Pattern Recognition 138 (2023) 109285

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Wse-MF: A weighting-based student exercise matrix factorization model

Xia Sun[a], Bo Li[a], Richard Sutcliffe[a,c,*], Zhizezhang Gao[b], Wenying Kang[a], Jun Feng[a]

[a] School of Information Science and Technology, Northwest University, Xi'an 710127, China
[b] School of Statistics, Beijing Normal University, Beijing 100875, China
[c] School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe Park, Colchester, CO43SQ, UK

ARTICLE INFO

Article history:
Received 26 November 2021
Revised 6 December 2022
Accepted 27 December 2022
Available online 11 January 2023

Keywords:
Educational data mining
Personalized exercise prediction
Matrix factorization

ABSTRACT

Students who have been taught new ideas need to develop their skills by carrying out further work in their own time. This often consists of a series of exercises which must be completed. While students can choose exercises themselves from online sources, they will learn more quickly and easily if the exercises are specifically tailored to their needs. A good teacher will always aim to do this, but with the large groups of students who typically take advantage of open online courses, it may not be possible. Exercise prediction, working with large-scale matrix data, is a better way to address this challenge, and a key stage within such prediction is to calculate the probability that a student will answer a given question correctly. Therefore, this paper presents a novel approach called Weighting-based Student Exercise Matrix Factorization (Wse-MF) which combines student learning ability and exercise difficulty as prior weights. In order to learn how to complete the matrix, we apply an iterative optimization method that makes the approach practical for large-scale educational deployment. Compared with eight models in cognitive diagnosis and matrix factorization, our research results suggest that Wse-MF significantly outperforms the state-of-the-art on a range of real-world datasets in both prediction quality and time complexity. Moreover, we find that there is an optimal value of the latent factor K (the inner dimension of the factorization) for each dataset, which is related to the relationship between skills and exercises in that dataset. Similarly, the optimal value of hyperparameter c_0 is linked to the ratio between exercises and students. Taken as a whole, we demonstrate improvements to matrix factorization within the context of educational data.

1. Introduction

With the rapid informatization of education, student learning data obtained from online education platforms [1] provides many opportunities for developing new means of assessment [2] and new forms of intelligent tutoring [3]. Much of this information is in the form of matrices.

Students typically learn by practicing in their own time what they have learned within formal lectures, often delivered online. Large banks of online exercises can contribute by making such material available to students. However, students can learn more quickly and easily if exercises are chosen for them which specifically meet their needs. Because this is time-consuming for teachers to do, it is very valuable to automate the process by inferring the appropriate exercises from large-scale student-exercise datasets.

An important step within exercise recommendation is to estimate how likely it is that a student will be able to answer an exercise. If this probability is too low, then the exercise is probably too difficult and will discourage the student. On the other hand, if the probability is too high, then the exercise is too easy and will have insufficient instructional value to be worthwhile.

With improvements in computing capability, approaches based on deep learning have made rapid progress and this has resulted in data mining successes within educational research. For example, researchers have proposed Deep Knowledge Tracing (DKT) [4], Dynamic Key-Value Memory Networks (DKVMN) [5], Exercise-Enhanced Recurrent Neural Networks (EERNN) [6] and Exercise-aware Knowledge Tracing (EKT) [7] for predicting student performance [8]. However, since knowledge tracing builds a model based on the answer records of all students as a whole, it can only give

* Corresponding author.
E-mail addresses: raindy@nwu.edu.cn (X. Sun), rsutcl@nwu.edu.cn (R. Sutcliffe), kwy@stumail.nwu.edu.cn (W. Kang), fengjun@nwu.edu.cn (J. Feng).

https://doi.org/10.1016/j.patcog.2022.109285

an overall prediction for all students, rather than a personal one for individual students, making it inapplicable to customized exercise prediction.

Regarding individualized exercise prediction, the Deterministic Inputs Noisy AND Gate (DINA) model [9], as one of the Cognitive Diagnosis Methods (CDMs) [10–12], is one of the most popular prediction techniques in the education community, and its psychometric models have immense potential to provide rich and relevant information for instruction and learning. However, in real life, we need to pay attention to the high time complexity of DINA. The existing method focuses on explicit modeling of the 'slip' and 'guess' latent factors for each exercise, usually assumed $O(2^L)$ in time complexity where L is the number of skills; this can make it unsuitable for applications involving large datasets where better performance can only be achieved by using a large number of parameters.

In order to address the above problems, the Matrix Factorization (MF) model [13] for exercise prediction was introduced. MF is one of the commonly employed methods in pattern recognition, and can be used to predict the scores of students when a student-exercise performance matrix has been populated with some observed scores. MF, computed via Stochastic Gradient Descent (SGD), is one of the most prominent approaches; it is easier to train and requires no additional attributes. However, MF assigns binary weights (zero or one) to the predicted data, rather than real-valued weights, making it suboptimal. More recently, Devooght's study [14] confirms that using priors on the predicted values leads to an overall improvement in MF quality.

Inspired by the above work, we aim to predict the student score from the joint perspective of both students and exercises. In particular, students have a single underlying characteristic, namely learning ability, and there is also a parameter representation for the exercises, called difficulty. Whether a student can solve an exercise depends on the difficulty of the exercise and the learning ability of the student, and the combination of the two attributes can help achieve personalized learning. Therefore, we propose a novel MF model called Weighting-based Student Exercise Matrix Factorization (Wse-MF), which focuses on setting an explicit prior on predicted data, as well as incorporating a student-exercise weighting strategy, by taking into account the student learning ability and exercise difficulty from observed data. Considering computational complexity due to complex weights, we apply a new optimization, Student-Exercise element-wise Alternating Least Squares (SE-ALS), which is based on the existing eALS technique [15]. SE-ALS is more efficient than the commonly-used SGD approach. SGD uses a stochastic method for controlling the possible losses, and relies on a good learning rate, while SE-ALS uses coordinate descent to carry out optimization, updating a particular parameter at each step, which does not rely on the learning rate anymore. We summarize our key contributions as follows:

- We define a student's learning ability and exercise difficulty, and use them as a student-exercise weighting strategy. Therefore, our method is able to achieve individuation.
- We incorporate the student-exercise weighting strategy into the loss function and apply it to the predicted data, effectively improving the MF model for exercise prediction.
- We apply a new optimization called SE-ALS, based on student and exercise matrices, so that our Wse-MF is more efficient than DINA. Hence, we can train quickly and with low time complexity.
- We conduct experiments on three real-world datasets and show that our method consistently outperforms state-of-the-art CDM methods. Moreover, the proposed model has two important parameters, the latent factor K (the inner dimension of the factorization) and the weight of the predicted data, c_0. We show experimentally that the optimal value of K is related to the relationship between skills and exercises within a dataset, while c_0 is approximately equal to the ratio between the numbers of exercises and students.
- Overall, the work is a contribution to research on matrix factorization, demonstrated within the education field.

2. Related work

In this section, we introduce existing exercise prediction methods from three perspectives: knowledge tracing models, cognitive diagnosis models and matrix factorization models.

2.1. Knowledge tracing models

Knowledge tracing has been increasingly influenced in the past two decades by developments in the fields of pedagogy and computer science. In 1994, Bayesian Knowledge Tracing (BKT) [16] was the first application of knowledge tracking to the prediction of student performance scores, and Individualized Bayesian Knowledge Tracing [17] was developed from this by splitting BKT parameters. In 2015, Piech et al. proposed Deep Knowledge Tracing (DKT) [4], showing that tracking student learning and predicting answers greatly improves forecasting accuracy. In 2017, by automatically establishing the relationship between exercises and implicit knowledge points, Zhang et al. proposed a Dynamic Key-Value Memory Network (DKVMN) model [5] for student exercise prediction. This was followed by Su et al. in 2018 [6], who proposed a general Exercise-Enhanced Recurrent Neural Network (EERNN) framework for exploring both students records and the text content of exercises. In 2019, Liu et al. [7] extended EERNN to form the Exercise-aware Knowledge Tracing (EKT) framework, in order to quantify how much an exercise can affect the ability of a student to develop multiple skills during the learning process. In 2021, Zia et al. [18] proposed an improvement of BKT in which features are firstly engineered from the basic features, to derive hidden features and hence reveal information that was not apparent. Finally, Song et al. [19] proposed a Joint graph convolutional network deep Knowledge Tracing (JKT) framework to establish connections between exercises under cross-concepts and to capture high-level semantic information.

2.2. Cognitive diagnosis models

In educational psychology, Item Response Theory (IRT) is a modern psychometric theory which considers that students' responses to exercises have a special relationship with their potential characteristics. One of the underlying assumptions of IRT is that students all use the same skill to respond to each exercise. When exercises do not satisfy this unidimensionality assumption, Multidimensional Item Response Theory (MIRT) [20] enables the interaction of students with exercises to be modeled in a way which is capable of discriminating between levels of student ability, and levels of student proficiency with respect to these abilities. Cheng et al. [21] consider that traditional IRT only exploits student response results, and has difficulties in fully utilizing the semantics of question texts. Therefore, they first use a vector to represent student proficiency with regard to knowledge concepts, and represent question texts and skills by dense embeddings. Then, they use deep learning to determine the parameters of students and of questions, by exploiting question texts and the relationship between those texts and skills. Xu et al. [22] proposed two new probabilistic graph models to improve the accuracy of assessments based on the well-accepted cognitive diagnosis technique in MOOCs.

Table 1
An example Q-matrix.

Exercise	$Skill_1$	$Skill_2$	$Skill_3$	$Skill_4$	$Skill_5$
$exercise_1$	1	0	0	1	0
$exercise_2$	0	0	1	1	1
$exercise_3$	0	1	0	0	0

As mentioned earlier, the Cognitive Diagnosis Method (CDM) [11] is a technique based on IRT to predict the future performance of a particular student by an analysis of their previous work. With the CDM, students are characterized by their proficiency in specific skills. A Q-matrix [23] is previously determined by experts in the subject and specifies the skills which are required for each exercise. Table 1 is an example of a Q matrix, where 1 indicates that the exercise contains the skill, and 0 indicates that the exercise does not contain the skill. CDMs can be grouped into two classes, depending on whether the information they use is discrete or continuous. Among them, the DINA model [9], introduced above, uses a vector of latent binary variables, that is, it can only assign students their degree of mastery of a set of skills using discrete values. To address this limitation, a popular solution is probabilistic modeling of students' skills in order to simulate them as continuous values between 0 and 1 [24]. Liu et al. [10] combined fuzzy set theory and educational hypotheses in order to model a student; their fuzzy Cognitive Diagnosis Framework (fuzzyCDF) was applicable to both objective and subjective exercises [25].

2.3. Matrix factorization models

Matrix Factorization (MF) analysis has attracted more and more attention from researchers because it outperforms the state-of-the-art Collaborative Filtering (CF) technique. For example, He et al. [26] enhanced the flexibility of MF by allowing the use of nonuniform weights on missing data. Zhong et al. [27] proposed a Constrained Matrix Factorization which integrates the course average score into the objective function so as to make up the prediction deviation caused by the unbalanced course selection rate. Most traditional NMF-based algorithms are sensitive to noisy data; correntropy based semi-supervised NMF (CSNMF) [28] aims to solve this issue. In addition, Hu et al. [29] proposed a Feature Nonlinear Transformation Non-Negative Matrix Factorization with Kullback-Leibler Divergence (FNTNMF-KLD) for extracting the nonlinear features of a matrix in standard NMF. Jin et al. [30] described two new sparse matrix factorization methods, each with $L_{2,1}$ norm to explicitly force the row sparseness of the factor matrix. This approach can attain comparable performance to the deep learning-based matrix completion methods.

2.4. Analysis

In spite of the successes achieved within the previous work discussed above, the existing methods still display some limitations. For instance, although deep learning has been well developed in the field of exercises, the DKT and DKVMN models not only need to adjust many hyperparameters manually, but more importantly, the model parameters obtained from the training are overall predictions for all students, so they cannot predict suitable exercises for an individual student.

Specifically, in order to model the *slip* and *guess* latent factors, cognitive diagnosis usually assumes a time complexity of $O(2^L)$ where L is the number of skills. In consequence, adjusting the *slip* and *guess* for N exercises takes time $O(N \times 2^L)$ where N is the number of exercises. Thus, for one iteration which is adjusting all the parameters of the model, the time complexity is $O(M \times N \times 2^L)$

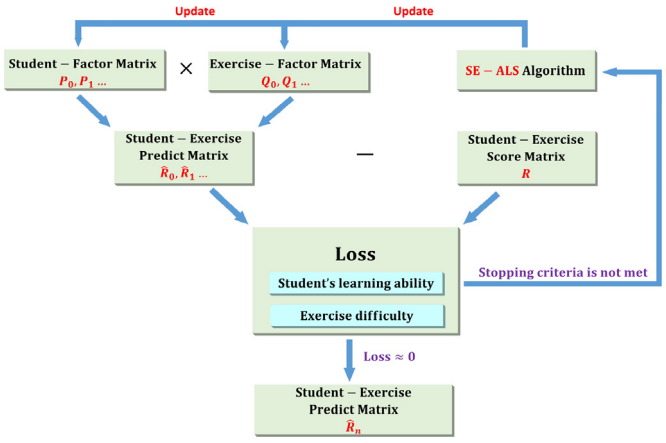


Fig. 1. Wse-MF Framework Overview: Schema for training exercise MF models.

where M is the number of students. Clearly, this means that the algorithm is not very practical when applied to large datasets involving huge numbers of students, exercises and performance records.

Furthermore, MF requires time to recursively compute and update the matrix, especially when the number of factors is large, and it ignores the student's knowledge state. So, by using data-driven methods to mine attributes of students and exercises from performance data, we want to apply an efficient method to solve personalized exercise prediction at low time complexity.

3. Proposed method

We will now introduce our general loss function for training exercise MF models, which is called Wse-MF (Fig. 1). In essence, we first randomly generate two matrices, namely the Student-Factor Matrix ($\boldsymbol{P_0}$) and the Exercise-Factor Matrix ($\boldsymbol{Q_0}$), and multiply the two to obtain the Student-Exercise Prediction Matrix ($\hat{\boldsymbol{R}}_{\boldsymbol{0}}$). Secondly we calculate the loss with the Student-Exercise Score Matrix ($\boldsymbol{R}$), taking into account the student's learning ability and the exercise difficulty. Then, we apply the SE-ALS algorithm [15] to optimize the objective function by setting a cache to achieve a significant increase in speed. It updates the Student-Factor Matrix ($\boldsymbol{P_1} \ldots$) and the Exercise-Factor Matrix ($\boldsymbol{Q_1} \ldots$) to obtain the Student-Exercise Prediction Matrix ($\hat{\boldsymbol{R}}_{\boldsymbol{1}} \ldots$). Finally, when the loss approaches 0, we obtain the new Student-Exercise Prediction Matrix ($\hat{\boldsymbol{R}}_{\boldsymbol{n}}$).

3.1. Student-exercise weighting strategy

All other factors being equal, students who have a high learning ability are more likely to answer an exercise correctly; similarly, exercises which have a greater difficulty are more likely to be done wrongly. To account for these effects, we evaluate student learning ability and exercise difficulty from a statistical perspective.

Student's learning ability: The CDM model mainly uses the Student-Exercise Score Matrix and the Exercise-Skill Correlation Q-matrix to obtain the student's mastery of the skills, so as to realize the exercise prediction. The DINA model, on the other hand, which is a kind of CDM, introduces exercise parameters *slip* and *guess*, as already mentioned: Students who possess all the required skills for an exercise can nevertheless make a slip and answer incorrectly; conversely, students who lack at least one of the required skills can still guess and hence answer correctly. DINA adopts the Expectation Maximization algorithm to obtain the two parameter estimates based on exercises and the students binary skill state [9]. The *slip* and *guess* update rules are as follows:

$$\boldsymbol{s_i} = \frac{I_{ij}^1 - R_{ij}^1}{I_{ij}^1} \tag{1}$$

$$\boldsymbol{g_i} = \frac{R_{ij}^0}{I_{ij}^0} \tag{2}$$

where I_{ij}^1 is the number of students who have all the required skills for exercise i and R_{ij}^1 is the number of students among I_{ij}^1 correctly answering exercise i. Conversely, I_{ij}^0 is the number of students lacking at least one of the required skills for exercise i and R_{ij}^0 is the number of students among I_{ij}^0 correctly answering exercise i.

However, the method of Zhu et al. [24] is an improvement relative to the DINA model because the skill state is no longer Boolean, but is now a continuous value between 0 and 1. As a result, the posterior probability $\hat{\partial}_{uj}$ that will now be obtained is as follows:

$$\begin{aligned}\hat{\partial}_{uj} &= \frac{\sum_{\partial_{uj}=1} P\left(\boldsymbol{\partial_j}|\boldsymbol{r_u}, \boldsymbol{s_i}, \boldsymbol{g_i}\right)}{\sum_{x=1}^{J} P\left(\boldsymbol{\partial_x}|\boldsymbol{r_u}, \boldsymbol{s_i}, \boldsymbol{g_i}\right)} \\ &= \frac{\sum_{\partial_{uj}=1} L\left(\boldsymbol{r_u}|\boldsymbol{\partial_j}, \boldsymbol{s_i}, \boldsymbol{g_i}\right) P(\boldsymbol{\partial_x})}{\sum_{x=1}^{J} P\left(\boldsymbol{\partial_x}|\boldsymbol{r_u}, \boldsymbol{s_i}, \boldsymbol{g_i}\right)} \\ &= \frac{\sum_{\partial_{uj}=1} \prod_{i=1}^{I} L\left(r_{ui}|\boldsymbol{\partial_j}, \boldsymbol{s_i}, \boldsymbol{g_i}\right) P(\boldsymbol{\partial_j})}{\sum_{x=1}^{J} P\left(\boldsymbol{\partial_x}|\boldsymbol{r_u}, \boldsymbol{s_i}, \boldsymbol{g_i}\right)}\end{aligned} \tag{3}$$

where $\boldsymbol{s_i}$ and $\boldsymbol{g_i}$ represent the *slip* and *guess* parameters of exercise i, and $\boldsymbol{r_u}$ represents student u's answer records. The numerator represents the state accumulation after the student u answers the exercise containing skill j, and the denominator represents the all-state accumulation based on student u's answer records.

A students average mastery over all exercises is now established by the probability of the student being at each skill state [24]:

$$l_{ui} = \sqrt[\sum_{j=1}^{J} Q_{ij}]{\prod_{j=1}^{J} In_{uij}} \tag{4}$$

where the In_{uij} is

$$In_{uij} = \begin{cases} 1, & Q_{ij} = 0 \\ \hat{\partial}_{uj} \times Q_{ij}, & Q_{ij} = 1 \end{cases} \tag{5}$$

where Q_{ij} is an element of the Q-matrix, $Q_{ij} = 1$ means that exercise i includes skill j, and $Q_{ij} = 0$ means that exercise i does not include skill j.

Finally, we use l_{ui} as that student's learning ability.

Exercise difficulty: Because exercises differ in the number and type of skills required, obtaining low-dimensional vectors of the exercises through matrix factorization also loses some information, including exercise difficulty. We use the method of Minn et al. [31] to find the differences between *success* and *failure* ratios on each skill, based on a student's previous performance, and transform each one into a value for the student u as follows:

$$\boldsymbol{success}\left(x_{uj}\right) = \sum_{i=1}^{N} \frac{\left(x_{uji} == 1\right)}{N_{uj}} \tag{6}$$

$$\boldsymbol{failure}\left(x_{uj}\right) = \sum_{i=1}^{N} \frac{\left(x_{uji}! = 1\right)}{N_{uj}} \tag{7}$$

$$d_{ui} = Q_{ij} * \left(\boldsymbol{failure}\left(x_{uj}\right) - \boldsymbol{success}\left(x_{uj}\right)\right) \tag{8}$$

where $\boldsymbol{success}\left(x_{uj}\right)$ and $\boldsymbol{failure}\left(x_{uj}\right)$ are the ratios of exercises requiring skill x_j being answered correctly or incorrectly, and N_{uj} is the total number of times student u has practiced skill x_j. x_{uji} means student u answers the exercise i including skill x_j correctly. Q_{ij} is the relationship between exercise i and skill x_j, and d_{ui} is the difficulty of exercise i for student u, which is transformed into the difficulty of skill x_j for student u through the Q-matrix. From Eq. (11) we can see that if $\boldsymbol{failure}\left(x_{uj}\right)$ is greater than $\boldsymbol{success}\left(x_{uj}\right)$, it indicates that students have a low degree of mastery of the skill x_j; then, through the Q-matrix mapping, exercises containing this skill x_j will be assigned a high difficulty.

3.2. Student-exercise weighting loss

In Section 3.1, we proposed the Student-Exercise Weighting Strategy, which defines the student's learning ability and exercise difficulty. In order to achieve accurate and personalized exercise prediction, we add it to the loss function as the weight of the predicted data. In this section, we first introduce the MF method, illustrating the issue of inefficiency which arises from the conventional loss function used. Then, we describe the Wse-MF model after adding the Student-Exercise Weighting Strategy.

3.2.1. MF method for prediction

The starting point of the MF method is a matrix $\mathbf{R} \in \mathbf{R}_{\mathbf{M}\times\mathbf{N}}$, where M is the number of students and N is the number of exercises. If student u carries out exercise i, then the corresponding element in the matrix is one. Otherwise it is zero. In our work, R is represented in the form of a student-exercise matrix. Matrix R is now factorized into $\mathbf{P} \in \mathbf{R}_{\mathbf{M}\times\mathbf{K}}$ and $\mathbf{Q} \in \mathbf{R}_{\mathbf{N}\times\mathbf{K}}$, in so doing generalizing the mapping between students and exercises. K is thus the inner dimension of the factorization and we refer to this as the *latent factor*. Prediction of exercise i for student u is given by the rule:

$$\hat{r}_{ui} = \boldsymbol{p_u}\boldsymbol{q_i}^T \tag{9}$$

More recent works learn factor vectors directly on known interaction data through a suitable objective function which minimizes prediction error. The proposed objective functions are usually regularized in order to avoid overfitting [32]. Typically, gradient descent is applied to minimize the objective function:

$$L = \sum_{i=1}^{N}\sum_{u=1}^{M}\left(r_{ui} - \hat{r}_{ui}\right)^2 + \lambda\left(\sum_{u=1}^{M}\|\boldsymbol{p_u}\|^2 + \sum_{i=1}^{N}\|\boldsymbol{q_i}\|^2\right) \tag{10}$$

The constant λ controls the extent of regularization, usually determined by cross validation. Minimization is typically performed by either stochastic gradient descent or alternating least squares.

3.2.2. Wse-MF model

It is well-known that the matrix R which models student interaction is usually very large and also very sparse, meaning that it is too time-consuming to predict the frequent nulls. To this end, we overcome the limitation of uniform weights in matrix factorization models and adopt our Student-Exercise Weighting as the value to be predicted. By expanding the first term of Eq. (10), our loss function is as follows:

$$\begin{aligned}L = &\sum_{(u,i)\in\boldsymbol{R}}\left(r_{ui} - \hat{r}_{ui}\right)^2 + \sum_{u=1}^{M}\sum_{i\notin\boldsymbol{r_u}} c_0 \frac{d_{ui}}{l_{ui}}\hat{r}_{ui} \\ &+\lambda\left(\sum_{u=1}^{M}\|\boldsymbol{p_u}\|^2 + \sum_{i=1}^{N}\|\boldsymbol{q_i}\|^2\right)\end{aligned} \tag{11}$$

where c_0 represents the weight of the predicted data, considering the number of students and exercises, and $\frac{d_{ui}}{l_{ui}}$ denotes the weight with which the exercise i is predicted by student u (we will demonstrate the effectiveness of c_0 and the Student-Exercise Weighting Strategy in Sections 4.4.2 and 4.4.3). As can be seen, the first term is concerned with the prediction error of the relevant entries. The second term expresses the loss of predicted data in terms of d_{ui} and l_{ui}, which can take different values independently: When d_{ui} is high and l_{ui} is low, that means the exercise i is difficult and the student u is assigned low learning ability; we hope

the predicted value is close to 0, because otherwise $\frac{d_{ui}}{l_{ui}}$ will increase the cost function. In contrast, when d_{ui} is low and l_{ui} is high, that means the exercise i is easy and the student u is assigned high learning ability; now we hope the predicted value is close to 1 and makes the loss become close to 0. Furthermore, there are two additional cases: When d_{ui} and l_{ui} are both low, or when they are both high, they only depend on the first term of the loss function. So, our loss function can better deal with the error problem of the first two cases. The last term is to solve the overfitting problem caused by MF. It is not directly minimizing the loss function, but is adding a normalization factor.

3.3. Inference

Our goal is to find a vector p_u for each student u, and a vector q_i for each exercise i. These vectors will be known as the student factors and the exercise factors, respectively. This is the MF technique which is popular for SGD when using explicit feedback, with two important distinctions: 1) Our cost function contains $M \times N$ terms, where M is the number of students and N is the number of exercises. For typical datasets $M \times N$ can easily reach a few billion. This huge number of terms prevents the use of most direct optimization techniques such as SGD. 2) SGD needs a careful and time-consuming search for the learning rate. In contrast to SGD, ALS requires no learning rate to be determined. What is more, it can be observed that when either the student factors or the exercise factors are fixed, the cost function becomes quadratic so its global minimum can be readily computed. Based on the above analysis, we focus on the SE-ALS algorithm [15] in our MF model. We optimize one parameter with the other fixed, then optimize the other parameter with the first one fixed, repeating the process iteratively until a joint optimum is reached. First, we write the p_{uf} update rule as Eq. (12) by separating the observed data part:

$$p_{uf} = \frac{\sum_{i=1}^{N}\left(r_{ui} - \hat{r}_{ui}\right)q_{if} - \sum_{i \notin \boldsymbol{r_u}} c_0 \frac{d_{ui}}{l_{ui}} q_{if}\hat{r}_{ui}}{\sum_{i \in \boldsymbol{r_u}} q_{if}^2 + \sum_{i \notin \boldsymbol{r_u}} c_0 \frac{d_{ui}}{l_{ui}} q_{if}^2 + \lambda} \tag{12}$$

Similarly, we can derive the update rule for q_{if}:

$$q_{if} = \frac{\sum_{u=1}^{M}\left(r_{ui} - \hat{r}_{ui}\right)p_{uf} - \Sigma_{u \notin \boldsymbol{r_i}} c_0 \frac{d_{ui}}{l_{ui}} p_{uf}\hat{r}_{ui}}{\sum_{u \in \boldsymbol{r_i}} p_{uf}^2 + \sum_{u \notin \boldsymbol{r_i}} c_0 \frac{d_{ui}}{l_{ui}} p_{uf}^2 + \lambda} \tag{13}$$

The second term of Eq. (11) is more time-consuming because the items are to be predicted. So, the sum on predicted data can be formulated as the difference between the sum on all items and the sum on observed data. In this way, the sum on predicted data disappears from the computations:

$$\sum_{u=1}^{M}\sum_{i \notin \boldsymbol{r_u}} c_0 \frac{d_{ui}}{l_{ui}} \hat{r}_{ui}^2 = \sum_{u=1}^{M} c_0 \frac{d_{ui}}{l_{ui}} \hat{r}_{ui}^2 - \sum_{(u,i) \in \boldsymbol{R}} c_0 \frac{d_{ui}}{l_{ui}} \hat{r}_{ui}^2 \tag{14}$$

So, the second term in the numerator of Eq. (12) is

$$\sum_{i \notin \boldsymbol{r_u}} c_0 \frac{d_{ui}}{l_{ui}} q_{if}\hat{r}_{ui} = \sum_{i=1}^{N} c_0 \frac{d_{ui}}{l_{ui}} q_{if}\hat{r}_{ui} - \sum_{i \in \boldsymbol{r_u}} c_0 \frac{d_{ui}}{l_{ui}} q_{if}\hat{r}_{ui} \tag{15}$$

and the second term in the denominator of Eq. (12) is

$$\sum_{i \notin \boldsymbol{r_u}} c_0 \frac{d_{ui}}{l_{ui}} q_{if}^2 = \sum_{i=1}^{N} c_0 \frac{d_{ui}}{l_{ui}} q_{if}^2 - \sum_{i \in \boldsymbol{r_u}} c_0 \frac{d_{ui}}{l_{ui}} q_{if}^2 \tag{16}$$

The complete Wse-MF Algorithm is shown as Algorithm 1 below.

4. Experiments

In this section, we will conduct experiments with the aim of answering the following questions:

Algorithm 1 Wse-MF model using SE-ALS with M, N, $\mathbf{R}$, K, λ and $\frac{d_{ui}}{l_{ui}}$.

```
Procedure Student-Exercise Weighting Matrix Factorization(M, N, R, K, λ and d_ui/l_ui)
Randomly initialize P and Q;
While stopping criteria is not met do
    for (u, i) ∈ R do r̂_ui ← Eq. (9)
    for u ← 1 to M do //update student factors
        for f ← 1 to K do
            p_uf ← Eq. (12)
        end
    end
    for i ← 1 to N do //update exercise factors
        for f ← 1 to K do
            q_if ← Eq. (13)
        end
    end
end
return P and Q;
end Procedure
```

Table 2
Dataset Summary.

Dataset	Students	Skills	Exercise		Records	Avg. Skills per Exercise
			Obj.	Subj.		
FrcSub	536	8	20	0	10,720	2.8
Math1	4209	11	15	5	84,180	3.4
Math2	3911	16	16	4	78,220	3.2

RQ1 Do our proposed Wse-MF methods outperform the state-of-the-art methods?

RQ2 How do different parameter settings (e.g. λ, the latent factor, and the weight of the predicted data) affect Wse-MF?

RQ3 Is our proposed Student-Exercise weighting strategy effective?

First, we introduce the three datasets, provide the experimental settings and specify the baselines, before answering the above three questions in Section 4.4.

4.1. Datasets

The experiments are conducted on three real-world datasets which are widely used. The first dataset comprises the scores of middle school students performing fraction subtraction exercises [23,33]. The second and third datasets contain the results of two mathematics examinations taken by students in high school; these include both objective and subjective exercises.[1] Each of the datasets consists of a score matrix and a Q-matrix. Each Q-matrix is specified by experts in the appropriate subject field. A brief summary of these datasets is shown in Table 2. Note that scores in the *FrcSub* matrix are all either 0 or 1, while those in the *Math1* and *Math2* matrices are real values between 0 and 1.

4.2. Experiment settings

For **RQ1**, we select three evaluation indicators to measure the effectiveness of our model: *accuracy (ACC)*, *root mean square error (RMSE)* and *mean absolute error (MAE)*. Using *ACC*, we train our model on 80% of the interactions, withholding 20% for testing, and the data is evenly and randomly

[1] http://staff.ustc.edu.cn/~qiliuql/data/math2015.rar.

distributed. Since exercises in the *Math1* and *Math2* datasets contain subjective questions, and their score matrices contain values from 0 to 1, the task can be regarded as a regression problem and hence we conduct our experiments at the 20%, 40%, 60% and 80% test ratios, using *RMSE* and *MAE* as the evaluation metrics. Generally, the smaller the values of these metrics are, the better the results we have. Furthermore, as the datasets contain a large number of objective questions, this can also be considered a classification task. We construct our experiments using *leave-one-out* evaluation and multiple divisions with the same test ratio. Since the performance does not vary much from one division to the next, we choose the average as the experimental result. In our experiments, we compare the proposed approach with eight standard methods as described below. For the purpose of comparison, we record the best performance of each algorithm by tuning its parameters. Moreover, in order to answer **RQ2** and **RQ3**, we also conduct experiments on the parameters and innovations of our model.

4.3. Baselines

The eight baseline methods are as follows:

1. **Item Response Theory (IRT)**
   Birnbaum [34], Rasch [35] propose a cognitive diagnosis method modeling examinees' latent traits and using parameters of problems like difficulty and discrimination.
2. **Deterministic Inputs Noisy AND Gate (DINA)**
   De La Torre [9] build a cognitive diagnosis method modeling students' skill proficiency with a Q-matrix and incorporating the slip and guess factors of exercises.
3. **Probabilistic Matrix Factorization (PMF)**
   Minh and Salakhutdinov [36] construct a latent factor model projecting students and problems into a low-dimensional space.
4. **Non-negative Matrix Factorization (NMF)**
   Desmarais [37] proposes a latent non-negative factor model, which can be viewed as a topic model.
5. **k-Nearest Neighbor (kNN)**
   Zhao [38] calculates the similarity between students and finds similar exercises to recommend.
6. **Probabilistic Matrix Factorization and Cognitive Diagnosis (PMF-CD)**
   Zhu et al. [24] combine the complementary advantages of PMF and cognitive diagnosis, taking both the individual and the common study status of students into account.
7. **fuzzy Cognitive Diagnosis Framework (fuzzyCDF)**
   Liu et al. [10] fuzzify the skill proficiency of examinees and combine fuzzy set theory and educational hypotheses to model the examinees' mastery of the problems based on their proficiency.
8. **Personalized Exercise Recommendation via Implicit Skills (ERviaIS)**
   Kang et al. [39] use exercise-concept pairs to discover implicit relations between skills with the aid of Dynamic Key-Value Memory Networks (DKVMN) and predict students' performance by combining the updated Q-matrix with a cognitive diagnosis model.

For the CDM and MF methods, we analyzed the *time complexity*. For the DINA, PMF, kNN, PMF-CD and ERviaIS methods, with evenly divided data, we constructed the experiments on the 20% test ratio, using *ACC* as the evaluation metric; For the IRT, DINA, PMF, NMF, kNN, PMF-CD and fuzzyCDF methods, with evenly divided data, we constructed the experiments on the 20%, 40%, 60% and 80% test ratios, using *MAE* and *RMSE* as the evaluation metrics.

Table 3
Time Complexity of models.

Model	Time complexity
DINA	$O\left(MN2^L\right)$
MF	$O(MNK + \|R\|K)$
Wse-MF	$O\left((M+N)K^2 + \|R\|K\right)$

M is the number of students, N is the number of exercises, L is the number of skills, K is the latent factor (inner dimension of the factorization), and $|R|$ is the number of training data in student-exercise matrix R.

Table 4
ACC of models.

Dataset Model	Frcsub ACC	Math1	Math2
DINA	0.34	0.40	0.36
kNN	0.53	0.56	0.51
PMF	0.64	0.60	0.66
PMF-CD	0.74	0.65	0.66
ERviaIS	0.76	0.66	0.67
Wse-MF	**0.83**	**0.72**	**0.80**

Best results in each group are in bold.

4.4. Results and discussion

4.4.1. Performance comparison (***RQ1***)

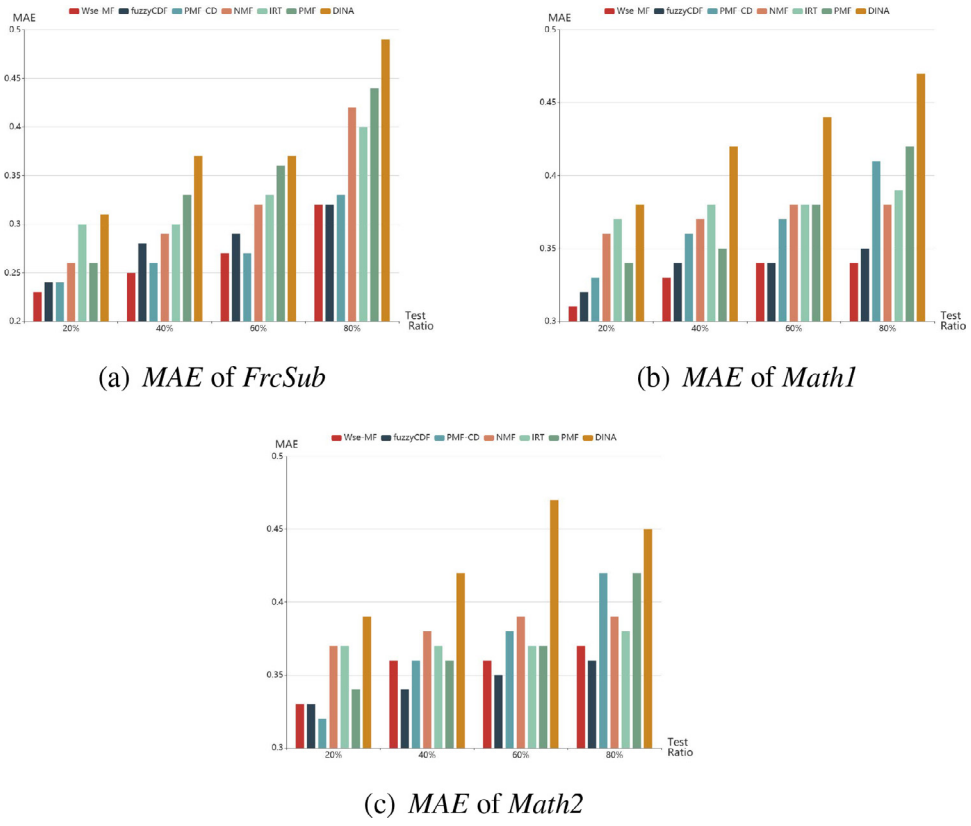
In this section, we show the experimental results of the models at multiple test ratios on three datasets. Next, we analyze the performance of the Wse-MF approach relative to the other methods using three measures, *Time complexity*, *ACC* and *MAE/RMSE*.

Time complexity: Table 3 shows the time complexity of Wse-MF as compared with the DINA and MF methods. The complexities of DINA and MF are, respectively, $O\left(MN2^L\right)$ and $O(MNK + |R|K)$, where, as seen earlier, *M, N, L* denote the numbers of students, exercises, and skills, K is the inner dimension of the factorization, and $|R|$ is the number of training data in matrix R. Note that in the MF model, the inner dimension of the factorization process is artificially set and its value is often smaller than both M and N. MF methods are mainly affected by the number of students and the amount of exercises; DINA is also affected by exponential growth in the number of skills, which has the biggest impact because of the Q-Matrix. Therefore, the time complexity of DINA depends on the number of skills. However, as actual online education platforms may contain huge numbers of exercises and students, the DINA and MF methods are unsuitable for large-scale data.

In contrast, we see that our approach achieves the best performance. The time complexity of Wse-MF is $O\left((M+N)K^2 + |R|K\right)$ and is little influenced by M and N due to the application of the SE-ALS optimization strategy. In Wse-MF, by performing optimization at the element level, the expensive matrix inversion can be avoided. We can speed up learning by using the cache introduced from SE-ALS [15] to avoid the massive repeated computations for which the time complexity is $O\left(K^3\right)$ [40], so that updating a student or an exercise costs $O\left(K^3 + MK^2\right)$ or $O\left(K^3 + NK^2\right)$. So, this quadratic level, $O\left((M+N)K^2 + |R|K\right)$, makes our method suitable for large-scale learning. Specifically, our approach trains more quickly than cognitive diagnosis methods. We think the reason is that cognitive diagnosis methods update a parameter to minimize the objective function of the current status, and moreover the update time is spent modeling skills.

Therefore, the Wse-MF model based on matrix factorization applies the SE-ALS optimization strategy with the lowest time complexity.

ACC: Table 4 shows the prediction results of Wse-MF and the other approaches on the three datasets. We can see that the *ACC*



(a) *MAE* of *FrcSub* (b) *MAE* of *Math1*

(c) *MAE* of *Math2*

Fig. 2. We conduct experiments on three datasets with a 20%–80% test ratio and show the *MAE* of the models. When the testing data ratio ranges from 20% to 80%, the *MAE* of our method is lower than other methods, except that it is up to 0.02 higher than fuzzyCDF when the test ratio is from 40% to 80% on the *Math2* dataset.

scores of Wse-MF on the three datasets are 83%, 72% and 80%. These scores are better than all the other models.

Compared with the matrix factorization methods, we believe that the benefits mainly come from the Student-Exercise Weighting Strategy (see Section 4.4.3), as the traditional PMF and PMF-CD methods do not use an approach based on weights associated with individual student-exercise pairs. So, the Wse-MF method has a better prediction effect by virtue of its loss function and because it considers both the learning ability of each student and the difficulty of each exercise.

MAE/RMSE: Figs. 2 and 3 compare Wse-MF with the CDM and MF methods at multiple test ratios on the three datasets using *MAE* and *RMSE* as evaluation indicators. For datasets *FrcSub* and *Math1*, Wse-MF outperforms the other methods as measured by *MAE* and *RMSE*; however, Wse-MF has poor performance on *Math2*. Under the 20% test ratio the Wse-MF score, as measured by both *MAE* and *RMSE*, is lower than (superior to) that of all the other methods. Furthermore, under the 40% test ratio, Wse-MF is also superior to the other methods as measured by *MAE* and *RMSE*, with the exception of its score on the *Math2* dataset, where fuzzyCDF is superior by just 0.02.

Generally, Wse-MF performs best on *Frcsub* while still performing well on the other two datasets. The reason may be that the model can accurately predict an objective exercise, and under the premise of not using fuzzy sets, it can also better predict a subjective exercise; however, on the *Math1* and *Math2* datasets, Wse-MF performs less well than fuzzyCDF because it does not use the fuzzy set hypothesis to distinguish between subjective and objective exercises like fuzzyCDF.

Finally, when changing the test ratio from 40% to 80%, the prediction performance is not better, but is still acceptable. In fact, it suffers from the 'cold start' problem: When the training data ratio declines from 80% to 20%, that is to say the training matrix becomes highly sparse, there may be students who have few or even no exercise records from which to predict difficulty. At this stage, less training data leads to a reduction in training times, but at the same time, the prediction error becomes large.

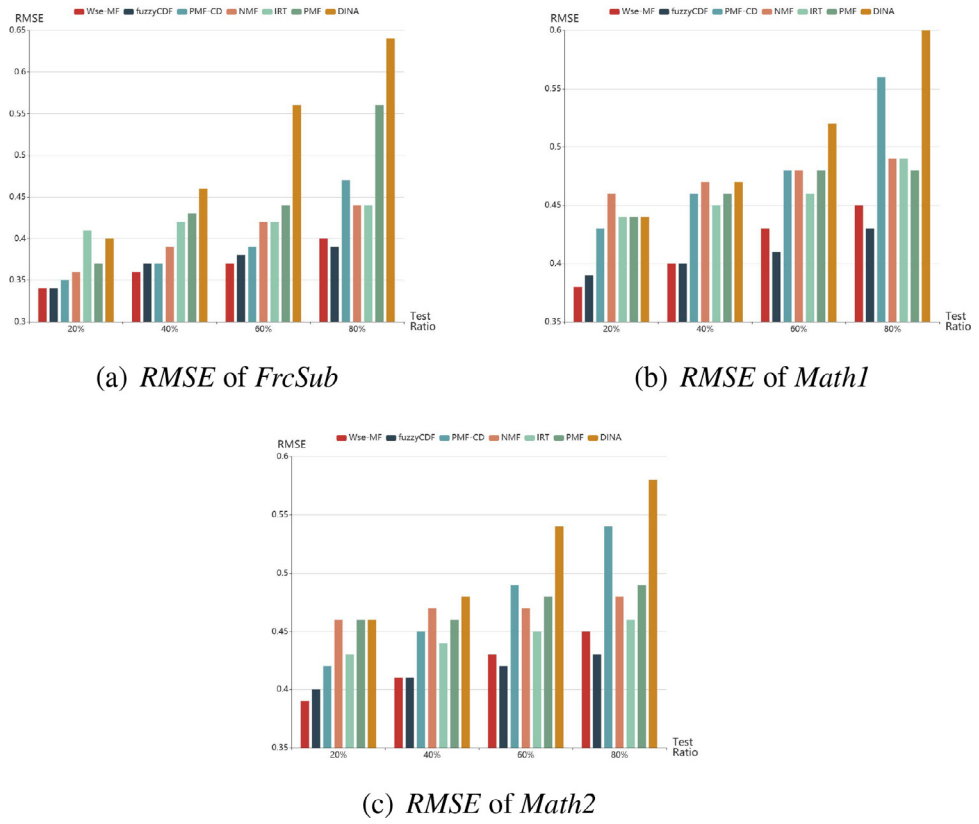
All the above evidence demonstrates that Wse-MF has a good ability to predict student performance by taking full advantage of the exercise records. This suggests that a matrix factorization model based on student learning ability and exercise difficulty is suitable for exercise prediction datasets in the field of education. Moreover, the evidence also provides a good idea for educational researchers, namely adding prior knowledge (students' learning ability and exercise difficulty) to the model. Thirdly, Wse-MF achieves rapid prediction through applying the SE-ALS optimization strategy.

4.4.2. Study of Wse-MF (***RQ2***)

As the latent factor K plays an important role in matrix factorization, we explore its impact on performance. We also analyze the influence of the parameters c_0 and λ.

Impact of latent factor K: In order to explore whether there is a suitable latent factor that can achieve the best prediction results, we experiment with different values. In particular, under the 20% test ratio on the three datasets, we chose values from 1 to 20 for the experiments, because 20 is less than the number of students and exercises. Figure 4 summarizes the experimental results. We found that there is indeed a certain factor to make the three evaluation indicators optimal for the three datasets, i.e., $K = 1$ for *Frcsub*, $K = 2$ for *Math1*, and $K = 14$ for *Math2*.

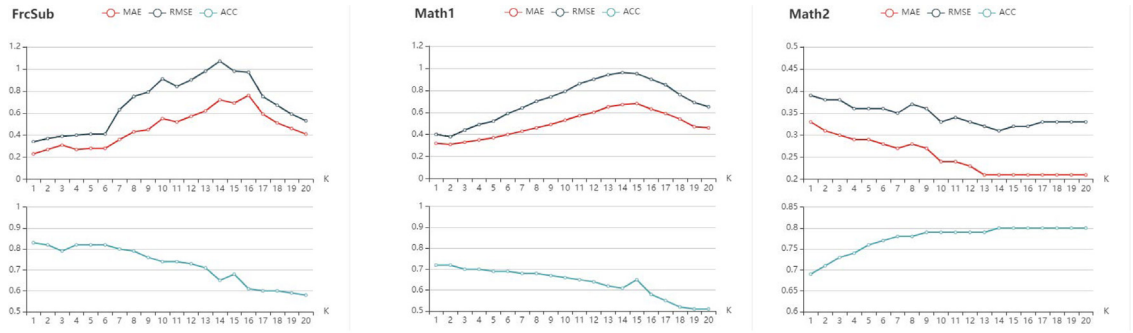
We guessed that the latent factor is closely related to the number of skills per exercise, because the potential space that maps students to exercises must have a certain connection with skills, and they are also hidden in the attribute information of students and exercises. However, according to the dataset summary



(a) *RMSE* of *FrcSub* (b) *RMSE* of *Math1*

(c) *RMSE* of *Math2*

Fig. 3. We conduct experiments on three datasets with a 20%–80% test ratio and show the *RMSE* of the models. When the test data ratio ranges from 20% to 40%, the *RMSE* of our method is lower than other methods. However, when the test ratio is between 60% and 80%, the *RMSE* is up to 0.02 higher than for fuzzyCDF.



(a) Impact of K on *FrcSub* (b) Impact of K on *Math1* (c) Impact of K on *Math2*

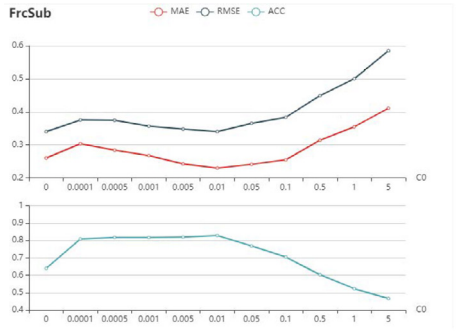
Fig. 4. We conduct experiments with K from 1 to 20 on the three datasets and show the *MAE*/*RMSE*/*ACC* of the models. Our method performs well when K is 1 on *FrcSub*, 2 on *Math1*, and 14 on *Math2*.

in Table 2 and the experimental results in Fig. 4, it seems that the factor is not directly related to skills per exercise. Therefore, we carried out an analysis of the Q-matrix and obtained further statistics.

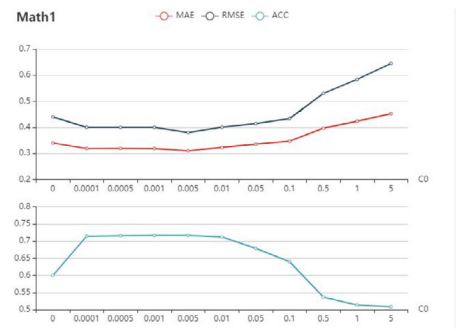
According to the Q-matrices for the three datasets, we can conclude that the average skills contained in each exercise (2.8 for *Frcsub*, 3.4 for *Math1*, and 3.2 for *Math2*) is approximately 3 and the number of exercises (20) is also equal in all three datasets. However, the number of those skills shared between the exercises varies by dataset. *Frcsub* has 8 skills, *Math1* has 11 skills, and *Math2* has 16 skills. If an exercise contains around 3 skills, less will be shared for *Math2* than for *Frcsub* and *Math1*, because the number of skills in *Math2* is higher. This could account for the higher optimal value of K for *Math2*, since more dimensions are needed to map the exercises to the skills in the matrix.

Impact of the weight of predicted data c_0: In order to observe the influence of the hyperparameter c_0 on the experimental results, we set c_0 from 0 to 5 in the experiments. Figure 5 shows the three optimal values of c_0, i.e., 0.01 for *Frcsub*, 0.005 for *Math1*, and 0.005 for *Math2*.

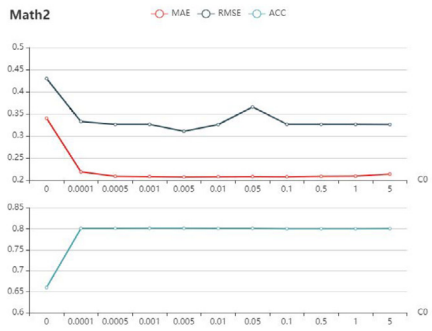
In Eq. (11), macroscopically, we interpreted c_0 as the weight of predicted data. In fact, according to the results in Fig. 5, we found the best c_0 and deduced that the role of c_0 is to balance the number of students and exercises, because the proportion of students' learning ability to exercise difficulty is not balanced in the Student-Exercise Weighting Strategy. For instance, on the *FrcSub* dataset, there are 536 students and 20 exercises. When students learning ability and exercises difficulty are normalized by *Softmax*, those exercises with a large number of students will have a smaller average learning ability, and those students with a small number of exercises will have a higher average difficulty, which



(a) Impact of c_0 on *FrcSub*



(b) Impact of c_0 on *Math1*



(c) Impact of c_0 on *Math2*

Fig. 5. We conduct experiments with c_0 from 0 to 5 on the three datasets and show the *MAE/RMSE/ACC* of the models. Our method performs well when c_0 is 0.01 on *FrcSub*, 0.005 on *Math1*, and 0.005 on *Math2*.

Table 5
Impact of λ.

Dataset	Frcsub			Math1			Math2		
λ	MAE	RMSE	ACC	MAE	RMSE	ACC	MAE	RMSE	ACC
1	0.24	0.35	0.82	0.32	0.39	0.71	0.21	0.32	0.79
0.1	0.23	0.34	0.82	0.32	0.38	0.71	0.20	0.32	0.80
0.01	**0.23**	**0.34**	**0.83**	**0.31**	**0.38**	**0.72**	**0.20**	**0.31**	**0.80**

Best results in each group are in bold.

Table 6
Effectiveness of Student-Exercise Weighting Strategy.

Dataset	Frcsub			Math1			Math2		
Model	MAE	RMSE	ACC	MAE	RMSE	ACC	MAE	RMSE	ACC
PMF	0.26	0.37	0.64	0.32	0.44	0.60	0.34	0.46	0.66
We-MF	0.26	0.37	0.82	0.32	0.40	0.71	0.22	0.34	0.80
Ws-MF	0.24	0.35	0.82	0.32	0.40	0.71	0.21	0.33	0.80
Wse-MF	**0.23**	**0.34**	**0.83**	**0.31**	**0.38**	**0.72**	**0.20**	**0.31**	**0.80**

Best results in each group are in bold.

leads to an order of magnitude difference. So, we guessed that the value of c_0 is approximately equal to the ratio between the number of exercises and students ($20/536 \approx 0.01$). This is also the case for the *Math1* and *Math2* datasets, i.e., $20/4209 \approx 0.005$ for *Math1* and $20/3911 \approx 0.005$ for *Math2*.

Impact of λ: In order to find the optimal value of parameter λ and its influence on the experimental results, we set λ from 0.01 to 1 in the experiments. Table 5 shows that the optimal value of λ is 0.01. As λ increases, the prediction results of Wse-MF decrease slightly, but it can be observed that the decrease is small. From Eq. (10), this parameter is located in the regularization item and it is only to prevent overfitting. So, it has little effect on the experimental results.

4.4.3. Effectiveness of student-exercise weighting strategy (***RQ3***)

To investigate how the Student-Exercise Weighting Strategy affects the performance, we consider Wse-MF variants using different strategies. In particular, we remove l_{ui}, d_{ui} or both from Eq. (11); in other words, we set them to $\frac{d_{ui}}{1}$, $\frac{1}{l_{ui}}$, or we completely delete the second term from the equation. We refer to the resulting models as Ws-MF, We-MF and PMF. In fact, it is illogical only to remove two items, because the model then degenerates into matrix factorization, so we select the most representative PMF model for comparison. We show the results in Table 6 and obtain the following findings:

- Wse-MF is always superior to its variants. We attribute the improvements to the combination of student learning ability and exercise difficulty, that is the Student-Exercise Weighting Strategy, which makes predicted data dependent on the prior knowledge from attributes of students and exercises.
- In most cases, Ws-MF outperforms We-MF. This shows that student learning ability is more important than exercise difficulty. After all, the number of students is much larger than the number of exercises, so we can model student attributes better.

4.4.4. Potential applications outside education

Two key aspects of the work presented here are, firstly a novel loss function based on student learning ability and exercise difficulty, and secondly, an optimization to allow faster training with lower time complexity. Both of these could be applicable in other application fields, provided that certain assumptions held. Concerning the loss function, learning ability and exercise difficulty are scalar values manipulated in matrix form. In other domains, provided that comparable information existed within an isomorphic framework, a similar function could potentially be used.

Regarding the proposed optimization, this is not education-specific. Therefore, we consider there is the possibility to apply a similar approach within another field.

5. Conclusion

In this paper, we proposed a matrix factorization model to simulate student data within an exercise prediction framework and showed that the MF technique is indeed effective under certain assumptions, one of these being that all students attempt the same exercises. MF is easily applicable to a wide variety of contexts by specifying only the input data. Aspects of our approach could also potentially be applied outside the education field, as discussed above. In contrast to previous work that applied MF, we produced a general Wse-MF approach, which can model individual students and hence predict the effect of individualized exercises. In particular, both observed data and predicted data were considered in the loss function, and heuristic knowledge was effectively utilized to predict results. To address the key efficiency challenge, we applied the learning algorithm SE-ALS, which effectively optimizes parameters just at the element level, while leaving the others fixed. Finally, we have mentioned the 'cold start' problem in Section 4.4.1. Our method performs poorly at the 80% test ratio, so in future work we will improve the model by means of incremental matrix updates and then apply it to data derived from online education platforms.

Declaration of Competing Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Data availability

Data will be made available on request.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (No. 61877050) and the Major Research and Development Projects in Shaanxi Province of China (No. 2019ZDLGY03-10).

References

[1] A. Anderson, D. Huttenlocher, J. Kleinberg, J. Leskovec, Engaging with massive online courses, in: Proceedings of the 23rd International Conference on World Wide Web, ACM, 2014, pp. 687–698.
[2] P.D. Nichols, S.F. Chipman, R.L. Brennan, Cognitively Diagnostic Assessment, Routledge, 2012.
[3] H. Burns, C.A. Luckhardt, J.W. Parlett, C.L. Redfield, Intelligent Tutoring Systems: Evolutions in Design, Psychology Press, 2014.
[4] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L.J. Guibas, J. Sohl-Dickstein, Deep knowledge tracing, in: Advances in Neural Information Processing Systems, 2015, pp. 505–513.
[5] J. Zhang, X. Shi, I. King, D.-Y. Yeung, Dynamic key-value memory networks for knowledge tracing, in: Proceedings of the 26th International Conference on World Wide Web, International World Wide Web Conferences Steering Committee, 2017, pp. 765–774.
[6] Y. Su, Q. Liu, Q. Liu, Z. Huang, Y. Yin, E. Chen, C. Ding, S. Wei, G. Hu, Exercise-enhanced sequential modeling for student performance prediction, in: The Thirty-Second AAAI Conference on Artificial Intelligence(AAAI-18), 2018, pp. 2435–2443.
[7] Q. Liu, Z. Huang, Y. Yin, E. Chen, H. Xiong, Y. Su, G. Hu, EKT: exercise-aware knowledge tracing for student performance prediction, IEEE Trans. Knowl. Data Eng. (2019). 1–1
[8] A. Sapountzi, S. Bhulai, I. Cornelisz, C. van Klaveren, Dynamic models for knowledge tracing & prediction of future performance, Data Anal. (2018) 131.
[9] J. De La Torre, DINA model and parameter estimation: a didactic, J. Educ. Behav. Statist. 34 (1) (2009) 115–130.
[10] Q. Liu, R. Wu, E. Chen, G. Xu, Y. Su, Z. Chen, G. Hu, Fuzzy cognitive diagnosis for modelling examinee performance, ACM Trans. Intell. Syst.Technol. (TIST) 9 (4) (2018) 48.
[11] J. Leighton, M. Gierl, Review of cognitively diagnostic assessment and a summary of psychometric models, in: Cognitive Diagnostic Assessment for Education: Theory and Applications, 2007, pp. 3–18.
[12] L. DiBello, L. Roussos, W. Stout, Review of cognitively diagnostic assessment and a summary of psychometric models, in: C.R. Rao, S. Sinharay (Eds.), Handbook of Statistics, Vol. 26, 2007, pp. 970–1030. Psychometrics
[13] Y. Koren, R. Bell, C. Volinsky, Matrix factorization techniques for recommender systems, Computer 42 (8) (2009) 30–37.
[14] R. Devooght, N. Kourtellis, A. Mantrach, Dynamic matrix factorization with priors on unknown values, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2015, pp. 189–198.
[15] X. He, H. Zhang, M.-Y. Kan, T.-S. Chua, Fast matrix factorization for online recommendation with implicit feedback, in: Proceedings of the 39th International ACM SIGIR conference on Research and Development in Information Retrieval, ACM, 2016, pp. 549–558.
[16] A.T. Corbett, J.R. Anderson, Knowledge tracing: modeling the acquisition of procedural knowledge, User Model. User-Adapt. Interact. 4 (4) (1994) 253–278.
[17] M.V. Yudelson, K.R. Koedinger, G.J. Gordon, Individualized Bayesian knowledge tracing models, in: International Conference on Artificial Intelligence in Education, Springer, 2013, pp. 171–180.
[18] A. Zia, J. Nouri, M. Afzaal, Y. Wu, X. Li, R. Weegar, A step towards improving knowledge tracing, in: 2021 International Conference on Advanced Learning Technologies (ICALT), 2021, pp. 38–39.
[19] S. Song, J. Li, Y. Tang, T. Zhao, Y. Chen, Z. Guan, JKT: a joint graph convolutional network based deep knowledge tracing, Inf. Sci. 580 (2021) 510–523.
[20] M.D. Reckase, Multidimensional item response theory models, in: Multidimensional Item Response Theory, Springer, 2009, pp. 79–112.
[21] S. Cheng, Q. Liu, E. Chen, Z. Huang, Z. Huang, Y. Chen, H. Ma, G. Hu, DIRT: deep learning enhanced item response theory for cognitive diagnosis, in: Proceedings of the 28th ACM International Conference on Information and Knowledge Management, 2019, pp. 2397–2400.
[22] J. Xu, Q. Li, J. Liu, P. Lv, G. Yu, Leveraging cognitive diagnosis to improve peer assessment in MOOCs, in: IEEE Access, 2021, pp. 50466–50484.
[23] K.K. Tatsuoka, Analysis of Errors in Fraction Addition and Subtraction Problems, Final report, Kikumi K. Tatsuoka, 252 ERL, 103 S. Mathews St., Univ. of Illinois, Urbana, IL 61801., 1984.
[24] T. Zhu, Z. Huang, E. Chen, et al., Cognitive diagnosis based personalized question recommendation, Chin. J. Comput. 40 (01) (2017) 176–191.
[25] Z. Liu, B.J. Jansen, Subjective versus objective questions: perception of question subjectivity in social Q&A, in: International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction, Springer, 2015, pp. 131–140.
[26] X. He, J. Tang, X. Du, H. Richang, T. Ren, C. Tat-Seng, Fast matrix factorization with nonuniform weights on missing data, IEEE Trans. Neural Netw. Learn. Syst. 8 (2020) 2791–2804.
[27] S. Zhong, L. Huang, C.D. Wang, J.H. Lai, Constrained matrix factorization for course score prediction, in: 2019 IEEE International Conference on Data Mining (ICDM), 2020, pp. 1510–1515.
[28] S. Peng, S. Wee, B. Chen, Z. Lin, Robust semi-supervised nonnegative matrix factorization for image clustering, Pattern Recognit. 111 (2021).
[29] L. Hu, N. Wu, X. Li, Feature nonlinear transformation non-negative matrix factorization with Kullback-Leibler divergence, Pattern Recognit. 132 (2022).
[30] X. Jin, J. Miao, Q. Wang, G. Geng, K. Huang, Sparse matrix factorization with L2,1 norm for matrix completion, Pattern Recognit. 127 (2022).
[31] S. Minn, Y. Yu, M.C. Desmarais, F. Zhu, J.-J. Vie, Deep knowledge tracing and dynamic student classification for knowledge tracing, in: 2018 IEEE International Conference on Data Mining (ICDM), IEEE, 2018, pp. 1182–1187.
[32] A. Paterek, Improving regularized singular value decomposition for collaborative filtering, in: Proceedings of KDD Cup and Workshop, Vol. 2007, 2007, pp. 5–8.
[33] L.T. DeCarlo, On the analysis of fraction subtraction data: the DINA model, classification, latent class sizes, and the q-matrix, Appl. Psychol. Meas. 35 (1) (2010) 8–26.
[34] A.L. Birnbaum, Some latent trait models and their use in inferring an examinee's ability, Statist. Theories Mental Test Scores (1968).
[35] G. Rasch, On general laws and the meaning of measurement in psychology, in: Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 4, 1961, pp. 321–333.
[36] A. Mnih, R.R. Salakhutdinov, Probabilistic matrix factorization, in: Advances in Neural Information Processing Systems, 2008, pp. 1257–1264.
[37] M.C. Desmarais, Mapping question items to skills with non-negative matrix factorization, ACM SIGKDD Explor. Newsl. 13 (2) (2012) 30–36.
[38] Y. Zhao, Recommended exercises based on collaborative filtering algorithm (in Chinese), Sci. Technol. Econ. Mag. (005) (2016) 136. https://www.doc88.com/p-7824554938674.html?r=1
[39] W. Kang, L. Zhang, B. Li, J. Chen, X. Sun, J. Feng, Personalized exercise recommendation via implicit skills, in: Proceedings of the ACM Turing Celebration Conference-China, ACM, 2019, p. 77.
[40] Y. Hu, Y. Koren, C. Volinsky, Collaborative filtering for implicit feedback datasets, in: 2008 Eighth IEEE International Conference on Data Mining, Ieee, 2008, pp. 263–272.

Xia Sun received a PhD from Xian Jiaotong University, China, in 2006. She is a Professor in the School of Information Science and Technology at Northwest University, Xian, China. Her current research interests include natural language processing and machine learning. Recent projects have included causality extraction from educational data, and relationship extraction from bioinformatics text. She has served

as a reviewer for IEEE Transactions on Industrial Informatics, IEEE Transactions on Neural Networks and Learning Systems and Chinese Journal of Electronics. She has also reviewed for the IEEE International Conference on Computational Science and Engineering). She is the co-author of 40 articles and is editor or co-editor of four books.

Richard Sutcliffe received a PhD from University of Essex in 1989. He is an Associate Professor at Northwest University China. His research interests are in the areas of Natural Language Processing, Information Retrieval and Music Information Retrieval. Recent projects have included persuasive conversational agents, public sector message classification, analysis of classical music texts, and personality and translation ability. He has reviewed for Artificial Intelligence Review, Computational Linguistics, Computers and the Humanities, Information Processing and Management, Information Retrieval Journal, Journal of Natural Language Engineering, Journal Traitement Automatique des Langues. Conferences he has reviewed for include ACL, CIKM, COLING, IJCNLP, LREC, NAACL-HLT, and SIGIR. He is the co-author of 108 articles and is co-editor of three books and ten conference proceedings.

Jun Feng received a PhD from City University of Hong Kong in 2006. She is a Professor in the School of Information Science and Technology at Northwest University. Her research areas include pattern recognition and machine learning, especially in the fields of medical imaging analysis and intelligent education. Recent projects have included medical image analysis with deep learning, and intelligent education based on AI and Brain-Human Interaction. She has reviewed for many journals, including TSP, JIVP, MTAP, JDIM, CJC, JCAD, OPE, and INFPHY. Conferences she has reviewed for include IEEE-VR, MICCAI, SIGCSE, IWCSE, and CompEd. She is a member of IEEE and ACM, and is co-author of 132 articles and co-editor of three books.