A Comparison of Student Skill Knowledge Estimates

Elizabeth Ayers[1], Rebecca Nugent[1], and Nema Dean[2]
{eayers, rnugent}@stat.cmu.edu, {nema}@stats.gla.ac.uk
[1]Department of Statistics, Carnegie Mellon University
[2]Department of Statistics, University of Glasgow

A fundamental goal of educational research is identifying students' current stage of skill mastery (complete/partial/none). In recent years a number of cognitive diagnosis models have become a popular means of estimating student skill knowledge. However, these models become difficult to estimate as the number of students, items, and skills grows. There exist alternatives such as sum-scores and the capability matrix. While initial theoretical work on sum-scores has been done, the behavior of sum-scores and the capability matrix is not well understood with respect to each other or to estimates from cognitive diagnosis models. In this paper we compare the performance of the three estimates of student skill knowledge under a variety of clustering methods using simulated data with varying levels of missing values.

1 Introduction

A fundamental goal of educational research is identifying students' current stage of skill mastery (complete/partial/none). In addition, finding groups of students with similar skill set profiles is important to provide feedback for classroom instruction. In recent years a number of cognitive diagnosis models [3,8] have become a popular means of estimating student skill knowledge. However, these models become difficult and time-consuming to estimate as the number of students, items, and skills increases [8]. Two alternative estimates, sum-scores [3,6] and the capability matrix [1], can be used to estimate student skill knowledge in (near to) real time. Estimates are subsequently clustered to identify similar skill set profiles.

While initial theoretical work on sum-scores has been done [3], the behavior and performance of sum-scores and the capability matrix is not well understood in comparison with each other or with estimates from cognitive diagnosis models. The performance of the methods when missing values occur is also of interest. Moreover, which clustering method to employ is an open question. In this work we take a step back and compare the performance of three estimates of student skill knowledge under a variety of clustering methods. In Section 2, we describe the three different estimates of student skill knowledge. In Section 3, we give a brief introduction to the clustering methods used. In Section 4, we show results from a simulation study incorporating varying amounts of missing data. Finally, in Section 5, we offer conclusions and thoughts on future work.

2 Estimates of Student Skill Knowledge

While there may be several possible methods to estimate student skill knowledge, this paper will consider one traditional Bayesian estimation procedure and two simpler statistics. First, we introduce notation that will be common among the methods. We begin by

assembling the skill dependencies of each item into a Q-matrix [2,12]. The Q-matrix, also referred to as a transfer model or skill coding, is a $J \times K$ matrix where $q_{jk} = 1$ if item j requires skill k and 0 if it does not, J is the total number of items, and K is the total number of skills. The Q-matrix is usually an expert-elicited assignment matrix. This paper assumes the Q-matrix is known and correct.

There are (at least) two ways in which Q-matrices can differ. First, each item could require only a single skill or multiple skills. A Q-matrix can then be comprised of all single skill items, single and multiple skill items, or all multiple skill items. Second, the Q-matrix may have a balanced or unbalanced design. In a balanced design, all single skill items occur the same number of times, and each combination of skills occurs the same number of times. For example, if K = 3 and J = 30 one possible balanced design would be: five single skill items for each skill, four double skill items for each pair of skills, and three triple skill items. A design could be unbalanced in two ways. Either all skills or combinations of skills are present but do not occur the same number of times or there are missing skills or combinations of skills.

$$Q = \begin{bmatrix} q_{1,1} & q_{1,2} & \cdots & q_{1,K} \\ \vdots & \ddots & & \vdots \\ q_{J,1} & q_{J,2} & \cdots & q_{J,K} \end{bmatrix}, \quad Y = \begin{bmatrix} y_{1,1} & y_{1,2} & \cdots & y_{1,J} \\ \vdots & \ddots & & \vdots \\ y_{N,1} & y_{N,2} & \cdots & y_{N,J} \end{bmatrix}$$

We then assemble student responses in a $N \times J$ response matrix Y where y_{ij} indicates both if student i attempted item j and whether or not they answered item j correctly and N is the total number of students. If student i did not answer item j then $y_{ij} = NA$. The indicator $I_{y_{ij} \neq NA} = 0$ expresses this missing value. If student i attempted item j ($I_{y_{ij} \neq NA} = 1$), then $y_{ij} = 1$ if they answered correctly, or 0 if they answered incorrectly.

2.1 DINA Model Estimates

The first method of estimating student skill knowledge uses a common conjunctive cognitive diagnosis model. The deterministic inputs, noisy "and" gate model (DINA; [8]) models student responses as

$$P(Y_{ij} = 1 \mid \eta_{ij}, s_j, g_j) = (1 - s_j)^{\eta_{ij}} g_j^{1-\eta_{ij}} \tag{1}$$

where $\alpha_{ik} = I_{\{\text{Student } i \text{ has skill } k\}}$ indicates if student i possesses skill k, $\eta_{ij} = \prod_{k=1}^{K} \alpha_{ik}^{q_{jk}}$ indicates if student i has all skills needed for item j, $s_j = P(Y_{ij} = 0 \mid \eta_{ij} = 1)$ is the slip parameter, and $g_j = P(Y_{ij} = 1 \mid \eta_{ij} = 0)$ is the guess parameter. If a student is missing any of the required skills, the probability that they will answer an item correctly drops due to the conjunctive assumption.

We estimate the student skill knowledge parameters of the DINA model, the α_{ik}, using Markov Chain Monte Carlo methods with the program WinBUGS (Bayesian Inference Using Gibbs Sampling, [9]). In the model, the α_{ik} are 0/1 indicating whether or not student i has mastered skill k. Our estimates will be $\hat{\alpha}_{ik} \in [0, 1]$. We can think of the $\hat{\alpha}_{ik}$ as the probability that student i has mastered skill k.

2.2 Sum-scores

The second estimate we consider is the sum-score method of [3,6]. Here $W_i = (W_{i1}, W_{i2}, ..., W_{iK})$ is a vector of sum-scores where the k^{th} component is defined as

$$W_{ik} = \sum_{j=1}^{J} y_{ij} q_{jk}, \tag{2}$$

where y_{ij} and q_{jk} are the corresponding entries from the response matrix Y and Q-matrix. Thus, the components of W_i are simply the number of items student i answered correctly for each skill k. When an item requires more than one skill it will contribute to more than one component of W_i. The range of W_{ik} may be different for each k if the skills are required by a different number of problems.

2.3 Capability Matrix

Finally, we consider the *capability matrix* defined in [1]. The capability matrix B is an $N \times K$ matrix where B_{ik} is the proportion of correctly answered items involving skill k that student i attempted. Thus,

$$B_{ik} = \frac{\sum_{j=1}^{J} I_{y_{ij} \neq NA} \cdot y_{ij} \cdot q_{jk}}{\sum_{j=1}^{J} I_{y_{ij} \neq NA} \cdot q_{jk}}, \tag{3}$$

where y_{ij} and q_{jk} are the corresponding entries from the response matrix Y and Q-matrix. The capability matrix expands on sum-scores by accounting for the number of items requiring skill k that student i answered. In this manner the statistic scales for the number of items in which the skill appears as well as for missing data. If a student has not seen all of the items requiring a particular skill, we still derive an estimate based on the available information. If student i completes no items involving skill k, then $B_{ik} = NA$. In this case, we impute an uninformative value (e.g., 0.5, mean, median) to map students to the hypercube. Exploring the performance of these imputation choices is ongoing. For this paper we assume that the data are complete or that missing B-values are appropriately imputed.

We can note that both the DINA model estimates and the B-matrix values map students into a K-dimensional hypercube (for each dimension, zero indicates total lack of skill mastery, one is complete skill mastery, and values in between are less certain). The 2^K corners of the hypercube correspond to natural skill set profiles $C_i = \{C_{i1}, C_{i2}, ..., C_{iK}\}, C_{ik} \in \{0, 1\}$.

Additionally, we can note theoretical connections between the sum-scores and B-matrix values. If there are no missing response values y_{ij}, then

$$W_{ik} = J_k B_{ik}, \tag{4}$$

where J_k is the number of items that require skill k. When all students have answered all questions and there is a balanced Q-matrix design (i.e., $J_1 = J_2 = ... = J_K$), the two estimates will mapto the same (scaled) feature space. In this case, we expect the two estimates to perform similarly. However, when there is either missing data or an unbalanced

Q-matrix design, the space to which the estimates map will be different. In this case, we cannot guarantee that performance will be similar.

3 Clustering Methods

To identify groups of students with similar skill set profiles, we cluster the student skill knowledge estimates. In this paper we will compare the performance of three common clustering methods: hierarchical agglomerative clustering, K-means, and model-based clustering. In the sections below we briefly introduce each of these methods.

3.1 Hierarchical Agglomerative Clustering

Hierarchical agglomerative clustering (HAC; [10]) links groups in order of closeness to form a tree structure from which a clustering solution can be extracted. Euclidean distance is most commonly used to measure the distance between groups. The method also requires the user to specify how to measure the distance between groups. We will use "complete" linkage where the distance between any two groups is defined as the largest distance between two observations, one from each group. In HAC, all observations begin as their own group. The two closest groups are merged and all inter-group distances are recalculated. We continue merging groups and recalculating distances until a single group with all observations is formed. Once the tree structure is formed, we can extract the desired number of clusters G by cutting the tree at a height corresponding to G branches.

3.2 K-means

K-Means [5] is a popular iterative descent algorithm for data $X = \{\underline{x}_1, \underline{x}_2 ..., \underline{x}_n\}$, $\underline{x}_i \in \Re^K$. It uses squared Euclidean distance as a dissimilarity measure and tries to minimize within-cluster distance and maximize between-cluster distance. For a given number of clusters G, K-Means searches for cluster centers m_g and assignments A that minimize the criterion

$$\min_A \sum_{g=1}^{G} \sum_{A(i)=g} ||\underline{x}_i - m_g||^2.$$

The algorithm alternates between optimizing the cluster centers for the current assignment (by the current cluster means) and optimizing the cluster assignment for a given set of cluster centers (by assigning to the closest current center) until convergence (i.e. cluster assignments do not change). It tends to find compact, spherical clusters and requires *a priori* both the number of clusters G and a starting set of cluster centers. The final cluster assignment can be sensitive to the choice of centers; a common method for initializing K-Means is to randomly choose G observations.

3.3 Model-based Clustering

Model-based clustering [4, 11] is a parametric statistical approach that assumes: the data $X = \{\underline{x}_1, \underline{x}_2, ..., \underline{x}_n\}$, $\underline{x}_i \in \Re^K$ are an independently and identically distributed sample from an unknown population density $p(\underline{x})$; each population group g is represented by a

Table 1: Clustering the DINA Model Estimates of Student Skill Knowledge

N	J	K	Q-matrix design	DINA	HAC	K-means	MBC	MBC 2^K
250	30	3	Single, bal	1.000	1.000	0.8739	0.9966	1.000
				(0.0054)	*(0.0054)*	*(0.0736)*	*(0.0895)*	*(0.0349)*
250	30	3	Both, bal	0.9793	0.9781	0.8367	0.8915	0.9632
				(0.0179)	*(0.0200)*	*(0.1192)*	*(0.0882)*	*(0.1087)*
250	30	3	Both,unbal, all	0.9657	0.9657	0.7789	0.9129	0.9350
				(0.0285)	*(0.2920)*	*(0.0941)*	*(0.0505)*	*(0.0758)*
250	30	3	Both,unbal,miss	0.9240	0.9131	0.7696	0.8811	0.9132
				(0.0395)	*(0.0427)*	*(0.0858)*	*(0.0696)*	*(0.0428)*
250	30	3	Mult, bal	0.4677	0.5127	0.5012	0.5282	0.4979
				(0.0292)	*(0.0443)*	*(0.0578)*	*(0.0690)*	*(0.0411)*
250	30	3	Mult, unbal, all	0.4629	0.4874	0.4948	0.5130	0.4790
				(0.0430)	*(0.0536)*	*(0.0816)*	*(0.0736)*	*(0.0495)*
250	30	3	Mult, unbal, miss	0.3239	0.4070	0.3835	0.4266	0.4090
				(0.0380)	*(0.0596)*	*(0.0521)*	*(0.0837)*	*(0.0630)*
500	68	5	Both, bal	0.9463	0.9428	0.7132	0.8348	0.9243
				(0.0184)	*(0.0188)*	*(0.0428)*	*(0.1123)*	*(0.0488)*
500	68	5	Both, unbal, miss	0.8724	0.8729	0.6665	0.8213	0.8624
				(0.0247)	*(0.0219)*	*(0.0466)*	*(0.0960)*	*(0.0226)*
300	40	7	Single	0.9041	0.8891	0.7674	0.3050	0.8881
				(0.0262)	*(0.0286)*	*(0.0409)*	*(0.1203)*	*(0.0282)*

(often Gaussian) density $p_g(\underline{x})$; and $p(\underline{x})$ is a weighted mixture of these density components, i.e. $p(\underline{x}) = \sum_{g=1}^{G} \pi_g \cdot p_g(\underline{x}; \theta_g)$ where $\sum \pi_g = 1$, $0 < \pi_g \leq 1$ for $g = 1, 2, ..., G$, and $\theta_g = (\mu_g, \Sigma_g)$ for Gaussian components. The method chooses the number of components G by maximizing the Bayesian Information Criterion (BIC) and estimates the means and variances (μ_g, Σ_g) via maximum likelihood. While it may assume Gaussian components, its flexibility on their shape, volume, and orientation allows student groups of varying shapes and sizes. When multiple students map to the same location, model-based clustering is known to overfit the data by using spikes with near singular covariance in these locations [4]. To alleviate this concern, we jitter the student skill estimates by a small amount (0.01). The effect on our results is minimal.

4 Simulation Study

To compare the skill knowledge estimates and clustering methods described above we did a simulation study using generated data from the DINA model (Equation 1). The Q-matrix design is varied to include balanced and unbalanced combinations of single and multiple skill items. Then, for a fixed Q-matrix design, we simulate 20 different student populations. Skill difficulties are always set to equal medium difficulty; inter-skill correlations are set to zero. These choices evenly spread students among the 2^K natural skill set profiles $[0, 1]^K$. For each student population, we generate true skill set profiles C_i. We then draw slip and guess parameters from a random uniform distribution ($s_j \sim$ Unif(0,0.30); $g_j \sim$

Table 2: Clustering the Sum-scores Estimates of Student Skill Knowledge

N	J	K	Q-matrix design	HAC	K-means	MBC	MBC 2^K
250	30	3	Single, bal	0.9910	0.8549	0.9191	0.9957
				(0.0110)	*(0.0960)*	*(0.2899)*	*(0.0071)*
250	30	3	Both, bal	0.7644	0.8156	0.9321	0.9442
				(0.1095)	*(0.1110)*	*(0.1181)*	*(0.0515)*
250	30	3	Both,unbal, all	0.6398	0.7707	0.6970	0.8494
				(0.0889)	*(0.0951)*	*(0.2138)*	*(0.0713)*
250	30	3	Both,unbal,miss	0.6482	0.6728	0.7066	0.7661
				(0.0511)	*(0.0650)*	*(0.2064)*	*(0.1095)*
250	30	3	Mult, bal	0.3950	0.4720	0.4383	0.4375
				(0.0339)	*(0.0648)*	*(0.0675)*	*(0.0517)*
250	30	3	Mult, unbal, all	0.3862	0.4606	0.4380	0.4481
				(0.0533)	*(0.0670)*	*(0.0696)*	*(0.0428)*
250	30	3	Mult, unbal, miss	0.2689	0.2827	0.3314	0.3099
				(0.0273)	*(0.0848)*	*(0.0352)*	*(0.0347)*
500	68	5	Both, bal	0.4006	0.5859	0.5893	0.6523
				(0.0560)	*(0.0442)*	*(0.1223)*	*(0.0432)*
500	68	5	Both, unbal, miss	0.4104	0.54412	0.6010	0.6265
				(0.0373)	*(0.0366)*	*(0.0537)*	*(0.0397)*
300	40	7	Single	0.7348	0.6474	0.0973	0.7080
				(0.0526)	*(0.0456)*	*(0.0362)*	*(0.0453)*

Unif(0,0.15)). Given profiles and slip/guess parameters, we generate the student response matrix Y.

As we know the true underlying skill set profiles C_i, we can calculate their agreement with the clustering partitions using the Adjusted Rand Index (ARI; [7]), a common measure of agreement between two partitions. The expected value of the ARI is zero and the maximum value is one, with larger values indicating better agreement.

Tables 1, 2, and 3 show the clustering results for the DINA model estimates, sum-scores, and the capability matrix, respectively. In each table, N is the number of students, J is the number of items, and K is the number of skills. The Q-matrix design describes the Q-matrix used when generating the student responses (see Section 2 for more details). Here *single* indicates that there were only single skill items, *both* indicates that there were both single and multiple skill items, and *mult* indicates that there were only multiple skill items. Also, *bal* indicates that the Q-matrix had a balanced design. An unbalanced design is denoted by *unbal* and *all* or *miss* shows whether all combinations were present or if some were missing. For the DINA model estimates (Table 1), we rounded the $\hat{\alpha}_{ik}$ to 0/1 to find the closest skill set profile. For the remaining methods in Table 1 and for all methods in Tables 2 and 3 we cluster the unrounded $\hat{\alpha}_{ik}$. When using HAC and K-means, we set the number of clusters equal to 2^K as suggested by [3]. For MBC we search over an appropriate range; MBC 2^K indicates that we set the number of clusters to 2^K. For each set of 20 simulations,

Table 3: Clustering the Capability Matrix Estimates of Student Skill Knowledge

N	J	K	Q-matrix design	HAC	K-means	MBC	MBC 2^K
250	30	3	Single, bal	0.9910	0.8190	0.9957	0.9957
				(0.0104)	*(0.0835)*	*(0.0071)*	*(0.0071)*
250	30	3	Both, bal	0.7644	0.7947	0.9353	0.9411
				(0.1095)	*(0.1056)*	*(0.1583)*	*(0.0300)*
250	30	3	Both,unbal, all	0.7273	0.8082	0.6252	0.8281
				(0.0867)	*(0.1227)*	*(0.1719)*	*(0.1543)*
250	30	3	Both,unbal,miss	0.6698	0.7390	0.4563	0.6693
				(0.0813)	*(0.0778)*	*(0.1267)*	*(0.1628)*
250	30	3	Mult, bal	0.4045	0.4530	0.4586	0.4499
				(0.0347)	*(0.0508)*	*(0.0624)*	*(0.0382)*
250	30	3	Mult, unbal, all	0.3899	0.4585	0.4518	0.4580
				(0.0509)	*(0.0550)*	*(0.0822)*	*(0.0589)*
250	30	3	Mult, unbal, miss	0.2700	0.3638	0.2803	0.2840
				(0.0291)	*(0.0737)*	*(0.0620)*	*(0.0457)*
500	68	5	Both, bal	0.4096	0.5711	0.5951	0.6647
				(0.0504)	*(0.0543)*	*(0.1284)*	*(0.0928)*
500	68	5	Both, unbal, miss	0.4327	0.5435	0.5560	0.6291
				(0.0405)	*(0.0350)*	*(0.2027)*	*(0.1050)*
300	40	7	Single	0.7399	0.6437	0.0906	0.7109
				(0.0545)	*(0.0402)*	*(0.0168)*	*(0.0409)*

we report the median ARI and the standard deviation.

First, we examine performance differences across Q-matrix designs. The first Q-matrix has only three skills; each skill occurs in 10 single skill items. The ARI for all three methods of estimation and all clustering methods is 1 in nearly all cases. Across the methods, K-means has the lowest ARI. This is not surprising as we randomly select $2^K = 8$ observations as the starting centers. A more informed set of starting centers (i.e., the natural skill set profiles) may lead to better performance. For the $K = 3$ examples, the ARI is higher when there are only single skill items compared to when there are both single and multiple skill items and only multiple skill items. The lone exception is MBC with sum-scores (*Single, bal* = 0.9191, *Both, bal* = 0.9321). The standard deviation in this case (0.2899) is rather large and indicates a wide range of ARI values for these 20 simulated datasets.

We now take a closer look at Q-matrices with at least some multiple skill items. We can note that the performance of all three clustering methods is better (as indicated by a higher ARI) when there are both single and multiple skill items in the Q-matrix, compared to only multiple skill items (also true across all three methods of estimation). In addition, when the Q-matrix has a balanced design, as opposed to an unbalanced design, the recovery of the true skill set profiles is better. In general, the performance of the three estimates of the student skill knowledge is similar across the clustering methods. This similar performance is particularly interesting since using sum-scores and the capability matrix yield large com-

Table 4: Clustering the DINA Model Estimates of Student Skill Knowledge for $N = 250, J = 30, K = 3$ with Missing Response Data

Q-matrix design	% missing	DINA	HAC	K-means	MBC	MBC 2^K
Both, bal	0	0.9793	0.9781	0.8367	0.8915	0.9632
Both, bal	10	0.4584	0.4690	0.4750	0.4725	0.4754
Both, bal	20	0.4326	0.4550	0.4581	0.4544	0.4567
Both, bal	30	0.4006	0.4340	0.4276	0.4267	0.4306
Both, bal	40	0.3513	0.3825	0.3850	0.3655	0.3681
Both, unbal, miss	0	0.9240	0.9131	0.7696	0.8811	0.9132
Both, unbal, miss	10	0.9084	0.9057	0.7516	0.8274	0.8009
Both, unbal, miss	20	0.8775	0.8651	0.7294	0.7560	0.7578
Both, unbal, miss	30	0.8193	0.8160	0.7256	0.7052	0.6948
Both, unbal, miss	40	0.7694	0.7746	0.7181	0.6515	0.6114

Table 5: Clustering the Sum-Score Estimates of Student Skill Knowledge for $N = 250, J = 30, K = 3$ with Missing Response Data

Q-matrix design	% missing	HAC	K-means	MBC	MBC 2^K
Both, bal	0	0.7644	0.8156	0.9321	0.9442
Both, bal	10	0.6255	0.7671	0.8280	0.8489
Both, bal	20	0.5000	0.6717	0.4854	0.7526
Both, bal	30	0.4191	0.5855	0.4131	0.5309
Both, bal	40	0.3168	0.5072	0.2951	0.3867
Both, unbal, miss	0	0.6482	0.6728	0.7066	0.7661
Both, unbal, miss	10	0.5744	0.6091	0.3608	0.6563
Both, unbal, miss	20	0.4834	0.5556	0.3264	0.5414
Both, unbal, miss	30	0.3686	0.4876	0.2725	0.3961
Both, unbal, miss	40	0.3266	0.4203	0.2514	0.2624

putational savings when compared to estimating the DINA model using WinBUGS (up to 700 times faster; [1]). Moreover, in this simulation study the data are generated from the DINA model; we would expect the Bayesian estimation to perform well in this best-case scenario. For sum-scores and the capability matrix to perform as well as, and better than in some cases, the DINA model is noteworthy.

The above results are for student populations with complete response data. In practice, missing responses (unanswered questions) will be ubiquitous. We chose two Q-matrix designs with $N = 250$, $J = 30$, and $K = 3$ (*Both, bal* and *Both, unbal, miss*) and removed 0, 10, 20, 30, and 40% of the student responses completely at random for each of the 20 student populations. Results can be seen in Tables 4, 5, and 6. Note that the 0% missing corresponds to the previously shown results. Again, we report the median ARI. The standard deviations are not shown due to space limitations. They ranged from 0.03 to 0.16 and were generally ordered as DINA model (lowest), capability matrix, and sum-score (highest).

Table 6: Clustering the Capability Matrix Estimates of Student Skill Knowledge for $N = 250, J = 30, K = 3$ with Missing Response Data

Q-matrix design	% missing	HAC	K-means	MBC	MBC 2^K
Both, bal	0	0.7644	0.7947	0.9353	0.9411
Both, bal	10	0.6682	0.7894	0.6633	0.8786
Both, bal	20	0.6028	0.7491	0.5350	0.7655
Both, bal	30	0.6022	0.7141	0.5021	0.5505
Both, bal	40	0.4842	0.6103	0.3948	0.4086
Both, unbal, miss	0	0.6698	0.7390	0.4563	0.6693
Both, unbal, miss	10	0.6032	0.6980	0.4766	0.5473
Both, unbal, miss	20	0.5761	0.6629	0.4687	0.4654
Both, unbal, miss	30	0.5351	0.6251	0.4764	0.4775
Both, unbal, miss	40	0.5108	0.5658	0.4144	0.4335

In general, as the amount of missing data increases, the ARI decreases across all three estimation methods and all methods of clustering. However, some methods show more substantial decreases than others. When using the capability matrix, K-means shows relatively stable performance for both Q-matrix designs. For the *Both, unbal, miss* design, HAC and MBC also show stable performances. When using sum-scores, the performance drops more noticeably across all clustering methods which may reflect that the capability matrix scales for the number of questions answered while sum-scores do not. In the *Both, bal* case, the performance of the capability matrix estimates is generally better than both the DINA model estimates and the sum-scores (particularly true for K-means). For HAC, sum-scores and the capability matrix perform similarly (both better than the DINA model estimates). For the *Both, unbal, miss* case, the performance of the DINA model estimates is better than both sum-scores and the capability matrix estimates. When using the capability matrix estimates, K-means clustering performs best; its ARI values are only slightly lower than those of the DINA model.

5 Conclusions

Simulated examples show that recovery of the true skill set profiles is best when only single skill items occur. For Q-matrices with multiple skill items, recovery is improved if there are also single skill items present. These results hold across all three clustering methods and all three estimates of student skill knowledge. In addition, we note that the more computationally attractive capability matrix and the sum-score estimates perform similarly to the Bayesian estimation of the DINA model.

However, when there are missing responses, the performance of the estimation procedures changes. In general, the ARI values decrease as the percent of missingness increases (across all estimation and clustering methods). When the Q-matrix has a *Both, bal* design, the capability matrix estimates perform better than both the DINA model and sum-score estimates. In the *Both, unbal, miss* design, the DINA model estimates perform better than

sum-scores and the capability matrix estimates.

These results can be used to guide the design of exams and tutor problems. For better estimation of student skill knowledge, single skill items should be included for each skill. In addition, students should be encouraged to finish all items. Whether or not it is by design, when students use online tutors, for example, they often do not complete all the items. In this case, it is particularly important for single skill items to be included. In the presence of missing responses, however, care should be taken when choosing an estimation method and a clustering method. The best choice is not obvious.

While there are benefits of using the capability matrix and/or sum-scores, we note that if an item requires multiple skills and a student answers incorrectly, all skills required by the item will receive a penalty, even if the student has mastered one (or more) of the skills. In future work, we will explore the behavior of alternative estimates that better account for multiple skill items. Possible methods could use empirical performance on single skill items or weight by the number of skills required by the incorrectly answered item.

References

[1] Ayers, E, Nugent, R, Dean, N. "Skill Set Profile Clustering Based on Student Capability Vectors Computed from Online Tutoring Data". *Educational Data Mining 2008: 1st International Conference on Educational Data Mining, Proceedings* (refereed). R.S.J.d. Baker, T. Barnes, and J.E. Beck (Eds), Montreal, Quebec, Canada, June 20-21, 2008. p.210-217.

[2] Barnes, T.M. (2003). *The Q-matrix Method of Fault-tolerant Teaching in Knowledge Assessment and Data Mining*. Ph.D. Dissertation, Department of Computer Science, North Carolina State University.

[3] Chiu, C. (2008). *Cluster Analysis for Cognitive Diagnosis: Theory and Applications*. Ph.D. Dissertation, Educational Psychology, University of Illinois at Urbana Champaign.

[4] Fraley, C. and Raftery, A. Mclust: Software for model-based cluster analysis. *Journal of Classification*, 1999, 16, 297-306.

[5] Hartigan, J. and Wong, M.A. A k-means clustering algorithm. *Applied Statistics*, 1979, 28, 100-108.

[6] Henson, J., Templin, R., and Douglas, J. Using efficient model based sum-scores for conducting skill diagnoses. *Journal of Education Measurement*, 2007, 44, 361-376.

[7] Hubert, L. and Arabie, P. Comparing partitions. *Journal of Classification*, 1985, 2, 193-218.

[8] Junker, B.W. and Sijtsma K. Cognitive Assessment Models with Few Assumptions and Connections with Nonparametric Item Response Theory. *Applied Psych Measurement*, 2001, 25, 258-272.

[9] Lunn, D.J., Thomas, A., Best, N., and Spiegelhalter, D. WinBUGS – a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing*, 2000, 10, 325–337.

[10] Mardia, K.V., Kent, J.T., and Bibby, J.M. *Multivariate Analysis*. Academic Press, 1979.

[11] McLachlan, G.J., and Basford, K.E. *Mixture Models: Inference and Applications to Clustering*. Marcel Dekker, 1988.

[12] Tatsuoka, K.K. (1983). Rule Space: An Approach for Dealing with Misconceptions Based on Item Response Theory. *Journal of Educational Measurement*. 1983, Vol. 20, No. 4, 345-354.