earlbaum.
- Raykov, T., & Marcoulides, G. A. (2011). *Introduction to psychometric theory*. London: Routledge.

## 8. The Scoring and Norming sub-program

### 8.1. Summary

The Scoring and norming module (a) provides the descriptive statistics of the raw test scores, (b) transforms the individual raw scores, and (c) creates the test normative table.

There are two computing options, explicitly labelled as step 1 and 2 (once the data file is imported), because the order of the process is important in this module:



Step 1: Individual Transformation

In this function the user has to select which items will be used for the analysis, enter the reliability estimate (regardless of how it has been obtained) and choose the confidence level (68%, 95% or 99%) and the linear transformed scale to be displayed. The reliability can be obtained from different sources and by different methods (for example, Cronbach's alpha can be obtained from the CIA sub-program, or the Split-Half estimate can be obtained from the Preprocessor sub-program).



Step 2: Normative Table

This function generates the normative table of the instrument. It uses the data from the "Individual Transformation" tab, which is obtained using the function with the same name. The user can specify which scale will be used to generate the normative table (Standard Z, scale T, scale IQ or Stanine) and also whether the data needs to be grouped or not in the table. An extended table with error bands for all the transformed scores is also available as an option

There is an option for saving all the individual transformed scores, on the results section:



Save Individual Transformation

This function allows the user to save all the individual transformations in a text file, including all the scales and their confidence intervals. This feature requires the "Individual Transformations" function to have been executed previously.

The output is organized in two parts. The first part, which is obtained by clicking on “individual transformation”, linearly transforms the raw scores of each individual. The raw scores are obtained from the item set which is specified by the user (the default is the whole test), and the possible transformed scores are: standard or typical (0,1), T (50,10), IQ (100, 15) and stanine (5,2) (linear transformations) and Percentile ranks (nonlinear transformations). In addition to the transformed score, the module provides the corresponding confidence interval and the percentile band at the confidence level specified by the user. The scores, both raw and transformed, can be saved in a text file by clicking on “save individual transformation”.

The second part of the output is obtained by clicking on “normative table”. Three pieces of information are provided: **(a)** the descriptive statistics of the raw scores together with the corresponding histogram, **(b)** the frequency table, and **(c)** the normative table which uses both linear (standard, T, IQ or Stanine) and nonlinear (percentile) transformations. Tables **(b)** and **(c)** can be non-grouped or grouped at the user’s request. The non-grouped table uses all the observed scores, and is recommended for short tests which have a limited range of values. Grouped frequency tables are preferable when the range of raw score values is large.

The extended table with error bands can be obtained by marking this option in the “normative table” menu. For each possible test score (or class mark in the grouped case), the extended table provides not only the point transformed estimate (linear and percentile), but also the corresponding confidence band. We strongly emphasize the use of the extended table in real applications, as the error bands indicate the confidence that can be placed on the transformed score, and so, the extent to which valid inferences from this score are warranted.

## 8.2. Foundations and Details

The Scoring and norming module implements standard procedures for scaling and norming unidimensional test scores based on classical test theory. It has two main aims: (a) to scale the individual raw scores, and (b) to build the normative table of the test according to the user’s specifications.

With regard to score transformations (scaling) we follow the distinctions made by Petersen, Kolen, and Hoover (1993) and consider the primary scores obtained by linear transformations of the raw scores, and the auxiliary scores, which in the Scoring and norming module are the percentile ranks. In both cases the transformation aims to make the resulting scores simpler and more informative, and the ultimate goal is to make interpreting each individual score more straightforward. The additional information that is gained with the transformation is of two types: normative meaning and score precision.

The linear transformations computed in Scoring and norming are the simpler, unadjusted transformations obtained after the raw scores have been standardized in the sample group provided by the user. Apart from the standard scores, which are the initial transformation, this module can provide (at the user's choice): McCall's T scores (mean=50, sd.=10), Weschler IQ scores (mean=100, sd.=15), and Stanines (Scoring and norming use the linear approximation with mean=5, and sd.=2). In all cases, the normative information provided is essentially the distance of the respondent's score from the reference group mean (above or below it) in standard deviation units (e.g. a T score of 70 means that the score of this respondent is two standard deviations above the group mean).

When the module is used in practical sessions, it is important to choose a sample that can be properly considered as a normative sample. Otherwise, the normative information that the transformed scores convey may be meaningless or misleading. We should also point out that standardization changes neither the shape of the distribution of the raw scores or the relative location of the respondent with respect to the reference group.

The percentile rank of a given (integer) raw score is obtained in this module as the rounded percent of the scores in the distribution that lie below the midpoint of the score interval. They are, thus, directly interpretable as the percentage of respondents from the group whose scores fall below the corresponding score. Because rank percentiles provide purely normative information it is important to use a sample that can be properly considered to be normative. We emphasize two results in the practical sessions. First, that percentile ranks provide only ordinal information. And second, that the sampling error of these scores is generally far larger than that of the raw or linearly transformed scores.

The additional information about scale precision is provided by Scoring and norming in the form of confidence intervals around each individual score (raw, linearly transformed, or percentile rank). The classical standard error of measurement is obtained as:

$$se = sd \sqrt{1 - r_{xx}}$$

where  $r_{xx}$  is the reliability of the test scores and  $sd$  the standard deviation of the corresponding scale scores.

The confidence interval (CI) is now computed as:

$$X_i \pm z se$$

where  $z$  is the normal deviate value corresponding to the chosen confidence level. Scoring and norming allows three  $z$  values to be used:  $z=1$  (68% CI or confidence band),  $z=1.65$  (90% CI), and  $z=1.96$  (95% CI).

The confidence interval so far described is based on the so-called traditional approach (e.g. Charter & Feldt, 2001). The main characteristics of this type of interval are that: (a) it is centered on the observed individual score, (b) it uses the classical standard error of measurement obtained from the reliability estimate, and (c) assumes that measurement errors are distributed normally. Assumption (a) is justified by the result that the observed score is an unbiased estimate of the true score, whereas assumption (c) is quite tenable from the conceptualization of random measurement error. The results of the interval so defined can be interpreted as a 'coverage' probability, i.e. the probability that the true score of individual  $i$  lies within the limits of the interval (Charter & Feldt, 2001). Thus, in a 90% confidence interval, the interpretation is that there is a 90% probability that the constructed interval includes the true score of the individual. (Strictly speaking this notion of probability only applies before the measure is taken).



In our practical sessions, we emphasize that for a given score to be meaningfully interpreted, the CI must be narrow, as only in this case we have certainty that the observed score is close to the true score . We also discuss the role of reliability in narrowing the CI.

The lower and upper limits of the chosen CI are next converted to percentiles providing the corresponding percentile bands. These bands can be interpreted as the range of percentile ranks that are likely to represent the corresponding true score at the chosen confidence level. They can also be interpreted as the range of percentile ranks at which the individual would be expected to fall under repeated testing.

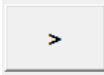
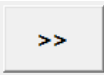
In line with the two aims discussed above, the module has two steps. In the first step, the program provides raw and linearly transformed scores for each individual in the sample, together with the corresponding CIs. and percentile bands. The program can also display all the linear transformations discussed above (standard, T, IQ and Stanine) but only one is displayed. So, the user need to provide a reliability estimate, and choose the confidence level and the desired linear transformation to be displayed (standard, T, IQ, or Stanine).

In the second step, the program provides three pieces of information: **(a)** the descriptive statistics of the raw scores together with the corresponding histogram, **(b)** the frequency table corresponding to the raw scores, and **(c)** the normative table which is organized in three columns: raw score or class interval, linearly transformed raw score, and percentile rank (computed as defined above). Only one linear transformation is displayed and the user must decide which transformation to use. The user must also decide whether the table needs to display all the possible scores (this is recommended for short tests which have a limited range of values) or whether the scores will be grouped in classes. In the latter case, the user can choose among three levels of grouping: 2, 5, and 10. Grouped frequency tables are preferable when the range of raw score values is large.

The normative table can be extended at the user's choice so that each transformed score (both linear and percentile) includes also the corresponding 68% confidence interval or confidence band. This is quite a relevant information: it informs the user about the uncertainty (or lack of accuracy) of the transformed score and so the confidence with which valid and meaningful inferences can be made when interpreting it.

### 8.3. Illustrative example

In this example, we again use the database of 338 respondents and 35 items, which has been used in the previous CIA and IFA illustrative examples. First of all, we have to clear all previous results and data by clicking on the respective buttons to ensure that no information remains in the macro's memory. Now it is time to import the file containing the raw data by clicking on "Import Text File", and selecting the right file and the appropriate column separator. As can be seen, this module has two computing options. We start by clicking on "Individual Transformation" to obtain the raw and transformed scores of all the participants, which is a required step before the normative table is created.

First, you have to select which items you want to use to compute the total scores. By default, all the items are excluded. You can select individual items by clicking on  or all items at once by clicking on . Fill in the reliability estimate field with the reliability of the raw data. This can be computed using the CIA module (Cronbach's alpha), the IFA module (Omega index) or the pre-processor module (to obtain split-half or test-retest estimates). In this example we use a reliability value of 0.91, which was obtained using the split-half method. We shall use a confidence interval of 68% ( $\pm$  Standard Error) here, but 90% and 95% levels are also available. Finally, we have to select which scores will be presented, where the options are: standard Z, I.Q. scores, T scores or Stanines. There is also the option for presenting the confidence interval of each of those scales only. In this example we are going to select the T scores. The window with this configuration is shown in Figure 10.

Select Items ×

**Excluded-Items**

>

>>

**Selected-Items**

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

Enter the reability estimate:

Chose the confidence level:

Show columns

<input type="radio"/> Standard Z	<input type="radio"/> IQ-Scale
<input type="radio"/> C.I (Z)	<input type="radio"/> C.I (IQ)
<input checked="" type="radio"/> T-Scale	<input type="radio"/> Stanine
<input type="radio"/> C.I (T)	<input type="radio"/> C.I (St)

Cancel

Continue

Figure 10. The configuration used in this example for the Individual Transformation function

The output shows the score on each item, the total raw score, the percentile and the transformed scores in the previously selected scale. The confidence interval is also provided for each score. For example, the first participant obtained a total score of 79 with a confidence interval between 74.49 and 83.51. This score represents a percentile 1, where only 1% of the population are expected to have lower scores. In T scores he/she obtained 21(CI=18.12; 24.12), almost three standard deviations below the mean.

Now, all we have to do to create the normative table is to click on “Step 3: Normative Table”. The options displayed are presented in Figure 11.

Select transformation and table options

Standard - Z
  Non-Grouped Data

Scale - T
  Grouped Data (2)

Scale - IQ
  Grouped Data (5)

Stanine
  Grouped Data (10)

Choose a score: Total

Extended table with error bands

Cancel Continue

Figure 11. The options available when creating the normative table

As can be seen, we have to select (a) which scale to use for the table, (b) whether to group the data for the normative table, (c) which scores will be used for creating the normative table (selection only available when using in combination with the Scoring Multiple Choice Tests module) and (d) select if the extended table with error bands should be printed. In this example, we use T scores in (a), and group the scores at level 2 in (b) because the expected range of scores is relatively large and (d) because we are interested in the error bands of the scores. The chosen option in (b) creates a table that is not too lengthy and which fits well on a single page without losing too much information. As well as the normative table built according to specifications (a), (b) and (d), the output displays the descriptive statistics, histogram, and percentile curve plot of the test scores.

The normative table generated will look similar to the one presented in Figure 12, which is the output of the example data using a 5-point grouping.

Raw Score	T-Score	C.I.	Pc	Percentile band
74	18	(18; 18)	0	(1; 2)
84	24	(24; 24)	2	(1; 4)
96	32	(32; 32)	5	(1; 12)
102	36	(36; 36)	8	(1; 26)
107	40	(40; 40)	12	(1; 38)
112	43	(43; 43)	23	(3; 52)
117	46	(46; 46)	36	(7; 64)
122	50	(50; 50)	48	(10; 75)
127	53	(53; 53)	62	(21; 81)
132	56	(56; 56)	75	(34; 85)
137	60	(60; 60)	85	(46; 87)
142	63	(63; 63)	91	(61; 89)
148	67	(67; 67)	96	(74; 90)
155	72	(72; 72)	98	(83; 90)
165	78	(78; 78)	100	(87; 90)

Figure 12. The extended normative table using T-Scores and grouping by 5

#### References

- Charter, R.A. & Feldt, L.S. (2001). Confidence intervals for true scores: Is there a correct approach?. *Journal of Psychoeducational Assessment*, 19, 350-364.
- Petersen, N. S., Kolen, M. J., & Hoover, H. D. (1993). Scaling, norming and equating. In R. L. Linn (Ed.), *Educational measurement* (3rd Edition, pp. 221–263). Phoenix, AZ: Oryx Press.

## 9 The IRT Basics (IRT-B) sub-program

### 9.1 Summary

The IRT-B module performs item calibration and individual scoring on the basis of the one-parameter and the two-parameter logistic models. Unlike the remaining CTT modules, we recommend that IRT-B be used only for teaching purposes. For research purposes or accurate individual assessment, a more sophisticated IRT program such as BILOG-MG or IRT-Pro is recommended. As a teaching tool, however, IRT-B has interesting features and provides useful information.

The IRT-B module only contains one computing button:



This function performs all the analysis at once, using the data in the “Data” tab. Determining which items will be used to perform the analysis and which model will be used (1 Parameter or 2 Parameter) are the only requirements. The output will be generated in two tabs: “BasicIRT” and “Individual Scores”.

---

The output is organized in two parts, which generally coincide with the calibration and scoring stages. The first part first provides the CTT-based item analysis results (including the alpha reliability estimate). Next, the IRT item estimates: discriminations (slopes), and locations are displayed. This first part also provides: (a) the test information values, (b) the descriptive statistics of the raw and IRT scores, and (c) a mean-squared residual statistic for each item in order to assess the appropriateness of the model chosen. Two graphs are available in this part: the item characteristic curve (for the selected items), and the test information function curve.

The second part of the output provides: (a) the raw scores and (b) the individual trait estimates for each respondent together with the corresponding standard errors obtained using Bayes EAP estimation. The test characteristic curve, which is the graph that relates both types of scores, is also included in this part of the output.

### 9.2. Foundations and Details

The IRT-Basics (IRT-B) sub-program aims to provide a didactic introduction to the most basic models in Item Response Theory: the one- and the two-parameter logistic models (e.g. Lord, 1980). Of all the PT modules, IRT-B is the most suitable for instructional purposes although it should be pointed out that the program implements simple and heuristic procedures for item calibration that are not claimed to provide optimal estimates. We have checked that the IRT-B results are generally close to those obtained with more sophisticated programs such as BILOG or MULTILOG. However, we do not recommend IRT-B for use in research or individual assessment.

The analytical approach used in IRT-B has two main stages (calibration and scoring) and a preliminary stage which performs CTT analysis, and allows the most basic relations between CTT and IRT to be explored. One example we give in our practical sessions is that if the classic discrimination indices (item-rest correlations) vary widely between items, we know that the one-parameter model would not be appropriate for this data. The statistics computed in the first stage are: the mean item scores (location or difficulty indices), the item-rest correlations, and the point estimate of coefficient alpha (see the CIA module for more details).

The first and second stages are the standard calibration and scoring stages used in conventional IRT assessment. In the calibration stage, the item parameters are estimated. In the scoring stage, the item estimates are taken as fixed and known, and are used to obtain individual scores and standard errors for each respondent.

The calibration stage implements the simple heuristic procedures based on Lord's (1980) formulas but incorporating the corrections proposed by Schmidt (1977). Let  $P$  be the item mean or proportion of item endorsement (i.e. the classical difficulty index),  $Q=1-P$ ,  $r_{xx}$  the reliability estimate for the test scores, and  $r_{jx}$ , the item-rest correlation (note that all these statistics are obtained in the first-stage CTT analysis). Furthermore, let  $z$  be the normal deviate corresponding to the  $P$  and  $Q$  partition under the standard normal distribution, and  $h(z)$  the ordinate corresponding to  $z$ . The item-test biserial correlation corrected for unreliability is then obtained by

$$rbc_j = \frac{r_{jX} \sqrt{P_j Q_j}}{\sqrt{r_{XX}} h(z_j)}$$

And the discrimination and difficulty indices in the two-parameter model are then obtained as

$$a_j = \frac{rbc_j}{\sqrt{1 - rbc_j^2}}$$

and

$$b_j = \frac{z_j}{rbc_j}$$

If the one-parameter model is to be estimated, the common discrimination index and the difficulty indices are obtained from the average of the  $rbc_j$  values.

In the scoring stage, IRT-B computes Bayes expected a posteriori scores (EAP, Bock & Mislevy, 1982). This is a relatively simple and non-iterative approach that has many advantages and so is implemented in many commercial programs. The EAP point estimate is defined as the expectation of the posterior distribution:

$$EAP = \hat{\theta}_i = E(\theta | \mathbf{x}_i) = \frac{\int \theta L(\mathbf{x}_i | \theta, \mathbf{b}, \mathbf{a}) g(\theta) d\theta}{\int_{\theta} L(\mathbf{x}_i | \theta, \mathbf{b}, \mathbf{a}) g(\theta) d\theta}$$

where  $L$  is the Likelihood function. Our implementation uses standard settings to obtain this estimate. First, the prior for  $\theta$  is taken to be standard normal. Second, the integral expression above is approximated numerically by quadrature. More specifically, IRT-B uses rectangular quadrature in 20 equally-spaced points.

In addition to the EAP point estimate, IRT-B computes the posterior standard deviations (PSDs), which serve as standard errors (e.g. Bock & Mislevy, 1982).

$$PSD(\hat{\theta}_i) = \text{sqrt}(E(\theta^2 | \mathbf{x}_i) - \hat{\theta}_i^2).$$



They are approximated by quadrature in the same way as the point estimates. Overall, then, for each respondent IRT-B provides an estimate of his/her trait level together with the standard error corresponding to this estimate. A global assessment of the accuracy with which the test measures at different trait levels is provided by the test information function in which the amount of information is computed at 20 equally-spaced points between -4 and +4 and the graphical test information curve is obtained by joining the resulting points. In practical teaching, we emphasize the relevance of assessing this curve in order to determine the type and purposes of the test that is analysed. For example, a generic or omnibus personality test is expected to show a relatively flat information curve centred around the mean level of the trait distribution. In contrast, a selection test is expected to show a narrow and peaked information curve centred around the cut-off selection point.

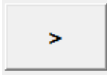
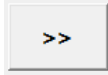
Finally, the module implements a mean-square goodness of model-data fit measure to assess the appropriateness of the chosen model on an item-by-item basis. If we denote by  $\hat{P}_j(\hat{\theta}_i)$  the expected score on item  $j$  for respondent  $i$  evaluated by using the item and person estimates, the mean squared residual for item  $j$  is obtained as

$$MSR(j) = \frac{\sum_{i=1}^N (X_{ij} - P_j(\hat{\theta}_i))^2}{N P_j(\hat{\theta}_i)(1 - P_j(\hat{\theta}_i))}$$

The expected value of MSR under the null hypothesis of item fit is 1. According to Wright and Linacre (1994) MSR values in the range 0.5 to 1.5 indicate that the chosen model fits the item and that this item is productive for measurement.

Overall IRT-B requires a minimum of choices to be run. As in all the PT modules, the user must first select the item set to be analysed and the model to be fitted (one-parameter or two-parameter). Next, three graphical displays are available on option: the item characteristic curve (ICC) for a given item, the test information graph described above, and the test characteristic curve (TCC). The main interest of the ICC is didactic, and we use it to graphically represent the estimated item parameters: discrimination (steepness of the ICC) and difficulty (location of the ICC) parameters. As for the TCC, it is a good tool for (a) explaining the non-linear relation that generally exists between trait levels and expected scores, and (b) illustrating the lack of discriminating power that the raw scores usually have at the extremes of the scale.

### 9.3. Illustrative example

For this example we use a database of 1441 participants who were administered a temperament questionnaire made up of 53 binary items. It can be found in the file “data\_IRT.dat”. As with the other modules, it is recommendable to clear data and results before the file is imported to prevent any interference from previous data or analysis. Now it is time to import the file containing the raw data by clicking on “Import Text File” and selecting the appropriate column separator. The next step is to click on “Compute” and select the items that you want to run the analysis on. By default, all the items are excluded from the analysis. You can select the individual items by clicking on  or all items at once by clicking on . In this example we select all the items because we want to analyze the complete test. We also have to select the model we want to use: the 1-Parameter Model (Rasch) or 2-Parameter Model. In our example we select the 2-Parameter Model because we expect the classic discrimination indices (item-rest correlations) to vary considerably between items.

Under request, the program can provide several graphical displays (see Figure 13).

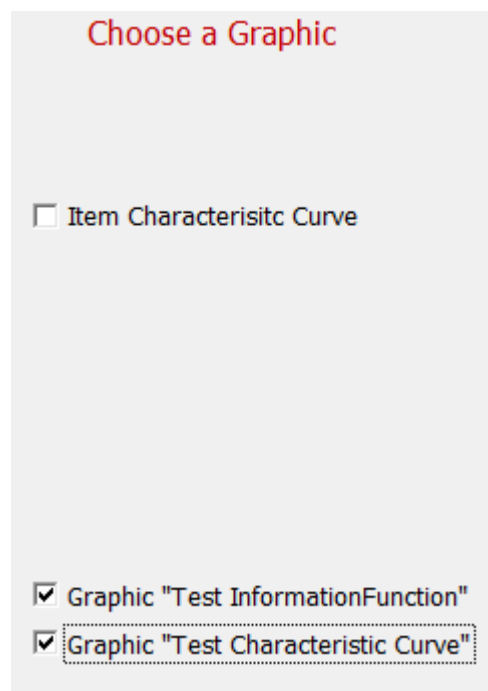


Figure 13. The three possible graphic representations available: Item characteristic curve (for only one selected item at a time), the test information curve or the test characteristic curve.

They are all optional, but for instructional purposes it is recommended to select at least the test information curve and the test characteristic curve. After selecting which plots are to be drawn we click on Continue and wait for the results. Unlike the other modules, this operation can take a while to compute because of the sophistication of the operations, but usually no more than two minutes (although the time does depend heavily on the number of participants and items).

As we have mentioned above, the output is organized in two parts (two tabs). The first tab provides the CTT-based item analysis results and the IRT item estimates. It gives the descriptive statistics of the items (mean and standard deviation), a summary of the test, the corrected item-rest correlation of each item and the Cronbach alpha index. In our practical sessions we stress the need to inspect the item-rest correlation values (low or even negative) and to assess the variability between these correlations. In this example, we detect some items with poor item-rest correlation values. For example, item number 13 is inversely correlated with the other items ( $r = -.330$ ), and some items have values lower than  $r = .150$ : item 9 ( $r = .107$ ), item 11 ( $r = .125$ ), item 17 ( $r = .131$ ), item 5 ( $r = .136$ ) and item 14 ( $r = .138$ ). When the 53 items are used, the Cronbach alpha is only 0.85 because these items have such poor item-rest correlations. Before continuing, we exclude the six items mentioned in an attempt to improve the internal consistency of the questionnaire, and maybe the Cronbach alpha index.

We click on Compute again and repeat the process with the remaining 47 items. Now the item-total correlations change, and the Cronbach alpha is 0.864, a small increase on the value obtained with the full questionnaire. However, considering that Cronbach alpha is highly dependent on the number of items, any improvement with fewer items is commendable.

The IRT part of the output begins by presenting the IRT-based item discriminations and the item difficulties. As expected, the discriminations between the items vary considerably, which supports our decision to select the 2-parameter model. As far as difficulty is concerned, the questionnaire is quite balanced, with some easy items (for example, item 26) and some hard items (for example, item 49). Next we can see the Item Characteristic Curve of the chosen item if we request that plot. Subsequently, the descriptive statistics of the raw and IRT scores are presented, including the mean and standard deviation. For the 47-item version of the questionnaire, the mean value is 24.57 and the standard deviation is 8.31 in the raw score.

The test information values and the graphic representation can also be displayed (if requested) (see Figure 14).

Interval	Information
-4,0	1,267
-3,6	1,602
-3,2	2,035
-2,7	2,602
-2,3	3,361
-1,9	4,403
-1,5	5,861
-1,1	7,887
-0,6	10,455
-0,2	12,493
0,2	12,075
0,6	9,717
1,1	7,227
1,5	5,262
1,9	3,850
2,3	2,869
2,7	2,185
3,2	1,695
3,6	1,335
4,0	1,064

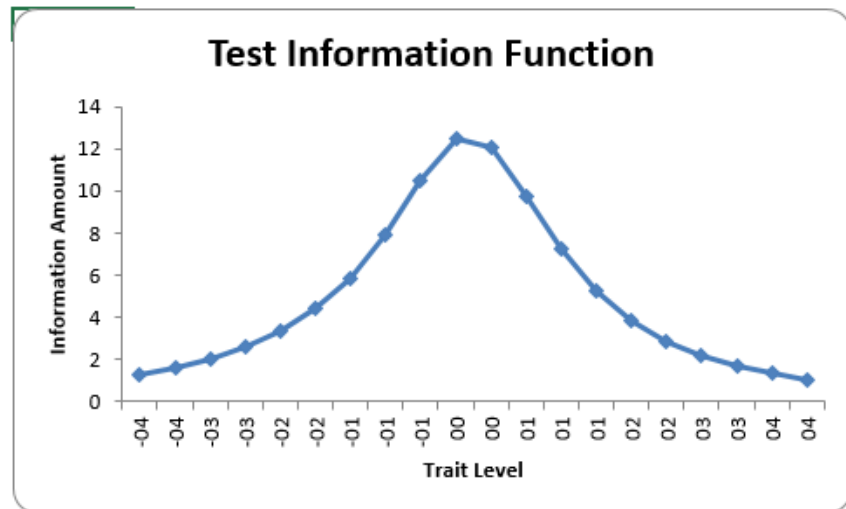


Figure 14. Test information function graph and the trait level in intervals of 0.4. The highest information values were between trait levels of -0.2 and 0.2.

The graph shows that the information curve is not too peaked, has the mode at about the mean score (i.e. near zero) and is relatively symmetrical around this point. This is the expected profile of a personality measure intended for the general population for – for example – screening purposes.

Finally, at the bottom of this tab we can see the mean-squared residual (MSR) statistic for each item, which is very useful for assessing the appropriateness of the chosen model.

On the next tab, “Individual Scores”, the output provides a list of the raw scores of each participant as well as the individual trait estimates (TH scores, which are presented in z scale) and the corresponding standard errors. We use this table to show the students why the test information values are important in individual assessment, and focus on those individuals who have the most or the fewest errors. As we explain, the more similar the trait estimate of one participant is to the point where the test gives most information, the lower the corresponding standard error will be. In our case, the test gives the most information at levels between -0.2 and 0.2, so all the subjects with scores close to 0 will have less standard error than the subjects with scores more distant from zero. This tab also presents the Test Characteristic Curve (Figure 15) if

requested, which can be useful for explaining why the trait estimates discriminate better at the two ends of the graph in comparison with raw scores.

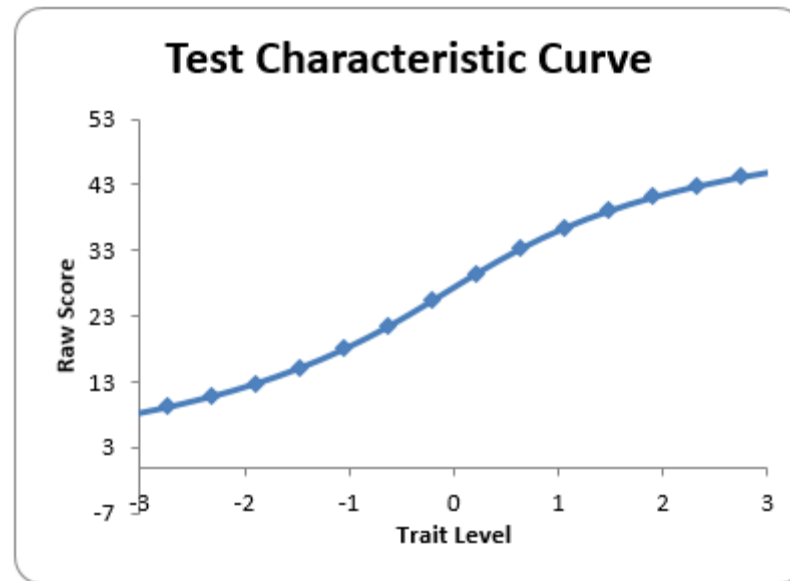


Figure 15. Test Characteristic Curve of the example data.

For example, at the lower end of the graph, two individuals with very similar raw score (for example 7 and 10) actually have quite different trait estimates. This can be seen graphically as the line has gone up very little (raw score) but has shifted further to the right (trait level). At average trait levels (corresponding to the near mean score of the raw scores), the discrimination is better because as mentioned above, the questionnaire gives the most information at levels near 0.

This graphic is based on the table above, where the user can see which expected score (in raw score metrics) corresponds to each individual trait estimate (TH score). For example, a participant with a mean level of the trait (TH score near 0, between -0.2 and 0.2) is expected to have a raw score between 25.430 and 29.558, which are also values around the mean of the raw scores (27.332).

## References

- Bock, R.D. & Mislevy, R.J. (1982). Adaptive EAP estimation of ability in a microcomputer environment. *Applied Psychological Measurement*, 6, 431-444.
- Lord, F. M. (1980). *Applications of Item Response Theory to Practical Testing Problems*. New Jersey: Lawrence Erlbaum.

- Schmidt, F. L. (1977). The Urry method of approximating the item parameters of latent trait theory. *Educational and Psychological Measurement*, 37, 613-620.
- Wright, B.D. & Linacre, J.M. (1994). Reasonable mean-square fit values. *Rasch Measurement Transactions*, 8, 370.

## 10 The Multiple-Choice Analysis (MCA) sub-program

### 10.1. Summary

The MCA sub-program is intended for performance and ability tests that use a multiple-choice item format, with the same number of options for all the items, and that is administered under the following conditions. First, there are no time limits, so that every examinee can attempt every item in the test. Second, examinees are not requested to respond to all of the items. These conditions are submitted to be quite common in most exams the users of PT are familiar with. Furthermore, in these exams, students are usually told that there is a penalty for wrong responses so that guessing is not encouraged. For data collected in the conditions so far described, the MCA function: (a) performs item and option analysis, (b) obtains reliability estimates, and (c) scores examinees using different formula scores.



Step 1: Item  
Analysis

This function performs the item analysis, which are the same as in CIA but with some additions like the number of correct responses, errors and omissions and the difficulty indices. It also computes the score transformation for each participant, similar to the obtained when using the Scoring and Norming module but with the corrections described in the next section.



Step 2: Option  
Analysis

This function performs an analysis (including graphical displays) of the functioning of the different items options (distractors and correct response).

---

The output is organized in three parts. The first part describes the item analysis results, which are the same as in the CIA function with additional information about certain score descriptives (correct responses, errors and omissions) as well as the corrected difficulty indices. The second part describes the option analysis results. The third part, finally, provides uncorrected or corrected (at the user's choice) scores for each examinee.

## 10.2. Foundations and Details

As mentioned at the beginning, the item analysis implemented in the MCA function is the same as that in the CIA function with certain additional results. So, the reader is referred to the CIA part of the guide for the foundation of this type of analysis. Standard summary results include item means, standard deviations, inter-item correlations, squared item-rest multiple correlations, corrected item-total correlations (i.e. discrimination indices), the estimated value of coefficient alpha when each item is deleted from the set, and the alpha estimate from the full scale with the corresponding confidence intervals.

The additional information provided by MCA at this part, are the number of correct responses ( $A$ ), errors ( $E$ ), and omissions ( $O$ ) per item, as well as the corrected item difficulty index. For items with  $k$  response options, the corrected item difficulty provided by the program is obtained as:

$$p_c = \frac{A - \frac{E}{k-1}}{A + E + O} = \frac{A - \frac{E}{k-1}}{Nt}$$

where  $Nt$  is the total number of sample examinees. The formula above is the standard correction for chance success (e.g. Davis, 1951) and is obtained on the assumption that when the examinee who responds the item does not possess enough knowledge to permit selection of the correct answer he/she guesses blindly among all the choices. Another form to arrive at the corrected index above is to request that the  $p_c$  value in a group in which all  $Nt$  examinees would respond at random should be zero. Overall, the  $p_c$  index is intended to be an unbiased estimate of the standard difficulty index  $p$  i.e. the proportion of individuals in the population for which the test is intended that know the correct answer to the item, and so, it should be interpreted in the same way (see the CIA guide).

The option analysis in the second part of the MCA output provides numerical and graphical summaries of the functioning of the different item options including the omitted responses. First, the marginal proportions of endorsement for each option are provided. For the correct option these proportions are indeed



the uncorrected item difficulty indices  $p$ . For the remaining options they provide information about the functioning of the options as a distractors. In an 'ideal' item, the proportions of endorsement for those who failed the item would be evenly distributed among the non-correct options (e.g. Horst, 1933). For example, assume that the item was answered by all the examinees and that 60% provided the correct answer. So, the uncorrected difficulty index is  $p=.6$ . Ideally we should then expect the remaining 40% to be evenly distributed among the different options. An option that is chosen by no one or by too few examinees is ineffective as a distractor and should be replaced. With regards to the uncorrected difficulty index, it has been shown that values slightly above than the midway point between chance ( $1/k-1$ ) and 100 per cent correct answers maximize item discrimination (Lord & Novick, 1968).

The second type of summaries provided in the option analysis are based on the concept that the proportions of endorsement for each option are expected to vary as a function of the ability level. More specifically, for the correct option the proportion of endorsement is expected to increase with ability, whereas it is expected to decrease for the incorrect options (i.e. distractors). A graphical analysis for testing this condition is carried out as follows for each item: (a) examinees are sorted into quartiles based on their overall scores (from lowest to highest), and (b) the proportion of endorsement for each option is obtained in each quartile group. Each option is printed with a different color and pattern, and the correct answer is always displayed in a solid green line. Items with good discrimination power should show the following pattern: as the level of the individuals increases (higher quartiles), the endorsement of the correct answer should improve lineally, while the endorsement of the distractors should decrease systematically. And items with no relation with the participant's proficiency will show a different pattern, probably one where the endorsement of the correct answer remains similar for all the quartiles. As a numerical index, for each option, the product-moment correlation between the four proportions of endorsement and their respective quartiles (1, 2, 3, and 4) is obtained. According to the rationale above, this correlation should be positive for the correct option and negative for the distractors. The analysis so far described can be obtained for any of the overall scores described below.

We turn now to the third part of the MCA output

*“Religion, politics and formula scoring are areas where two informed people often hold opposing ideas with great assurance.” (Lord, 1975, p. 7)*

Two type of individual scores are provided in the MCA output: (a) uncorrected sum scores and (b) raw scores corrected for guessing with standard penalty ( $E/k-1$ ) (i.e. raw score corrected for chance success). As Lord’s quote above suggests, considerable debate exists regarding which is the most appropriate scoring in MC items, and this is indeed a complex issue that also depends on the test instructions and the test-taking behavior of the examinees which, in turn, can be safely assumed to be highly impacted by individual differences (e.g. Prieto & Delgado, 1999). Only a brief discussion will be provided in this guide.

When guessing is present, uncorrected sum scores are (upwardly) biased ability estimates. Furthermore, guessing is expected to increase error variance, and so, uncorrected scores are expected to have decreased reliability and validity. Empirical studies, however, often fail to obtain these decreasing effects and, when obtained, they are usually quite meager. These result, together with (a) the strong assumptions made by the usual correction formulas, (b) the fact that correction differentially affect some types of examines, and (c) the result that when there are not omitted responses, the uncorrected scores and the corrected scores become perfectly correlated (Lord & Novick, 1968, p. 307) have lead some authors to recommend avoiding corrected formula scores and use simple raw scores as the most appropriate schema (e.g. Wurz, 1999).

Among the available correction formulas, the most usual is that corrected for chance success using the standard penalty ( $1/k-1$ ). The correction can be derived by assuming that: (a) there is no partial knowledge, so, each respondent either know or not know the correct response; (b) when the correct response is not known, the examinee guesses totally at random; and (c) each incorrect response is due to guessing. A simple algebraic mechanism for obtaining the penalty in this scenario is to impose that the expected score should be zero when all the items are answered at random.

Assumptions (a), (b) and (c) above are possibly too restrictive and unrealistic in most settings, particularly (a). When partial knowledge exists, the examinee should be able to discard some of the options

(distractors) provided by the item (Horst, 1933), and so, the formula for chance success would be an undercorrection. In other words, the corrected score will still be an upwardly biased estimate. To sum up, under the assumption of partial knowledge, the item penalty should exceed  $1/k-1$  if guessing behavior is to be discouraged (Budescu & Bar Hillel, 1993, Espinosa & Gardeazabal, 2010).

For any of the chosen scoring formulas, the raw or corrected scores can be next linearly and nonlinearly transformed by using the procedures implemented in the “Scoring and Norming” sub-program. Normative tables based on the chosen scores can also be obtained (see the Scoring and Norming part of the user’s guide for details).

### 10.3 Illustrative example

Results by García, Ponsoda & Sierra (2010) based on multiple choice exams suggest that stable and reliable estimates of the item difficulty can be obtained even in samples as low as 50. However, samples as large as 400 might be needed to reach the same stability and reliability results for the discrimination indices. This result may be a limitation in multiple-choice item analysis in which small samples (e.g. classrooms) are commonly used.

This is the case in our example, where the database contains the answers of 95 participants on 50 items which were part from a real exam. The items have 3 options available (from 1 to 3) with only one correct answer for each item. The omissions were allowed and are displayed as 0 values.

The database file has one particularity: unlike the other examples files, this one has no separation character, it contains the answers to the 50 items together, with no separator. So, when importing the file, we have to specify the option “Items are of fixed width”, where there is only 1 character per item, and the number of items are 50 (Figure 16).

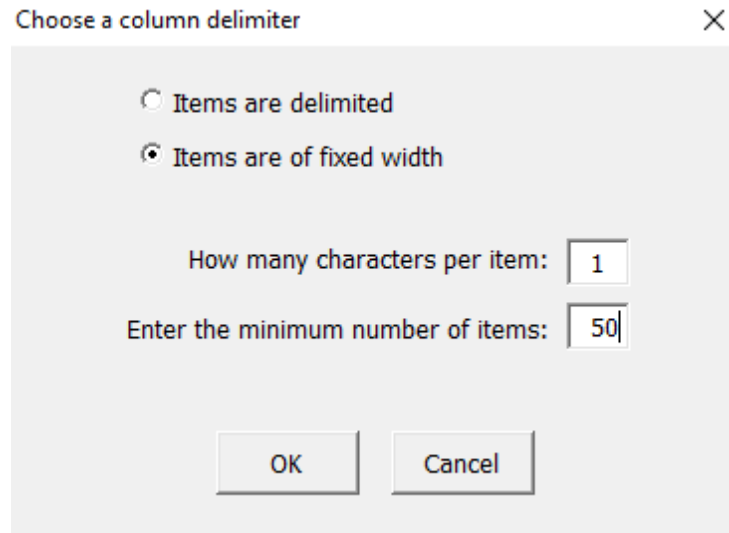


Figure 16. Import selection figure with the required configuration.

Once the dataset file is imported, it is time to perform the Item Analysis function, where we have to specify the number of available options (3) and the correction pattern. In this case, the correct response pattern was the following:

3312321131232211323121123213323233211232

The output is presented in different tabs. The first one contains the Item Analysis section, which show the correct responses, the errors and the omissions for each item. It also show the difficulty indices for each one of them, and in our example we can see that there are some difficult items, like the 38 and 31 (corrected difficulties of 0.11 and 0.14) and others are very easy to guess like items 14 and 34 (corrected difficulties of 0.95 and 0.88), while the majority have medium values.

Next tab shows the individual transformed scores for each participant with the two available formulas:

the direct score (summation of the correct answers) and the corrected formula score of  $A - \frac{E}{k-1}$ . Next, the CIA analysis are presented. In the Descriptive tab we have the inter-item correlation matrix and the classical item descriptives for each item as well as the scale summary, where we can see that the average score of the responders was 23.31. The following tab contains the reliability indices. We can see that the standardized Cronbach Alpha is 0.645, which is pretty low for an ability test. The corrected item-total correlations are low to mediocre, but there are some items with acceptable discriminating power like item 36, others with no discriminating power at all like item number 14, and some others which even have a negative correlation with the rest, like item number 26.

Finally, we are going to compute the second part of the analysis by pressing on Option Analysis, where we have to select the score to analyze: Direct Number-Right Score or corrected formula score. In this example, we are going to select the first one, the direct summation of the item scores. The option analysis tab will be displayed, containing a table with the proportion of endorsement of each option at each proficiency level (i.e. quartiles). The options labeled in green color are the correct ones. The product-moment correlation between the four proportions of endorsement and their respective quartiles (1, 2, 3, and 4) is also obtained. We are going to see some examples, starting with an item with “good” discrimination power, item number 36. In the Figure 17 we can see the endorsement proportions, the product-moment correlation and its graphical representation, where the tendency it is quite obvious: as proficiency level increases (higher quartiles), the higher the probability to choose the correct answer. Thus, in the Q1 only 20% of responders choose the correct option, and in Q4 almost all of them do it (95.24%). This is also very easy to see in the plot, where the green solid line increase almost lineally when increasing the proficiency level of the responders. Also, the product-moment correlation works as expected, with a very high positive correlation for the correct option and negative for the distractors and the omissions.

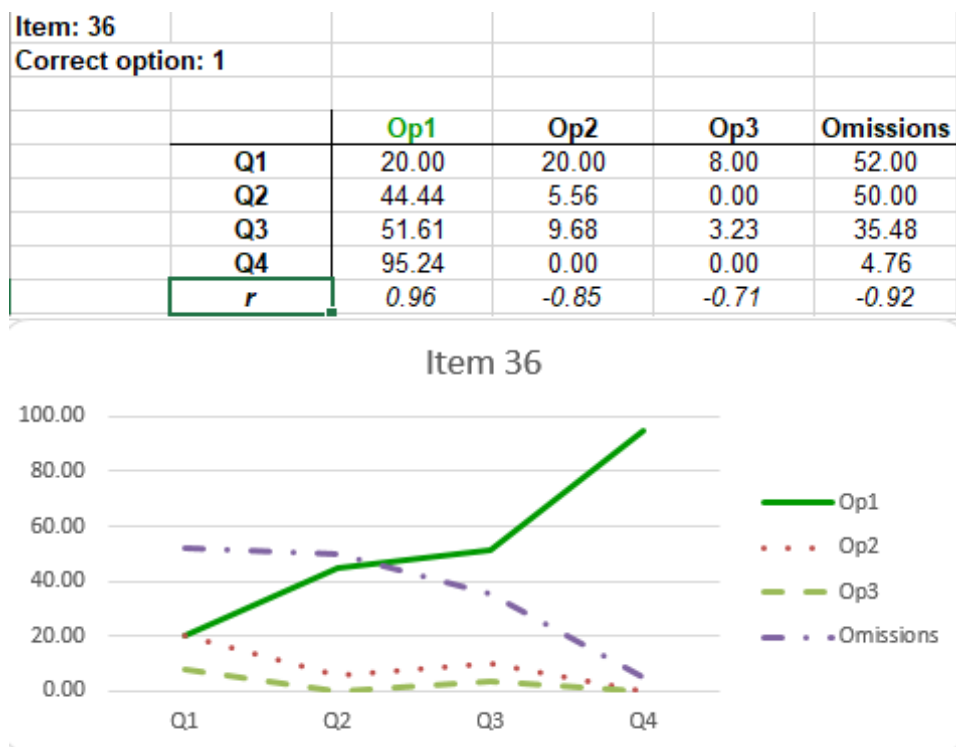


Figure 17. Option Analysis for item number 36.

However, when we look at a “bad” item like item number 14, with a near zero discriminating power value, the results, which are shown in Figure 18, are quite different. Almost everybody answers this item correctly no matter their level of proficiency. In fact, no one has chosen one of the distractors (option number 1), and barely the other one. That it the reson why the product-moment correlations are almost uninterpretable (for the option 1 is even uncalculable), and this item becomes useless for assesing the level of proficiency of the participants.

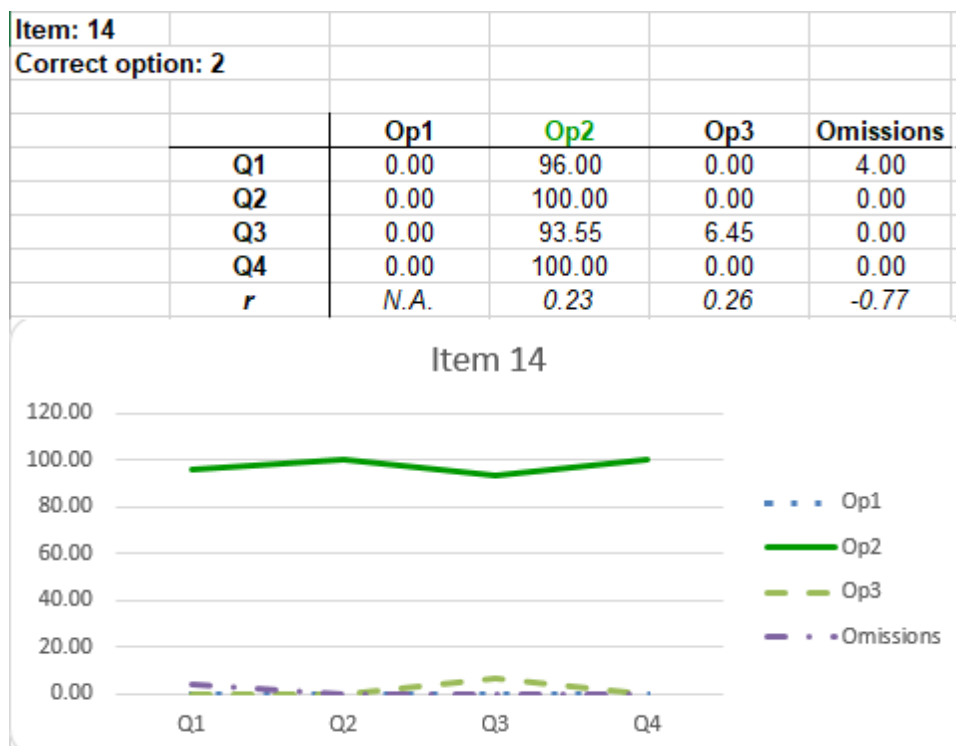


Figure 18. Option Analysis for item number 14.

## References

- Budescu, D., & Bar-Hillel, M. (1993). To guess or not to guess: A decision-theoretic view of formula scoring. *Journal of Educational Measurement*, 30, 277-291.
- Davis, F. B. (1951). Item selection techniques. In E.F. Lindquist (Ed.) *Educational measurement*, pp.266-328. Washington: American Council of Education.
- Espinosa, M. P., & Gardezabal, J. (2010). Optimal correction for guessing in multiple-choice tests. *Journal of Mathematical psychology*, 54, 415-425.

- García, C. G., Ponsoda, V., & Sierra, A. (2010). ¿Cómo evaluamos? Análisis de ítems de opción múltiple y su relación con errores en la construcción. In *Actas del XI Congreso de Metodologías de las Ciencias Sociales y de la Salud: Málaga, 15-18 septiembre de 2009* (pp. 344-349).
- Horst, P. (1933). The difficulty of a multiple choice test item. *Journal of educational psychology*, 24, 229.
- Kurz, T. B. (1999). A review of scoring algorithms for multiple-choice tests. *Paper presented at the annual meeting of the Southwest Educational Research Association, San Antonio, TX.*
- Lord, F. M. (1975). Formula scoring and number-right scoring. *Journal of Educational Measurement*, 12, 7-11.
- Lord, F.M., & Novick, M.R. (1968). *Statistical theories of mental test scores*. Reading (MA): Addison-Wesley.
- Prieto, G., & Delgado, A. R. (1999). The role of instructions in the variability of sex-related differences in multiple-choice tests. *Personality and individual differences*, 27, 1067-1077.