Methods to find the number of latent skills

Behzad Beheshti
Département de génie
informatique et génie logiciel
Polytechnique Montréal
C.P. 6079, succ. Centre-Ville
Montréal, Québec, Canada
behzad.beheshti@polymtl.ca

Michel C. Desmarais
Département de génie
informatique et génie logiciel
Polytechnique Montréal
C.P. 6079, succ. Centre-Ville
Montréal, Québec, Canada
michel.desmarais@polymtl.ca

Rhouma Naceur
Département de génie
informatique et génie logiciel
Polytechnique Montréal
C.P. 6079, succ. Centre-Ville
Montréal, Québec, Canada
rhouma.naceur@polymtl.ca

ABSTRACT

Identifying the skills that determine the success or failure to exercises and question items is a difficult task. Multiple skills may be involved at various degree of importance, and skills may overlap and correlate. In an effort towards the goal of finding the skills behind a set of items, we investigate two techniques to determine the number of dominant latent skills. The Singular Value Decomposition (SVD) is a known technique to find latent factors. The singular values represent direct evidence of the strength of latent factors. Application of SVD to finding the number of latent skills is explored. We introduce a second technique based on a *wrapper* approach. Linear models with different number of skills are built, and the one that yields the best prediction accuracy through cross validation is considered the most appropriate. The results show that both techniques are effective in identifying the latent factors over synthetic data. An investigation with real data from the fraction algebra domain is also reported. Both the SVD and *wrapper* methods yield results that have no simple interpretation.

1. INTRODUCTION

A critical component of student models is the skills mastery profile. Personalization of the learning content relies heavily on this component in many, if not most intelligent tutoring systems. The more precise the skills mastery profile is, the more appropriate this personalization process will be.

However, finding the latent skills underlying exercises or questions items is non-trivial because of a number of reasons.

One reason is that multiple skills may be involved at various degree of importance with regards to a single item. This is in fact typical of most items. For example solving a simple fraction algebra problem may require knowledge of a few algebra rules, each rule representing a specific skill. More general skills such as vocabulary and grammar rules may be involved in language related task, etc.

Another difficulty is that skills may overlap and they will therefore correlate. Highly correlated skills result in similar response patterns to a set of items.

Finally, the nature of the items and the difficulty of mastering some skills will result in *slip* and *guesses.* Those will be reflected as noise that will make the identification of the latent skills more difficult.

Most of the time, the latent skills underlying question items are defined by experts. Models such as Knowledge Tracing [2], Constraint-based Modeling [7], or Performance Factor Analysis [8], are well known examples that require expert defined mapping of skills to latent factors. Some studies have looked at means to help this process.

Suraweera et al. have used an ontology-based approach to facilitate the item to skill mapping and the more general task of building the domain model [9].

Others have studied the mapping of items to skills with data driven algorithms with some success [1; 3; 11]. Their results show that mappings can be successfully derived in certain conditions of low noise (*slip* and *guess*) relative to the latent factors. However, these studies assume that the number of skills are known in advance, which is rarely the case. Although some of the the latent skills may be relatively obvious, the obvious skills only set a minimum number. That minimum does not preclude that other skills may come into play and have a strong effect also.

Of course, we do not need to identify all the skills behind an item in order to use the item outcome for assessment purpose. As long as we can establish a minimally strong tie from an item to a skill, this is a sufficient condition to use the item in the assessment of that skill. But knowledge that there is a fixed number of determinant factors to predict item outcome is a useful information. For example, if a few number of skills, say 6, are meant to be assessed by a set of 20 questions items, and we find that the underlying number of determinant latent factors behind these items is very different than 6, then it gives us a hint that our 6-skills model may not be congruent with the assessment result.

This study aims at identifying this number. It aims at finding means to estimate how many latent factors are influencial enough to determine the item success. We explore two techniques towards this end: Singular Value Decomposition (SVD) and a *wrapper* selection feature based on Nonnegative Matrix Factorization (NMF). We describe these techniques in more details and report the results of our experiments to validate their effectiveness for estimating the number of latent skills[1].

[1] The reader interested in more details is referred to the code that was used in this study: `http://www.professeurs.polymtl.ca/michel.desmarais/Papers/EDM2012/scripts.html`

2. SVD-BASED METHOD

Singular Value Decomposition (SVD) is a well known matrix factorization technique that decomposes any matrix, $\mathbf{A}$, into three sub-matrices:

$$\mathbf{A} = \mathbf{U}\mathbf{D}\mathbf{V}^T \tag{1}$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthonormal matrices and their column vectors respectively represent the eigenvectors of $\mathbf{A}\mathbf{A}^T$ and $\mathbf{A}^T\mathbf{A}$. $\mathbf{D}$ is a diagonal matrix that contains the singular values. They are the square root of the eigenvalues of the eigenvectors and are sorted in a descending order.

Because the singular values represent scaling factors of the unit eigenvectors in equation (1), they are particularly useful in finding latent factors that are dominant in the data. This is demonstrated with simulated data below. First we describe the simulated data and the results of applying SVD on the students item outcome results matrix $\mathbf{R}$.

2.1 Simulated data

The synthetic data is generated by defining a Q-matrix of 21 items that combine 6 skills. The 21 items are represented as columns in figure 1. They span the space of all pairwise combinations of skills (first 15 columns) plus 6 single skill items (last 6 columns).

Skills	Items
1	111110000000000100000
2	100001111000000010000
3	010001000111000001000
4	001000100100110000100
5	000100010010101000010
6	000010001001011000001

Figure 1: Conjunctive Q-matrix composed of 21 items that span all combinations of 6 skills for pairs of skills and single skills

Figure 1's Q-matrix is used to generate simulated data and we assume a *conjunctive* model (all skills are necessary to answer the item correctly). The data contains the 21 question items and 200 simulated student responses over these items. The six skills are assigned an increasing degree of difficulty from 0.17 to 0.83 on a standard normal (Gaussian) scale, and each student is assigned a skill vector based on a {0,1} sampling with a probability corresponding to this difficulty (or *easiness* in fact, since higher values bring greater chances of skill mastery). The choice of these difficulty values stems from the need to have a mean student success score around 50%–60%: because 15 of the 21 items require the conjunction of two skills, mean skill mastery must be substantially higher than 50% to obtain average results around 50%–60%.

Once a skills mastery profile is assigned to students, represented by a matrix $\mathbf{S}$, an *ideal response matrix* is generated according to the product $\neg\mathbf{R} = \mathbf{Q}\neg\mathbf{S}$, where $\mathbf{Q}$ is a *conjunctive* Q-matrix (more details about this model are given later, see equation (3) below). Then, slip and guess factors are used to generate noise in the ideal response pattern by randomly changing a proportion of the item success and failures outcomes according respectively to slip and guess values. The slip and guess values of respectively 0.1 and 0.2 will result in approximately 15% of the item outcomes being inconsistent with the ideal response matrix (15% corresponds to a weighted average of 0.1 and 0.2).

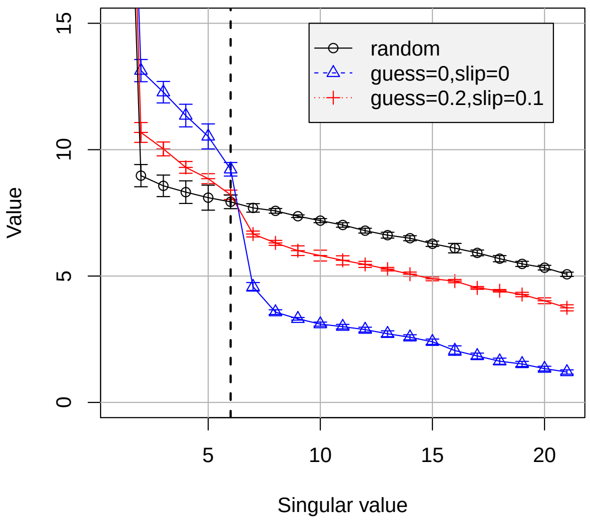

Figure 2: Singular values of simulated data for a 21 items test. Unit standard error bars for a 10-fold simulations is drawn for each line. A vertical dashed line at singular value 6 corresponds to the number of underlying latent skill factors.

2.2 Results

The results of the SVD method are shown in figure 2. The x is the index of the singular value, and the y axis is its actual value. Recall that the singular values of SVD indicate the strength of latent factors.

Three conditions are reported in figure 2. The y values at 1 on the x scale are truncated on the graph to allow a better view of the interesting region of the graph, but the highest value is from the [guess=0, slip=0] condition and the lowest is for the *random* condition. The random curve condition can be obtained by simulating random $\{0, 1\}$ values and ensuring that the overall average score of the results matrix reflects the original's data average. In this random condition, the slope from singular value 2 to 21 remains relatively constant, suggesting no specific number of skills. In condition [guess=0, slip=0], a sharp drop occurs between singular values of 6 and 7. Then the slope remains relatively constant from values 8 to 21. The largest drop is clearly at value 6 which corresponds to the underlying number of skills. In the third condition [guess=0.2, slip=0.1], the largest drop still remains visible between 6 and 7, but not as sharp as for the noiseless condition, as expected.

In other experiments with various number of skills, not reported here due to space constraints, we observed similar patterns. Another observation is that the random curve intersects with the other two after the number of underlying latent skills (after 6 in figure 2's experiment).

Therefore, the SVD method does allow for the identification of the number of skills with synthetic data, at least up to the [guess=0.2, slip=0.1] level.

3. WRAPPER-BASED METHOD

We introduce a second method to determine the number of dominant skills behind items based on a *wrapper* approach. In statistical learning, the *wrapper* approach refers to a general method for selecting the most effective set of variables by measuring the predictive performance of a model with each variables set (see [6]). In our context, we assess the predictive performance of linear models embedding different number of latent skills. The model that yields the best predictive performance is deemed to reflect the optimal number of skills.

3.1 A Linear Model of Skills Assessment

The wrapper method requires a model that will predict item outcome. A linear model of skills is defined for that purpose on the basis of the following product of matrices:

$$\mathbf{R} = \mathbf{Q}\mathbf{S} \tag{2}$$

where the $\mathbf{R}$ matrix contains observable student results with item rows and student columns, and the $\mathbf{S}$ matrix is the skills (rows) per students (columns) mastery profile (see for e.g., [3]). Matrix $\mathbf{Q}$ is the Q-matrix that maps items (rows) to skills (columns). Normalizing row sums of $\mathbf{Q}$ to 1 would yield values of 1 in the results matrix, $\mathbf{R}$, if all skills necessary to succeed an item is mastered by the corresponding individual. Equation (2) represents a *compensatory* interpretation of skills modeling, where each skill contributes additively to the success of an item.

A *conjunctive* model can be defined according to the following equation [1; 4] :

$$\neg\mathbf{R} = \mathbf{Q}\neg\mathbf{S} \tag{3}$$

where the operator $\neg$ is the Boolean negation, which is defined as a function that maps a value of 0 to 1 and any other value to 0. This equation will yield values of 0 in $\mathbf{R}$ whenever an examinee is missing one or more skills for a given item, and yield 1 whenever all necessary skills are mastered by an examinee.

3.2 Overview of the method

To estimate the optimal number of skills, the *wrapper* model can either correspond to equation (2) or (3). We will focus our explanations around equation (2), but they obviously apply to (3) if $\mathbf{R}$ and $\mathbf{S}$ are negated.

This model states that, given estimates of $\mathbf{Q}$ and $\mathbf{S}$, we can predict $\mathbf{R}$. We refer to these estimates as $\hat{\mathbf{Q}}$ and $\hat{\mathbf{S}}$, and to the predictions as $\hat{\mathbf{R}} = \hat{\mathbf{Q}}\hat{\mathbf{S}}$. The goal is therefore to derive estimates of $\hat{\mathbf{Q}}$ and $\hat{\mathbf{S}}$ with different number of skills and measure the residual difference between $\hat{\mathbf{R}}$ and $\mathbf{R}$.

First, $\hat{\mathbf{Q}}$ is learned from an independent set of training data. Then, $\hat{\mathbf{S}}$ is learned from the test data, and the residuals are computed[2].

An estimate of $\hat{\mathbf{Q}}$ is obtained through Non-negative Matrix Factorization (NMF). Details on applying this technique to the problem of deriving a Q-matrix from data is found in [3] and we limit our description to the basic principles and issues here.

NMF decomposes a matrix into two matrices composed solely of non-negative values. Its structure is equivalent to equation (2). The technique requires to choose a rank for the decomposition, which corresponds in our situation to the number of skills (i.e. number of columns of $\mathbf{Q}$ and number of rows of $\mathbf{S}$). Because NMF constrains $\mathbf{Q}$ and $\mathbf{S}$ to non-negative values, their respective interpretation as a Q-matrix and a as student skills assessments is much more natural than other matrix factorization techniques such as Principal Component Analysis, for example. However, multiple solutions exists to this factorization and there are many algorithms that can further constrain solutions, namely to force sparse matrices. Our experiment relies on the R package named NMF and the Brunet algorithm [5].

Once $\hat{\mathbf{Q}}$ is obtained, then the values of $\hat{\mathbf{S}}$ can be computed through linear regression. Starting with the overdetermined system of linear equations:

$$\mathbf{R} \;=\; \hat{\mathbf{Q}}\hat{\mathbf{S}} \tag{4}$$

which has the same form as the more familiar $\mathbf{y} = \mathbf{X}\beta$ (except that $\mathbf{y}$ and β are generally vectors instead of matrices), it follows that the linear least squares estimate is given by:

$$\hat{\mathbf{S}} \;=\; (\hat{\mathbf{Q}}^T\hat{\mathbf{Q}})^{-1}\hat{\mathbf{Q}}^T\mathbf{R} \tag{5}$$

Equation (5) represents a linear regression solution which minimizes the residual errors ($||\mathbf{R} - \hat{\mathbf{Q}}\hat{\mathbf{S}}||$).

3.3 Prediction Accuracy and the Number of Skills

We would expect the model with the correct number of skills to perform the best, and models with fewer skills to underperform because they lack the correct number of latent skills to reflect the response patterns. Models with greater number of skills than required should match the performance of the correct number model, since they have more representative power than needed, but they run higher risk of over-fitting the data and could therefore potentially show lower accuracy in a cross-validation. However, the skills matrix $\hat{\mathbf{S}}$ obtained through equation (5) on the test data could also result in over-fitting that will *increase* accuracy this time. We return to this issue in the discussion.

We use the same simulated data as described for the SVD method in section 2.1, where six skills are used to generate data according to the Q-matrix of figure 1. For this experiment, we only report the condition of guess=0.2 and slip=0.1.

Figure 3 shows the percentage of correct predictions of the models as a function of the number of skills. Given that predictions are $\{0, 1\}$, the percentage can be computed as $||\mathbf{R} - \hat{\mathbf{Q}}\hat{\mathbf{S}}||/mn$, where m and n are the number of rows and columns of $\mathbf{R}$.

The results confirm the conjectures above: the predictive accuracy increases until the underlying number of skills is reached, and it almost stabilizes thereafter. Over-fitting of $\hat{\mathbf{S}}$ with the test data is apparently not substantial.

It is interesting to note that the accuracy increments of figure 3 are relatively constant between each skill up to 6. This

[2] Note that computing $\hat{\mathbf{S}}$ from the test data raises the issue of over-fitting, which would keep the accuracy growing with the number of skills regardless of the “real” number of skills. However, this issue is mitigated by using independent learning data for $\hat{\mathbf{Q}}$, without which, we empirically observed, the results would deceive us: in our experiments using both $\hat{\mathbf{S}}$ and $\hat{\mathbf{Q}}$ from NMF while increasing the rank of the factorization (number of skills), ends up increasing prediction accuracy even after we reach beyond the “real” number of skills. This can reasonably be attributed to over-fitting.

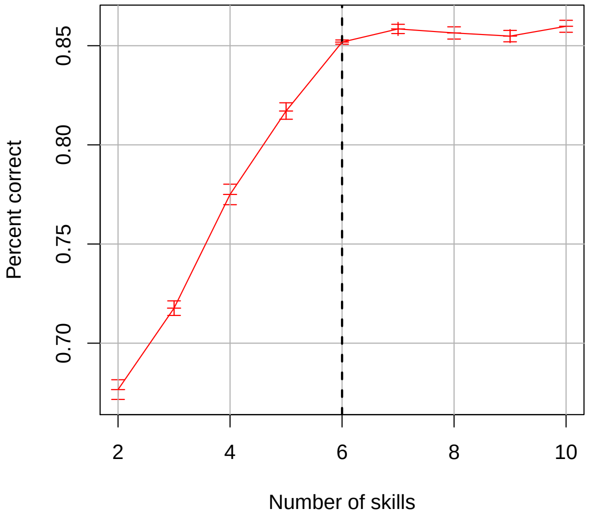

Figure 3: Precision of student results predictions from estimated skill matrix (equation (5)). Error bars are the standard error of the accuracy curves. Experiment is done with simulated data with 6 skills and *slip* and *guess* values of 0.1 and 0.2 respectively.

is also what we would expect since every skill in the underlying Q-matrix has an equivalent weight to all others. We expect that differences in increments indicate differences in the weights of the skills. This could either stem from the structure of the Q-matrix (for e.g., more items can depend on one skill than on another), or on the criticality of the skill over its item outcome.

4. APPLICATION OF THE METHODS ON REAL DATA FROM FRACTION ALGEBRA

Simulated data reveals that both the SVD and wrapper methods provide effective means to identify the number of latent skills. Are these means as effective in identifying skills with real data? This can depend on a number of factors. One factor is the degree to which a skill is determinant to the success of an item. General high level skills can only add to the chances of success, they are not decisive. More specific skills can be decisive, but there may be alternative skills that also account for an item success (e.g. a different method of solving a problem). Finally, noise from slips and guesses will undermine the ability of any method that attempts to identify the number of latent skills.

Therefore, an answer to the above question, i.e. whether we can identify the number of latent skills, is only valid within a given context, where the factors mentioned above take on a particular combination. So any conclusion will have to take into account this limitation in its generalization.

We investigate the question with data from Vomlel [10] on fraction algebra problems. This data set is composed of 20 question items and answers from 148 students. A Bayesian Network linking items to skills was defined by experts for the 20 items. It can readily be transformed into the Q-matrix shown in figure 4.

Items

Skills	1	2	3	4	7	8	9	10	11	12	13	14	15	16	17	19	20
1 CL	1	1									1						
2 CMI						1	1					1			1		1
3 CIM								1	1								
4 CD			1	1	1									1		1	
5 MT										1	1	1					
6 AD													1	1	1		
7 SB																1	1

Figure 4: Conjunctive Q-matrix of Fraction Algebra data composed of 7 skills and 17 items. Item numbers refer to the original data items.

This Q-matrix is a subset of the whole Q-matrix from the Bayesian Network in Vomlel's study. It was chosen based on four fundamental skills of fraction algebra :

1 CL: cancelling out
2 CIM: conversion to mixed numbers
3 CMI: conversion to proper fractions
4 CD: finding common denominator

A total of 15 items are involved those skills. Because some items involved other skills, 3 more skills are added through conjunction, for a total of 7 skills:

5 AD: addition
6 SB: subtraction
7 MT: multiplication

And 2 more items involving these added skills are also added, for a total of 17 items. Six out of the 17 items involve a conjunction of 2 skills, whereas all other items are single skill.

Note that contrary to the synthetic data, skills are not expected to have equal weight in the prediction results, as some are only involved in two items, whereas others are involved in five items.

The SVD and wrapper methods are applied to the data in an attempt to derive the number of underlying skills. For the SVD method, the factorization is conducted on the full data set since this method does not rely on a cross validation process. For the wrapper method, the data is split in half for training, half for testing. Both approaches follow the methodology described in sections 2 and 3.

4.1 SVD method

Results of applying the SVD method to the fraction algebra data is reported in figure 5. Apart from the usual steep slope from singular value 1 to 2, there is no clear indication of the number of skills in this figure when we look at a change of slope as we had with the simulated data experiment. However, the *random* and *real* curves meet at singular value 2, which, according to the results from simulated data, would suggest that the number of latent skills is 2. However, this not consistent with the expert Q-matrix. It is also counter-intuitive since we would expect that more than two skills in fraction algebra problems would cover the skills described above.

We could also conclude that there is a continuum of skills, and/or that the data is too noisy to show any effect of skills.

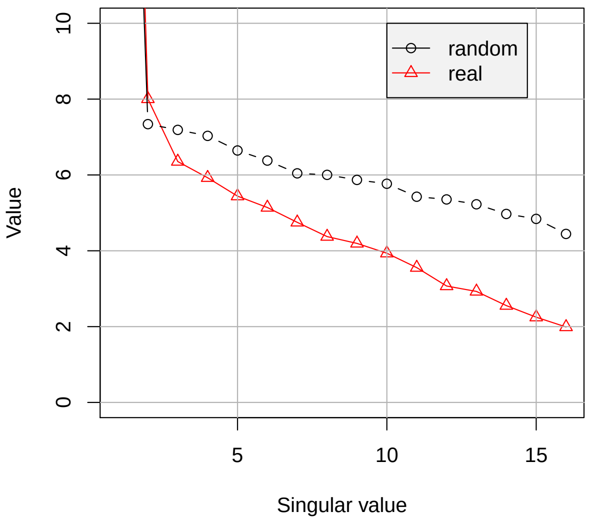

Figure 5: SVD results over fraction algebra data. The random and real curve at skill 1 are not shown but they are respectively 30 and 35.

Let us turn to the wrapper method before speculating any further on these unexpected results.

4.2 Wrapper method

For the wrapper method, the data set is divided into two random samples of half the size of the original 148 students. One half is used for deriving the Q-matrix and the other in deriving the skills matrix, $\hat{\mathbf{S}}$, and measuring the accuracy of the predictions. This procedure is the same as the one used for the simulated data. As we explain below, a large number of folds (50) have to be run in order to obtain stable results.
Figure 6 reports the results of the wrapper method. We observe a sharp drop after skill 2, which suggests that a peak was reached at that point[3]. In that respect, it confirms the 2-skill findings of the SVD method.
However, we also observe a steady increase of accuracy starting from 3 skills, up to 8 skills, and a gradual decrease of skill contribution to performance starting from 4 skills. Except for the unexpected drop after 2 skills, this finding is close to the 7 skills defined by experts. And the fact that some skills have a greater weight on the performance is also consistent with the gradual decrease of contribution up to 8 skills.
Concerning the decrease after 9 skills, this can be explained by over-fittins in the NMF Q-matrix induction ($\hat{\mathbf{Q}}$) with the training data. In simulated data, the sample size was apparently large enough to shield the results from the over-fitting issue, but the smaller sample size of the real data may raise this issue here. Moreover, as the number of latent factors approaches the number of items in the data (17), the over-fitting issue becomes even more significant.
Drawing conclusions from this experiment with real data is obviously hard. Both the SVD and the wrapper methods seem to suggest that 2 skills would are plausible, but the wrapper method also points to an 8 skills set that is more consistent with the expert Q-matrix.

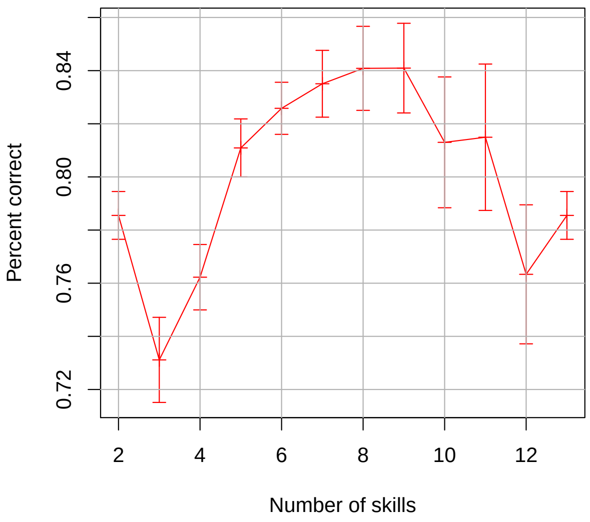

Figure 6: Wrapper method applied to the fraction algebra data set. The error bars represent the standard error of 50 folds results.

5. DISCUSSION

Both the SVD and the wrapper methods provide strong cues of the number of underlying skills with simulated student test data. However, for the Vomlel data set, both methods yield results that are much more ambiguous. Instead of the 7 skills that were identified by experts over the 17 items set, the SVD method suggests only 2 skills if we rely on the intersection with the random data curve, and no clear number if we look for a change of slope after skill 2. The wrapper method shows data that is also consistent with 2 skills to the extent that a drop of accuracy is observed at 3 skills, but a rise of accuracy up to 8 skill draws an interpretation closer to the experts' 7 skills set.
An important difference between the SVD and the wrapper methods has to do with the independence of skills. For SVD, orthogonality of the singular matrices $\mathbf{U}$ and $\mathbf{V}$ in equation (1) forces latent factors to be independent. NMF does not require latent factors to be independent. The orthogonality constraint of may limit the application of the SVD method with respect to real skills and might explain some of the difference between the two methods. The skills from the synthetic data of the first experiment were independent and the Q-matrix had an homogeneous pattern for each skill, and therefore the effect of dependence between skills could not come into play.
Obviously, the study calls for more investigations. The findings from one set of data from the real world may be

[3] The implementation of the method does not allow a computation of the accuracy for a single skill, but we can reasonably assume that a single skill model would perform worst than a 2-skills model.

highly different from another set. More studies should be conducted to assess the generality of the findings. Other investigations are called for to find ways to improve these methods and to better understand their limits when faced with real data. In particular, we need to know at which level of noise from guess and slip factors do the methods break down, and what is the ratio of latent skills to data set size that is critical to avoid over-fitting of the wrapper method. One improvement that can be brought to the wrapper method is to use a cross validation to derive the skills matrix. This would require the use of two sets of items, one for testing and one for assessing the student's skills. This comes at the cost of a greater number of items, but it avoids the problem of over-fitting that leads to accuracy increases.

6. ACKNOWLEDGEMENTS

This project was supported by funding from the MATI institute (www.matimtl.org) and by Canada's NSERC discovery program.

7. REFERENCES

[1] T. Barnes. The Q-matrix method: Mining student response data for knowledge. In *AAAI'2005 Workshop on Educational Data Mining*, pages technical report WS–05–02, 2005.

[2] A. T. Corbett and J. R. Anderson. Knowledge tracing: Modeling the acquisition of procedural knowledge. *User Modeling and User-Adapted Interaction*, 4(4):253–278, 1995.

[3] M. C. Desmarais. Conditions for effectively deriving a q-matrix from data with non-negative matrix factorization. In C. Conati, S. Ventura, T. Calders, and M. Pechenizkiy, editors, *4th International Conference on Educational Data Mining, EDM 2011*, pages 41–50, Eindhoven, Netherlands, June 6–8 2011.

[4] M. C. Desmarais, B. Beheshti, and R. Naceur. Item to skills mapping: deriving a conjunctive q-matrix from data. In *11th Conference on Intelligent Tutoring Systems, ITS 2012, Chania, Greece, 14–18 June 2012.*

[5] R. Gaujoux and C. Seoighe. A flexible R package for nonnegative matrix factorization. *BMC Bioinformatics*, 11(367):http://www.biomedcentral.com/1471--2105/11/367, 2010.

[6] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182, 2003.

[7] A. Mitrovic, M. Mayo, P. Suraweera, and B. Martin. Constraint-based tutors: A success story. In L. Monostori, J. Váncza, and M. Ali, editors, *IEA/AIE*, volume 2070 of *Lecture Notes in Computer Science*, pages 931–940. Springer, 2001.

[8] P. I. Pavlik, H. Cen, and K. R. Koedinger. Performance factors analysis –a new alternative to knowledge tracing. In *Proceeding of the 2009 conference on Artificial Intelligence in Education*, pages 531–538, Amsterdam, The Netherlands, 2009. IOS Press.

[9] P. Suraweera, A. Mitrovic, and B. Martin. A knowledge acquisition system for constraint-based intelligent tutoring systems. In C. K. Looi, G. McCalla, B. Bredeweg, and J. Breuker, editors, *Proceedings of the 12th International Conference on Artificial Intelligence in Education, AEID'2005*, pages 638–645, Amsterdam, July 18–22 2005. IOS Press.

[10] J. Vomlel. Bayesian networks in educational testing. *International Journal of Uncertainty, Fuzziness and Knowledge Based Systems*, 12:83–100, 2004.

[11] T. Winters, C. R. Shelton, and T. Payne. Investigating generative factors of score matrices. In R. Luckin, K. R. Koedinger, and J. E. Greer, editors, *Artificial Intelligence in Education, Building Technology Rich Learning Contexts That Work, Proceedings of the 13th International Conference on Artificial Intelligence in Education, AIED 2007, July 9-13, 2007, Los Angeles, California, USA*, volume 158 of *Frontiers in Artificial Intelligence and Applications*, pages 479–486. IOS Press, 2007.