The Disappearing Beta Trick

The essential attributes of a Rasch model are sufficient statistics and separable parameters, which allow, but don't guarantee, specific objectivity. Well, actually sufficient statistics come pretty close if they really are sufficient to capture all the relevant information in the data. We will come back to this in the discussion of what Rasch called *control of the model* and most of us call *goodness of fit*. The current topic is a demonstration, more intuitive than mathematical, of how to manipulate the model to estimate item difficulties.

The process begins with the basic Rasch model for how likely the person wins when <u>one</u> person takes <u>one</u> dichotomous item:

$$15. \quad Prob(Correct\ Response \mid B_\nu, \Delta_i) = \frac{B_\nu}{B_\nu + \Delta_i}.$$

B_ν represents the ability of person ν and Δ_i the difficulty of item i. The complementary expression for when the item wins instead of the person is:

$$16. \quad Prob(Incorrect\ Response \mid B_\nu, \Delta_i) = \frac{\Delta_i}{B_\nu + \Delta_i}.$$

And, of course, the two outcomes, right or wrong, cover the universe of possibilities so the probabilities add to one.

If one person takes <u>two</u> items, there are four possible outcomes: both right, both wrong, the first right and the second wrong, or the first wrong and the second right. If the responses are independent, then the probabilities for the four outcomes can be computed with the product of the individual, independent probabilities. The four possibilities, in Table III.1, cover all the possible outcomes of the two item test so these probabilities must also sum to one but that takes a little more algebra to demonstrate.

The upper left (both wrong) and the lower right (both right) provide no information about the relative difficulties of the two items; they only say that both are either too easy or too hard for our person; nothing about which of the two might be easier or harder. We are only interested in the two possibilities with one item correct and the other not, which are the upper right and lower left cells. So our universe has gotten smaller. We are making our analysis *conditional* on being in one of the two shaded cells, i.e., *conditional on a raw score of exactly one item correct*.

Table III.1: Probabilities for a Two-Item Test

		Item 1	
		Incorrect	**Correct**
ITEM 2	**Incorrect**	$\frac{\Delta_1}{B_\nu + \Delta_1} \frac{\Delta_2}{B_\nu + \Delta_2}$	$\frac{B_\nu}{B_\nu + \Delta_1} \frac{\Delta_2}{B_\nu + \Delta_2}$
	Correct	$\frac{\Delta_1}{B_\nu + \Delta_1} \frac{B_\nu}{B_\nu + \Delta_2}$	$\frac{B_\nu}{B_\nu + \Delta_1} \frac{B_\nu}{B_\nu + \Delta_2}$

The *conditional* probability that item *1* is the correct one given a total score of 1 is the probability in the upper right cell divided by the sum of the upper right and lower left. This forces the probabilities for our new universe to again sum to one and the result reduces miraculously.

$$17. \qquad \textit{Prob(Item 1 Correct |1 or 2 Correct)} = \frac{\dfrac{\mathrm{B}_\nu \Delta_2}{(\mathrm{B}_\nu + \Delta_1)(\mathrm{B}_\nu + \Delta_2)}}{\dfrac{\mathrm{B}_\nu \Delta_1 + \mathrm{B}_\nu \Delta_2}{(\mathrm{B}_\nu + \Delta_1)(\mathrm{B}_\nu + \Delta_2)}} = \frac{\Delta_2}{\Delta_1 + \Delta_2}.$$

Of course, there is an analogous expression for the probability that item *2* is the one correct given a total score of *1*.

$$18. \qquad \textit{Prob(Item 2 Correct |1 or 2 Correct)} = \frac{\Delta_1}{\Delta_1 + \Delta_2}.$$

The person's ability has disappeared completely from the expression for item difficulty. This was possible because the parameters were separated and is also a somewhat backdoor demonstration that raw score is the sufficient statistic. The sufficiency argument follows because we restricted ourselves to the cases with a raw score of one. This is a very small illustration of Rasch's very Big Idea.

Person ability has completely vanished from the expressions for item difficulty.

Specific objectivity (aka, *person-freed* item estimation) means that, while it is necessary to involve people to obtain data, it does not matter what people. Within reason.

Doing the Math: Connecting the Model to Data

If many people take the same two items, counts can be tabulated for the four possible outcomes. This provides very straightforward estimates for the probabilities in the Table III.1 and for expressions 17 and 18. The probability in the upper right can be estimated by:

$$19. \qquad P_{10} = \frac{N_{10}}{N},$$

and similarly for the lower left cell, where N_{10} is the count of people who answered item 1 correctly and item 2 incorrectly, N_{01} is the count of people who answered item 1 incorrectly and item 2 correctly, and N is the total number tested.

From expression 17,

$$20. \qquad \frac{N_{10}/N}{N_{01}/N + N_{10}/N} = \frac{N_{10}}{N_{01} + N_{10}} = \frac{D_2}{D_1 + D_2}.$$

The Latin D replaced the Greek Δ once we had brought real data into the equation to indicate estimates of parameters rather than parameters themselves. A similar expression can be wrought from equation 18. At first glance, they seem to provide two equations with two unknowns, but they reduce to the same thing.

$$21. \qquad \frac{D_1}{D_2} = \frac{N_{01}}{N_{10}}, \qquad \text{or its reciprocal.}$$

This expression makes intuitive sense. D_1 is the difficulty of item *1* and N_{01} is the count of people who found item 1 harder than item 2; N_{10} is the count of people who found item 1 easier than item 2. If the two counts are equal, then the items are equally difficult and D_1 equals D_2. If N_{01} is large compared to N_{10}, then D_1 is large compared to D_2. The abilities of the people in the sample have no effect on either of these ratios and for good reason. Whether or not an item is easier or harder than some other item has nothing to do with the ability of any people who might or might not take the items.

There are some loose ends that should be tidied up. First, it may not be obvious that expression 20 for a group of people follows immediately from expression 17 for one person. We just need to invoke the mathematical face of a Rasch model: *Separability*. If the probability of Item 1 correct and Item 2 incorrect for some arbitrary person is given by the upper right cell of Table III.1, the expected number of people with that pattern is found by summing the probabilities for everyone in the group:

$$22. \qquad N_{10} = \sum_{v} P_{10} = \sum_{v} \frac{\mathrm{B}_v \Delta_2}{(\mathrm{B}_v + \Delta_1)(\mathrm{B}_v + \Delta_2)} .$$

Because Δ_2 does not have the subscript for the people, it can be moved in front of the summation symbol; every term involving the people will cancel out as before whether we are talking about one person or everyone in the US, leaving expressions 17, 18, and 21.

Next, the equations have been kept uncluttered by expressing the parameters and their estimates in the exponential metric rather than logits. The model would look more familiar, although a little messier and a lot harder to type, if e^{β_v} were substituted for B_v and e^{δ_i} were substituted for Δ_i. Using logits, expression 21 becomes:

$$23. \qquad d_1 - d_2 = \ln(N_{01}) - \ln(N_{10}) \qquad \text{or its negative.}$$

Finally, in order to actually attach numbers to d_1 and d_2 with one equation and two unknowns, we need to adopt some convention for where zero is. It can take many forms; e.g., $d_1 = k$, or $d_2 = k$, or more generally $c_1 d_1 + c_2 d_2 = k$, where c and k are convenient constants. The most convenient is $d_2 = 0$. The most common is $d_1 + d_2 = 0$ where c_i and k take the very convenient values of 1 and 0 respectively. Then we can finish up with:

$$24. \qquad d_1 = \frac{\ln(N_{01}) - \ln(N_{10})}{2} \qquad \text{and} \qquad d_2 = -d_1 = \frac{\ln(N_{10}) - \ln(N_{01})}{2} .$$

This is one possible convention that gives actual values to d_1 and d_2. Another popular choice is to anchor one of the items, say, $d_2 = x$. Then from (23),

$$25. \qquad d_1 = \ln(N_{01}) - \ln(N_{10}) + x \qquad \text{and} \qquad d_2 = x.$$

With either convention, the relative difficulties are the same and both preserve the relationship of expression 23: $d_1 - d_2 = \ln(N_{01}) - \ln(N_{10})$. Any other choice for c_i and k would be just as legitimate and useable; perhaps not as convenient nor conventional.

If You Ever Use Tests Longer than Two Items

The previous section would be cute but not be very profound if it did not hold for tests longer than two items. Choppin's Pairwise algorithm extends the logic to all possible pairs of items. For each pair *i* and *j*, count the people who passed *i* but failed *j* and vice versa, just like we did for items 1 and 2 in the preceding sections. Modifying that notation a little, n_{ij} is the count of the people who attempted both *i* and *j* and who passed *j* but failed *i*. In the old notation, we had n_{01} instead of n_{ij} if item 1 is *i* and item 2 is *j*; and n_{ji} replaces n_{10}.

If there are a total of *L* items, the n_{ij} can be arranged in an *LxL* matrix, call it ***N****, in which row *i* contains the counts where item *i* was <u>incorrect</u> and column *j* contains counts where item *j* was <u>correct</u>.

The matrix ***N**** of these counts is converted to a matrix ***R*** of logarithms of ratios by dividing n_{ij} by n_{ji} and taking the natural log. More succinctly,

26. $$r_{ij} = \ln(n_{ij} / n_{ji}) = \ln(n_{ij}) - \ln(n_{ji}) = d_i - d_j \,.$$

If all the off-diagonal elements of ***N**** contain non-zero entries[1], then every element of ***R*** will contain a value $r_{ij} = d_i - d_j$. This is exactly the same (with slightly different notation) as expression 23. As a matter of housekeeping, every diagonal element of ***R*** should be defined as zero, which in fact represents the case $d_k - d_k \equiv 0$. Obtaining the total for each row, including the zero on the diagonal,

27. $$\sum_{j=1}^{L} r_{ij} = \sum_{j=1}^{L}\left(d_i - d_j\right) = Ld_i - \sum_{j=1}^{L} d_j \,.$$

Imposing the generalized convention that $\sum_{j=1}^{L} d_j = 0$ and rearranging a little:

28. $$d_i = \frac{\sum_{j=1}^{L} r_{ij}}{L} \,.$$

Extracting the estimate for any δ_i is trivial in the complete case where ***N**** and ***R*** are full: the estimate d_i is the average of row *i* of ***R***. Once again, problem solved with no mathematical gymnastics.

Completing the Sum: A Non-iterative Solution

Expression 27 applies to the situation in which all off-diagonal elements of ***N**** are non-zero. This simply means that every item was paired with every other item and that the item sometimes won and sometimes lost compared to the other member of the pair. The world is not always so accommodating.

The solution of the previous section can be expressed succinctly, if somewhat more eruditely, in matrix notation as:

29. $$\mathbf{A}\underline{\mathbf{d}} = \underline{\mathbf{S}},$$

---

[1] The diagonal, where i = j, will always be zero because the two conditions cannot hold simultaneously.

where ***A*** is an *LxL* matrix of known (but not yet disclosed) coefficients; ***S*** is an Lx1 vector containing the sum of each row of the matrix ***R***; and ***d*** is an *Lx1* vector of the item difficulty estimates d_i that we are pursuing. In the complete case, ***A*** is an *LxL* diagonal matrix with the constant *L* along the diagonal. This is just a more pedantic way of expressing the answer of the last section, which was, *find the row averages*.

If the matrices ***N**** and consequently ***R*** are not complete, expression 29 is still the answer but the matrix ***A*** is not a simple diagonal. For example, if element (r_{ik}) is missing because either n_{ik} or n_{ki} or both were zero, then the sums of both rows *i* and *k* are deficient. Part of equation 27 is missing. Because the element $(r_{ik})=(d_i - d_k)$ is not there, the row sum is changed:

30. $$\sum_{j=1;\, j\neq k}^{L} r_{ij} = Ld_i - \sum_{j=1}^{L} d_j - (d_i - d_k),$$

which starts with expression 27 and subtracts the piece that isn't there. Writing the expression in this form makes it easy to adopt the same convention, i.e., $\sum_{j=1}^{L} d_j = 0,$ and to envision a modified version of the ***A*** matrix.

31. $$(L-1)d_i + d_k = \sum_{j=1}^{L} r_{ij} \ ,$$

which compares to expressions 27 and 29. More missing elements would mean a smaller coefficient for d_i and additional d_k to be added back in.

To repeat in general terms, the matrix ***A*** required by expression 29 begins with an *LxL* matrix with *L* for each diagonal element and *0* in each off-diagonal element. After the ***A***-matrix has been constructed assuming no missing cells, the off-diagonal elements of the ***N****-matrix are scanned[2] row-by-row for situations where n_{ij} is zero, and if one is found,

- Subtract 1 from the diagonal element *(i,i)* of ***A***, and
- Add 1 to the off-diagonal element *(i,j)* of ***A***.

The rows of A will always sum to *L*. Then the estimates of the item difficulties can be obtained with expression 29, whenever a solution exists. Typically, a solution will exist unless a row (and column) contains no non-zeros or there are blocks of items that are not connected to other blocks. This is the same as saying two forms are not linked through common items or common persons.

Demonstration of Pairwise Calculations

The data presented in Table III.2 demonstrate the required calculations for a simple case. The data were simulated using 500 examinees with logit ability equal to zero and five items with logit difficulties of (-3, -1, 0, 1, 3) with no random component. The ***N****-matrix is the number expected to pass one and fail one item in each pairing. The value 17 in the second column of the first row means 17 examinees failed item 1 and passed item 2. The value 128 in the first column of the

[2] As described here, the procedure scans every row in its entirety. More efficient, but more difficult to describe, algorithms can be readily devised that would only scan the upper or lower triangle of the matrix. The small gain in computing efficiency hardly makes it worth the effort.

second row indicates 128 examinees passed item 1 but failed item 2. There are no empty cells so the solution is easy.

Table III.2: Demonstration of Pairwise Calculations (N = 500; Ability = 0.0)

Logit Difficulties	*N**-matrix of counts				
-3	0	**17**	12	6	1
-1	**128**	0	67	36	6
0	238	183	0	67	12
1	348	267	183	0	17
3	454	348	238	128	0

The ***R***-matrix below is computed from the ***N****-matrix above. For example, the value in the first row, second column of ***R*** is $ln\ (n_{12} / n_{21}) = ln\ (17 / 128) = -2.019$. Analogously, the value in the second row, first column is $ln\ (n_{21} / n_{12}) = ln\ (128 / 17) = 2.019$. We now have data indicating that item 1 is two logits easier than item 2. The final two steps in the calculation are to sum the five values in each row and, since there are no missing values, divide by 5.

Table III.3: Demonstration of Pairwise Calculations (N = 500; Ability = 0.0)

R-Matrix of Log Ratios					Row Sum	Recovered Parameters
0	**-2.019**	-2.987	-4.060	-6.118	-15.185	**-3.037**
2.019	0	-1.005	-2.004	-4.060	-5.050	**-1.010**
2.987	1.005	0	-1.005	-2.987	0	**0**
4.060	2.004	1.005	0	-2.019	5.050	**1.010**
6.118	4.060	2.987	2.019	0	15.185	**3.037**

The values inside Table III.3 are the comparisons of each pair of items; exactly what we would have gotten if we had treated this as 10 two-item tests. The row averages in the last column consolidate all the information and express the difficulty for each item as its distance from the center. This happened because we chose the convention $\sum_{j=1}^{L} d_j = 0$; if we choose a different convention the numbers would be different but the relationships the same.

To illustrate the estimates really do not depend on the ability distribution of the sample, the demonstration was repeated for four levels of ability (0.0, 1.0, 2.0 and 3.0). The recovered parameters are shown in Table III.4 for the four cases.

Table III.4: Recovered Parameters for Four Abilities

Original Difficulty	Ability 0	1	2	3
-3	-3.037	-3.024	-3.031	-2.992
-1	-1.010	-0.999	-1.000	-0.972
0	0.000	-0.017	-0.002	0.026
1	1.010	1.025	1.041	1.016
3	3.037	3.016	2.992	2.921

With a sample mean of 3.5 logits, the situation became unbalanced enough to leave one cell (i.e., cell 1,5) of the ***N****-matrix empty. This in turn leaves two cells (*1,5* and *5,1*) of the ***R***-matrix undefined. It is still possible to obtain estimates but it involves a little more effort.

Table III.5: Demonstration with Ability = 3.5 (N = 500)

Logit Difficulty	*N**-Matrix of Counts				
-3	0	1	1	1	**0**
-1	5	0	5	5	3
0	15	14	0	14	9
1	38	38	37	0	24
3	188	187	183	174	0

R-Matrix of Log Ratios					Row Sum	Recovered Difficulties
0	-1.609	-2.708	-3.638		-7.955	-2.730
1.609	0	-1.030	-2.028	-4.132	-5.581	-1.116
2.708	1.030	0	-0.972	-3.012	-0.246	-0.049
3.638	2.028	0.972	0	-1.981	4.657	0.931
	4.132	3.012	1.981	0	9.126	2.964

When some counts are zero, at least some of the five equations must be solved simultaneously and so the ***A***-matrix of coefficients contains some non-zero off-diagonal elements. Completely filled rows are as easy as ever. For example from table III.4,
$d_2 = -5.581 / 5 = -1.116$.

The ***A***-matrix of coefficients, expression 28, needs to reflect the empty cells and the reduced number of terms in the summations. Using the data from Table III.5:

$$32. \quad \begin{bmatrix} 4 & 0 & 0 & 0 & 1 \\ 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 5 & 0 \\ 1 & 0 & 0 & 0 & 4 \end{bmatrix} \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ d_4 \\ d_5 \end{bmatrix} = \begin{bmatrix} -7.955 \\ -5.581 \\ -0.246 \\ 4.657 \\ 9.126 \end{bmatrix}.$$

The equations involving items 1 and 5 are readily solved with any of a variety of tools for solving simultaneous equations. This example is simple enough to do by hand.

Because the estimation procedure is based on conditioning out ability, one might ask if the demonstration works only if it is based on groups with homogeneous abilities. Table III.6 demonstrates that using a sample with a mixture of abilities does not affect the result. This is a more general demonstration of Rasch's *Specific Objectivity*.

Table III.6: Demonstration of Pairwise Calculations for Mixed Ability Sample (0, 1, 2, 3, & 3.5)

Generating Difficulties	*N**-Matrix of Counts					Total Count	Generating Abilities
-3	0	8	6	4	1	500	0, 1, 2, 3, & 3.5
-1	56	0	37	24	7		
0	117	99	0	53	16		
1	206	178	145	0	34		
3	422	379	326	252	0		

R-Matrix of Log Ratios					Row Sum	Recovered Difficulties
0	-1.998	-2.937	-4.119	-6.068	-15.122	-3.024
1.998	0	-0.986	-1.992	-4.015	-4.996	-0.999
2.937	0.986	0	-1.005	-3.002	-0.083	-0.017
4.119	1.992	1.005	0	-1.992	5.124	1.025
6.068	4.015	3.002	1.992	0	15.078	3.016

All these demonstrations were done with no random error component in the simulations, so one might expect to recover the generating parameters exactly. The issue preventing this is counts of examinees must be whole numbers. Had we used the theoretical Σp unrounded for N_{ij}, the recovery would have been perfect.

While we keep insisting that the distribution of ability does not matter, the ***N****-matrix in Table III.5 illustrates the inefficiency of an off-target sample. While the estimates are fully conditional so that they "do not depend on the ability distribution of the examinees", the estimates will be based on fewer examinees with an off-target sample, implying poorer estimates. While tables III.3 and III.5 began with the same number (500), approximately 90% provided useful data comparing items 1 and 5 in the first case and less than 40% did in the second case. Only six of the 500 cases provided data for the comparison of items 1 and 2 in Table III.5. The very low counts in the first row of Table III.5 contributed to significant rounding error and to a recovered parameter that differed by 0.27 logits from the original.

The solution is, don't give people inappropriate tests.