

Modelling a Dichotomously Scored Multiple Choice Test with the Rasch Model

Ray Adams and Margaret Wu. 22 August 2010

Multiple choice items are perhaps the most widely applied tool in testing. This is particularly true in the case of the testing of the cognitive abilities or achievements of a group of students.¹ The analysis of the basic properties of dichotomous items and of tests containing a set of dichotomous items is the simplest application of ConQuest. This sample analysis, shows how ConQuest can be used to fit Rasch's simple logistic model (Rasch, 1980) to data gathered with a multiple choice test. ConQuest can also generate a range of traditional test item statistics.²

The files used in this sample analysis are:³

<code>ex1.cqc</code>	The command statements.
<code>ex1.dat</code>	The data.
<code>ex1.lab</code>	The variable labels for the items on the multiple choice test.
<code>ex1.shw</code>	The results of the Rasch analysis.
<code>ex1.itn</code>	The results of the traditional item analyses.

The data used in this tutorial comes from a 12-item multiple-choice test that was administered to 1000 students. The data have been entered into the file `ex1.dat`, using one line per student. A unique student identification code has been entered in columns 1 through 5, and the students' responses to each of the items have been recorded in columns 12 through 23. The response to each item has been allocated one column; and the codes a, b, c and d have been used to indicate which alternative the student chose for each item. If a student failed to respond to an item, an M has been entered into the data file. An extract from the data file is shown in Figure 1.

¹ The term 'student' or 'students' is used to indicate the object of the measurement process, that is, the entity that is being measured. This term has been chosen because most of the sample analyses are set in an educational context where the object of measurement is typically a student. The methods, however, are applicable well beyond the measurement of students.

² The analysis of dichotomous tests with traditional methods is usually referred to as classical test theory.

³ The last two files are created when the command file is executed.

	1	2
	12345678901234567890123	(column numbers) ⁴
40016	acdabaeada	acd
655	acdcccceba	ca
31140	eccdbcebb	acb
.	.	.
50321	dabcMcebd	aca
30782	acddbcebb	acc
.	.	.
.	.	.

Figure 1 Extract from the Data File **ex1.dat**

In this sample analysis, Rasch's ([1960] 1980) simple logistic model will be fitted to the data, and traditional item analysis statistics are generated. The contents of a command file that can be used to analyse these data are shown in Figure 2.

```
1. 5 Datafile ex1.dat;  
2. Format id 1-5 responses 12-23;  
3. Labels << ex1.lab;  
4. Key acddbcebbacc ! 1;  
5. Model item;  
6. Estimate;  
7. Show >> ex1.shw;  
8. Itanal >> ex1.itn;  
9. Plot icc;  
10. Plot mcc ! legend=yes;
```

Figure 2 Sample Command File for a Dichotomous Test

1. The `datafile` statement indicates the name and location of the data file. Any file name that is valid for the operating system you are using can be used here.
2. The `format` statement describes the layout of the data in the file `ex1.dat`. This `format` statement indicates that a field that will be called `id` is located in columns 1 through 5 and that the responses to the items are in columns 12 through 23 of the data file. Every `format` statement must give the location of the responses. In fact, the explicit variable `responses` must appear in the `format` statement or ConQuest will not run. In this particular sample analysis, the responses are those made by the students to the multiple choice items; and, by default, `item` will be the implicit variable name that is used to indicate these responses. The levels of the `item` variable (that is, item 1, item 2 and so on) are implicitly identified through their location within the set of responses (called the response block) in the `format` statement; thus, in this sample analysis, the data for item 1 is located in column 12, the data for item 2 is in column 13, and so on.

⁴ In each of the listings of the data file, column so you can easily identify the data column. The actual ConQuest data file does not have any column labels.

⁵ In each of the listings of the command file, each statement is labelled so that it can be easily referred to in the text. The actual ConQuest command file does not have any statement labels.

EXTENSION: *The item numbers are determined by the order in which the column locations are set out in the response block. If you use the following:*

`format id 1-5 responses 12-23 ;`

item 1 will be read from column 12. If you use:

`format id 1-5 responses 23,12-22 ;`

item 1 will be read from column 23.

TIP: *In some testing contexts, it may be more informative to refer to the response variable as something other than `item`. Using the variable name `task` or `question` may lead to output that is better documented. Altering the name of the response variable is easy. If you want to use the name `tasks` rather than `item`, simply add an option to the `format` statement as follows:*

`format id 1-5 responses 12-23 ! tasks (12) ;`

The variable name `tasks` must then be used to indicate the response variable in other ConQuest commands. For example in statement 5 in Figure 2.

- The `labels` statement indicates that a set of labels for the variables (in this case, the items) is to be read from the file `ex1.lab`. An extract of `ex1.lab` is shown in Figure 3. (This file must be text only; if you create or edit the file with a word processor, make sure that you save it using the text only option.) The first line of the file contains the special symbol `===>` (a string of three equals signs and a greater than sign) followed by one or more spaces and then the name of the variable to which the labels are to apply (in this case, `item`). The subsequent lines contain two pieces of information separated by one or more spaces. The first value on each line is the level of the variable (in this case, `item`) to which a label is to be attached, and the second value is the label. If a label includes spaces, then it must be enclosed in double quotation marks (`" "`). In this sample analysis, the label for item 1 is `BSMMA01`, the label for item 2 is `BSMMA02`, and so on.

TIP: *Labels are not required by ConQuest, but they improve the readability of any ConQuest printout, so their use is strongly recommended.*

```
====> item
1      BSMMA01
2      BSMMA02
3      BSMMA03
4      BSMMA04
5      BSMMA05
6      BSMMA06
7      BSMSA07
8      BSMSA08
9      BSMSA09
10     BSMSA10
11     BSMSA11
12     BSMSA12
```

Figure 3 Contents of the Label File `ex1.lab`

4. The `key` statement identifies the correct response for each of the multiple choice test items. In this case, the correct answer for item 1 is `a`, the correct answer for item 2 is `c`, the correct answer for item 3 is `d`, and so on. The length of the argument in the `key` statement is 12 characters, which is the length of the response block given in the `format` statement.

If a `key` statement is provided, ConQuest will recode the data so that any response `a` to item 1 will be recoded to the value given in the `key` statement option (in this case, 1). All other responses to item 1 will be recoded to the value of the `key_default` (in this case, 0). Similarly, any response `c` to item 2 will be recoded to 1, while all other responses to item 2 will be recoded to 0; and so on.

5. The `model` statement must be provided before any traditional or item response analyses can be undertaken. When undertaking simple analyses of multiple-choice tests, as in this example, the argument for the `model` statement is the name of the variable that identifies the response data that are to be analysed (in this case, `item`).
6. The `estimate` statement initiates the estimation of the item response model.

NOTE: *The order in which commands can be entered into ConQuest is not fixed. There are, however, logical constraints on the ordering. For example, **show** statements cannot precede the **estimate** statement, which in turn cannot precede the **model**, **format** or **datafile** statements.*

7. The `show` statement produces a sequence of tables that summarise the results of fitting the item response model. In this case, the redirection symbol (`>>`) is used so that the results will be written to the file `ex1.shw` in your current directory. If redirection is omitted, the results will be displayed on the console (or in the output window for the GUI version).
8. The `itanal` statement produces a display of the results of a traditional item analysis. As with the `show` statement, the results are redirected to a file (in this case, `ex1.itn`).
9. The `plot icc` statement will produce 12 item characteristic curve plots, one for each item. The plots will compare the modelled item characteristic curves with the

empirical item characteristic curves. Note that this command is not available in the console version of ConQuest.

- The plot mcc statement will produce 12 category characteristic curve plots, one for each item. The plots will compare the modelled item characteristic curves with the empirical item characteristic curves (for correct answers) and will also show the behavior of the distractors. Note that this command is not available in the console version of ConQuest.

RUNNING THE MULTIPLE CHOICE SAMPLE ANALYSIS

To run this sample analysis, start the GUI version. Open the file ex1.cqc and choose **Run**→**Run All**.

Alternatively, you can launch the console version of ConQuest, by typing the command⁶

```
ConQuestCMD ex1.cqc
```

ConQuest will begin executing the statements that are in the file ex1.cqc and as they are executed, they will be echoed on the screen (or output window). When ConQuest reaches the estimate statement, it will begin fitting Rasch's simple logistic model to the data, and as it does so it will report on the progress of the estimation. This particular sample analysis will take 46 iterations to converge. Figure 4 is an extract of the information that is provided during the estimation (in this case, the changes in the estimates after four iterations).

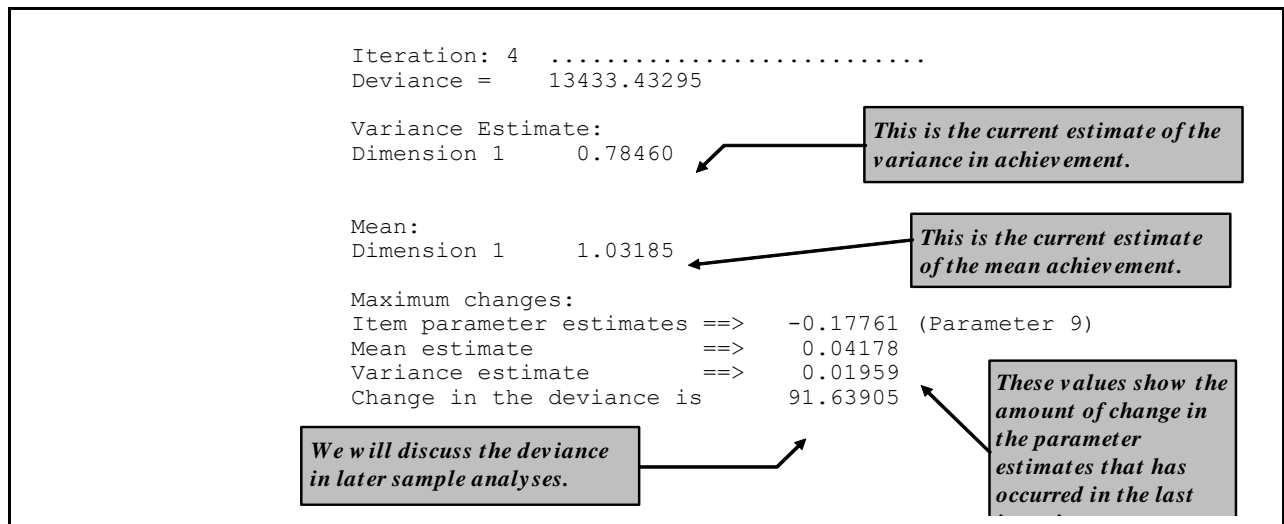


Figure 4 Reported Information on the Progress of Estimation

After the estimation is completed, the two statements that produce text output (show and itanal) will be processed and then, in the case of the GUI version two sets of 12 plots will be produced. In this case, the show statement will produce all six of its tables. All of these tables will be in the file ex1.shw. The contents of the first table are shown in Figure 5.

This table is provided for cross-referencing and record-keeping purposes. It indicates the data set that was analysed, the format that was used to read the data, the model that was requested and the sample size. It also provides the number of parameters that were estimated, the number of iterations that the estimation took, and the reason for the termination of the estimation. The

⁶ If you wish to launch ConQuest in this fashion on command-based systems, ConQuestCMD.exe must be in the directory you are working in or a path must have been set up; otherwise, you must type the entire path name.

deviance is a statistic that indicates how well the item response model has fit the data; it will be discussed further in future sample analyses.

As Figure 5 shows, in this analysis 13 parameters were estimated. They are: (a) the mean and variance of the latent achievement that is being measured by these items and (b) 11 item difficulty parameters. Following the usual convention of Rasch modelling, the mean of the item difficulty parameters has been made zero, so that a total of 11 parameters is required to describe the difficulties of the 12 items.

```
=====
ConQuest: Generalised Item Response Modelling Software  Tue Oct 03 10:19 2006
SUMMARY OF THE ESTIMATION
=====
Estimation method was: Gauss-Hermite Quadrature with 15 nodes
Assumed population distribution was: Gaussian
Constraint was: DEFAULT
The Data File: ex1.dat
The format: id 1-5 responses 12-23
The regression model:
Grouping Variables:
The item model: item
Sample size: 1000
Final Deviance:      13274.87603
Total number of estimated parameters: 13
The number of iterations: 46
Termination criteria: Max iterations=1000, Parameter Change= 0.00010
                    Deviance Change= 0.00010
Iterations terminated because the deviance convergence criteria was reached
Random number generation seed:      1.00000
Number of nodes used when drawing PVs: 2000
Number of nodes used when computing fit: 200
Number of plausible values to draw: 5
Maximum number of iterations without a deviance improvement: 100
Maximum number of Newton steps in M-step: 10
Value for obtaining finite MLEs for zero/perfects:      0.30000
=====
```

Figure 5 Summary Table

Figure 6 shows the second table from the file `ex1.shw`. This table gives the parameter estimates for each of the test items along with their standard errors and some diagnostics tests of fit. The estimation algorithm and the methods used for computing standard errors and fit statistics are discussed in Chapter 12 of the ConQuest 2 manual (Wu, Adams, Wilson and Haldane, 2007). In brief, the item parameter estimates are marginal maximum likelihood estimates obtained using an EM algorithm, the standard errors are asymptotic estimates given by the inverse of the hessian, and the fit statistics are residual-based indices that are similar in conception and purpose to the weighted and unweighted fit statistics that were developed by Wright and Stone (1979) and Wright and Masters (1982) for Rasch's simple logistic model and the partial credit model respectively.

For the MNSQ fit statistics we provide a ninety-five percent confidence interval for the expected value of the MNSQ (which under the null hypothesis is 1.0). If the MNSQ fit statistic lies outside that interval then we reject the null hypothesis that the data conforms to the model. If the MNSQ fit statistic lies outside the interval then the corresponding T statistics will have an absolute value that exceeds 2.0.

At the bottom of the table an item separation reliability and chi-squared test of parameter equality are reported. The separation reliability is as described in Wright and Stone (1979). This indicates how well the item parameters are separated; it has a maximum of one and a minimum of zero. This value is typically high and increases with increasing sample sizes. The

null hypothesis for the chi-square test is equality of the set of parameters. In this case equality of all of the parameters is rejected because the chi-square is significant. This test is not useful here, but will be of use in other contexts, where parameter equivalence (eg rater severity) is of concern.

```

=====
ConQuest: Generalised Item Response Modelling Software Tue Oct 03 10:19 2006
TABLES OF RESPONSE MODEL
=====
TERM 1: item
=====

```

VARIABLES		UNWEIGHTED FIT					WEIGHTED FIT		
item	ESTIMATE	ERROR^	MNSQ	CI	T	MNSQ	CI	T	
1	BSMMA01	0.364	0.050	0.87	(0.91, 1.09)	-3.1	0.90	(0.94, 1.06)	-3.4
2	BSMMA02	-0.177	0.052	0.99	(0.91, 1.09)	-0.3	0.98	(0.92, 1.08)	-0.4
3	BSMMA03	-0.024	0.051	0.89	(0.91, 1.09)	-2.5	0.93	(0.93, 1.07)	-2.1
4	BSMMA04	0.837	0.049	0.97	(0.91, 1.09)	-0.6	0.97	(0.95, 1.05)	-1.2
5	BSMMA05	1.179	0.049	1.10	(0.91, 1.09)	2.2	1.07	(0.95, 1.05)	2.9
6	BSMMA06	-0.313	0.052	1.01	(0.91, 1.09)	0.3	1.01	(0.92, 1.08)	0.3
7	BSMSA07	-0.391	0.053	1.09	(0.91, 1.09)	2.0	1.04	(0.92, 1.08)	0.8
8	BSMSA08	-0.326	0.053	1.15	(0.91, 1.09)	3.3	1.09	(0.92, 1.08)	2.2
9	BSMSA09	-0.966	0.056	0.91	(0.91, 1.09)	-2.2	0.97	(0.88, 1.12)	-0.5
10	BSMSA10	-0.391	0.053	1.11	(0.91, 1.09)	2.4	1.06	(0.92, 1.08)	1.4
11	BSMSA11	-0.499	0.053	0.91	(0.91, 1.09)	-2.0	0.97	(0.91, 1.09)	-0.7
12	BSMSA12	0.707*	0.172	1.02	(0.91, 1.09)	0.4	1.02	(0.95, 1.05)	0.6

This table provides parameter estimates for every term in the model statement. In this sample analysis, we have just one term: item.

The asterisk indicates that this parameter has been constrained.

Figure 6 Item Parameter Estimates

EXTENSION: In Rasch modelling, it is usual to identify the model by setting the mean of the item difficulty parameters to zero. This is also the default behaviour for ConQuest, which automatically sets the value of the 'last' item parameter to ensure an average of zero. In ConQuest, however, you can, as an alternative, choose to set the mean of the latent ability distribution to zero. To do this, use the `set` command as follows:

```

set constraints=cases ;

```

If you want to use a different item as the constraining item, then you can read the items in a different order. For example:

```

format id 1-5 responses 12-15, 17-23, 16;

```

would result in the constraint being applied to the item in column 16. But be aware, it will now be called item 12, not item 5, as it is the twelfth item in the response block.

The third table in the `show` statement's output (not shown for the sake of brevity) gives the estimates of the population parameters. In this case, these are simply estimates of the mean of

the latent ability distribution and of the variance of that distribution. In this case, the mean is estimated as 1.070, and the variance is estimated as 0.866.

This table also provides a set of reliability indices.

The fourth table in the output, Figure 7, provides a map of the item difficulty parameters.

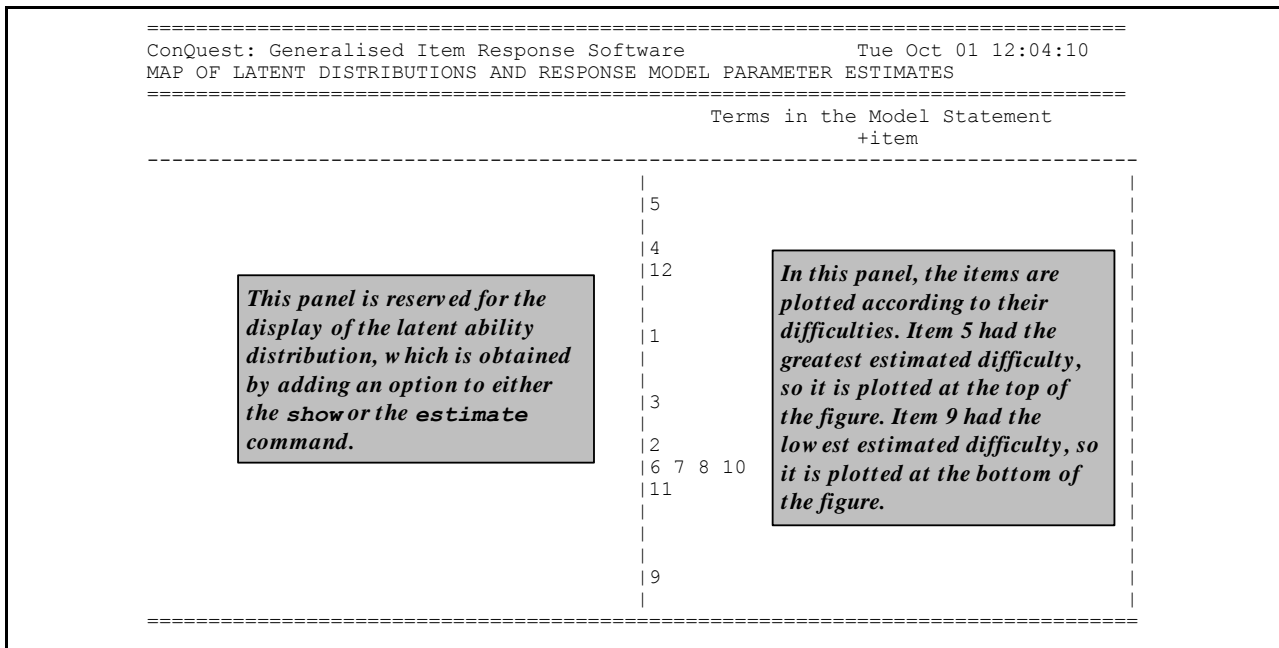


Figure 7 The Item and Latent Distribution Map for the Simple Logistic Model

The file `ex1.shw` contains one additional table, labelled Map of Latent Distributions and Thresholds. In the case of dichotomously scored items and a model statement with a single term⁷, these maps provide the same information as that shown in Figure 7, so they are not discussed further.

The traditional item analysis is invoked by the `itanal` statement, and its results have been written to the file `ex1.itn`. The `itanal` output includes a table showing classical difficulty, discrimination, and point-biserial statistics for each item. Figure 8 shows the results for item 2.

Summary results, including coefficient alpha for the test as a whole, are printed at the end of the file `ex1.itn` as shown in Figure 9. Discussion of the usage of the statistics can be found in any standard text book, such as Crocker and Algina (1986).

⁷ In this case the single term was "item".

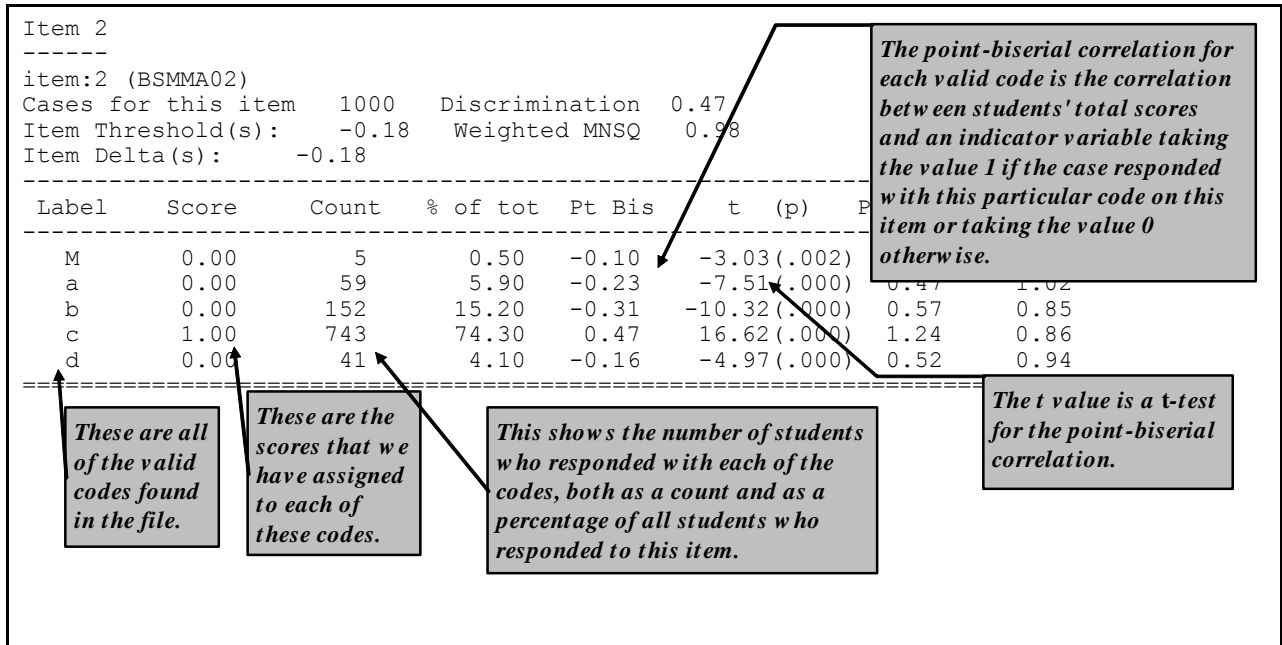


Figure 8 Example of Traditional Item Analysis Results

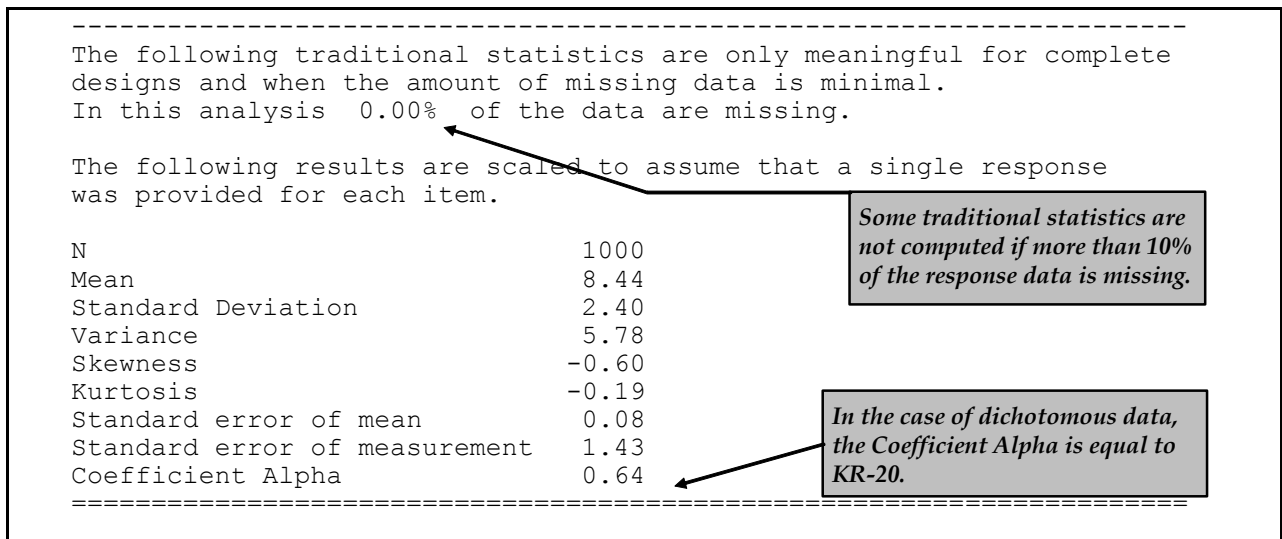


Figure 9 Summary Statistics from Traditional Item Analysis Results

TIP: Whenever a **key** statement is used, the `itana1` statement will display results for all valid data codes. If a **key** statement is not used, the `itana1` statement will display the results of an analysis done after recoding has been applied.

Figure 10 shows one of the 12 plots that were produced by the `plot icc` command. The ICC plot shows a comparison of the empirical item characteristic curve (the broken line, which is based directly upon the observed data) with the modelled item characteristic curve (the smooth line).

Figure 11 shows a matching plot produced by the `plot mcc` command. In addition to showing the modelled curve and the matching empirical curve, this plot shows the

characteristics of the incorrect responses – the distractors. In particular it shows the proportion of students in each of a sequence of ten ability groupings⁸ that responded with each of the possible responses.

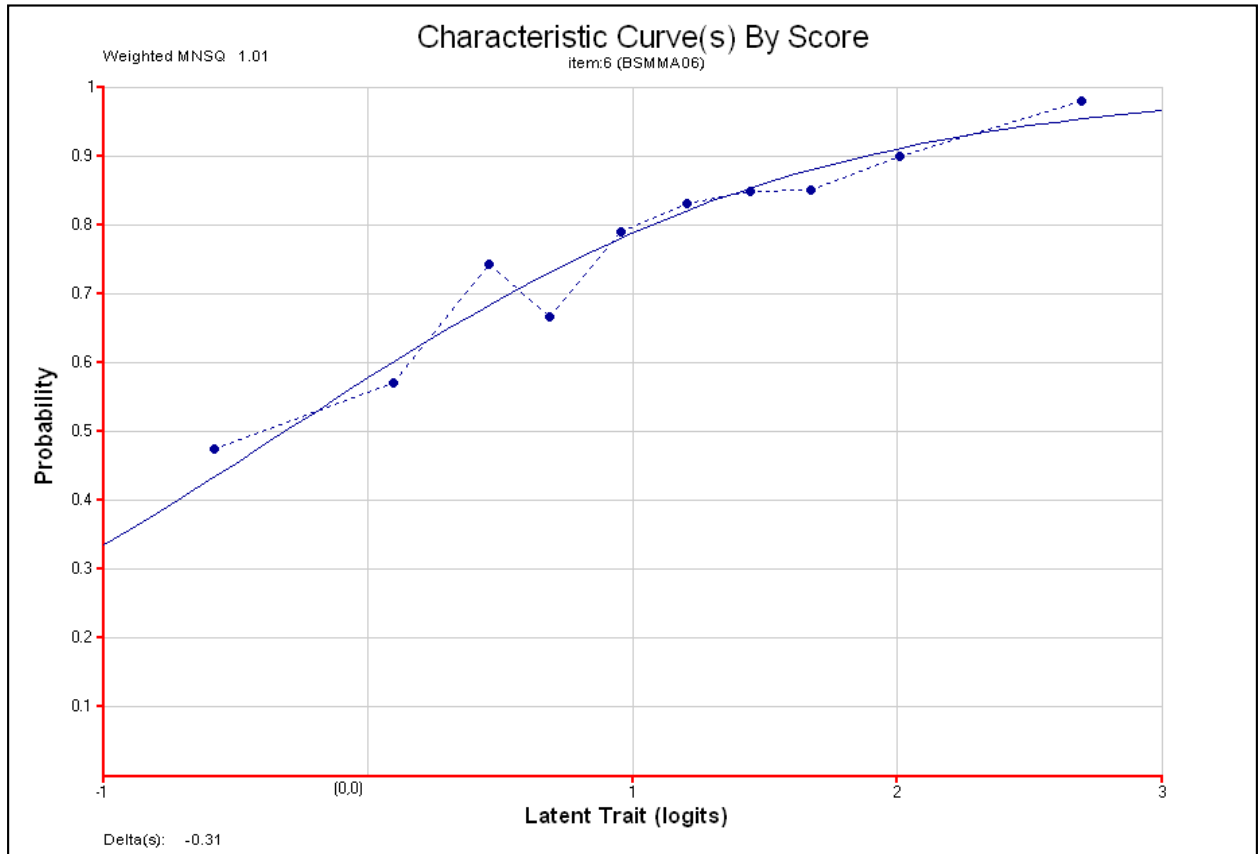


Figure 10 Modelled and Empirical Item Characteristic Curves for Item 6

SUMMARY

This tutorial shows how ConQuest can be used to analyse a multiple-choice test with Rasch's simple logistic model and classical techniques. Some key points covered in this tutorial are:

- the `datafile`, `format` and `model` statements are prerequisites for data set analysis.
- the `key` statement provides an efficient method for scoring multiple choice tests.
- the `estimate` statement is used to fit an item response model to the data.
- the `itanal` statement generates traditional item statistics.
- the `plot` statement displays graphs which illustrate the relationship between the empirical data and the model's expectation.

⁸ Ten ability groupings is a default setting that can be altered.

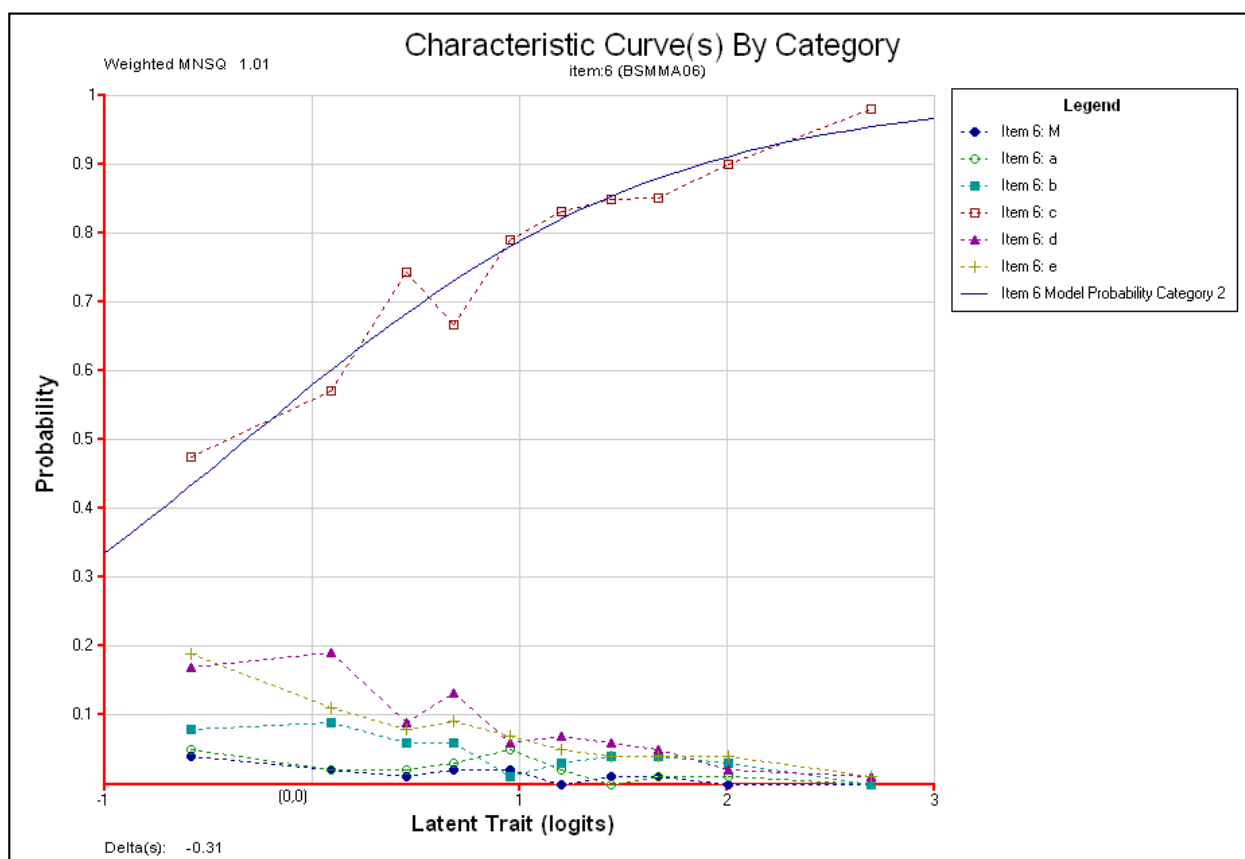


Figure 11 Modelled and Empirical Category Characteristics Curves for Item 6

EXTENSION: *ConQuest can fit other models to multiple choice tests, including models such as the two parameter logistic model, the nominal response model and the ordered partition model.*

REFERENCES

- Crocker, L., and Algina, J. 1986. *Introduction to Classical and Modern Test Theory*. Orlando, Florida: Holt Rinehart and Winston.
- Rasch, G. 1980. *Probabilistic Models for Some Intelligence and Attainment Tests*. Chicago: University of Chicago Press (original work published 1960).
- Wright, B. D., and Masters, G. N. 1982. *Rating Scale Analysis: Rasch Measurement*. Chicago: MESA Press
- Wright, B. D., and Stone, M. 1979. *Best Test Design: Rasch Measurement*. Chicago: MESA Press.
- Wu, M. L., Adams, R. J., Wilson, M. R., Haldane, S.A. (2007). *ACER ConQuest Version 2: Generalised item response modelling software* [computer program]. Camberwell: Australian Council for Educational Research.