

Appendices A–G for
**The Theory and Practice of Item Response
 Theory, Second Edition**

by R. J. de Ayala
 Guilford Publications, Inc.

Appendix A. Maximum Likelihood Estimation of Person Locations	1
Estimating an Individual's Location: Empirical Maximum Likelihood Estimation	2
Estimating an Individual's Location: Newton's Method for MLE	3
R Function for MLE of θ with the Rasch Model	10
Revisiting Zero Variance Binary Response Patterns	10
Appendix B. Maximum Likelihood Estimation of Item Locations	14
R function for MLE of δ with the Rasch Model	17
Appendix C. The Normal Ogive Models	21
Conceptual Development of the Normal Ogive Model	21
The Relationship between IRT Statistics and Traditional Item Analysis Indices	26
Relationship of the Two-Parameter Normal Ogive and Logistic Models	29
Extending the Two-Parameter Normal Ogive Model to a Multidimensional Space	32
Appendix D. Computerized Adaptive Testing	35
A Brief History	36
Fixed-Branching Techniques	37
Variable-Branching Techniques	37
Advantages of Variable-Branching over Fixed-Branching Methods	38
IRT-Based Variable-Branching Adaptive Testing Algorithm	39
Appendix E. Linear Logistic Test Model (LLTM)	46
Example of LLTM Calibration Using eRm	49
Appendix F. Mixture Models	61
Latent Class Analysis	61
Mixture Rasch Model	64
Example: Application of the Mixture Rasch Model to Writing Problem Data, CMLE, WINMIRA	66
Example: Application of the Mixture Rasch Model to Writing Problem Data, CMLE, psychomix	77

(continued)

Appendix G. Miscellanea	89	
Using Principal Axis for Estimating Item Discrimination		89
Infinite Item Discrimination Parameter Estimates	90	
Example: NOHARM Unidimensional Calibration	91	
An Approximate Chi-Square Statistic for NOHARM	95	
Relative Efficiency, Monotonicity, and Information	97	
FORTRAN Formats	99	
Odds, Odds Ratios, and Logits	100	
The Person Response Function	104	
Linking: A Temperature Analogy Example	107	
Should DIF Analyses Be Based on Latent Classes?	108	
The Separation and Reliability Indices	110	
Dependency in Traditional Item Statistics and Observed Scores	111	
Conditional Independence Using Q_3	117	
Standalone NOHARM Calibration of Interpersonal Engagement Instrument, M2PL Model	119	
CFI, GFI, M_2 , RMSEA, TLI, and SRMR	125	
An Introduction to Kernel Equating	127	
Correspondence between the Rasch Model and a Loglinear Model	129	
R Introduction	136	

Appendix A

Maximum Likelihood Estimation of Person Locations

In this appendix we demonstrate using the likelihood of an individual's observed response vector to estimate their location. We present two approaches, the first is a simplistic approach, whereas the second is a more sophisticated strategy that is commonly used. For both approaches we assume that we know the item parameters.

In general, the probability of a response vector, \underline{x} , is given by

$$p(\underline{x} | \theta, \underline{\vartheta}) = \prod_{j=1}^L p_j(\theta)^{x_j} (1 - p_j(\theta))^{(1-x_j)}, \quad (\text{A.1})$$

where p_j is short for $p(x_j | \theta, \alpha, \delta_j)$, x_j is the binary response to item j , L is the number of items on the instrument, $\underline{\vartheta}$ is a matrix containing the item parameters (e.g., α and δ_j s), and “ \prod ” is the product symbol. Once the responses are observed this expression becomes a likelihood function (Hambleton & Swaminathan, 1985). In other words, the likelihood of person i 's observed response vector, \underline{x}_i , is given by

$$L(\underline{x}_i | \theta_i, \underline{\vartheta}) = \prod_{j=1}^L p_j^{x_{ij}} (1 - p_j)^{(1-x_{ij})}$$

and

$$\ln L(\underline{x}_i | \theta_i, \underline{\vartheta}) = \sum_{j=1}^L [x_{ij} \ln(p_j) + (1 - x_{ij}) \ln(1 - p_j)], \quad (\text{A.2})$$

where $\ln L(\underline{x}_i | \theta_i, \underline{\vartheta})$ is the log likelihood function. The location of the maximum of the log likelihood function is the same as for the likelihood function. In the following we use the log likelihood function and for notational convenience symbolize it as $\ln L$.

ESTIMATING AN INDIVIDUAL'S LOCATION: EMPIRICAL MAXIMUM LIKELIHOOD ESTIMATION

Empirical maximum likelihood estimation (MLE) is a comparatively crude method of determining the location of the maximum of a likelihood function. Its main advantage is that it does not require knowledge of a function's derivatives and therefore is useful for initial or exploratory work. In this approach the maximum may be determined by performing a binary search of the $\ln L$ throughout the θ range of interest (this is conceptually equivalent to the bisection method used in numerical analysis). We start by setting a lower bound (LB) and an upper bound (UB) at, say -3.0 and 3.0 , respectively. This range is bisected (the initial $\hat{\theta}$ is $\theta_0 = 0.0$) and we determine whether $\ln L$ is greater above or below $\hat{\theta}_0$. If $\ln L$ is less than its value at $\hat{\theta}_0$, then the next iteration has a new UB set at 0 (i.e., $\hat{\theta}_0$) and the range between this new UB and the LB is bisected. Therefore, the revised $\hat{\theta}_1$ is -1.5 , the halfway point between -3.0 and 0.0 . Again, we determine whether $\ln L$ is greater above or below $\hat{\theta}_1$ and the lower/upper bound is appropriately reset. This process continues until the θ at which $\ln L$ has its maximum is determined to a desired degree of accuracy. Applying this approach to the log likelihood for the pattern 11000 is shown in Figure A.1. The vertical line in the body of the graph shows that the location of the maximum of the log likelihood for the response pattern 11000 occurs at approximately -0.85 . This value would be our $\hat{\theta}$ for this response pattern.

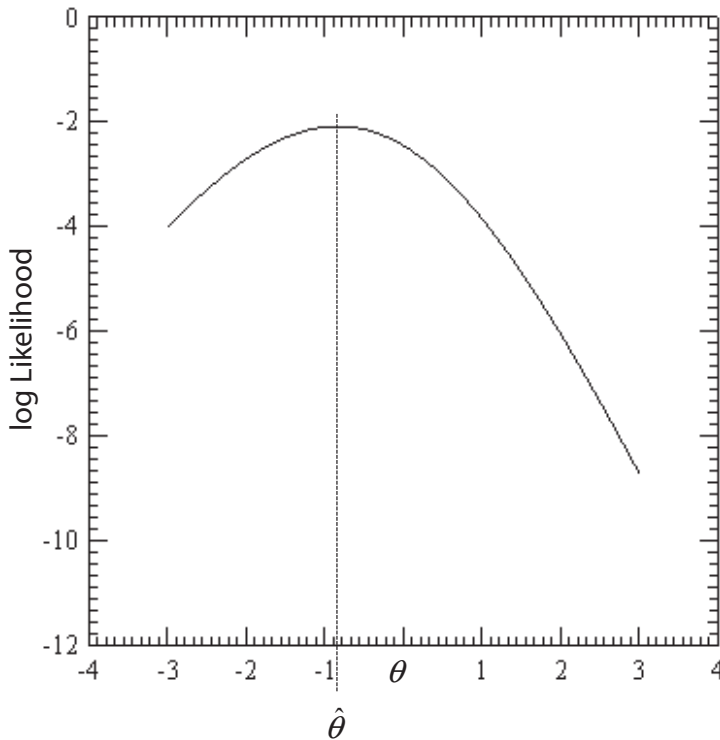


FIGURE A.1. Log likelihood function for the pattern 11000.

ESTIMATING AN INDIVIDUAL'S LOCATION: NEWTON'S METHOD FOR MLE

The empirical MLE approach is inefficient and does not provide us with a standard error of estimate (i.e., an index of the accuracy of our estimate). Its primary advantage is that it can be applied without knowledge of the derivatives of the likelihood function. However, a more sophisticated approach involving the derivatives of the likelihood function provides the sample standard error of estimate. The idea of a likelihood and the maximum likelihood method is presented by Fisher (1971a, 1971b). In the following we first describe the method and then apply it in the IRT context.

To understand this approach, examine the $\ln L$ function shown in Figure A.2. We have drawn a series of lines tangent to the function that vary in their respective slopes (these are the lines labeled (a), (b), and (c)). As we progress from line (a) to line (c) we see that the slope is greatest for line (a) and decreases until for line (c) it is 0. As such, to find the location of the maximum of the function we simply need to determine where the slope of the tangent line is equal to zero. Symbolically,

$$\text{slope} = \frac{\text{change in } Y}{\text{change in } X} = \frac{\Delta Y}{\Delta X} = 0$$

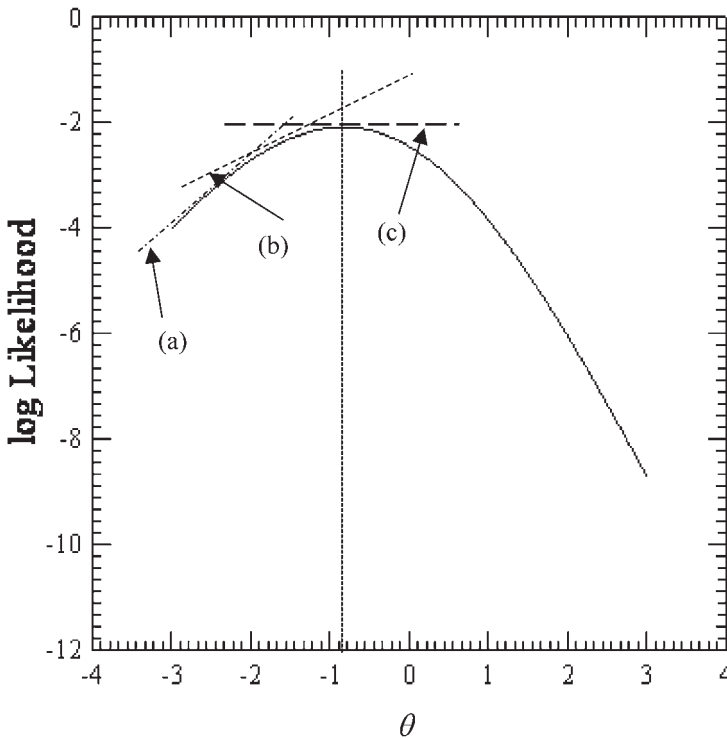


FIGURE A.2. Log likelihood function for the pattern 11000.

or, alternatively, the slope is zero when

$$\Delta Y = 0.$$

To find the maximum of the $\ln L$ function we use an iterative process. The $\hat{\theta}$ at which $\ln L$ is maximized (i.e., the slope = 0) is found by iterating through a series of $\hat{\theta}$ s, with each iteration's $\hat{\theta}$ reflecting a refinement over the previous iteration's $\hat{\theta}$. The process continues to entertain improved $\hat{\theta}$ s until the difference between two successive $\hat{\theta}$ s is considered to be unimportant. This approach to finding the root of an equation is called Newton's method and is a commonly used method for solving equations.^{1,2}

The bisection method described above worked by bracketing a range of θ and searching the bracket for the location of the maximum of $\ln L$. This location is subsequently improved or refined by halving the bracket and re-performing the search. By making the brackets progressively smaller across iterations, one could find the location of the maximum to a desired degree of accuracy. Newton's method works in a similar iterative fashion. Conceptually, Newton's method consists of a series of progressively smaller right triangles (rather than brackets). One of these triangles is shown in Figure A.3; the right triangle is inverted. The hypotenuse of the right triangle in Figure A.3 corresponds to one of the tangent lines shown in Figure A.2 (e.g., line (a)). The horizontal leg of the triangle (the "adjacent leg" to the angle ω) reflects the change in the horizontal axis, ΔX , whereas the vertical leg of the triangle (the "opposite leg" to the angle ω)

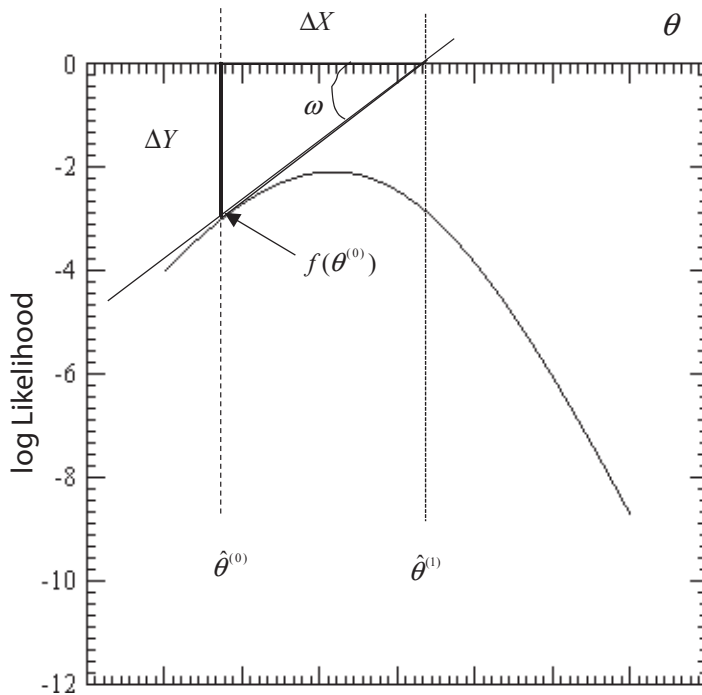


FIGURE A.3. One-step in the Newton method.

reflects the change in the vertical axis, ΔY . Moreover, the “height” of the function at a point (e.g., $\hat{\theta}^{(0)}$) is the length of the opposite leg at that point and is given by $f(\hat{\theta}^{(0)})$. Stated symbolically, $\Delta Y = f(\hat{\theta}^{(0)})$.

To start Newton’s method requires an initial “guesstimate” or provisional estimate ($\hat{\theta}^{(0)}$) of the maximum’s location, and we stop when $\Delta Y = 0$. That is, we stop when the opposite leg has zero length—we have found the maximum of the function. The equality “ $\Delta Y = 0$ ” means that $\Delta Y = 0$ is true to some desired degree of accuracy.

To improve the initial estimate, $\hat{\theta}^{(0)}$, requires knowledge of a few facts:

1. The tangent (tan) of an angle, ω , is equal to the ratio of the opposite leg over the adjacent leg. Thus, $\tan(\omega) = \frac{\text{opposite leg}}{\text{adjacent leg}}$.
2. The first derivative of a function is the slope of a line tangent to the function and is symbolized as $f'(x)$.
3. The line that is tangent to the function in Figure A.3 is the triangle’s hypotenuse with slope $\Delta Y/\Delta X$.
4. The point at which the tangent line crosses the abscissa defines the length of the adjacent leg, ΔX , or $\Delta X = (\hat{\theta}^{(0)} - \hat{\theta}^{(1)})$.

Given Fact #1 and that $\Delta Y =$ “opposite leg” and $\Delta X =$ “adjacent leg,” this means that

$$\tan(\omega) = \frac{\text{opposite leg}}{\text{adjacent leg}} = \frac{\Delta Y}{\Delta X}$$

Therefore, $\tan(\omega)$ is the slope of the line tangent to the function, and given Fact #3, we know that this tangent line is the right triangle’s hypotenuse. Combining Facts #1–#4, recalling that $\Delta Y = f(\hat{\theta}^{(0)})$, and by substitution, one has that

$$\tan(\omega) = \text{slope} = f'(\hat{\theta}^{(0)}) = \frac{\Delta Y}{\Delta X} = \frac{f(\hat{\theta}^{(0)})}{\hat{\theta}^{(0)} - \hat{\theta}^{(1)}} \tag{A.3}$$

Solving for (isolating) $\hat{\theta}^{(1)}$ yields

$$\hat{\theta}^{(1)} = \hat{\theta}^{(0)} - \frac{f(\hat{\theta}^{(0)})}{f'(\hat{\theta}^{(0)})}. \tag{A.4}$$

Stated in words, Equation A.4 says that the value $\hat{\theta}^{(0)}$ may be improved upon by projecting the tangent line from the point $f(\hat{\theta}^{(0)})$ toward the abscissa (i.e., from Fact #2 we know that the tangent line is the first derivative at $\hat{\theta}^{(0)}$). The tangent line’s point of intersection with the abscissa produces a new estimate, $\hat{\theta}^{(1)}$. The change from $\hat{\theta}^{(0)}$ to the new $\hat{\theta}$ is ΔX (i.e., $\Delta X = (\hat{\theta}^{(0)} - \hat{\theta}^{(1)})$). A single application of Equation A.4 to improve $\hat{\theta}^{(0)}$ is one step or iteration. After conducting one iteration we may or may not be at the location of the maximum of the log likelihood function. However, Equation A.4 may be reapplied to “construct” a (we hope smaller) new right triangle using $\hat{\theta}^{(1)}$ in lieu of $\hat{\theta}^{(0)}$ (e.g., the hypotenuse of this new right triangle would be line (b) in Figure A.2).

This idea of conducting multiple iterations may be symbolized by rewriting Equation A.4 as

$$\hat{\theta}^{(t+1)} = \hat{\theta}^{(t)} - \frac{f(\hat{\theta}^{(t)})}{f'(\hat{\theta}^{(t)})} \quad (\text{A.5})$$

where t stands for the t^{th} iteration and T is the maximum number of iterations (i.e., $t = 1 \dots T$). Equation A.5 says that we can improve the t^{th} estimate of the location of the maximum by changing it by an amount equal to $\frac{f(\hat{\theta}^{(t)})}{f'(\hat{\theta}^{(t)})}$. As mentioned above, when $\Delta Y = 0$ (i.e., $f(\hat{\theta}^{(t)}) = 0$), then the location of the maximum has been found and is our $\hat{\theta}$. Stated another way, we have found the maximum when the step size $\frac{f(\hat{\theta}^{(t)})}{f'(\hat{\theta}^{(t)})} = 0$ or, alternatively, $\hat{\theta}^{(t+1)} = \hat{\theta}^{(t)}$. Therefore, after calculating $\hat{\theta}^{(t+1)}$ we check to see if $\hat{\theta}^{(t+1)} = \hat{\theta}^{(t)}$. If the answer is “yes,” then the iterations stop because we have found the location of the maximum; the location is $\hat{\theta}^{(t+1)}$. However, if the answer is “no,” then we can improve the current estimate by calculating a new step size. In effect, we refine our estimate of the location of the maximum by stepping along the log likelihood function in steps of size $\frac{f(\hat{\theta}^{(0)})}{f'(\hat{\theta}^{(0)})}$. If the function is well behaved (e.g., it is not flat), then the step size becomes progressively smaller as the iterations proceed. The signs of $f(\hat{\theta}^{(t)})$ and $f'(x)$ do not have to be the same.³

Because of how decimal values are represented on a computer it is difficult to test for an equality (e.g., $\hat{\theta}^{(t+1)} = \hat{\theta}^{(t)}$ or $f(\hat{\theta}^{(t)}) = 0$). Therefore, the difference between successive $\hat{\theta}$ s (i.e., $(\hat{\theta}^{(t+1)} - \hat{\theta}^{(t)})$) is checked to see if the $\hat{\theta}$ s are “indistinguishable” from one another. If they are indistinguishable, then the process is said to have *converged* and we have determined the location of the maximum. What is considered indistinguishable (i.e., what defines a “zero” change) is given by the *convergence criterion* (e.g., $\Xi = 0.001$). Therefore, when $(\hat{\theta}^{(t+1)} - \hat{\theta}^{(t)}) < \Xi$ is true, then we have a *converged solution*, $\hat{\theta}^{(t+1)}$ is the estimate of location of the maximum, and convergence is achieved in $t+1$ iterations. However, when $(\hat{\theta}^{(t+1)} - \hat{\theta}^{(t)}) > \Xi$, then we do not have a converged solution and another iteration is performed. How many iterations one performs depends on the maximum number of iterations, T (e.g., $T = 25$). As a result, there are two criteria that must be met before another iteration is performed (i.e., $(\hat{\theta}^{(t+1)} - \hat{\theta}^{(t)}) > \Xi$ and $t < T$). If $(\hat{\theta}^{(t+1)} - \hat{\theta}^{(t)}) > \Xi$ and if t equals T , then the estimation process stops even though it has not converged, and we have a *nonconverged solution*.

In applying Newton’s method to IRT the function of interest is the log likelihood function. That is, the function $f(\hat{\theta}^{(t)})$ that is set to 0 is the first derivative of the $\ln L$ function, $f(\hat{\theta}^{(t)}) = \frac{\partial}{\partial \theta} \ln L(\underline{\mathbf{x}} | \theta)$. Therefore, the maximum of the log likelihood function is found (if it exists) when

$$\frac{\partial}{\partial \theta} \ln L(\underline{\mathbf{x}} | \theta) = 0 \quad (\text{A.6})$$

is true.

If we look at the form of the step size in Equation A.5, we see that it has the form of a ratio of a function to its derivative (i.e., $\frac{f(\hat{\theta}^{(t)})}{f'(\hat{\theta}^{(t)})}$). In the IRT case, $f(\hat{\theta}^{(t)})$ is the first derivative of $\ln L$. As a result, the step size is equal to the first derivative over its derivative (i.e., the second derivative). Therefore, with respect to θ , the step size is

$$\frac{f(\hat{\theta}^{(t)})}{f'(\hat{\theta}^{(t)})} = \frac{\frac{\partial}{\partial \theta} \ln L(\underline{\mathbf{x}} | \theta)}{\frac{\partial^2}{\partial \theta^2} \ln L(\underline{\mathbf{x}} | \theta)} \quad (\text{A.7})$$

By substituting Equation A.7 into Equation A.5, our formula for improving our estimate of the location of the maximum of person i 's log likelihood function becomes

$$\hat{\theta}_i^{(t+1)} = \hat{\theta}_i^{(t)} - \frac{\frac{\partial}{\partial \theta} \ln L(\underline{\mathbf{x}} | \theta_i^t)}{\frac{\partial^2}{\partial \theta^2} \ln L(\underline{\mathbf{x}} | \theta_i^t)}. \quad (\text{A.8})$$

The equations for $\frac{\partial}{\partial \theta} \ln L(\underline{\mathbf{x}} | \theta_i^t)$ and $\frac{\partial^2}{\partial \theta^2} \ln L(\underline{\mathbf{x}} | \theta_i^t)$ vary from model to model. The simplest forms of these belong to the Rasch model. Therefore, as an example of applying Newton's method to IRT, the Rasch model is used.

For the Rasch model the t^{th} iteration of the derivatives are (cf. Hambleton & Swaminathan, 1985; Wright & Stone, 1979)

$$\frac{\partial}{\partial \theta} \ln L(\underline{\mathbf{x}} | \theta_i^t) = X_i - \sum_{j=1}^L p_{ij}^{(t)} \quad (\text{A.9})$$

and

$$\frac{\partial^2}{\partial \theta^2} \ln L(\underline{\mathbf{x}} | \theta_i^t) = -\sum_{j=1}^L p_{ij}^{(t)}(1 - p_{ij}^{(t)}), \quad (\text{A.10})$$

where p_{ij} is the probability of a response of 1 on item j by person i according to Equation 2.2. By substitution of these identities into Equation A.8 we have

$$\hat{\theta}_i^{(t+1)} = \hat{\theta}_i^{(t)} - \frac{X_i - \sum_{j=1}^L p_{ij}^{(t)}}{-\sum_{j=1}^L p_{ij}^{(t)}(1 - p_{ij}^{(t)})}. \quad (\text{A.11})$$

Equation A.11 is applied until we have a converged solution or we reach the maximum number of iterations with $\hat{\theta}_i^{(t+1)}$ value as our estimate of person i 's location, $\hat{\theta}_i$. If our

solution converges, then our $\hat{\theta}$ is the location of the maximum of the log likelihood function (i.e., $\hat{\theta}$ maximizes the likelihood of obtaining the response pattern).

Focusing on the numerator of the step size, $X_i - \sum_{j=1}^L p_{ij}^{(t)}$, we see that it has the form of an observed score, X_i , minus the expected (trait) score ($\sum_{j=1}^L p_{ij}^{(t)}$); the expected score is based on the provisional estimate of the person's location ($\hat{\theta}_i^{(0)}$), the item parameters, and the model. In effect, the estimation tries to minimize the difference between what one would expect or predict on the basis of the model and what is observed. We can also see from the numerator that there is no information about the pattern of 0s and 1s in person i 's response vector. The estimation of θ is driven solely by trying to modify θ to make $\sum_{j=1}^L p_{ij}^{(t)}$ as close a match as possible to the observed score, X_i . The denominator of the step size is the sum of the predicted item variances. In the foregoing it is assumed that the δ_j s are known. (Given that our interpretation of the numerator of Equation A.11 is similar to that of the numerator of the chi-square statistic, it is not surprising that there is an alternative estimation method based on the chi-square statistic (Berkson, 1944, 1955; Baker, 1991).)

As an example, assume that we are interested in estimating the θ that has the highest likelihood of producing the pattern 11000 (Table A.1). Moreover, assume that our item locations are $\delta_1 = -1.9000$, $\delta_2 = -0.6000$, $\delta_3 = -0.2500$, $\delta_4 = 0.3000$, and $\delta_5 = 0.4500$. Our convergence criterion is 0.0001. To start our estimation we need an initial guesstimate as to where the function has its maximum. There are various ways of providing this guesstimate. For example, we could assume that the individuals who produce this pattern 11000 are of average proficiency, and therefore the initial guesstimate would be $\hat{\theta}^{(0)} = 0.0$. Alternatively, we can take test performance into account in making our guesstimate. For instance, we can convert X into its corresponding z -score or it may be transformed into a logit correct by $\ln(X/(L - X))$ (Wright & Stone, 1979).

Using this latter approach our guesstimate for $X = 2$ would be $\hat{\theta}^{(0)} = \ln(2/(5 - 2)) = -0.40546510811$. Given this $\hat{\theta}^{(0)}$ we calculate the first and second derivatives (columns

TABLE A.1. MLE Iteration History for Solving $\frac{\partial}{\partial \theta} \ln L(11000 | \theta) = 0$

Iteration	$\hat{\theta}^{(t)}$	$\frac{\partial}{\partial \theta} \ln L(\underline{x} \theta^{(t)})$	$\frac{\partial^2}{\partial \theta^2} \ln L(\underline{x} \theta^{(t)})$	Step size	$\hat{\theta}^{(t+1)}$
1	-0.40546510811	-0.45534004562	-1.07642581519	0.42301107907	-0.82847618718
2	-0.82847618718	-0.00955017713	-1.02205392943	0.00934410294	-0.83782029012
3	-0.83782029012	-0.00000834401	-1.02026427006	0.00000817828	-0.83782846840
4	-0.83782846840	-0.00000000001	-1.02026269393	0.00000000001	-0.83782846841

3 and 4) as well as their ratio (column 5, labeled “Step size,” Equation A.7). As indicated in Equation A.11 this step size is subtracted from $\hat{\theta}^{(0)}$ to produce an improved estimate, $\hat{\theta}^{(1)}$, shown in column 6 (e.g., $\hat{\theta}^{(1)} = -0.82847618718$). In iteration 2 this $\hat{\theta}^{(1)}$ is improved upon by recalculating the values of the first and second derivatives, forming a new step size, and producing a new improved estimate, $\hat{\theta}^{(2)} = -0.83782029012$. These steps are repeated for the remaining iterations. Because iteration 3’s step size of 0.00000817828 is less than our convergence criterion, we have a converged solution and the estimation process stops. The $\hat{\theta}$ after the third iteration, $\hat{\theta}^{(3)} = -0.83782846840$, would be our final estimate of the location of the maximum of $\ln L(\underline{x} = 11000)$; that is, $\hat{\theta} = -0.8378$.⁴ For a pedagogical reason we perform a fourth iteration to show how little change there is from iteration 3’s results. Figure A.4 contains a graphical representation of the steps shown in Table A.1.

As mentioned above, one advantage of the Newton method over the empirical MLE is the ability to obtain the sample standard error of estimate. The standard error of $\hat{\theta}$ is

$$s_e(\hat{\theta}) = \frac{1}{\sqrt{\sum_{j=1}^L \alpha^2 p_j (1 - p_j)}} \tag{A.12}$$

where for the Rasch model α equals 1 and p_j is conditional on $\hat{\theta}$. To calculate p_j one uses the final $\hat{\theta}$ and the item parameters. As can be seen from Equation A.12, the magnitude of the standard error of $\hat{\theta}$ is influenced, in part, by the instrument’s length. For this example the standard error for $\hat{\theta} = -0.8378$ is 0.9900. This value of almost 1

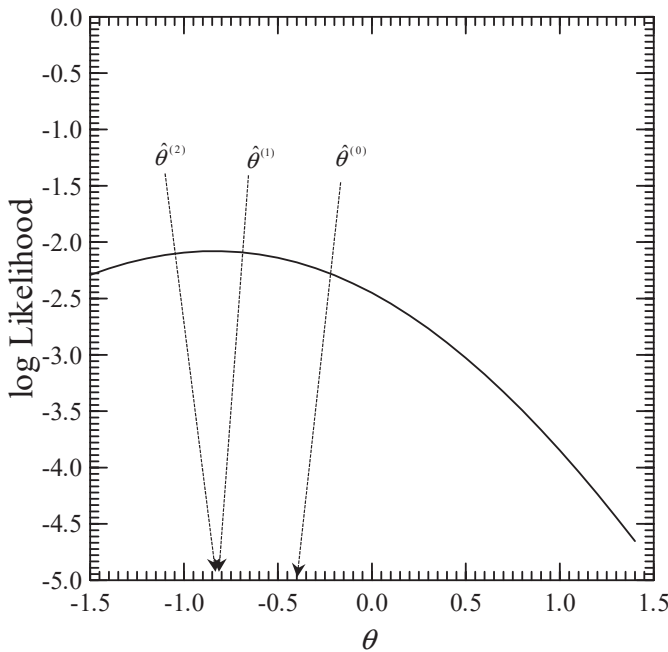


FIGURE A.4. Graphical representation of MLE for the pattern 11000.

logit may be considered on the large side. Its magnitude is due, in part, to the use of five items, the items' locations, and the location of the person's estimate with respect to the items' location (i.e., this $\hat{\theta}$ falls in a gap between δ_1 and δ_2).

In general, a standard error of estimate consists of two components. The first component is the *bias* in the estimate (i.e., the signed difference between the parameter and its estimate), whereas the second component is the *mean squared error* (MSE) in the estimate (i.e., the unsigned squared difference between the parameter and its estimate); the root mean squared error (RMSE) is the square root of the MSE. The relationship between SEE, MSE, and bias is

$$s_e = \sqrt{MSE - bias^2} \quad (\text{A.13})$$

R FUNCTION FOR MLE OF θ WITH THE RASCH MODEL

As a pedagogical tool we provide an R function that will estimate θ given item location parameters and a person response vector \underline{x} . In addition to estimating θ , it graphically displays the corresponding $\ln L$ function. Table A.2 shows the code. To use the function we provide values for \underline{x} and delta (i.e., δ_j s). For example,

```
delta=seq(-2,2,1)           # 5 items located at -2, -1, 0, 1, 2
x = c(1,1,0,0,0)
Raschthetahat_SE(x,delta)
```

For this example response vector our $\hat{\theta} = -0.591$ with a $s_e(\hat{\theta}) = 1.096$. (Note: these item locations are not the same as those used in Table A.1 example.) If we repeatedly call `Raschthetahat_SE` passing to it one of the response vectors associated with $X = 1$, followed by one of the response vectors corresponding to a $X = 2$, and so on up to $X = 4$ we see the $\ln L$ starting on the left side of the continuum and progressing towards the right side with the corresponding maximum locations moving up the continuum.

REVISITING ZERO VARIANCE BINARY RESPONSE PATTERNS

Figure 2.8 shows the log likelihood for a perfect response pattern, 11111. This pattern as well as the pattern 00000 have zero response variability. From Figure 2.8 we see the function's trajectory becoming asymptotic as θ increases and, for all intents, the function becomes relatively flat. Therefore, Equation A.8's step size does not decrease and the $\hat{\theta}$ will "drift off" toward infinity. Mathematically, the numerator of the step size in Equation A.11 equals 0 only when $p_{ij} = 1$ for all items (i.e., $X_i = X_i - \sum_{j=1}^L p_{ij}^{(i)}$). However, p_{ij} equals 1 only for an infinitely large $\hat{\theta}$. Therefore, the use of Newton's method for estimating θ does not provide finite estimates for zero or perfect scores. (The $\ln L$ for $X = 0$ would be the mirror image of the $\ln L$ presented in Figure 2.8.)

TABLE A.2. R Function for MLE of θ Based on the Rasch Model

```

Raschthetahat_SE=function(x,delta) {
# arguments: x - person response vector
#             (needed only for plot.  without plot pass only obs scr X & delete
#             calculation of the observed score within the funciton)
#             delta - vector containing as many item locations as there are
#             responses in x
# Call: Raschthetahat_SE(x,delta)

# for plot: set abscissa to have 9 tick marks, ordinate to have 5, &
#             character labels to be 2
  par(lab=c(9,5,2))

# general initializations for Rasch person estimation
  maxit=20      # maximum number of iterations allowed; in the text this is 'T'
  ccrit=0.001   # convergence criterion
  L = length(x) # nitems

  X = sum(x)    # calc observed score

# To mimic the typical MLE implementation uncomment the follow if statement.
# Deal w/ zero variance response vectors; change X = 0 to be X = 0.5 &
# change X= L to be X = L -0.5.  The resulting person location estimates are
# not MLEs, but pseudo MLEs.
#   if(X==0) {
#     X=0.5
#   } else {
#     if (X==L) {
#       X=L-0.5
#     }
#   }

# t_est is theta hat; "log" in R is the "ln" function
  t_est=log(X/(L-X))      # initial value for t_est (theta hat null)

# estimation (only need X, not x)
  it = 1
  converged = FALSE

  while ((it <= maxit) & (! converged)) {
    expctdX = 0.0
    expctdVar = 0.0
    for (j in 1:L) {
      p = 1/(1+exp(-1.0*(t_est-delta[j])))
      expctdX = expctdX + p # Equation A.9
      expctdVar = expctdVar + (1-p)*p      # Essentially Equation A.10
    } # for j loop

    step=(X - expctdX)/-expctdVar

    t_est=t_est - step      # Equation A.5

# To see the iteration table uncomment the following two lines
#   if (it==1) {print('iteration pre_t      1st      2nd      step
#                     post_t') }
#   cat(sprintf("%12.d %10.5f %10.5f %10.5f %10.5f %10.5f",it,(t_est+step), (X -
#                     expctdX), (-expctdVar),step,t_est),"\n")

```

(continued)

TABLE A.2. (continued)

```

converged = abs(step) < ccrit # is stepsize < convergence criterion?

# if we have converged or we have reached the maximum of iterations, calc Std Error
if (converged | it == maxit) {
  se = 1/sqrt(expctdVar) # Equation A.12
}

it = it + 1
} # while it & step loop

# produce lnL plot
mintheta=-4.0; maxtheta=4.0; incr=0.1 # initializations
nvals=(abs(mintheta)+abs(maxtheta))/incr+1
t = seq(mintheta, maxtheta, incr)
lnL = rep(0.0, nvals)

for (k in 1:nvals) {
  lnLike = 0.0
  for (j in 1:L) {
    p = 1/(1+exp(-1.0*(t[k]-delta[j])))
    lnLike = lnLike + x[j]*log(p) + (1-x[j])*log(1 - p)
  } # for j loop
  lnL[k] = lnLike
} # for k loop

cat(plot(t,lnL,xlab="theta",type="l",ylab="lnL",xlim=c(mintheta,maxtheta)), "\n")

cat(paste("theta est",t_est, "\n"))
cat(paste("SEE",se, "\n"))
cat(paste("Converged ",converged, "\n"))

Raschthetahat_SE =c(t_est,se)
}

```

In a software MLE implementation one will obtain a faux $\hat{\theta}$ for zero variance response vector. This faux $\hat{\theta}$ is not the location of the maximum, but an artifact of a convergence criterion that is “too liberal” relative to the lnL’s asymptotic nature. This is easily demonstrated using the `Raschthetahat_SE` function; `maxit` should be set to a large value. By increasing the desired precision of $\hat{\theta}$ (e.g., changing `ccrit` to be 0.00001, then 0.000001, etc.) one finds the faux $\hat{\theta}$ drifting towards extreme values.

NOTES

1. Both the empirical MLE and Newton’s method for MLE are predicated on the assumption that the likelihood function’s shape is determined by some unknown parameter, θ .
2. This method is also referred to as Newton–Raphson. Newton’s method was developed about 1669 and, apparently, Raphson independently developed a simplified version of Newton’s method in 1690 (Gautschi, 1997). Therefore, this method is typically referred to as Newton–Raphson, although some (e.g., Gerald & Wheatley, 1984; Gautschi, 1997) refer to it as Newton’s

method. However, both Newton's and Raphson's algorithms were algebraic and did not involve derivatives (Gautschi, 1997). Simpson in 1740 (cited in Gautschi, 1997) introduced the calculus description to Newton's method, and the modern version of Newton's approach seems to have appeared first in a paper by Fourier in 1831 (Gautschi, 1997).

3. The slope is equal to zero whenever a function has a maximum or a minimum. The sign of the second derivative, $f''(x) = \frac{\partial^2}{\partial \theta^2} \ln L(\underline{x} | \theta)$, indicates whether a minimum or a maximum at θ has been obtained. Specifically, if $f''(x) < 0.0$, then it is a maximum. However, a given function may have multiple maxima/minima as well as local maxima/minima. Local and multiple maxima/minima arise when the function has multiple bends rather than a single bend as shown in Figure A.2. A local maximum (or minimum) occurs when a location is found at which the slope of the function is 0, but this location does not correspond to the highest point on the function. For example, imagine a function that is increasing, reaches a crest, bends downward into a valley, and then bends upward out of the valley to a second crest that is higher than the first crest. The first crest would be a local maximum and the second would be the function's true maximum; the floor of the valley would be a minimum. Evidence about whether the solution is at a local maximum/minimum rather than at a true maximum/minimum may be obtained by using different initial starting estimates. If the various solutions produce the same estimate, then most likely a true maximum has been found.

4. If the *standard score* of $X = 2$ had been used as the $\hat{\theta}^{(0)}$, the results would still have converged in three iterations and $\hat{\theta}^{(3)} = -0.83782846841$.

Appendix B

Maximum Likelihood Estimation of Item Locations

In Appendix A we discuss the logic and mathematics of Newton's method for locating the maximum of the $\ln L$ as well as demonstrate Newton's method for estimating a person's location. In this appendix we assume the reader is familiar with Newton's method and show its use to estimate an item's location, δ . Analogous to what is done in the estimation of person locations in Appendix A, we assume that the persons' θ s are known.

In estimating a person's location with the IPL model, the data in Table 2.1 are reduced to six rows ($X = 0, 1, \dots, 5$) and only the row totals for four observed scores ($X = 1, 2, 3, 4$) provide information for estimating θ using MLE. Similarly, in estimating an item's location it is not the pattern of 1s and 0s on the item, but the item score (i.e., the item total), q_j , for item j that provides all the information needed for estimating its δ (cf. Rasch, 1980). (The pattern of responses to an item is also ignored in calculating the traditional item difficulty index [i.e., an item's P-value, P_j].) Accordingly, the item score q_j embodies all the information for estimating the item's location and is a sufficient statistic for estimating δ_j .

Conceptually, the likelihood function for an item specifies the likelihood of observing a particular q_j , given the possible values of δ . For instance, how likely is it that 13,319 individuals got item 1 correct if the item is located at -3.0 , or if it is located at -2.9 , or at 3.0 ? As is the case with estimating person locations and zero variance response vectors, if $q_j = 0$ or $q_j = N$, then the likelihood function does not have a maximum and there is no finite estimate of the item's δ . Stated another way, if $q_j = 0$, then $\delta_j = \infty$, and if $q_j = N$, then $\delta_j = -\infty$. In principle, the likelihood function for an item would be obtained in a way similar to the way it is obtained for estimating a person's location. However, a logarithmic transformation of L is typically performed to produce a log likelihood function, $\ln L$ (e.g., see Equation A.2).

The application of Newton's method in Appendix A produced an equation that allowed one to successively refine the location estimate of the maximum of the $\ln L$ (see Equation A.8). Applying Newton's method to obtain the $\hat{\delta}$ involves making the appropriate substitutions for the first and second derivatives of the log likelihood function

with respect to δ into Equation A.8. These derivatives may be found in the literature (e.g., Baker & Kim, 2004; Hambleton & Swaminathan, 1985; Wright & Stone, 1979). Upon making these substitutions into Equation A.8 we have an equation to obtain an improved $\hat{\delta}_j$ for the j th item at the $(t + 1)$ iteration

$$\hat{\delta}_j^{(t+1)} = \hat{\delta}_j^{(t)} - \frac{f(\hat{\delta}_j^{(t)})}{f'(\hat{\delta}_j^{(t)})} = \hat{\delta}_j^{(t)} - \frac{\frac{\partial}{\partial \delta} \ln L(\underline{x} | \delta_j^t)}{\frac{\partial^2}{\partial \delta^2} \ln L(\underline{x} | \delta_j^t)} \quad (\text{B.1})$$

Because the derivatives in Equation B.1 are with respect to δ , they are different from those seen in Appendix A, Equations A.9 and A.10. Specifically, we have

$$\frac{\partial}{\partial \delta} \ln L(\underline{x} | \delta_j^t) = -q_j + \sum_{i=1}^N p_{ij}^{(t)}$$

and

$$\frac{\partial^2}{\partial \delta^2} \ln L(\underline{x} | \delta_j^t) = -\sum_{i=1}^N p_{ij}^{(t)}(1 - p_{ij}^{(t)}).$$

Therefore, upon substitution of these derivatives into Equation B.1 one obtains

$$\hat{\delta}_j^{(t+1)} = \hat{\delta}_j^{(t)} - \frac{\left(\sum_{i=1}^N p_{ij}^{(t)}\right) - q_j}{-\sum_{i=1}^N p_{ij}^{(t)}(1 - p_{ij}^{(t)})} \quad (\text{B.2})$$

As is the case with estimating person location via MLE, the numerator reflects a difference between the observed item score (the number of responses of 1 on the item) and the expected/predicted score for the item based on the provisional $\hat{\delta}_j$, the known θ s, and the model. We see that the numerator does not contain any information about the pattern of 0s and 1s in item j 's response vector. As a result, the estimation of δ is driven solely by trying to modify $\hat{\delta}$ to make $\sum p_{ij}$ as close a match as possible to the item score q_j . In obtaining a solution one seeks to minimize this discrepancy by iteratively improving the $\hat{\delta}_j$ s until $(\hat{\delta}_j^{(t+1)} - \hat{\delta}_j^{(t)}) < \Xi$, where Ξ is the convergence criterion. In addition to Ξ , one typically has a maximum number of iterations that can be executed as a stop-gap criterion. Therefore, our estimation continues until either Ξ is satisfied or we reach the maximum number of iterations.

Strictly speaking, because δ is unknown it is not possible to calculate p_{ij} . However, the t^{th} provisional estimate of δ is treated as known in order to calculate an estimate of p_{ij} in Equation B.2. In addition, for purposes of estimation, and because all persons with the same observed score have the same θ , we can approximate $\sum p_{ij}$ by

$$\sum_{X=1}^{L-1} n_X p_{Xj}$$

where n_X is the number of persons obtaining an observed score of X . (Because $X = 0$ and $X = L$ do not produce finite estimates they are omitted from the summation and

the sum runs from 1 to $L - 1$, not from 0 to L .) Therefore, in implementations of the Newton method $\sum p_{ij}$ is replaced by $\sum_{X=1}^{L-1} n_X p_{Xj}$ and $\sum_{i=1}^N p_{ij}^{(t)}(1 - p_{ij}^{(t)})$ is replaced by $\sum_{X=1}^{L-1} n_X p_{Xj}^{(t)}(1 - p_{Xj}^{(t)})$.

Newton's method converges more quickly if one starts in the neighborhood of the final solution. A starting location can be obtained by transforming the item scores to their corresponding standard scores or by using a modified logit incorrect (Wright & Stone, 1979)

$$\tilde{\delta}_j^{(0)} = \ln\left(\frac{N - q_j}{q_j}\right) - \sum \ln\left(\frac{N - q_j}{q_j}\right) / L \quad (\text{B.3})$$

The first term is essentially a logit incorrect (i.e., the number of responses of 0 over the number of responses of 1), whereas the second term is its average across items. Therefore, Equation B.3 provides a "centered" starting value. These provisional estimates have a mean of 0.

We use the first item on our example's mathematics test (Chapter 2) to demonstrate the MLE of an item location. As mentioned above, the person locations are assumed to be known. For this example, we obtain our provisional person location estimates by using $\ln(X/(L - X))$; see Appendix A. Therefore, for persons with an $X = 1$ our θ_1 is -1.38629 , for $X = 2$ our $\theta_2 = -0.40547$, for $X = 3$ our $\theta_3 = 0.40547$, and for $X = 4$ our $\theta_4 = 1.38629$. Table B.1 contains the MLE iteration history. The convergence criterion is set to 0.0001 (i.e., $\Xi = 0.0001$). We see that convergence is achieved on the fourth iteration. Our MLE estimate of item 1's location is $\hat{\delta}_1 = -2.044$. The accuracy of this estimate can be ascertained via its standard error, $s_e(\hat{\delta})$

$$s_e(\hat{\delta}) = \frac{1}{\sqrt{\sum_{X=1}^{L-1} n_X p_{Xj}(1 - p_{Xj})}} \quad (\text{B.4})$$

By using the $\hat{\delta}_1$ obtained from the last iteration in Equation B.4, our sample standard error for $\hat{\delta}_1$, $s_e(\hat{\delta}_1)$, is 0.0242. (As is the case for estimation of person locations one can

TABLE B.1. MLE Iteration History for Solving $\ln L$

Iteration	$\hat{\delta}_1^{(r)}$	$\frac{\partial}{\partial \delta} \ln L(\underline{x} \delta^{(r)})$	$\frac{\partial^2}{\partial \delta^2} \ln L(\underline{x} \delta_1^{(r)})$	Step size	$\hat{\delta}_1^{(r+1)}$
1	-0.827790295	-2782.277903	-2816.302721	0.987918622	-1.815708917
2	-1.815708917	-414.1668931	-1928.518792	0.214759065	-2.030467982
3	-2.030467982	-22.95006925	-1715.301008	0.013379616	-2.043847599
4	-2.043847599	-0.087983322	-1702.153013	5.16894E-05	-2.043899288

TABLE B.2. MLE $\hat{\delta}_s$ and Corresponding SEEs for the Five-Item Instrument			
Item	$\hat{\delta}$	$s_e(\hat{\delta})$	Number of iterations
Uncentered			
1	-2.0438	0.0242	4
2	-0.1648	0.0179	3
3	0.3208	0.0178	3
4	1.2477	0.0195	4
5	1.5579	0.0206	4
Centered			
1	-2.2274	0.0242	4
2	-0.3484	0.0179	3
3	0.1373	0.0178	3
4	1.0641	0.0195	4
5	1.3744	0.0206	4

talk about the amount of information a sample provides for estimating an item's location by taking the square of the reciprocal of Equation B.4.)

Table B.2 shows the results of applying the above procedure to the remaining items on the instrument with the values also being mean centered to address indeterminacy of scale. As can be seen, our items are located throughout the continuum ranging from roughly 2 logits below 0 to 1.4 logits above 0. Our standard errors are on the order of two one-hundredths or less indicating reasonably accurate item location estimates. We should note that these item location estimates should not be interpreted in an absolute sense. That is, if we estimate these item locations with a different sample of examinees, we would most likely obtain a different set of estimates that, assuming model–data fit, would be highly linearly related to the estimates in Table B.2. This issue is discussed in Chapter 3 in the section entitled “Indeterminacy of Parameter Estimates.”

R FUNCTION FOR MLE OF δ WITH THE RASCH MODEL

As a pedagogical tool we provide an R function that will estimate δ . In addition to estimating δ , it graphically displays the corresponding $\ln L$ function. Table B.3 shows the code. To use the function we provide the data file and the item of interest. For example, for item 2 we would call the function

```
RaschItem_SE(mathdata, 2)
```

TABLE B.3. R Function for MLE of δ Based on the Rasch Model

```

RaschItem_SE=function(x,item) {
# arguments: x - response data matrix
#             (needed only for plot.  without plot pass only item scr q & delete
#             calculation of the item score within the function)
#             item - item of interest

# Call: RaschItem_SE(x,item)

# for plot: set abscissa to have 9 tick marks, ordinate to have 5, &
#           character labels to be 3
par(lab=c(9,5,3))

# general initializations for Rasch person estimation
maxit=20L      # maxiterations as an integer
ccrit=0.001    # convergence criterion
N = length(x[,item]) # n persons
L = length(x[1,])  # Length: # of items
L_1 = L -1

q = as.integer(colSums(x)) # calc item score
X=rowSums(x) # calc observed scores, X

nX = as.integer(table(X)) # frequency of each obs. score
adjustment = nX[1]+nX[L+1] # remove zero response vectors

t_est=rep(-99.9, L_1)      # determine provisional person estimates
for (i in 1:L_1) {
  t_est[i]=log(i/(L-i))
}

Nadj= N - adjustment
delta_est=0.0             # determine initial value for delta_est; Equation B.3
for (j in 1:L) {
  q[j] = q[j] - adjustment
  delta_est = delta_est + log((Nadj-q[j])/q[j])
}
delta_est=log((N-q[item])/q[item]) - delta_est/L

# estimation
it = 1
converged = FALSE

while ((it <= maxit) & (! converged) ) {
  expctdq = 0.0
  expctdVar = 0.0
  for (i in 1:L_1) {
    # offset due to X being 0-based & indexing is 1-based
    p = 1/(1+exp(-1.0*(t_est[i]- delta_est)))
    expctdq = expctdq + nX[i+1]*p          # essentially numerator of Equation B.2
    expctdVar = expctdVar + nX[i+1]*p*(1-p) # essentially denominator of Eq B.2
  } # for j loop

  expctdq = expctdq - q[item]

  step=expctdq/-expctdVar

```

(continued)

TABLE B.3. (continued)

```

delta_est =delta_est - step      # Equation B.2
  expctdVar = expctdVar + nX[i+1]*p*(1-p)  # essentially denominator of Eq B.2
} # for j loop

expctdq = expctdq - q[item]

step=expctdq/-expctdVar

delta_est =delta_est - step      # Equation B.2
# To see the iteration table uncomment the following two lines
# if (it==1) {print('iteration  pre_d      1st      2nd      step
                post_d' ) }
# cat(sprintf("%12.d  %10.5f %12.5f %12.5f %10.5f %10.5f",it, (delta_est+step),
                expctdq, (-expctdVar),step,delta_est),"\n")

converged = abs(step) < ccrit

if (converged | it == maxit) {
  se = 1/sqrt(expctdVar)      } # essentially Equation B.4

it = it + 1
} # while it & step loop

# produce lnL plot
maxdelta = 4.0; mindelta= -4.0; incr = 0.1      # initializations
nvals=(abs(maxdelta)+abs(mindelta))/incr+1
lnL = rep(0.0,nvals)
delta = seq(mindelta,maxdelta,incr)

t_est=rep(-99.9, Nadj)
u=rep(-99.9,Nadj)

k = 1
for (i in 1:N) {
  if((X[i]>0) & (X[i] < L)) {      # remove zero variance response vectors
    t_est[k]=log(X[i]/(L-X[i]))
    u[k]=x[i,item]
    k = k + 1
  } # if
} # for i

for (k in 1:nvals) {
  lnLike = 0.0

  for (i in 1:Nadj) {
    p = 1/(1+exp(-1.0*(t_est[i]-delta[k])))
    lnLike = lnLike + u[i]*log(p) + (1-u[i])*log(1 - p)
  } # for i loop

  lnL[k] = lnLike
} # for k loop

```

(continued)

TABLE B.3. (continued)

```

cat(plot(delta,lnL,main=paste("item ",item), xlab="delta", type="l", ylab="lnL",
    xlim=c(mindelta,maxdelta)),"\n")

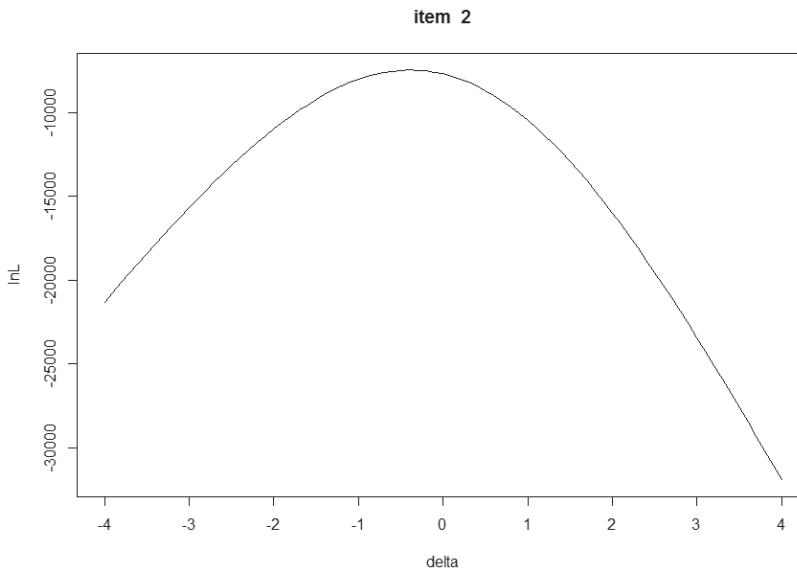
cat(paste("delta est",delta_est,"\n"))
cat(paste("SEE",se,"\n"))
cat(paste("Converged ",converged,"\n"))

RaschItem_SE=c(delta_est,se)
}

```

The result of our call yields $\hat{\delta}_2 = -2.0438$ ($s_e(\hat{\delta}_2) = 0.0242$) with its $\ln L$ function shown in Figure B.1. If we repeatedly call `RaschItem_SE` passing to it each successive item we obtain the uncentered estimates shown in Table B.2 (top panel).

The processes shown in Appendices A and B form the foundation of the joint maximum likelihood estimation (JMLE) algorithm. In the JMLE context (Chapter 3) the item locations would be estimated before persons. The resulting centered $\hat{\delta}$ s would be used to estimate the $\hat{\theta}$ s (see Appendix A) and cycle 1 ends. In the subsequent cycle the $\hat{\theta}$ s would be used in lieu of the provisional ones to re-estimate the $\hat{\delta}$ s. These improved $\hat{\delta}$ s would then be used to re-estimate the $\hat{\theta}$ s and cycle 2 ends. Cycle 3 would use the improved $\hat{\theta}$ s to re-estimate the $\hat{\delta}$ s and the improved $\hat{\delta}$ s would be used to re-estimate the $\hat{\theta}$ s. These cycles continue until the change between successive cycles' $\hat{\delta}$ s and successive cycles' $\hat{\theta}$ s are less than the convergence criterion.

**FIGURE B.1.** $\ln L$ for item 2.

Appendix C

The Normal Ogive Models

CONCEPTUAL DEVELOPMENT OF THE NORMAL OGIVE MODEL

Our conceptual development of the IRT normal ogive model begins with a discussion of the relationship between the observed 0/1 responses and the variable being measured. In the current context, the latent variable of interest (e.g., neuroticism, narcissism, mathematics proficiency) is measured by asking a series of questions. The responses to these questions are transformed to be a 0 or a 1. For instance, a person may be asked, “Given $X = 3 + 5$, what is the value of X ?” In this case, the individual’s response is coded as 1 if the response is 8, otherwise it is coded to 0.

One may ask, “How does the 0/1 ‘response’ on an item relate to the latent variable being measured?” To answer this question assume that a continuous latent variable, Ω_j , determines an individual’s response to an item j . Large values of this item latent variable Ω_j indicate a greater tendency to produce a response (x_j) of 1 than do smaller values. This continuous variable is dichotomized at some point, τ_j , such that at and above this point the continuous latent variable’s values are recoded as a 1, and below which they are recoded as a 0. For example, in Figure C.1 person 1 is located (μ_{1j}) beyond item j ’s cutpoint (threshold, τ_j). Therefore, the shaded area under the function beyond the cutpoint is the probability (π_{1j}) of a response of 1 to item j by person 1. The unshaded portion gives the probability of a response of 0 to this item by this person. This is the mechanism by which the observed 0/1 responses arise. Note that in contrast to a true dichotomy (i.e., a variable that has two mutually exclusive and jointly exhaustive possibilities), the dichotomization of this continuous variable results in an artificial dichotomy.

How does the unobserved variable Ω_j that determines the performance on item j relate to the latent variable of interest, θ ? The latent variable Ω_j is a function of a common factor θ across all the items on an instrument plus an error factor that is unique to item j (Lord, 1980). Moreover, the regression of Ω_j on θ is linear (Lord & Novick, 1968). Figure C.2 depicts this regression for item j using a standard simple linear regres-

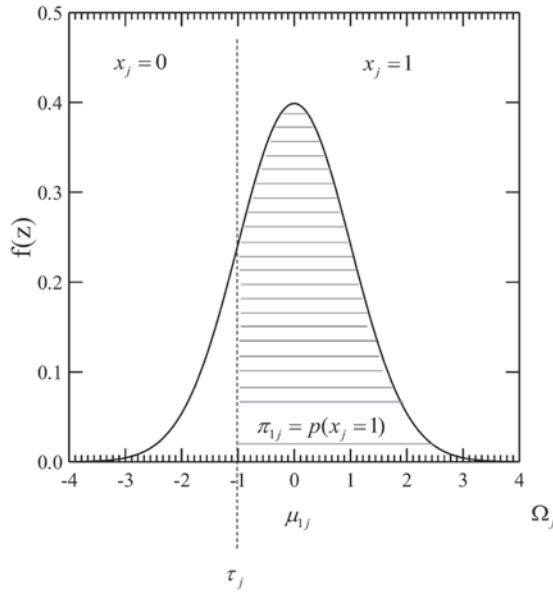


FIGURE C.1. Person 1’s location relative to item j ’s cutpoint.

sion presentation. As can be seen, the conditional distributions of Ω_j for fixed (predictor) values of θ are assumed to be normally distributed with mean $\mu_{j|\theta}$ and variance $\sigma_{j|\theta}^2$; because $\sigma_{j|\theta}^2$ is constant or homoscedastic across all conditional distributions it is symbolized as σ_j^2 . Note that although we are assuming that the latent variable Ω_j is normally distributed, we are *not* assuming that the people are normally distributed. That is, the continuous latent variable Ω_j reflects the distribution of the responses to item j by a person who is presented the item an infinite independent number of times.

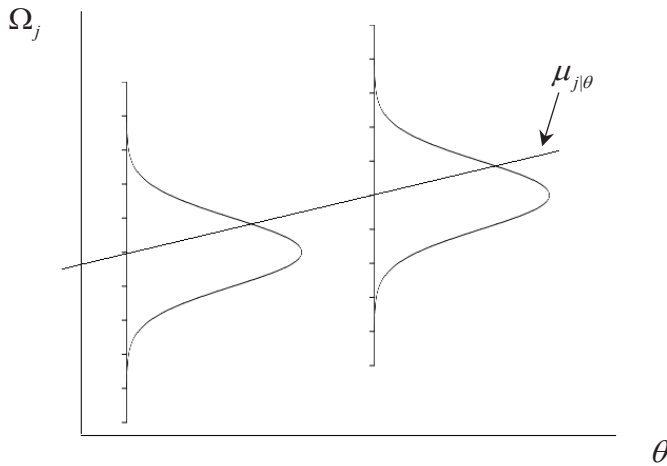


FIGURE C.2. Regression of Ω_j on θ for item j .

Combining the ideas behind Figure C.1 with those underlying Figure C.2 results in Figure C.3. As we see, these two conditional distributions reflect different probabilities of obtaining a response of 1 conditional on θ and item j 's threshold. For a low θ value (i.e., the left conditional distribution) the item's cutpoint results in an area that is substantially less than that for a higher θ value (i.e., the right conditional distribution). This implies that the item discriminates across the θ continuum and the degree of discrimination is related to the slope of the regression line.

To find the probability of a response of 1 (π_{1j}) for the right conditional (normal) distribution, one converts the τ_j to its corresponding z-score

$$z_{\tau_j} = \frac{\tau_j - \mu_{1j\theta}}{\sigma_j} \tag{C.1}$$

and determines the area at and above z_{τ_j} . For convenience the metrics for Ω_j and θ are standardized so that their marginal distributions have means of 0 and standard deviations of 1 (Lord, 1980). Consequently, one may use the standard unit normal distribution to determine the area that falls at and above z_{τ_j} (e.g., π_{1j}).¹ This area would be the probability of a response of 1 on item j conditional on θ . For instance, if $z_{\tau_j} = -1$, then $\pi_{1j} = 0.84$. In a similar fashion, the probability of a response of 1 for the left conditional distribution (π_{2j}) could be obtained. Assuming $z_{\tau_j} = 1$ for this latter distribution, then $\pi_{2j} = 0.16$. Thus, z_{τ_j} and τ_j are related to the difficulty of endorsing the item. Using the standard unit normal curve distribution to obtain these probabilities is tantamount to performing the integration from z_{τ_j} to ∞ under the unit normal distribution. This may be represented symbolically as

$$\pi(x_j = 1) = \int_{z_{\tau_j}}^{\infty} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right) dz . \tag{C.2}$$

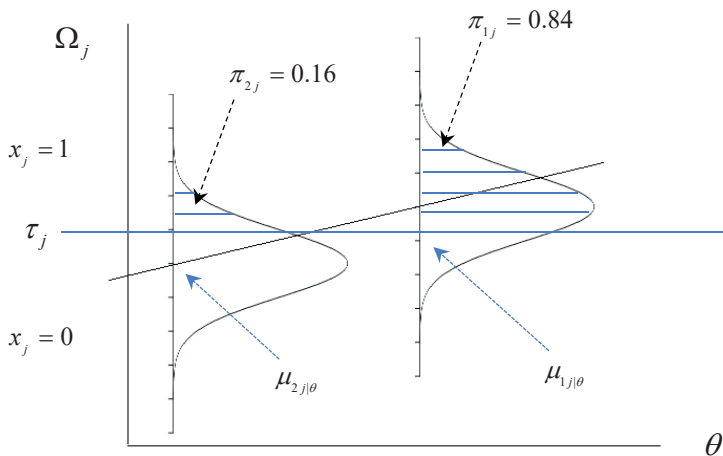


FIGURE C.3. Regression of Ω_j on θ and its relationship to item j 's cutpoint.

Figure C.3 shows the regression line for predicting Ω_j from θ . Because Ω_j and θ have been standardized, the intercept equals 0 and the regression equation for predicting Ω_j from θ simplifies to

$$\mu'_{ij|\theta} = \rho_j \theta. \quad (\text{C.3})$$

Therefore, our regression coefficient equals the correlation, ρ_j , between Ω_j and θ . Following Lord (1980), item j 's conditional standard deviation (the standard error) is $\sigma_{j|\theta} = \sqrt{1 - \rho_j^2}$. By substitution of Equation C.3 and $\sigma_{j|\theta}$ into Equation C.1, one obtains

$$z_{\tau_j} = \frac{\tau_j - \mu'_{ij|\theta}}{\sigma_j} = \frac{\tau_j - \rho_j \theta}{\sqrt{1 - \rho_j^2}} \quad (\text{C.4})$$

Lord and Novick (1968) define item j 's discrimination parameter, α_j , and its location, δ_j , in terms of the steepness of the regression line for predicting Ω_j from θ and the conditional variability about this regression

$$\alpha_j \equiv \frac{\rho_j}{\sqrt{1 - \rho_j^2}} \quad (\text{C.5})$$

Because τ_j is related to the difficulty of endorsing an item and τ_j / ρ_j is the point on the continuum where the probability of a response of 1 is 0.5, item j 's location, δ_j , is defined as

$$\delta_j \equiv \tau_j / \rho_j. \quad (\text{C.6})$$

By substitution of Equations C.5 and C.6 into Equation C.4 one obtains, upon simplification,

$$-z_{\tau_j} = \alpha_j (\delta_j - \theta). \quad (\text{C.7})$$

That is, the location of item j 's standardized threshold that delimits the response of 1 from that of 0 is a function of how well item j discriminates and its location on the latent variable of interest.

We can extend the idea embodied in Figure C.3 to a series of the conditional distributions of Ω_j . For each of these Equation C.2 can be used to calculate the probability of a response of 1 on item j . The graphing of these probabilities as a function of θ would produce an S-shaped curve or an ogive (Figure C.4). These probabilities may be traced by the *standard normal ogive function*²

$$\pi(x_j = 1) = \int_{-\infty}^z \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right) dz \quad (\text{C.8})$$

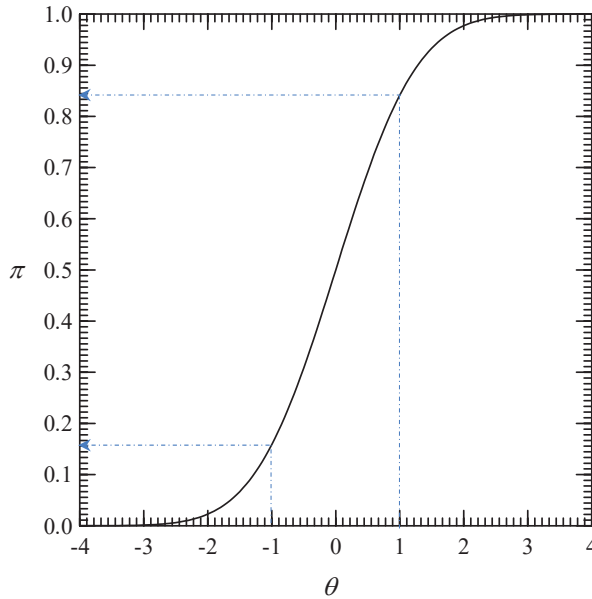


FIGURE C.4. Normal ogive with $\pi_{1j} = 0.84$ and $\pi_{2j} = 0.16$ overlaid.

In the current context the z in Equation C.8 is replaced by z_{τ_j} (i.e., $z_{\tau_j} = -(-z_{\tau_j})$) so that by substitution of Equation C.7 into Equation C.8 one obtains the *two-parameter normal ogive model* (Lord, 1952)

$$\begin{aligned} \pi(x_j = 1) &= \int_{-\infty}^{\alpha_j(\theta - \delta_j)} \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-(z^2)}{2}\right) dz \\ &= \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\alpha_j(\theta - \delta_j)} \exp\left(\frac{-(z^2)}{2}\right) dz \end{aligned} \tag{C.9}$$

where α_j and δ_j are the discrimination and location parameters for item j , respectively. (We use π instead of p to represent the probability from the two-parameter normal ogive.) It should be noted that the model in Equation C.9 does *not* make “any assumption about the distribution of” θ in the total group administered the instrument (Lord, 1980, p. 32). As is the case with the 2PL model, the item is located at the point where the probability of a response is 0.5, because when $\theta = \delta_j$ the integral has the limits $\int_{-\infty}^{z=0}$ and evaluates to 0.5. The term α_j is proportional to the slope of the IRF at δ_j . (Specifically, the slope is $\alpha_j/\sqrt{2\pi}$.)³ In contrast to the use of the logit model (e.g., for the 2PL model), the model in Equation C.9 is a use of the probit model.⁴

Birnbaum (1968) modified the model in Equation C.9 to include a lower nonzero asymptote parameter, χ_j , to address the observation that even “subjects of very low ability will sometimes give correct responses to multiple-choice items, just by chance”

(Birnbaum, 1968, p. 404). This model is referred to as the *three-parameter normal ogive model*

$$\pi(x_j = 1 | \theta, \alpha_j, \delta_j, \chi_j) = \chi_j + (1 - \chi_j) \int_{-\infty}^{\alpha_j(\theta - \delta_j)} \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right) dz, \quad (\text{C.10})$$

where χ_j is the pseudo-guessing (lower asymptote) parameter. Because the slope under the three-parameter normal ogive model involves the pseudo-guessing parameter as the pseudo-guessing value increases the IRF's slope decreases. (Specifically, the slope is $\alpha_j(1 - \chi_j) / \sqrt{2\pi}$; Lord, 1975.) As is the case with the logistic models, one can obtain the one-parameter normal ogive model by fixing χ_j to zero and holding α_j constant across items.

THE RELATIONSHIP BETWEEN IRT STATISTICS AND TRADITIONAL ITEM ANALYSIS INDICES

In traditional item analysis the proportion of correct responses to an item is the item's measure of difficulty. This proportion is typically referred to as the item's P-value, P_j , with large values indicating easy items and small P_j values reflecting difficult items. Moreover, there are several indices for assessing an item's discrimination power. Two of these are the item's point biserial and biserial correlation coefficients.

We begin by focusing on the biserial correlation as the discrimination index. Recall that the biserial correlation coefficient is a measure of the association between a continuous normally distributed variable and another continuous normally distributed variable that has been dichotomized (e.g., a variable such as Ω_j). To specify the relationship between the biserial correlation and the IRT discrimination parameter, we need to make two assumptions. First, because the biserial correlation assumes that both the dichotomized and the continuous variables are normally distributed we need to assume that both Ω_j and the *latent variable* θ are normally distributed. The second assumption is that there is no guessing on the item. Under these assumptions, Tucker (1946) and Lord and Novick (1968) show that the biserial correlation between the responses to an item j and the latent trait θ is related to the item's discrimination parameter by

$$\rho_{b_j} = \frac{\alpha_j}{\sqrt{1 + \alpha_j^2}}. \quad (\text{C.11})$$

Therefore, as the item discrimination increases so does the correlation between the item response and the latent variable.

Because θ is unknown it is not possible to calculate the biserial correlation between the responses to an item j and the latent trait θ . However, to the extent the observed score is a reasonable proxy or measure of θ (e.g., the instrument is of sufficient length and homogeneity; Urry, 1974), then one may calculate the biserial correlation, r_b , between the responses to item j and the observed score to estimate ρ_b . In this regard,

the relationship between item j 's traditional discrimination index, r_{b_j} , and the IRT discrimination parameter, α_j , may be expressed as (cf. Lord, 1980; also see Equation C.5)

$$\alpha_j \cong \frac{r_{b_j}}{\sqrt{1 - r_{b_j}^2}}. \tag{C.12}$$

Therefore, as the correlation between the item and the observed score increases, α_j also increases. The traditional item discrimination index can also be expressed in terms of the IRT item discrimination parameter by rearranging Equation C.12

$$r_{b_j} \cong \frac{\alpha_j}{\sqrt{1 + \alpha_j^2}}. \tag{C.13}$$

As mentioned above, a second traditional discrimination index is the point biserial correlation. The point biserial correlation gives the association between a true dichotomy and a normally distributed continuous variable. We can relate the point biserial correlation between the binary responses to an item and θ to obtain the IRT discrimination parameter. To do this we use the relationship between the point biserial and the biserial correlations. This relationship requires knowing the height of the standard unit normal curve at the dichotomizing point. At and above this point or threshold, τ_j , the response to the item is a 1 with an area represented by the shaded region in Figure C.1, π_j . The height of the standardized normal distribution at the threshold is given by

$$Y(\tau_j) = \frac{1}{\sqrt{2\pi}} \left[\exp\left(-\frac{\tau_j^2}{2}\right) \right]. \tag{C.14}$$

By using the covariance between the dichotomized item and the latent trait θ ,

$$\sigma_{j\theta} = \rho_{b_j} (Y(\tau_j))$$

and dividing it by the standard deviation of the dichotomized item (i.e., $\sigma_j = \pi_j(1 - \pi_j)$) one obtains the point biserial, ρ_{pb_j} , as a function of the biserial correlation

$$\rho_{pb_j} = \frac{\sigma_{j\theta}}{\sigma_j} = \rho_{b_j} \left[\frac{Y(\tau_j)}{\sqrt{\pi_j(1 - \pi_j)}} \right]. \tag{C.15}$$

As is the case with the biserial correlation, if the observed score is a reasonable proxy or measure of θ , then the point biserial correlation, r_{pb} , between the responses to item j and the observed scores serves as an estimate of ρ_{pb} ; in this case the item's P_j is used instead of π_j . By solving for ρ_b in Equation C.15 we can transform our estimated r_{pb} to its corresponding r_b and then apply Equation C.12. The point biserial is more appropriate than the biserial for situations that involve guessing.

We now turn our attention to the relationship between item j 's location, δ_j , and its traditional item difficulty index, P_j . Recall that P_j is the proportion of respondents cor-

rectly responding to an item. If we assume that θ is normally distributed (specifically, $N(0,1)$), then the proportion of respondents correctly answering item j corresponds to an area under this distribution. This area is delimited by a cutpoint, z_{t_j} , and the relationship between z_{t_j} and P_j is depicted in Figure C.5. By using Equation C.6, assuming that there is no guessing on item j , and because we have only sample information, we can express the relationship between the traditional item difficulty index and the item's location as (cf. Lord, 1952, 1980; Tucker, 1946)

$$\delta_j \equiv \frac{\Phi^{-1}(1 - P_j)}{r_{b_j}} = \frac{z_{t_j}}{r_{b_j}}, \quad (\text{C.16})$$

where z_{t_j} is the standard unit normal deviate that delimits an area to its left equal to $1 - P_j$ and an area P_j to its right (Figure C.5) and $\Phi^{-1}(\bullet)$ is the inverse (cumulative) normal function.⁵ Because high values of P_j indicate the same thing as low values of δ_j (e.g., “easiness”) we use the complement of P_j (i.e., $1 - P_j$) so that z_{t_j} may be interpreted similar to δ_j . The relationship between P_j and δ_j is dependent on how well the item discriminates. When all items discriminate equally well, then as P_j increases δ_j decreases.

We can use Equations C.12 and C.16 to “estimate” the IRT parameters α_j and δ_j . To demonstrate this we use the traditional item statistics from our mathematics data to estimate their corresponding IRT parameters. The traditional item difficulty and discrimination indices for item 1 are $P_1 = 0.887$ and $r_{b_1} = 0.407$, respectively. Therefore, with $(1 - P_1) = 1 - 0.887 = 0.113$ we obtain $z_{t_1} = -1.210727$. (In Excel function we use

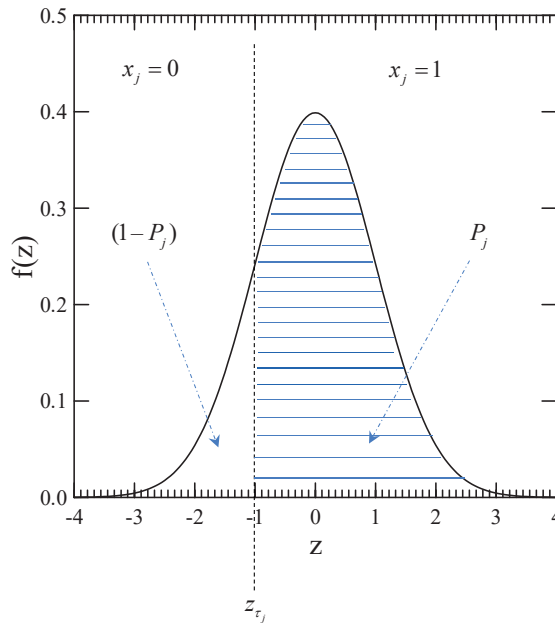


FIGURE C.5. Relationship of z_{t_j} , $(1 - P_j)$, and P_j .

NORM.INV((1-0.887), 0, 1) or in R $qnorm(0.113, 0, 1) = -1.210727$.) Substituting these values into Equation C.16 yields an IRT location estimate of

$$\hat{\delta}_1 = \frac{-1.210727}{0.407} = -2.97476$$

and a discrimination estimate of

$$\tilde{\alpha}_1 \cong \frac{r_{b_1}}{\sqrt{1-r_{b_1}^2}} = \frac{0.406}{\sqrt{1-0.407^2}} = 0.4456.$$

The relationships between P_j and δ_j as well as between α_j and r_{b_j} , may appear to provide a convenient approach for estimating α_j and δ_j . However, there are various reasons why the JMLE (Chapter 3) and MMLE (Chapter 4) techniques are to be preferred over using Equations C.12 and C.16. First, recall that both P_j and r_{b_j} are sample dependent, whereas α_j and δ_j are sample independent. Second, Equations C.12 and C.16 hold only when the latent variable is normally distributed and there is no guessing on the items. Third, because the observed score, X , contains error and θ does not, the X and θ are distributed differently, and these approximations using Equations C.12 and C.16 “fall short of accuracy” (Lord, 1980, p. 33). Fourth, Equations C.12 and C.16 do not provide standard errors for the α_j and δ_j estimates. As such, we do not know how accurately the parameters are being estimated. Moreover, research has shown that the approximation approaches of Equations C.12 and C.16 do not produce estimates that are as accurate as the MLE approach. For instance, Jensen (1976) compared these approximation approaches with MLE and found that the MLE estimates were more highly linearly related to their parameters than the estimates based on Equations C.12 and C.16. Specifically, for MLE we have $r_{\alpha\hat{\alpha}} = 0.863$ and $r_{\delta\hat{\delta}} = 0.971$, whereas using Equations C.12 and C.16 we have $r_{\alpha\hat{\alpha}} = 0.798$ and $r_{\delta\hat{\delta}} = 0.963$. Similar results were reported by Swaminathan and Gifford (1983). Furthermore, the accuracy of the estimates increased as sample size and test length increased, and decreased as α increased. The foregoing notwithstanding, Equations C.12 and C.16 can be used to provide provisional estimates or starting values for MMLE and JMLE.

RELATIONSHIP OF THE TWO-PARAMETER NORMAL OGIVE AND LOGISTIC MODELS

Because of the normal ogive model’s long history there was a desire with the introduction of the logistic form to make its results similar to those obtained from the normal ogive. The scaling constant $D = 1.702$ makes the logistic model’s values similar to those of the normal ogive model. (See Camilli [1994] for a discussion on the origin of D .)

The introduction of the scaling constant D into Equation 5.1 give us

$$p(x_j = 1 | \theta, \alpha_j, \delta_j) = \frac{e^{D\alpha_j(\theta - \delta_j)}}{1 + e^{D\alpha_j(\theta - \delta_j)}} \tag{C.17}$$

Equation C.17 yields probabilities from the logistic distribution function that are similar to those produced by Equation C.9. In effect, the use of D aligns, as closely as possible, the logistic function with the standard normal ogive function by changing the slope of logistic ogive. The standard normal ogive and the logistic functions intersect at a probability of 0.50.

To demonstrate the correspondence between the logistic and normal ogive two-parameter models we calculate the probability of a response of 1 when $\alpha = 1.5$ and $\delta = 1.0$ (item 2 from Figure 5.1). To calculate Equation C.9 we use the Excel function =NORM.DIST(($\alpha_j(\theta - \delta_j)$), 0,1,TRUE); see Endnote 1. For example, for $\theta = 0.5$ we have

Equation C.9:
$$\pi(x_j = 1) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{1.5(0.5-1.0)} \exp\left(\frac{-(z^2)}{2}\right) dz = 0.22663$$

Equation C.17:
$$p(x_j = 1) = \frac{e^{1.702[1.5(0.5-1)]}}{1 + e^{1.702[1.5(0.5-1)]}} = 0.21815$$

Equation 5.1:
$$p(x_j = 1) = \frac{e^{1.5(0.5-1)}}{1 + e^{1.5(0.5-1)}} = 0.32082$$

As can be seen, the discrepancy Equations C.9 and C.17 is about 0.01. This close correspondence is exemplified in this item's IRFs (Figure C.6); NORM_2P is two-parameter

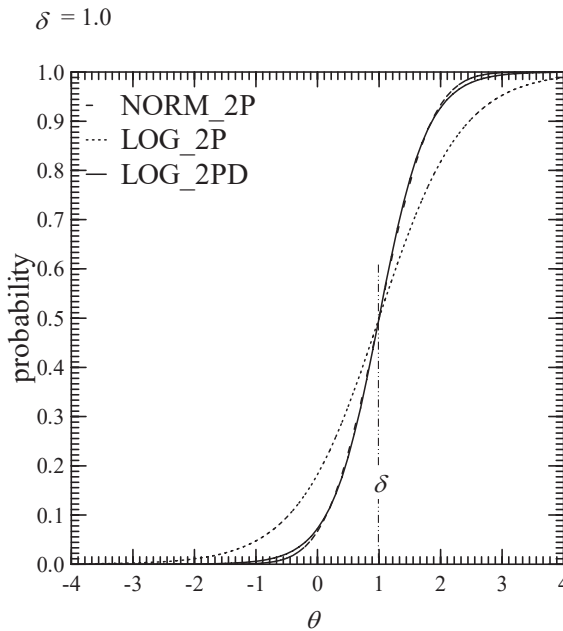


FIGURE C.6. IRFs for two-parameter normal ogive and logistic with and without D ; $\alpha = 1.5$ and $\delta = 1.0$.

normal ogive (Equation C.9), LOG _ 2P is the 2PL model (Equation 5.1), and LOG _ 2PD is the 2PL model incorporating the scaling constant D (Equation C.17). The two-parameter normal ogive and the 2PL model with D are virtually indistinguishable in the neighborhood around δ .

The introduction of D into the model changes the formulas for the slope and item information. With respect to the two-parameter logistic model's slope (i.e., the first derivative of the model, p'_j), the introduction of D results in the first derivative becoming

$$p'_j = D\alpha_j p_j(1 - p_j). \tag{C.18}$$

By substitution for p_j and noting that α_j is defined at $\theta = \delta$, the slope becomes

$$p'_j = D\alpha_j \frac{1}{4} = 0.425\alpha_j. \tag{C.19}$$

Therefore, α_j is proportional to the slope of the tangent line to the IRF at δ_j .

As noted in Chapter 5, Equation 5.2, the general formulation for item information is

$$I_j(\theta) = \frac{(p'_j)^2}{p_j(1 - p_j)} \tag{C.20}$$

By substitution of Equation C.18 for p'_j into Equation C.20, one arrives at the item information function for the two-parameter logistic model

$$I_j(\theta) = D^2\alpha_j^2 p_j(1 - p_j). \tag{C.21}$$

Using the logistic forms offers some advantages over using the normal forms of IRT models. For instance, one advantage of using Equation C.17 is the elimination of the integral in Equations C.9 and C.10 to make the mathematics simpler. A second advantage of the model shown in Equation C.17 is that, unlike with the normal ogive IRF, there are sufficient statistics for estimating person location (Lord, 1980).

When the logistic and normal ogive models provide good fit, “parameter estimates in logistic models are about 1.6 – 1.8 times those in probit models” (Agresti, 1990, p. 104). Therefore, the model shown in Equation C.17 may be viewed as a mathematically convenient, close approximation to the (classical form of the) two-parameter normal ogive model. However, it should be noted that D 's value is not concerned with model–data fit. In this regard, and because of the indeterminacy of the metric, D may be set to any other convenient value (e.g., 1) without adversely affecting model–data fit. As a consequence, in this book the logistic model *without* the use of D (e.g., Equation 5.1) is considered to be an intrinsically useful model and we are not concerned with approximating the normal ogive form. By not using D the calibration results are on the *logistic metric*; the use of D ensures that the results are on the *normal metric*. In those situations where it is necessary to invoke D (e.g., in comparisons involving NOHARM) the reader is alerted to the use of D .

EXTENDING THE TWO-PARAMETER NORMAL OGIVE MODEL TO A MULTIDIMENSIONAL SPACE

As mentioned above, the latent variable Ω_j is a function of a common factor θ across all items on an instrument plus a term that is unique to item j . As such, the two-parameter normal ogive model may be seen as related to a unidimensional common factor analysis model (e.g., see McDonald, 1967, 1997). (It is this relationship that is used in Appendix G “Using Principal Axis for Estimating Item Discrimination.”) Moreover, this relationship may be extended to a nonunidimensional common factor model. In this respect, Ω_j is a function of multiple weighted θ s

$$\Omega_j = \underline{\rho}_j \underline{\theta} + E_j \quad (\text{C.22})$$

where $\underline{\rho}_j'$ is a row vector of factor loadings (i.e., $\underline{\rho}_j' = \{\rho_{j1}, \dots, \rho_{jF}\}$), $\underline{\theta}$ is a vector of person locations (i.e., $\underline{\theta} = \{\theta_1, \dots, \theta_F\}$), and E_j is item j 's unique factor. To develop our multidimensional model we begin with the two-parameter normal ogive model. Assume that E_j and $\underline{\theta}$ are normally distributed and that Ω_j has been standardized to have a mean of 0 and a variance of 1. Therefore, Ω_j is also normally distributed; this is stated as an assumption above. Then, from above, we have the probability of a response of 1 is

$$\pi(x_j = 1 | \underline{\theta}) = \pi(\Omega_j > \tau_j | \underline{\theta}) = \Phi(z_{\tau_j}), \quad (\text{C.23})$$

where $\Phi(\square)$ is the cumulative normal distribution function. Focusing on z_{τ_j} we have

$$z_{\tau_j} = \alpha_j(\theta - \delta_j) = -\alpha_j \delta_j + \alpha_j \theta. \quad (\text{C.24})$$

By substitution of Equations C.5 and C.6 into Equation C.24 we can express our item's intercept in terms of the item's loading and its threshold

$$\gamma_j = -\alpha_j \delta_j = -\left[\frac{\rho_j}{\sqrt{1-\rho_j^2}} \right] \delta_j = -\left[\frac{\rho_j}{\sqrt{1-\rho_j^2}} \right] \left[\frac{\tau_j}{\rho_j} \right] = -\left[\frac{\tau_j}{\sqrt{1-\rho_j^2}} \right]. \quad (\text{C.25})$$

(That is, because factor loadings are the biserial correlations of the responses with θ , Equation C.5 may be interpreted as expressing the item's discrimination in terms of its loadings.) When we substitute Equations C.25 and C.5 into Equation C.24 we arrive at

$$z_{\tau_j} = -\alpha_j \delta_j + \alpha_j \theta = \frac{\tau_j + \rho_j \theta}{\sqrt{1-\rho_j^2}} \quad (\text{C.26})$$

We now have a vehicle to incorporate multiple dimensions. Specifically and following McDonald (1997), we can express Equation C.26 in terms of F -dimensional vectors of factor loadings, $\underline{\rho}_j'$, and person locations, $\underline{\theta}$ (also see McDonald, 1999; McDonald & Mok, 1995)

$$z_{\tau_j} = \frac{\tau_j + \underline{\rho}_j' \underline{\theta}}{\sqrt{1 - \underline{\rho}_j' \underline{\Sigma} \underline{\rho}_j}}, \quad (\text{C.27})$$

where $\underline{\Sigma}$ is a covariance matrix and $1 - \underline{\rho}_j' \underline{\Sigma} \underline{\rho}_j$ is the item's unique variance across the F -dimensions (i.e., $1 - \sum \rho_j^2 = 1 - h^2$). By substitution of Equations C.5, C.25, and C.26 into Equation C.23 we arrive at a *multidimensional two-parameter normal ogive model*

$$\pi(x_j = 1 | \underline{\theta}) = \Phi(z_{\tau_j}) = \Phi\left(\frac{\tau_j + \underline{\rho}_j' \underline{\theta}}{\sqrt{1 - \underline{\rho}_j' \underline{\Sigma} \underline{\rho}_j}}\right) = \Phi(\gamma_j + \underline{\alpha}' \underline{\theta}). \quad (\text{C.28})$$

In this parameterization the intercept and slopes (discriminations) are obtained by

$$\gamma_j = \frac{\tau_j}{\sqrt{1 - \underline{\rho}_j' \underline{\Sigma} \underline{\rho}_j}} \quad (\text{C.29})$$

and

$$\alpha_j = \frac{\rho_j}{\sqrt{1 - \underline{\rho}_j' \underline{\Sigma} \underline{\rho}_j}}, \quad (\text{C.30})$$

respectively. Conversely, we have that

$$\tau_j = \frac{\gamma_j}{\sqrt{1 + \underline{\rho}_j' \underline{\Sigma} \underline{\rho}_j}} \quad (\text{C.31})$$

and

$$\rho_j = \frac{\alpha_j}{\sqrt{1 + \underline{\rho}_j' \underline{\Sigma} \underline{\rho}_j}}. \quad (\text{C.32})$$

NOTES

1. The Excel (Microsoft Corporation, 2018) function =NORM.DIST(z_{τ_j} , 0, 1, TRUE) or the R function pnorm(z_{τ_j} , 0, 1, TRUE) can be used to obtain π_{1j} . For the area above z_{τ_j} the function's value is subtracted from 1 (e.g., $=1 - \text{NORM.DIST}(z_{\tau_j}, 0, 1, \text{TRUE})$).

2. Because the distribution is symmetric $\int_{-\infty}^z = \int_z^{\infty}$ Equation C.8 follows from Equation C.2. Below we use the symbol $\Phi(\square)$ to represent this standard normal ogive function.

3. The value of σ_j in the calculation of z_{τ_j} affects the magnitude of z_{τ_j} (Equation C.4). As σ_j increases then z_{τ_j} decreases, all other things being equal. In addition, as the z_{τ_j} s decrease the corresponding π_j s decrease. Therefore, the corresponding IRF's slope decreases, all things being equal. Conversely, as σ_j decreases then z_{τ_j} increases and the IRF's slope increases. As

such, because α is proportional to the IRF's slope there is an inverse relation between α and σ_j (i.e., $\alpha = 1/\sigma_j$). Because the metric is standardized to have a mean of 0 and a standard deviation of 1, the unit of measurement becomes the standard deviation unit. As a result, σ_j is referred to as a scale parameter and $1/\sigma_j$ is sometimes called *dispersion* (cf. Bock & Lieberman, 1970; Thurstone, 1925).

4. Similar models are presented by Lawley (1943, 1944), Tucker (1946), and Thurstone (1925). For example, given the cumulative normal ogive function in Equation C.8 we have z is the unit normal deviate that delimits an area corresponding to the probability of a response of 1. Let z for person i and item j , z_{ij} , be defined as

$$z_{ij} = \frac{(\theta_i - \mu_j)}{\sigma_j} \quad (\text{C.33})$$

where θ_i is person i 's location on the latent variable, μ_j and σ_j are the mean and standard deviation of the normal curve with respect to item j , respectively; this distribution is assumed to be normal with

$$\sigma_j = \frac{1}{\alpha_j} \quad (\text{C.34})$$

If we take our total sample of individuals and divide it into subgroups and redefine the standard deviation in Equation C.33 to be the standard deviation of a subgroup, σ_i , with mean μ_i , then its substitution into Equation C.2 gives Thurstone's mental age model; we're assuming that each subgroup is normally distributed. That is, Thurstone (1925; e.g., see p. 441) developed a model based on the cumulative normal distribution to determine the proportion of individuals of a specified age group correctly responding to an item.

5. To determine the z_{τ_j} corresponding to $\Phi^{-1}(P_j)$ the Excel function =NORM.INV(P_j , 0, 1) or the R function qnorm(P_j , 0, 1) can be used.

Appendix D

Computerized Adaptive Testing

“The facts are clear. From the point of view of measurement, tailored testing offers little, if any, advantage over the best that can be done with conventional testing” (Green, 1970, p. 184). Although Professor Green reached this conclusion based on the research on computerized adaptive testing (CAT) in 1970, he proceeded to present an argument against the perspective that CAT provides little advantage over conventional “paper-and-pencil” testing. In this appendix we provide a brief introduction to CAT from a proficiency assessment perspective. However, it should be noted that CAT can be applied to other psychological domains.

Computerized testing initially used the computer to simulate a paper-and-pencil test administration. This approach of administering items to an examinee without taking into account their responses is sometimes called a *linear test*. Therefore, the computerized linear test and the conventional paper-and-pencil testing procedure administer the same items to every examinee in a fixed fashion regardless of the examinee’s responses to the items. Because the examinees most likely vary in the proficiency being measured, some items are too difficult for certain examinees, whereas others are too easy. This undermines the effectiveness of the test, but is inevitable whenever the items administered are not tailored to the individual examinee. In contrast, and in the most simplistic terms, with CAT the items administered are selected for the examinee, given the most current information about the examinee’s proficiency and the items available in the pool. Although no method of administering items and scoring dichotomous responses can produce better measurement than that achieved by a “standard test” at a proficiency level equal to zero (on the theta scale), an adaptive test tries to achieve this level of accuracy *throughout* the proficiency range (Lord, 1971a). In so doing it achieves equiprecise measurement across the continuum. Additional advantages of CAT over conventional paper-and-pencil tests are a comparative (potential) test length reduction of 80% and the capacity to administer questions that take advantage of the computerized administration mode and that could not be administered with a conventional paper-and-pencil test.

The concept of adapting an instrument to an individual can be traced back over a century. Throughout this time computerized adaptive testing has had many different names, such as *tailored testing*, *response-contingent testing*, *sequential testing*, and *programmed testing*. Regardless of what the concept has been called, it has primarily been concerned with minimizing the measurement errors associated with estimating an individual's location. We begin with a brief history of adaptive testing and then proceed to discussing CAT.

A BRIEF HISTORY

The first adaptive test is considered to be the individually administered Binet–Simon intelligence test developed in the early 1900s (Weiss, 1982). In this test the particular subtests administered were chosen on the basis of the examinee's current ability level as determined during the testing procedure. That is, if an examinee passed all or any of the subtests within an ability level, then a higher-ability level of subtests is subsequently administered. Conversely, if an examinee fails all subtests at a given ability level, then the test is terminated. Therefore, the Binet test is adaptive with respect to ability level. Binet's procedure differs from present-day tailored testing in that it requires the examinee to answer all the questions associated with a particular ability level (Wood, 1973) and its administration requires a highly trained examiner rather than a computer.

In the 1940s two procedures, the staircase method and the sequential analysis system, were developed (Wood, 1973). The sequential analysis system has seen some use in mastery testing (e.g., see Reckase, 1980). The staircase method is analogous to the methods used by psychophysicists. Experimental psychologists' have used adaptive testing procedures in their psychophysical experiments for decades. Their methods, called adaptive convergence procedures, include the method of adjustment and the method of limits (Weiss, 1983).

In 1951, Hick presented all the ingredients of adaptive testing as it is now understood (Wood, 1973). In his article he stated that an intelligence test should be a branch process, with all questions having a 0.5 chance of being correctly answered. Patterson (1922; cited in Wood, 1973) took a pool of items and arranged them in such a way that an examinee, starting with an average difficulty item, would receive a harder item if they correctly answered the previous item and an easier item if they had answered the item incorrectly. Fixed-branching methods like those used by Patterson, and using traditional item statistics, were used during most of the 1960s.

In 1970, Lord outlined some test theory for tailored testing. Based on the study of various testing algorithms used in that article, he stated that better measurement could be obtained by selecting and administering, for example, the 60 most discriminating items as a conventional test rather than administering, for example, 500 items in a tailored testing procedure. It is ironic that Lord's (1970) work and its indisputably poor results marked the beginning of the integration of IRT with tailored testing procedures. In this regard, there have been a number of procedures developed.

The literature contains various taxonomies for grouping the different types of adaptive testing strategies (e.g., Hambleton & Swaminathan, 1985; Lord, 1970; Reckase, 1977; Vale, Albing, Foote-Lennox, & Foote-Lennox, 1982). These taxonomic schemes differ in their organization and terminology. For example, Reckase (1977) differentiates among methods depending on whether the adaptive testing method uses a mathematical model for determining the examinee's path through the item pool. Specifically, if the method uses a mathematical model for item selection, then the technique is classified as model-based; otherwise, the method is classified as a structure-based method. The former item selection type may be called variable-branching item selection, whereas the latter may be called fixed-branching item selection. It is this latter terminology that we adopt in the following discussion.

FIXED-BRANCHING TECHNIQUES

Fixed-branching strategies use a predetermined or fixed routing procedure through an item pool. The arrangement of items in the pool in conjunction with the routing method define the item selection process (Patience, 1977). The item pool size is determined by the procedure used. Fixed-branching procedures may be implemented either on a computer or as a paper-and-pencil test. There are many possible fixed-branching techniques, and their number is limited only by the ingenuity of the test designer. Examples include, but are not limited to, the flexilevel test (Lord, 1971b, 1971c), the stradaptive test (Weiss, 1973), the pyramidal test (Larkin & Weiss, 1975), random-walk techniques (Lord, 1970), and the two-stage test (Cleary, Linn, & Rock, 1968; Lord, 1971d, 1980).¹ These approaches may or may not use an IRT model for person location estimation. When they do not, the proficiency estimate is a simple function of the responses to items and the items' characteristics. For instance, the estimated proficiency is the number of correct responses, a weighted composite of the items administered (e.g., the average of the difficulties of the items administered or the average of the difficulties of the items correctly answered), or a function of the difficulty of the last item administered and the difficulty of the item that would have been administered next (Reckase, 1977; Lord, 1970) such as the mean difficulty of the last and next items.

VARIABLE-BRANCHING TECHNIQUES

Variable-branching procedures usually use an IRT model for person location estimation. (One could use a different model, such as a latent class model.) The item selection process is designed to maximize the information about an examinee's location. Two commonly used techniques are to select items that produce a specified probability of a correct response for an examinee's location estimate or maximize the information function (Patience, 1977; Reckase, 1977). Because of the computations required for item selection and person location estimation, variable-branching procedures are typically

implemented on a computer. (A paper-and-pencil tailored test based on the Rasch model is presented by Fischer and Pendl [1980].) Typically, MLE, EAP, or MAP is used for estimating an individual's location. (In general, an individual's observed score is usually inappropriate as a proficiency estimate because each examinee may respond to different items and different numbers of items [Reckase, 1977].) The item pool is designed to maximize the computer program's efficiency in searching for a particular item to administer. Most of the current research in computerized adaptive testing uses variable-branching techniques.

ADVANTAGES OF VARIABLE-BRANCHING OVER FIXED-BRANCHING METHODS

Variable-branching procedures eliminate some of the problems encountered with fixed-branching methods. For instance, non-IRT-based fixed-branching tests use item characteristics that are dependent on the particular sample of examinees used in their calculation. Therefore, the item characteristics may (and probably will) vary from sample to sample and result in more error in the proficiency estimates. A second problem with these fixed-branching non-IRT techniques is that the proficiency estimates are expressed on a different metric than the item difficulty parameter estimates (Weiss, 1982). As a result, it is difficult to select items that use all the information in the examinee's response and that are of appropriate difficulty for the examinee. Third, unlike some of the fixed-branching methods, IRT-based variable-branching procedures produce proficiency estimates that are independent of the particular subset of items administered to an examinee. As a consequence, different items can be selected for administration for each examinee and the resulting proficiency estimates are on the same metric (Weiss, 1982). Furthermore, adaptive tests can be designed to cover as wide a range of ability as desired. Lord designed a test that placed examinees from fourth grade up to graduate school on the same score scale (Lord, 1977).

A fourth problem with fixed-branching tests concerns the method of test termination. Fixed-branching tests typically terminate when a preset number of items are administered. Therefore, the degree of precision in ability estimation is not controlled by the examiner. Because with IRT-based variable-branching tests the standard error of the person estimate is directly related to the test's reliability, a test can be terminated when a predetermined level of precision is reached. In other words, a test is terminated when a particular degree of reliability is attained (Urry, 1977).²

A fifth issue involves item selection. Whereas fixed-branching methods typically use a predefined item selection algorithm, the use of IRT parameters permits items to be selected on the basis of more than just their difficulty levels (Weiss, 1982). Consequently, item selection can simultaneously take into account the item's difficulty, its discrimination, and the pseudo-guessing parameter as well as other considerations (e.g., content). In addition, the first item administered can be based on considerations other than the item happens to be of median difficulty.

A further consideration is test security. Because variable-branching methods are typically computerized, they are harder to compromise than noncomputerized fixed-branching techniques. For example, there are no test booklets that can be stolen, item pools can be encrypted, and so on. Moreover, the greater flexibility in item selection of variable-branching adaptive testing methods reduces the chances of an examinee receiving the same test more than once in a test-retest situation.

IRT-BASED VARIABLE-BRANCHING ADAPTIVE TESTING ALGORITHM

Under certain conditions CAT leads to improved measurement relative to conventional paper-and-pencil tests. These conditions are (1) an appropriate item response model, (2) accurate estimates of item parameters, (3) the construction of a good item pool, and (4) efficient unidimensional (or multidimensional) procedures for adaptive testing (Urry, 1977). Although in the following discussion we assume a unidimensional model, it is possible to use CAT with multidimensional models. The reader interested in multidimensional CAT is referred to Luecht (1996), Seagall (1996), Reckase (2009), and van der Linden (1999).

Conceptually, IRT-based variable-branching strategies consist of selecting and administering the item that is expected to most improve the current proficiency estimate. In general, these items are selected such that the examinee is expected to have about a 50% chance of correctly answering the items. The premise for this item selection strategy is that a test is most effective in measuring an examinee's proficiency "when the examinee knows the answers to only about half of the test items" (Lord, 1970, p. 140).

The CAT algorithm consists of four basic components: (1) the selection of the first item to administer, (2) the scoring or processing of the examinee's response to obtain a location estimate, (3) the selection of another item for administration (this may or may not be the same as that used for the first component), and (4) stopping criterion/criteria for terminating a test. In general, the basic decision rule for item selection (i.e., the third component) is to select items that are progressively more appropriate for the examinee than those administered beforehand. Thus and in the context of proficiency assessment, if an examinee correctly answers an item, then the examinee's ability is most likely higher than the answered item's location. Consequently, for the next item to be more appropriate than the previous one, its location should be higher (i.e., "harder"). Conversely, if an examinee incorrectly answers an item, then the examinee's ability is most likely lower than the incorrectly answered item's location. As a result, for the next item to be more appropriate than the previous one, its location should be lower (i.e., "easier"). (The terms "harder" and "easier" are *relative* to the examinee's current ability estimate.)

CAT implementation requires a pool of items from which items are selected. Initially, this item pool can be created by administering conventional paper-and-pencil examinations, calibrating the data with the appropriate model, and linking the separate calibrations (see Chapter 11). Subsequently, items may be pretested within the CAT examinations to augment/replenish the item pool. Item pool size varies as a function

of the item characteristics, test security concerns, the nature of the examination (e.g., high-stakes), breadth of content to be covered, and so on. A rule of thumb is that the number of items should be at least 8 to 12 times the average CAT length. For example, for an examination that averages 25 items, this guideline would say the item pool should have 200 to 300 items. The items' parameter estimates are treated as known when estimating an examinee's proficiency.

The computerized adaptive test typically begins with making a guesstimate as to the examinee's initial location. For example, we could assume that the examinee is of average proficiency (i.e., $\hat{\theta} = 0$), use ancillary information about the examinee to provide an initial location estimate (e.g., from a subtest), or randomly select the initial location guesstimate from within a θ range, such as -0.50 to 0.50 . In general, the examinee's initial $\hat{\theta}$ should be in the region corresponding to the median of the item pool difficulty distribution. This would allow movement through the pool in either direction while minimizing problems stemming from "topping-out" or "bottoming-out" of the item pool after only a few items (Patience & Reckase, 1980). (*Topping-out* refers to having an examinee location estimate that is so high that there are no items in the pool that are appropriate for administration. Conversely, *bottoming-out* occurs when the examinee's location estimate is less than the least difficult item in the item pool.) Weiss (1982) has stated that, on the basis of his personal experience, most adaptive tests are shortened by only a few items with the use of accurate initial location estimates. Stated more positively, the more accurate the initial person location estimate, the more quickly the adaptive test will converge to the individual's proficiency estimate.

Once we have an initial person location estimate we can select the first item for administration. The approach to select first item interacts with the method used for obtaining the examinee's initial location estimate. As a result, there are a number of strategies that can be used for selecting this first item. For example, on the basis of the initial location estimate, the algorithm may select the most informative item in the item pool. However, in this case if we assume that each examinee is of average proficiency, then we will always administer the same first item. In practice one needs to be concerned with overexposing items. Therefore, to avoid overexposing the first item, the algorithm may randomly select the first item from a set of items that are roughly equally informative (i.e., in terms of item information). Some other first item selection possibilities are to simply randomly select an item of average difficulty; using the item that is most informative for a θ value corresponding to the mode of the item pool total information distribution; using the item that is most informative for a θ corresponding to the median of the item pool total information distribution; and selecting the item on the basis of external information. If we had used the randomly assigned/ θ range approach, then we could simply select the most informative item for our guesstimate. If the tailored test is reasonably long (e.g., 25 items), then the choice of the initial item has almost no effect on the standard error of the final person location estimate (Lord, 1977).

After the examinee responds to the administered item, the response is scored and this information is used to estimate the examinee's θ . Any of the approaches discussed in this book, such as MLE, EAP, or MAP, can be used. The implementation of MLE and EAP for CAT is identical to that presented in Appendix A and Chapter 4, respec-

tively. With either EAP or MAP we can estimate the person's location after scoring their response to the first item. However, this is not the case with MLE. With MLE it is not possible to obtain an estimate of the person's location until they have provided both correct and incorrect responses; for polytomous data the responses need to be in different categories (see Dodd, Koch, and de Ayala [1989]). Therefore, when we have zero-variance response vectors we need to modify our initial θ estimate without using MLE in order to select the next item. We present three strategies that, in effect, may be considered fixed-branch approaches.

One approach is to set the new $\hat{\theta}$ equal to the previous $\hat{\theta}$ plus or minus a fixed amount (e.g., step size = 0.3 logits). That is, if the examinee correctly responded to the first item, then the new $\hat{\theta}$ equals the initial estimate plus this fixed amount; otherwise the new $\hat{\theta}$ equals the initial estimate minus the fixed amount. In either case, the item administered is based on the new $\hat{\theta}$. A variant of this fixed step size approach is to use a variable step size. In this strategy, the step size used with the first item is successively divided in half until we have a response vector with correct and incorrect responses. For example, if the first step size is 0.30, the second step size would be 0.15, the third step size would be 0.075, and so on. This variable step size approach seeks to minimize the possibility of topping- or bottoming-out. An alternative variable step size approach is to simply select an item that is midway between the administered item's location and an item with an extreme location. In other words, if the person correctly responds to the administered item, then the next administered item would be midway between the administered item's location and the most difficult item. Conversely, if the person incorrectly responds to the administered item, then the next item would be midway between the easiest item and the administered item's location. Either approach is repeated until the examinee has provided both a correct and an incorrect answer.

Once we obtain a new $\hat{\theta}$, the next item administered is (1) the most informative item in the pool for the current proficiency estimate, (2) the item that yields the greatest weighted information, or (3) the item that will lead to the greatest reduction in the posterior distribution's variance.³ The first of these three strategies is known as the maximum global information or the maximum information search and selection technique (MISS; Kingsbury & Weiss, 1983). The last two selection strategies are associated with Bayesian estimation; MISS may be used with either MAP or EAP estimation.

This process of administering items, scoring the responses, and re-estimating the examinee's location continues until some termination criterion is satisfied. In this regard, there are two types of CAT examinations. With the *variable-length* CAT examination the length of the examination may differ across examinees. In this case, the termination criterion is either that the examinee's SEE ($s_e(\hat{\theta})$) is less than the maximum SEE criterion or that there are no more items remaining in the pool with information values greater than some minimum value; also see Jensen (1974). Typically, these criteria are used in conjunction with a maximum test length in case the minimum item information (or maximum SEE) criterion is not satisfied. The second type of CAT examination is a *fixed-length* CAT test. In this CAT the adaptive test terminates after a fixed number of items are administered. As a result, all examinees have an examination of the same length. We summarize the CAT algorithm in Figure D.1.

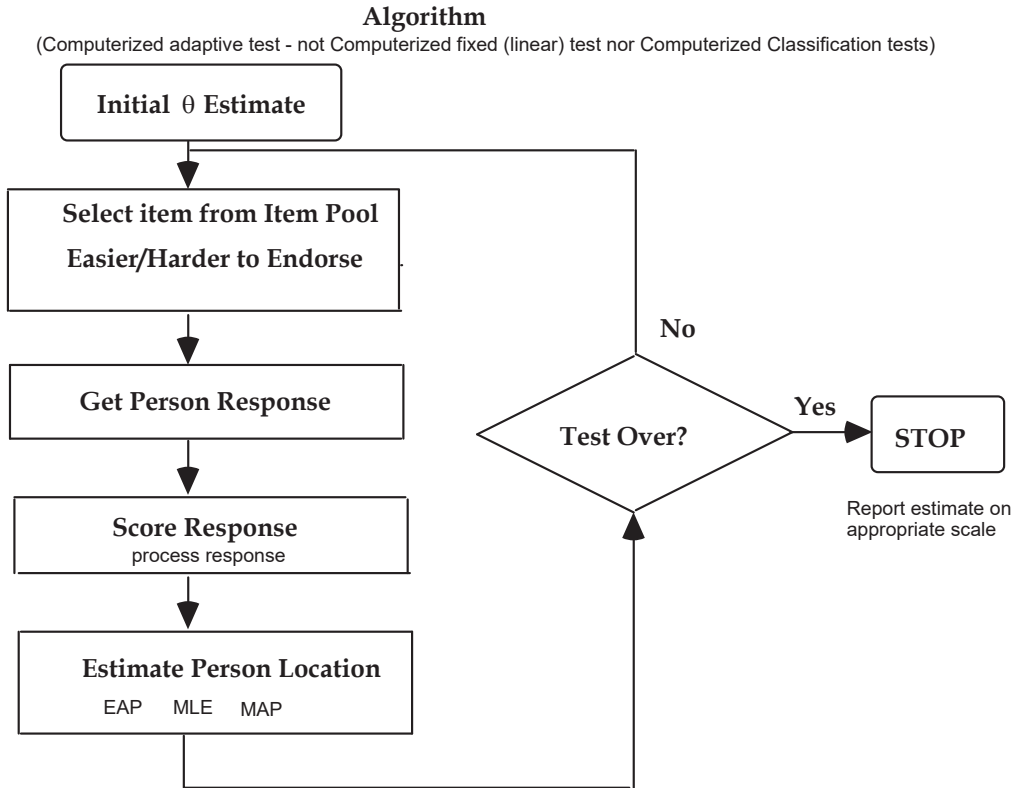


FIGURE D.1. Schematic of CAT algorithm.

In Table D.1 we present the results from a variable-length CAT examination simulation. For our simulation we use MLE with a fixed step size of 0.3. Items are selected on the basis of MISS with a minimum information criterion of 0.9. The maximum test length is 30 items and the item pool size is 240 items. The table's rows represent each item administered with the item's number, $\hat{\alpha}_j$, and $\hat{\delta}_j$ presented in the second, third (A), and fourth columns (B), respectively. For example, the first item administered is item #229 with an $\hat{\alpha}_{229} = 2.580$ and a $\hat{\delta}_{229} = 0.047$. To begin our CAT examination we assume the simulee is of average proficiency (i.e., $\hat{\theta} = 0$; PREVIOUS ESTIMATE column, row 1). The most informative item in the pool for $\hat{\theta} = 0$ is item #229 with an item information of 1.658 (INFO column, row 1). Because our examinee correctly responded to this item (RESP column, row 1) we add the step size of 0.3 to the initial $\hat{\theta}$ of 0 to obtain a REVISED ESTIMATE of 0.3. Based on this $\hat{\theta} = 0.3$ we select the second item to be administered. The most informative item is item #181 with an information value of 1.889 (INFO column, row 2). Again, the person correctly responds to the item and again we add the step size to our current $\hat{\theta}$.

We keep adding the step size until we administer the fifth item, at which point the examinee incorrectly responds (RESP column, row 5). Therefore, after administering five items we are able to use MLE because we now have a response vector that has at

least one correct response and one incorrect response; the actual response vector is $\underline{x} = 11110$. Our MLE $\hat{\theta}$ is 1.350 (REVISED ESTIMATE column, line 5) with a $s_e(\hat{\theta})$ of 0.429 (SEE column, line 5); the SEE = -99.900 for the first four items is used as a placeholder when we cannot use MLE. The process of item selection, re-estimation of θ , and checking to see if there are any more items with information values greater than 0.9 continues until we have administered 24 items. Examining the REVISED ESTIMATE and SEE columns in conjunction with the RESP column shows the algorithm homing in on the final proficiency estimate. After the administration of the 24th item (item #205), there are no items remaining in the pool that satisfy the minimum information criterion of 0.90 and the CAT examination stops. Our person location estimate at termination is $\hat{\theta} = 1.098$ with a $s_e(\hat{\theta}) = 0.192$. Of course, this $\hat{\theta}$ may be transformed to another metric such as the expected trait score, $\mathcal{E}T$, by Equation 4.23, or to some other target metric by Equation 4.17.

The foregoing is an oversimplification of CAT item selection. In practice, item selection involves content balancing, the pretesting of items, ensuring that items are not overused (i.e., exposure rate control), and so on. (More information on exposure rate control may be found in Georgiadou, Triantafyllou, and Economides [2007]; Hetter and

TABLE D.1. CAT Examination Audit Trail for One Person

ESTIMATED THETA= 1.098
INITIAL THETA= 0.000

ORDINAL POSITION	ITEM # ADMINISTERED	A	B	PREVIOUS ESTIMATE	RESP	REVISED ESTIMATE	INFO	SEE
1	229	2.580	0.047	0.000	1	0.300	1.658	-99.90
2	181	2.954	0.560	0.300	1	0.600	1.889	-99.90
3	182	2.424	0.508	0.600	1	0.900	1.451	-99.90
4	201	2.963	1.169	0.900	1	1.200	1.880	-99.90
5	189	2.772	1.174	1.200	0	1.350	1.918	0.429
6	146	2.876	1.746	1.350	0	1.252	1.520	0.372
7	143	2.436	1.358	1.252	1	1.415	1.459	0.351
8	232	2.431	1.349	1.415	0	1.282	1.468	0.315
9	193	2.446	1.037	1.282	0	1.146	1.369	0.292
10	196	2.220	1.313	1.146	0	1.076	1.191	0.279
11	215	2.168	1.149	1.076	0	1.004	1.168	0.268
12	141	2.383	0.646	1.004	1	1.051	1.189	0.257
13	34	2.145	1.071	1.051	0	0.987	1.150	0.249
14	224	2.401	0.614	0.987	1	1.027	1.187	0.240
15	186	2.407	0.551	1.027	1	1.058	1.061	0.233
16	131	2.179	1.382	1.058	0	1.021	1.051	0.227
17	86	2.045	0.983	1.021	0	0.969	1.044	0.221
18	187	2.043	1.022	0.969	0	0.924	1.040	0.216
19	133	2.082	0.632	0.924	1	0.957	0.989	0.211
20	138	1.940	0.919	0.957	1	0.997	0.939	0.207
21	161	2.189	1.428	0.997	1	1.062	0.966	0.203
22	172	2.004	1.219	1.062	1	1.107	0.980	0.199
23	129	1.912	1.005	1.107	0	1.067	0.905	0.195
24	205	1.914	0.932	1.067	1	1.098	0.901	0.192

Sympson [1997]; McBride and Martin [1983]; Stocking and Lewis [1998]; Stocking and Lewis [2000]; van der Linden and Veldkamp [2004]; and Way [1998].) In some cases, for test security concerns the first few items may be randomly selected from item sets; these sets contain items with similar characteristics. In general, examinees are not permitted to return to items and change answers nor to omit items. Therefore, additional implementation concerns include whether examinees should be permitted to omit items, revisit answered items, or mark items for review, or change answers to administered items; the handling of examinees who have been unable to finish; the handling of nonconvergence with MLE; and item pool characteristics (e.g., information distribution, size, etc.). There are variants of the item information approach for selecting items that may be considered for a CAT implementation. For more information on CAT, as well as a discussion of some of these issues, see Drasgow and Olson-Buchanan (1999), Parshall, Spray, Kalohn, and Davey (2002), Reckase (1989), Sands, Waters, and McBride (1997), van der Linden and Glas (2000), van der Linden and Glas (2010), and Wainer et al. (2000), as well as the International Association for Computerized Adaptive Testing (IACAT) at <http://www.iacat.org/>.

NOTES

1. A two-stage test consists of a short routing test that determines which second-stage test is most appropriate for the examinee. The second stage consists of multiple tests that vary in their difficulty, but each test is homogeneous in terms of difficulty. The examinee takes the routing test and, depending on their location estimate, is administered a second-stage test of appropriate difficulty. A variant of this approach uses three stages (cf. Fischer & Pendl, 1980).

The pyramidal test is sometimes called a multistage or multilevel test. Conceptually, the item pool is structured as a binary tree or Pascal's triangle. Each item in the pyramid has a left-hand branch that leads to an easier item and a right-hand branch that leads to a more difficult item. The first item administered is of median difficulty and is at the apex of the pyramid. Each subsequent row (i.e., level) has its items ordered from easy to hard as one progresses across the row from left to right. As an examinee answers items, they progress down through the levels of the binary tree. Which item is administered next depends on the correctness of examinee's response to the currently administered item. If the examinee correctly responds to an item, then they are administered a more difficult item (i.e., the item on the right-hand branch); otherwise the item is an easier item (i.e., the item on the left-hand branch).

For the flexilevel test the item pool is, conceptually, an inverted V, with the item of median difficulty at the apex. The left branch of the V contains items in increasing order of easiness, whereas the right branch contains items in increasing order of difficulty. The items in each branch are numbered, beginning with 1, up to $(L - 1)/2$. The testing procedure begins with the item of median difficulty. The subsequent item selection uses a simple decision rule: If an examinee correctly responds to an item, then they are administered the next lowest numbered right-branch item that has not previously been answered. Conversely, if the examinee incorrectly responds to an item, then the next item administered is the next lowest numbered left-branch item that has not previously been answered. Consequently, if the examinee correctly responds to the first item, then they answer the first item in the right branch. Otherwise, their second item is the first item in the left branch. Assuming the examinee correctly responds to the first two

items, then the next item administered is the second item in the right branch, and so on. If they incorrectly responds to this item, then the fourth item administered is the first item in the left branch, and so on. The test terminates after $(L + 1)/2$ have been administered. For example, if we have a 71-item pool, then the test terminates after the examinee takes $(71 + 1)/2 = 36$ items. The flexilevel may be implemented as a paper-and-pencil test or on a computer (e.g., see de Ayala, Dodd, & Koch, 1990).

2. An alternative way of presenting this termination criterion is in terms of the ability estimate's standard error. Specifically, the adaptive test is terminated once the examinee's ability estimate's standard error is less than an acceptable maximum standard error. For instance, the termination standard error may be set at 0.30; this value is determined from the item pool characteristics. Once an item is administered that results in an examinee's ability estimate's standard error falling at or below 0.30 the adaptive test is terminated. As is the case with maximum information search and selection technique (discussed above), a second criterion based on the maximum number of items that can be administered is also used. In other words, an adaptive test is terminated whenever the examinee's ability estimate's standard error is equal to or less than the criterion value or the maximum number of items administered is reached (whichever occurs first). We call this standard error focused criterion the *target standard error termination* criterion although it could be referred to as a *maximum standard error termination* criterion because it specifies the maximum standard error considered acceptable. (This standard error stopping rule has also been referred to as the minimum standard error criterion.)

3. For a Rasch model-based CAT the item located closest to the examinee's current proficiency estimate is selected because all items have the same maximum item information.

Appendix E

Linear Logistic Test Model (LLTM)

In this book, we treat the individual as a “black box.” However, there are IRT models that can be used to attempt to take into account cognitive processes. One such model is Fischer’s (1973) *linear logistic test model* (LLTM). The LLTM is an extension of the Rasch model designed to incorporate item characteristics that describe performance on the items. These item characteristics can be used to account for variability in the item locations (e.g., why an item is more difficult than another item). Moreover, item characteristics can be cognitive operations/skills required to correctly respond to an item, item features, item response format, instructional conditions item position, and so on (see Embretson, 1984; Kubinger, 2009). In those cases where the item characteristics (e.g., cognitive structure underlying the item set) are theory driven one may consider using the term “explanatory” (e.g., as in explanatory IRT models). However, if there is no theoretical framework, then simply using item characteristics as predictors should not warrant the use of the term “explanatory.” This is particularly true for non-experimental settings where it is impossible to isolate the variables of interest to determine whether the observed relationship is spurious. Moreover, our item characteristics may be proxies for or convenient abstractions of the true cause(s). Nevertheless, these proxies may be useful for making predictions that describe performance on the item. In these cases, it may be prudent to not consider the item characteristics has having an explanatory or *causal* interpretation.

In the LLTM, the item location parameter is constrained to be a linear function of a common set of basic parameters that describe the relevant item characteristics

$$\delta_j = \sum_{s=1}^S q_{js} \eta_s + C, \quad (\text{E.1})$$

where η_s is a basic parameter (i.e., item characteristic) associated with elementary component s , S is the number of components, q_{js} is the weight of component s for item j , and C is a normalization constant equal to

$$C = \frac{-\sum_j \sum_s q_{js} \eta_s}{L}. \quad (\text{E.2})$$

C is the mean of the location estimates prior to dealing with the indeterminacy of the metric (Baker, 1993a). The q_{js} s might be the hypothetical frequencies with which each component influences the solution of each item j , or may simply reflect whether a component is necessary for responding to an item.

Incorporating Equation E.1 into the Rasch model (Equation 2.2) we obtain the LLTM

$$p_j = \frac{\exp(\theta_i - \delta_j)}{1 + \exp(\theta_i - \delta_j)} = \frac{\exp(\theta_i - \sum_{s=1}^S q_{js} \eta_s + C)}{1 + \exp(\theta_i - \sum_{s=1}^S q_{js} \eta_s + C)}. \quad (\text{E.3})$$

As can be seen, the LLTM includes the decomposition of δ_j into a weighted linear composite of parameters that correspond to the components that describe the performance on item j . When the number of components equals the number of items on the instrument, then the LLTM is equivalent to the Rasch model (Embretson, 1984). The LLTM has been extended to rating scale and partial credit data (Fischer & Ponocny, 1994, 1995).

The η_s may correspond to *cognitive operations* required to solve an item, instructional conditions (characterized by their efficacy) experienced by the individual before attempting the item, or the “difficulties” of the cognitive operations (see Embretson, 1984). In short, these components can reflect hypotheses about the psychological structure of the item. Consequently, Equation E.1 shows that the item’s location is the result of the (weighted) cognitive operations required to respond to the item. The values of the η_s s provide information about the relative contribution of a component to the item’s location (Baker, 1993a). In effect, the η_s s are regression weights. The cognitive structure underlying the item set is determined prior to the calibration of the data or as part of the instrument creation process. Additionally, if one has η_s estimates, then one can construct items to reflect one or more basic operations to locate an item in a particular region of the continuum. For more information on cognitive structure/processing and how it can be used for instrument development, see Embretson (1985, 1996), Frederiksen, Mislevy, and Bejar (1993), and Irvine and Kyllonen (2002).

Fischer (1973) presents an example in which to correctly answer mathematics items the examinee must take the first derivative of a series of functions. The first two questions on the 29-item test are

1. $x^3(x^2 + 1)^5$
2. $\frac{x^2 - 3}{5x + 4}$

The basic rules/operations that are necessary to solve the problems are

- | | |
|------------------------------------|--------------|
| 1. Differentiation of a polynomial | 5. $\sin(x)$ |
| 2. Product rule | 6. $\cos(x)$ |
| 3. Quotient rule | 7. $\exp(x)$ |
| 4. Compound functions | 8. $\ln(x)$ |

For instance, for the first question one needs to use rules 1, 2, and 4, whereas for the second question one uses rules 1 and 3. The operations associated with which items are indicated by the corresponding q_{js} s. The q_{js} s for the test may be collected into a S by L weight matrix, \underline{Q} ; $S < L$. For example, for the first two items and the 29th item our \underline{Q} would be

$$\underline{Q} = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ & & & : & & & & \\ 1 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \end{bmatrix},$$

where the columns represent the eight operations and the rows reflect the items. In this case, the entries in \underline{Q} reflect whether the operation is needed (“1”) or not needed (“0”) for a particular item. For example, the first column indicates that rule 1 is used for items 1, 2 and 29 and the second column shows that rule 2 is used only for item 1, but not for item 2 or item 29, and so on.

Fischer’s calibration results showed that for the test

$$\begin{array}{ll} \hat{\eta}_1 = -0.199 & \hat{\eta}_5 = -0.626 \\ \hat{\eta}_2 = 0.061 & \hat{\eta}_6 = -0.759 \\ \hat{\eta}_3 = -0.290 & \hat{\eta}_7 = 0.020 \\ \hat{\eta}_4 = -1.750 & \hat{\eta}_8 = -0.388 \end{array}$$

with $C = 2.066$. (Note the $\hat{\eta}_s$ s are constant across the items.) Therefore, given

$$\delta_j = \sum_{s=1}^S q_{js} \eta_s + C$$

the item location estimates for items 1 and 2 are

$$\hat{\delta}_1 = 1(-0.199) + 1(0.061) + 0(-0.290) + 1(-1.750) + 0(-0.626) + 0(-0.759) + 0(0.02) + 0(-0.388) + 2.066 = 0.178$$

$$\hat{\delta}_2 = 1(-0.199) + 0(0.061) + 1(-0.290) + 0(-1.750) + 0(-0.626) + 0(-0.759) + 0(0.02) + 0(-0.388) + 2.066 = 1.577$$

Although there are specialized programs for obtaining estimates for the LLTM (see Seliger & Fischer, 1994), it is possible to perform the analysis using a two-step approach

that doesn't require specialized programs. This approach produces results that are similar to those of the specialized programs. The first step in this approach is the fitting of the Rasch model (Equation 2.2). The second step involves regressing the resulting $\hat{\delta}_j$ s on the component variables (i.e., $\hat{\delta}_j = b_s q_{js}$; see Embretson and Daniel [2008], Green and Smith [1987]). The resulting regression coefficients are the estimates of the η_s s (i.e., $\hat{\eta}_s = \hat{b}_s$, the effect of characteristic s).

The usefulness of LLTM for a particular instrument depends on the accuracy of the hypothesized cognitive structure underlying the item set (i.e., the \underline{Q} matrix). Baker (1993a) examined the effects of the misspecification of the \underline{Q} matrix. He found that the parameter estimates' accuracy depended on the sparseness of the \underline{Q} matrix as well as the sample size. Even a small degree of misspecification had a large impact on the estimates. He concluded that "because specifying the $\langle \underline{Q} \rangle$ matrix is a judgmental task, it must be done with great care" (p. 209).

Moreover, the LLTM assumes an item's location is perfectly predictable by the weighted item characteristics. However, this assumption may not be tenable and it may be prudent to include a random error term in Equation E.1. Janssen, Schepers, and Peres (2004) present such a model. In Chapter 13 we discuss a LLTM with a random error term. The reader interested in the application of an LLTM-like model would be well served by considering the Janssen et al. (2004) model.

The advantage of using a specialized program for performing an LLTM calibration (e.g., the R package `eRm`; Mair, Hatzinger, & Maier, 2018) is the fit information provided. This is particularly important when one is evaluating different cognitive structures for an item set. That is, because competing theories may lead to alternative cognitive structures for an item, the LLTM may be used for evaluating these competing theoretical explanations. For additional application examples see Embretson (1993), Embretson and Wetzel (1987), Fischer and Formann (1982), and Spada and McGaw (1985).

EXAMPLE OF LLTM CALIBRATION USING `eRm`

Nutrition literacy is defined as ". . . as the degree to which individuals can obtain, process, and understand the basic health (nutrition) information and services they need to make appropriate health (nutrition) decisions" (Silk et al., 2008, p. 4). Zoellner et al. (2011) found relationships between participants' diet quality and their nutrition literacy, age, gender, as well as participation in the Supplemental Nutrition Assistance Program (SNAP). We conceptualize nutrition literacy as a continuous latent variable and proceed through the steps of scale construction to develop a ten-item multiple choice format instrument that measures participants' nutrition literacy; for example, see Boateng, Neilands, Frongillo, Melgar-Quiñonez, and Young (2018). An example item from the scale could be

Butter has lots of fat that can increase cholesterol.

- (a) monounsaturated
- (b) polyunsaturated

- (c) saturated
- (d) trans

(This item is similar to one found on Diamond’s [2004] Nutrition Literacy Scale.) Our items are created involving two components. The first component reflects whether the item included technical terminology (e.g., terms such as trans fat, saturated fat, polyunsaturated fat), whereas the second component has to do with food safety (e.g., temperatures, food storage, food handling). Our weight (design) matrix ($\underline{\mathbf{Q}}$) is $L \times S$

$$\underline{\mathbf{Q}} = \begin{bmatrix} 1 & 0 \\ 1 & 0 \\ 1 & 1 \\ 1 & 1 \\ 1 & 1 \\ 0 & 1 \\ 0 & 1 \\ 1 & 1 \\ 0 & 1 \\ 0 & 1 \end{bmatrix},$$

where the entries in $\underline{\mathbf{Q}}$ reflect whether the item involves the facet (“1”) or not (“0”). For instance, items 1 and 2 include just technical terminology (i.e., $q_{11} = q_{21} = 1$, $q_{12} = q_{22} = 0$), items 3–5 include both technical terminology and food safety (e.g., $q_{31} = 1$, $q_{32} = 1$), and item 6 reflects just food safety (i.e., $q_{61} = 0$, $q_{62} = 1$), and so on. Our scoring of the responses results in higher scores reflecting greater literacy than do lower scores. We collect data from 1000 individuals.

To calibrate our data we use the R package `eRm` (also see Mair and Hatzinger [2007]); `eRm` will also fit other models, such as, the partial credit and rating scale models using CMLE.¹ Our data file, `LLTM.dat`, consists of 1000 binary responses occupying columns 9–18. The responses are read using the `read.fortran` function to perform a fixed formatted read with the FORTRAN format statement of `10I1` (i.e., 10 integers each occupying one column; see FORTRAN Formats below for more information). Table E.1 shows our session.

After importing the data we examine the data frame (`lltmdat`) contents to verify the data were read correctly. Subsequently, we remove the case id variable (`id`) from `lltmdat`. We first perform a simple Rasch calibration followed by the LLTM calibration.

Our simple Rasch model calibration uses the `RM` function (`rasch = RM(lltmdat)`). The `summary(rasch)` function provides estimation information followed by our item parameter estimates. Because our problem required 18 iterations and 50 iterations is the default maximum we know that we have a converged solution. The item location estimates are given on a difficulty scale ($\hat{\delta}_j$) and an easiness scale ($\hat{\delta}_j^E$)

TABLE E.1. Rasch and LLTM Calibrations in R Using eRm

```

> library(eRm)
> lltmdat=read.fortran("LLTM.dat",c("1I8","10I1"))           # read fixed format data

> # replace default variables names (i.e., V1, ..., V11) with meaningful names
> names(lltmdat)= c('id', 'i1','i2','i3','i4','i5','i6','i7','i8','i9','i10')
> head(lltmdat,6)
      id i1 i2 i3 i4 i5 i6 i7 i8 i9 i10
1      1  1  1  0  1  0  0  1  0  0  0
2      2  1  1  0  0  1  1  0  0  0  0
3      3  1  1  1  1  0  0  0  0  0  0
4      4  1  1  1  0  0  1  0  0  0  0
5      5  1  1  0  1  1  1  0  0  0  0
6      6  1  1  1  0  0  0  0  0  0  0

> tail(lltmdat,4)
      id i1 i2 i3 i4 i5 i6 i7 i8 i9 i10
997    997  1  1  0  1  0  1  0  1  1  0
998    998  1  1  0  0  1  0  0  0  0  0
999    999  1  1  1  1  1  1  0  0  0  0
1000  1000  1  1  0  0  0  0  1  0  0  0

> lltmdat=within(lltmdat, rm(id))                               # remove id from data frame

> # Rasch model calibration
> rasch=RM(lltmdat)
> summary(rasch)

Results of RM estimation:

Call:  RM(X = lltmdat)

Conditional log-likelihood: -3539.35
Number of iterations: 18
Number of parameters: 9

Item (Category) Difficulty Parameters (eta): with 0.95 CI:
  Estimate Std. Error lower CI upper CI
i2    -1.504    0.080   -1.660   -1.348
i3    -1.091    0.073   -1.233   -0.948
i4    -0.437    0.066   -0.567   -0.307
i5    -0.078    0.065   -0.206    0.050
i6     0.306    0.066    0.178    0.435
i7     0.426    0.066    0.296    0.555
i8     0.889    0.069    0.753    1.024
i9     1.557    0.078    1.404    1.710
i10    2.088    0.090    1.912    2.264

Item Easiness Parameters (beta) with 0.95 CI:
  Estimate Std. Error lower CI upper CI
beta i1     2.155    0.096    1.967    2.344
beta i2     1.504    0.080    1.348    1.660
beta i3     1.091    0.073    0.948    1.233
beta i4     0.437    0.066    0.307    0.567
beta i5     0.078    0.065   -0.050    0.206
beta i6    -0.306    0.066   -0.435   -0.178
beta i7    -0.426    0.066   -0.555   -0.296

```

(continued)

TABLE E.1. (continued)

```

beta i8    -0.889    0.069   -1.024   -0.753
beta i9    -1.557    0.078   -1.710   -1.404
beta i10   -2.088    0.090   -2.264   -1.912

> raschfit=LRtest(rasch,splitcr="median")
> raschfit      # show fit stats
  Andersen LR-test:
  LR-value: 10.856
  Chi-square df: 9
  p-value: 0.286

> # the following plot produces Fig. E1: "Wright Map", Item-Person map
> plotPImap(rasch)

> # the following plot produces Fig. E2: Empirical & Predicted IRFs
> plotICC(rasch,item.subset = 1:1, empICC = list("raw"), mplot = TRUE,
legpos = FALSE, ask = FALSE)

> # the following plot produces Fig E3: graphical check for misfitting items
> plotGOF(raschfit,conf=list(ia=FALSE,col="black"))

> # LLTM calibration
> Q=matrix(c(1,1,1,1,1,0,0,1,0,0,0,0,1,1,1,1,1,1,1,1),ncol=2) # create Q matrix
> lltm=LLTM(lltmdat,Q) # LLTM calibration
> summary(lltm)

Results of LLTM estimation:

Call: LLTM(X = lltmdat, W = Q)

Conditional log-likelihood: -3947.843
Number of iterations: 8
Number of parameters: 2

Basic Parameters eta with 0.95 CI:
      Estimate Std. Error lower CI upper CI
eta 1    1.129    0.050    1.032    1.227
eta 2   -1.597    0.073   -1.740   -1.454

Item Easiness Parameters (beta) with 0.95 CI:
      Estimate Std. Error lower CI upper CI
beta i1    1.129    0.050    1.032    1.227
beta i2    1.129    0.050    1.032    1.227
beta i3   -0.467    0.099   -0.662   -0.273
beta i4   -0.467    0.099   -0.662   -0.273
beta i5   -0.467    0.099   -0.662   -0.273
beta i6   -1.597    0.073   -1.740   -1.454
beta i7   -1.597    0.073   -1.740   -1.454
beta i8   -0.467    0.099   -0.662   -0.273
beta i9   -1.597    0.073   -1.740   -1.454
beta i10  -1.597    0.073   -1.740   -1.454

> print((GsqrR=(-2*lltm$loglik)) # done for pedagogical reason, G sqr reduced model
[1] 7895.687

> print((GsqrF=(-2*rasch$loglik)) # done for pedagogical reason, G sqr full model
[1] 7078.7

```

(continued)

TABLE E.1. (continued)

```

> # Calculate G square
> print((Gsqr=GsqrR-GsqrF)) # or more concisely: (-2*lltm$loglik)-
  (-2*rasch$loglik)
  [1] 816.9862

> print((GsqrDF=rasch$npar-lltm$npar)          # obtain dfs
> GsqrDF
  [1] 7

> qchisq(.95,df=GsqrDF)                       # obtain critical value
  [1] 14.06714

> cor(lltm$betapar,rasch$betapar)             # obtaining corr betw Rasch & LLTM
  [1] 0.8478063

> # the following plot produces Fig. E4
> plot(lltm$betapar,rasch$betapar,xlim=c(-3,3),ylim=c(-3,3),ylab="Rasch Easiness
  Location",xlab="LLTM Easiness Location")

> PersonEstLLTM=person.parameter(lltm)        # abstract MLE of person location est
> PersonEstLLTM

Person Parameters:

Raw Score  Estimate Std.Error
0 -3.0247922      NA
1 -2.0120574  1.1165795
2 -1.0679091  0.8683352
3 -0.4063419  0.7692289
4  0.1438121  0.7197200
5  0.6443882  0.6994835
6  1.1340521  0.7043722
7  1.6511136  0.7398455
8  2.2574839  0.8299285
9  3.1286312  1.0808306
10 4.0648431      NA

> summary(PersonEstLLTM)

Estimation of Ability Parameters

Collapsed log-likelihood: -43.13987
Number of iterations: 10
Number of parameters: 9

ML estimated ability parameters (without spline interpolated values):
      Estimate Std. Err.    2.5 %    97.5 %
theta P1    0.1438121  0.7197200 -1.2668133  1.5544374
theta P2    0.1438121  0.7197200 -1.2668133  1.5544374
theta P3    0.1438121  0.7197200 -1.2668133  1.5544374
theta P4    0.1438121  0.7197200 -1.2668133  1.5544374
theta P5    0.6443882  0.6994835 -0.7265742  2.0153506
      :
theta P999  1.1340521  0.7043722 -0.2464921  2.5145963
theta P1000 -0.4063419  0.7692289 -1.9140028  1.1013191

> plot(PersonEstLLTM)          # produces Fig. E5: plot estimates against X

```

) where $\hat{\delta}_j = -\hat{\delta}_j^E$. Examining the Difficulty Parameters table shows that item 1 is absent. To obtain its value we can either take the negative of item 1's $\hat{\delta}_1^E$ (beta i1) so that $\hat{\delta}_1 = -1 * \hat{\delta}_1^E = -1 * 2.155 = -2.155$ or take the negative of the sum of the remaining items $\hat{\delta}_1 = -\sum_{j=2}^L \hat{\delta}_j = -2.156$ (rounding error). Accordingly, we have $\hat{\delta}_1 = -2.155$, $\hat{\delta}_2 = -1.504$, $\hat{\delta}_3 = -1.091$, etc. or on the easiness scale (Easiness Parameters) we have $\hat{\delta}_1^E = 2.155$, $\hat{\delta}_2^E = 1.504$, $\hat{\delta}_3^E = 1.091$, and so on. (To obtain person location estimates we would use `PersonEstRasch= person.parameter(rasch)`.) To compare how our respondents' distribution relates to our scale's item distribution we request a (Wright) item-person map (`plotPImap(rasch)`; Figure E.1). This plot shows each item's $\hat{\delta}_j^E$ as a dot with respect to the item label (i.e., i1, i2, ..., i10) and the distribution of respondents (top panel: Person Parameter Distribution). As can be seen, our scale measures across the continuum and maps to almost all of our (except the extreme) respondents.

We use the LRtest with a median split (criterion) for the observed scores to determine if the data fit the Rasch model. Because the likelihood ratio test's p -value is greater than 0.05 it appears the data are consistent with the Rasch model. From the Conditional log-likelihood line we have $\ln L = -3539.35$ so that our G_F^2

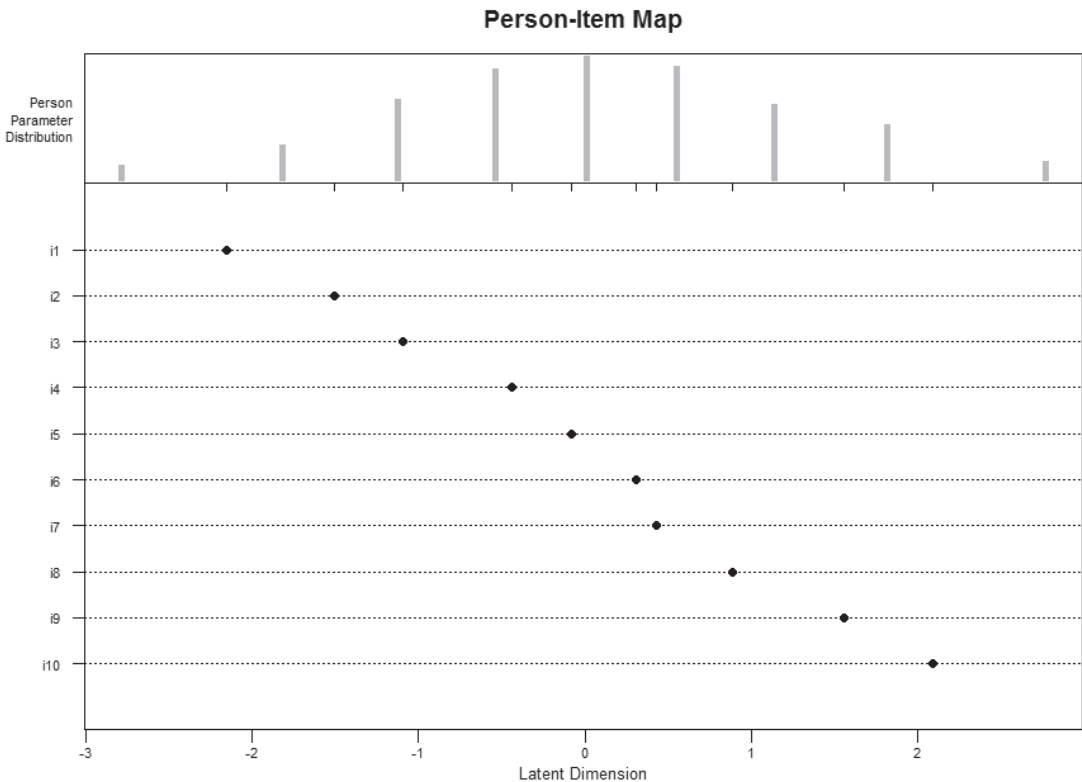


FIGURE E.1. Item-Person map (Wright map).

$= -2\ln L = -2(-3539.35) = 7078.7$ for $L - 1 = 9$ location parameters; we will use this below.

In addition to this model-level fit, we can graphically examine our item-level fit using two graphical techniques. The first is the `plotICC` function. As an example, for item 1 we use `plotICC(rasch, item.subset = 1:1, empICC = list("raw"), etc.)` with the `empICC` argument (Figure E.2). As can be seen, we have good agreement between the predicted IRF (line) and the empirical IRF (circles).

The second graphical approach is the goodness of fit (GOF) plot. For the GOF the respondents are first split into two groups based on the mean and then the item location estimates from each group are plotted against one another (Figure E.3). Accordingly, the GOF plot simultaneously examines all the items rather than the item-wise approach of the empirical/predicted plot. We use the `plotGOF` function with confidence ellipses (`plotGOF(raschfit, conf=list(ia=FALSE,col="black"))`). Ideally, the items (the small circles) would fall on the identity (diagonal) line and thereby indicate perfect fit. Items appearing further away from the line reflect poorer fitting items. However, because of estimation error we do not expect that all items would fall on the identity line even in a perfect fit situation. The use of confidence ellipses takes into account estimation error. Thus, if an item's ellipse includes ("covers") the identity line we consider the item to exhibit fit. Because all of our items' confidence ellipses cover the line we conclude that we have item-level fit as well as additional evidence of model-data fit.

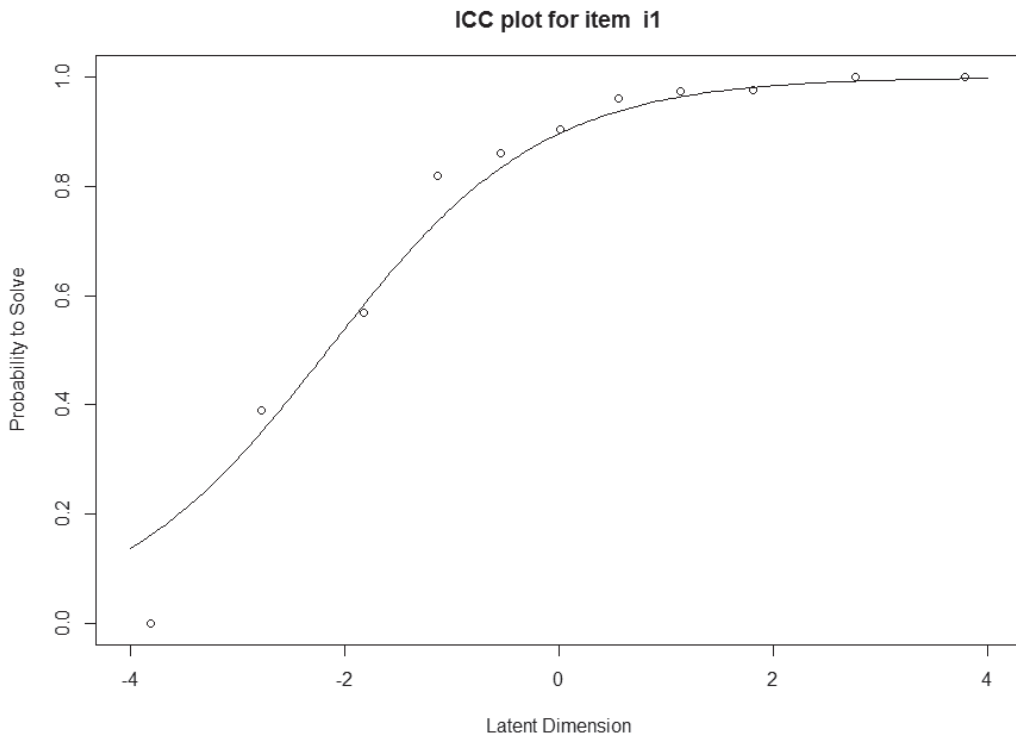


FIGURE E.2. Rasch empirical and predicted IRFs.

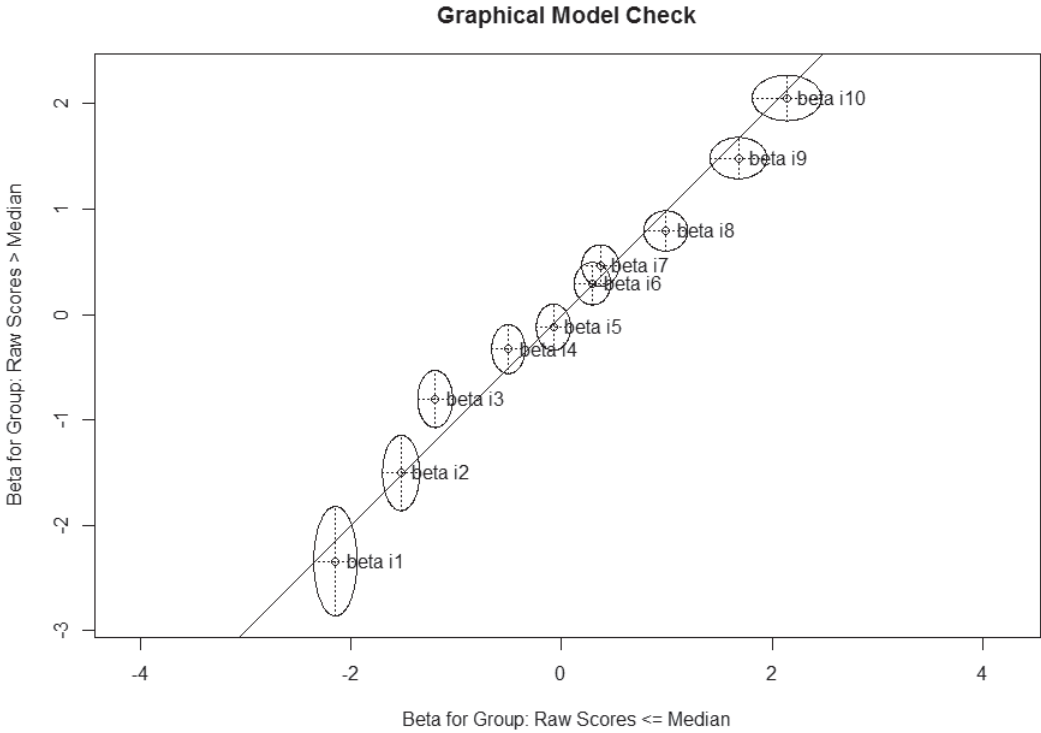


FIGURE E.3. Split half plot.

Given our model-data fit we proceed to perform our LLTM analysis by creating the Q matrix ($Q = \text{matrix}(c(1, 1, 1, 1, 1, 0, 0, \dots, 1), ncol=2)$) and providing it along with the data frame as arguments to the LLTM function ($\text{LLTM}(\text{lltmdat}, Q)$). As above, we use the `summary` function to see our results. Our LLTM's $G_R^2 = -2\ln L = -2(-3947.843) = 7895.687$ with two basic parameters. Because the LLTM is a restricted (i.e., reduced) version of the Rasch model we can use the likelihood ratio (LR) statistic to determine whether the additional complexity of the Rasch model leads to a significant improvement in fit over the LLTM. Our statistic is

$$\Delta G^2 = -2 \ln \left(\frac{L_R}{L_F} \right) = (-2 \ln L_R) - (-2 \ln L_F) = G_R^2 - G_F^2,$$

where G_R^2 is $-2\ln L$ for our reduced model (LLTM) and G_F^2 represents the $-2\ln L$ our full model (Rasch). Assuming the Rasch model holds then our statistic is asymptotically distributed as a chi square with $df = L - 1 - S$ (Fischer & Formann, 1982); that is, the null hypothesis is that the data are consistent with the LLTM and the Rasch model is the alternative hypothesis. (We can perform these calculations in R by $G_{\text{sqrR}} = (-2 * \text{lltm}\$ \text{loglik})$ to obtain G_R^2 and $G_{\text{sqrF}} = (-2 * \text{rasch}\$ \text{loglik})$ to obtain G_F^2 .) Our ΔG^2 ($G_{\text{sqr}} = G_{\text{sqrR}} - G_{\text{sqrF}}$)

$$\Delta G^2 = 7895.687 - 7078.7 = 816.9862$$

Because our ΔG^2 exceeds a critical value of 14.06714 with $df = 10 - 1 - 2 = 7$ the Rasch model fits significantly better than the LLTM. Given that the LLTM model has only two parameters (i.e., η_1, η_2) it is not surprising that its $-2\ln L$ is larger than with the Rasch model.

As is typically the case, the Rasch model-data fit is comparatively better than with the LLTM (see Fischer & Formann, 1982). Fischer and Formann (1982) suggest calculating the correlation between the Rasch model's and LLTM's item location estimates. A high correlation reflects that the \underline{Q} matrix is accounting for the variability observed in the item locations. For our example, the correlation is 0.8478 (`cor(lltm$betapar, rasch$betapar)`). Therefore, approximately 72% ($r^2 = 0.8478^2 = 0.7188$) of the variability in the Rasch $\hat{\delta}_j^E$ s is shared with those of the LLTM. Figure E.4 shows that the correlation's magnitude is an accurate reflection of the linear relationship between the two sets of item location estimates. Thus, our \underline{Q} provides a reasonable reflection of the nutrition literacy item locations.

Our basic parameter estimates are $\hat{\eta}_1 = 1.129$ and $\hat{\eta}_2 = -1.597$ with item location estimates on the easiness scale of $\hat{\delta}_1^E = 1.129, \hat{\delta}_2^E = 1.129, \hat{\delta}_3^E = -0.467$, and so on. Our basic parameters indicate the technical terminology component is easier than food safety component. Thus, any items that have only the technical terminology characteristic will tend to be easier than those items that have only the food safety characteristic.

To obtain our person location estimates we use the `person.parameter` function and store the estimates in the `PersonEstLLTM` object (`PersonEstLLTM = person.parameter(lltm)`). The estimate column contains our $\hat{\theta}$ s. Because the

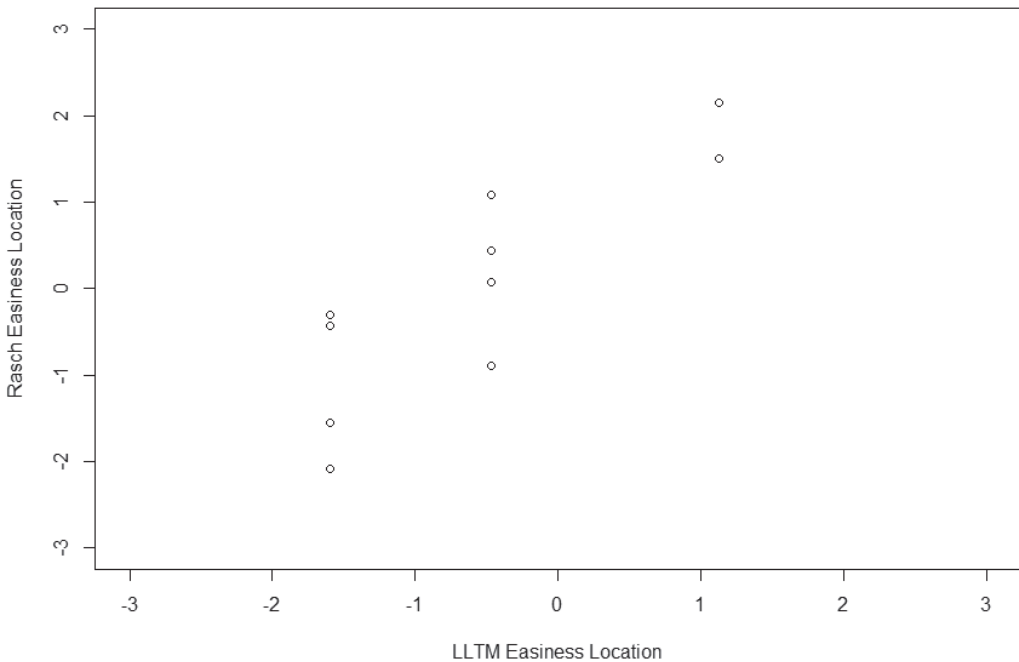


FIGURE E.4. Scatterplot of Rasch $\hat{\delta}_j^E$ s against LLTM $\hat{\delta}_j^E$ s.

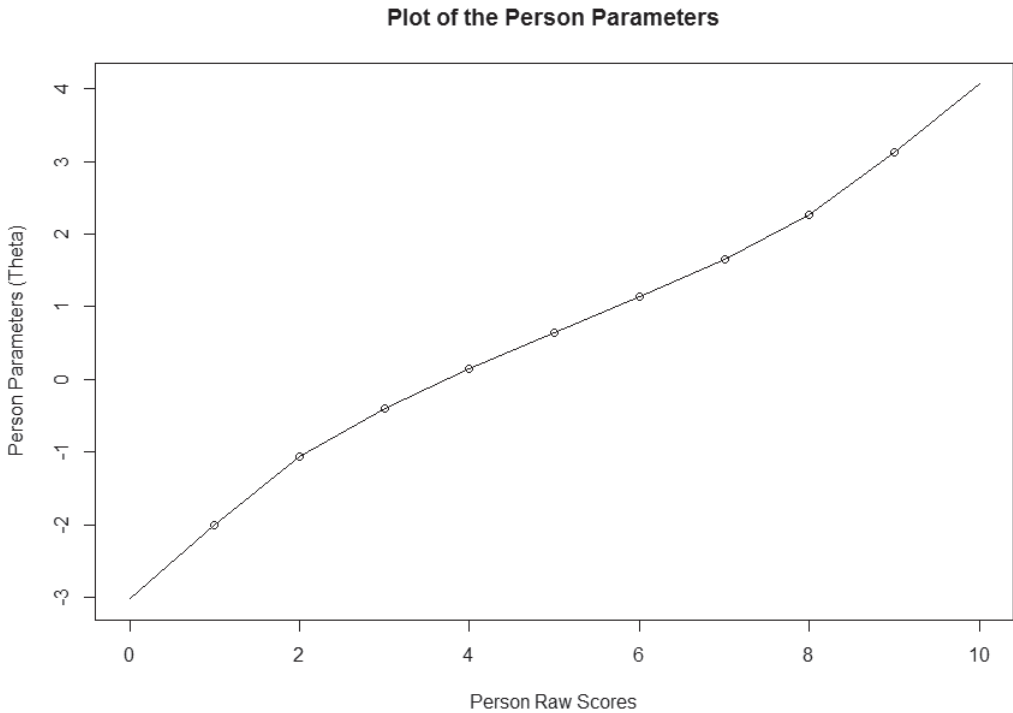


FIGURE E.5. Scatterplot of $\hat{\theta}$ against X .

LLTM belongs to the Rasch family of models there are $L - 1$ possible $\hat{\theta}$ s. Displaying the contents of `PersonEstLLTM` shows the possible values (e.g., $X = 1$ corresponds to $\hat{\theta} = -2.0120574$, $s_e(\hat{\theta}) = 1.1155795$). In Figure E.5 we have a plot of our 9 possible $\hat{\theta}$ s against their corresponding observed scores. To obtain the $\hat{\theta}$ for each case we use the `summary` function (`summary(PersonEstLLTM)`). As can be seen, the first four respondents are estimated to be located at 0.1438121 ($s_e(\hat{\theta}) = 0.7197200$) and are approximately average in nutrition literacy.

NOTES

1. An alternative to using `eRm` is to use the regression approach discussed above or SAS procedure `glimmix`. Tables E.2 and E.3 shows the command file and part of the corresponding output. The `Solutions for Fixed Effects` table shows that $\eta_1 = 1.3985$ and $\eta_2 = -1.0746$. To obtain the item location estimates one applies $\sum q_{js}\eta_s$. For example, $\hat{\delta}_1^E = q_{11}\eta_1 + q_{12}\eta_2 = 1(1.3985) + 0(-1.0746) = 1.3985$, $\hat{\delta}_3^E = q_{31}\eta_1 + q_{32}\eta_2 = 1(1.3985) + 1(-1.0746) = 0.3239$, and so on. These estimates correlate 0.981 with those of `eRm`.

TABLE E.3. proc glimmix for Rasch Calibration

The command file:

```
proc glimmix data=stacked_data noclprint method=laplace;
  title ""Rasch" fixed items & random P; w/o intercept ";
  class item person;
  model x(event='1')= item /noit dist=binary link=logit solution;
  random intercept/subject=person solution;
  output out=fit_statistics2 resid(ilink)=residual2
         variance(ilink)=variance2;
run;
```

Abridged output:

Covariance Parameter Estimates			
Cov Parm	Subject	Estimate	Standard Error
Intercept	person	0.5463	0.05629

Solutions for Fixed Effects						
Effect	Item	Estimate	Standard Error	DF	t Value	Pr > t
item	1	2.2517	0.1062	8991	21.19	<.0001
item	2	1.5975	0.08780	8991	18.19	<.0001
item	3	1.1825	0.08005	8991	14.77	<.0001
item	4	0.5239	0.07268	8991	7.19	<.0001
item	5	0.1636	0.07135	8991	2.29	0.0219
item	6	-0.2213	0.07149	8991	-3.10	0.0020
item	7	-0.3405	0.07190	8991	-4.74	<.0001
item	8	-0.8009	0.07521	8991	-10.65	<.0001
item	9	-1.4629	0.08504	8991	-17.20	<.0001
item	10	-1.9860	0.09780	8991	-20.31	<.0001

Solution for Random Effects						
Effect	Subject	Estimate	Std Err Pred	DF	t Value	Pr > t
Intercept	person 1	-0.3128	0.5229	8991	-0.60	0.5497
Intercept	person 2	-0.3128	0.5229	8991	-0.60	0.5497
Intercept	person 3	-0.3128	0.5229	8991	-0.60	0.5497
Intercept	person 4	-0.3128	0.5229	8991	-0.60	0.5497
Intercept	person 5	-0.04071	0.5215	8991	-0.08	0.9378
Intercept	person 999	0.2313	0.5226	8991	0.44	0.6581
Intercept	person 1000	-0.5871	0.5267	8991	-1.11	0.2650

TABLE E.3. proc glimmix for Rasch Calibration

The command file:

```
proc glimmix data=stacked_data noclprint method=laplace;
  title ""Rasch" fixed items & random P; w/o intercept ";
  class item person;
  model x(event='1')= item /noit dist=binary link=logit solution;
  random intercept/subject=person solution;
  output out=fit_statistics2 resid(ilink)=residual2
         variance(ilink)=variance2;
run;
```

Abridged output:

Covariance Parameter Estimates			
Cov Parm	Subject	Estimate	Standard Error
Intercept	person	0.5463	0.05629

Solutions for Fixed Effects						
Effect	Item	Estimate	Standard Error	DF	t Value	Pr > t
item	1	2.2517	0.1062	8991	21.19	<.0001
item	2	1.5975	0.08780	8991	18.19	<.0001
item	3	1.1825	0.08005	8991	14.77	<.0001
item	4	0.5239	0.07268	8991	7.19	<.0001
item	5	0.1636	0.07135	8991	2.29	0.0219
item	6	-0.2213	0.07149	8991	-3.10	0.0020
item	7	-0.3405	0.07190	8991	-4.74	<.0001
item	8	-0.8009	0.07521	8991	-10.65	<.0001
item	9	-1.4629	0.08504	8991	-17.20	<.0001
item	10	-1.9860	0.09780	8991	-20.31	<.0001

Solution for Random Effects						
Effect	Subject	Estimate	Std Err Pred	DF	t Value	Pr > t
Intercept	person 1	-0.3128	0.5229	8991	-0.60	0.5497
Intercept	person 2	-0.3128	0.5229	8991	-0.60	0.5497
Intercept	person 3	-0.3128	0.5229	8991	-0.60	0.5497
Intercept	person 4	-0.3128	0.5229	8991	-0.60	0.5497
Intercept	person 5	-0.04071	0.5215	8991	-0.08	0.9378
Intercept	person 999	0.2313	0.5226	8991	0.44	0.6581
Intercept	person 1000	-0.5871	0.5267	8991	-1.11	0.2650

Appendix F

Mixture Models

Mixture IRT models address situations involving a mixture of latent subpopulations. These subpopulations are qualitatively different but within which a measurement model based on a continuous latent variable holds. In this modeling framework, one can characterize respondents by both their location on a continuous latent variable as well as by their latent subpopulation (class) membership. Although non Rasch models can be used in mixture modeling our presentation will focus on the Rasch mixture model.

Mixture models may be applicable when the Rasch/IPL model may not fit the population of interest, but does fit subpopulations. For instance, one may observe varying item discrimination because the calibration sample reflects a mixture of (homogeneous) subpopulations. In this case, when we apply the Rasch/IPL model to each subpopulation we might obtain model–data fit. As such, a mixture Rasch provides an alternative approach for addressing the lack of model–data fit. In addition, mixture models may be useful with multidimensional data. As mentioned in the first chapter, in some cases the latent space may be conceptualized as consisting of both latent classes and latent continua. This section first presents a general introduction to latent class analysis (LCA) and then introduces the integration of LCA with IRT; LCA is briefly introduced in Chapter 1.

LATENT CLASS ANALYSIS

In contrast to IRT’s assumption of a continuous latent variable, in LCA the latent construct is assumed to be categorical. Specifically, the latent construct consists of a set of mutually exclusive and exhaustive *latent classes* that account for the manifest relationships between any two or more items on an instrument (Stouffer, 1950). That is, we believe that our sample consists of a mixture of qualitatively different types of persons. The latent classes are subpopulations of individuals that are homogeneous with respect to the variable of interest. These subpopulations are not manifest groupings (e.g., high-versus low-proficiency groups, gender, ethnicity), but are unobserved. Moreover, these latent classes may or may not be ordered. LCA may be used with either dichotomous or

polytomous response data that are assumed to be a manifestation of two or more latent classes of respondents.

As an example, assume we wish to measure anxiety by administering a scale, such as, the Taylor Manifest Anxiety Scale (Taylor, 1953). We conceptualize our latent variable anxiety as categorical in nature. The latent class analysis of our response data might lead us to classify respondents into qualitatively different latent groups so, for example, one class may be interpreted as representing individuals with incapacitating anxiety and a second class reflecting respondents with transient anxiety. Because LCA involves comparing individuals in terms of their latent class memberships, rather than their locations on a continuous latent variable, we talk about the respondents in terms of their similarities within a class or their dissimilarities across classes, but not in terms of degree of their anxiety.

LCA characterizes an item in terms of the probability of a randomly drawn individual from a particular latent class providing a particular response. To develop the basic latent class model assume that we administer an instrument developed to measure general anxiety. For simplicity let the items be scored to produce dichotomous responses. Each item j on the instrument is characterized by a conditional item probability ($\pi_{j\nu}$) in each of G latent classes ($\nu = 1 \dots G$). This conditional item probability specifies the probability that a respondent in latent class ν obtains a response of 1 on item j . For instance, in a proficiency context the conditional item probabilities would reflect the item's difficulty for a latent class. Additionally, each latent class ν is characterized by its latent class proportion (π_ν). In other words, π_ν is the proportion of respondents in the sample that belong to latent class ν . (Note that a double script on π indicates a conditional item probability, whereas a single subscript reflects a latent class proportion.) Because the latent class set is exhaustive the sum of all the latent class proportions is 1 ($\sum_{\nu=1}^G \pi_\nu = 1$). The latent class proportions and conditional item probabilities are the person and item parameters estimated in LCA.

To obtain a basic model we define the probability of a response vector \underline{x}_i for a latent class ν as

$$p(\underline{x}_i | \nu) = \prod_{j=1}^L \pi_{j\nu}^{x_{ij}} (1 - \pi_{j\nu})^{(1-x_{ij})}, \quad (\text{F.1})$$

where $p(\underline{x}_i | \nu)$ is the *conditional* probability for respondent i 's response vector \underline{x}_i and $\pi_{j\nu}$ is the conditional item probability for item j in latent class ν . Stated in words, Equation F.1 specifies the chance of observing the response vector \underline{x}_i given that respondent i is in latent class ν . These probabilities are conditional on the respondent's latent class membership and are a function of the each item's conditional item probability for the class ($\pi_{j\nu}$) and respondent i 's response to the item. Because LCA assumes conditional independence Equation F.1 is an application of the multiplication rule for independent events. Consequently, item responses are assumed independent within a latent class (i.e., conditional on latent class membership).

Equation F.1 is the basis for predicting to which latent class a respondent is most

likely to belong. To predict a respondent's class membership given their response vector we need to know the probabilities for each latent class. That is, what is the probability of having observed respondent i 's response vector if they belong to latent class 1, what is the probability of the observed response vector if they are in latent class 2, and so on. To obtain this overall probability for respondent i 's response vector (i.e., *irrespective* of latent class) we need to take into consideration the latent class sizes (i.e., π_ν). These latent class proportions serve to weight the conditional probabilities of respondent i 's response vector. Therefore,

$$p(\underline{x}_i) = \sum_{\nu=1}^G \pi_\nu \cdot p(\underline{x}_i | \nu) = \sum_{\nu=1}^G \pi_\nu \left[\prod_{j=1}^L \pi_{j\nu}^{x_{ij}} (1 - \pi_{j\nu})^{(1-x_{ij})} \right], \quad (\text{F.2})$$

where $p(\underline{x}_i)$ is the *unconditional* probability for individual i 's response vector \underline{x}_i , $p(\underline{x}_i | \nu)$ is the conditional probability for respondent i 's response vector \underline{x}_i given by Equation F.1, and the other terms are defined above. Equation F.2 tells us the probability of observing individual i 's responses regardless of their class membership, whereas Equation F.1 tells us the probability of observing respondent i 's responses given a latent class. By way of analogy, Equation F.1 tells us the probability of randomly selecting a person who is left-handed from a particular gender, whereas Equation F.2 tells us the probability of randomly selecting a left-handed person regardless of their gender.

As mentioned above, LCA involves comparing individuals in terms of their latent class memberships. Accordingly, we need to predict latent class membership for each member of our sample. To make this prediction we determine to which latent class each respondent has the highest probability of belonging. This probability is determined by using Equations F.1 and F.2 as well as Bayes' Theorem to obtain Equation F.3. Equation F.3 gives us the (posterior) probability of membership in latent class ν given person i 's responses

$$p(\nu | \underline{x}_i) = \frac{p(\underline{x}_i | \nu) \cdot \pi_\nu}{\sum_{\nu=1}^G \pi_\nu \left[\prod_{j=1}^L \pi_{j\nu}^{x_{ij}} (1 - \pi_{j\nu})^{(1-x_{ij})} \right]}. \quad (\text{F.3})$$

Equation F.3 is calculated for each latent class given a respondent's responses. Each respondent is assigned to whichever latent class has the largest membership probability. Because we calculate the probability of a given response pattern for each latent class all persons with the same response pattern are classified in the same latent class.

We can extend these ideas to an instrument with L items and m possible (i.e., polytomous) responses to obtain the probability of an individual's particular response pattern \underline{x} given their membership in latent class ν by

$$p(\underline{x} | \nu) = \prod_{j=1}^L \prod_{k=1}^m (\pi_{k|j\nu})^{x_{kj}}, \quad (\text{F.4})$$

where the $\pi_{k|j\nu}$ is the conditional probability and reflects the conditional probability

of the observed response k given item j and the individual's membership in latent class ν . The exponent x_{kj} equals 1 if and only if $x_j = k$. The unconditional probability of the observed response vector \underline{x} is the weighted average of the conditional probabilities across the G latent classes

$$p(\underline{x}) = \sum_{\nu=1}^G \pi_{\nu} p(\underline{x} | \nu). \quad (\text{F.5})$$

One may conceive of a situation where, with a large number of *ordered* latent classes, there would be little difference between conceptualizing the latent variable as continuous or as categorical. In point of fact, Lindsay, Clogg, and Grego (1991) show that a latent class model with $G \geq (L + 1)/2$ latent classes gives the same estimates of item parameters as the Rasch model; also see Masters (1985).¹ For example, for a data set with $L = 4$ items, a latent class model with at least three latent classes would provide item characterizations “equivalent” to those of the Rasch model that uses only item location parameters. Both Clogg (1995) and Dayton (1998) provide readable introductions to LCA.

MIXTURE RASCH MODEL

Some empirical situations involve a mixture of latent subpopulations such that there are qualitative differences between the subgroups but within each subpopulation there is a continuous latent variable. To provide some context assume we develop a temperament scale that uses a force-choice format with two alternatives that represent opposite poles (e.g., extraversion or introversion). For example, one item's stem might be “When I feel drained from a long work week”: (a) “I like to spend time by myself to ‘recharge.’” or (b) “I like to get together with friends and go out to ‘recharge.’” The “a” response is coded as a 1 (i.e., the introversion choice is coded as 1), 0 otherwise. Thus, the right side of the E-I (the theta scale) continuum would reflect introversion and the left side would be associated with extraversion. However, if our sample reflected a mixture of latent subpopulations that differ with respect to self-disclosure (one subpopulation reflects “non-discriminatory self-disclosure,” whereas the other reflects “selective self-disclosure”), then these two classes could interact with how individuals respond on the temperament E-I scale. Stated another way, an individual's propensity to endorse one choice over the other is affected by whether they are a member of the “selective self-disclosure” class or the “non-discriminatory self-disclosure” class. We can model this situation by using mixture IRT—an integration of IRT and LCA.

In the current context of mixture models, assume the data consist of a mixture of subpopulations that are different from one another in *kind*. Each of these subpopulations might be best represented as a latent class of individuals. Within each of these classes there is a latent continuum on which the individuals within the class may be placed and ordered. That is, there are both qualitative *and* quantitative differences among individuals with respect to the same items (e.g., Rost, 1990). In effect, we have IRT model-data fit within each class, but not across the classes. Therefore, each item has parameter estimates for each class. (In the simplest case, there is only one latent class and the calibra-

tion sample contains only members from a single class. As a result, one has model-data fit with a simple IRT model.) The gist of the application of mixture models is to “unmix” the data into a set of classes, determine the classes’ sizes, assign individuals to classes, and estimate the person and item parameters for each class.

A mixture model involves parameters having to do with latent classes and IRT item and person parameters. As mentioned above these latent classes are mutually exclusive and jointly exhaustive. The symbolic representation of the mixture model for person i and item j is similar to Equation F.6, but with the conditional response probability given by an IRT model

$$p_{ij} = \sum_{\nu=1}^G \pi_{\nu} p_{ij\nu}. \quad (\text{F.6})$$

The IRT model may be a dichotomous model, such as the Rasch model (Mislevy & Verhelst, 1990; Rost, 1990, 1991), or a model for ordered response categories (Rost, 1991) or unordered response categories (Bolt, Cohen, & Wollack, 2001).² For example, for the mixed IPL model $p_{ij\nu}$ is

$$p(x_j = 1, \theta_{\nu}, \alpha_{\nu}, \delta_{j\nu}) = \frac{e^{\alpha_{\nu}(\theta_{\nu} - \delta_{j\nu})}}{1 + e^{\alpha_{\nu}(\theta_{\nu} - \delta_{j\nu})}}, \quad (\text{F.7})$$

where α_{ν} is the common item discrimination in latent class ν , $\delta_{j\nu}$ is item j ’s location in latent class ν , and θ_{ν} is the person location in latent class ν . Equation F.7 specifies the probability of a response of 1 on item j as a function of an individual’s latent class membership (indexed by ν), their location on the continuum (θ_{ν}) within latent class ν , and item j ’s parameters for the relevant latent class (α_{ν} and $\delta_{j\nu}$). If $\alpha_{\nu} = 1.0$, then Equation F.7 becomes the *mixture Rasch model* or *mixed Rasch model* (Rost, 1990, 1991).³ The IRT assumptions discussed in Chapter 2 hold within each latent class.

As is the case when applying IRT and LCA to empirical data we need to first determine the latent structure of our data. To accomplish this we fit a series of models with an increasing number of latent classes. For instance, we would fit a one class model to determine if multiple classes are necessary and then proceed to fit a two class model, a three class model, and so on. Assessing the model-data fit across this series model would involve fit indices such as AIC, consistent AIC (CAIC, Bozdogan, 1987), BIC, SABIC, or a statistical significance test such as the chi-squared difference test (i.e., the likelihood-ratio chi-squared test); AIC, BIC, and SABIC are discussed in Chapter 5. (Additional information concerning fit statistics may be found in Nylund, Asparouhov, and Muthén [2007].) The chi-squared difference test is applied between successive hierarchical models (e.g., a two- versus a three-latent class model) to determine if the additional latent class is necessary. A non-significant chi-squared difference test indicates that the model with the smaller number of latent classes is preferred over the model with the larger number of classes. With respect to the information criteria, models with the smaller information index exhibits better relative fit than models with larger information index values.

In addition to our numeric indices we also use latent class interpretability to aid in determining the latent class structure. For example, if our indices indicate a three-class

model is preferred to a two-class model, but we cannot interpret the all of the classes then a three-class structure is suspect. Moreover, if we can interpret the two-class structure, then we would select the two-class model over the three-class model. Of course, available theory should inform the number of latent classes retained.

Specialized software such as MIRA (Rost & von Davier, 1992), WINMIRA (von Davier, 2001), and the R packages `mRm` (Preinerstorfer, 2016), `mixRasch`, `psychomix` (Frick, Leisch, Strobl, Wickelmaier, & Zeileis, 2020; Frick, Strobl, Leisch, & Zeileis, 2012) can be used to estimate the mixture model's parameters for some of the members of the Rasch family of models. WINBUGS (Spiegelhalter, Thomas, Best, & Lunn, 2004) may also be used to estimate the model's parameters (see Bolt et al., 2001).

In addition to their use in addressing varying item discrimination, mixture models have been applied to the situation in which the latent classes represent different problem-solving strategies (Mislevy & Verhelst, 1990). For instance, imagine that a test item consists of a three-dimensional object. Participants are then shown a second object that may be the first object, albeit from a different perspective. The participants are asked whether the second object is the same as the first. In this example, one class could consist of individuals who employ a mental rotational strategy to solve the problem, whereas a second class might consist of people employing analytical reasoning to detect feature(s) that match without performing the rotation.

As a second potential application example, assume one administers an instrument designed to measure social anxiety to a sample of individuals. This sample may consist of a mixture of three latent populations or classes. Class interpretation shows that one class is comprised of persons who suffer from major depression, another class as consisting of persons who suffer from hebephrenia, and a third class contains persons who are neither of these types of individuals. Although it may be possible to measure social anxiety on a unidimensional continuum within each class, it may not be possible to place each of these individuals on a single social anxiety unidimensional continuum.

EXAMPLE: APPLICATION OF THE MIXTURE RASCH MODEL TO WRITING PROBLEM DATA, CMLE, WINMIRA

For this example we analyze data from a study designed to understand the nature of writing problems. Specifically, we wish to identify students that might have misconceptions and/or are utilizing potentially erroneous strategies that lead to writing problems. Along with a measure of writing knowledge (i.e., planning, organization, multiple drafts, revision strategies), the student participants are given either a verbal or pictorial writing prompt to assess their writing ability. Two raters holistically judge the participants' writing samples for verbal complexity (age-appropriate vocabulary, sentence structure) and syntactic correctness (age-appropriate spelling, handwriting, capitalization, and punctuation). Responses on writing knowledge are scored as correct or incorrect, whereas verbal complexity and syntactic correctness are each judged on a four-point scale with larger values indicating greater competence than lower values. Our data come from a sample of secondary schools. We collect data on 1174 student participants, seventy-nine

percent of whom are white with females accounting for 52% of the sample. Other demographic information is also collected.

In this example we use WINMIRA (von Davier, 2001) for estimation followed by a reanalysis using the R package `psychomix`; Endnote 4 shows a *Mplus* command file for performing a two-class mixture Rasch model analysis. WINMIRA is designed for the Rasch model for dichotomous and polytomous data and utilizes a graphical user interface. Thus, there is no syntax to present. Through the completion of a series of menus and dialogs we import our data, select our items to be analyzed, specify the number of latent classes, the mixed Rasch model to use, as well as output options. The data may be in a delimited ASCII format (tab, space, comma, etc.) or a SPSS data file. By default, WINMIRA uses the random start value of 4321 for its EM algorithm; this value can be changed to address local minima concerns. Example screen shots are shown in Figure F.1.

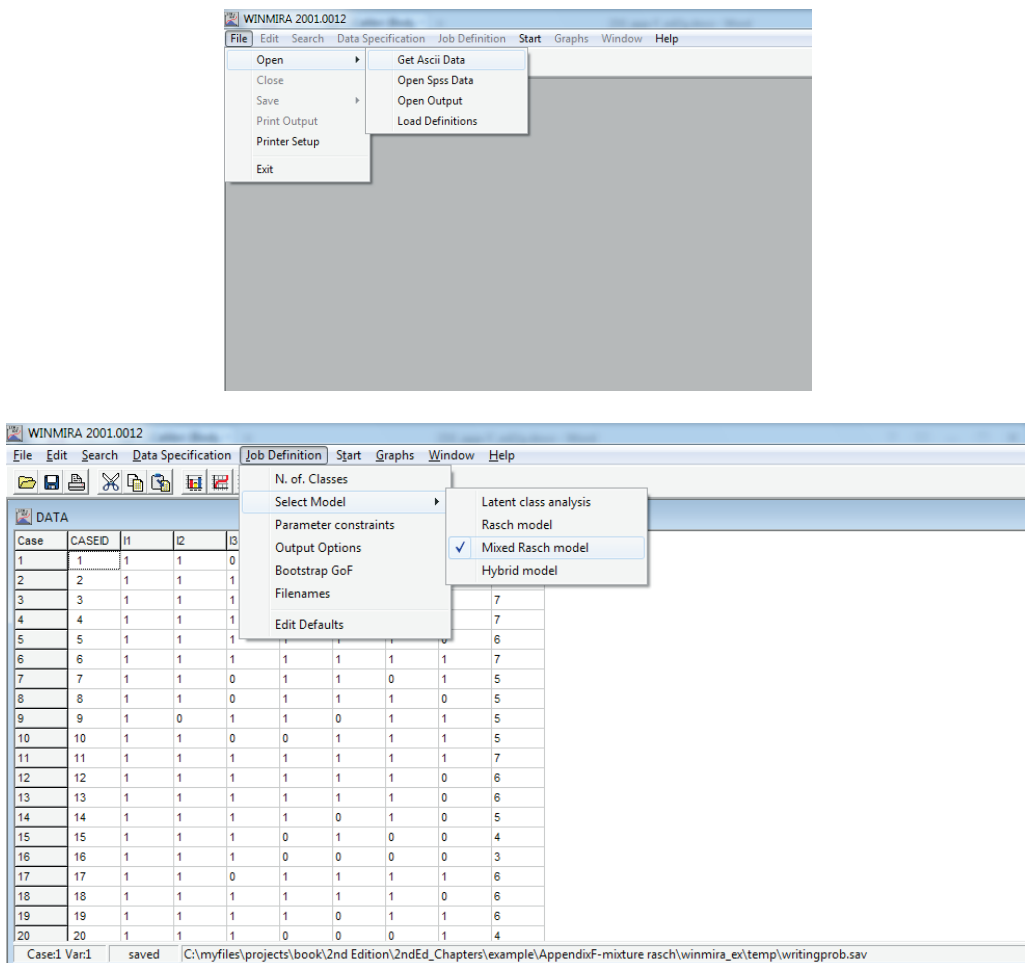


FIGURE F.1. WINMIRA Screen Shots for retrieving a csv data file (top) and specification of a mixture IRT model (bottom).

We examine multiple class models ranging from one to three classes. The program creates one ASCII output file per latent class with the extension “.OU#” where # is either “T” or a numeral. From the Job Definition menu’s Output Options item’s dialog we check several check boxes (e.g., ‘categories probabilities,’ ‘item threshold parameters,’ ‘person parameter estimates,’ ‘add person parameters,’ ‘item fit (Q-index)’). The information criteria for the one- through three-class solutions are presented in Table F.1 with the abridged output from the two-class solution shown in Table F.2.

For each of our models we check for a converged solution by verifying that the Number of iterations needed is less than the max. number of iterations. For example, the two-class solution required 493 iterations with a maximum of 1000 iterations (see Table F.2). Table F.1 shows that all three information criteria indicate that the two-class solution exhibits the best relative model-data fit.⁵ Therefore, we proceed to interpret the two-class solution to ensure that these classes are meaningful.

Our interpretation shows that latent class one consists of 826 participants who are knowledgeable about the writing process with an average number correct score on the writing knowledge measure of 5.00. Moreover, these participants could apply this knowledge in writing performance as exhibited in the verbal complexity and syntactic correctness of their writing samples (verbal $M = 3.05$, syntactic $M = 3.05$) regardless of prompt type. In contrast, the second class is composed primarily of participants ($n = 348$) who were less knowledgeable about the writing process (writing knowledge $M = 1.65$) and whose writing tends to not be seen as verbally complex and syntactically correct (verbal $M = 1.98$, syntactic $M = 1.86$) as members of the first latent class. Additionally, it appears that latent class 2 members tend to do better with pictorial prompts than with verbal prompts; latent class 1 members’ performance did not seem to be affected by the type of prompt. With latent class proportions of $\pi_1 = 0.69$ and $\pi_2 = 0.31$ we see that latent class 1 is almost twice the size of latent class 2 indicating that the majority of the participants are expected to be proficient in writing. On the basis of the combination of the fit information and our ability to interpret the two latent class solution we accept the two-class model for these data.

The output consists of general information about data input, descriptive statistics, followed by sections on each latent class, and then goodness of fit information (e.g., AIC, BIC) at the model level (Table F.2). Within each class section we have student location estimation (Expected Score Frequencies and Personparameters <sic>) followed by item parameter estimation information: estimated item locations, standard errors, and item-level fit information.

The Expected Score Frequencies and Personparameters section shows that a student that obtained a Rawscore of 1 in CLASS 1 of 2 is unlikely to be observed because we expect that 0.05% (i.e., $0.41/(0.69217*1174)*100\%$) of latent class 1 participants will have a number correct score of 1. Nevertheless, all participants with a $X = 1$ will be estimated to be located at -2.053 on the writing ability continuum ($\hat{\theta}_{11} = -2.053$) with a $s_e(\hat{\theta}) = 1.117$; we use Warm’s weighted likelihood estimates (Warm, 1989), WLE estimate, because these are available for all possible number correct scores and are less biased than the MLEs.

TABLE F.1. Model-Data Fit Indices, Latent Class Proportions, and LC Characteristic

Index	G = 1	G = 2	G = 3		
AIC	9172.74	9066.71	9075.41		
BIC	9213.28	9152.87	9207.18		
CAIC	9221.28	9169.87	9233.18		
π_c	G = 1	G = 2	G = 3*	(*off due to rounding)	
1	1	0.69	0.35		
2		0.31	0.33		
3			0.31		
G² analysis					
	1	2	3		
lnL	-4578.37	-4516.35	-4511.70		
# parameters	8	17	26		
-2lnL	9156.74	9032.70	9023.4		
G ²		124.04	9.30		
df		9	9		
p		0.0000	0.4101		
LC characteristics for G = 2.					
LC	Verbal Complexity	Syntactic Correctness	Prompt Type		WK
1	3.05	3.05	Verbal Pictorial	54.5% 45.5%	5.00
2	1.98	1.86	Verbal Pictorial	23.7% 76.3%	1.65
LC		Prompt Type			
		Verbal Complexity	Syntactic Correctness		
1	Verbal Pictorial	3.08 3.02	3.08 3.01		
2	Verbal Pictorial	1.00 2.31	1.89 1.85		

TABLE F.2. Abridged WINMIRA Output for G = 2 Solution

```
// WINMIRA 2001 1.45
// (c) 2000,2001 by Matthias von Davier
//           IPN - institute for science education
//           Olshausenstrasse 62
//           24098 Kiel
//           Germany
:
Filenames:

    data: C:\...\writingprob.sav
    output: C:\ ...\writingprob.OU1
    patterns: C:\ ...\writingprob.PAT

number of persons      :   1174
number of items       :     8
number of classes     :     2
max. number of iterations : 1000
accuracy criterion    : 0.0005
random start value    : 4321

item labels and sample frequencies:          ← category response frequencies (e.g., 987
no. | label      | cats | 0 | 1 | N      correct responses item 1)
-----|-----|-----|---|---|-----|
1 | I1           | 2 | 187 | 987 | 1174
2 | I2           | 2 | 245 | 929 | 1174
3 | I3           | 2 | 429 | 745 | 1174
4 | I4           | 2 | 449 | 725 | 1174
5 | I5           | 2 | 645 | 529 | 1174
6 | I6           | 2 | 695 | 479 | 1174
7 | I7           | 2 | 867 | 307 | 1174

saturated likelihood      :          -4459.4158    ← theoretical maximum based on
number of different patterns :              89    saturated modela
number of possible patterns :              128    ← = mt

Number of iterations needed: 493

fitted model: (MIRA) Mixed Rasch Model with smoothed score frequencies:
according to the ordinal (partial credit) model in 2 latent classes.

Classes are sorted by class size!
Final estimates in CLASS 1 of 2 with size 0.69217          ←  $\hat{\pi}_1$ 
=====

Expected Score Frequencies and Personparameters:          ←  $\hat{\theta}_i$ 
score frequency | person parameters and standard errors:
Raw- | Expected | MLE- | std. error | WLE- | std. error
score | freq.    | estimate | MLE | estimate | WLE
-----|-----|-----|-----|-----|-----
0 | 0.01 | ***** | ***** | -3.549 | 1.739
1 | 0.41 | -2.329 | 1.189 | -2.053 | 1.117
2 | 7.21 | -1.207 | 0.971 | -1.095 | 0.958
3 | 56.27 | -0.347 | 0.895 | -0.302 | 0.892
4 | 194.15 | 0.431 | 0.879 | 0.416 | 0.879
5 | 295.94 | 1.238 | 0.933 | 1.140 | 0.922
6 | 199.31 | 2.277 | 1.150 | 1.988 | 1.069
7 | 59.30 | ***** | ***** | 3.389 | 1.680
```

(continued)

TABLE F.2. (continued)

WLE estimates : Mean = 1.218 Var = 0.858 stdev = 0.927
 marginal error variance = 1.044 stdev = 1.022
 anova reliability = 0.451
 Andrichs reliability = -0.216

WLE = Warm's modified likelihood estimates,
 MLE = Standard maximum likelihood estimates.

Raw-score : Mean = 4.985 Stdev = 1.068

Smoothed Score Distribution descriptives:
 location: tau = 8.669
 dispersion: delta = 5.003
 approx. error: RMSEA = 0.018

expected category frequencies and item scores:

Item label	Item's		relative category frequencies	
	Score	Stdev	0	1
I1	0.97	0.18	0.035	0.965
I2	0.93	0.25	0.068	0.932
I3	0.77	0.42	0.232	0.768
I4	0.80	0.40	0.200	0.800
I5	0.60	0.49	0.403	0.597
I6	0.56	0.50	0.439	0.561
I7	0.36	0.48	0.638	0.362
Sum:	4.98			

threshold parameters: ordinal (partial credit) model

$\leftarrow \hat{\delta}_j$

item label	item location	threshold parameters
I1	-2.08405	
I2	-1.38368	
I3	0.02594	
I4	-0.16186	
I5	0.83262	
I6	0.97914	
I7	1.79190	

standard errors of item parameters:

itemlabel	location	threshold parameters
I1	0.1943332	
I2	0.1437841	
I3	0.0906126	
I4	0.0948103	
I5	0.0804422	
I6	0.0798488	
I7	0.0832941	

(continued)

TABLE F.2. (continued)

item fit assessed by the Q-index

itemlabel	Q-index	Zq	p(X>Zq)	
I1	0.2476	0.0069	0.49725	-...Q!....+
I2	0.2353	-0.0873	0.53477	-....Q.....+
I3	0.1687	-0.0925	0.53686	-....Q.....+
I4	0.2278	0.0207	0.49175	-...Q!....+
I5	0.1921	0.0130	0.49483	-...Q!....+
I6	0.2004	0.1753	0.43044	-...Q!....+
I7	0.2127	-0.0331	0.51320	-....Q.....+

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

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

Final estimates in CLASS 2 of 2 with size 0.30783 : $\leftarrow \hat{\pi}_2$

Expected Score Frequencies and Personparameters:

score frequency | person parameters and standard errors: $\leftarrow \hat{\theta}_2$

Raw- score	Expected freq.	MLE- estimate	std. error MLE	WLE- estimate	std. error WLE
0	46.42	*****	*****	-3.414	1.682
1	102.83	-2.296	1.154	-2.016	1.077
2	117.68	-1.242	0.944	-1.148	0.934
3	69.58	-0.412	0.891	-0.392	0.890
4	21.25	0.380	0.896	0.349	0.895
5	3.35	1.228	0.958	1.129	0.947
6	0.27	2.315	1.171	2.037	1.096
7	0.01	*****	*****	3.482	1.710

WLE estimates : Mean = -1.429 Var = 1.093 stdev = 1.046
 marginal error variance = 1.187 stdev = 1.089
 anova reliability = 0.480
 Andrichs reliability = -0.085

WLE = Warm's modified likelihood estimates,
 MLE = Standard maximum likelihood estimates.

Raw-score : Mean = 1.800 Stdev = 1.132

Smoothed Score Distribution descriptives:

location: tau = -8.301
 dispersion: delta = 4.045
 approx. error: RMSEA = 0.040

expected category frequencies and item scores:

Item label	Item's		relative category frequencies	
	Score	Stdev	0	1
I1	0.56	0.50	0.439	0.561
I2	0.48	0.50	0.525	0.475
I3	0.33	0.47	0.666	0.334
I4	0.21	0.40	0.794	0.206

(continued)

TABLE F.2. (continued)

I5		0.12		0.33		0.878		0.122
I6		0.06		0.25		0.936		0.064
I7		0.04		0.19		0.964		0.036

Sum: | 1.80

threshold parameters: ordinal (partial credit) model

← $\hat{\delta}_2$

item label	item location	threshold parameters
------------	---------------	----------------------

I1		-1.71525
I2		-1.35372
I3		-0.72570
I4		-0.04528
I5		0.59334
I6		1.31102
I7		1.93558

standard errors of item parameters:

itemlabel	location	threshold parameters
-----------	----------	----------------------

I1		0.1258122
I2		0.1240231
I3		0.1284062
I4		0.1443556
I5		0.1724884
I6		0.2237938
I7		0.2909052

item fit assessed by the Q-index

itemlabel	Q-index	Zq	p(X>Zq)					
I1		0.2148		0.1099		0.45626		-...Q!....+
I2		0.1861		0.0493		0.48034		-...Q!....+
I3		0.0982		0.4272		0.33461		-..Q!....+
I4		0.2187		-0.5004		0.69160		-....!Q....+
I5		0.1615		0.2476		0.40223		-...Q!....+
I6		0.1430		-0.6462		0.74094		-....!Q....+
I7		0.1840		-0.3810		0.64838		-....!Q....+

-?:p<0.05, +?:p>0.95
-!:p<0.01, +!:p>0.99

item discrimination index:

itemlabel	discr. index
-----------	--------------

I1		0.41087
I2		0.43146
I3		0.24690
I4		0.54753
I5		0.28137
I6		0.32598
I7		0.15607

(continued)

TABLE F.2. (continued)

person fit index descriptives:

```

mean      :      -0.0308950
std.dev.  :      0.9904008

skewness  :      -0.4775509
kurtosis  :      -0.6160654

```

statistics of expected class membership:

class	exp. size	mean prob.	1	2
1	0.657	0.937	0.937	0.063
2	0.296	0.898	0.102	0.898

Goodness of fit statistics:

	estimated model	saturated model
Log-Likelihood :	-4516.35	-4459.42
Number of parameters :	17	127
geom. mean likelihood :	0.57719881	0.58121180

Information Criteria:

AIC-Index :	9066.71	9172.83
BIC-Index :	9152.87	9816.49
CAIC-Index :	9169.87	9943.49

Power Divergence GoF statistics:

	emp. value	chi-square p-value
Cressie Read :	103.63	p= 0.6529
Pearson Chisquare :	105.87	p= 0.5936

```

=====
Likelihood ratio      :      113.88  p= 0.3809
Freeman-Tukey Chi^2  :      157.05  p= 0.0022
Degrees of freedom    :      110

```

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

```

obs.patterns/cells = 0.6953125000000000000
number of zero cells = 39

```

The data might be very sparse, please do not use the chi square p-value approximation for the Power Divergence Goodness of Fit Statistics. Consider to use the parametric bootstrap procedure instead. In addition, several start values should be used (see defaults menu) in order to examine the occurrence of local likelihood maxima.

*The saturated model has one parameter per response pattern/score probability.

Continuing with latent class 1 we see from the expected category frequencies and item scores section that items 1 and 7 were correctly answered by 97% and 36% of the participants, respectively. Our estimates of these items' locations are found in the threshold parameters: ordinal (partial credit) model section. For instance, item 1 is relatively easy in this class with an estimated location of -2.084 ($\hat{\delta}_{11} = -2.08405$; $s_e(\hat{\delta}_{11}) = 0.194$), whereas item 7 is comparatively harder with an estimated location of 1.792 ($\hat{\delta}_{71} = 1.79190$; $s_e(\hat{\delta}_{71}) = 0.083$).

Although we are using a mixture Rasch model items and people are on the same continuum within a latent class. Thus, we can predict that a participant that is member of LC 1 and who correctly answers only one item (i.e., $X = 1$, $\hat{\theta}_{11} = -2.053$) has a low probability (0.0209) of correctly answering item 7 (i.e., an item located farther up the continuum from their location), but has about a 50:50 chance of correctly answering item 1 (i.e., an item located at about the same point as their ability).⁶ (Our estimated participant and item locations can be transformed to eliminate negative values or to a scale that has intrinsic meaning such as a proportion or a number correct scale.) Similar information for latent class 2 is found in the Final estimates in CLASS 2 of 2 with... section.

In addition to model-level data fit information, WINMIRA also provides item-level fit information in the item fit assessed by the Q-index section. The Q-index (Rost & von Davier, 1994) has a range of 0 to 1 where small values are good. A Q-index = 0 reflects perfect fit (a Guttman pattern), a Q-index = 0.5 indicates random response behavior, and a Q-index = 1 indicates perfect misfit for the model. In contrast to a descriptive use of the Q-index, we can use a transformed Q-index for significance testing. This transformed Q-index, Z_q, is asymptotically normal, standardized, and centered at 0.

In the item fit assessed by the Q-index section we find both the Q-index and Z_q for each item in each latent class. For instance, in latent class 1 we see that Q-index ranges from 0.1687 to 0.2476. Thus, we have evidence supporting item-data fit for each of our items in this latent class. Similarly, for latent class 2 the Q-index values show that we have item-data fit. From a significance testing perspective we want non-significant Z_{qs}. As can be seen, we have non-significant Z_{qs} for each of our items in each of our latent classes. Therefore, in terms of fit our two-class model exhibits the best relative fit of our three models and our Q-index values show item-level fit in each of our latent classes. For comparison, our one-class solution shows that item 3 exhibits misfit (Q-index = 0.1691; Z_q = 1.7517 with $p(X > Z_q) = 0.03991$).

WINMIRA provides several latent class-oriented plots. For example, Figure F.2 presents a double-Y graphical depiction of our proficiency estimates and their frequency distributions for each of our latent classes. As can be seen, latent class 1 participants tend towards being distributed at and above $X = 4$, whereas latent class 2 participants tend to obtain fewer items correct. However, for a given number correct score the participants are estimated to be similar in writing proficiency. For example, for participants in latent class 1 that correctly answered four items we would estimate their writing proficiency to be 0.416 (see dash-dot line, top panel). However, for latent class 2 participants

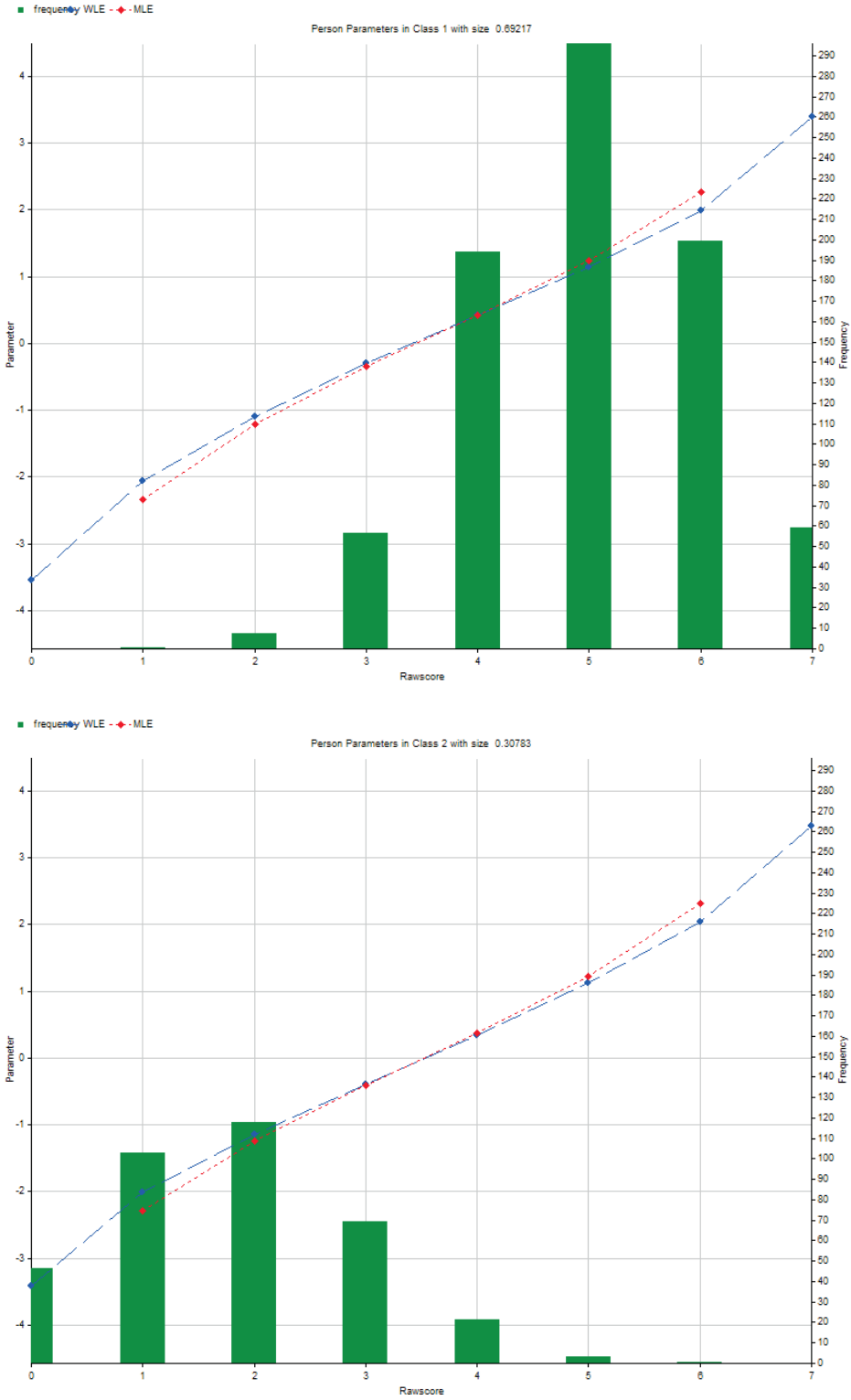


FIGURE F.2. Latent proficiency locations and frequency distribution of X.

who had a number correct score of 4 our estimated location for them is 0.349 (see dash-dot line, bottom panel). Because of the class structure we know these latter participants would tend to respond correctly to the pictorial prompts, but not verbal prompts, as well as the types of problems we would see in their writing (i.e., few related ideas, stories that lack cohesiveness, etc.).

To examine how item responses vary across latent classes we can examine category probability plots. For instance, Figure F.3 shows that latent class 2 members tend to provide more incorrect than correct responses on items 2–7 (i.e., as item difficulty increases in class 2), whereas latent class 1 members show the opposite pattern except on item 7.

The previous two plots figures are “person-oriented.” From an item perspective one can examine how item thresholds function within classes. For our dichotomous data these category thresholds are our item locations. Figure F.4 shows our item parameter plot. As can be seen, overall the profiles for the two classes are very similar.

Above we mention that we checked the `add person parameters` check box. The results are shown in Figure F.5 (i.e., columns `PERSPAR` to `Z2`). Our person location estimates and their standard errors are found in the columns `PERSPAR` and `STDERR`, respectively. For example, for `ID = 1` we have $\hat{\theta}_{11} = 0.4161$, $s_e(\hat{\theta}_{11}) = 0.8789$ and for `ID = 1174` we have $\hat{\theta}_{1174,2} = -1.1483$, $s_e(\hat{\theta}_{1174,2}) = 0.9342$. The columns `P1` and `P2` contain the LC assignment probabilities (e.g., for `ID = 1` we have `P1 = 0.9281` and `P2 = 0.0719`; the numeral represents the LC) with the largest shown in `MAXPI`. The most probable class for the first case is latent class 1 (`MAXCLASS = 1.0000`) because `P1 > P2`. The `NEWFIT` columns show an approximately normally distributed person fit index. For the cases shown our values reflect response patterns that are consistent with the model.

EXAMPLE: APPLICATION OF THE MIXTURE RASCH MODEL TO WRITING PROBLEM DATA, `CMLE`, `psychomix`

We reanalyze our data using the R package `psychomix`. Because the estimation algorithm in `psychomix` is not the same as used in `WINMIRA` and each program utilizes random number generators their respective results will be close, but not necessarily identical. Table F.3 shows our R session.

After reading our data into the data frame `writingX` and verifying the data were correctly read we obtain the frequency distribution of our observed score X . We have $42 + 55 = 97$ zero-variance response vectors. Because we will subsequently use `writingX` we copy only the response vectors to a second object `writing` (i.e., we remove the case `ID` and `X`).

In our call to `raschmix` we use the model formula specification approach. Specifically, the outcome variables (`I1`, `I2`, ... , `I7`) precede the tilde with any concomitant/covariates following it. In our case, there are no covariates and so we simply specify a “1.” We specify the fitting of one- to three-latent class models ($k = 1:3$). We could use the `which` argument to have `raschmix` automatically select the model with the

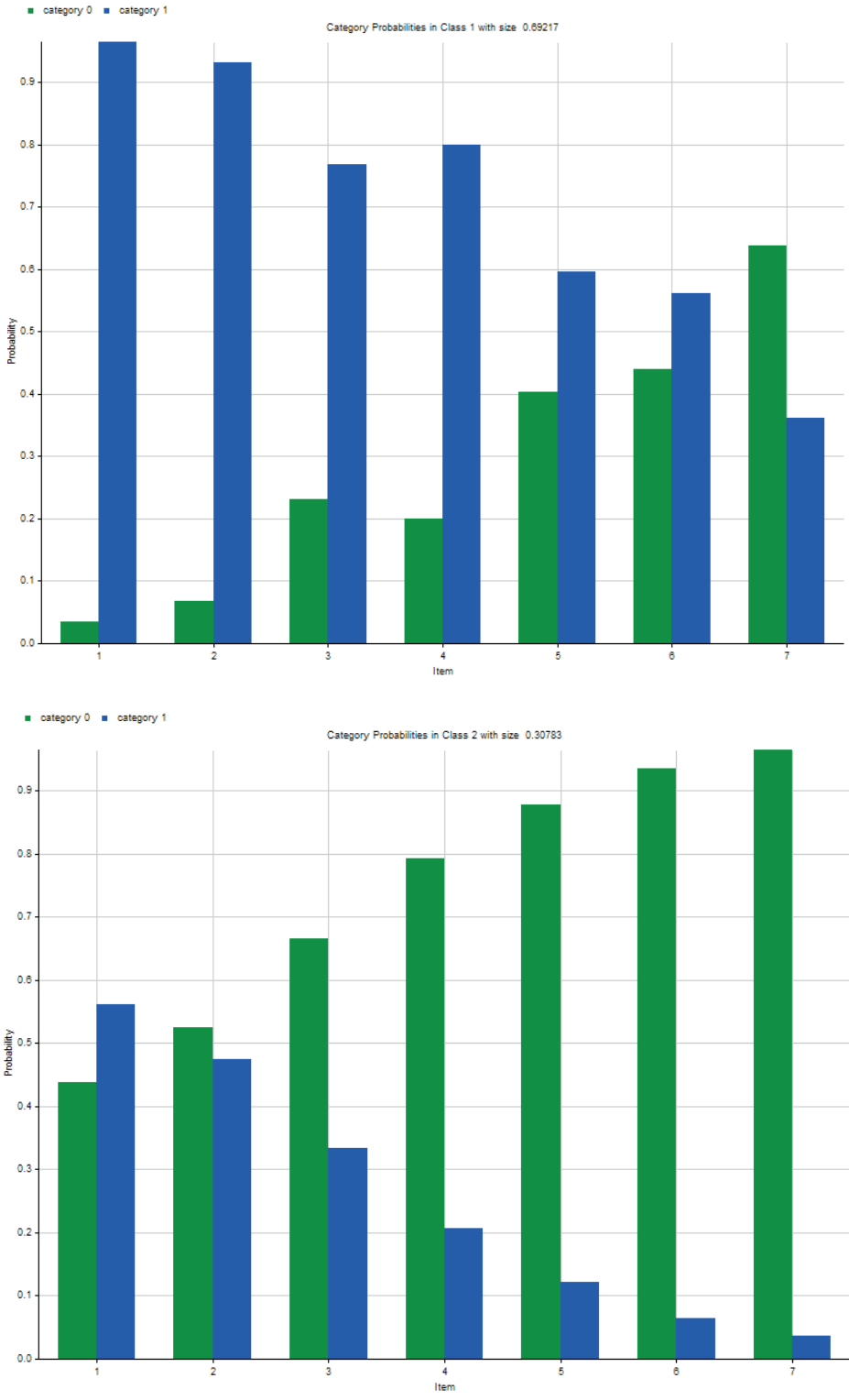
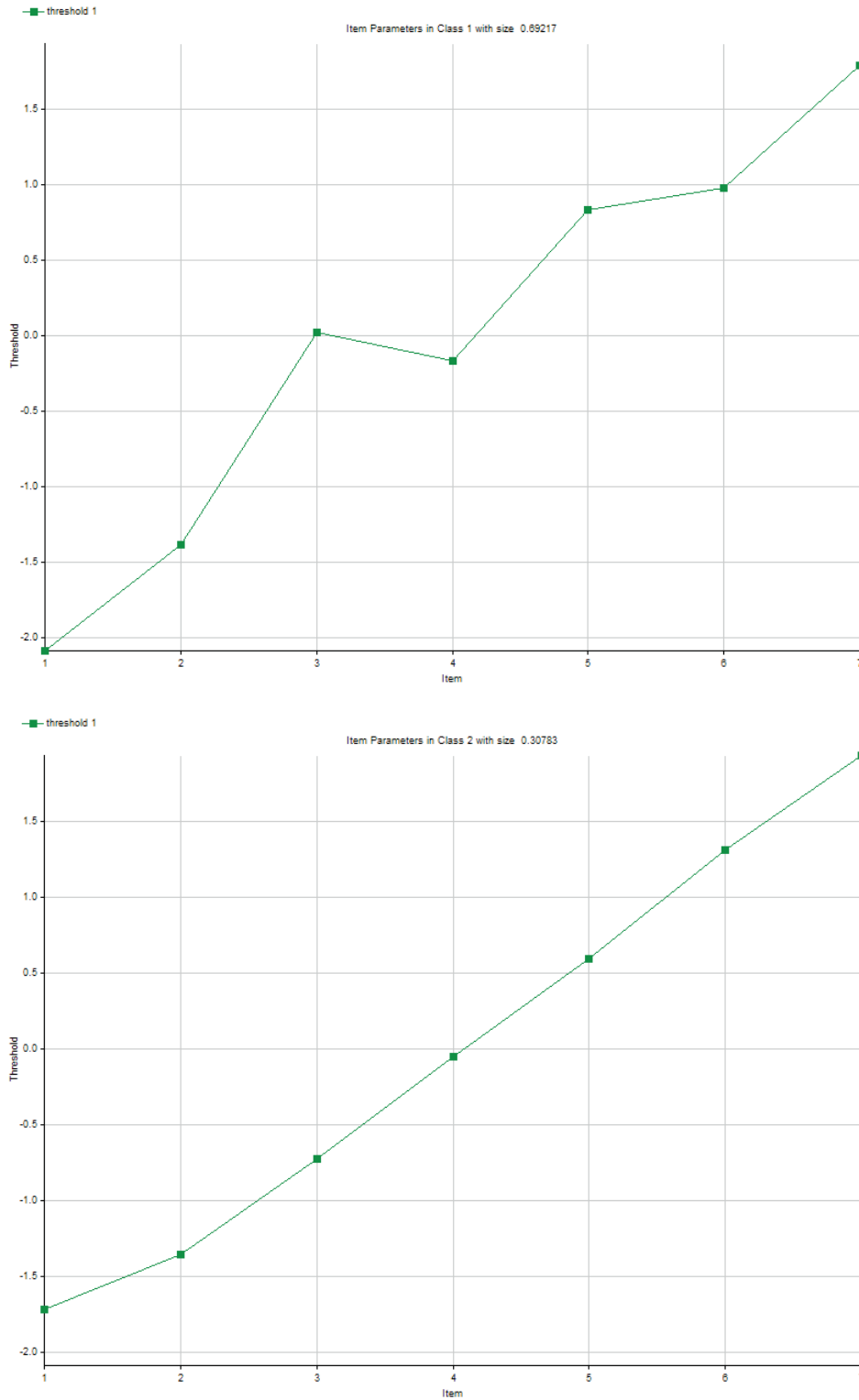


FIGURE F.3. Category probability plots.

**FIGURE F.4.** Item parameter plots.

DATA

Case	CASED	I1	I2	I3	I4	I5	I6	I7	X	PERSPAR	STDERR	OLDFIT	NEWFIT1	NEWFIT2	MAXPI	MAXCLAS	P1	Z1	P2	Z2
1	1	1	1	0	1	1	0	0	4	0.4161	0.8769	0.3413	0.1759	0.1742	0.9281	1.0000	0.9281	0.2032	0.0719	-0.2008
2	1	1	1	1	1	1	0	0	5	1.1402	0.9216	0.8861	0.9464	0.9442	0.9839	1.0000	0.9839	0.9435	0.0161	0.9899
3	1	1	1	1	1	1	1	1	7	3.3887	1.6795	.	.	.	0.9998	1.0000
4	1	1	1	1	1	1	1	1	7	3.3887	1.6795	.	.	.	0.9998	1.0000
5	1	1	1	1	1	1	1	0	6	1.9884	1.0688	0.8645	0.9040	0.9038	0.9985	1.0000	0.9985	0.9040	0.0015	0.7243
6	1	1	1	1	1	1	1	1	7	3.3887	1.6795	.	.	.	0.9998	1.0000
7	1	1	0	1	1	1	0	1	5	1.1402	0.9216	-0.8855	-0.9626	-0.9617	0.9934	1.0000	0.9934	-0.9595	0.0066	-1.2903
8	1	1	0	1	1	1	1	0	5	1.1402	0.9216	-0.8622	-0.2978	-0.2975	0.9945	1.0000	0.9945	-0.2943	0.0055	-0.8930

Case:19 Var:20 modified C:\myfiles\projects\book\2nd Edition\2ndEd_Chapters\example\AppendixF-mixture rasch\winmira_ex\temp\writingprob.sav

DATA

Case	CASED	I1	I2	I3	I4	I5	I6	I7	X	PERSPAR	STDERR	OLDFIT	NEWFIT1	NEWFIT2	MAXPI	MAXCLAS	P1	Z1	P2	Z2
1165	1	0	0	0	0	0	0	0	1	-2.0156	1.0774	0.7388	0.7397	0.7391	0.9949	2.0000	0.0051	0.6394	0.9949	0.7396
1166	1	0	0	0	0	0	0	0	1	-2.0156	1.0774	0.7388	0.7397	0.7391	0.9949	2.0000	0.0051	0.6394	0.9949	0.7396
1167	1	1	0	0	1	0	0	0	3	-0.3021	0.8921	0.7231	0.7159	0.6896	0.5706	1.0000	0.5706	0.9323	0.4294	0.3670
1168	0	0	0	0	0	0	0	0	0	-3.4142	1.6815	.	.	.	0.9998	2.0000
1169	1	1	1	0	1	0	0	0	3	-0.3021	0.8921	0.7231	0.7159	0.6896	0.5706	1.0000	0.5706	0.9323	0.4294	0.3670
1170	1	1	1	0	0	0	0	0	2	-1.1483	0.9342	0.9697	1.0552	1.0444	0.9246	2.0000	0.0754	0.9661	0.9246	1.0507
1171	1	1	1	0	0	0	0	0	2	-1.1483	0.9342	0.9697	1.0552	1.0444	0.9246	2.0000	0.0754	0.9661	0.9246	1.0507
1172	1	0	0	0	1	0	0	0	2	-1.1483	0.9342	-0.2118	-0.3959	-0.3915	0.9184	2.0000	0.0816	-0.1870	0.9184	-0.4097
1173	1	1	1	1	0	0	0	0	3	-0.3919	0.8905	0.9292	0.9918	0.9576	0.6420	2.0000	0.3580	0.6942	0.6420	1.1045
1174	0	1	1	1	0	0	0	0	2	-1.1483	0.9342	0.0439	-0.1470	-0.1465	0.9741	2.0000	0.0259	-0.8259	0.9741	-0.1284

Case:19 Var:20 modified C:\myfiles\projects\book\2nd Edition\2ndEd_Chapters\example\AppendixF-mixture rasch\winmira_ex\temp\writingprob.sav

FIGURE F.5. Data file augmented by person information.

TABLE F.3. psychomix R Session for Analysis of Writing Problem Data

```

> library(psychomix)
> packageVersion("psychomix")
[1] '1.1.8'
> set.seed(99999)

> writingX=read.table("WritingProb.dat",header=T)

> head(writingX,n=5)
  id I1 I2 I3 I4 I5 I6 I7 X
1  1  1  1  0  1  1  0  0  4
2  2  1  1  1  1  1  0  0  5
3  3  1  1  1  1  1  1  1  7
4  4  1  1  1  1  1  1  1  7
5  5  1  1  1  1  1  1  0  6

> tail(writingX,n=5)
  id I1 I2 I3 I4 I5 I6 I7 X
1170 1170 1  1  0  0  0  0  0  2
1171 1171 1  1  0  0  0  0  0  2
1172 1172 1  0  0  1  0  0  0  2
1173 1173 1  1  1  0  0  0  0  3
1174 1174 0  1  1  0  0  0  0  2

> table(writingX$X)
 0  1  2  3  4  5  6  7
42 116 112 134 212 292 211  55

> # remove X & case ID from data frame
> writing=within(writingX,rm(X)); writing=within(writing,rm(id))

> # use formula approach to specify model
> writingMix=raschmix(i1+i2+i3+i4+i5+i6+i7~1,data=writing,k=1:3,score="meanvar")
 1 : * * *
 2 : * * *
 3 : * * *

> writingMix                                     # failure to converge with 3 class model
Call:
raschmix(formula = I1 + I2 + I3 + I4 + I5 + I6 + I7 ~ 1, data = writing,
  k = 1:3, scores = "meanvar")

  iter converged k k0   logLik      AIC      BIC      ICL
1     2      TRUE 1  1 -4541.771 9103.543 9153.362 8536.908
2    71      TRUE 2  2 -4514.177 9066.355 9161.012 9072.554
3   200     FALSE 3  3 -4507.962 9071.924 9211.418 9255.865

> # repeat analysis increasing default max iterations to a 1000; use nonformula
approach
> dl=as.matrix(writing)
> print((writingMix=raschmix(data=dl,k=1:3,score="meanvar",control=list(iter=
1000))))
 1 : * * *
 2 : * * *
 3 : * * *

```

(continued)

TABLE F.3. *(continued)*

```

Call:
raschmix(data = dl, k = 1:3, scores = "meanvar", control = list(iter = 1000))

  iter converged k k0    logLik      AIC      BIC      ICL
1     2      TRUE 1  1 -4541.771 9103.543 9153.362 8536.908
2    73      TRUE 2  2 -4514.177 9066.353 9161.010 9072.605
3   215      TRUE 3  3 -4507.897 9071.794 9211.288 9215.493

> writing2LC=getModel(writingMix, which = "2")

> summary(writing2LC)
Call:
raschmix(data = dl, k = 2, scores = "meanvar", control = list(iter = 1000))

      prior size post>0 ratio
Comp.1  0.54  682   1077 0.633
Comp.2  0.46  395   1077 0.367

Item Parameters:
      Comp.1      Comp.2
i1 -2.2544134 -1.69223340
i2 -1.4055979 -1.32317135
i3  0.2100710 -0.66747131
i4 -0.2086226 -0.01229296
i5  0.9011347  0.64267087
i6  0.9373187  1.23569648
i7  1.8201096  1.81680168

'log Lik.' -4514.177 (df=19)
AIC: 9066.353   BIC: 9161.01

> writing2LC@nobs                                # Sample size used: extreme scores removed
[1] 1077

> writing1LC=getModel(writingMix,which="1")
> logLik(writing1LC)                             # lnL for 1 class
'log Lik.' -4541.771 (df=10)

> logLik(writing2LC)                             # lnL for 2 classes
'log Lik.' -4514.151 (df=19)

> writing3LC=getModel(writingMix,which="3")
> logLik(writing3LC)                             # lnL for 3 classes
'log Lik.' -4507.897 (df=28)

> plot(writing2LC,pos="topleft")                  # produces Figure F.6

> # LC membership for non-zero variance response vectors; 2 class model; N'=1077
> grp = data.frame(clusters(writing2LC))

> histogram(writing2LC)                          # produces Figure F.7

> N = length(writing[,1])                        # number of cases incl 0-variance response vectors
> L = length(writing[1,])                        # number of items
> freq = as.numeric(table(writingX$X))          # determine the # of 0-variance response vectors, trimN
> trimN = freq[1]+freq[L+1]                      # 1-based indexing: freq[1] is X=0, freq[L+1] is X=7

```

(continued)

TABLE F.3. (continued)

```

> CensoredWriting=as.data.frame(matrix(-99.9,nrow=(N-trimN),ncol=9))      # initialize
> g=1L
> trim zero-variance response vectors
> for (i in 1:N) {
+   if((writingX$X[i]>0) & (writingX$X[i] < L)) {
+     CensoredWriting[g,]=writingX[i,]; g=g+1
+   } # if
+ } # for i

> names(CensoredWriting) = c("id",paste0("I",1:7),"X")                    # use meaningful names
> names(grp) = c("LC")

> Nprime=length(CensoredWriting$X)                                       # check - should match writing2LC@nobs
> Nprime
[1] 1077

> CensoredWriting=cbind(CensoredWriting,grp)                               # merge latent class membership

> # merge assignment probabilities
> CensoredWriting=cbind(CensoredWriting, posterior(writing2LC))

> # LC: latent class membership; '1' & '2' are LC=1 & LC=2 assignment probabilities
> head(CensoredWriting,6)
  id I1 I2 I3 I4 I5 I6 I7 X LC      1      2
1  1  1  1  1  0  1  1  0  0  4  1 0.7416410 0.2583590
2  2  2  1  1  1  1  1  0  0  5  1 0.7330276 0.2669724
3  3  5  1  1  1  1  1  1  0  6  1 0.7982802 0.2017198
4  4  7  1  1  0  1  1  0  1  5  1 0.8681011 0.1318989
5  5  8  1  1  0  1  1  1  0  5  1 0.8989814 0.1010186
6  6  9  1  0  1  1  0  1  1  5  1 0.8147455 0.1852545

> tail(CensoredWriting,5)
  id I1 I2 I3 I4 I5 I6 I7 X LC      1      2
1073 1170  1  1  0  0  0  0  0  2  2 0.06020964 0.9397904
1074 1171  1  1  0  0  0  0  0  2  2 0.06020964 0.9397904
1075 1172  1  0  0  1  0  0  0  2  2 0.06698696 0.9330130
1076 1173  1  1  1  0  0  0  0  3  2 0.20688811 0.7931119
1077 1174  0  1  1  0  0  0  0  2  2 0.01495640 0.9850436

> write.csv(CensoredWriting, file = "WritingProbXLC.csv")av

```

lowest information criterion (e.g., which="BIC"). However, we believe it is prudent to examine all the results to determine our best model. `psychomix` allows the use of either a saturated score model or a mean-variance score model parameterization of the raw score probabilities distribution. To mirror our WINMIRA approach we specify the mean-variance score model (score="meanvar"). `psychomix` provides a progress indicator during its execution. As it completes each stage for a model it displays an "*" (i.e., "1 : *," "1 : * *," "1 : * * *"); the "1:..." are for the one-class model, the "2:..." are for the two-class model, and so on. Displaying our output object, `writing-Mix`, shows that we failed to achieve convergence with the three-class model because the permissible number of iterations was reached.

In our second call we demonstrate a simpler model specification approach that takes advantage of the fact our model does not have concomitant/covariates. This specification requires that we only provide the data matrix (`writingMix= raschmix(data=d1,...)`). To address our first attempt's nonconvergence we increase the maximum number of iterations to 1000 (`control=list(iter=1000)`) in our second call. As can be seen, convergence for the three-class model was achieved in 215 iterations.

Our model-data fit information shows AIC is lowest for the two-class model, whereas BIC and ICL are lowest for the one-class model.⁷ Given AIC's tendency to suggest models with more rather than fewer classes we would, given BIC, select the one-class model. Why do these model-fit results differ from those of WINMIRA? As mentioned above there are differences in the implementations (e.g., the number of model parameters vary), but there are also slightly different data being analyzed. `raschmix` removes all zero-variance response vectors so we have $N' = 1077$ respondents and not $N = 1174$.⁸ For completeness, we note that the G^2 for the two- versus one-class models is significant ($G^2 = 55.188$, $p = 0.0000$, $df = 9$) thus supporting the use of the two- over the one-class model, whereas it is not significant for the three- versus two-class models ($G^2 = 12.560$, $p = 0.1835$, $df = 9$). For pedagogical reasons we proceed with the two-class solution.

We extract the two-class solution from the `writingMix` output object using the `getModel` function. With latent class proportions of $\pi_1 = 0.63$ and $\pi_2 = 0.37$ we see that latent class 1 is almost twice the size of latent class 2. Our item location estimates follow in the Item Parameters table. For example, item 1 is the easiest in each class (i.e., $\hat{\delta}_{11} = -2.2544$ and $\hat{\delta}_{12} = -1.6922$), whereas item 7 is estimated to be the most difficult in each class, $\hat{\delta}_{71} = 1.8201$ and $\hat{\delta}_{72} = 1.8168$. Similar information is graphically obtained from the item profile (Figure F.6). Our estimated locations show correlations with WINMIRA's of 0.9978 for LC 1 and 0.9994 for LC 2.

The `clusters` function provides us with the latent class assignments for each of our 1077 cases (see `writing2LC@nobs`) with the `posterior` function providing the corresponding posterior probabilities used for these assignments. (We store the latent class assignments in the `grp` object for further use.) Figure F.7 shows the posterior probabilities for each latent class. Generally speaking, we would like to have our respondents either high or low in each class. In the two-class solution (top) this pattern is somewhat evident except at the upper end of LC 1 and the lower end of LC 2. For pedagogical purposes we present the three-class solution in the bottom panel. As can be seen, the U-shape pattern is absent for two of the classes.

To match our latent class assignments to the corresponding respondents requires that we remove the zero-variance response vectors from our original data frame, `writingX`. To this end, we adopt a general approach to coding by using variables rather than "hard-coding" values (e.g., `for` loop). We first determine the sample size and instrument length using the `length` function followed by calculating the number of zero-variance response vectors, `trimN`. Our third step is to initialize a data frame, `CensoredWriting`, which will contain our results. Subsequently, we copy our original case data from `writingX` to `CensoredWriting` when the case satisfies $0 < X < 7$. We use the names

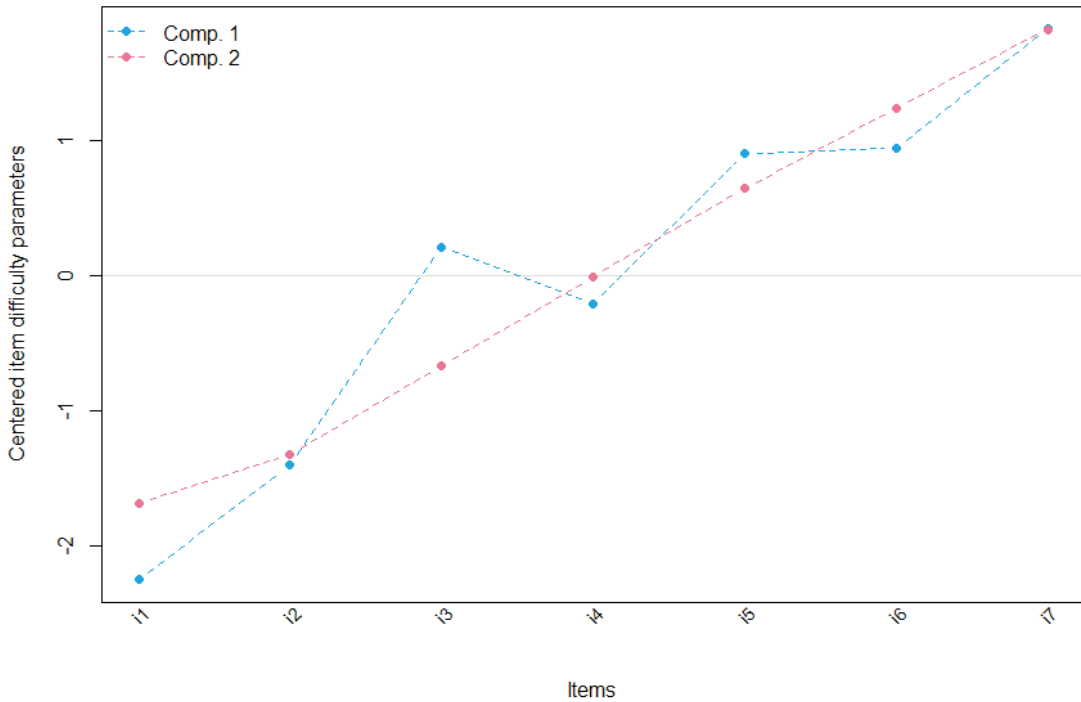


FIGURE F.6. Estimated Item Parameter profile.

function to assign meaningful variable names. Our final step is to use the `cbind` function to merge the LC assignment object `grp` and the posterior probabilities (`cbind(..., posterior(writing2LC))`) to `CensoredWriting`. The results are shown using the `head` and `tail` functions as well as written to an external csv file.

NOTES

1. Strictly speaking, a latent class model with $G \geq (L + 1)/2$ latent classes gives the same estimates of item parameters as the Rasch model does under conditional maximum likelihood estimation. In addition to the relationship between LCA and the Rasch model, the Rasch model is related to a log-linear model (e.g., Baker & Subkoviak, 1981; Kelderman, 1984; also see Holland, 1990a, 1990b). That is, the Rasch model may be expressed as a log-linear model for the probabilities of each unique response pattern (Cressie & Holland, 1983). In this case, it is possible to estimate the parameters via log-linear analysis (see Mellenbergh & Vijn, 1981, as well as Kelderman, 1984).

2. Yamamoto (1989) developed a HYBRID model that eliminates the constraint that the same item response model hold in each latent class. Thus, one may have different IRT models in each class or have an IRT model in one class, but not in another class. Boughton and Yamamoto (2007) show how the HYBRID model may be applied to the analysis of speededness. The WINMIRA software package can estimate the HYBRID model.

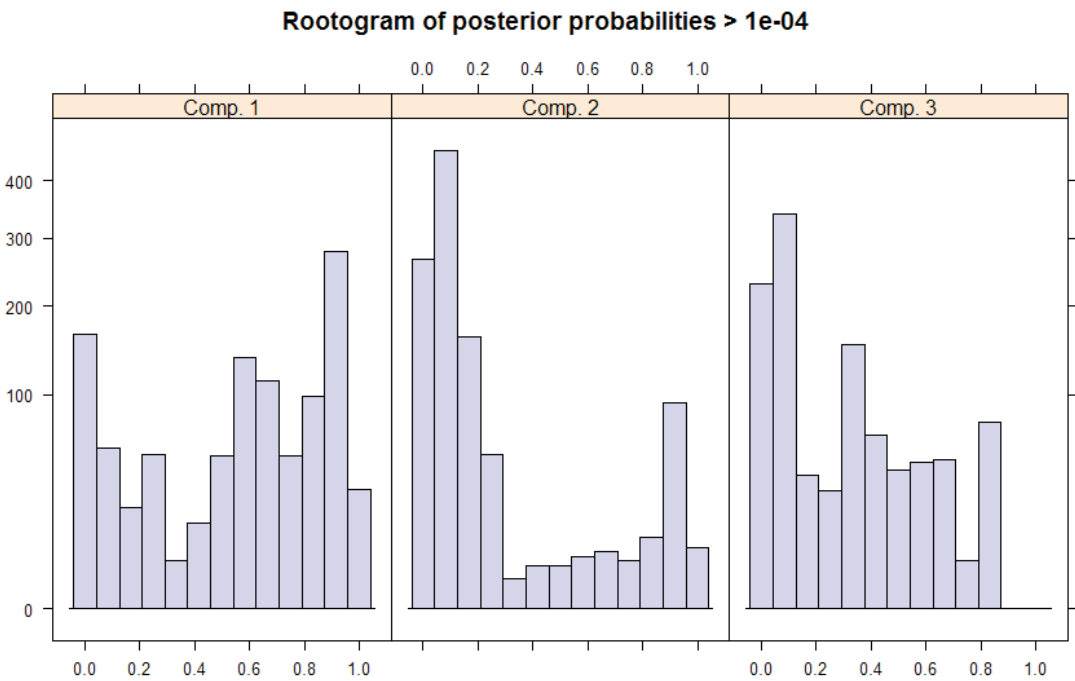
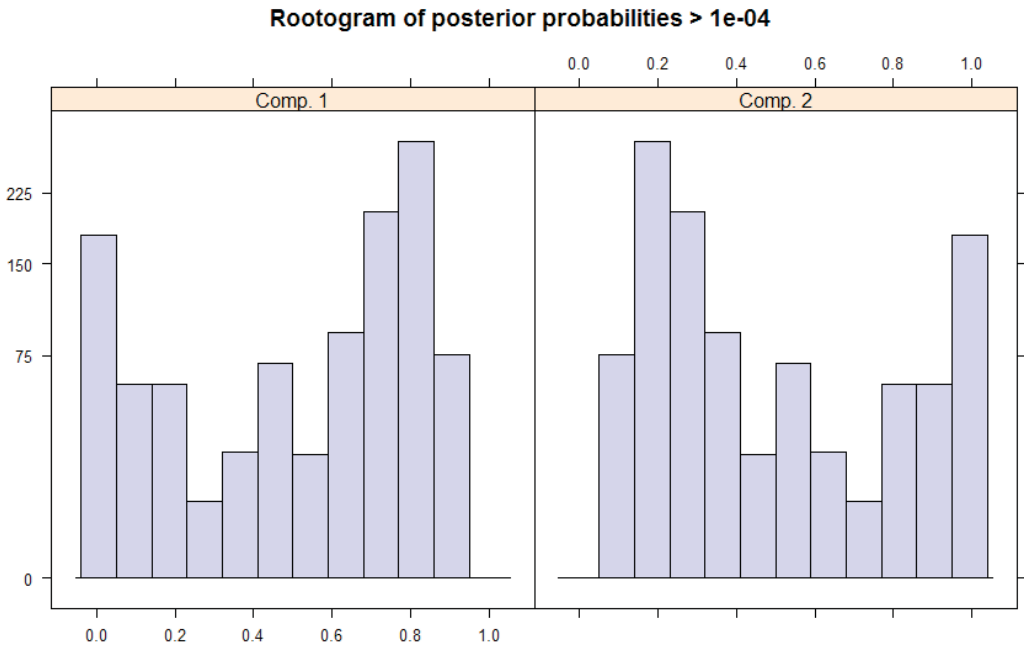


FIGURE F.7. Posterior probabilities for two- (top) and three- (bottom) class solutions. *Note.* LC = 1 : Comp. 1; LC = 2 : Comp. 2.

3. In the literature one finds synonymous terms, such as, Rasch mixed model or Rasch mixture model. We will precede the IRT model (e.g., “Rasch”) with either “mixture” or “mixed” to be consistent with the format used throughout this book in which the “type” comes first (e.g., “partial credit” Rasch model, “modified one-parameter” logistic model, “dichotomous” Rasch model, “two-parameter logistic” (2PL) model).

4. To perform our mixture Rasch model analysis using *Mplus* we begin by specifying a single latent class model and proceed to two- and three-latent class models. This is the syntax for $G = 2$:

```
Title: mix IRT - 2 LCs
Data: file="<filename.csv>";
Variable:names=i1-i10;
categorical=i1-i7;
classes=c(2);
analysis:type=mixture;
algorithm=integration;
starts=200 50;
model: %overall%
f by i1-i7*(1);
[f@0];
%c#1%
f by i1-i7@1;
f;
[i1$1-i7$1];
%c#2%
f by i1-i7@1;
f;
[i1$1-i7$1];
plot: type=plot3;
output: tech1 tech8;
```

The general format is the specification of the variables and their nature (`names=i1-i7` and `categorical=i1-i7`), the two class model (`classes=c(2)`), the use of categorical and continuous latent variables (`type=mixture`), followed by the model specification. We start with an overall model specification of what is in common across our latent classes. That is, our latent variable (f) is being measured by each of our ten items (i.e., i_1, \dots, i_{10}). Each of these items has a starting value of 1 and we fix the latent variable’s mean to 0 in all classes (`[f@0]`); also addresses model identification. The class specific parts follow. Class 1 (i.e., `%c#1%`) is presented first and followed by the second class (i.e., `%c#2%`). We want to estimate a model with a constant discrimination (i.e., 1) within each class set with each item measuring the continuous factor (`f by i1-i7@1`) within a class, but allow item locations to vary. (Technically, allowing the item thresholds to vary, but these thresholds are transformed to the item locations. Similarly, setting the item loadings to be 1, but the loadings are also transformed to be our item discriminations.)

5. As an example of obtaining AIC and BIC we use Equations 5.11 and 5.12. For the two-class model we have $N_{\text{parm}} = 17$ and $N = 1174$

$$\text{AIC} = -2\ln L + 2N_{\text{parm}} = -2(-4516.35) + 2(17) = 9066.70$$

and

$$\text{BIC} = -2\ln L + \ln(N)N_{\text{parm}} = -2(-4516.35) + \ln(1174)(17) = 9152.86.$$

6. This interpretation comes from $\hat{\theta}_1 = -2.053$ and $\hat{\delta}_{71} = 1.79190$ and Equation F.7; $\alpha = 1.0$:

$$p_{71} = \frac{e^{(\hat{\theta}_1 - \hat{\delta}_{71})}}{1 + e^{(\hat{\theta}_1 - \hat{\delta}_{71})}} = \frac{e^{(-2.053 - 1.79190)}}{1 + e^{(-2.053 - 1.79190)}} = 0.0209,$$

and again with $\hat{\delta}_{11} = -2.08406$:

$$p_{11} = \frac{e^{(\hat{\theta}_1 - \hat{\delta}_{11})}}{1 + e^{(\hat{\theta}_1 - \hat{\delta}_{11})}} = \frac{e^{(-2.053 - (-2.08406))}}{1 + e^{(-2.053 - (-2.08406))}} = 0.5078.$$

7. The integrated completed likelihood (Biernacki, Celeux, & Govaert, 2000; ICL) maximizes the integrated likelihood to select the “best” model; the ICL is also known as integrated classification likelihood. The ICL can be obtained by using BIC as an approximation to the integrated likelihood of the complete data (a.k.a., ICL_{BIC}). That is, the ICL modifies BIC’s penalty to reflect the quality of classification as measured by entropy (E):

$$\text{ICL}_{\text{BIC}} = -2\ln L + \ln(N)N_{\text{parm}} - 2E,$$

where $E = \sum_{i=1}^N \sum_{v=1}^G p(v | \mathbf{x}_i) \ln(p(v | \mathbf{x}_i))$. As is the case with the information indices, smaller ICL_{BIC} values indicate better relative fit than do larger values.

Because E is negative it is sometimes presented as its opposite with “+2E” replacing “-2E” in ICL_{BIC} ; in certain contexts E is calculated using base 2. E does not have an upper bound. Thus, it is sometimes rescaled to be [0,1]: $E' = 1 - \frac{-E}{N \ln(\nu)}$ ($\nu > 1$) with values approaching 1 reflecting “good” values. One guideline for interpreting E' uses 0.8 to reflect the minimum for a “good” value. When $E \approx 0$ (or $E' = 1$) the classes are well separated and the classification of respondents is essentially perfect given the model. *Mplus* provides E' (entropy) in its output.

8. As an example of obtaining AIC and BIC we use Equations 5.11 and 5.12. For the two-class model we have $N_{\text{parm}} = 19$ and $N' = 1077$

$$\text{AIC} = -2\ln L + 2N_{\text{parm}} = -2(-4514.177) + 2(19) = 9066.35$$

and

$$\text{BIC} = -2\ln L + \ln(N')N_{\text{parm}} = -2(-4514.177) + \ln(1077)(19) = 9161.01.$$

Appendix G

Miscellanea

This appendix contains various sections that provide either background information, alternative/additional analyses, or models not covered in the chapters of this book. For instance, we discuss using principal axis estimation as well as provide background information on odds and odds ratios and FORTRAN formats. We think that this “nether land” appendix is important, but not necessarily required to understand the material in the chapters.

USING PRINCIPAL AXIS FOR ESTIMATING ITEM DISCRIMINATION

One may use the relationship between the two-parameter normal ogive model and factor analysis (see Appendix C “Extending the Two-Parameter Normal Ogive Model to a Multidimensional Space”) to obtain item parameter estimates (Lord & Novick, 1968; Mislevy, 1986b; Takane & de Leeuw, 1987). This alternative approach for estimating the discriminations involves performing a principal axis analysis of the tetrachoric correlation matrix for the data.¹ PRELIS (Jöreskog & Sörbom, 1999) is used to obtain the tetrachoric correlation matrix, \mathbf{R}_T , shown in Table G.1. (Alternative ways to obtain the tetrachoric correlation matrix are to use an SPSS macro by Enzmann [2002]; the R packages `psych` (Revelle, 2018), `polycor` (Fox, 2019), or `sirt`; or, because the tetrachoric correlation is a special case of the polychoric correlation, SAS’s `PLCORR` keyword with `PROC FREQ`.)

An exploratory principal axis analysis of \mathbf{R}_T yields a one factor solution ($\lambda_1 = 2.1177$, $\lambda_2 = 0.1626$, $\lambda_3 = 0.0742$, $\lambda_4 = 0.0369$). If we specify the extraction of a single factor, we obtain a common factor that accounts for 42.4% of the common variance with the following factor loadings (a_j s) for items 1 through 5: $a_1 = 0.5258$, $a_2 = 0.76660$, $a_3 = 0.6884$, $a_4 = 0.6748$, and $a_5 = 0.5100$. The difference between \mathbf{R}_T and the reproduced \mathbf{R}_T yields residuals that are all less than $|0.0005|$.

Given that the factor loadings are the biserial correlations of the responses with θ (Lord, 1980), we can use the factor loadings with Equation C.12 (Appendix C “Extend-

TABLE G.1. Tetrachoric Correlation Matrix, R_T , for the Mathematics Test Data

Item intercorrelations				
1	2	3	4	5
1.000				
0.453	1.000			
0.368	0.502	1.000		
0.313	0.519	0.476	1.000	
0.240	0.373	0.368	0.369	1.000

ing the Two-Parameter Normal Ogive Model to a Multidimensional Space”) to obtain estimates of α ; $\hat{\delta}_j$ s are obtained via Equation C.16. These α s are on the normal metric. Using Equation C.12 we obtain $\hat{\alpha}_1 = 0.6181$, $\hat{\alpha}_2 = 1.1938$, $\hat{\alpha}_3 = 0.9490$, $\hat{\alpha}_4 = 0.9145$, and $\hat{\alpha}_5 = 0.5928$. (As a comparison the BILOG normal metric estimates are $\hat{\alpha}_1 = 0.733$, $\hat{\alpha}_2 = 1.199$, $\hat{\alpha}_3 = 0.928$, $\hat{\alpha}_4 = 0.922$, and $\hat{\alpha}_5 = 0.587$; these estimates are obtained using the command file shown in Table 5.1, but with the subcommand LOG omitted from the GLOBAL line.) If we place our $\hat{\alpha}$ s on the logistic metric by multiplying them by D , we obtain $\hat{\alpha}_1 = 1.0519$, $\hat{\alpha}_2 = 2.0319$, $\hat{\alpha}_3 = 1.6151$, $\hat{\alpha}_4 = 1.5564$, and $\hat{\alpha}_5 = 1.0090$. The correlation between these $\hat{\alpha}$ s with those from the Chapter 5 example, “Application of the 2PL Model to the Mathematics Data, MMLE, BILOG-MG,” shows a strong linear relationship between this approach and those obtained by MMLE, $r = 0.9784$.

INFINITE ITEM DISCRIMINATION PARAMETER ESTIMATES

In some situations it is possible to experience difficulty in estimating an item’s discrimination parameter with either JMLE or MMLE. Specifically, for some items the estimate of α may drift off to infinity. The situation where α does not have a finite estimate, is an example of a *Heywood case* (i.e., an improper solution); for example, see Bock and Aitkin (1981), Christofferson (1975), and Swaminathan and Gifford (1985). To explain why this difficulty in estimating α is related to factor analysis’s Heywood case, recall that the total variance (σ^2) of each variable can be decomposed into variability that is in common across variables (*common variance*) plus variance specific to the variable (*specific variance*) and *error variance*. Specific and error variances collectively constitute *unique variance*. The complement of an item’s unique variance is its communality, h_j^2 ; that is, $h_j^2 = 1 - \text{unique } \sigma_j^2$. A variable’s communality specifies the proportion of a variable’s variance that is attributable to the common factors.

Although the bounds for h_j^2 are 0.0 and 1.0, the estimate of h_j^2 , \hat{h}_j^2 , may not be within these bounds. Because h_j^2 is a measure of common variance, the estimate reflects

negative common variance whenever $\hat{h}_j^2 < 0.0$. In addition, if $\hat{h}_j^2 > 1.0$, then one has a Heywood case (Heywood, 1931). In terms of variance, a Heywood case reflects negative unique variance, because for the equality in “ $1 = h_j^2 + \text{unique } \sigma_j^2$ ” to hold, the item’s unique σ_j^2 must be negative.

We can estimate a communality using the triad approach (Harman, 1960)

$$\hat{h}_j^2 = \frac{r_{jk}r_{jl}}{r_{kl}}, \quad (\text{G.1})$$

where k and l are the two variables with the highest correlation with the variable of interest. For example, if we apply this approach to the values in Table G.1, we obtain an $\hat{h}_2^2 = 0.5473$ for the second item. However, if we change r_{34} to be 0.25 (Table G.1), then $\hat{h}_2^2 = 1.0422$ (i.e., a Heywood case) and its corresponding unique σ_j^2 is negative ($= 1 - 1.0422 = -0.0422$).

Because h_j^2 equals the sum of squared loadings across the F factors, that is

$$\hat{h}_j^2 = \sum_f a_{jf}^2 \quad (\text{G.2})$$

and, with a unidimensional situation, $\sqrt{\hat{h}_j^2} = a_j = r_{j\theta}$, we can use the communality for item j to estimate α_j via Equation C.12. Applying Equation C.12 to item 2 produces a nonfinite $\hat{\alpha}_2$. The relationship between a_j (or h^2) and α_j is best represented by a J-curve such that as a_j (or h^2) approaches 1.0 α_j goes to ∞ . Consequently, if \hat{a}_j (or \hat{h}^2) ≥ 1 , then α_j does not have a finite value.

This difficulty in estimating α_j does not occur in all data sets. In addition, by using a prior distribution for the estimation of α_j one may avoid the problem of obtaining a nonfinite $\hat{\alpha}_j$. An alternative strategy is to impose an upper limit on the values that the $\hat{\alpha}_j$ may take on (i.e., a kludge). This is the approach used in LOGIST. Unless otherwise specified by the user, BILOG imposes a prior distribution when estimating α_j with the 2PL and 3PL models; BILOG also imposes a prior for estimating the IRF’s lower asymptote, χ_j , for the 3PL model. The use of a prior in estimating α_j can be seen in the Phase 2 output in the CALIBRATION PARAMETERS section, the line PRIOR DISTRIBUTION ON SLOPES: YES. Although, in general, the use of a prior distribution (i.e., a Bayesian approach) produces estimates that may be regressed toward the prior’s mean, the use of a prior with discrimination parameter estimation “has less serious implications than in the case of” (Lord, 1986, p. 161) person and item location parameters.

EXAMPLE: NOHARM UNIDIMENSIONAL CALIBRATION

In Chapter 3 we mention that NOHARM can provide not only dimensionality information, but also calibration results. In this section we discuss the two-parameter calibration results that were omitted from Tables 3.1 and 3.10; NOHARM may also be used for obtaining results for the one-parameter and the three-parameter models. The input file shown in Table 3.9 produced the output shown below in Table G.2.

TABLE G.2. One-Dimensional Output Including Item Parameter Estimates**NOHARM4 results**

```

                                N O H A R M
                                Fitting a (multidimensional) Normal Ogive
                                by Harmonic Analysis - Robust Method

Input File : math1Dcasedata.cmd
Title : EXPLORATORY ANALYSIS, math.dat (raw data), 1D

Number of items      = 5
Number of dimensions = 1
Number of subjects   = 19601

An exploratory solution has been requested.

Sample Product-Moment Matrix

      1      2      3      4      5
1    0.887
2    0.607  0.644
3    0.531  0.442  0.566
4    0.401  0.352  0.317  0.427
5    0.360  0.302  0.275  0.223  0.387

:

Final Constants
      1      2      3      4      5
1.438  0.551  0.229 -0.256 -0.335

Final Coefficients of Theta
      1
1    0.637
2    1.106
3    0.947
4    0.966
5    0.610

:

Threshold Values
      1      2      3      4      5
1.213  0.369  0.166 -0.184 -0.286

Unique Variances
      1      2      3      4      5
0.711  0.450  0.527  0.517  0.729

Factor Loadings
      1
1    0.537
2    0.742
3    0.688
4    0.695
5    0.521

```

(continued)

TABLE G.2. (continued)

LORD'S PARAMETERIZATION - for the unidimensional case

=====

Vector A : Discrimination parameters

1	2	3	4	5
0.637	1.106	0.947	0.966	0.610

Vector B : Difficulty parameters

1	2	3	4	5
-2.258	-0.498	-0.242	0.265	0.550

noharm.sirt results

```
> noharmld=noharm.sirt(mathdata,dimensions=1,lower=0,optimizer="optim",
  reliability=T)
> summary(noharmld)
-----
sirt 3.4-64 (2019-05-03 18:33:11)
R version 3.6.0 (2019-04-26) i386, mingw32 | nodename=RRRR-PC | login=rj

Call:
noharm.sirt(dat = mathdata, dimensions = 1, lower = 0, optimizer = "optim",
  reliability = T)
Date of Analysis: 9999-99-99 17:38:39
Time difference of 0.0500021 secs
Computation Time: 0.0500021

Function 'noharm.sirt'

:
RMSEA                                : 0.028

Green-Yang Reliability Omega Total : 0.633
:

Item Parameters - Latent Trait Model (THETA) Parametrization
Loadings, Constants, Asymptotes and Descriptives

      F1 final.constant lower upper item.variance    N    p
i01 0.637          1.438    0     1          1.406 19601 0.887
i02 1.106          0.551    0     1          2.223 19601 0.644
i03 0.947          0.229    0     1          1.897 19601 0.566
i04 0.966         -0.256    0     1          1.933 19601 0.427
i05 0.610         -0.335    0     1          1.372 19601 0.387

Item Parameters - Common Factor (DELTA) Parametrization
Loadings, Thresholds, Uniquenesses and Asymptotes

      F1 threshold lower upper uniqueness
i01 0.537     -1.213    0     1          0.711
i02 0.742     -0.369    0     1          0.450
i03 0.688     -0.166    0     1          0.527
i04 0.695      0.184    0     1          0.517
i05 0.521      0.286    0     1          0.729
```

(continued)

TABLE G.2. (continued)

```

--- Parameter table ---
  mat row col index fixed   est lower
  1   F   1   1     1     0 0.637 -Inf
  2   F   2   1     2     0 1.106 -Inf
  3   F   3   1     3     0 0.947 -Inf
  4   F   4   1     4     0 0.966 -Inf
  5   F   5   1     5     0 0.610 -Inf
  6   P   1   1     NA    1 1.000  NA
    
```

Chapter 10’s “Estimation of the M2PL Model” section contains a brief overview of NOHARM’s estimation approach. In the current context, the unidimensional two-parameter model may be seen as a special case of the M2PL. The reader interested in greater estimation detail is referred to McDonald (1967, 1997) and McDonald and Mok (1995). Our NOHARM results are on the normal metric. As such, if we wish to have the $\hat{\alpha}_j$ s on the logistic metric we would need to multiply the $\hat{\alpha}_j$ s by $D = 1.702$.

We first discuss NOHARM4 and then `noharm.sirt`. In NOHARM4 our estimates are found at the end of the output in the section labeled `LORD’S PARAMETERIZATION` - for the unidimensional case. The subsection `Vector A : Discrimination parameters` contains the item discrimination estimates. Our item discrimination estimates are $\hat{\alpha}_1 = 0.637$, $\hat{\alpha}_2 = 1.106$, $\hat{\alpha}_3 = 0.947$, $\hat{\alpha}_4 = 0.966$, and $\hat{\alpha}_5 = 0.610$. The item locations estimates are found in the subsection labeled `Vector B: Difficulty parameters`. As can be seen, the item location estimates are $\hat{\delta}_1 = -2.258$, $\hat{\delta}_2 = -0.498$, $\hat{\delta}_3 = -0.242$, $\hat{\delta}_4 = 0.265$, and $\hat{\delta}_5 = 0.550$. The values in these two subsections are determined from values presented above in the output. For instance, the values labeled as `Final Coefficients of Theta` are the item discrimination estimates (e.g., $\hat{\alpha}_1 = 0.637$, $\hat{\alpha}_2 = 1.106$, etc.). Dividing these values into the negative of the corresponding values labeled `Final Constants` produces the item location estimates (i.e., $\hat{\delta}_j = -\hat{\gamma}_j / \hat{\alpha}_j$). Thus, for item 1 we have $\hat{\delta}_1 = -(1.438/0.637) = -2.258$, for item 2 we have $\hat{\delta}_2 = -(0.551/1.106) = -0.498$, and so on. The Pearson correlation coefficients between the NOHARM estimates and those from BILOG (Chapter 5) are 0.9574 for the $\hat{\alpha}_j$ s and 0.9995 for the $\hat{\delta}_j$ s.

The values from the `Threshold Values` section are estimates that correspond to the z_{t_j} shown in Figure C.5. As a result, these threshold values are related to the items’ P_j s and may be obtained in the way described above in Appendix C “The Relationship between IRT Statistics and Traditional Item Analysis Indices” (i.e., using the inverse of the cumulative unit normal distribution for a P_j). NOHARM obtains these values by taking the `Final Constants` values and multiplying them by the square root of the values in the `Unique Variances` section. For example, the threshold value for item 1 would be obtained as $1.438(\sqrt{0.711}) = 1.2125$.

As discussed above in the “Using Principal Axis for Estimating Item Discrimination” section, item discriminations may be estimated by using factor loadings. Using the values provided in the `Factor Loadings` section and dividing them by the square

root of the unique variances provides the item discrimination estimates. For instance, for item 1 the corresponding factor loading is 0.537 and the unique variance is 0.711. Therefore, 0.537 divided by the square root of 0.711 equals 0.637.

Unlike NOHARM4, `noharm.sirt` provides a reliability estimate. As discussed above, nonlinearity is likely to be present when working with dichotomous data. The `noharm.sirt` reliability estimate (Green & Yang, 2009) allows for nonlinear relationships between items and the latent variable. As can be seen, for our data with only five items our reliability estimate is 0.633. As a comparison, Cronbach's alpha is 0.6077. However, see Sijtsma (2009) for comments about issues with Cronbach's alpha.

To obtain our item location estimates we can use the relationships discussed above. Examining the Item Parameters - Latent Trait Model (THETA) Parametrization [sic] table we find our items' P_{js} (p ; e.g., $P_1 = 0.887$) and the item discrimination estimates (F1), $\hat{\alpha}_1 = 0.637$, $\hat{\alpha}_2 = 1.106$, $\hat{\alpha}_3 = 0.947$, $\hat{\alpha}_4 = 0.966$, and $\hat{\alpha}_5 = 0.610$. To obtain our item location estimates we divide these values into the corresponding final.constant values. For example, $\hat{\delta}_1 = -(1.438/0.637) = -2.257$, and so forth. From the Item Parameters - Common Factor (DELTA) Parametrization [sic] table we find our factor loadings (F1), threshold values, and unique variances that could also be used to obtain the item discriminations. Chapter 3 discusses how to interpret NOHARM's fit information to assess model-data fit.

AN APPROXIMATE CHI-SQUARE STATISTIC FOR NOHARM

In addition to using NOHARM's RMS and GFI to determine data dimensionality, one could also use Maydeu-Olivares and Joe's (2006) M_2 statistic or Gessaroli and De Champlain's (1996) *approximate* chi-square statistic, χ_{GD}^2 . The former statistic is distributed as a χ^2 , although at present its calculation is not easily performed. The latter statistic tests the null hypothesis that the off-diagonal elements of the residual matrix are zero (i.e., the number of dimensions is correctly specified in the model). Although Maydeu-Olivares (2001) has indicated that Gessaroli and De Champlain's statistic is not distributed as a χ^2 , it may still be useful to provide some rough evidence to support a unidimensional or a multidimensional model of the data. For example, research has found χ_{GD}^2 to be useful to correctly identify dimensionality with sample sizes of 250, 500, and 1000 (e.g., De Champlain & Gessaroli, 1998; Gessaroli & De Champlain, 1996; also see Finch & Habing, 2005).

Because χ_{GD}^2 is based on evaluating the off-diagonal elements of the symmetric residual matrix, there are $(L^2 - L)/2$ unique off-diagonal item pairs. To calculate χ_{GD}^2 we use the items' observed proportion of responses of 1 (P^O) and the residual matrix's values to obtain an estimated residual correlation for each unique item pair. This estimated "residual correlation" between items j and v is

$$r_{jv}^* = \frac{P_{jv}^R}{\sqrt{[P_j^O(1-P_j^O)][P_v^O(1-P_v^O)]}}, \quad (\text{G.3})$$

where, P_j^O is item j 's observed proportion of responses of 1, P_v^O is the observed proportion of responses of 1 for item v , and P_{jv}^R is the residual proportion of individuals who responded 1 to both items j and v (i.e., P_{jv}^R is the difference between the observed proportion of individuals who responded 1 to both items and what would be expected on the basis of the model). The P_{jv}^R s come from the NOHARM program's residual matrix and the P^O s may be obtained from the main diagonal of \mathbf{P} ; see Equation 3.4.

Prior to summing the estimated residual correlations across all unique item pairs, each residual correlation is transformed using the Fisher r-to- \dot{z} transformation

$$\dot{z}_{jv} = 0.5 \ln \left[\frac{1 + r_{jv}^*}{1 - r_{jv}^*} \right] \quad (\text{G.4})$$

to stabilize their variances. The weighted sum of the squared Fisher \dot{z}_{jv} s gives an approximate chi-square for the solution

$$\chi_{GD}^2 = (N - 3) \sum_{j=2}^L \sum_{v=1}^{j-1} (\dot{z}_{jv})^2, \quad (\text{G.5})$$

where N is the sample size and L is the instrument length. The null hypothesis is that after fitting the model the resulting residual correlation matrix contains off-diagonal elements equal to zero. χ_{GD}^2 would be calculated for each dimensional solution and its significance evaluated with $df = (L^2 - L)/2 - \text{Nparm}$, where Nparm is the number of estimated independent parameters in the solution's nonlinear factor analytic model. Failure to reject null hypothesis provides evidence that the corresponding dimensional solution represents a reasonable structure for the data vis á vis the observed correlations. Determining Nparm 's value depends on the number of dimensions, the number of items, and the number of constraints. In the exploratory case the simplest way to obtain Nparm is to count the number of unique estimates in NOHARM's `Final Coefficients of Theta` table.

As mentioned above, the gist of the null hypothesis is that one has correctly specified the number of dimensions (e.g., unidimensionality) in the calibration model. As a result, one would like to obtain a nonsignificant χ_{GD}^2 in order to have supporting evidence. De Champlain and Tang (1997) have suggested using the number and proportion of $|z_{jv}^*| > 2$ to provide additional evidence supporting the result of the hypothesis test, as well as for diagnosing why a particular solution does not correspond to the specified dimensional model.

The use of the $(N - 3)$ weighting factor in Equation G.5 shows this statistic's performance is influenced by sample size. The degree of influence is greater for extremely large or extremely small samples than for sample sizes of, say, 500 to 1000. For our example's large sample size we would expect its influence to lead us to falsely reject the correct dimensional solution. In addition to `sirt.noharm`, software for calculating χ_{GD}^2 and its probability is available from De Champlain and Tang (1997). Alternatively, a spreadsheet program could be used to calculate the χ_{GD}^2 and its significance evaluated using critical values or a function like EXCEL's `CHIDIST`.

RELATIVE EFFICIENCY, MONOTONICITY, AND INFORMATION

If we apply a monotonic transformation to θ to create a new metric θ^* then the transformed information function ($I(\theta^*, y)$) equals the untransformed information function ($I(\theta, y)$) divided by the square of the derivative of the transformation (Lord, 1974b)²

$$I(\theta^*, y) = \frac{I(\theta, y)}{\left[\frac{\partial \theta^*}{\partial \theta} \right]^2}. \quad (\text{G.6})$$

Equation G.6 implies that the location of the maximum information may be changed by a transformation of the metric. Moreover, the shape of the information function may be changed as $\partial \theta^* / \partial \theta$ varies across the continuum (Lord, 1980).

In contrast, the RE is invariant under a monotonic transformation. For example, assume that a monotonic transformation is applied to the 1PL and 2PL model calibration results from Chapter 5. This monotonic transformation is based on the exponential function and is $\xi^* = Ke^{c\xi}$ with $\alpha_j^* = \alpha_j/c$ where ξ is either θ or δ_j , K and c are two constants that, for convenience, are set to 1, and the asterisks indicate the parameter estimates are on the transformed metric (cf. Lord, 1980). For convenience, the item parameter estimates from Chapter 5 are presented in Table G.3.

The information functions on the transformed metric are given by the 1PL and 2PL models' information functions from the untransformed metric (e.g., Figure 5.8) divided by $(e^\theta)^2$ (i.e., for this transformation the denominator of Equation G.6 is $(e^\theta)^2$). These transformed 1PL and 2PL models' information functions $I(\theta^*, 2PL)$ and $I(\theta^*, 1PL)$, respectively, are shown in Figure G.1. Our latent variable metric shifts from $-4 \leq \theta \leq 4$ to $0.05 \leq \theta^* \leq 55$. It is clear these information functions are neither unimodal nor symmetric and do not have the same maxima as those in Figure 5.8. Moreover, both models provide their maximum information at the lower end of the transformed metric. However, the RE plot of these information functions (Figure G.2) shows the same pattern as seen in Figure 5.9. Therefore, relative efficiency is unaffected by a monotonic transformation of the metric (Lord, 1980).

TABLE G.3. 1PL and 2PL Models' Item Parameter Estimates for the Mathematics Data

Item	1PL model ($\hat{\alpha} = 1.421$)	2PL model	
	$\hat{\delta}_i$	$\hat{\alpha}_i$	$\hat{\delta}_i$
1	-1.925	1.226	-2.107
2	-0.581	1.992	-0.499
3	-0.264	1.551	-0.254
4	0.284	1.544	0.270
5	0.443	0.983	0.560

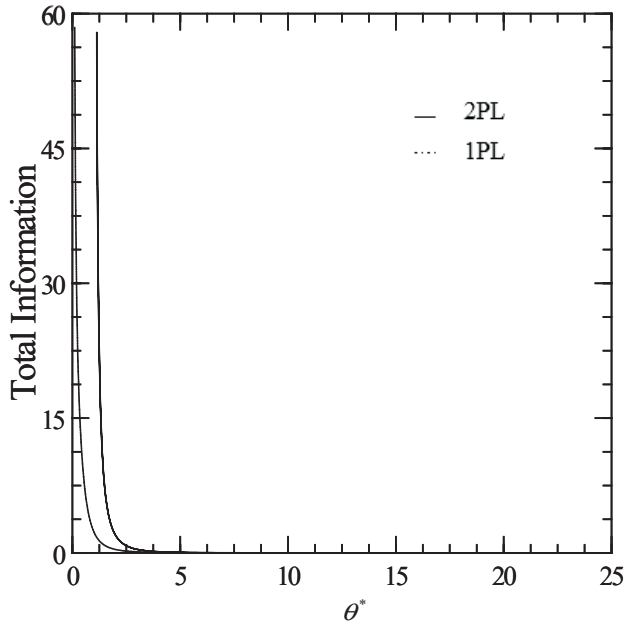


FIGURE G.1. Total information function for 1PL and 2PL model monotonically transformed metrics. 1PL information function ($I(\theta^*, 1PL) = 1PL$) shifted to the right by 1 unit to avoid superimposition with 2PL information function ($I(\theta^*, 2PL) = 2PL$).

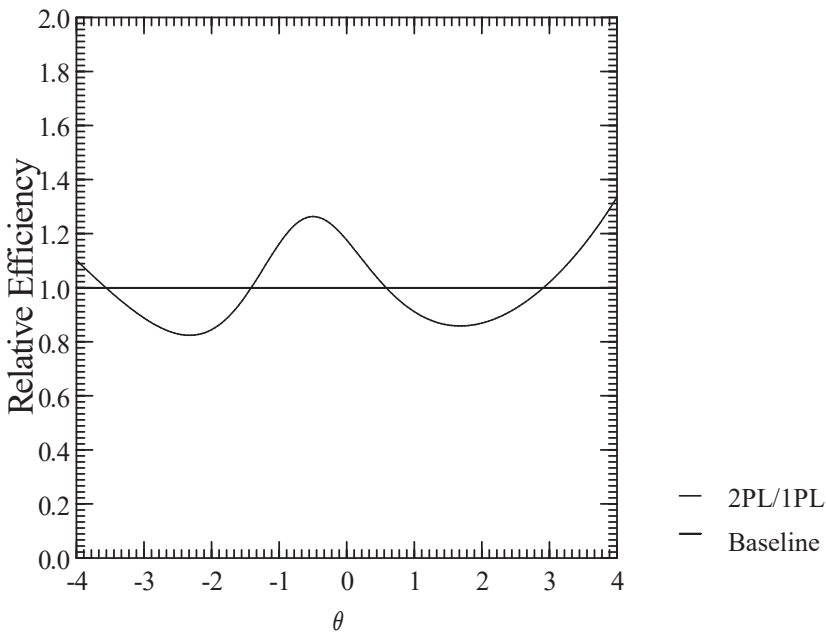


FIGURE G.2. Relative efficiency plot for monotonically transformed metric based on 1PL* and 2PL* models.

FORTRAN FORMATS

A FORTRAN format statement specifies how a FORTRAN program (e.g., PARSCALE, BILOG, EQUATE) should read and interpret the contents of a data file. This format begins and ends with a parenthesis. Everything within the parentheses consists of format descriptors. Some common general descriptors are presented in Table G.4; there are many more.

As an example, assume that our format statement is (10A1, T1, 5(1X,1A1)). To interpret this statement the parenthesized term is decomposed into the components separated by the commas. Therefore, beginning in column 1 our interpretation is

<u>Format descriptor</u>	<u>Interpretation</u>
“10A1”	means read 10 alphanumeric variables that are each 1 column wide
“T1”	means go to column 1
“5(1X,1A1)”	means repeat the parenthesized terms 5 times

The parenthesized term is further broken down into two segments:

“1X”	means skip 1 column
“1A1”	means read 1 alphanumeric variable that is 1 column wide

Repeat the above two segments 5 times

That is, “5(1X,1A1)” is a shorthand way of specifying “1X,1A1,1X,1A1,1X,1A1,1X,1A1,1X,1A1.”

TABLE G.4. Some General FORTRAN Format Descriptors

Code	Meaning	Syntax	Example	Example's interpretation
A	specifies alphanumeric data	nAc	5A1	Read 5 columns as alphanumeric and each alphanumeric occupies one column
X	skip	nX	3X	Skip 3 columns
F	specifies floating point data	nFc.d	2F4.2	Read 2 real numbers that each occupy at most 4 columns and contain 2 decimal places
I	specifies integer data	nIc	1I6	Read 1 integer that occupies at most 6 columns
T	tab	Tc	T23	Go to column 23
/	skip to a new line for the current record			

Note: n = repeat factor, c = total number of columns, d = number of decimal places.

Let us apply this (10A1,T1,5(1X,1A1)) to a line of data where the data begin with a space and successive values are separated by a space

0 0 0 1 1

(i.e., using \mathfrak{b} to indicate a blank space our data are “ $\mathfrak{b}0\mathfrak{b}0\mathfrak{b}0\mathfrak{b}1\mathfrak{b}1$ ”). Because the format (10A1,T1,5(1X,1A1)) first specifies the reading of 10 characters, each occupying one column (i.e., 10A1), the entire string “0 0 0 1 1” is read with the values interpreted as alphanumeric (e.g., characters). The next format descriptor, T1, instructs the program to return to column 1 (i.e., the first blank space). The final component, 5(1X,1A1), specifies the rereading of the same 10 characters (i.e., “0 0 0 1 1”). That is, skip a column and then read a column and repeat this five times. Again, the values are interpreted as alphanumeric (e.g., characters).

Some calibration programs (e.g., BILOG) allow the user to specify a case identification field. Specifically, the first format descriptor is associated with this identification field (e.g., 10A1 refers to the case identification field). For instance, with the format (10A1,T1,5(1X,1A1)) we are using an individual’s response pattern as their identification field and then specifying where to find the responses to be analyzed (i.e., T1, 5(1X,1A1)).

Sometimes each case has identification information, such as the person’s Social Security number (SSN). For example, assume that each line of data consists of a person’s SSN (with hyphens) followed by a space and then their response vector (e.g., 123-45-6789 00011). For this layout the, say, BILOG format statement would be (11A1,1X,5A1). The 11A1 would read the SSN (including hyphens), 1X would skip the blank space following the SSN, and the 5A1 would read five consecutive columns of responses. Although the example’s responses consist of the integers 0 and 1, by treating these data as alphanumeric (i.e., using the A in the format) the program allows flexibility in the data coding. For instance, our format statement could be used with a response vector that contains letters, such as “T” and “F” (e.g., 123-45-6789 FFFTT).

ODDS, ODDS RATIOS, AND LOGITS

Probabilities may be expressed in terms of the odds of an event occurring. Some treatments of IRT use the odds of an event in their discussions and we use them in Chapters 9 and 12. The *odds* of an event b occurring (i.e., a “success”) is given by

$$\text{odds}(b) = \frac{p(b)}{1 - p(b)} \quad (\text{G.7})$$

Equation G.7 states that the odds of b occurring is equal to the ratio of the probability of the event b occurring to the probability that the event b does not occur. In other words, the odds of an event expresses the likelihood of the event occurring relative to its not occurring. If the event b is as likely to occur as to not occur, then the $\text{odds}(b) = 1$ (i.e., $p(b) = 1 - p(b) = 0.5$). Obviously, if the event b is less likely to occur than to not occur, then the $\text{odds}(b)$ are less than 1. Conversely, if the event b is more likely to occur than to not occur, then the $\text{odds}(b)$ are greater than 1.

By rearranging Equation G.7 one may obtain the probability of event b occurring expressed in terms of the odds of b by

$$p(b) = \frac{\text{odds}(b)}{1 + \text{odds}(b)}. \quad (\text{G.8})$$

Equation G.8 also shows that when an event b is as likely to occur as not (i.e., the $\text{odds}(b) = 1$), then $p(b) = 0.50$.

As an example of calculating odds, assume that the probability of b occurring (i.e., a “success”) is 0.75. The corresponding odds of b occurring are 3 (i.e., $0.75/0.25$) or, alternatively, the odds are 3 to 1 that b occurs as opposed to not occurring; odds are implicitly compared to 1. Conversely, the odds of b not occurring (\bar{b}) would be

$$\text{odds}(\bar{b}) = \frac{1 - p(b)}{p(b)}. \quad (\text{G.9})$$

In terms of our example, the odds of b not occurring is $0.25/0.75$ or 1 to 3 (i.e., 0.333 to 1).

As a second example, let $p(b)$ be given by

$$p(b) = p(x_j = 1 | \theta, \alpha, \delta_j) = \frac{e^{\alpha(\theta - \delta_j)}}{1 + e^{\alpha(\theta - \delta_j)}}. \quad (\text{G.10})$$

That is, the probability of b occurring refers to a response of 1 (i.e., a “success”) on item j . Therefore, we can talk about the odds of a response of 1 occurring versus a response of 0 on an on item j . For instance, if $\theta = -2.0$ and the item is located at -1.3 (i.e., $\delta_j = -1.3$; $\alpha = 1$), then according to Equation G.10 the probability of a response of 1 (i.e., success) is 0.3318. Expressing this probability in terms of odds we have that the odds of success on the item are roughly 1 to 2. That is, the odds are

$$\text{odds}(b) = \frac{p(b)}{1 - p(b)} = \frac{0.3318}{0.6682} = 0.4966$$

In words, we have that the odds of a correct response are 0.4966 to 1 or multiplying each of these values by 2 gives us odds of 1 to 2. If this is a proficiency item, then these odds indicate that for people located at -2 we expect a correct response to this item for every two incorrect responses. Conversely, the odds of an *incorrect* response is approximately 2 to 1. That is,

$$\text{odds}(\bar{b}) = \frac{1 - p(b)}{p(b)} = \frac{0.6682}{0.3318} = 2.0138$$

Given that probabilities are always positive and sum to 1 for all the events in an event class, the odds of an event must be positive. Moreover, although probabilities fall within the range 0 to 1, their conversion to odds results in the range of odds being 0 to ∞ with a value of 1 reflecting no difference between the event occurring and not occurring. Because of this asymmetry in the odds scale (i.e., the “no difference point” occurs at 1) the odds of an event are sometimes transformed to the (natural) logarithmic scale (i.e., $\ln(\text{odds}(b))$). On the log scale a value of 0 reflects no difference between the

event occurring and not occurring, a positive values indicate that the odds of success are greater than of failure, and negative values reflect that the odds of failure are greater than the odds of success. This *logit transformation* gives the *log odds* or the *logit* of the event occurring. Therefore, applying this transformation to Equation G.7 one has

$$\ln(\text{odds}(b)) = \text{logit}(p(b)) = \ln \left[\frac{p(b)}{1-p(b)} \right]. \quad (\text{G.11})$$

The transformation “ $\text{logit}(p(b))$ ” is sometimes called the logit link function. By substituting Equation G.10 for $p(b)$ in Equation G.11 one obtains, upon simplification, that the odds for a response of 1 (i.e., event b) occurring are³

$$\text{odds}(b) = \frac{p(b)}{1-p(b)} = \frac{\frac{e^{\alpha(\theta-\delta)}}{1+e^{\alpha(\theta-\delta)}}}{\frac{1}{1+e^{\alpha(\theta-\delta)}}} = \left[\frac{e^{\alpha(\theta-\delta_j)}}{1+e^{\alpha(\theta-\delta_j)}} \right] \left[\frac{1+e^{\alpha(\theta-\delta_j)}}{1} \right] = e^{\alpha(\theta-\delta_j)}. \quad (\text{G.12})$$

As a result, the log odds are (after applying the quotient rule and substitution of Equation G.10)

$$\begin{aligned} \text{logit}(p(b)) &= \ln \left[\frac{p(b)}{1-p(b)} \right] = \ln(p(b)) - \ln(1-p(b)) \\ &= \ln \left[\frac{e^{\alpha(\theta-\delta)}}{1+e^{\alpha(\theta-\delta)}} \right] - \ln \left[1 - \frac{e^{\alpha(\theta-\delta)}}{1+e^{\alpha(\theta-\delta)}} \right] \\ &= \ln \left[\frac{e^{\alpha(\theta-\delta)}}{1+e^{\alpha(\theta-\delta)}} \right] - \ln \left[\frac{1}{1+e^{\alpha(\theta-\delta)}} \right] \\ &= \left(\ln \left[e^{\alpha(\theta-\delta)} \right] - \ln \left[1+e^{\alpha(\theta-\delta)} \right] \right) - \left(\ln[1] - \ln \left[1+e^{\alpha(\theta-\delta)} \right] \right) \quad (\text{G.13}) \\ &= \ln \left[e^{\alpha(\theta-\delta)} \right] - \ln \left[1+e^{\alpha(\theta-\delta)} \right] + \ln \left[1+e^{\alpha(\theta-\delta)} \right] \\ &= \ln \left[e^{\alpha(\theta-\delta_j)} \right] \\ &= \alpha(\theta - \delta_j) = \alpha\theta + \gamma_j, \end{aligned}$$

where $\gamma_j = -\alpha\delta_j$.

Equation G.13's term “ $\gamma_j + \alpha\theta$ ” shows that the logit transformation has the effect of linearizing the nonlinear relationship between the continuous θ and the probability of the event of a response of 1 (i.e., $p(b) = p(x_j = 1 | \theta, \alpha, \delta_j)$). As such, α reflects the slope of the logit regression line and may be interpreted as the change in the logit corresponding to a one-unit change in θ . The constant or intercept, γ_j , is simply the predicted logit value when $\theta = 0$. Alternatively, $\gamma_j + \alpha\theta$ may be interpreted as the term “ $\alpha\theta$ ” specifying how much better in prediction one can do over the baseline odds pro-

vided by the intercept, γ_j . Stated another way, and for simplicity letting $\alpha = 1$, the logit $\gamma_j + \alpha\theta$ indicates how much knowing *only* the person's location improves our capacity to predict a response of 1 over and above just knowing the item's location (i.e., $\delta_j = -\gamma_j$).

In terms of IRT, " $\alpha(\theta - \delta_j)$ " (or $\gamma_j + \alpha\theta$) specifies the *logit success* and equals the weighted difference between the person and item locations; a logit is also known as a logistic deviate. Therefore, a person's θ in logits is their natural log odds for obtaining a response of 1 on items of the kind chosen to define the zero point on the scale, and an item's δ in logits is its natural log odds for a response of 0 on that item from persons with zero ability (Wright & Stone, 1979).

Just as we can obtain the probability of b from the odds of b , we can rearrange Equation G.13 to obtain the probability of b from the log odds of b (i.e., logits)

$$\ln \left[\frac{p(b)}{1-p(b)} \right] = \alpha(\theta - \delta_j).$$

Applying the natural exponential function to both sides, one has

$$\text{odds}(b) = \left[\frac{p(b)}{1-p(b)} \right] = e^{\alpha(\theta - \delta_j)} = e^{\alpha\theta + \gamma_j} = e^{\gamma_j} e^{\alpha\theta}.$$

Solving for $p(b)$, one obtains

$$p(b) = \frac{e^{\alpha\theta + \gamma_j}}{1 + e^{\alpha\theta + \gamma_j}} = \frac{e^{\alpha(\theta - \delta_j)}}{1 + e^{\alpha(\theta - \delta_j)}}. \quad (\text{G.14})$$

Because $e^{\alpha(\theta - \delta_j)}$ equals the *odds*(b) (see Equation G.12), Equation G.14 is the conversion of the odds of a response of 1 to their corresponding probability (i.e., Equation G.8).

The " $\gamma_j + \alpha\theta$ " term is the simple linear regression (SLR) model for predicting the criterion Y 's conditional means (i.e., the means of Y 's distribution for fixed values of θ). In other words, the SLR prediction equation is

$$\mathcal{E}(Y | \theta) = \beta_0 + \beta_1 \theta, \quad (\text{G.15})$$

where $\beta_0 = \gamma_j$ and $\beta_1 = \alpha$; also see Chapter 13. Analogously, because the mean of a binary variable is the proportion of 1s, the proportions shown in Figure 2.2 may be considered a series of conditional means (i.e., $\mathcal{E}(x | \theta) = p(z)$) and Equation G.12 may be written as

$$\mathcal{E}(x | \theta) = p(\theta) = \frac{e^{\gamma + \alpha\theta}}{1 + e^{\gamma + \alpha\theta}}, \quad (\text{G.16})$$

where x is our binary criterion (response) variable and θ is our predictor. In short, Equation G.16 may be recognized as the typical representation of a logistic regression model.

We now discuss odds ratios. A natural extension of asking about the odds of a single event is to ask about how the odds of one event relate to the odds of another event. In this regard, the above ideas may be extended to describe the association between

the odds of two events. This measure of association is the odds ratio (OR or Ω). The odds ratio is, as the name implies, the ratio of two odds (e.g., $odds(b)$ and $odds(a)$). For instance, the odds ratio of b to a is

$$\Omega_{b,a} = \frac{odds(b)}{odds(a)} = \frac{p(b)/1-p(b)}{p(a)/1-p(a)} \quad (G.17)$$

An odds ratio is asymmetrical about 1 with a range of 0 to ∞ . An $\Omega = 1$ indicates that both events are equally likely, with a value less than 1 indicating that the odds of success for a are greater than the odds for b , and a value greater than 1 reflecting that the odds of success for b are greater than the odds for a . For instance, if $\Omega = 5$, then the odds of success for b is five times the odds of success for a . As is the case with odds, the odds ratio is sometimes transformed to a logarithmic scale ($\ln(\Omega)$) to eliminate its inherent asymmetry. The transformed odds ratio is centered at 0 (i.e., a 0 reflects no difference between the events occurring) with values greater than 0 reflecting that the log odds of success for b are greater than the log odds for a , and values less than 0 reflecting that the log odds of success for a are greater than the log odds for b .

THE PERSON RESPONSE FUNCTION

The *person response function* (PRF) provides a graphical approach to examining person-model fit; the PRF is also known as the *person characteristic curve* (Weiss, 1973). The idea of a PRF may be traced back to the works of Thorndike and Thurstone in the early part of the 20th century (Engelhard, 1990). The PRF presents the relationship of the probability of a person's response pattern and the item locations. In this regard, the PRF is the person analog to the item response function. Similar to person fit statistics, the PRF can be used to identify misfitting individuals. Additionally, the PRF may be used to identify a particular item or set of items for which person-item fit is problematic as well as to provide diagnostic information, such as inattention, guessing, identifying copying, and so on (Trabin & Weiss, 1983; Wright, 1977b). The performance of PRF has been compared with that of the I_z index (Nering & Meijer, 1998). They found that although I_z performed well, and in some cases better than PRF, the PRF was useful in determining the reason for a misfitting response vector.

In general, the PRF is assumed to be a nonincreasing function of the item δ_s . Departures from this monotonicity assumption are taken as indicators of person-model misfit for all or some subset of the instrument's items. However, this assumption may be unreasonable if the items are multidimensional or the items cannot be ordered in the same way for all individuals (Sijtsma & Meijer, 2001).

To examine person fit, we compare a person's observed PRF (OPRF) with their expected PRF (EPRF). Trabin and Weiss (1983) argue that the shape of the OPRF provides diagnostic information concerning guessing behavior, carelessness, the precision with which the person is measured, and dimensionality information. Figure G.3 contains

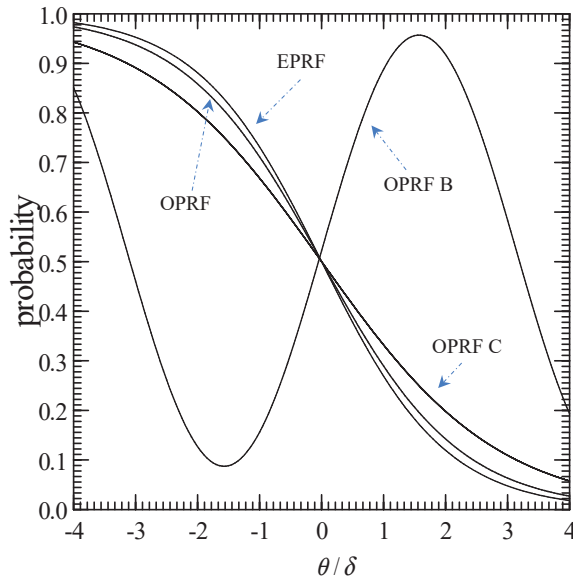


FIGURE G.3. Expected and three possible observed person response functions.

an example of each of these diagnoses and is adapted from Trabin and Weiss (1983). In the following, assume a proficiency assessment situation and the EPRF is appropriate for the three OPRFs. We would interpret the close agreement between the EPRF and OPRF curves as indicating good fit. Furthermore, the steepness of the OPRF reflects that this person is more precisely measured than they would have been if the OPRF had been less steep. Also shown are two additional problematic OPRFs labeled “OPRF B” and “OPRF C.” Because the right side of person’s OPRF C is greater than their EPRF, this person is correctly responding to items that they are expected to incorrectly answer. Trabin and Weiss interpret this to indicate guessing behavior on the items located at the upper end of the continuum. Moreover, because the observed proportion of correct responses for easy items (i.e., the left side of OPRF C) is less than would be expected according to the person’s EPRF, one has evidence of carelessness. OPRF B reflects a person exhibiting inconsistent response behavior because they are incorrectly answering easy items and correctly answering difficult items. Because this person’s OPRF reflects a deviation from a unidimensional response pattern, Trabin and Weiss (1983) suggest it reflects multidimensionality in the person.

One approach to obtaining the OPRF is to put the instrument’s item into strata ($s = 1, \dots, S$); it is preferable that all strata have the same number of items. Because each stratum consists of L_s items that are similar to one another in terms of location, the strata can be ordered on the basis of their average locations.⁴ Assuming dichotomous data and the strata approach, we can obtain an individual’s OPRF by calculating, for each stratum, the proportion of items for which the person had a response of 1. Subsequently, these proportions are plotted as a function of stratum location level. To obtain the individual’s EPRF one uses their $\hat{\theta}$ and the item parameter estimates to calculate the person’s p_j for

each item in each stratum. The average of these probabilities for the L_s items that define each stratum specifies the individual's expected proportion of Is for the stratum. These averaged probabilities are then plotted as a function of the strata's locations to obtain the EPRF for the individual. To form a curve one connects the average probabilities (or in the case of the OPRF, the proportions) for each stratum. The OPRF may be overlaid on the EPRF to facilitate comparison of the two curves. In some cases it may be necessary to smooth the OPRF to reduce irregularities by, for example, using spline smoothing. Alternative approaches for PRF analysis use kernel smoothing or logistic regression to avoid the creation of strata (cf. Emmons, Sijtsma, & Meijer, 2004).

With item fit analyses we suggest that one use a combination of graphical approaches (e.g., empirical vs. predicted IRFs) and statistical indices for determining item fit. This same philosophy can be applied to assessing person fit. Specifically, a numerical index is used to identify individuals who merit further inspection and the PRF is the graphical approach for performing this inspection.

There are various indices that could be used to identify individuals for whom OPRFs and EPRFs should be created and examined. Meijer and Sijtsma (2001) present a review of some of these indices and recommend the UB statistic and Klauer and Rettig's (1990) chi-square statistic.

The UB statistic (Smith, 1985) is

$$UB = \frac{1}{S-1} \sum_{s=1}^S \frac{(O_s - E_s)^2}{V_s}, \quad (G.18)$$

where $O_s = \sum_{j=1}^{L_s} x_j$, $E_s = \sum_{j=1}^{L_s} p_j$, and $V_s = \sum_{j=1}^{L_s} p_j(1-p_j)$. This statistic may be standardized by a cube root transformation (Smith, 1985). As such, a standard normal table could be used to provide screening values that would identify individuals who warrant further scrutiny.

An alternative to the UB statistic is Klauer and Rettig's (1990) chi-square statistic. Their standardized person fit statistic is asymptotically distributed as a χ^2 with the number of strata minus one as the *df*. Their statistic is

$$\chi_{SC}^2 = \sum_{s=1}^S \frac{[W_s(\hat{\theta})]^2}{I_s(\hat{\theta})}, \quad (G.19)$$

where for the 3PL model the available information for estimating θ , $I_s(\hat{\theta})$, is given by

$$I_s(\hat{\theta}) = \frac{\alpha_j^2(1-p_j)}{p_j[(p_j - \chi_j)^2 / (1 - \chi_j)^2]} \quad (G.20)$$

and

$$W_s(\hat{\theta}) = \sum_{j=1}^{L_s} \alpha_j \frac{\exp(\alpha_j(\hat{\theta} - \delta_j))}{1 + \exp(\alpha_j(\hat{\theta} - \delta_j))} [x_j - p_j]. \quad (G.21)$$

Although Equations G.18 and G.19 are also appropriate for the 1PL and 2PL models. That is, for the 2PL model $\chi_j = 0$ and for the 1PL model one sets $\alpha_j = 1$ and $\chi_j = 0$.

Klauer and Rettig (1990) have evaluated the significance of χ_{SC}^2 with a significance level of 0.10; a significant χ_{SC}^2 indicates a misfitting person. Other indices that can be used to identify individuals for further (graphical) scrutiny may be found in Drasgow, Levine, and Williams (1985), Levine and Drasgow (1988), Meijer and Sijtsma (2001), Reise (2000), and van der Flier (1982).

LINKING: A TEMPERATURE ANALOGY EXAMPLE

Linking is analogous to comparing temperature on the Celsius scale with temperature on the Fahrenheit scale. Because of differences in the origin and units of the two metrics, a meaningful direct comparison cannot be made between a temperature of, for example, 32 degrees on the Celsius metric, with a temperature of 32 degrees on the Fahrenheit scale without first transforming one metric to the other. This linear transformation (e.g., $^{\circ}F = 1.8(^{\circ}C) + 32$) amounts to (1) aligning the origins of the metrics to one another (i.e., the zero points) so that the “zero point” is equivalent in meaning on both scales (e.g., add 32 to the temperature on the Celsius scale so that $0^{\circ}C$ is equivalent to $32^{\circ}F$), and (2) transforming the units in one metric to be the same size as the units on the other metric (e.g., 1 unit on the Celsius scale is equivalent to 1.8 units on the Fahrenheit scale). In short, the linear transformation from one metric to another involves one constant having to do with the units and another constant dealing with the different origins.

To show the parallel between this analogy and IRT metric alignment, we link the Celsius scale to the Fahrenheit scale using “mean-sigma” approach (see Chapter 11). Table G.5 presents the annual monthly high temperature readings from Fargo, North Dakota and Tucson, Arizona in both Celsius ($^{\circ}C$) and Fahrenheit ($^{\circ}F$). For both data sets the Celsius scale is the initial metric and the Fahrenheit scale is the target metric; the temperature readings are analogous to item locations.

To transform the Fargo Celsius readings to Fahrenheit, we obtain the metric transformation coefficients for the Fargo data by Equation 11.7

$$\zeta = \frac{s_{\delta^*}}{s_{\delta}} = \frac{24.8}{13.8} = 1.80$$

and by Equation 11.6

$$\kappa = \bar{\delta}^* - \zeta(\bar{\delta}) = 51.3 - 1.80(10.7) = 32$$

Therefore, by substitution into Equation 11.1 we have

$$\xi^* = \zeta(\xi) + \kappa = 1.80(\xi) + 32$$

where $\xi = ^{\circ}C$ and $\xi^* = ^{\circ}F$.

In Chapter 11 we stated the transformation should be independent of the groups of individuals used to develop the transformation (i.e., the transformation is unique). To

TABLE G.5. Temperature Analogy for Metric Linking

Fargo Daily High			Tucson Daily High		
Month	°F	°C	Month	°F	°C
1	15.4	-9.2	1	63.5	17.5
2	20.6	-6.3	2	67.0	19.4
3	33.5	0.8	3	71.5	21.9
4	52.6	11.4	4	80.7	27.1
5	66.8	19.3	5	89.6	32.0
6	75.9	24.4	6	97.9	36.6
7	82.6	28.1	7	98.3	36.8
8	81.6	27.6	8	95.3	35.2
9	69.6	20.9	9	93.1	33.9
10	58.4	14.7	10	83.8	28.8
11	37.2	2.9	11	72.2	22.3
12	21.9	-5.6	12	64.8	18.2
M	51.3	10.7	M	81.5	27.5
SD	24.8	13.8	SD	13.3	7.4

demonstrate this, we use our Tucson data. Our metric transformation coefficients with the Tucson data are

$$\zeta = \frac{s_{\delta^*}}{s_{\delta}} = \frac{13.3}{7.4} = 1.80$$

and

$$\kappa = \bar{\delta}^* - \zeta(\bar{\delta}) = 81.5 - 1.80(27.5) = 32$$

Because the linking equation, $\xi^* = 1.80(\xi) + 32$, transcends our data sets the alignment of the metrics is successful. Thus, the transformation results in all values on a common metric. These principles for handling different temperature scales also apply for aligning IRT metrics.

SHOULD DIF ANALYSES BE BASED ON LATENT CLASSES?

In Chapter 12 we discuss traditional DIF analyses. These analyses create two groups with known manifest characteristics (e.g., female and male subsamples). Therefore, there is a *de facto* (perhaps innocuous) assumption that individuals within a manifest

group are homogeneous. It can be argued that this assumption may not always be tenable, is not necessary to make, and that violation of the assumption may lead to false DIF conclusions. For instance, there may be a subgroup of the Focal group that is disadvantaged by one or more items, but the rest of the Focal group is not disadvantaged. In this case, the subgroup is not at all like the majority of the Focal group (i.e., the Focal group is not homogeneous). However, the relative sizes of the Focal subgroup and the majority Focal group may result in the masking of DIF for one or more items. Consequently, the items do not appear to be exhibiting DIF when, in fact, they do for the subgroup.

Consider, for example, the use of race or ethnic background for manifest grouping. This strategy treats all members of the group as equivalent and ignores intramanifest group differences. An Asian American manifest group lumps, for example, Filipino, Korean, Indonesian, Taiwanese, and Asian Indians (to name but a few) together. Similarly, a Hispanic/Latinx focal group would include Cubans, Guatemalans, Mexican Americans, Peruvians, Columbians, Argentines, Puerto Ricans, and so on. These culturally distinct groups are also potentially confounded with recency of immigration. The same could be said of a Caucasian manifest group, as well as an African American manifest group. An African American manifest group would include recent immigrants from Haiti, Nigeria, Trinidad, and Ghana, as well as African Americans whose families have lived in the United States for hundreds of years. Similarly, the homogeneity of males and of females may also be questioned.

de Ayala, Kim, Stapleton, and Dayton (2003) proposed that DIF analyses should focus on latent classes (LCs), not manifest groups. By focusing on LCs one avoids the assumption that manifest groups are homogeneous. Thus, our data may reflect a mixture of multiple latent populations or classes. Within each of these latent classes there are quantitative individual differences, but the classes are qualitatively different. Within a class there is a latent continuum, and this continuum is wholly or in part different from those in other classes. Therefore, our modeling of the data involves both latent classes and latent continua; see "Mixture Models" in this Appendix F. There is a multidimensional aspect to this DIF conceptualization, albeit different from that seen in the multidimensional item response theory interpretation of DIF (e.g., see Ackerman, 1996; Camilli, 1992; Reckase, 1997b). (Frederickx, Tuerlinckx, De Boeck, and Magis [2010] present an alternative in which one has a DIF class and a non-DIF class.)

In the simplest multiclass situation the sample consists of a mixture of two latent classes. If the latent classes are functionally equivalent to the manifest groups (i.e., 100% of the Reference group members belong to one latent class and 100% of the Focal group members are in another latent class), then the manifest groups are homogeneous and the current approach to DIF analysis is appropriate. (Obviously, this would also be true if the data consisted of a one-class structure.) However, if the latent classes are not isomorphic with the manifest groups, then the latent classes contain a mixture of members from the different manifest groups. For example, one latent class may consist of 80% Reference manifest group members, whereas the other latent class may contain 80% Focal group members. According to this conceptualization, DIF analyses may be improved by determining the latent class structure first and then using this information for conducting the DIF analysis.

THE SEPARATION AND RELIABILITY INDICES

In Chapter 3 we state that the person SEPARATION index gives an indication of how well the instrument can separate or distinguish persons in terms of their latent variable locations. In this section we provide some of the technical aspects of this index and the RELIABILITY index discussed in Chapters 3 and 7. Both of these indices may be calculated for people and items. We first treat these indices for respondents and then for items.

The person SEPARATION index is the ratio of the ADJ.SD to RMSE (Wright & Masters, 1982) with a lower bound of 0, no upper bound, and is expressed in standard error units. According to Wright and Masters (1982), the person ADJ.SD provides an estimate of the “true” person standard deviation from which measurement error-caused bias has been removed. (Measurement error is that part of the total variability unaccounted for by the model.) The ADJ.SD is

$$ADJ.SD(\theta) = \sqrt{SD_{\hat{\theta}}^2 - RMSE(\hat{\theta})^2} . \quad (G.22)$$

For example, using the results from the top-half of Table 3.3, we have an observed SD for people of $SD_{\hat{\theta}} = S.D. = 1.39$. Our “average measurement error” for nonextreme examinees is $REAL\ RMSE = 1.34$ gives us an $ADJ.SD = \sqrt{1.39^2 - 1.34^2} = 0.37$. Therefore, the SEPARATION ratio is $SEP = ADJ.SD(\hat{\theta}) / RMSE(\hat{\theta}) = ADJ.SD / RMSE = 0.37 / 1.34 = 0.28$; these calculated values match (within rounding) those from the table. (These equations also apply to the MODEL RMSE line.) The SEPARATION ratio is related to the number of “statistically different performance strata that the test can identify in the sample” (Wright, 1996). The number of distinct strata is $(4 * SEP + 1) / 3$ (Fischer, 1992). For instance, a SEP of 3 implies $INT((4 * 3 + 1) / 3) = 4$ strata where INT indicates to drop the decimal portion of the number. Because “large” SEPARATION ratio values represent a “large” number of strata they are considered better than small ones. In our case, our SEPARATION ratio is a poor value; this is primarily due to our test’s short length.

Because the SEPARATION index does not have a finite upper bound, it is sometimes beneficial to transform it. The SEPARATION ratio is directly related to the bounded SEPARATION RELIABILITY index (*REL*) such that as the number of strata increase so does coefficient alpha (Fisher, 1992). For example, a SEP = 3 is associated with a REL of 0.90. Specifically, Linacre and Wright (2001) state that the SEP and REL indices are related as

$$REL = \frac{SEP^2}{1 + SEP^2} \quad (G.23)$$

and

$$SEP = \sqrt{\frac{REL}{1 - REL}} . \quad (G.24)$$

The person SEPARATION RELIABILITY is another way to estimate reliability (see Linacre, 1996, 1997). REL tells us about the consistency or reproducibility of the $\hat{\theta}$ s. One way of looking at this index is that it indicates the consistency (reproducibility) of the $\hat{\theta}$ s across (mirror) instruments designed to measure the same latent variable. Its range

is from 0 to 1, with values close to or at 1 considered better than values approaching or at 0. According to Fischer (1992) a $REL < 0.5$ indicates that the differences between $\hat{\theta}$ s are mainly due to measurement error. As was the case with the SEPARATION index, the RELIABILITY index (Wright & Masters, 1982) is based on the $ADJSD(\hat{\theta})$

$$REL = \frac{ADJ.SD(\hat{\theta})^2}{SD_{\hat{\theta}}^2}. \quad (G.25)$$

As an example, using the analysis from Table 3.3 we have that the RELIABILITY for the REAL RMSE (nonextreme) line in the table is $RELIABILITY = 0.39^2/1.39^2 = 0.08$. This value indicates that the mathematics instrument is not doing a good job of distinguishing people. As a result, there is little reason to believe that we would obtain the same ordering of people with a different set of items measuring mathematics proficiency (i.e., the proportion of observed sample variance that is not due to measurement error is quite low).

As is the case with INFIT and OUTFIT, we can calculate SEPARATION and RELIABILITY for items. The item SEPARATION index gives an indication of how well the instrument can separate or distinguish items in terms of their latent variable locations. The premise of this index is that one would like items to be sufficiently well separated (i.e., in terms of their locations) to identify the direction and the meaning of the latent variable (Wright & Masters, 1982). As such, we would like to see little estimation error. This last aspect is assessed by the (item) RELIABILITY index. These two indices are calculated in a manner somewhat parallel to that used with persons. Specifically, the

$$\text{item } ADJ.SD(\hat{\delta}) = \sqrt{SD_{\hat{\delta}}^2 - V(RMSE(\hat{\delta})^2)}, \quad (G.26)$$

with

$$\text{item } SEP = \frac{ADJ.SD(\hat{\delta})}{RMSE(\hat{\delta})} \quad (G.27)$$

and

$$\text{item } REL = \frac{ADJ.SD(\hat{\delta})^2}{SD_{\hat{\delta}}^2}. \quad (G.28)$$

V is an “overall test-to-sample fit mean square” (Wright & Masters, 1982, p. 92). An item RELIABILITY of 1.00 indicates the instrument is creating a well-defined variable. This is the case for our test with an item $REL = 1.00$ (see the bottom-half of Table 3.3).

DEPENDENCY IN TRADITIONAL ITEM STATISTICS AND OBSERVED SCORES

In Chapter 3 we discuss the concept of invariance. This property is desirable and useful because it frees the practitioner from the specific characteristics of the instrument and samples used. Below we demonstrate that this property is not present in the application of CTT, but is exhibited in IRT.

Assume that we administer a 10-item instrument to each of two groups. Although our instrument measures mathematical reasoning (MR), the following demonstration also applies to other types of instruments, such as a delinquency scale or a survey of attitudes on global warming. The examinees' responses are scored "correct" and "incorrect." The first group consists of 1000 examinees and is high on the MR continuum, with an average number correct of 6.741 items ($SD = 1.844$), whereas the second group of 1000 examinees is low on this continuum ($M = 1.551$, $SD = 1.195$). We refer to the former group as the high group and the latter group as the low group.

In a traditional item analysis we calculate various indices, such as a reliability estimate, item discrimination indices, and the item difficulty index, to name just a few. One item discrimination index is the correlation between the responses on an item and the observed scores (e.g., the point biserial), and the traditional measure of an item's difficulty is the proportion of correct responses on an item (i.e., the item's P -value, P_j , or its mean). Using the high-group data, the instrument's coefficient alpha is 0.556, and for the low group the coefficient alpha is 0.404. Table G.6 contains the traditional item statistics for the two groups. As we see, the items have different characteristics across the two groups. For example, in general the items discriminate (i.e., the r_c s) better in the MR high group than in the low group. Furthermore, the item difficulties (i.e., the P_j s) indicate that the items are easier in the high group than in the low group; low values of P_j indicate a difficult item and high P_j values reflect an easy item. Therefore, our interpretations of these item indices would be conditional on the sample. For instance, item

TABLE G.6. Traditional and IRT Item Statistics

Item	Traditional				IRT	
	High MR		Low MR		High MR	Low MR
	P_j	r_c	P_j	r_c	$\hat{\delta}$	$\hat{\delta}$
1	0.751	0.256	0.082	0.144	-1.369	2.840
2	0.973	0.128	0.491	0.268	-4.201	0.044
3	0.365	0.278	0.011	0.090	0.691	5.052
4	0.693	0.248	0.060	0.127	-1.015	3.208
5	0.724	0.271	0.069	0.186	-1.199	3.046
6	0.494	0.306	0.022	0.125	0.030	4.319
7	0.978	0.078	0.596	0.256	-4.428	-0.476
8	0.297	0.308	0.009	0.046	1.072	5.260
9	0.856	0.240	0.169	0.197	-2.179	1.910
10	0.610	0.268	0.042	0.089	-0.560	3.612

Note. r_c is the corrected item-total correlation between an item's responses and the observed scores; P_j is the proportion of correct responses to item j .

1 is an “easy” item *if* it is administered to the high group, but it is a “hard” item when administered to the low group. (This is analogous to the situation in which the unit for measuring a box is the length of a string based on the shortest dimension of the box; see Chapter 1.) Moreover, this item is a poor discriminator in the low group, but a more reasonable (although not good) discriminator in the high group.

Given that the item statistics vary as a function of the sample used in their calculation, we might ask, “What is the relationship between the item statistics across the two samples?” The correlation between the item discriminations for the high and low groups is -0.852 and between the item difficulties it is 0.796 . Figure G.4 shows that the *linear* relationship between the item difficulties across groups is not as strong as the correlation of 0.796 might suggest. That is, the two groups’ item difficulties are nonlinearly related and are influenced to a large extent by the characteristics of the sample on which they were calculated.

We now turn to applying our IPL model to these data. After separately calibrating the instrument for the high and low groups, we obtain two sets of $\hat{\delta}s$; see the two right-most columns in Table G.6. As would be expected from our indeterminacy discussion (Chapter 3), the two sets of $\hat{\delta}s$ are not equal. For instance, item 1 is estimated to be located at -1.369 with the high group, but the item is estimated to be located at 2.840 for the low group. However, a closer examination reveals that the relative positions among the item location estimates is essentially the same across the two. The correlation between these estimates for the high and low groups is 0.999 ; the scatterplot is presented in Figure G.5. Unlike the P-value scatterplot, Figure G.5 shows that the near perfect correlation of 0.999 accurately reflects the linear relationship between the two sets of estimates. This very strong linear relationship shows that our IRT item characterizations transcend the sample characteristics, whereas the traditional indices do not.

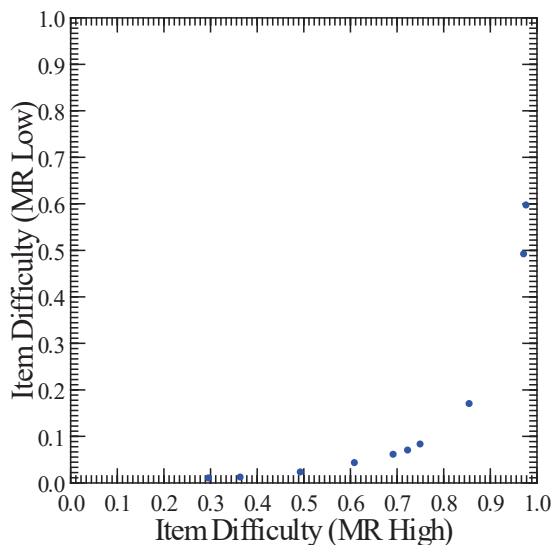


FIGURE G.4. Scatterplot of traditional item difficulties.

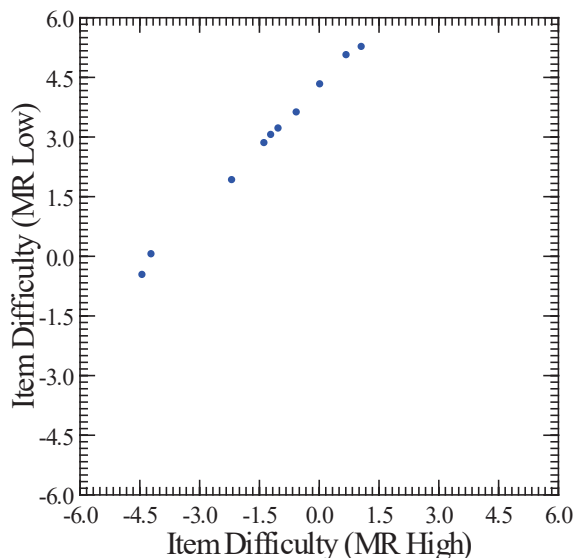


FIGURE G.5. Scatterplot of IRT item locations.

Moreover, although our interpretation of an item’s traditional difficulty as “easy” or “hard” is conditional on the proficiency of the examinee group, with IRT we do not need this qualification after we align the metrics (Chapter 11).⁵

As is the case with most estimation problems, sample size is an important determinant of the quality of estimation. Therefore, we halve the sample size to show that the invariance of our estimates continues to exist. With half as many examinees in each sample, the correlation between the traditional item discriminations for the high and low groups decreases to -0.284 , whereas the correlation between the traditional item difficulties becomes 0.782 ; the nonlinearity seen in Figure G.4 continues to exist. In contrast, the correlation between the IRT item location estimates remains strong ($r = 0.978$). As such, the effect of examinee characteristics (e.g., high vs. low proficiency) continue to affect the traditional item indices, but still do not affect our IRT item parameter estimates.

So far we have been concerned with how sample characteristics affect our item statistics. We now turn to the complementary question: “How do the characteristics of our instrument (e.g., the difficulty of an examination) affect the person location estimates?” To examine this question, assume that an item pool of 40 vocabulary items is divided into two tests. The first test consists of the 20 easiest items with a mean difficulty ($\bar{\delta}$) of -1.0141 , a $SD = 1.0582$ and minimum and maximum difficulties of -2.4390 and 0.5470 , respectively. The second test contains the 20 hardest items ($\bar{\delta} = 1.4529$, $SD = 0.7578$, minimum = -0.5690 , maximum = 2.9510). In the following, the first test is referred to as the Easy test, whereas the second test is known as the Hard test.

Through a Monte Carlo study we simulate the administration of these two tests to 1000 examinees randomly sampled from a normal distribution and whose locations (θ s) are known.⁶ Examining the responses, we see that the first examinee correctly answered 12 and 6 items on the Easy and Hard examinations, respectively. It is self-

evident that a person's observed score is affected by the easiness or difficulty of the examination. In fact, comparing the observed scores for the 1000 examinees across the Easy/Hard tests using a paired t -test shows the two sets of observed scores are significantly different from one another ($t = 101.215$, $p = 0.000$); the correlation between the observed scores from the Hard examination and those from the Easy examination is 0.713. Therefore, an examinee's observed score is dependent on the instrument's characteristics. Of course, whether a given score is an accurate assessment of an examinee's location on the construct is a validity question.

To estimate the locations of the 1000 examinees we use the Rasch model with EAP (Chapter 4). The estimate of the first examinee's location ($\hat{\theta}_1$) is -0.3955 according to the Easy test and 0.3790 according to the Hard test. Similar to what is seen with the observed scores, there are two different person location estimates for each person. However, if there is model-data fit these estimates should be strongly linearly related and we can linearly transform the $\hat{\theta}$ s from the Easy test metric to the Hard test metric or vice versa. (How this is done is demonstrated in Chapter 4 in the discussion of metric transformation.) Whether the $\hat{\theta}$ represents an examinee's location on the construct of interest is still a validity question.

The Pearson correlation between the Hard test $\hat{\theta}$ s and those from the Easy test is 0.743. This correlation is similar to what we see with the observed scores. As such, we have not shown that our IRT person location estimates are not influenced by the instrument's characteristics. However, there are two primary reasons for the magnitude of this correlation. The first is that the tests provide information over a limited range of the θ continuum, and the second is the *asymptotic* nature of estimation. We discuss each of these reasons in turn.

To understand the first reason, compare the test's difficulty range with the range of person locations. Because the examinees are normally distributed we expect that approximately 99% of the examinees to be located from -3 to 3 . However, the Easy test does not have items above 0.547 and the Hard examination does not have items located below -0.569 . Therefore, for both examinations we need to estimate person locations that are beyond the range represented by the examination (e.g., estimating a person located above the Easy test's most difficult item, $\delta = 0.547$). At this point it appears that this argument may also be used to explain the CTT results. Therefore, it still remains to be shown that despite the Easy and Hard examinations' item location distributions we can obtain $\hat{\theta}$ s that are highly linearly related—in effect, obtaining person location estimates that are not influenced by the instrument's characteristics.

The second reason, the asymptotic nature of estimation, addresses the issue of estimating person locations that are not influenced by the instrument's characteristics. In our current situation we have only 20 observations (i.e., items) for estimating the examinees' locations. In contrast, for estimating the item locations there are 1000 observations (i.e., examinees) available. It is in this discrepancy in the number of observations for estimation that we find the explanation for the magnitude of the correlation.

The $\hat{\theta}$ s are *asymptotically* unbiased which, in effect, means that one needs a large number of items to compensate for the tests' truncated item location distributions. (If the tests contained items that spanned the full range of interest, then the issue of test

length would not be as much consequence as it is in this example.) To demonstrate this issue we increase the number of items on each examination while still restricting the range of the items to be the same as the 20-item tests. Specifically, we increase the Hard and Easy test lengths to 100 items while restricting their respective difficulty ranges to match those from the corresponding 20-item tests. This means that for the 100-item Easy test there are no items more difficult than 0.547 and for the 100-item Hard examination there are no items easier than -0.569 . Using these 100-item tests, the persons' locations are re-estimated. The correlation between the re-estimated $\hat{\theta}$ s from the Easy test and those from the Hard test increases to 0.933. This is a substantial improvement over our 20-item Easy and Hard tests' results and shows that person estimation is not adversely affected by the range of item locations of the instrument. If we continue to increase the length of each test to 250, 500, and 1000 items, then the respective correlations between the Easy and Hard tests' $\hat{\theta}$ s become 0.969, 0.982, and 0.987. Therefore, a test's level of difficulty does not adversely affect the person location estimation and our estimates of person location are "free" of the instrument's characteristics. Obviously, increasing the test length would not eliminate the test dependency issue seen with CTT.

As we would expect, as the number of items increases, the corresponding standard errors, $s_e(\hat{\theta})$ s, decrease. For example, the mean $s_e(\hat{\theta})$ s for the 20-item tests are 0.502 for the Easy test and 0.514 for the Hard test. However, if we lengthen the tests to 100 items, then the mean $s_e(\hat{\theta})$ decreases to 0.248 for the Easy test and to 0.262 for the Hard test. Further increasing the test length to 250 and 500 items results in the mean $s_e(\hat{\theta})$ s falling to 0.162 (Easy)/0.165 (Hard) and 0.109 (Easy)/0.110 (Hard), respectively. With 1000 items the mean $s_e(\hat{\theta})$ decreases to 0.063 for the Easy test and to 0.061 for the Hard test.

The preceding two observations concerning item and person location estimation may be summarized as *specific objectivity*. Loosely speaking, specific objectivity means that what one is interested in measuring does not affect the measuring instrument and the measuring instrument does not affect what is being measured.⁷ When this level of objectiveness is realized then it is "possible to generalize measurement beyond the particular instrument used, to compare objects measured on similar but not identical instruments, and to combine or partition instruments to suit new measurement requirements" (Wright, 1968, p. 87). Wright (1968) has referred to the capability of obtaining item parameter estimates that are not influenced by the sample of individuals as *person-free test calibration*; this is also known as *item-parameter invariance* (Lord, 1980) and *object-free instrument calibration* (Wright, 1968). Moreover, Wright refers to the capacity to estimate a person's location "free" of the instrument's characteristics as *item-free person measurement*; this is also known as *person-parameter invariance* (Lord, 1980) and *instrument-free object measurement* (Wright, 1968). ("Person-free" and "item-free" should not be taken literally.) Therefore, IRT's invariance property is the realization of Thurstone's (1928) idea that "the scale must transcend the group measured" (p. 547). For an instrument to be accepted as valid, then it must not be seriously affected in its measuring function by the object of measurement, and "to the extent that its measuring function is affected, the validity of the instrument is impaired or limited" (p. 547).

CONDITIONAL INDEPENDENCE USING Q_3

To use Q_3 for evaluating the conditional independence assumption we first calculate Q_3 , determine a screening value, and then perform our comparison. With a five-item instrument there are 10 Q_3 values (i.e., $L(L - 1)/2$) to calculate. Although a statistical package can be used to calculate Q_3 , we use a spreadsheet program. To calculate Q_3 we need our person location estimates to calculate the expected responses, $p_j s$. For instance, in our 3PL model calibration (see Table 6.1) we requested that the EAP $\hat{\theta} s$ be saved to a file (i.e., "MATH3PL_EAP.SCO"). In addition to these $\hat{\theta} s$, we import our response data and the item parameter estimates. Using the $\hat{\theta} s$ and the item parameter estimates we calculate the $p_j s$ for each respondent. Comparing our observed responses to the corresponding $p_j s$ we determine the residuals ($x_{ij} - p_j(\hat{\theta}_i)$) for each item and each person. The Pearson product-moment correlation function is used to calculate the correlations (i.e., the $Q_3 s$) among the unique residual pairings. Table G.7 contains these $Q_3 s$ for the mathematics data example; the scatterplots (not presented) corresponding to these values were inspected for anomalies, but none were found. Not surprisingly, given the instrument's length, only one of the $Q_3 s$ is positive; the average Q_3 is -0.1799 .

As noted above, Yen's (1993) suggestion of a 0.2 screening criterion was in the context of instruments that had at least 17 items (i.e., an expected Q_3 value of $-1/(L - 1) = -0.0625$). However, the expected Q_3 value for our 5-item mathematics test is -0.25 . This value is substantially farther away from 0 than the expected Q_3 value in Yen's (1993) study. Therefore, although a ± 0.2 screening criterion may be useful when the expected Q_3 value is comparatively close to 0 (e.g., with 35 or more items), with only 5 items this criterion is less useful. As a result, rather than use the 0.2 screening criterion we determine the screening value for our 5-item instrument. To identify our screening criterion we conduct a simulation that showed the magnitude of $Q_3 s$ that

TABLE G.7. Q_{3s} for the Math Data Set; Q_{3s} Are in Parentheses

	Items				
	1	2	3	4	5
1	1.0000				
2	-0.1810 (0.0328)	1.0000			
3	-0.0985 (0.0097)	-0.3037 (0.0922)	1.0000		
4	-0.0915 (0.0084)	-0.2134 (0.0455)	-0.2087 (0.0436)	1.0000	
5	-0.0581 (0.0034)	-0.1366 (0.0187)	-0.0988 (0.0098)	-0.1566 (0.0245)	1.0000

might be expected to be observed if the data conformed to the model's conditional independence assumption.

This simulation can be performed with a statistical package (e.g., SAS) or a programming language (e.g., FORTRAN, R). In our case, the simulation consists of randomly sampling $N = 19,601$ standard unit normal z s from a normal distribution. These z s are the persons' θ s; these "persons" are referred to as simulees. To generate a simulee's item response we use the item's parameter estimates from Table 6.3 and the simulee's θ to calculate the probability of a response of 1 according to our 3PL model. This probability (p_j) is compared with a uniform random number. If the uniform random number is less than or equal to the calculated probability, then the simulee's response to the item is coded as 1, otherwise it is coded as 0. This process is repeated for each item and for each simulee. In effect, we have created a parallel data set with the same number of respondents and items as our empirical data except that these data are, by definition, conditionally independent.

Because in practice we have only an estimate of θ we obtain each simulee's EAP location estimate using the item parameter estimates and the simulee's response vector. These EAP $\hat{\theta}$ s along with our item parameter estimates are used to determine the expected response for each item for each simulee. To obtain our residuals we calculate the expected response (i.e., p_j) and compare it to the corresponding simulated response for each item and each simulee. The intercorrelations among these item residuals are calculated and recorded. The entire process, from sampling the N standard unit normal z s to calculating the Q_{3s} , is repeated a few thousand times (e.g., 5000). With five items there are 10 unique Q_{3s} (i.e., $L(L-1)/2$) or $10 \times 5000 = 50,000$ Q_{3s} across the 5000 replications. The Q_3 value corresponding to the bottom 5% (i.e., 5% total) of the Q_3 (null) distribution is -0.293485 with minimum and the maximum Q_{3s} are -0.3194 and -0.0171 , respectively.

Having obtained our screening value (-0.2935) we return to our empirical data. Our Q_{3s} show that we have one item pair (items 2 and 3) that has an absolute value exceeding the screening value of 0.2935. That is, after fitting the unidimensional 3PL model to the data the items in this item pair has almost 10% of their residual variability in common. (This item pair may or may not be found to exceed the screening criterion with either the 1PL or 2PL models.) As we did in Chapter 6 we also use a "gap" approach informed by \bar{Q}_3 to identify values that reflect item dependence ($\bar{Q}_3 = -0.1541$). Figure G.6 contains our dot density plot for Q_3 . As can be seen, we have a cluster of item pairs in the range of approximately -0.22 to -0.05 that are clustered about \bar{Q}_3 . Our item pair 2-3 shows a sizable gap to the item pair cluster. This is also the item pair flagged by Q_3^p . The italicized Q_{3s} (Table G.7) are the additional item pairs flagged by Q_3^p . Comparing the Q_3^p results with those of Q_3 shows that it appears that not controlling for variability shared across residuals can potentially mask item pairs that should be furthered examined for conditional dependence. Although the item pair 2-3 may be considered to be exhibiting item dependence, evidence of conditional dependence in the remaining nine pairs is absent. How one deals with items that are considered sufficiently dependent to be problematic post-administration is discussed in Chapter 6.

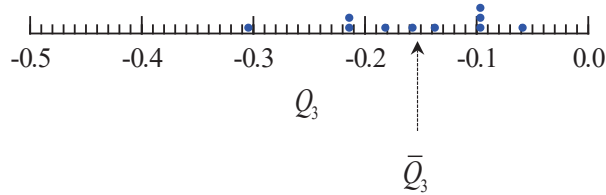


FIGURE G.6. Dot density plot for Q_3 .

STANDALONE NOHARM CALIBRATION OF INTERPERSONAL ENGAGEMENT INSTRUMENT, M2PL MODEL

To perform our analysis we prepare an ASCII input file (Table G.8) that is subsequently submitted to NOHARM. The first command line in the input file contains a title for the analysis with the remaining lines specifying the analysis setup. The second line “10 2 1000 0 1 0 0 0” specifies that there are 10 items and a two-dimensional analysis based on 1000 cases, that the input data consist of binary responses, to perform an exploratory analysis, and that NOHARM should generate its starting values and print the correlation, covariance, and residual matrices, respectively. The subsequent line allows the user to provide the IRF’s lower asymptote value for each of the 10 items. Because we are fitting a two-parameter model this line contains one zero for each item. However, if we were using the M3PL model we would provide an estimate of the lower asymptotes on this line. These estimates may be obtained by calibrating the data with the 3PL model and using the corresponding χ_j estimates as input for the M3PL model calibration. This approach has been found to work well with two-dimensional data, but not as well with four-dimensional data (DeMars, 2007).

Following this line are the binary responses for the 1000 individuals. (Note that when the data consist of binary response vectors, the number of cases specified on line 2 is used for reading the data. If the number of cases specified on line 2 does not match the number of response lines, then an error will occur [i.e., “Unexpected end-of-file when...”].)

TABLE G.8. Two Dimensional Input Command File (name = intrprsnl.inp)

```

Interpersonal Ex, 2 dimensions, raw data input, exploratory
10 2 1000 0 1 0 0 0
0 0 0 0 0 0 0 0 0 0
1 1 1 0 1 0 0 0 0 0
1 1 0 1 1 0 0 0 0 0
1 1 1 0 0 0 0 0 0 1
1 1 0 1 0 0 1 0 0 0
1 1 0 1 0 0 0 1 0 0
:
```

Table G.9 contains the corresponding output. The beginning of the output contains echoes of the input specifications, initial starting values, the covariance matrix, and so on, followed by the `Results` section. We first discuss model-data fit before returning to the item parameter estimates.

As mentioned in Chapter 3, NOHARM produces a residual matrix to facilitate assessing model-data fit. The residual matrix is the discrepancy between the observed covariances and those after the model is fitted to the data. Therefore, the ideal situation is where these discrepancies are zero. Our residuals are comparatively small vis-à-vis the observed item covariances, with almost all residual magnitudes in the thousandths' place or less. To summarize the residual matrix, NOHARM provides the root mean square (RMS). As we discuss in Chapter 3, the RMS is the square root of the average squared difference between the observed and predicted covariances. Therefore, small values of RMS reflect good fit. This overall measure of model-data misfit is evaluated by comparing it to four times the reciprocal of the square root of the sample size (i.e., the "typical" standard error of the residuals). For these data this criterion is 0.1265. The observed RMS of 0.0033523 indicates that we have evidence of model-data fit. For this example the residual matrix, the RMS, and the `GFI` all indicate good model-data fit. Therefore, we proceed to examine the item parameter estimates.

The output following the residual matrix contains the common factor parameterization of the MIRT model. Whether this information is used depends on one's purpose (e.g., see McDonald, 1997; Reckase, 1997a). The sections labeled `Factor Loadings`, `Varimax Rotated Factor Loadings`, `Unique Variances`, and `Promax (oblique) Rotated Factor Loadngs [sic]` contain the values based on the common factor model. Intermixed within these sections are two additional sections titled `Varimax Rotated Coefficients of Theta` and `Promax Rotated Coefficients of Theta` that contain the common factor model's reparameterization into the MIRT model.

The estimated MIRT model is a reparameterization of the common factor model (see Appendix C). Therefore, the common factor model's loadings may be related to the MIRT model item parameters through the item unique variances. The items' unique variances are given on the line labeled `Unique Variances` and are equal to an item's communality subtracted from 1 (i.e., unique variance = $1 - \rho_j \Sigma \rho_j$). For example, for item 1 we have that its loadings on factors 1 and 2 are 0.809 and 0.0, respectively. Therefore, its estimated communality is $\hat{h}^2 = 0.8092 + 0.02 = 0.654$ and its unique item variance is $1 - 0.654 = 0.346$. This is item 1's value in the `Unique Variances` section.

Dividing each item's estimated `Threshold Values` (i.e., from Appendix C; $\hat{\tau}_j$) by the square root of its corresponding unique variance yields the item's intercepts (i.e., Equation C.29). As an example, for item 1 we have that its estimated threshold $\hat{\tau}_1 = 0.536$ and, as a result, its

$$\hat{\gamma}_1 = 0.536 / \sqrt{0.346} = 0.911$$

(i.e., item 1's value in the `Final Constants` section). In terms of the `Factor Loadings` matrix, we can divide each item's factor loading by the square root of its unique

TABLE G.9. Output File for the Self-Efficacy to Engage in Making Interpersonal Engagement Behavior Instrument^a

```

                                N O H A R M
                                Fitting a (multidimensional) Normal Ogive
                                by Harmonic Analysis - Robust Method

Input File : intrprsnl.inp

Title :      Interpersonal Ex, 2 dimensions, raw data input, exploratory

Number of items      = 10
Number of dimensions = 2
Number of subjects   = 1000

An exploratory solution has been requested.

Sample Product-Moment Matrix
      1      2      3      4      5      6      7      8      9
1      0.704
2      0.460  0.557
3      0.505  0.403  0.614
4      0.480  0.380  0.426  0.608
5      0.415  0.323  0.356  0.346  0.511
:

Item Covariance Matrix
      1      2      3      4      5      6      7      8      9
1      0.208
2      0.068  0.247
3      0.073  0.061  0.237
4      0.052  0.041  0.053  0.238
5      0.055  0.038  0.042  0.035  0.250
:

                                =====
                                Results
                                =====

Success. The job converged to the specified criterion.

Final Constants
      1      2      3      4      5      6      7      8      9      10
0.911  0.177  0.382  0.324  0.031 -0.828 -0.520 -0.627 -0.992 -1.018 ←  $\hat{\gamma}_i$ s

Final Coefficients of Theta
      1      2
1      1.374  0.0
2      0.724 -0.024
3      0.858 -0.020
4      0.581  0.254
5      0.494  0.079
6      0.739  0.562
7      0.650  0.610
8      0.538  0.505
9      0.591  0.587
10     0.606  0.288

```

(continued)

TABLE G.9. (continued)

Final Correlations of Theta

	1	2
1	1.000	
2	0.0	1.000

Residual Matrix (lower off-diagonals)

	1	2	3	4	5	6	7	8	9
2	-3.1e-4								
3	0.002	-0.002							
4	0.003	0.002	-0.005						
5	-0.005	0.003	0.002	0.001					
6	-0.002	-0.001	0.002	0.001	-3.1e-4				
7	-0.001	0.002	-0.001	0.008	-0.005	-0.004			
8	0.002	7.0e-5	0.003	-0.006	-1.6e-4	0.003	0.002		
9	2.9e-4	-0.003	-0.001	-0.002	0.005	1.8e-5	-0.004	0.003	
10	0.001	-0.002	-4.3e-4	-0.005	0.004	0.003	0.004	-0.010	0.004

Sum of squares of residuals (lower off-diagonals)	=	0.0005057	
Root mean square of residuals (lower off-diagonals)	=	0.0033523	← The RMSR
Tanaka index of goodness of fit	=	0.9984227	← The GFI

Threshold Values

1	2	3	4	5	6	7	8	9	10
0.536	0.143	0.290	0.274	0.028	-0.607	-0.388	-0.504	-0.762	-0.845

Unique Variances

1	2	3	4	5	6	7	8	9	10
0.346	0.656	0.576	0.714	0.800	0.537	0.557	0.647	0.590	0.689

Factor Loadings

	1	2
1	0.809	0.0
2	0.586	-0.019
3	0.651	-0.015
4	0.490	0.214
5	0.442	0.071
6	0.542	0.412
7	0.485	0.455
8	0.433	0.407
9	0.454	0.451
10	0.503	0.239

Varimax Rotated Factor Loadings

	1	2
1	0.739	0.327
2	0.544	0.220
3	0.602	0.249
4	0.362	0.394
5	0.375	0.243
6	0.328	0.596
7	0.259	0.613
8	0.231	0.547
9	0.233	0.596
10	0.364	0.422

(continued)

TABLE G.9. (continued)

Varimax Rotated Coefficients of Theta $\leftarrow \hat{\alpha}_{i,f}s$

	1	2
1	1.257	0.557
2	0.672	0.271
3	0.793	0.329
4	0.428	0.467
5	0.420	0.272
6	0.448	0.813
7	0.348	0.821
8	0.288	0.680
9	0.303	0.776
10	0.438	0.509

Promax (oblique) Rotated Factor Loadngs

	1	2
1	0.810	-0.001
2	0.610	-0.031
3	0.670	-0.025
4	0.241	0.329
5	0.360	0.108
6	0.062	0.632
7	-0.045	0.699
8	-0.041	0.625
9	-0.071	0.693
10	0.225	0.367

Factor Correlations

	1	2
1	1.000	
2	0.760	1.000

Promax Rotated Coefficients of Theta

	1	2
1	1.376	-0.002
2	0.753	-0.038
3	0.883	-0.033
4	0.285	0.389
5	0.402	0.121
6	0.084	0.863
7	-0.061	0.937
8	-0.051	0.776
9	-0.093	0.901
10	0.271	0.442

^aThe text following the " \leftarrow " is provided to help the reader understand the corresponding input.

variance to obtain the item's estimated discrimination parameter on the corresponding dimension (i.e., Equation C.30). For instance, item 2's loading on dimension 1 is $\hat{\rho}_{2,1} = 0.586$ and its

$$\hat{\alpha}_{2,1} = 0.586 / \sqrt{0.656} = 0.724$$

For this item's relationship to the second dimension, we have $\hat{\rho}_{2,2} = -0.019$ and its

$$\hat{\alpha}_{2,2} = -0.019 / \sqrt{0.656} = -0.024$$

Both of these $\alpha_{j,f}$ estimates are found in the Final Coefficients of Theta matrix. Similarly, dividing the Varimax Rotated Factor Loadings by the corresponding square root of the item's unique variance yields the Varimax Rotated Coefficients of Theta. The entries in this latter matrix yield the same p_j as those from the Final Coefficients of Theta section after accounting for rotation. For example, if $\underline{\theta}' = (1.5, -1.0)$, then using the estimated parameters for items 1 and 2 one obtains $p_1 = 0.95129$ and $p_2 = 0.78364$. Rotating the Factor Loadings matrix by approximately 23.872° yields the values in the Varimax Rotated Factor Loadings matrix and, as a result, this rotation angle is reflected in the Varimax Rotated Coefficients of Theta. Therefore, after rotating the $\underline{\theta}$ by this amount, our rotated person locations are $\underline{\theta} = (2.6417, -2.2626)$. Using this $\underline{\theta}$ and the rotated item parameter estimates for items 1 and 2, we obtain $p_1^* = 0.95129$ and $p_2^* = 0.78364$. The Varimax Rotated Coefficients of Theta estimates may be more meaningful in some situations.

The items' intercept (constant) estimates, $\hat{\gamma}_j$ s, are found in the section titled Final Constants. This section shows that our estimates are $\hat{\gamma}_1 = 0.911$, $\hat{\gamma}_2 = 0.177$, \dots , $\hat{\gamma}_{10} = -1.018$. The items' discrimination parameter estimates, $\hat{\alpha}_{j,f}$ s, are (first) shown in the Final Coefficients of Theta section. As would be expected from the discussion of indeterminacy, one sees that the first item's value on the second factor is fixed at zero to address the solution's rotational indeterminacy. We also note that some values are negative. Barring a mistake (e.g., a miskeyed correct response) or a functional form anomaly we want our discrimination parameter estimates to be positive. Consequently, we rotate our solution (i.e., the values in the Final Coefficients of Theta section) without loss of information. The Varimax Rotated Coefficients of Theta section contains our rotated estimates. The item discrimination parameter estimates for the first dimension are $\hat{\alpha}_{1,1} = 1.257$, $\hat{\alpha}_{2,1} = 0.672$, \dots , $\hat{\alpha}_{10,1} = 0.438$ (i.e., column 1). For the second dimension the estimates of the discrimination parameters are $\hat{\alpha}_{1,2} = 0.557$, $\hat{\alpha}_{2,2} = 0.271$, \dots , $\hat{\alpha}_{10,2} = 0.509$. Comparing these estimates to those of `sirt.noharm`'s F1 and F2 (see Table 10.1) we find that dimension 1 estimates correlate over 0.999 with F2 rotated estimates and dimension 2's estimate also correlate over 0.999 with F1 rotated estimates (i.e., the dimensions here are reversed from the `sirt.noharm` results).

Because these estimates are on the normal metric, we need to multiply them by $D = 1.702$ to place them on the logistic metric of the M2PL model. The Final

Correlations of Theta section gives the correlation between the θ_{js} . Table G.10 presents the estimates and other summary information.

CFI, GFI, M_2 , RMSEA, TLI, AND SRMR

In Chapter 3 we saw that NOHARM produces several model-data fit indices. We now provide slightly greater detail on these indices than above. Tanaka's (1993) goodness-of-fit index (GFI; labeled *Tanaka Index*) involves the sample covariance matrix, \underline{C} , and the residual covariance matrix, \underline{C}_{res} (McDonald & Mok, 1995). Specifically, the goodness of fit index is

$$GFI = 1 - \frac{Tr(\underline{C}_{res}^2)}{Tr(\underline{C}^2)}, \quad (G.29)$$

where Tr is the matrix's trace (i.e., the sum of the main diagonal's elements). A GFI of 1 indicates perfect fit. McDonald (1999) states that a minimum GFI of 0.90 indicates an acceptable level of fit and a minimum value of 0.95 indicates "good" fit.

A second index, root mean square residual (RMSR), is the square root of the average squared difference between the observed and predicted covariances with small values indicating good fit. This overall measure of model-data misfit may be evaluated by comparing it to four times the reciprocal of the square root of the sample size (i.e., the "typical" standard error of the residuals; McDonald, 1997). For example, for our math data with 19,601 respondents we have $4/\sqrt{19601} = 0.0286$.

TABLE G.10. Summary Statistics for the Interpersonal Engagement Behavior Instrument

	normal metric						logistic metric	
	$\hat{\alpha}_{i1}$	$\hat{\alpha}_{i2}$	$\hat{\gamma}_i$	\hat{A}_i	$\hat{\Delta}_i$	ω_{i1}^o	$\hat{\alpha}_{i1}$	$\hat{\alpha}_{i2}$
1	1.257	0.557	0.911	1.374	-0.663	23.9	2.139	0.948
2	0.672	0.271	0.177	0.725	-0.244	22.0	1.144	0.461
3	0.793	0.329	0.382	0.859	-0.445	22.5	1.350	0.560
4	0.428	0.467	0.324	0.633	-0.511	47.5	0.728	0.795
5	0.420	0.272	0.031	0.500	-0.062	32.9	0.715	0.463
6	0.448	0.813	-0.828	0.928	0.892	61.1	0.762	1.384
7	0.348	0.821	-0.520	0.892	0.583	67.0	0.592	1.397
8	0.288	0.680	-0.627	0.738	0.849	67.0	0.490	1.157
9	0.303	0.776	-0.992	0.833	1.191	68.7	0.516	1.321
10	0.438	0.509	-1.018	0.672	1.516	49.3	0.745	0.866

Three more indices for assessing model-data fit are the Root Mean Square Error of Approximation (RMSEA; Steiger, 1990), the Standardized Root Mean Square Residual (SRMR; Hu & Bentler, 1998, 1999), and Gessaroli and De Champlain's (G&D; Gessaroli & De Champlain, 1996) chi square statistic (G&D is discussed above). Following Hu and Bentler (1998, 1999) and McDonald and Mok (1995) RMSEA and SRMR are defined as

$$\text{RMSEA} = \sqrt{\frac{(\chi^2 - df)/N}{df}}, \quad (\text{G.30})$$

$$\text{SRMR} = \sqrt{\frac{2 \sum_j \sum_k \left(\frac{s_{jk} - \hat{\sigma}_{jk}}{s_{kk} s_{jj}} \right)^2}{L(L+1)}}, \quad (\text{G.31})$$

where $\hat{\sigma}_{jk}$ and s_{jk} are the reproduced and observed covariances, respectively, between items j and k , s_{jj} and s_{kk} are the observed standard deviations, with χ^2 and df reflecting the model under consideration. According to MacCallum et al. (1996) and Browne and Cudeck (1993; cited in MacCallum et al., 1996) a RMSEA less than 0.05 indicates a "close fit," a value from 0.05 to 0.08 reflects "good/fair" fit, a value from 0.08 to 0.10 indicates "mediocre" fit, and values greater than 0.10 reflecting "poor" fit; Hu and Bentler (1999) suggest a RMSEA a cutoff value "close to" 0.06. With respect to SRMR, Hu and Bentler (1999) consider a SRMR a cutoff value "close to" or less than 0.08 to be good fit.

Two additional indices used by `mirt` (e.g., Chapter 4) are the Tucker-Lewis Index (TLI or non-normed fit index, NNFI; Tucker & Lewis, 1973) and the comparative fit index (CFI; Bentler, 1990). TLI is defined as

$$\text{TLI} = \frac{\left(\chi_0^2/df_0 - \chi_1^2/df_1 \right)}{\left(\chi_0^2/df_0 - 1 \right)}, \quad (\text{G.32})$$

where χ_0^2 and df_0 correspond to the baseline (null) model and χ_1^2 and df_1 are for the comparison model. Although TLI has a range of 0 to 1 it is possible to obtain values outside this range. In these cases, TLI is set to appropriate boundary value. Values "close to" 0.95 or greater indicate good fit (Hu & Bentler, 1999).

CFI is given by

$$\text{CFI} = 1 - \frac{\max[(\chi_1^2 - df_1), 0]}{\max[(\chi_0^2 - df_0), (\chi_1^2 - df_1), 0]}, \quad (\text{G.33})$$

where $(\chi_1^2 - df_1)$ and $(\chi_0^2 - df_0)$ are for the comparison and baseline (null) models, respectively. CFI has a range of 0 to 1 with values "close to" 0.95 or larger indicating good fit (Hu & Bentler, 1999)

The M_2 statistic belongs to the family of limited information goodness-of-fit statistics. In contrast to other statistics that use a full contingency table that may have many small or zero frequency cells, M_2 uses the one- and two-way marginal tables that are less likely to have small cell frequencies. (There are m^L cells in the full contingency table where m is the number of response categories (e.g., with binary data and 5 items $2^5 = 32$ cells).) Research has shown the M_2 statistic maintains appropriate Type I error rates under varying degrees of model misspecification. M_2 is asymptotically distributed as a χ^2 with, for binary data, $df = 2^L - (\text{number of item parameters}) - 1$.

AN INTRODUCTION TO KERNEL EQUATING

For situations in which a parametric method's assumptions are untenable *nonparametric* equating methods may be fruitful. In general, nonparametric approaches make fewer assumptions of the underlying mathematical form than do parametric approaches and estimate item response functions directly from observed scores (see Lei, Dunbar, & Kolen [2004]; von Davier, Holland, & Thayer [2004]). One of these nonparametric approaches, kernel equating (KE), is described by von Davier et al. (2004) as "a unified approach to test equating based on a flexible family of equipercenile-like equating functions that contains the linear equating function as a special case" (p. 45). In contrast to traditional equipercenile equating that uses linear interpolation to continuize the discrete test score distributions, KE uses a Gaussian kernel to perform this continuization (Duong & von Davier, 2008; Kolen & Brennan, 2004; Mao et al., 2006; von Davier et al., 2004). In this regard, fewer mathematical assumptions are made by using this kernel than with linear interpolation (von Davier et al., 2004).

There have been only a few studies that directly compare KE to traditional methods. These studies have been based on empirical test data and have provided evidence that KE is a viable alternative to traditional methods (e.g., Lei, Dunbar & Kolen, 2004; Mao, von Davier et al., 2006; von Davier et al., 2004). Specifically, KE appears to perform very similar to, and in some circumstances better than, traditional linear and equipercenile methods (Mao et al., 2006; von Davier, Holland, & Thayer, 2004; von Davier et al., 2006). Other applications of kernel smoothing have demonstrated equivalent performance to parametric continuous response IRT methods (Ferrando, 2004). However, research also suggests that KE is affected by factors such as sample size and test length (Lee, 2007). KE provides an opportunity to use one family of equating methodologies for multiple types of equating designs.

There are two mathematical methods that can be used for KE. One is chain equating (CE), whereas the other is post-stratification equating (PSE). In CE one first links Form X to the anchor test A , followed by linking A to Form Y . In contrast, with PSE the marginal distributions of both X and Y in the target population are estimated first and then the equating function is computed (von Davier et al., 2004).

The process of performing KE, for CS and PSE approaches, involves five steps: (1) pre-smoothing (optional), (2) estimation of the score probabilities, (3) continuiza-

tion, (4) equating, and (5) calculating the standard error of equating. The pre-smoothing step involves estimating the score probabilities for a particular equating design by fitting a statistical model (e.g., a log linear model) to each of the form–anchor test score distributions (i.e., (X, A) in subpopulation P and (Y, A) in subpopulation Q). The resulting score distribution is used for the remaining equating process as well as providing the matrix that can be used for calculating the standard error of equating (Chen, 2012; von Davier et al., 2006).

In the second step the score probability distributions based on the target population T are estimated using the smoothed score distributions obtained from the pre-smoothing in step 1 (von Davier et al., 2006). The third step is the continuization step and involves the process of transforming the discrete score distributions for X and Y on population T into continuous score distributions over the entire score range by using kernel smoothing (von Davier et al., 2006). It is this step that involves the Gaussian kernel smoothing for the distribution of X and Y as well as the continuization constants or bandwidths. In effect, this step consists of selecting these continuization constants.

The bandwidth can be thought of as the width of the segmentation of a distribution within each segment the data are transformed. The specific transformation is a function of which kernel density estimator is used (e.g., a Gaussian kernel). Consequently, the bandwidth serves as a smoothing parameter with values close to zero yielding very little smoothing and increasing values resulting in correspondingly increased degrees of smoothing. Thus, the bandwidth affects the shape of the resulting continuous approximation (Holland & Thayer, 1989). See von Davier et al. (2004), Cid and von Davier (2015), Häggstrom and Wiberg (2014), and Lee (2007) for more information.

When a bandwidth is approximately 0.3 the equating functions “agree closely with the traditional equating functions” (Holland & Thayer, 1989, p. 11). Large bandwidth values (e.g., 5 or 10) yield equating functions that are progressively more linear in form and that begin to approximate the traditional linear equating functions (i.e., bandwidths that approach infinity) (Holland & Thayer, 1989). More generally speaking, setting the bandwidths to be greater than 10σ results in approximating linear equating and setting bandwidths to be less than 0.1σ produces equipercentile equating (von Davier et al., 2004). A form’s bandwidth value can be specified by the researcher or algorithmically determined to minimize the mean square error (MSE).

The equating of scores occurs in step 4 and involves computing the KE functions. The KE function for equating X to Y on T is given by

$$\hat{e}_Y(x) = e_Y(x; \bar{R}, \bar{S}) = G_{h_Y}^{-1}(F_{h_X}(x; \bar{R}); \bar{S}) = G_{h_Y}^{-1}(F_{h_X}(x_j)), \quad (\text{G.34})$$

where $\hat{e}_Y(x)$ is the KE function for equating X to Y , and \bar{R} and \bar{S} are the estimated vectors of score probabilities of r_j and s_k for populations P and Q , respectively. Similarly, the KE function for equating Y to X on T is given by

$$\hat{e}_X(y) = e_X(x; \bar{R}, \bar{S}) = F_{h_X}^{-1}(G_{h_Y}(y_k; \bar{R}); \bar{S}) = F_{h_X}^{-1}(G_{h_Y}(y_k)). \quad (\text{G.35})$$

These equations provide the equating functions that can be linear or nonlinear (e.g., the equipercentile case) depending on the bandwidths.

In step five the standard error of equating (*SEE*) and the standard error of equating difference (*SEED*) are calculated. The *SEE* specifies the degree of uncertainty in the estimated equating functions, whereas *SEED* is the difference between the two equating function *SEEs*. The general formula for calculating *SEE* for equating \bar{X} to \bar{Y} is

$$SEE_Y(x) = \hat{\sigma}_Y(x) = \sqrt{\text{Var}(\hat{e}_Y(x))} \quad (\text{G.36})$$

and for the equating \bar{Y} to \bar{X} *SEE* is given by

$$SEE_X(y) = \hat{\sigma}_X(y) = \sqrt{\text{Var}(\hat{e}_X(y))} . \quad (\text{G.37})$$

The calculations for *SEE* differ for PSE and CE. For example, utilizing CE one has $SEE_{Y(CE)}(x) = \sqrt{[SEE_Y(e_A(x))]^2 + [e'_Y(e_A(x))SEE_A(x)]^2}$. For both PSE and CE it is assumed that the bandwidths, h_x and h_y , are fixed values and not functions of the estimated score probabilities \bar{R} and \bar{S} (von Davier et al., 2004). *SEE* calculations are most appropriate for large sample sizes and may not be valid with small samples (von Davier et al., 2004).

Research has shown several factors may affect the accuracy of KE equating. For example, the fit of kernel smoothing has been found to improve with increased sample sizes (Lee, 2007). This research also suggests that kernel smoothing improves with increased test length. Research comparing KE to linear, equipercentile, and TCC equating methods shows that KE performs well (e.g., de Ayala, Smith, & Norman Dvorak, 2018; Mao et al., 2006; von Davier et al., 2006) and to TCC equating. Software for performing kernel equating include KE (von Davier, Holland, & Thayer, 2004) or the R packages *kequate* (Andersson, Bränberg, & Wiberg, 2013, 2020) and *SNSequate*.

With *kequate* there are three phases. In the first phase, the package's *kefreq* function is used to create the frequency distributions. The second phase involves fitting a series of generalized linear models via the *glm* function. These models vary in complexity from nonadditive to additive models as well as involving varying powers. For instance, one model might include the terms $X, X^2, X^3, X^4, A, A^2, A^3$, and XA where X represents the test form, A represents the anchor test, and XA is a first-order interaction. From these models the best model is selected for equating on the basis of, for example, the smallest AIC or BIC. After which the model is applied to each data set for the equating of Form Y scores to Form X scores using the *kequate* function (phase 3).

CORRESPONDENCE BETWEEN THE RASCH MODEL AND A LOGLINEAR MODEL

In Chapter 2 we introduced our math data set (see Table 2.1) as patterns with the corresponding frequencies of occurrence. Another way to summarize our responses is as a contingency table. For example, Table G.11 presents the crossing of all five items and X .

TABLE G.11. Contingency Table for Math Data										
I1	I2	I3	I4	I5	X					
					0	1	2	3	4	5
0	0	0	0	0	691	0	0	0	0	0
				1	0	184	0	0	0	0
			1	0	0	158	0	0	0	0
				1	0	0	41	0	0	0
		1	0	0	0	235	0	0	0	0
				1	0	0	87	0	0	0
			1	0	0	0	65	0	0	0
				1	0	0	0	15	0	0
	1	0	0	0	0	242	0	0	0	0
				1	0	0	79	0	0	0
			1	0	0	0	92	0	0	0
				1	0	0	0	28	0	0
		1	0	0	0	0	134	0	0	0
				1	0	0	0	52	0	0
			1	0	0	0	0	63	0	0
				1	0	0	0	0	40	0
1	0	0	0	0	0	2280	0	0	0	0
				1	0	0	571	0	0	0
			1	0	0	0	462	0	0	0
				1	0	0	0	166	0	0
		1	0	0	0	0	1053	0	0	0
				1	0	0	0	412	0	0
			1	0	0	0	0	370	0	0
				1	0	0	0	0	187	0
	1	0	0	0	0	0	1685	0	0	0
				1	0	0	0	626	0	0
			1	0	0	0	0	702	0	0
				1	0	0	0	0	500	0
		1	0	0	0	0	0	1682	0	0
				1	0	0	0	0	1219	0
			1	0	0	0	0	0	2095	0
				1	0	0	0	0	0	3385

As can be seen, our zero variance response vectors ($X = 0, X = 5$) consist simply of zero frequencies except for the intersection of the cells corresponding to a pattern of 00000 or of 11111. Of these cases, we have 691 individuals with $X = 0$ and 3385 respondents with $X = 5$ for a total of 4076 cases with uninformative response vectors vis à vis estimating θ via MLE. Consequently, there are 15,525 remaining cases that have useful response vectors.

We can fit a generalized linear model to these data using a log link function with a categorical systematic component and assuming a Poisson distribution for the random component (i.e., the distribution of responses); see Agresti (1990). In this context, our *linear* model (i.e., linear in parameters) using a *log* link is called a loglinear model.⁸ As such and generally speaking, our table consists of three factors: responses, items, and scores. We have L items ($j = 1, 2, \dots, L$) each of which has two responses ($k = 1, 2; m_j = 2$). As mentioned in Chapter 2, with the Rasch model we have $L - 1$ unique ability estimates. Because all individuals obtaining the same X obtain the same $\hat{\theta}$ we can collect these respondents into score groups. Consequently, when using MLE for person estimation we have at most $L - 1$ score groups using. However, because it is possible that we may not observe each X from 1 to $L - 1$ let o represent the number of uninformative response vectors (i.e., $o \geq [(L + 1) - (L - 1)]$). Thus, our data consist of $(L + 1 - o)$ score groups ($i = 1, 2, \dots, (L + 1 - o)$); for Table G.11 $o = 2$. Let a table's observed frequencies be f_{ijk} and the corresponding expected frequencies by ϕ_{ijk} with

$$\phi_{ijk} = \pi \pi_i^{\text{scoregrp}} \pi_j^{\text{item}} \pi_k^{\text{response}} \pi_{ij}^{\text{scoregrp*item}} \pi_{ik}^{\text{scoregrp*response}} \pi_{jk}^{\text{item*response}} \pi_{ijk}^{\text{scoregrp*item*response}} \quad (\text{G.38})$$

or alternatively,

$$\begin{aligned} \ln(\phi_{ijk}) = & \lambda_{\cdot} + \lambda_i^{\text{scoregrp}} + \lambda_j^{\text{item}} + \lambda_k^{\text{response}} + \\ & \lambda_{ij}^{\text{scoregrp*item}} + \lambda_{ik}^{\text{scoregrp*response}} + \lambda_{jk}^{\text{item*response}} + \\ & \lambda_{ijk}^{\text{scoregrp*item*response}}, \end{aligned} \quad (\text{G.39})$$

where λ_{\cdot} is the overall effect (constant); $\lambda_i^{\text{scoregrp}}$, λ_j^{item} , $\lambda_k^{\text{response}}$, are the main effects of the score group, item, and response, respectively; $\lambda_{ij}^{\text{scoregrp*item}}$, $\lambda_{ik}^{\text{scoregrp*response}}$, $\lambda_{jk}^{\text{item*response}}$, are the first-order interaction effects, and $\lambda_{ijk}^{\text{scoregrp*item*response}}$ is the second-order interaction of score group, item and response. Because Equation G.39 is the saturated model it contains all main and interaction effects and will yield $\phi_{ijk} = f_{ijk}$. Equation G.39 may also be written as

$$\begin{aligned} \ln(\mu_{ijk}) = & \lambda_{\cdot} + \lambda_i^{\text{scoregrp}} + \lambda_j^{\text{item}} + \lambda_k^{\text{response}} + \\ & \lambda_{ij}^{\text{scoregrp*item}} + \lambda_{ik}^{\text{scoregrp*response}} + \lambda_{jk}^{\text{item*response}} + \\ & \lambda_{ijk}^{\text{scoregrp*item*response}} \end{aligned} \quad (\text{G.40})$$

to reflect our expected cell counts and a GLM notation.

Following Mellenbergh and Vijn (1981) we have as our logit for a response of 1

$$\ln \left[\frac{p_{ij}}{1-p_{ij}} \right] = \ln \left[\frac{\ln(\mu_{ij1})}{1-\ln(\mu_{ij1})} \right] = \ln \left[\frac{\ln(\mu_{ij1})}{\ln(\mu_{ij0})} \right], \quad (\text{G.41})$$

where $\ln(\mu_{1j0}) = 1 - \ln(\mu_{1j1})$. Applying the quotient rule for logarithms we have

$$\ln \left[\frac{p_{ij}}{1-p_{ij}} \right] = -\ln(\mu_{1j1}) - \ln(\mu_{1j0}). \quad (\text{G.42})$$

By appropriate substitution of Equation G.40 into G.42 we obtain

$$\begin{aligned} \ln \left[\frac{p_{ij}}{1-p_{ij}} \right] = & [\lambda_{\cdot} + \lambda_i^{\text{scoregrp}} + \lambda_j^{\text{item}} + \lambda_1^{\text{response}} + \\ & \lambda_{ij}^{\text{scoregrp*item}} + \lambda_{i1}^{\text{scoregrp*response}} + \lambda_{j1}^{\text{item*response}} + \lambda_{ij1}^{\text{scoregrp*item*response}}] \\ & - [\lambda_{\cdot} + \lambda_i^{\text{scoregrp}} + \lambda_j^{\text{item}} + \lambda_0^{\text{response}} + \\ & \lambda_{ij}^{\text{scoregrp*item}} + \lambda_{i0}^{\text{scoregrp*response}} + \lambda_{j0}^{\text{item*response}} + \lambda_{ij0}^{\text{scoregrp*item*response}}], \end{aligned} \quad (\text{G.43})$$

where for the minuend $k = 1$, for the subtrahend $k = 0$, and with the constraints

$$\begin{aligned} \sum_j^L \lambda_j^{\text{item}} &= \sum_{i=1}^{L+1-o} \lambda_i^{\text{scoregrp}} = \sum_{k=0}^1 \lambda_k^{\text{response}} = 0, \\ \sum_{j=1}^L \lambda_{ij}^{\text{scoregrp*item}} &= \sum_{i=1}^{L+1-o} \lambda_{ij}^{\text{scoregrp*item}} = \sum_{j=1}^L \lambda_{jk}^{\text{item*response}} = \sum_{k=0}^1 \lambda_{jk}^{\text{item*response}} \\ &= \sum_{i=1}^{L+1-o} \lambda_{ik}^{\text{scoregrp*response}} = \sum_{k=0}^1 \lambda_{ik}^{\text{scoregrp*response}} = 0 \\ \sum_j^L \lambda_{ijk}^{\text{scoregrp*item*response}} &= \sum_{i=1}^{L+1-o} \lambda_{ijk}^{\text{scoregrp*item*response}} = \sum_{k=0}^1 \lambda_{ijk}^{\text{scoregrp*item*response}} = 0. \end{aligned}$$

After cancellation of like terms; substitution of the equivalencies $\lambda_1^{\text{response}} = -\lambda_0^{\text{response}}$, $\lambda_1^{\text{scoregrp*response}} = -\lambda_0^{\text{scoregrp*response}}$, $\lambda_1^{\text{item*response}} = -\lambda_0^{\text{item*response}}$, $\lambda_1^{\text{scoregrp*item*response}} = -\lambda_0^{\text{scoregrp*item*response}}$; and factoring we obtain the saturated model

$$\ln \left[\frac{p_{ij}}{1-p_{ij}} \right] = 2\lambda_1^{\text{response}} + 2\lambda_{i1}^{\text{scoregrp*response}} + 2\lambda_{j1}^{\text{item*response}} + 2\lambda_{ij1}^{\text{scoregrp*item*response}}. \quad (\text{G.44})$$

The terms $\lambda_{j1}^{\text{item*response}}$, $\lambda_{i1}^{\text{scoregrp*response}}$, and $\lambda_{ij1}^{\text{scoregrp*item*response}}$ reflect our item and person effects as well as their interaction effect on the logit, respectively. As mentioned above, the saturated model fits the data perfectly. Thus, if we assume the highest-order interac-

tion equals 0 we obtain a more parsimonious model that may be nonsignificantly different from the saturated model

$$\ln \left[\frac{p_{ij}}{1-p_{ij}} \right] = 2\lambda_1^{\text{response}} + 2\lambda_{i1}^{\text{scoregrp*response}} + 2\lambda_{j1}^{\text{item*response}}. \quad (\text{G.45})$$

By letting our person (i.e., score group) and item effects be represented as $\theta_i = 2\lambda_{i1}^{\text{scoregrp*response}}$, $\delta_j^E = 2\lambda_{j1}^{\text{item*response}}$, and noting that $2\lambda_1^{\text{response}}$ is constant (C) we have

$$\ln \left[\frac{p_{ij}}{1-p_{ij}} \right] = C + \theta_i + \delta_j^E \quad (\text{G.46})$$

with constraints $\sum_{j=1}^L \delta_j^E = \sum_{i=1}^{L+1-o} \theta_i = 0$.

Equation G.46 is essentially equivalent to Equation G.13 ($\delta_j = -\delta_j^E$) with C “absorbed” into the parameters (Mellenbergh & Vijn, 1981). As such, from the perspective of estimating the parameters the models are equivalent. To demonstrate this point we fit a loglinear model to our math contingency table (Table G.11) using the R `glm` function.⁹ Table G.12 contains our R session.

After reading our data we calculate the observed score (`mathdat$X = rowSums(mathdat)`) and obtain its frequency distribution using the `table` function. We have 691 cases with $X = 0$ and 3385 individuals with $X = 5$. Because we compare our estimates to those from those using JMLE we remove these 4076 zero variance response vectors from our data frame using the `subset` function. Our new data frame (`mathdatR`) has 15,525 cases (`nrow(mathdatR)`). Using our five items and X we create our contingency table (`cntngncytblNoZeroVar`) for analysis (`with(..., table(i1,i2,i3,i4,i5,X))`). Comparing our contingency table (`fTable(...)`) with the one presented in Table G.11 shows that the only difference is the elimination of the zero variance response vectors. Our last preparatory step is to convert our contingency table to a data frame (`cntngncytblNoZeroVar_df=as.data.frame(...)`). As a result, a new variable containing the cell frequencies (`Freq`) is created. It is this variable that we use in our modeling.

We pass our table data frame, specifying a model containing each of our items and X as well as the Poisson distribution (`llNoZeroVar = glm(Freq ~ i1+i2+i3+i4+i5+X, ..., family = poisson)`). We obtain convergence in 7 iterations. Our estimates are $\hat{\delta}_1^E = 2.22436$, $\hat{\delta}_2^E = 0.38511$, $\hat{\delta}_3^E = -0.01378$, $\hat{\delta}_4^E = -0.74904$, and $\hat{\delta}_5^E = -0.98964$. By default the `glm` function models the 1 value. Consequently, to convert our estimates from the easiness scale we multiply them by -1 (i.e., $\delta_j = -\delta_j^E$). Therefore, we convert these easiness estimates to the difficulty metric. To compare our estimates to those of BIGSTEPS (Table 3.4, JMLE, $N = 15,525$) we apply the mean-sigma method (see Chapter 11) to link this metric to that of BIGSTEPS's. Our transformed estimates are $\hat{\delta}_1^* = -2.22723$, $\hat{\delta}_2^* = -0.23364$, $\hat{\delta}_3^* = 0.19872$, $\hat{\delta}_4^* = 0.99568$, and $\hat{\delta}_5^* = 1.25647$. Comparing the two sets of estimates shows a mean absolute deviation of 0.00765 with a correlation of 0.99992; the corresponding plot shows the points falling essentially on a straight line.

TABLE G.12. glm Session for the Loglinear Calibration of the Mathematics Data

```

> sessionInfo()
  R version 3.6.0 (2019-04-26)
> mathdat=read.table("math.dat",col.names=c(paste0("i",1:5)))

> mathdat$X=rowSums(mathdat)                                # calculate observed (summed) score

> table(mathdat$X)
  0    1    2    3    4    5
691 3099 4269 4116 4041 3385

> mathdatR=subset(mathdat, (X > 0) & (X < 5))             # eliminate X=0 & X=5 cases
> nrow(mathdatR)                                           # number of cases
[1] 15525

> table(mathdatR$X)
  1    2    3    4
3099 4269 4116 4041

> cntngncytblNoZeroVar=with(mathdatR, table(i1,i2,i3,i4,i5,X))

> dim(cntngncytblNoZeroVar)                                # contingency table's dimensions 2x2x2x2x2x4
[1] 2 2 2 2 2 4

> ftable(cntngncytblNoZeroVar)
      X      1      2      3      4
i1 i2 i3 i4 i5
0  0  0  0  0      0      0      0      0
      1      184      0      0      0
      1  0      158      0      0      0
      1      0      41      0      0      0
  1  0  0      235      0      0      0
      1      0      87      0      0      0
      1  0      0      65      0      0
      1      0      0      0      15      0
  1  0  0  0      242      0      0      0
      1      0      79      0      0      0
      1  0      0      92      0      0
      1      0      0      0      28      0
  1  0  0      0      134      0      0
      1      0      0      52      0      0
      1  0      0      0      63      0
      1      0      0      0      0      40
  1  0  0  0  0      2280      0      0      0
      1      0      571      0      0      0
      1  0      0      462      0      0
      1      0      0      166      0      0
  1  0  0      0      1053      0      0
      1      0      0      412      0      0
      1  0      0      0      370      0
      1      0      0      0      187      0
  1  0  0  0      0      1685      0      0
      1      0      0      626      0      0
      1  0      0      0      702      0
      1      0      0      0      500      0
  1  0  0      0      0      1682      0
      1      0      0      0      0      1219
      1  0      0      0      0      2095
      1      0      0      0      0      0

```

(continued)

TABLE G.12. (continued)

```

> cntngncytblNoZeroVar_df=as.data.frame(cntngncytblNoZeroVar)

> # note the 0s & 1s reflect category labels NOT the responses

> head(cntngncytblNoZeroVar_df,5)
  i1 i2 i3 i4 i5 X Freq
1  0  0  0  0  0  1   0
2  1  0  0  0  0  1 2280
3  0  1  0  0  0  1  242
4  1  1  0  0  0  1   0
5  0  0  1  0  0  1  235

> tail(cntngncytblNoZeroVar_df,5)
  i1 i2 i3 i4 i5 X Freq
124 1  1  0  1  1  4  500
125 0  0  1  1  1  4   0
126 1  0  1  1  1  4  187
127 0  1  1  1  1  4   40
128 1  1  1  1  1  4   0

> # glm models ref of 1 - this indicated by the second digit on the item label
  in the
> # Coefficients table. For example, i11 means variable i1 using a reference
  value of 1

> llNoZeroVar = glm(Freq ~ i1+i2+i3+i4+i5+X, data = cntngncytblNoZeroVar, family =
  poisson)

> summary(llNoZeroVar )
Call:
glm(formula = Freq ~ i1 + i2 + i3 + i4 + i5 + X, family = poisson,
    data = cntngncytblNoZeroVar)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-33.570  -16.358   -7.471   -3.604   74.378

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept)  3.41816    0.03363 101.642 <2e-16 ***
i11          2.22436    0.02705  82.246 <2e-16 ***
i21          0.38511    0.01635  23.554 <2e-16 ***
i31         -0.01378    0.01605  -0.859    0.39
i41         -0.74904    0.01719 -43.573 <2e-16 ***
i51         -0.98964    0.01806 -54.807 <2e-16 ***
X2           0.32030    0.02360  13.572 <2e-16 ***
X3           0.28380    0.02378  11.933 <2e-16 ***
X4           0.26541    0.02388  11.116 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 64405  on 127  degrees of freedom
Residual deviance: 46602  on 119  degrees of freedom
AIC: 46839

Number of Fisher Scoring iterations: 7

> llNoZeroVar$converged
[1] TRUE
# convergence achieved


```

R INTRODUCTION

Downloading R

Go to: <https://www.r-project.org/>

Click on download R



The R Project for Statistical Computing

Getting Started

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To [download R](#), please choose your preferred [CRAN mirror](#).

If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

News

- **R version 3.3.1 (Bug in Your Hair) prerelease versions** will appear starting Saturday 2016-06-11. Final release is scheduled for Tuesday 2016-06-21.
- **R version 3.3.0 (Supposedly Educational)** has been released on 2016-05-03.
- **R version 3.2.5 (Very, Very Secure Dishes)** has been released on 2016-04-14. This is a rebadging of the quick-fix release 3.2.4-revised.
- **Notice XQuartz users (Mac OS X)** A security issue has been detected with the Sparkle update mechanism used by XQuartz. Avoid updating over insecure channels.
- The **R Logo** is available for download in high-resolution PNG or SVG formats.
- **useR! 2016**, will take place at Stanford University, CA, USA, June 27 - June 30, 2016.
- **The R Journal Volume 7/2** is available.
- **R version 3.2.3 (Wooden Christmas-Tree)** has been released on 2015-12-10.
- **R version 3.1.3 (Smooth Sidewalk)** has been released on 2015-03-09.

[Home]

Download
CRAN

R Project
About R
Logo
Contributors
What's New?
Mailing Lists
Bug Tracking
Development Site
Conferences
Search

R Foundation
Foundation
Board
Members
Donors
Donate


Documentation
Manuals
FAQs
The R Journal
Books
Certification
Other

Pick your mirror site (you'll have to scroll to get to the USA section):

CRAN Mirrors	
The Comprehensive R Archive Network is available at the following URLs, please choose a location close to you. Some statistics on the status of the mirrors can be found here: main page , windows release , windows old release .	
0-Cloud	
https://cloud.r-project.org/	Automatic redirection to servers worldwide, currently sponsored by Rstudio
http://cloud.r-project.org/	Automatic redirection to servers worldwide, currently sponsored by Rstudio
Algeria	
http://cran.usthb.dz/	University of Science and Technology Houari Boumediene
Argentina	
http://mirror.fcaglp.unlp.edu.ar/CRAN/	Universidad Nacional de La Plata
Australia	
http://cran.csiro.au/	CSIRO
http://cran.ms.unimelb.edu.au/	University of Melbourne
Austria	
https://cran.wu.ac.at/	Wirtschaftsuniversität Wien
http://cran.wu.ac.at/	Wirtschaftsuniversität Wien

<p>USA</p> <p>http://stat-www.st-andrews.ac.uk/cran/</p> <p>https://cran.cnr.berkeley.edu/</p> <p>http://cran.cnr.berkeley.edu/</p> <p>http://cran.stat.ucla.edu/</p> <p>http://mirror.las.iastate.edu/CRAN/</p> <p>http://ftp.usg.iu.edu/CRAN/</p> <p>https://rweb.crmda.ku.edu/cran/</p> <p>http://rweb.crmda.ku.edu/cran/</p> <p>https://cran.mtu.edu/</p> <p>http://cran.mtu.edu/</p> <p>http://cran.wustl.edu/</p> <p>http://archive.linux.duke.edu/cran/</p> <p>http://cran.case.edu/</p> <p>http://iis.stat.wright.edu/CRAN/</p> <p>http://ftp.osuosl.org/pub/cran/</p> <p>http://lib.stat.cmu.edu/R/CRAN/</p> <p>http://cran.mirrors.hoobly.com/</p> <p>https://mirrors.nics.utk.edu/cran/</p> <p>http://mirrors.nics.utk.edu/cran/</p> <p>https://cran.revolutionanalytics.com/</p> <p>http://cran.revolutionanalytics.com/</p> <p>https://cran.fhcr.org/</p> <p>http://cran.fhcr.org/</p> <p>Venezuela</p> <p>http://camoruco.ing.uc.edu.ve/cran/</p> <p>Many of these sites can also be accessed using FTP.</p> <p>If you want to host a new mirror at your institution, please have a look at the CRAN Mirror HOWTO.</p>	<p>St Andrews University</p> <p>University of California, Berkeley, CA</p> <p>University of California, Berkeley, CA</p> <p>University of California, Los Angeles, CA</p> <p>Iowa State University, Ames, IA</p> <p>Indiana University</p> <p>University of Kansas, Lawrence, KS</p> <p>University of Kansas, Lawrence, KS</p> <p>Michigan Technological University, Houghton, MI</p> <p>Michigan Technological University, Houghton, MI</p> <p>Washington University, St. Louis, MO</p> <p>Duke University, Durham, NC</p> <p>Case Western Reserve University, Cleveland, OH</p> <p>Wright State University, Dayton, OH</p> <p>Oregon State University</p> <p>Statlib, Carnegie Mellon University, Pittsburgh, PA</p> <p>Hoobly Classifieds, Pittsburgh, PA</p> <p>National Institute for Computational Sciences, Oak Ridge, TN</p> <p>National Institute for Computational Sciences, Oak Ridge, TN</p> <p>Revolution Analytics, Dallas, TX</p> <p>Revolution Analytics, Dallas, TX</p> <p>Fred Hutchinson Cancer Research Center, Seattle, WA</p> <p>Fred Hutchinson Cancer Research Center, Seattle, WA</p> <p>Universidad de Carabobo Venezuela</p>
--	---

Pick your platform:



The Comprehensive R Archive Network

Download and Install R

Precompiled binary distributions of the base system and contributed packages. Windows and Mac users most likely want one of these versions of R:

- [Download R for Linux](#)
- [Download R for \(Mac\) OS X](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all Platforms

Windows and Mac users most likely want to download the precompiled binaries listed in the upper box, not the source code. The sources have to be compiled before you can use them. If you do not know what this means, you probably do not want to do it!

- The latest release (2016-05-03, Supposedly Educational) [R-3.3.0 tar.gz](#), read [what's new](#) in the latest version.
- Sources of [R alpha and beta releases](#) (daily snapshots, created only in time periods before a planned release).
- Daily snapshots of current patched and development versions are [available here](#). Please read about [new features and bug fixes](#) before filing corresponding feature requests or bug reports.
- Source code of older versions of R is [available here](#).
- [Contributed extension packages](#)

Questions About R

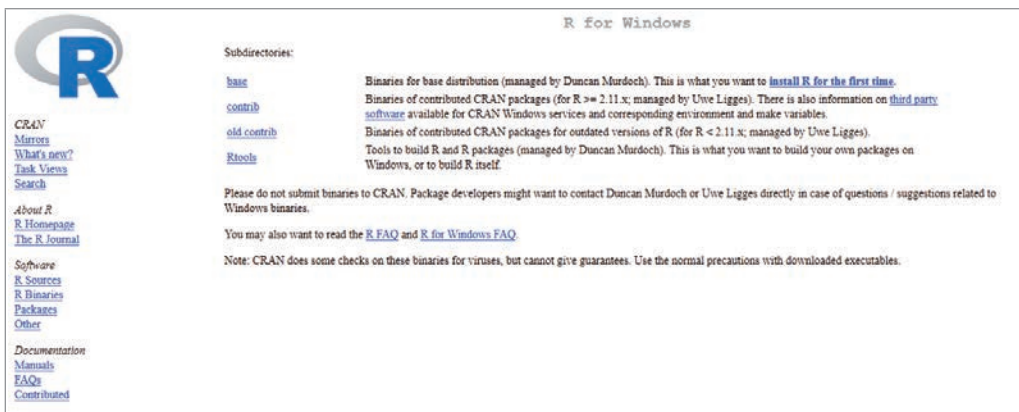
- If you have questions about R, like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

What are R and CRAN?

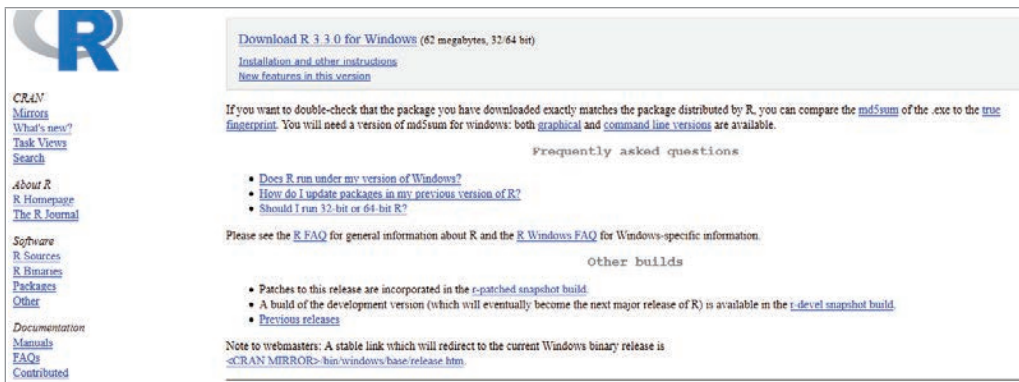
R is 'GNU S', a freely available language and environment for statistical computing and graphics which provides a wide variety of statistical and graphical techniques: linear and nonlinear modelling, statistical tests, time series analysis, classification, clustering, etc. Please consult the [R project homepage](#) for further information.

CRAN is a network of ftp and web servers around the world that store identical, up-to-date, versions of code and documentation for R. Please use the CRAN [mirror](#) nearest to you to minimize network load.

This is what a Windows site looks like:



Clicking on install R for the first time produces:



Clicking on the icon that appears on the desktop after the download will install the package.

One can perform analyses and/or data manipulations from within the R environment or from a shell such as RStudio. RStudio has editing and management features that are either not available or more convenient to use than those in R. Using RStudio is recommended.

- RStudio can be downloaded from <https://www.rstudio.com/> for free.
- Reference Card: <https://cran.r-project.org/doc/contrib/Short-refcard.pdf>
- Codeschool (interactive): <http://tryr.codeschool.com/levels/1/challenges/2>
- RStudio Introduction: <https://www.youtube.com/watch?v=jPk6-3prknk>
- Getting started with R and RStudio: <https://www.youtube.com/watch?v=IVKMsaWju8w>

Installing Packages in R

In addition to the base R routines one will install additional packages that contain routines (i.e., functions) to accomplish a specific analysis or task. If you are using RStudio one simply selects `Install Packages...` (found in the `Tools` menu) and types the package name in the `Packages` (separate multiple with space or comma) field. Below we outline the steps within R.

As an example to install the `mirt` package we would use:

```
> install.packages("mirt")
```

You will be asked to select a mirror site from which to download the package. Here's part of the process:

```
> install.packages("mirt")
Installing package into 'C:/Users/Ralph/Documents/R/win-library/3.4'
(as 'lib' is unspecified)
--- Please select a CRAN mirror for use in this session ---
trying URL 'https://mirror.las.iastate.edu/CRAN/bin/windows/contrib/3.4/mirt_1.26.3.zip'
Content type 'application/zip' length 2395876 bytes (2.3 MB)
downloaded 2.3 MB

package 'mirt' successfully unpacked and MD5 sums checked

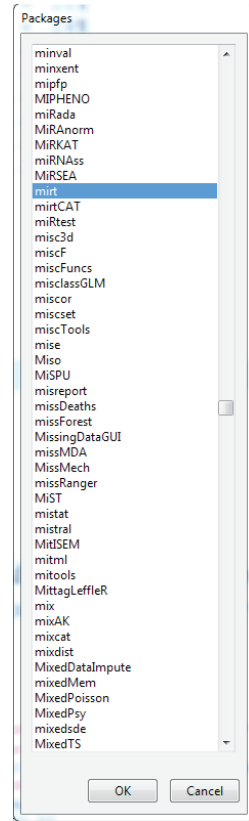
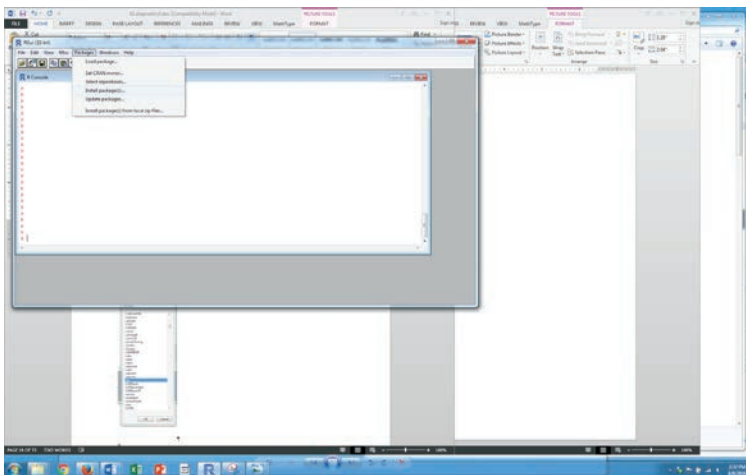
The downloaded binary packages are in
  C:\Users\Ralph\AppData\Local\temp\RtmpAnIXdN\downloaded_packages
. |
```

(Two useful plotting packages are `ggplot2` and `lattice`. Each of these would have to be installed as done with `mirt`.)

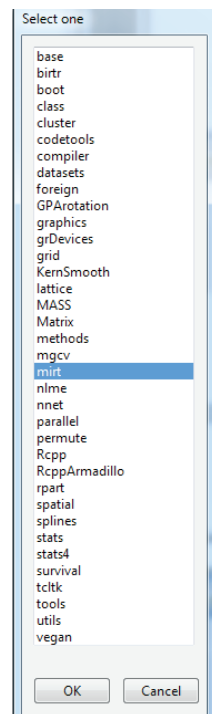
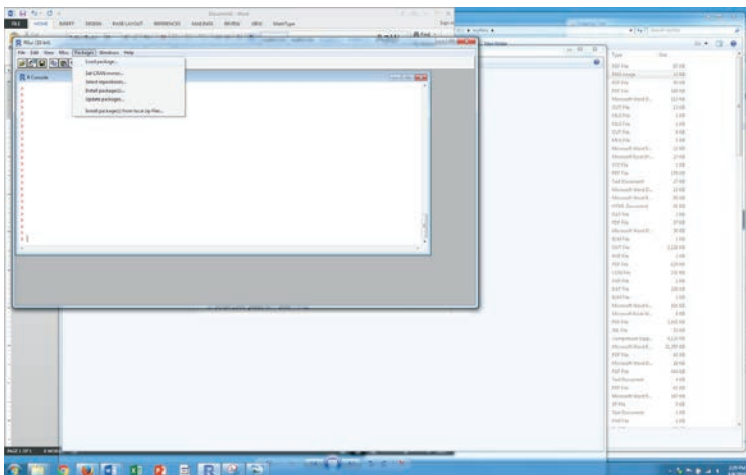
The `mirt` package now exists on the harddrive. To use it we need to tell R to load it into our session using the `library` command (quotes are not needed in the current R version):

```
> library("mirt")
Loading required package: stats4
Loading required package: lattice
> |
```

An alternative to typing a command to install the `mirt` package – using the GUI



Loading the `mirt` package through the GUI:



Note 1: Spelling of each command is specific.

For example, `> install.package("Hmisc")`

`Error: could not find function "install.package"`

This error was caused because the command was misspelled (i.e., `'package'` instead of `'packages'`).

Note 2: The case used with each command is important

For example, `> install.package("hmisc")`

`Warning messages:`

`1: package 'hmisc' is not available (for R version 3.3.0)`

`2: Perhaps you meant 'Hmisc' ?`

This error was caused because the `"h"` was not capitalized.

For example, `> Install.packages("Hmisc")`

`Error: could not find function "Install.packages"`

This error was caused because the `"I"` was capitalized and should not have been.

Note 3: One uses double quotes with the `install.packages` command; the `library` command will work with or without double quotes

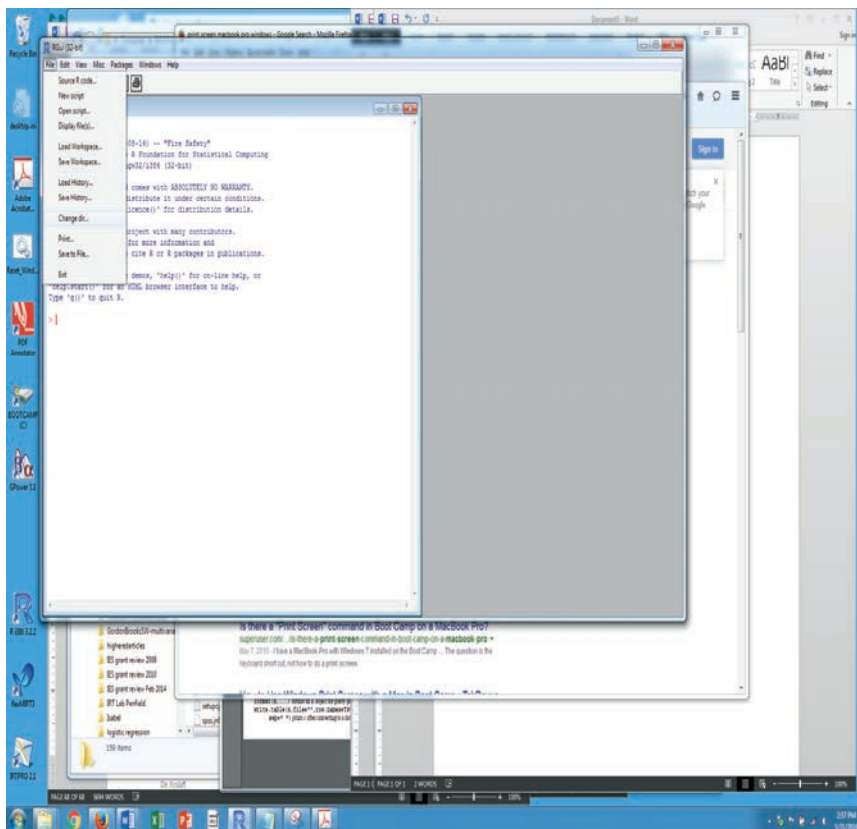
Note 4: Installing the `latticeExtra` package will be necessary for some of the plotting routines in `mirt`.

Note 5: A useful Reference "card" containing R commands is found at: <https://cran.r-project.org/doc/contrib/Short-refcard.pdf>

Using R

Upon starting R a workspace is created that, by default, can be saved with the .RData extension. To see what exists in the workspace use the `ls()` function. To quit R use the File menu, type `q()`, or simply close the window. Note 1: Commands are typed after “>” and R *commands* are case sensitive.

Change the default directory to be the folder containing your data file.



(In RStudio select the “Set Working Directory\Choose Directory...” submenu item from the Session menu.)

Alternatively, one can use the `setwd` function. Because in R a backward slash (i.e., “\”) is the escape character in characters strings one needs to use a double backward slash in path names or a forward slash (i.e., “/”). For example, `setwd("c:\\folder_name1\\folder_name2")` or `setwd("c:/folder_name1/folder_name2")`.

You can use the `getwd()` function to determine the working directory.

Here's part of the beginning and end of our data file, `math.dat`, to show the data's format. As can be seen the data are separated by a space (i.e., space delimited) and we have variable names on first line.

```
i1 i2 i3 i4 i5
 1  1  0  0  0
 1  1  1  0  0
:
 1  1  1  0  0
 1  0  0  0  0
```

To read my data set, `math.dat`, we use the `read.table` command with the `header` subcommand to read the variable names that appear on the first line of the file (i.e., `mathdata = read.table("math.dat", header=TRUE)`); if our data were tab or comma delimited we would use the field separator option (e.g., for tab delimited `read.table("math.dat", header=T, sep="\t")`, for comma delimited `read.table("math.dat", header=T, sep=",")` or `read.csv`; in Europe: `read.csv2`). We then display the first five and last three cases using the `head` and `tail` commands, respectively, to verify the data were read in correctly.

```
> head(mathdata, n=5)
  i1 i2 i3 i4 i5
1  1  1  0  0  0
2  1  1  1  0  0
3  1  0  0  0  0
4  1  1  1  0  0
5  1  0  1  1  0
```

```
> tail(mathdata, n=3)
  i1 i2 i3 i4 i5
19599 1  1  1  1  1
19600 1  1  0  1  1
19601 1  1  1  1  0
```

Note: The *file name* in the `read.table` command is not case sensitive. For example, `MATH.DAT`, `math.dat`, `MatH.Dat`, etc.) are considered the same. However, if we had typed `"MATHdata"` in the `head` command then R would tell me that the *data frame*, `MATHdata`, was not found because in the `read.table` command the data frame was called `mathdata`.

Updating R

New versions of R are periodically released. Potentially a new package may not execute with an older version of R (i.e., the package may not be backward compatible). The most convenient way to address these releases is to use `updateR()`. To use `updateR()` one first needs to install it:

```
> install.packages("installr")
```

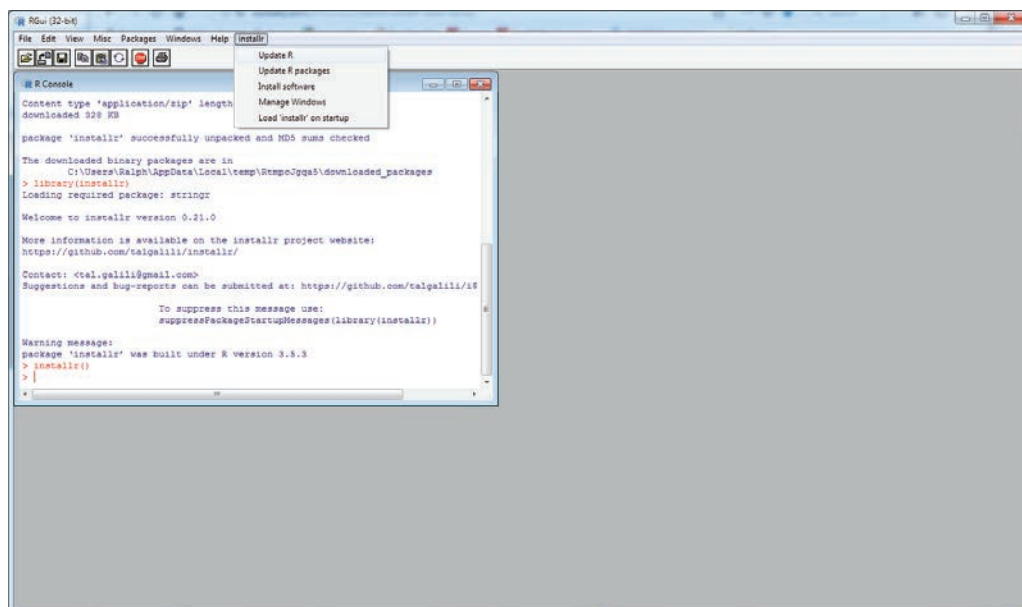
To use `updateR()` one loads the library:

```
> library(installr)
```

To update/upgrade your version you can use the command:

```
> installr()
```

Or you can use the GUI interface. Select `Update R` from the `installr` menu:



`Update R` will tell you the most current version of R and the version of R you are running. Alternatively, to determine your version of R:

```
> R.Version
```

This process will also allow you to update any packages you have installed.

NOTES

1. A tetrachoric correlation coefficient specifies the association between two variables that are continuous and assumed to be normally distributed, but that are artificially dichotomized. These variables, X and Y , may be the dichotomization of a manifest variable(s) in a sample (e.g., X = “males 30 years and older” vs. “males below 30” and Y = “females 30 years and older” vs. “females below 30”) or may be a theoretical dichotomization as discussed in Appendix C “Conceptual Development of the Normal Ogive Model” (i.e., the responses of 0 and 1 are assumed to arise from dichotomizing two continuous normally distributed latent variables). Consequently, the coefficient is an estimate of the linear relationship between the two continuous variables if the correlation was calculated using the two continuous variables. Cross-classifying the two dichotomous variables creates a 2×2 contingency table. This table is graphically represented in Figure G.7 with the variables’ normal distributions on the table’s margins. With respect to the variables’ normal distributions, the symbol z is the standard score corresponding to the p proportion of 1s for variable Y (i.e., p is the marginal proportion of 1s and $(1 - p)$ is the marginal proportion of 0s for variable Y). The height of the unit normal curve at z is denoted by Y . In an analogous fashion, and with respect to the variable X , z' delimits the p' proportion of 1s and the ordinate value at z' is symbolized as Y' . The cross-classification of the 1s and 0s for variables X and Y leads to a fourfold table with the cells’ frequencies labeled by the letters A, B, C, and D.

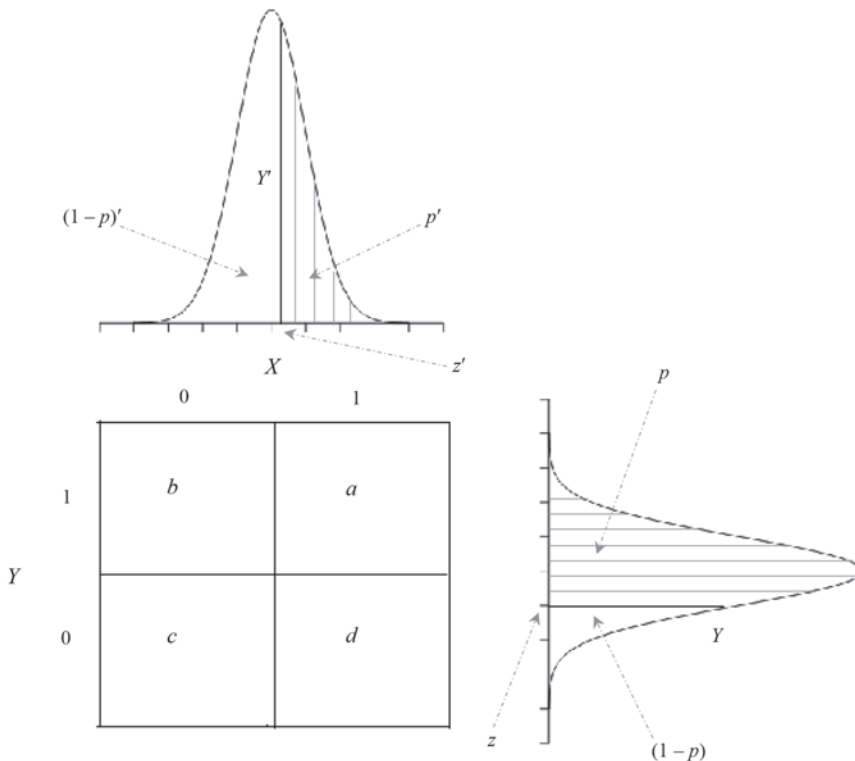


FIGURE G.7. Relationship between two continuous normally distributed variables and their dichotomization.

One equation for calculating the tetrachoric correlation, r_T , involves the power series (Guilford & Fruchter, 1978)

$$r_T = r_T + r_T^2 \left(\frac{zz'}{2} \right) + r_T^3 \left(\frac{(z^2 - 1)(z'^2 - 1)}{6} \right) + \dots \cong \frac{ad - bc}{YY'N^2}, \quad (\text{G.47})$$

where N is the number of cases and all other terms are defined above. Equation G.47 shows that r_T contains the unknowns z and z' as well as Y and Y' . Therefore, obtaining r_T is a complex process and several approaches have been developed. For example, Divgi (1979) and Bonett and Price (2005) contain two different methods for estimating the tetrachoric correlation coefficient.

There are several situations that may lead to problems in the accuracy of the r_T estimate. For instance, if the dichotomized variable(s) has extreme proportions (e.g., p and/or p' is 0.90), guessing is present, and/or the normality assumption is not tenable, then one obtains a biased estimate of the true population relationship. As a result, the magnitude of the observed coefficients may be inappropriately large and outside the range -1 to 1 . In addition, there is an increased chance of observing non-Gramian matrices (i.e., a matrix with negative eigenvalue[s]) when factor analyzing tetrachorics.

As previously mentioned, and as is the case with the analysis of phi coefficients, it is possible to observe difficulty factors with the factor analysis of a tetrachoric correlation matrix (Gourlay, 1951). In the situation where items may be correctly answered on the basis of guessing, then the tetrachoric correlation is adversely affected. However, Carroll (1945) provides an approach for correcting tetrachoric correlations for chance success; see Reckase (1981; cited in Green et al., 1984) concerning problems with overcorrecting tetrachoric correlations. An approach for testing assumptions for tetrachoric correlations is presented by Muthén and Hofacker (1988). Guilford and Fruchter (1978) suggest that "for estimating the *degree* [italics added] of correlation . . . it is recommended that N be at least 200, and preferably 300" (p. 315), as well as to avoid calculating r_T when there is a zero frequency in one cell.

2. A monotonic transformation preserves the inequalities of the untransformed values. That is, a transformation, say $f(\square)$, is monotonic if for $x_0 < x_1$ one has that $f(x_0) < f(x_1)$. The graph of $f(\square)$ as a function of x would appear as a line that either increases or plateaus, but never decreases (i.e., a monotonically increasing function). Conversely, $f(\square)$ is a monotonic transformation if for $x_0 < x_1$, then $f(x_0) > f(x_1)$. In this latter case the graph of $f(\square)$ as a function of x would appear as a line that either decreases or plateaus, but never increases (i.e., a monotonically decreasing function). Examples of monotonic transformation are $x^* = 1/x$, $x^* = e^x$, $x^* = \ln(x)$, and sometimes $x^* = x^2$; in the case of x^2 one needs the restrictions of either $x \geq 0$ or $x \leq 0$.

3. The probability of a response of 0 for the 1PL model is

$$\begin{aligned} p(x_j = 0 | \theta, \alpha, \delta_j) &= 1 - \frac{e^{\alpha(\theta - \delta_j)}}{1 + e^{\alpha(\theta - \delta_j)}} = \left[\frac{1 + e^{\alpha(\theta - \delta_j)}}{1 + e^{\alpha(\theta - \delta_j)}} \right] - \left[\frac{e^{\alpha(\theta - \delta_j)}}{1 + e^{\alpha(\theta - \delta_j)}} \right] \\ &= \frac{1 + e^{\alpha(\theta - \delta_j)} - e^{\alpha(\theta - \delta_j)}}{1 + e^{\alpha(\theta - \delta_j)}} = \frac{1}{1 + e^{\alpha(\theta - \delta_j)}}. \end{aligned} \quad (\text{G.48})$$

Sometimes the 1PL model's exponent is written to contain the response, x_j . That is,

$$p(x_j = 1 | \theta, \alpha, \delta_j) = \frac{e^{x_j \alpha(\theta - \delta_j)}}{1 + e^{\alpha(\theta - \delta_j)}}. \quad (\text{G.49})$$

Equation G.49 can be used to calculate the probability of a response of 0 and a response of 1.

4. Although the strata are ordered in terms of their average item location, this does not mean that the probability of a response of 1 will always have a direct relationship with the strata's order. In the context of proficiency assessment this ordering would be with respect to the stratum's average difficulty. Whenever we use a model that allows for item discrimination and/or the IRF's lower asymptote to vary, then it is possible to have crossing IRFs. When IRFs cross, then some individuals have a higher probability of a response of 1 on a more difficult item than on an easier item. For example, Figure G.8 contains the IRFs for two items that differ in their discriminations and difficulties. Specifically, item 1 has an $\alpha_1 = 2$ and $\delta_1 = 0.5$, whereas item 2 has an $\alpha_2 = 0.8$ with $\delta_2 = 1.5$. As can be seen, an individual with a θ above the IRFs' intersection point (e.g., $\theta = 1.0$) has a higher probability of correctly answering the easier item 1 than the more difficult (in terms of δ) item 2. This result is consistent with ordering the items according to difficulty (i.e., $\delta_2 > \delta_1$). However, for θ s below the IRFs' intersection point (e.g., at $\theta = -1.0$) the probability of a correct response on the more difficult item 2 is greater than on the easier item 1. Stated another way, for low-proficiency people item 2 (the "hard" item) is actually "easier" than item 1 because their probability of a correct response is higher for item 2 than it is for item 1. This item level observation may be extended to strata ordered by average item location. Specifically, each of our respondents may not have a probability of a response of 1 that decreases as the average stratum difficulty increases.

5. The different location estimates for the same item across groups is a reflection of the indeterminacy of metric issue discussed above. Recall each metric is defined with respect to the sample that was used. Consequently, the metric for the high group is not the same as for the low group. The difference in metrics is reflected in the points falling above the identity line (Figure G.5); the mean location for the high and low groups are -1.316 and 2.882 , respectively. Ignoring this issue leads one to interpret the estimated item location of -1.369 as an "easy" item and the item estimated to be at 2.840 as a "difficult" item. However, once the low-group metric and the

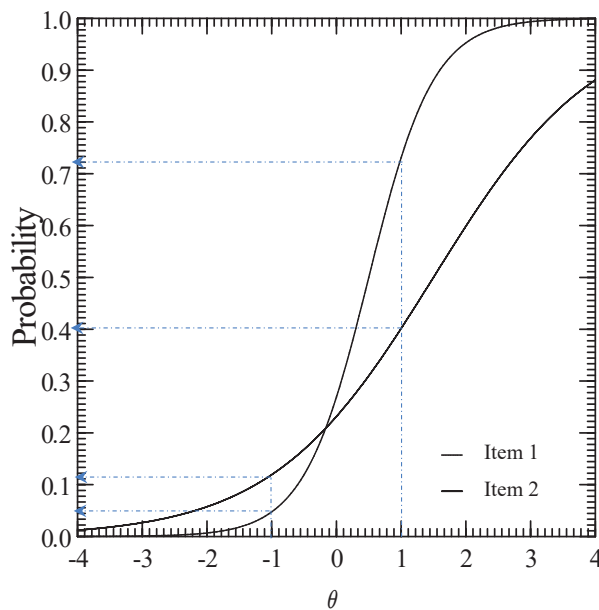


FIGURE G.8. IRFs for two items with different α_i s.

high-group metric are aligned with one another the interpretation of whether the item is “easy” or “difficult” is consistent across groups. One way of thinking about this is to consider the low-group metric to be analogous to the Celsius scale and the high-group metric to be analogous to the Fahrenheit scale. Whether a temperature of 30° is interpreted to be “hot” or “cold” depends on whether it is on the Celsius or Fahrenheit scale. However, once the two scales are aligned, the interpretation of 30° is the same regardless of which temperature scale one is referencing. This linking of the Celsius and Fahrenheit scales is accomplished using the (perfect) linear relationship that exists between the two scales. The linking of the different metrics is discussed in Chapter 11.

6. The best way to examine this question requires knowledge of the examinees’ true locations, θ_s . In this way the quality of the estimated person locations may be directly assessed by comparing the estimates with their true values. An experiment in which the researcher knows parameter values (e.g., examinees’ true locations) and then uses a computer to model the behavior of the construct of interest is known as a computer *simulation study*. If a simulation study uses a model (e.g., a regression model, the Rasch model) to generate the (simulated) data, then the technique is known as a *Monte Carlo* simulation study. Therefore, to address the question, “How do the characteristics of the instrument affect the person location estimates?” we conduct a *Monte Carlo* study. For this simulation 40 z -scores are randomly sampled from a unit normal distribution; this may be accomplished by using a normal distribution random number generator from a statistical package or spreadsheet program. These z -scores are considered to be the items’ location parameters, δ_s . In addition, 1000 z -scores are randomly sampled from a unit normal distribution. These z -scores serve as the person location parameters, θ_s ; these pseudo “people” are sometimes referred to as *simulees*.

The response data for each simulee is obtained in two phases. In the first phase the simulee’s probability of a response of 1 to item j is calculated according to a model (e.g., the Rasch model) using the appropriate parameters (e.g., δ_j and the simulee’s θ). In the second phase this probability is compared to a uniform random number [0, 1]. If the random number is less than or equal to the probability of the response 1, then the simulee’s response to item j is coded as 1, otherwise it is coded as 0. These phases are repeated to obtain the simulee’s responses to the remaining items on the instrument. The entire process is repeated for each of the remaining simulees. Harwell, Stone, Hsu, and Kirisci (1996) and Paxton, Curran, Bollen, Kirby, and Chen (2001) contain more information on conducting Monte Carlo studies.

7. Divgi (1986) argues that because of maximum likelihood bias in person locations one cannot have instrument-free estimation whenever finite instruments differ in their item locations. Moreover, it is not clear that we can *always* have a measure of a person’s location that is unaffected by the items on an instrument. For instance, consider an item that would be considered to be at the synthesis level of Bloom’s taxonomy of educational objectives for the cognitive domain. When an examinee encounters such an item it is possible that the process of synthesizing the relevant information leads to the person learning something that previously they did not know. As such, the examinee’s location shifts from where it would have been if they had not encountered the item. Therefore, we would have one ($\hat{\theta}$) if the person is administered the synthesis item and a different ($\hat{\theta}$) if they had been given a different (e.g., a knowledge level) item, albeit in the synthesis item’s content domain. In short, whether the item is at the synthesis or knowledge level affects our person location and its estimate. Because the item-person interaction is not immutable the measurement of individuals is not always independent of the administered items. Therefore, item-invariant measurement must be interpreted to refer to the *result* of the person–item interaction and not that the item does not affect the person. Moreover, although

we may accumulate invariance evidence across different groupings (e.g., male versus females, high versus low ability) this does not mean that we would necessarily obtain invariance evidence across other possible groupings, languages, and/or in other cultures. As such, because our invariance evidence is conditional on the specific groups, language, and culture that are used we suggest the use of the term *conditional invariance* (e.g., conditional person-parameter invariance and conditional item-parameter invariance). The best that we can do is accumulate evidence that supports our contention of invariant measurement.

8. Goodman (1978) and other use the term log-linear model.

9. Our corresponding SPSS syntax is given below. We first compute our observed score X followed by eliminating our zero variance response vectors ($X = 0$ and $X = 5$) before calling GENLOG. Consequently, our call to GENLOG involves a contingency table with 15,525 examinees. Because by default SPSS models 0 our estimates are on the difficulty scale.

```
COMMENT compute observed score.
COMPUTE X=i1+i2+i3+i4+i5.
EXECUTE.

COMMENT select cases non zero variance response vectors.
USE ALL.
COMPUTE filter_$=(X>0 and X<5).
VARIABLE LABELS filter_$ 'X>0 and X<5 (FILTER)'.
VALUE LABELS filter_$ 0 'Not Selected' 1 'Selected'.
FORMATS filter_$ (f1.0).
FILTER BY filter_$.
EXECUTE.

COMMENT loglinear analysis.
GENLOG i1 i2 i3 i4 i5 X
/MODEL=POISSON
/PRINT=FREQ RESID ADJRESID ZRESID DEV ESTIM CORR COV ITERATION
/PLOT=RESID(ADJRESID) NORMPROB(ADJRESID)
/CRITERIA=CIN(95) ITERATE(20) CONVERGE(0.001) DELTA(.5)
/DESIGN i1 i2 i3 i4 i5 X.
```

Our estimates are $\hat{\delta}_1 = -2.224$, $\hat{\delta}_2 = -0.385$, $\hat{\delta}_3 = 0.014$, $\hat{\delta}_4 = 0.749$, and $\hat{\delta}_5 = 0.990$. Applying the mean-sigma method (see Chapter 11) to link this metric to that of BIGSTEPS's (Table 3.4, JMLE; $N = 15,525$) we obtain $\hat{\delta}_1^* = -2.223$, $\hat{\delta}_2^* = -0.230$, $\hat{\delta}_3^* = 0.203$, $\hat{\delta}_4^* = 1.000$, and $\hat{\delta}_5^* = 1.260$. Comparing the two sets of estimates shows a mean absolute deviation of 0.013 with a correlation of 0.99992; the corresponding plot shows the points falling essentially on a straight line. (Because the BIGSTEPS estimates are presented to two decimal places the accuracy of the mean absolute deviation is adversely affected.)