

Recognizing Uncertainty in the Q-Matrix via a Bayesian Extension of the DINA Model

Applied Psychological Measurement

36(6) 447–468

© The Author(s) 2012

Reprints and permission:

sagepub.com/journalsPermissions.nav

DOI: 10.1177/0146621612449069

<http://apm.sagepub.com>**Lawrence T. DeCarlo¹****Abstract**

In the typical application of a cognitive diagnosis model, the Q-matrix, which reflects the theory with respect to the skills indicated by the items, is assumed to be known. However, the Q-matrix is usually determined by expert judgment, and so there can be uncertainty about some of its elements. Here it is shown that this uncertainty can be recognized and explored via a Bayesian extension of the DINA (deterministic input noisy and) model. The approach used is to specify some elements of the Q-matrix as being random rather than as fixed; posterior distributions can then be used to obtain information about elements whose inclusion in the Q-matrix is questionable. Simulations show that this approach helps to recover the true Q-matrix when there is uncertainty about some elements. An application to the fraction-subtraction data of K. K. Tatsuoaka suggests a modified Q-matrix that gives improved relative fit.

Keywords

cognitive diagnosis, DINA, Q-matrix, Bayesian analysis, fraction-subtraction data

The goal in cognitive diagnosis, as the name suggests, is to “diagnose” which skills examinees have or do not have. This approach potentially offers more useful feedback to examinees than a simple overall score derived from classical test theory or item response theory, and so cognitive diagnosis models (CDMs) have become popular in recent years (for reviews, see DiBello, Roussos, & Stout, 2007; Fu & Li, 2007; Rupp & Templin, 2008b). A useful CDM that has been widely studied is the DINA (deterministic input noisy and) model (Haertel, 1989; Junker & Sijtsma, 2001; Macready & Dayton, 1977).

The use of a CDM requires the specification of a Q-matrix (K. K. Tatsuoaka, 1990); the Q-matrix indicates the set of skills that are required to answer a particular item correctly. The Q-matrix is usually determined by expert judgment, and so there can be uncertainty about some of its elements. Here, it is noted that this uncertainty can be recognized by using a Bayesian approach, with some elements of the Q-matrix specified as being random rather than as fixed. Posterior distributions can then be used to obtain information about the inclusion or exclusion of these elements, as shown in the following. The Bayesian approach allows one to explore

¹Columbia University, NY, USA

Corresponding Author:

Lawrence T. DeCarlo, Department of Human Development, Box 118, Teachers College, Columbia University, 525 West 120th Street, New York, NY 10027-6696, USA

Email: decarlo@tc.edu

possible modifications of the Q-matrix, ideally as suggested by expert judgment or by substantive considerations. It is shown that the approach is simple to implement, particularly with a reparameterized version of the DINA model. Simulations that explore benefits and limitations of the approach are presented, and the widely analyzed fraction-subtraction data of K. K. Tatsuoka (1990) are examined.

Uncertainty in the Q-Matrix

Most researchers who have worked with CDMs have recognized that there is often uncertainty with respect to at least some elements of the Q-matrix. For example, virtually every researcher who has analyzed the fraction-subtraction data has suggested possible modifications of the Q-matrix (e.g., DeCarlo, 2011; de la Torre, 2009; de la Torre & Douglas, 2008; Henson, Templin, & Willse, 2009; C. Tatsuoka, 2002; K. K. Tatsuoka, 1990), and so there is clearly uncertainty as to the correct specification of some elements. There are several approaches to this problem. One is to consider a set of alternative Q-matrices and to fit the models associated with these matrices; indices of relative fit, such as the Bayesian information criterion (BIC) or Akaike information criterion (AIC), can then be used to help determine the appropriate Q-matrix. This approach has been used by a number of researchers (e.g., Barnes, Bitzer, & Vouk, 2005; Cen, Koedinger, & Junker, 2005; DeCarlo, 2011; de la Torre & Douglas, 2008; Rupp & Templin, 2008a). A limitation of this approach, however, is that the number of possible Q-matrices grows quickly as the number of uncertain elements increases. For example, the following simulation examines a situation where there is uncertainty about 12 elements of the Q-matrix (out of 60 elements), which gives $2^{12} = 4,096$ possible Q-matrices, and so a large number of models would need to be fitted and compared.

An approach to this problem was offered by de la Torre (2008), in which he suggested using a sequential search algorithm, which helps to avoid the large number of possibilities associated with an exhaustive search. In particular, a sequential search algorithm was used with a fit statistic that minimizes the sum of the average slip and guess parameters. There are, however, some limitations to that approach. For example, de la Torre noted that the approach is based on a particular fit statistic, δ , and that other statistics might be more useful or appropriate. Another problem is that the sequential search algorithm might not lead to the best solution; this is suggested below in a reanalysis of the fraction-subtraction data of K. K. Tatsuoka (1990).

Another way to recognize uncertainty is via a Bayesian approach, as has been noted by several researchers. For example, J. Templin and Henson (2006) used a Bayesian approach to allow for uncertain elements in the Q-matrix, in the same manner as done here. Similarly, Muthén and Asparouhov (2010) discussed using a Bayesian approach to recognize uncertainty in factor analysis models (e.g., to recognize uncertainty about factor loadings) and other (structural equation and regression) models. It is shown here that a Bayesian approach is useful and is simple to implement, particularly with a reparameterized version of the DINA model.

The DINA Model and a Reparameterization

The basic idea underlying the DINA model is that, for each item, certain skills are needed for an examinee to answer the item correctly. The skills are assumed to be either present or absent, and so the skills are latent dichotomous variables that take on values of zero or one, typically denoted as α_{ik} for examinee i and skill k . The DINA model is a *conjunctive* model (Maris, 1999), in that it is assumed that an examinee must have *all* the required skills to answer an item correctly.

Let Y_{ij} be a binary variable that indicates whether the response of the i th examinee to the j th item is correct or incorrect (1 or 0), and let $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_K)$ denote the vector of K skills that are needed to solve the items. The Q-matrix consists of elements q_{jk} that specify which of the K skills are needed to solve the j th item. Thus, the Q-matrix elements consist of zeroes and ones, with a value of zero indicating that the k th skill is *not* needed, and a value of one indicating that the skill is needed. These ideas lead to the DINA model, where the probability that an examinee gets an item correct is

$$p(Y_{ij} = 1 | \alpha) = (1 - s_j)^{\eta_{ij}} g_j^{1 - \eta_{ij}},$$

with $\eta_{ij} = \prod_{k=1}^K \alpha_{ik}^{q_{jk}}$ (e.g., de la Torre, 2009; Henson et al., 2009; Junker & Sijtsma, 2001). The parameter s_j is the *slip* rate for examinee j , which is the probability that an examinee who has the requisite skills gets the item incorrect. The parameter g_j is the *guess* rate. It has previously been noted (e.g., DeCarlo, 2011; Junker & Sijtsma, 2001) that the DINA model is a type of signal detection model. In signal detection theory (SDT; Macmillan & Creelman, 2005), a “slip” is a *miss*: The examinee has the requisite skills, but he or she misses getting the item correct. A “guess” in SDT is related to the *false alarm* rate: The examinee does not have the requisite skills, but nevertheless gets the item correct (guessing is simply an interpretation of false alarms).

Substituting the second equation above into the first gives the DINA model,

$$p(Y_{ij} = 1 | \alpha) = (1 - s_j) \prod_{k=1}^K \alpha_{ik}^{q_{jk}} g_j^{1 - \prod_{k=1}^K \alpha_{ik}^{q_{jk}}}.$$

For the Bayesian generalization, uncertainty is recognized by allowing some of the elements of the Q-matrix (i.e., the q_{jk}) to be random rather than fixed, as done by J. Templin and Henson (2006) and discussed the following. The preceding form of the model, however, is somewhat complex (e.g., the q_{jk} are exponents of exponents). A simpler (but equivalent) form of the model can be obtained by reparameterizing it as follows:

$$\text{logit } p(Y_{ij} = 1 | \alpha) = f_j + d_j \prod_{k=1}^K \alpha_{ik}^{q_{jk}}. \quad (1)$$

Equation 1 is a reparameterization of the DINA model and has been referred to as the RDINA (reparameterized deterministic input noisy and) model (DeCarlo, 2011). The parameter f_j is the false alarm rate, as described earlier, whereas d_j is a *discrimination* (detection) parameter that indicates how well the item discriminates between the presence versus absence of the required skill set. The parameter d_j has an interpretation in SDT as a distance measure (e.g., DeCarlo, 2010; Macmillan & Creelman, 2005); a related discrimination parameter, although not a distance measure, has been discussed by Henson, Roussos, Douglas, and He (2008). Note that the DINA parameters are easily recovered from the RDINA parameters,

$$g_j = \frac{\exp(f_j)}{1 + \exp(f_j)},$$

where \exp is the exponential function, and

$$s_j = 1 - \frac{\exp(f_j + d_j)}{1 + \exp(f_j + d_j)}.$$

An attractive aspect of Equation 1 is that it is a simple logistic regression model with latent (multiplicative) predictors (i.e., the latent dichotomous α), and so it can easily be fit with software for latent class analysis, as previously shown (DeCarlo, 2011). In a similar vein, although the Bayesian generalization can be accomplished with either form of the model, the RDINA version is simple to implement, as shown in the Appendix.

Note that, when fitting the DINA or RDINA model, the fixed elements q_{jk} in Equation 1 are simply set to zero or one, according to the Q-matrix specification. For example, suppose that out of a set of four skills, Skills 1 and 3 are considered as necessary to solve the first item, but not Skills 2 or 4. The first row of the Q-matrix is then 1, 0, 1, 0, which means that the model for the first item is

$$\begin{aligned}\text{logit } p(Y_{i1} = 1 | \alpha_1, \alpha_2, \dots, \alpha_k) &= f_1 + d_1 \alpha_{i1}^1 \times \alpha_{i2}^0 \times \alpha_{i3}^1 \times \alpha_{i4}^0, \\ &= f_1 + d_1 \alpha_{i1} \alpha_{i3},\end{aligned}$$

which shows that the q_{jk} elements simply select which terms (α_{ik}) appear in the model. Note that the product term following d_1 is equal to one only if skills α_{i1} and α_{i3} are present; otherwise, it is equal to zero, which is the conjunctive aspect of the model. The Bayesian extension of Equation 1 differs in that some q_{jk} explicitly appear in the model as random parameters, rather than being set to zero or one.

Equation 1 is the “examinee-level” part of the model, which is concerned with how the examinees’ response patterns are related to the skill patterns. The full model includes a (higher order) model for the skill patterns:

$$p(Y_{i1}, Y_{i2}, \dots, Y_{iJ}) = \sum_{\alpha} p(\alpha_1, \alpha_2, \dots, \alpha_k) \prod_j p(Y_{ij} | \alpha_1, \alpha_2, \dots, \alpha_k), \quad (2)$$

where the first term on the right of the equal sign is the “skill-level” model and the second term is the examinee-level model. The skill-level model is concerned with relations among the skills, and the term $p(\alpha_1, \alpha_2, \dots, \alpha_k)$ is the probability of a particular skill pattern. A simple assumption is independence, in which case

$$p(\alpha_1, \alpha_2, \dots, \alpha_k) = \prod_k p(\alpha_k),$$

where $p(\alpha_k)$ are the probabilities of the K skills (latent class sizes in latent class analysis). Equation 2 also has a product term, which follows from an assumption of local independence of the indicators given the skills. Note that $p(Y_{ij} | \alpha_1, \alpha_2, \dots, \alpha_k)$ in Equation 2 can be obtained by applying the inverse of the logit function to Equation 1, where the inverse of the logit function, known as the *expit* function, is $\exp(a)/[1 + \exp(a)]$. This form of the model is used in the program given in the appendix.

In addition to the simple DINA and RDINA models with independence, various higher order models have also been considered in the literature on CDMs (e.g., de la Torre & Douglas, 2004; J. L. Templin, Henson, Templin, & Roussos, 2008); these models allow for structure in the skill-level part of the model, such as allowing for correlations among the skills. For example, the independence assumption can be replaced with a conditional independence assumption by including a latent continuous variable θ (e.g., examinee ability) in the skill-level part of the model as follows:

$$p(\alpha_1, \alpha_2, \dots, \alpha_k) = \int_{\theta} p(\alpha_1, \alpha_2, \dots, \alpha_k | \theta) p(\theta) d\theta.$$

The latent variable θ accounts for associations among the K skills, as in a factor analysis model. Replacing $p(\alpha_1, \alpha_2, \dots, \alpha_k)$ in Equation 2 with the preceding gives a higher order model for the skills, which has been referred to as the HO-DINA (higher order deterministic input noisy and) model (e.g., de la Torre & Douglas, 2004). In this case, it is assumed that the conditional probabilities of the skills, $p(\alpha|\theta)$, are independent conditional on θ :

$$p(\alpha_1, \alpha_2, \dots, \alpha_k|\theta) = \prod_k p(\alpha_k|\theta).$$

A simple logistic model has been used for the higher order model (e.g., DeCarlo, 2011; de la Torre & Douglas, 2004):

$$p(\alpha_k|\theta) = \frac{\exp(b_k + a_k\theta)}{1 + \exp(b_k + a_k\theta)}, \quad (3)$$

where b_k is an item difficulty parameter and a_k is an item discrimination parameter. Using the higher order model with Equations 1 and 2 gives a higher order version of the reparameterized version of the model (higher order reparameterized deterministic input noisy and [HO-RDINA]).

As previously shown, the RDINA and HO-RDINA models are simple to implement in software that can fit logistic models, or more generally, *generalized linear models* with latent predictors, such as LEM (Vermunt, 1997), Latent Gold (Vermunt & Magidson, 2005), and other software (e.g., GLLAMM; Rabe-Hesketh, Skrondal, & Pickles, 2004); for sample LEM and Latent Gold programs, see DeCarlo (2011). The software can also be used to consider various extensions of the models, such as allowing for more than two response categories (with nominal or ordinal responses) or allowing for skills with more than two categories (either nominal or ordinal). With respect to the (more complex) Bayesian extensions of the RDINA and HO-RDINA models considered here, the freely available software OpenBUGS (Thomas, O'Hara, Ligges, & Sturtz, 2006) can be used; the Appendix provides a sample program.

The Bayesian RDINA/HO-RDINA Models

Bayesian extensions of the above models allow some of the elements of the Q-matrix to be random rather than fixed. In particular, instead of fixing all the q_{jk} to zero or one, which is the usual approach to specifying the Q-matrix, some of the q_{jk} are treated as random variables, and in particular as Bernoulli variables with parameters p_{jk} . The choice of Bernoulli variables reflects the fact that the q_{jk} only take on values of zero or one. The current approach relaxes the assumption, for some elements of Q, that the q_{jk} are known. Posterior distributions can then be used to guide decisions about the q_{jk} in question.

The Bayesian version of the RDINA model can be specified in the same manner as earlier. To start, the Y_{ij} are conditionally independent and Bernoulli distributed:

$$Y_{ij} \sim \text{Bernoulli}(p_j),$$

where p_j is given by the inverse of Equation 1; that is,

$$p_j = p(Y_{ij} = 1 | \alpha) = \text{expit} \left(f_j + d_j \prod_{k=1}^K \alpha_{ik}^{q_{jk}} \right),$$

where the expit function is as defined above. In the Bayesian approach, the parameters f_j and d_j are specified as random. In addition, instead of being fixed, some of the elements of the Q-matrix are now treated as Bernoulli variables:

$$\tilde{q}_{jk} \sim \text{Bernoulli}(p_{jk}),$$

where the tilde (\sim) on q is used to indicate that the element is random (compare Lee, 2004); note that the use of tilde is avoided in the following when it is clear from the context that q is random. As before, elements of the Q-matrix that are assumed to be known are set to zero or one, according to the Q-matrix specification. For uncertain elements, however, \tilde{q}_{jk} is a Bernoulli variable with probability p_{jk} (note that p_j is the conditional response probability, whereas p_{jk} is the Q-element probability). A Beta prior, with hyperparameters α and β , is used for p_{jk} :

$$p_{jk} \sim \text{Beta}(\alpha, \beta).$$

For example, Beta(1,1) is the uniform distribution and is used as a prior below. As is well known in Bayesian analysis, the Beta distribution is a conjugate prior for the Bernoulli distribution, and so the posteriors of p_{jk} given \tilde{q}_{jk} are also Beta:

$$p_{jk} | \tilde{q}_{jk} \sim \text{Beta}(\alpha + \tilde{q}_{jk}, \beta + 1 - \tilde{q}_{jk}).$$

The posterior reflects the information that the data provide about whether \tilde{q}_{jk} is one or zero; it has a mean of

$$E(p_{jk} | \tilde{q}_{jk}) = \frac{\alpha + \tilde{q}_{jk}}{\alpha + \beta + 1}. \quad (4)$$

Equation 4 shows that, for a Beta(1,1) prior, the posterior mean for p_{jk} will be 1/3 if $\tilde{q}_{jk} = 0$ and 2/3 if $\tilde{q}_{jk} = 1$ (see the following results). The posterior density of \tilde{q}_{jk} or p_{jk} for the random elements of Q can be used to help determine whether a skill should be included in a particular item, that is, whether the “true” q_{jk} is zero or one. A decision can be implemented by simply rounding the posterior mean of \tilde{q}_{jk} , as done by J. Templin and Henson (2006) and below or, similarly, by using a cut point of 0.50 for the posterior mean of p_{jk} (i.e., q_{jk} is zero if $p_{jk} < .5$ and one otherwise).

Implementation in OpenBUGS

The Appendix provides a program that shows how to implement the Bayesian RDINA model using the freely available software OpenBUGS. The program includes uncertainty about the skills for Items 1, 8, and 13, as in the simulations below. The program shows that to include uncertain elements, the latent dichotomous variable α_k (denoted as x_k in the program) is raised to the power q_{jk} , where q_{jk} is a Bernoulli variable with parameter p_{jk} . Each uncertain element has its own probability, with a Beta(1,1) prior used for p_{jk} . In addition, “mildly” informative normal priors (mean of 0 and variance of 10; note that OpenBUGS uses the precision, which is the inverse of the variance) are used for f_j and d_j , and the bounds function, $I(0,)$, is used to restrict d to positive values (more informative priors could also be used, given that there is often information available about values of SDT parameters found in practice).

Note that, although the model could be fit as in Equation 1 by using a logit transform of the response probabilities, problems can arise with this approach when there are extreme probabilities, because the logits then go toward plus or minus infinity. This problem does not appear to

Table 1. True Q-Matrix and Bayesian Q-Matrix (Conditions 2 and 7) for Four Skills and Fifteen Items

Item No.	Q-matrix				Bayesian Q-matrix			
	α_1	α_2	α_3	α_4	α_1	α_2	α_3	α_4
1	1	0	0	0	q_{11}	q_{12}	q_{13}	q_{14}
2	0	1	0	0	0	1	0	0
3	0	0	1	0	0	0	1	0
4	0	0	0	1	0	0	0	1
5	1	1	0	0	1	1	0	0
6	1	0	1	0	1	0	1	0
7	1	0	0	1	1	0	0	1
8	0	1	1	0	q_{81}	q_{82}	q_{83}	q_{84}
9	0	1	0	1	0	1	0	1
10	0	0	1	1	0	0	1	1
11	1	1	1	0	1	1	1	0
12	1	1	0	1	1	1	0	1
13	1	0	1	1	q_{131}	q_{132}	q_{133}	q_{134}
14	0	1	1	1	0	1	1	1
15	1	1	1	1	1	1	1	1

Note: The Bayesian Q-matrix has complete uncertainty about the four skills for Items 1, 8, and 13.

arise if the probability version of the model (i.e., with the expit function) is used, as is done in the program in the Appendix.

Simulations: Uncertainty About Q-Matrix Elements

The utility of the Bayesian approach is examined in a number of small simulations. The basic RDINA model with independence is first examined in several conditions to determine if the Bayesian approach is of use in at least the simplest case. In the first simulation, there is uncertainty about 4 elements of the Q-matrix (for three items), with the remaining elements correctly specified. In the second simulation, there is complete uncertainty about the skills for three items, giving a total of 12 uncertain elements (20% of the 60 Q-matrix elements), as shown by the Bayesian Q-matrix in Table 1. The third simulation introduces a new aspect, in that there is again uncertainty about 12 elements, as in the second simulation; however, 6 other elements of the Q-matrix are now incorrectly specified in the fitted model, as shown in Table 2. Thus, the third simulation examines the effect of misspecification of other parts of the Q-matrix (i.e., parts that are not under question) on recovery of elements that are considered uncertain. One could argue that this is a realistic situation, in that it seems likely that in practice, some elements of the Q-matrix might be misspecified without the researcher's knowledge (i.e., the elements are not recognized as being uncertain).

The preceding conditions examine the utility of the Bayesian approach for determining uncertain elements in the Q-matrix, with the number of skills correctly specified. The next three conditions provide some information about a different question, which is whether the correct number of skills has been specified. In the fourth condition, the same data used in the first three simulations were used, and the fitted model used the correct Q-matrix for the first three skills; however, the fourth skill (which was necessary) was specified as being completely uncertain for all 15 items. In this situation, one is not sure if the fourth skill should be included, but in fact it is necessary. The fifth condition is the same as the fourth condition, except that five other elements of the Q-matrix were misspecified, as shown in Table 2 (although q_{74} was now

Table 2. Q-matrix With Complete Uncertainty for Three Items and Six Misspecified Elements (Conditions 3 and 8)

Item No.	Bayesian Q-matrix with misspecification			
	α_1	α_2	α_3	α_4
1	q_{11}	q_{12}	q_{13}	q_{14}
2	0	1	0 \rightarrow 1	0
3	0	0	1	0
4	0	0 \rightarrow 1	0	1
5	1	1	0	0
6	1	0	1	0
7	1	0	0	1 \rightarrow 0
8	q_{81}	q_{82}	q_{83}	q_{84}
9	0 \rightarrow 1	1	0	1
10	0	0	1	1
11	1	1	1 \rightarrow 0	0
12	1	1	0	1
13	q_{131}	q_{132}	q_{133}	q_{134}
14	0	1 \rightarrow 0	1	1
15	1	1	1	1

Note: Misspecified elements are shown with arrows, with original value followed by the misspecified value. For example, Skill 3 (α_3) for Item 2 is misspecified as 1, whereas it is 0 in the true Q-matrix shown in Table 1.

uncertain instead of misspecified). In the sixth condition, the Q-matrix was correctly specified for the first four skills, but a fifth (nonrequired) skill, with complete uncertainty about its elements, was introduced. In this situation, one is again not sure whether an additional skill is needed, but in this case it should not be included. The fourth through sixth conditions provide information about the use of the Bayesian approach in situations where there is uncertainty as to the number of skills.

Finally, the seventh and eighth conditions examined the utility of the approach in situations where the skills have a higher order structure. In particular, data were generated according to the HO-RDINA model given earlier. In the seventh condition, the fitted model had 12 uncertain elements, as in the second condition stated earlier, with the difference that the generated data had a higher order structure. In the eighth condition, the fitted model again had 12 uncertain elements and, in addition, 6 misspecified elements, as in the third condition stated earlier. The simulations provide information about a range of basic situations of interest.

Method

The simulations used a Q-matrix for 15 items that was previously used by Rupp and Templin (2008a) and DeCarlo (2011); the matrix is shown in Table 1. For the generated data, the population values chosen for the detection parameters d_j and the false alarm rates f_j were similar to estimates obtained for real data in previous research. For example, an analysis of Tatsuoka's fraction-subtraction data with RDINA (DeCarlo, 2011) gave estimates of d_j for 21 items that ranged from 1.7 to 6.9 with a mean of 4.3, and estimates of f_j that ranged from -4.8 to -0.1 with a mean of -2.3 . For the current simulation, values of d_j from 1.5 to 5.5 and f_j from -4 to 0 were used; the skill class sizes were 0.38, 0.50, 0.62, and 0.73, and the sample size was 1,000. The purpose here was not to examine parameter recovery (which appeared to be good), but

rather to explore whether the Bayesian approach helps to recover the true Q-matrix elements when there is uncertainty about them.

The Bayesian RDINA and HO-RDINA models were fitted using OpenBUGS. For each condition, the complete analysis was repeated for the 20 generated data sets. For each run, the data were loaded into OpenBUGS, and the syntax program (see the appendix) was then run with 5,000 burn-ins followed by 20,000 iterations for posterior sampling. With this number of iterations, the Monte Carlo errors (see Geyer, 1992), which are given in the OpenBUGS output, were generally less than 5% of the sample standard deviations (also given in the output), which has previously been suggested as a rule of thumb to assess convergence (Spiegelhalter, Thomas, Best, & Lunn, 2003). Multiple chains also appeared to converge.

In the first condition, four elements in the fitted RDINA model were specified as being uncertain; specifically, elements q_{13} , q_{14} , q_{82} , and q_{131} for Items 1, 8, and 13 were treated as Bernoulli variables. In the second condition, there was complete uncertainty about all the skills for 3 items (a kind of worst-case scenario), as shown by the Q-matrix on the right side of Table 1 with 12 uncertain elements; note that this would require an examination of 4,096 Q-matrices in an exhaustive search approach. In the third condition, there were again 12 uncertain elements in the Q-matrix, but 6 other elements were incorrectly specified in the fitted model, and in particular, three skills were incorrectly included and three skills were incorrectly excluded, as shown by the Q-matrix in Table 2.

The next three conditions examined situations where there was complete uncertainty about a skill. In the fourth condition, the fitted model used the correct Q-matrix (shown on the left side of Table 1); however, all elements for the fourth skill were treated as being uncertain, that is, the q_{j4} were random for all 15 items. This examines a situation where there is uncertainty about whether the skill is needed, when in fact it is needed. The fifth condition was the same except that five other elements of the Q-matrix were misspecified, as shown in Table 2 (with the difference that q_{74} was specified as uncertain rather than misspecified). For the sixth condition, the fitted model used the Q-matrix for the four skills, as shown in Table 1, but a fifth skill was also included, with all elements treated as uncertain, and so the q_{j5} were uncertain for all 15 items. This examines a situation where there is uncertainty about whether an additional skill should be included, when in fact it should not be included.

Finally, in the seventh and eighth conditions, data with a simple higher order structure for the skills were generated. These conditions provide information about Q-element recovery in the presence of a higher order skill structure. In particular, data for the HO-RDINA model were generated, with Equation 3 used to specify the higher order structure; the Q-matrix was the same as that shown in Table 1. The population values for the higher order parameters were suggested by results for an application of the HO-RDINA model (and restricted versions) to the fraction-subtraction data (DeCarlo, 2011), which found estimates of a_k that ranged from 0.7 to 4.0 (and 3.5 for a version with equal a_k) and estimates of b_k that ranged from -1 to 4. For the simulation, a common value of $a_k = 3$ was used with values of b_1 to b_4 of -1 , 0, 1, and 2, respectively (which give skill class sizes similar to those used for the RDINA simulation above). In Condition 7, the fitted HO-RDINA model (with equal a_k) had 12 elements of the Q-matrix specified as being uncertain, as in Condition 2. In Condition 8, the fitted model had 12 uncertain Q-elements, and 6 misspecified elements, as in Condition 3.

Results

Condition 1: Four uncertain Q-elements. The main tools used here are means and plots of the posterior distributions of q_{jk} and p_{jk} for the uncertain elements. For example, for the first generated data set, Figure 1 shows plots from OpenBUGS of the posterior distributions of q_{jk} and p_{jk} .

Table 3. Posterior Means of Q-Matrix Elements and Probabilities for 20 Replications (Condition 1)

Data set	q_{13}	q_{14}	q_{82}	q_{131}	p_{13}	p_{14}	p_{82}	p_{131}
1	0.227	0.269	1.000	1.000	0.407	0.423	0.666	0.664
2	0.321	0.323	1.000	1.000	0.442	0.442	0.666	0.666
3	0.001	0.013	1.000	1.000	0.336	0.338	0.667	0.667
4	0.218	0.708	1.000	1.000	0.405	0.570	0.666	0.666
5	0.006	0.508	1.000	1.000	0.335	0.506	0.665	0.665
6	0.032	0.322	1.000	1.000	0.343	0.441	0.667	0.666
7	0.001	0.028	1.000	1.000	0.337	0.343	0.668	0.668
8	0.638	0.174	1.000	1.000	0.544	0.390	0.666	0.666
9	0.183	0.751	1.000	1.000	0.392	0.585	0.666	0.668
10	0.035	0.101	1.000	1.000	0.345	0.366	0.668	0.668
11	0.058	0.209	1.000	1.000	0.353	0.404	0.667	0.669
12	0.082	0.223	1.000	1.000	0.363	0.407	0.668	0.668
13	0.334	0.329	1.000	1.000	0.443	0.444	0.666	0.668
14	0.039	0.571	1.000	1.000	0.346	0.525	0.672	0.665
15	0.137	0.060	1.000	1.000	0.378	0.354	0.666	0.664
16	0.505	0.492	1.000	1.000	0.503	0.495	0.664	0.668
17	0.154	0.007	1.000	1.000	0.386	0.334	0.668	0.664
18	0.160	0.430	1.000	1.000	0.387	0.477	0.667	0.668
19	0.090	0.250	1.000	1.000	0.364	0.416	0.667	0.666
20	0.652	0.727	1.000	1.000	0.552	0.575	0.666	0.667

Note: The true Q-values are 0, 0, 1, and 1. Posteriors that give incorrect q s are shown in bold.

The true Q-matrix elements in this case are 0, 0, 1, 1, and it is apparent that a similar pattern appears in the plots. For example, the first row of Figure 1 shows that the posteriors for \tilde{q}_{jk} are fractional and suggest values of zero. The second row shows that the posteriors of \tilde{q}_{jk} are both simply one. Thus, the posteriors clearly suggest the pattern 0, 0, 1, 1, which is correct. The same holds for the posterior distributions of p_{jk} , shown in the third and fourth rows of Figure 1, which tend toward “left-edge” triangles when the true q_{jk} is zero (third row) and “right-edge” triangles when the true q_{jk} is one (fourth row). Note that a left- or right-edge triangle shape occurs when the posteriors are Beta(1,2) or Beta(2,1), respectively, which gives posterior means of 0.33 and 0.67, as shown in Equation 4. The data in this case provide a single observation as to whether \tilde{q}_{jk} is 0 or 1. Note that, although the fourth row shows sharp triangles, which reflect the posterior means of 0.67 found for p_{jk} , the third row does not, but rather shows “left-leaning” plots with posterior means of p_{jk} of 0.41 and 0.42 (see Table 3), and so there is fractional evidence that the \tilde{q}_{jk} are zero, exactly as shown by the posterior means of \tilde{q}_{jk} in the first row.

Table 3 shows, for the first condition, the posterior means for the uncertain Q-matrix elements for all 20 data sets. The true q_{jk} elements are 0, 0, 1, and 1. The third and fourth columns of Table 3 show that the posterior means of \tilde{q}_{82} and \tilde{q}_{131} are 1.0 for all 20 data sets, and so the recovery rate is 100% for these elements. The first and second columns show that the posterior means of \tilde{q}_{13} and \tilde{q}_{14} generally indicate values of zero, with the exception of three cases for \tilde{q}_{13} , shown in bold, and five cases for \tilde{q}_{14} . It is also apparent from Table 3 that the same decisions are made regardless of whether the posterior means of \tilde{q}_{jk} or p_{jk} are used.

As noted above, a simple procedure is to round the Q-matrix element posterior mean (of \tilde{q}_{jk}) to zero or one. Table 4 shows, for the first three conditions, the results of this procedure. The first matrix in Table 4 (top left) shows the true Q-matrix elements for the 3 items. The next matrix (top right) shows, for the first condition, the percentage of cases (out of 20) where the Q-element was correctly recovered. Recovery is 100% (that the skill should be included) for

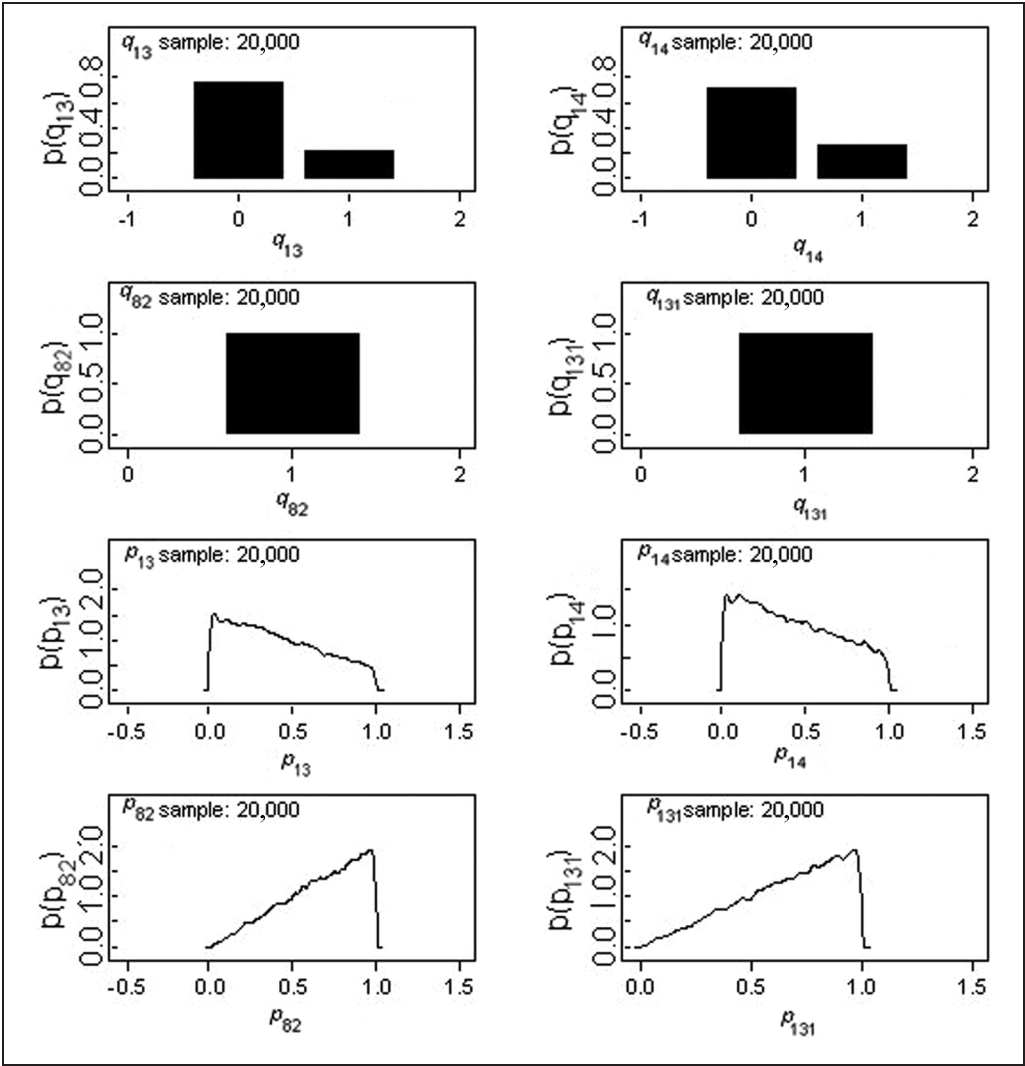


Figure 1. The top four plots are the posterior distributions for \tilde{q}_{jk} for the first generated data set in the first condition, with a plot for each of the four uncertain elements. The second four plots are the posterior distributions for p_{jk} .

Items 8 and 13 (Skills 2 and 1, respectively), 85% (that the skill should not be included) for the third skill of Item 1, and 75% (that the item should not be included) for the fourth skill of Item 1. Thus, using the mean of the posterior density to set q_{jk} to zero or one generally led to a correct decision. The table also shows that the detection of 1s in the Q-matrix (skill needed) appears to be better than the detection of 0s (skill not needed), as also shown in Table 3.

Condition 2: Twelve uncertain Q-matrix elements. In the second condition, there was complete uncertainty for all four skills for Items 1, 8, and 13, giving a total of 12 uncertain elements. Using the approach noted above (i.e., rounding the posterior mean of q_{jk}), the matrix at the bottom left of Table 4 shows that the four skills for Item 8 are correctly detected as being zero or one in 100% of the cases, whereas for Item 13, detection is correct for 90% to 100% of the

Table 4. Percent-Correct Recovery of Q-Matrix Elements for Conditions 1, 2, and 3

RDINA Item No.	True Q-matrix				Q-matrix (4 uncertain)			
	α_1	α_2	α_3	α_4	α_1	α_2	α_3	α_4
1	1	0	0	0	—	—	85	75
8	0	1	1	0	—	100	—	—
13	1	0	1	1	100	—	—	—

RDINA Item No.	Q-matrix (12 uncertain)				Misspecified Q-matrix			
	α_1	α_2	α_3	α_4	α_1	α_2	α_3	α_4
1	100	85	85	75	100	85	90	60
8	100	100	100	100	65	100	100	80
13	95	100	90	95	95	100	85	85

Note: RDINA = reparameterized deterministic input noisy and. Except for the true Q-matrix, entries indicate the percent-correct recovery of the element for 20 data sets. The misspecified Q-matrix had 12 uncertain elements and 6 misspecified elements.

cases. For Item 1, the need to include the first skill is detected correctly in 75% to 100% of the cases. Thus, although it was expected that the inclusion of a greater number of uncertain skills would lead to less accurate detection, the results in Table 4 suggest generally good detection for a situation with 12 uncertain skills.

Condition 3: Twelve uncertain Q-matrix elements and six misspecifications. In the above conditions, the specification of the Q-matrix was uncertain for some of the elements, but the other Q-elements were correctly specified. In Condition 3, there was again uncertainty about 12 Q-matrix elements, as in Condition 2; however, six other elements of the Q-matrix were incorrectly specified. As shown in Table 2, three elements were incorrectly excluded and three elements were incorrectly included. It is of interest to see how this affects recovery of the uncertain Q-matrix elements in the Bayesian approach.

The posterior means were again used to determine the Q-matrix elements. The lower right of Table 4 shows recovery of the uncertain elements. Recovery is somewhat better than expected, still being around 80% or higher for 10 of the 12 elements (and 100% for 4 elements); however, recovery is poor for two elements. Notably, it is 60% and 65% for elements q_{14} and q_{81} (both with true values of zero). This shows that misspecification of other elements of the Q-matrix can affect recovery rates for some of the uncertain elements.

Conditions 4, 5, and 6: Complete uncertainty about a skill. Table 5 shows recovery of the Q-elements for the fourth, fifth, and sixth conditions. The third column shows the percent-correct recovery for the fourth condition, where the fourth skill was necessary but had all 15 elements specified as being uncertain. The table shows that recovery is generally excellent, with 90% to 100% correct recovery for 14 of the 15 elements, whereas one element had only 75% recovery. The second column of Table 5 shows results for the fifth condition, which again had complete uncertainty about the fourth skill, but five other elements of the Q-matrix were also misspecified. In this case, recovery is quite poor for many elements. This shows that, when there is complete uncertainty about a skill, misspecification of other elements of the Q-matrix can have a profound effect on the recovery rates. In the sixth condition, a fifth skill with complete uncertainty was included. Given that the skill is not necessary, one might expect to find posteriors for q_{jk} that are all close to zero, suggesting that the skill should not be included; however, the posteriors instead tended to be around 0.5, and the resulting recovery rates were

Table 5. Recovery of Q-Matrix Elements for Conditions 4, 5, and 6

Item No.	Condition 4		Condition 5		Condition 6	
	True q_{j4}	% correct	True q_{j4}	% correct	True q_{j5}	% correct
1	0	75	0	80	0	40
2	0	100	0	100	0	65
3	0	95	0	25	0	70
4	1	100	1	45	0	60
5	0	100	0	90	0	55
6	0	100	0	15	0	35
7	1	90	1	25	0	40
8	0	100	0	25	0	65
9	0	90	0	20	0	60
10	1	100	1	75	0	45
11	0	100	0	15	0	45
12	1	100	1	30	0	50
13	1	95	1	100	0	40
14	1	95	1	90	0	40
15	1	100	1	95	0	40

Note: Entries indicate the proportion of times the element was correctly recovered for 20 data sets. For Condition 4, the fourth skill was specified as uncertain for all 15 items. Condition 5 was the same as Condition 4, except that five other elements were also misspecified. For Condition 6, the first four skills were correctly specified; however, a fifth skill was (incorrectly) included, with each element specified as uncertain.

Table 6. Recovery of Q-Matrix Elements for a HO-RDINA Model, Conditions 7 and 8

HO-RDINA Item No.	Q-matrix (12 uncertain)				Misspecified Q-matrix			
	α_1	α_2	α_3	α_4	α_1	α_2	α_3	α_4
1	90	80	55	55	90	90	60	55
8	100	100	100	100	100	100	100	90
13	95	100	80	70	100	100	60	60

Note: HO-RDINA = higher order reparameterized deterministic input noisy and.

uniformly poor (i.e., around 50%), as shown in Table 5. An interesting result is that the posterior means (not shown) of the class size for the nonnecessary Skill 5 tended to be large (i.e., $>.90$), which is consistent with an earlier conjecture that including a nonnecessary skill can lead to a class size close to one (DeCarlo, 2011).

Conditions 7 and 8: HO-RDINA. Data in these conditions (20 data sets) were generated according to the HO-RDINA model as described above. In Condition 7, the Bayesian HO-RDINA model (with a common value of a_k in Equation 3) was fit with 12 uncertain elements, as shown by the Bayesian Q-matrix in Table 1. The left side of Table 6 shows that recovery of many elements is quite good, with the same pattern as shown in the lower left part of Table 4 (Condition 2 for the RDINA model). A difference, however, is that some elements are poorly recovered, most notably elements q_{13} and q_{14} for Item 1, which are correctly detected as being zero only 55% of the time (and so the false alarm rate is quite high at 45%). The recovery rates for elements q_{133} and q_{134} are also lower compared with the results shown in the lower left part of Table 4 (Condition 2). In Condition 8, there were again 12 uncertain elements for the higher order model, but six other elements were misspecified (as in Condition 3). The right side of Table 6 shows that recovery was excellent for many elements, but was again poor for Skills 3

and 4 for Item 1 (where the skills are not necessary), and was also poor for Skills 3 and 4 for Item 13 (where the skills are necessary). Thus, as for the model without a higher order structure (Condition 3), the results for Condition 8 suggest that misspecification of other parts of the Q-matrix affects recovery rates for some of the uncertain elements. In addition, recovery rates for data with a higher order structure appear to be lower for some elements than those for data with an independence structure.

Discussion

The simulations show that the posterior distributions for the random Q-matrix elements provide useful information about which elements should or should not be included (i.e., whether \tilde{q}_{jk} should be zero or one). The approach generally led to correct recovery of elements in situations where there was uncertainty about 4 (7%) or 12 (20%) elements of the Q-matrix (Conditions 1 and 2), with the rest of the matrix correctly specified. When in addition, six elements of the Q-matrix were incorrectly specified (Condition 3), recovery of 12 uncertain elements was still generally quite good; however, recovery for some elements was apparently degraded. These results also appeared for data with a higher order structure (Conditions 7 and 8); recovery rates for some elements also appeared to be lower. These results differ somewhat from those reported by J. Templin and Henson (2006), who examined the DINA model with a higher order structure (using tetrachoric correlations for the skill patterns), in that they reported 100% recovery in three conditions where the percentage of uncertain elements was varied. The present results show excellent recovery in many cases, but not always 100%, and in fact recovery was quite poor for some elements in some conditions, as discussed earlier. A new finding, shown here, is that recovery rates can be adversely affected for some uncertain elements when other elements of the Q-matrix are not correctly specified.

With respect to situations that involve uncertainty about the number of skills, the Bayesian approach appears to be of some, but limited, utility. The results for Condition 4, for example, showed that when a skill was necessary, the Bayesian approach worked very well with respect to indicating which items the skill should load on. However, when (five) other elements of the Q-matrix were misspecified (Condition 5), the approach started to break down and recovery rates were poor for many elements. Furthermore, when a nonnecessary skill was introduced (Condition 6), the posteriors for the Q-elements were not zero, but tended to vary around 0.5, and so the recovery rates were poor. The results suggest that the Bayesian approach might be of limited utility with respect to determining when to stop adding skills: It correctly reveals the skill loadings when the rest of the Q-matrix is correctly specified, but less so when other elements are misspecified; it also does not clearly indicate when a skill is *not* needed.

Overall, the results are encouraging and suggest that it is worthwhile to examine the Bayesian approach in more extensive simulations, and to apply it to real-world data.

An Application: The Fraction-Subtraction Data

The Bayesian approach to Q-matrix determination is applied to the widely analyzed fraction-subtraction data of K. K. Tatsuoka (1990), which consists of 536 examinees. The Q-matrix, which in this case is for 15 of the items, is the same as that used by de la Torre (2008) in a prior study that was concerned with validity of the Q-matrix; the Q-matrix was adapted from Mislevy (1996) and is shown in Table 7. The labels given to the five skills are (a) performing basic fraction-subtraction operation, (b) simplifying/reducing, (c) separating whole numbers from fractions, (d) borrowing one from whole number to fraction, and (e) converting whole numbers to fractions.

Table 7. Fraction-Subtraction Data, 15 Items, Five Hypothesized Skills

No.	Item	Original Q-matrix					Bayesian Q-matrix				
		α_1	α_2	α_3	α_4	α_5	α_1	α_2	α_3	α_4	α_5
1	$\frac{3}{4} - \frac{3}{8}$	1	0	0	0	0	1	0	0	0	0
2	$3\frac{1}{2} - 2\frac{3}{2}$	1	1	1	1	0	1	1	1	1	0
3	$\frac{6}{7} - \frac{4}{7}$	1	0	0	0	0	1	0	0	0	0
4	$3 - 2\frac{1}{5}$	1	1	1	1	1	1	q_{42}	1	q_{44}	q_{45}
5	$3\frac{7}{8} - 2$	1	0	1	0	0	1	q_{52}	1	q_{54}	q_{55}
6	$4\frac{4}{12} - 2\frac{7}{12}$	1	1	1	1	0	1	1	1	1	0
7	$4\frac{1}{3} - 2\frac{4}{3}$	1	1	1	1	0	1	1	1	1	0
8	$\frac{11}{8} - \frac{1}{8}$	1	1	0	0	0	1	1	0	0	0
9	$3\frac{4}{5} - 3\frac{2}{5}$	1	0	1	0	0	1	0	1	0	0
10	$2 - \frac{1}{3}$	1	0	1	1	1	1	q_{102}	1	q_{104}	q_{105}
11	$4\frac{5}{7} - 1\frac{4}{7}$	1	0	1	0	0	1	0	1	0	0
12	$7\frac{3}{5} - \frac{4}{5}$	1	0	1	1	0	1	0	1	1	0
13	$4\frac{1}{10} - 2\frac{8}{10}$	1	1	1	1	0	1	1	1	1	0
14	$4 - 1\frac{4}{3}$	1	1	1	1	1	1	q_{142}	1	q_{144}	q_{145}
15	$4\frac{1}{3} - 1\frac{5}{3}$	1	1	1	1	0	1	1	1	1	0

Note: Skill labels are (a) performing basic fraction-subtraction operation, (b) simplifying/reducing, (c) separating whole numbers from fractions, (d) borrowing one from whole number to fraction, and (e) converting whole numbers to fractions.

Prior analyses of these data (e.g., DeCarlo, 2011; de la Torre, 2008) suggested that there is some uncertainty associated with items that involve whole numbers (without fractions), which consists of 4 items in Table 7 (Items 4, 5, 10, and 14) and so these 4 items are examined more closely here. All 4 items include Skills 1 and 3; however, they differ with respect to which of the other three skills (Skills 2, 4, and 5) are included, if any. Thus, the Q-matrix considered here has uncertainty with respect to whether Skills 2, 4, and 5 should be included in Items 4, 5, 10, and 14, which gives 12 uncertain elements. The Bayesian Q-matrix is shown on the right side of Table 7.

Note that one can make substantive arguments for the possible inclusion or exclusion of different skills in these (and other) items because of ambiguities with respect to solving items that involve whole numbers. For example, the first item with a whole number, Item 4, is $3 - 2\frac{1}{5}$, and the original Q-matrix considers all five skills as being involved in this item. There are, however, several different ways to solve this item. For example, one could simply convert both terms to fractions with common denominators and then subtract: $15/5 - 11/5 = 4/5$. Or one could borrow from the first whole number, convert to a fraction, and then subtract parts: $3 - 2\frac{1}{5} = 2\frac{5}{5} - 2\frac{1}{5}$, and so $2 - 2 = 0$ and $5/5 - 1/5 = 4/5$. Or one could first subtract the whole numbers, $3 - 2 = 1$, and then subtract the remaining fraction, $1/5$, from the converted remainder, $1 - 1/5 = 5/5 - 1/5 = 4/5$, and so on. The point is that the set of skills required for this simple item are questionable, and in general there are several approaches to solving the items that involve different combinations of skills. Of course, this is undesirable—ideally, a test for cognitive diagnosis should use items that clearly require certain skills and not others, without ambiguities, but that is easier said than done. In any case, the Bayesian approach recognizes that there is uncertainty about some elements of the Q-matrix and allows one in essence to simultaneously consider a large number of possible respecifications.

Table 8. Posterior Means of Q-Matrix Elements for the Fraction-Subtraction Data

Item	q_{j2}	q_{j4}	q_{j5}
4	0.005	1.000	1.000
5	0.980	0.000	1.000
10	0.013	1.000	1.000
14	0.999	1.000	1.000

Table 9. Information Criteria for Various Models: Fraction-Subtraction Data, $N = 536$

Model	No. of parameters	BIC	AIC	
RDINA	35	7,163.0	7,013.1	
RDINA from Bayesian analysis	35	7,120.5	6,970.5	
Effect of including an irrelevant attribute				
Model	No. of parameters	BIC	AIC	$\hat{g} + \hat{s}$
RDINA	35	7,163.0	7,013.1	.2461
RDINA (α_5 in Item 1)	35	7,188.0	7,038.1	.2379

Note: BIC = Bayesian information criterion; AIC = Akaike information criterion; RDINA = reparameterized deterministic input noisy and.

Results

OpenBUGS was again used with 5,000 burn-ins and 20,000 iterations. Table 8 shows the posterior means of q_{jk} for fits of the model with 12 uncertain elements. In this case, all the posterior means are close to zero or one. Note, for example, that the third and fourth rows of Table 8 suggest patterns of 0, 1, 1 and 1, 1, 1, respectively, which matches the original Q-matrix, and so the results suggest that Items 10 and 14 are correctly specified. However, the first and second rows suggest three changes to the Q-matrix. In particular, the table suggests that Skills α_2 and α_5 should be included in Item 5, and Skill α_2 should be dropped from Item 4.¹ Note that results of this sort could possibly be useful to substantive experts. For example, of the (at least) three strategies for solving Item 4 noted earlier, the Bayesian results (Skill 2 is not necessary) suggest that the third strategy might be the one used ($3 - 2 = 1$, $1 = 5/5$, $5/5 - 1/5 = 4/5$) because it involves the other skills but not Skill 2 (simplify/reduce).

To explore modifications of the Q-matrix suggested by the Bayesian analysis, Table 9 shows results for a fit of the original model and the model with a modified Q-matrix; in both cases, Latent Gold was used (using posterior mode estimation with Bayes's constants of 1; see DeCarlo, 2011). The table shows that BIC and AIC are smaller for the modified model (by more than 40), which indicates better relative fit compared with the original model. Table 10 shows parameter estimates for the original and modified model. For the modified Items 4 and 5, the discrimination parameters are slightly larger in magnitude than for the original model (whereas the false alarm rates are slightly lower in one case and higher in the other). Table 10 also shows that estimates of the skill probabilities (the latent class sizes) for the respecified Q-matrix are similar to those obtained for the original Q-matrix. The table also shows that the standard errors are large for the first item.

Another study of the validity of the Q-matrix that used the fraction-subtraction data and the same Q-matrix as used here was presented by de la Torre (2008). A sequential search algorithm was used along with a fit statistic, δ , that is basically the sum of the average slip and guess rates

Table 10. Parameter Estimates (SEs in Parenthesis) for the RDINA Model, Fraction-Subtraction Data

Item	Original Q-matrix					Modified Q-matrix				
	f_j		d_j			f_j		d_j		
1	−4.97 (1.91)		5.96 (1.91)			−4.90 (1.92)		5.92 (1.92)		
2	−1.31 (0.15)		3.33 (0.25)			−1.31 (0.15)		3.31 (0.25)		
3	−1.58 (0.34)		4.84 (0.43)			−1.47 (0.33)		4.76 (0.43)		
4	−1.93 (0.17)		3.83 (0.31)			−2.18 (0.21)		3.99 (0.32)		
5	−0.62 (0.19)		1.75 (0.23)			−0.49 (0.16)		2.33 (0.26)		
6	−3.35 (0.35)		4.59 (0.38)			−3.37 (0.35)		4.60 (0.38)		
7	−2.53 (0.24)		5.03 (0.35)			−2.53 (0.24)		5.01 (0.35)		
8	−1.70 (0.32)		4.07 (0.40)			−1.49 (0.32)		3.98 (0.38)		
9	−2.08 (0.38)		4.91 (0.45)			−1.96 (0.35)		4.80 (0.43)		
10	−1.62 (0.17)		4.17 (0.35)			−1.80 (0.19)		4.39 (0.36)		
11	−1.92 (0.32)		4.14 (0.37)			−1.87 (0.31)		4.13 (0.36)		
12	−3.45 (0.41)		5.30 (0.45)			−3.50 (0.43)		5.20 (0.46)		
13	−1.85 (0.18)		3.53 (0.25)			−1.86 (0.18)		3.53 (0.25)		
14	−3.75 (0.44)		5.16 (0.48)			−3.65 (0.41)		4.92 (0.45)		
15	−4.34 (0.57)		5.85 (0.60)			−4.44 (0.60)		5.95 (0.62)		
p_1	p_2	p_3	p_4	p_5		p_1	p_2	p_3	p_4	p_5
.79	.96	.93	.66	.78		.78	.94	.93	.68	.80
(0.02)	(0.02)	(0.02)	(0.03)	(0.03)		(0.02)	(0.02)	(0.02)	(0.03)	(0.03)

Note: RDINA = reparameterized deterministic input noisy and. p_1 to p_5 are the skill probabilities (latent class sizes) for the five skills.

(the approach was referred to as the “ δ -method”). Based on this analysis, de la Torre concluded that the original Q-matrix was supported. In contrast, the Bayesian approach used here suggested a modified Q-matrix that led to better relative fit, which suggests that the sequential search approach missed an improvement in the Q-matrix that was detected by the Bayesian approach.

A limitation of the δ -method was illustrated by de la Torre (2008), who included an irrelevant skill for one item (Skill 5 was included in Item 1). It was shown that this led to a smaller value of the proposed fit statistic (indicating better fit), although the inclusion was not theoretically plausible. However, using the Bayesian approach leads to the correct decision—the Bayesian RDINA model was fitted with uncertainty about Skill 5 in Item 1 and the posterior mean for q_{15} was close to zero, which (correctly) indicates that the attribute should not be

included. The lower part of Table 9 shows, for the model with an irrelevant skill, the δ fit statistics reported by de la Torre as well as information criteria. The table shows that BIC and AIC are larger for the incorrect modification, and so they correctly detect that the modification leads to relatively worse fit. Thus, the example illustrates a limitation of the δ -method, but supports the Bayesian approach, as well as the use of conventional information criteria, both of which pick up the error.

Summary and Conclusion

An approach to recognizing uncertainty in the Q-matrix is presented. Instead of specifying all the elements of the Q-matrix as being known (i.e., as zero or one), some elements are specified as being random (Bernoulli) variables. The probability parameters for the random elements are given a prior, such as the Beta distribution. The posterior of the element q , or its probability p , can then be used to help determine whether a Q-matrix element should be set to zero or one.

The utility of the approach was examined in several simulations. In the first two conditions, there was uncertainty about some elements of the Q-matrix (either 4 or 12 elements out of 60), with the rest of the elements correctly specified. In the third condition, a novel aspect was introduced in that there was again uncertainty about 12 elements of the Q-matrix, but 6 other elements were incorrectly specified. Conditions with complete uncertainty about a skill (i.e., for all items) were also examined, as well as data with a higher order structure. The results show that the Bayesian approach generally appears to be useful for helping to determine which skills should be included or excluded for each item. The best situation is when the skills are correctly specified for most or all of the Q-matrix, and there are questions about only some skills for some items. This situation appears to be somewhat robust to misspecifications of other parts of the Q-matrix, although misspecification leads to lower recovery rates for some elements. Similar results were found for data with a higher order structure, except that recovery rates for some elements were lower, particularly when other elements of the Q-matrix were misspecified. The approach appears to break down to some degree, however, when there is complete uncertainty about a skill, with other elements of the Q-matrix misspecified (Condition 5), or when an unnecessary skill is included (Condition 6). In short, the Bayesian approach appears to be useful for detecting which skills the items should load on when the Q-matrix and number of skills are at least generally correctly specified. Of course, the importance of theory in the development and respecification of the Q-matrix cannot be overemphasized.

There are also some limitations as well as directions for future research. For example, the present studies provide some basic groundwork, but the results are, of course, limited to the particular Q-matrices and parameter values examined. Simulations with other types of Q-matrices, a greater number of attributes, and other higher order structures need to be conducted. The use of more informative priors and a greater number of iterations should also be examined, particularly in the problematical situations found above. It might also be possible to use the approach in a more exploratory manner, for example, by considering most or all of the Q-matrix elements as being uncertain. However, the results found for Conditions 5 and 6, where there was complete uncertainty about a skill, raise some doubts about using the approach in this manner, although this remains to be explored.

Effects of using cut points in different ways for the random elements can also be examined. For example, a reviewer suggested a sequential version of the Bayesian approach, where after a first run, only elements with posterior means above or below certain cut points (e.g., below 0.2 or above 0.8) are set to zero or one. A second run can then be used with these elements fixed to see if this helps to set the remaining uncertain elements.

The results also have important practical implications. Given that CDMs are being more widely used in practice, the Bayesian approach offers a way for researchers to consider theory-guided modifications of their particular Q-matrix, without having to consider the dozens or hundreds of alternatives that arise when more than four elements are in question. The approach is simple to implement, and computational time is fairly short, which should encourage researchers to apply the approach in practical applications of CDMs.

Finally, it should be noted that although sophisticated models are helpful, there is also a need to collect data to *experimentally test* the Q-matrix under consideration. For example, for the fraction-subtraction task, one could instruct examinees as to which strategy to use (e.g., “solve all problems by converting to a common denominator”). Convincing evidence of the validity of a Q-matrix would be obtained if it could be shown that variations in the instructions (within or across examinees) lead to detectable differences in the Q-matrix. Although the fraction-subtraction data have been analyzed for decades, there has yet to be a basic demonstration of this sort, at least to the author’s knowledge.

Appendix

An OpenBUGS Program for the Bayesian RDINA (Reparameterized Deterministic Input Noisy And) Model

```
# Bayesian RDINA model in OpenBUGS
# Exploring 12 elements of the Q-matrix (see Table 4)
# Data is for 15 items, 4 skills, and 1000 observations

{

#priors for RDINA parameters f and d
for (j in 1:J) {
  d[j] ~ dnorm(0, 0.1) I(0,)
  f[j] ~ dnorm(0, 0.1)
}

#priors for skill class sizes
p1 ~ dbeta(1,1) p2 ~ dbeta(1,1) p3 ~ dbeta(1,1) p4 ~ dbeta(1,1)

#priors for uncertain elements
p11 ~ dbeta(1,1) q11 ~ dbern(p11) p12 ~ dbeta(1,1) q12 ~ dbern(p12)
p13 ~ dbeta(1,1) q13 ~ dbern(p13) p14 ~ dbeta(1,1) q14 ~ dbern(p14)
p81 ~ dbeta(1,1) q81 ~ dbern(p81) p82 ~ dbeta(1,1) q82 ~ dbern(p82)
p83 ~ dbeta(1,1) q83 ~ dbern(p83) p84 ~ dbeta(1,1) q84 ~ dbern(p84)
p131 ~ dbeta(1,1) q131 ~ dbern(p131) p132 ~ dbeta(1,1) q132 ~ dbern(p132)
p133 ~ dbeta(1,1) q133 ~ dbern(p133) p134 ~ dbeta(1,1) q134 ~ dbern(p134)

for (i in 1:N){
  x1[i] ~ dbern(p1) x2[i] ~ dbern(p2)
  x3[i] ~ dbern(p3) x4[i] ~ dbern(p4)
  pa[i,1] <- 1/(1 + exp(-f[1] - d[1]*(pow(x1[i],q11)*pow(x2[i],q12)*pow(x3[i],q13)*pow(x4[i],q14))))
  pa[i,2] <- 1/(1 + exp(-f[2] - d[2]*(x2[i])))
}
```

```

pa[i,3] <- 1/(1 + exp(-f[3] - d[3]*(x3[i])))
pa[i,4] <- 1/(1 + exp(-f[4] - d[4]*(x4[i])))
pa[i,5] <- 1/(1 + exp(-f[5] - d[5]*(x1[i]*x2[i])))
pa[i,6] <- 1/(1 + exp(-f[6] - d[6]*(x1[i]*x3[i])))
pa[i,7] <- 1/(1 + exp(-f[7] - d[7]*(x1[i]*x4[i])))
pa[i,8] <- 1/(1 + exp(-f[8] - d[8]*(pow(x1[i],q81)*pow(x2[i],q82)*pow(x3[i],q83)*pow(x4[i],q84))))
pa[i,9] <- 1/(1 + exp(-f[9] - d[9]*(x2[i]*x4[i])))
pa[i,10] <- 1/(1 + exp(-f[10] - d[10]*(x3[i]*x4[i])))
pa[i,11] <- 1/(1 + exp(-f[11] - d[11]*(x1[i]*x2[i]*x3[i])))
pa[i,12] <- 1/(1 + exp(-f[12] - d[12]*(x1[i]*x2[i]*x4[i])))
pa[i,13] <- 1/(1 + exp(-f[13] - d[13]*(pow(x1[i],q131)*pow(x2[i],q132)*pow(x3[i],q133)*pow(x4[i],q134))))
pa[i,14] <- 1/(1 + exp(-f[14] - d[14]*(x2[i]*x3[i]*x4[i])))
pa[i,15] <- 1/(1 + exp(-f[15] - d[15]*(x1[i]*x2[i]*x3[i]*x4[i])))
}

for (i in 1:N){
  for (j in 1:J) {
    y[i,j] ~ dbern(pa[i,j])
  }
}
}
#data
list(N=1000, J=15)

#inits
list(d=c(2,2,2,2,2,2,2,2,2,2,2,2,2,2,2),f=c(-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1,-1))

```

Author's Note

Links to software and sample programs are available at the author's website: <http://www.columbia.edu/~ld208>.

Acknowledgment

The author thanks Matthew S. Johnson for some helpful discussions.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

Note

1. It is interesting to note that the same results were found for a fit of the Bayesian version of the HO-RDINA (higher order reparameterized deterministic input noisy and) model (with free a_k).

References

- Barnes, T., Bitzer, D., & Vouk, M. (2005). Experimental analysis of the q-matrix method in knowledge discovery. *Proceedings of the 15th International Symposium on Methodologies for Intelligent Systems 2005*. Saratoga Springs, NY.
- Cen, H., Koedinger, K., & Junker, B. (2005). Learning factors analysis—A general method for cognitive model evaluation and improvement. In M. Ikeda, K. Ashley, & T. Chan (Eds.), *Intelligent tutoring systems: 8th International Conference* (pp. 164-175). Berlin, Germany: Springer.
- DeCarlo, L. T. (2010). On the statistical and theoretical basis of signal detection theory and extensions: Unequal variance, random coefficient, and mixture models. *Journal of Mathematical Psychology*, *54*, 304-313.
- DeCarlo, L. T. (2011). On the analysis of fraction subtraction data: The DINA model, classification, latent class sizes, and the Q-matrix. *Applied Psychological Measurement*, *35*, 8-26.
- de la Torre, J. (2008). An empirically based method of Q-matrix validation for the DINA model: Development and applications. *Journal of Educational Measurement*, *45*, 343-362.
- de la Torre, J. (2009). DINA model and parameter estimation: A didactic. *Journal of Educational and Behavioral Statistics*, *34*, 115-130.
- de la Torre, J., & Douglas, J. A. (2004). Higher-order latent trait models for cognitive diagnosis. *Psychometrika*, *69*, 333-353.
- de la Torre, J., & Douglas, J. A. (2008). Model evaluation and multiple strategies in cognitive diagnosis: An analysis of fraction subtraction data. *Psychometrika*, *73*, 595-624.
- DiBello, L. V., Roussos, L. A., & Stout, W. F. (2007). Review of cognitively diagnostic assessment and a summary of psychometric models. In C. R. Rao & S. Sinharay (Eds.), *Handbook of statistics, Volume 26: Psychometrics* (pp. 979-1030). Amsterdam, Netherlands: Elsevier.
- Fu, J., & Li, Y. (2007, April). *An integrative review of cognitively diagnostic psychometric models*. Paper presented at the annual meeting of the National Council on Measurement in Education, Chicago, IL.
- Geyer, C. J. (1992). Practical Markov chain Monte Carlo. *Statistical Science*, *7*, 473-483.
- Haertel, E. H. (1989). Using restricted latent class models to map the skill structure of achievement items. *Journal of Educational Measurement*, *26*, 301-321.
- Henson, R. A., Roussos, L., Douglas, J., & He, X. (2008). Cognitive diagnostic attribute-level discrimination indices. *Applied Psychological Measurement*, *32*, 275-288.
- Henson, R. A., Templin, J. L., & Willse, J. T. (2009). Defining a family of cognitive diagnosis models using log-linear models with latent variables. *Psychometrika*, *74*, 191-210.
- Junker, B. W., & Sijtsma, K. (2001). Cognitive assessment models with few assumptions, and connections with nonparametric item response theory. *Applied Psychological Measurement*, *25*, 258-272.
- Lee, P. M. (2004). *Bayesian statistics: An introduction* (3rd ed.). London, England: Hodder Education.
- Macmillan, N. A., & Creelman, C. D. (2005). *Detection theory: A user's guide* (2nd ed.). New York, NY: Cambridge University Press.
- Macready, G. B., & Dayton, C. M. (1977). The use of probabilistic models in the assessment of mastery. *Journal of Educational Statistics*, *2*, 99-120.
- Maris, E. (1999). Estimating multiple classification latent class models. *Psychometrika*, *64*, 187-212.
- Mislevy, R. J. (1996). Test theory reconceived. *Journal of Educational Measurement*, *33*, 379-416.
- Muthén, B., & Asparouhov, T. (2010). *Bayesian SEM: A more flexible representation of substantive theory*. Available from <http://www.statmodel.com>
- Rabe-Hesketh, S., Skrondal, A., & Pickles, A. (2004). *GLLAMM manual* (Technical Report No. 160, U.C. Berkeley Division of Biostatistics Working Paper Series). Berkeley: University of California. Retrieved from <http://www.bepress.com/ucbbiostat/paper160/>
- Rupp, A. A., & Templin, J. L. (2008a). The effects of Q-matrix misspecification on parameter estimates and classification accuracy in the DINA model. *Educational and Psychological Measurement*, *68*, 78-96.
- Rupp, A. A., & Templin, J. L. (2008b). Unique characteristics of diagnostic classification models: A comprehensive review of the current state-of-the-art. *Measurement*, *6*, 219-262.
- Spiegelhalter, D., Thomas, A., Best, N. G., & Lunn, D. (2003). *WinBUGS user manual*. Retrieved from <http://www.mrc-bsu.cam.ac.uk/bugs/>

- Tatsuoka, C. (2002). Data analytic methods for latent partially ordered classification models. *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, 51, 337-350.
- Tatsuoka, K. K. (1990). Toward an integration of item-response theory and cognitive error diagnosis. In N. Frederiksen, R. Glaser, A. Lesgold, & M. Shafto (Eds.), *Diagnostic monitoring of skill and knowledge acquisition* (pp. 453-488). Hillsdale, NJ: Erlbaum.
- Templin, J., & Henson, R. (2006, April). *A Bayesian method for incorporating uncertainty into Q-matrix estimation in skills assessment*. Paper presented at the annual meeting of the National Council on Measurement in Education, San Francisco, CA.
- Templin, J. L., Henson, R. A., Templin, S. E., & Roussos, L. (2008). Robustness of hierarchical modeling of skill association in cognitive diagnosis models. *Applied Psychological Measurement*, 32, 559-574.
- Thomas, A., O'Hara, B., Ligges, U., & Sturtz, S. (2006). Making BUGS open. *R News*, 6, 12-17. Retrieved from <http://mathstat.helsinki.fi/openbugs/FAQFrames.html>
- Vermunt, J. K. (1997). *LEM: A general program for the analysis of categorical data*. Tilburg, Netherlands: Tilburg University.
- Vermunt, J. K., & Magidson, J. (2005). *Technical guide for Latent Gold 4.0: Basic and advanced*. Belmont, MA: Statistical Innovations.