Context-Aware Recommendation-Based Learning Analytics Using Tensor and Coupled Matrix Factorization

A THESIS
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Faisal Almutairi

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF SCIENCE

Prof. Nicholas D. Sidiropoulos

August, 2016

Acknowledgements

I must express my honest gratitude to my research advisor, Professor Nicholas Sidiropoulos, for his advice and support. His patient guidance and valuable insights were essential, not only to make this work possible, but also to provide me with an inspiring and motivating research experience. I could not be more grateful for the learning opportunities I am receiving from his knowledge as a member of his group.

I am very thankful to the committee members, Professor Jarvis Haupt, Professor George Karypis, and Professor Shuzhong Zhang, for their time and effort.

I also would like to thank my labmates, Bo Yang, Ahmed Zamzam, John Tranter, Kejun Huang, Aritra Konar, Xiao Fu, Charilaos Kanatsoulis, Nikos Kargas and Cheng Qian, for our stimulating discussions that enlightened my first glance of research and provided insights for this work.

I would also like to thank my academic advisor, Professor Emad Ebbini for his guidance, which helped me to adjust to the atmosphere of graduate school during my first year of the master program.

Last but not least, I would like to thank my family, parents, and siblings for their immeasurable amount of unconditional love and support throughout my studies and my life.

Dedication

In memory of my brother Muhammad. Your kindness and support left fingerprints of grace on my life. You shan't be forgotten.

Abstract

In higher education, student retention and timely graduation are enduring challenges. Educational, advising, and counseling innovations and interventions are needed to address these challenges. With the rapidly expanding collection and availability of learning data and related analytics, student performance can be accurately monitored, and possibly predicted ahead of time, thus enabling early warning and degree planning 'expert systems' to provide disciplined decision support to counselors, advisors, educators – and even help guide students in semester-to-semester course selection. Previous work in educational data mining has explored matrix factorization techniques for grade prediction, albeit without taking contextual information into account. Temporal information should be informative as it distinguishes between the different class offerings and indirectly captures student experience as well. To exploit temporal and/or other kinds of context, we develop three approaches that leverage side information besides historical grades under the framework of Collaborative Filtering (CF). Two of the proposed methods build upon Coupled Matrix Factorization (CMF) with a shared latent matrix factor. The third method utilizes tensor factorization to model grades and their context. For each method, the latent factors obtained using matrix/tensor factorization lead to a compact model which we use not only to predict the unseen grades, but also the associated contextual information. We evaluate these approaches on grade datasets obtained from the University of Minnesota. Experimental results show that quite accurate prediction is possible using even simple models, while more advanced approaches outperform the prior art in predicting randomly missing entries.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

There has recently been growing interest in educational data mining [2] in general, and predicting student performance in particular [3], [4], [5], [6], [7], [8], [9], [10], [11]. The motivation behind these approaches is coming from their importance in practice. They support a variety of applications, from predicting student performance in class activities during the semester as an aid to the instructor [12], [3], which can be used in "early warning systems" to identify students who are on the verge of failing a class in order to take corrective action [5]. Other work has focused on predicting whether a student is able to perform a given task correctly [7], [13], [8], which can be used for class evaluation and exercise recommendation purposes. Moreover, leveraging course recommendation approaches as in [14], [15], [16], and methods for predicting final grades as in [10], [6], [4] can help to minimize the time-to-degree and build better academic planning tools. The work in this thesis fall under the last application as we aim to predict student performance at the course-level in terms of final grades in classes students haven't yet taken. This can help in semester-to-semester course selection, recommendation of 'bridge' courses, and early warning systems.

1.1 Background and and Related Work

Many researchers have proposed approaching the student performance prediction problem using **regression techniques** as in [17]. Recently, Elbadrawy, Studham, and Karypis [3] have presented collaborative multi-regression models which, unlike single

regression-based approaches [5], allow for cross-student information sharing. They cross-leverage the advantages of regression-based models in accounting for students' interactions with Learning Management Systems (LMS) and factorization-based models in creating student-specific predictions [3]. Polyzou and Karypis [6] proposed models that utilize a judiciously chosen subset of the historical information when predicting grades for a specific course or a specific student: Course-Specific Regression (CSR), or Student-Specific Regression (SSR), respectively. In CSR, a grade is predicted by the student's prior grades with a linear regression model that determines how much each one of her/his past grade contributes. This regression vector is estimated by a model that utilizes only rows corresponding to students who took the course to be predicted in the student $\times$ course grade matrix. As they pointed out, CSR uses the same regression vector for all students, which can be a limitation when applied on flexible academic programs where students have less common courses. To overcome this issue, in SSR, they eliminate courses that a student s hasn't taken from the grade matrix as well as students who haven't taken the target course c, or don't have enough common courses with s to estimate a regression model that is personalized for each student [6].

Researchers have adapted **recommender system techniques** to the student performance prediction problem [11] [7], [9], [8], [4], and the course recommendation problem [14], [15], [16]. Typically, the users' ratings for items are represented in a user $\times$ item matrix [18]. Similarly, in the setting of latent factor and Matrix Factorization (MF) models, the students' grades are usually tabulated in a sparse student $\times$ course matrix ,$\mathbf{G} \in \mathbb{R}^{n \times m}$, as in figure 1.1. In the context of grade prediction, various methods based on MF have been used to estimate the latent factors which produce representations for each student and course in order to be used to predict grades. The goal is to fill the missing values, which can be viewed as a matrix completion task [19]. The main idea here is to factor $\mathbf{G}$ as $\mathbf{G} \approx \mathbf{A}\mathbf{B}^T$, where $\mathbf{A} \in \mathbb{R}^{n \times F}$, $\mathbf{B} \in \mathbb{R}^{m \times F}$ and F is the model rank. Then missing entries in $\mathbf{G}$ are imputed based on the factors $\mathbf{A}\mathbf{B}^T$, where $\mathbf{A}$ and $\mathbf{B}$ are estimated from the data by minimizing a suitable loss function. Sweeney, Lester, and Rangwala [4] explored estimating those factors for grade data via Singular Value Decomposition (SVD) and showed that grade prediction improved when SVD is followed by k-NN (k-Nearest Neighbor) post-processing, as detailed in [20]. It has been shown that adding global and local biases (for every student and course) to the MF

model as in [6], [4] reduces the error in grade prediction.

Students \ Courses	*1*	*2*	*3*	.....	*m*
1	4		3.67	4	
2	3	3.33			
3			2		
⋮	3	2.67		3.67	
n		3.33			4

Figure 1.1: Student × course grade matrix

.

The models we propose in this work draw from factor analysis for matrices and tensors, but can also be viewed under a recommender system 'lens'. It is therefore useful to provide the following classification for recommender systems, before we proceed to explain the specific modeling and optimization approaches proposed as part of this thesis. Generally, models that are intended to predict the missing values (ratings or grades) are classified as follows [18]:

1. Collaborative Filtering (CF): in which predictions are calculated based on the historical ratings of all users collectively, either based on the similarities between users *(neighborhood-based)* or on latent factors and MF *(model-based)* [18].

2. Content-based recommendations: in which recommendation is provided depending on the similarity in the features of items that a user has rated or in the attributes of users who have rated the same items [21], [18].

3. Hybrid between CF and content-based models: various works have tried to cross-leverage the advantages of these two types of approaches. Our work here can be classified under this category.

The user and item features in CF models are learned from the data, which is assumed to exhibit a hidden low-dimension structure [18]. On the other hand, in content-based recommendations, these features are given, e.g., a user's gender or a movie's genre. In

the setting of student performance forecasting, the student's major or GPA, and the course's level or department are examples of such features.

In many situations, recommender system models that take additional **contextual information** into account provide more accurate recommendations as they are customized to each scenario [22], [23], [21], [24]. Some of these methods are extending the traditional MF to incorporate useful side information besides the typical historical grades (ratings). So-called context-aware recommender systems (CARS) can be categorized into three types: contextual *pre-filtering*, where the data is selected based on context; contextual *post-filtering*, where recommendations are filtered after they have been computed; and *contextual modeling*, where the context is accounted for while computing recommendations [25]. The proposed models here in this paper are under the last category as context is exploited within the model. Karatzoglou, Amatriain, and Baltrunas [21] generalized MF representation in the context of CF and model data as a User $\times$ Item $\times$ Context N-dimensional tensor where every type of context is introduced as a new mode in the tensor. As a special case, in predicting student performance the temporal information was exploited by modeling data as a three-way tensor (student $\times$ task $\times$ time) in [7].

1.2 Contributions

In this work, we extent the traditional two-dimensional MF model and propose three generic methods in the context of CF. Although these models can potentially be applied to any recommendation task (movies, products, places, etc.), we focus here on student grade prediction. In particular, we present two Coupled Matrix Factorization (CMF) models and one Low-Rank Tensor Factorization (LRTF) model. These models allow flexible integration of contextual information by modeling the context associated with each grade in a matrix (instead of a separate tensor mode) in a similar manner to the student $\times$ course matrix as shown in figure 1.1. In CMF models the grade and context matrices are factored simultaneously by constraining them to have the same student latent factor matrix (called CMFS) or course latent factor matrix (called CMFC). In the case of LRTF, we model data in a tensor which has the grade matrix as the first frontal slab and the context matrix behind it. We factor this tensor using Candecomp/Parafac

(CP) decomposition. In all cases, the obtained factors provide compact representation of the data which we use to predict student grades. The three proposed models allow for incorporating more than one context. Another advantage of our modeling approaches is that they predict not only the unseen grade or rating, but also the context in which the grade or rating will be earned/given. This is useful for forecasting course enrollments and other applications that can help schools to anticipate e.g., student enrollments and other contextual factors. The prevalent approach to incorporate context within the model in CF is to introduce a new tensor mode (index) for each context variable as in [21],[7], [26] which increases the tensor size exponentially in the number of context variables, and yields extremely sparse tensor data. The reason is that, usually, each grade (rating) is given in only one context. Very sparse tensors require high rank to approximate (think, e.g., of a diagonal matrix, which is full rank if the elements on the diagonal are nonzero). Our modeling, in contrast, maintains a common sparsity structure across the different matrices in CMF or tensor slabs in LRTF, facilitating low-rank modeling. Also, if context is modeled as a new mode and we wish to predict for next semester, we face the issue of predicting entire missing slab. Our modeling of the context allows us to deal with this "cold start" problem of an entire semester missing that we wish to predict.

The formulation of the problem we are interested in is discussed in chapter 2. The formulations and algorithms for CMF and LRTF models are presented in chapters 3 and 4, respectively. Our experimental setup is discussed in detail in chapter 5 and experimental results are summarized in chapter 6. We verify that the proposed models improve the baseline where no context is taken into account, and outperform other CF methods that have been recently used when predicting randomly missing grades.

Chapter 2

Problem Formulation

Given a grade dataset of (student, course) pairs, with contextual information, e.g., time, our main goal is to predict students' final grades in courses they haven't taken. In particular, we introduce three different models in chapters 3 and 4 in order to incorporate arbitrary side information alongside with historical grades to improve the accuracy of grade prediction. We denote the very sparse student $\times$ course grade matrix $\mathbf{G}_o$ which is comprised by the observed grades for n students in m courses. As for traditional MF techniques, $\mathbf{G}_o$ serves as the primary information source in our models [4].

In our experiments, we have tried two different types of side information: absolute time $\mathbf{T}_o$ in which the courses were taken, and student experience $\mathbf{E}_o$. Time in $\mathbf{T}_o$ is measured in semesters, while a student's experience in $\mathbf{E}_o$ is calculated by the number of semesters she/he has been in the program. The context is tabulated in a student $\times$ course matrix, in the same way as the grade matrix $\mathbf{G}_o$ using (student, course) pairs. $\mathbf{T}_o(i,j)/\mathbf{E}_o(i,j)$ reveals the temporal/experience information corresponding to the grade $\mathbf{G}_o(i,j)$. For the remainder of this thesis, we focus on the time context $\mathbf{T}_o$ in our formulation and results as we found that it is the most informative. Although the context matrix is formed in the same way, it is modeled and exploited differently among the three models.

We train our models (CMFS, CMFC, and LRTF) on the observed grades and their associated context after excluding the test set. After training each model, the task is to predict grades $\widehat{g}_{ij}$ in the test data. Those models predict not only missing grades, but also the context in which the grade is earned, $\widehat{t}_{ij}$.

Chapter 3

Coupled Matrix Factorization (CMF)

Latent factor and traditional MF techniques explained in section 1.1 have been used in the context of recommender systems, specifically in CF-based methods [27]. Researchers have adapted recommendation techniques, e.g., based on MF to address the grade prediction problem [9], [8] interpreting students, courses, and grades as users, items, and ratings, respectively. Those models have shown very good results in the context of grade prediction using only historical grades of students [6], [27].

Contextual information, such as time (e.g., measured in semesters) and student seniority (experience) should be informative when predicting grades. For instance, students who have taken the same courses with similar grades are more predictive for each other if they actually took these classes in the same semesters (same time) than if they have a big time gap. A student might fail a class if taking it in her/his freshman year, but might get an A taking the same class in her/his senior year. This is an example of how student seniority can help as side information.

The main idea of Coupled Matrix Factorization (CMF) is to incorporate and exploit those pieces of contextual information into the traditional latent factor and MF models without over-complicating the solution. There are two CMF models proposed in this thesis, which are presented in detail in this chapter. In the first section 3.1, we explain the CMF with common student factors (CMFS), while CMFC, CMF with common

course factors, is explained in section 3.2. Student and course bias terms are added to these two models in section 3.3. In the last section 3.4, algorithms to solve these two CMF models are presented in detail.

3.1 Coupled Matrix Factorization with Common Student Factors (CMFS)

In this model, CMF with a common student/user factors (CMFS), we enforce the students $\times$ courses grade matrix to have the same student latent factors as the students $\times$ courses context matrix. First, define a matrix $\mathbf{W} \in \mathbb{R}^{n \times m}$, such that:

$$\mathbf{W}(i,j) = \begin{cases} 1 & \text{if } (i,j) \in \mathcal{A}, \\ 0 & \text{otherwise.} \end{cases} \tag{3.1}$$

where $\mathcal{A}$ includes the indices of observed grades. In the sequel, we also use $\overline{\mathbf{W}} \in \mathbb{R}^{n \times m}$ to denote the complement of $\mathbf{W}$. Hence, $\overline{\mathbf{W}}$ has ones at the indices of the missing entries of the grade matrix and zeros elsewhere. Now, we can define the final grade and time matrix that include observed and missing entries, which are used throughout this thesis:

$$\mathbf{G} := \mathbf{W} \circledast \mathbf{G}_o + \overline{\mathbf{W}} \circledast \mathbf{G}_m, \qquad \mathbf{T} := \mathbf{W} \circledast \mathbf{T}_o + \overline{\mathbf{W}} \circledast \mathbf{T}_m \tag{3.2}$$

where $\circledast$ denotes the Hadamard (element-wise) product. Then, CMFS can be formulated as follows:

$$\min_{\mathbf{G}_m, \mathbf{T}_m, \mathbf{A}, \mathbf{B}_1, \mathbf{B}_2} \quad \|\mathbf{W} \circledast \mathbf{G}_o + \overline{\mathbf{W}} \circledast \mathbf{G}_m - \mathbf{A}\mathbf{B}_1^T\|_F^2 + \|\mathbf{W} \circledast \mathbf{T}_o + \overline{\mathbf{W}} \circledast \mathbf{T}_m - \mathbf{A}\mathbf{B}_2^T\|_F^2 \tag{3.3}$$

where $\mathbf{G}_o, \mathbf{T}_o \in \mathbb{R}^{n \times m}$ are the students $\times$ courses grade matrix of observed grades and the corresponding students $\times$ courses matrix of timestamps, respectively. Note this formulation (3.3) has only one context, $\mathbf{T}_o$, but we could add another context by following the same concept of shared student latent factors.
To reformulate CMFS in a simpler form, define:

$$\mathbf{X} := \left[\begin{array}{c:c} \mathbf{G} & \mathbf{T} \end{array}\right], \quad \mathbf{B}_c^T := \left[\begin{array}{c:c} \mathbf{B}_1^T & \mathbf{B}_2^T \end{array}\right], \tag{3.4}$$

where $\mathbf{X} \in \mathbb{R}^{n \times 2m}$. Then, (3.3) can be reformulated as follows:

$$\min_{\mathbf{G}_m, \mathbf{T}_m, \mathbf{A}, \mathbf{B}_c} \quad \|\mathbf{X} - \mathbf{A}\mathbf{B}_c^T\|_F^2 \tag{3.5}$$

Looking into equation (3.3) we can see that $\mathbf{A}$ and $\mathbf{B}_1$ are the low-rank factors of $\mathbf{G}$ and $\mathbf{A}$ and $\mathbf{B}_2$ are the low-rank factors of $\mathbf{T}$. Hence, the grades and the context share the same student latent factors matrix, $\mathbf{A}$. Here note that $\mathbf{X}$ is an implicit function of $\mathbf{G}_m, \mathbf{T}_m$. One approach is to fix those, solve for $\mathbf{A}$ and $\mathbf{B}_c$ in (3.5) by Singular Value Decomposition (SVD) of $\mathbf{X}$, then fix $\mathbf{A}$ and $\mathbf{B}_c$ and impute the missing entries (i.e., update $\mathbf{G}_m, \mathbf{T}_m$), and continue to alternate between the two types of updates – see section 3.4.

Intuitively, in order for two students to have the same latent factor representation (corresponding rows of $\mathbf{A}$), they must have the same grade transcripts *and* time profiles as illustrated in figure 3.1. The idea is that students sharing the same course timestamps in addition to grade transcripts are more predictive for each other than if they only have same grades. For instance, if students s_1 and s_2 correspond to identical rows of $\mathbf{A}$ (horizontal black lines) in figure 3.1, we can predict the missing entry in s_2's grade in $\mathbf{G}$ based on s_1's grade.

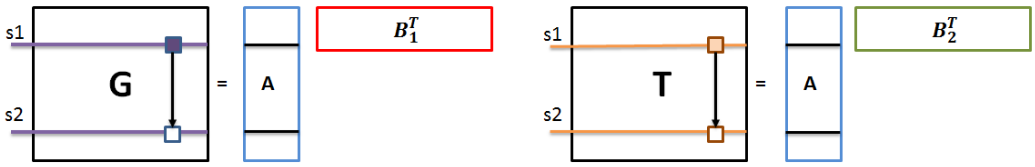

Figure 3.1: Illustration of the basic idea behind CMFS

3.2 Coupled Matrix Factorization with Common Course Factors (CMFC)

The context is exploited in this model, CMFC, by constraining it to have the same course latent factors as the students × courses grade matrix. CMFC is formulated as

follows:

$$\min_{\mathbf{G}_m,\mathbf{T}_m,\mathbf{A}_1,\mathbf{A}_2,\mathbf{B}} \quad \|\mathbf{W} \circledast \mathbf{G}_o + \overline{\mathbf{W}} \circledast \mathbf{G}_m - \mathbf{A}_1\mathbf{B}^T\|_F^2 + \|\mathbf{W} \circledast \mathbf{T}_o + \overline{\mathbf{W}} \circledast \mathbf{T}_m - \mathbf{A}_2\mathbf{B}^T\|_F^2 \tag{3.6}$$

Clearly, $\mathbf{A}_1$ and $\mathbf{B}$ are the low-rank factors of $\mathbf{G}$ (defined in (3.2)), and $\mathbf{A}_2$ and $\mathbf{B}$ are the low-rank factors of $\mathbf{T}$. In a similar manner to what we did to simplify CMFS, we define:

$$\mathbf{Y} := \left[\begin{array}{c:c} \mathbf{G}^T & \mathbf{T}^T \end{array}\right], \quad \mathbf{A}_c^T := \left[\begin{array}{c:c} \mathbf{A}_1^T & \mathbf{A}_2^T \end{array}\right], \tag{3.7}$$

where $\mathbf{Y} \in \mathbb{R}^{m \times 2n}$. Then, (3.6) can be reformulated as follows:

$$\min_{\mathbf{G}_m,\mathbf{T}_m,\mathbf{A}_c,\mathbf{B}} \quad \|\mathbf{Y} - \mathbf{B}\mathbf{A}_c^T\|_F^2 \tag{3.8}$$

Problem (3.8) can be solved by alternating between Singular Value Decomposition (SVD) of $\mathbf{Y}$ and imputation for missing entries by updating $\mathbf{G}_m$ and $\mathbf{T}_m$.

Studies have shown that grade statistics for a given course change over time, e.g., due to instructor variability, textbook changes, and broader trends such as grade inflation. In CMFC formulation, in order for two courses to have the same latent factor representation (corresponding columns of $\mathbf{B}^T$), they must have the same grade statistics *and* time profiles as illustrated in figure 3.2. This formulation will pick up the change of grades over time. The idea behind this model is that courses sharing the same course timestamps in addition to grade statistics are more predictive for each other than if they only have same grades. For example, if two courses, c_1 and c_2, correspond to identical columns of $\mathbf{B}^T$ (vertical black lines) in figure 3.2, we can predict the missing entry in c_2 grades in $\mathbf{G}$ based on the corresponding c_1 grade.

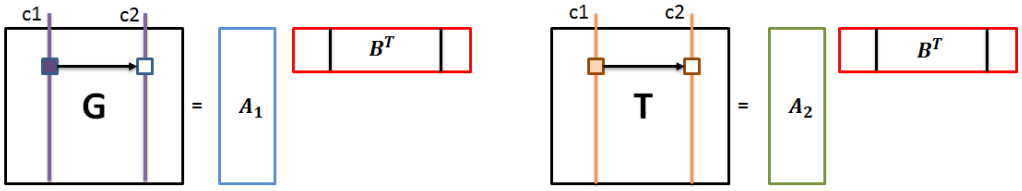

Figure 3.2: Illustration of the basic idea behind CMFC

3.3 Student and Course Biases for CMF Models

For more accurate prediction of grades, we add student and course bias terms to both the grade and context matrix. This is inspired by the improvement user and item biases impart on movie rating recommender systems [18], [28]. Modeling how a student is likely to perform (student bias) and how difficult is a certain course (course bias) has also been shown to be effective in grade prediction using different but related models and approaches [6], [3], [7]. After incorporating biases, the formulas of CMFS and CMFC (in order) become:

$$\begin{aligned}\min_{\mathbf{G}_m,\mathbf{T}_m,\mathbf{A},\mathbf{B}_1,\mathbf{B}_2,\mathbf{b}_s,\mathbf{b}_c,\mathbf{t}_s,\mathbf{t}_c} \quad & \|\mathbf{W} \circledast \mathbf{G}_o + \overline{\mathbf{W}} \circledast \mathbf{G}_m - \mathbf{A}\mathbf{B}_1^T - \mathbf{b}_s\mathbf{1}^T - \mathbf{1}\mathbf{b}_c^T\|_F^2 \\ & + \|\mathbf{W} \circledast \mathbf{T}_o + \overline{\mathbf{W}} \circledast \mathbf{T}_m - \mathbf{A}\mathbf{B}_2^T - \mathbf{t}_s\mathbf{1}^T - \mathbf{1}\mathbf{t}_c^T\|_F^2\end{aligned} \tag{3.9}$$

$$\begin{aligned}\min_{\mathbf{G}_m,\mathbf{T}_m,\mathbf{A}_1,\mathbf{A}_2,\mathbf{B},\mathbf{b}_s,\mathbf{b}_c,\mathbf{t}_s,\mathbf{t}_c} \quad & \|\mathbf{W} \circledast \mathbf{G}_o + \overline{\mathbf{W}} \circledast \mathbf{G}_m - \mathbf{A}_1\mathbf{B}^T - \mathbf{b}_s\mathbf{1}^T - \mathbf{1}\mathbf{b}_c^T\|_F^2 \\ & + \|\mathbf{W} \circledast \mathbf{T}_o + \overline{\mathbf{W}} \circledast \mathbf{T}_m - \mathbf{A}_2\mathbf{B}^T - \mathbf{t}_s\mathbf{1}^T - \mathbf{1}\mathbf{t}_c^T\|_F^2\end{aligned} \tag{3.10}$$

Where $\mathbf{b}_s \in \mathbb{R}^n$ is the student grade bias vector, $\mathbf{b}_c \in \mathbb{R}^m$ is the course grade bias vector, $\mathbf{t}_s \in \mathbb{R}^n$ is the student context (time) bias vector, and $\mathbf{t}_c \in \mathbb{R}^m$ is the course context bias vector. Therefore, after we train the two CMF models on the grade training set and its corresponding context, the prediction of the grade that a student i is going to obtain in a course j and the prediction of the time in which this grade will be earned are given in (3.11) for CMFS and in (3.12) for CMFC:

$$\widehat{g}_{i,j} = b_s(i) + b_c(i) + \mathbf{A}(i,:)\mathbf{B}_1^T(:,j), \qquad \widehat{t}_{i,j} = t_s(i) + t_c(i) + \mathbf{A}(i,:)\mathbf{B}_2^T(:,j) \tag{3.11}$$

$$\widehat{g}_{i,j} = b_s(i) + b_c(i) + \mathbf{A}_1(i,:)\mathbf{B}^T(:,j), \qquad \widehat{t}_{i,j} = t_s(i) + t_c(i) + \mathbf{A}_2(:,i)\mathbf{B}^T(:,j) \tag{3.12}$$

3.4 CMF Algorithm

In this section we provide the algorithm that solves the CMF models. Since the two formulas, CMFS and CMFC, can be solved in a very similar manner, we focus here on the instance of CMFS in equation (3.9). Recall $\mathbf{G}$ and $\mathbf{T}$ as defined in equation (3.2). Then, define:

$$\mathbf{G}_b := \mathbf{G} - \mathbf{b}_s\mathbf{1}^T - \mathbf{1}\mathbf{b}_c^T, \qquad \mathbf{T}_b := \mathbf{T} - \mathbf{t}_s\mathbf{1}^T - \mathbf{1}\mathbf{t}_c^T \tag{3.13}$$

$$\mathbf{X}_b := \left[\begin{array}{c:c} \mathbf{G}_b & \mathbf{T}_b \end{array}\right]. \tag{3.14}$$

Algorithm 1 CMFS equation (3.9)

1: Scaling: scale the context matrix $\mathbf{T}_o$ with ν – important for accurate prediction.
2: Initialization: impute missing entries in $\mathbf{G}_m$ with the average of the observed grades in $\mathbf{G}_o$; same for $\mathbf{T}_m$; $\mathbf{b}_s$=$\mathbf{b}_c$=$\mathbf{t}_s$=$\mathbf{t}_c$=$\mathbf{0}$
3: **Repeat**
4: Update $\mathbf{G}_b$ and $\mathbf{T}_b$ using (3.13)
5: $\mathbf{A}, \mathbf{B}_c$ (defined in (3.4)) $\leftarrow$ SVD($\mathbf{X}_b$)
6: Update grade bias vectors $\mathbf{b}_s = \frac{(\mathbf{G}-\mathbf{A}\mathbf{B}_1^T-\mathbf{1}\mathbf{b}_c^T)\mathbf{1}}{m}$, and $\mathbf{b}_c = \frac{(\mathbf{G}-\mathbf{A}\mathbf{B}_1^T-\mathbf{b}_s\mathbf{1}^T)^T\mathbf{1}}{n}$,
7: Update context bias vectors $\mathbf{t}_s = \frac{(\mathbf{G}-\mathbf{A}\mathbf{B}_2^T-\mathbf{1}\mathbf{t}_c^T)\mathbf{1}}{m}$, and $\mathbf{t}_c = \frac{(\mathbf{G}-\mathbf{A}\mathbf{B}_2^T-\mathbf{t}_s\mathbf{1}^T)^T\mathbf{1}}{n}$
8: Impute missing values of $\mathbf{G}$ by updating $\mathbf{G}_m = \mathbf{A}\mathbf{B}_1^T + \mathbf{b}_s\mathbf{1}^T + \mathbf{1}\mathbf{b}_c^T$
9: Impute missing values of $\mathbf{T}$ by updating $\mathbf{T}_m = \mathbf{A}\mathbf{B}_2^T + \mathbf{t}_s\mathbf{1}^T + \mathbf{1}\mathbf{t}_c^T$
10: **until convergence** (the normalized difference of the cost function of two successive iterations $< \epsilon$)

Chapter 4

Low-Rank Tensor Factorization (LRTF)

Considerable work has been done in recent years towards incorporating context with the goal of building more personalized recommender systems, for which context turns out playing an important role. The context-aware CF model based on *Tensor Factorization* introduced in [21] models data as a User × Item × Context tensor. A similar approach was used in the context of grade prediction by modeling the student performance data as a three-mode tensor (student × task × time) [7]. Although this technique improves the prediction accuracy by exploiting the context, it increases the sparsity of the data by introducing context as a new mode, therefore increasing the rank required for accurate approximation, especially without imputation. The reason is that a very sparse tensor with a random sparsity pattern requires rank in the order of the number of nonzero elements, as a rank-one tensor is spent explaining each available element, and there is very slow error "roll-off" as one increases the model's rank.

We propose instead a Low-Rank Tensor Factorization (LRTF) model, where a context matrix (e.g., time $\mathbf{T}_o$) is introduced as another slab behind the main historical grades matrix $\mathbf{G}_o$. The rest of this chapter is structured as follows. In section 4.1, we review Candecomp/Parafac decomposition, our LRTF formulation is explained in section 4.2, and an algorithm to solve this formulation is presented in section 4.3.

4.1 Candecomp/Parafac (CP) Decomposition

In this section we summarize the basics of CP decomposition as it is essential in our LRTF formulation. Decomposing a three-way tensor as a sum of outer products of rank-one three-way tensors as a data analysis technique was proposed independently by Carroll and Chang [29] (they called it Candecomp) and Harshman [30] (who called it Parafac). The CP decomposes a three-way array $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ into a sum of F rank-one tensors [31], i.e.,

$$\underline{\mathbf{X}} \approx \sum_{f=1}^{F} \mathbf{a}_f \circ \mathbf{b}_f \circ \mathbf{c}_f \tag{4.1}$$

Where $\circ$ is the outer product, F is a positive integer, $\mathbf{a}_f \in \mathbb{R}^I$, $\mathbf{b}_f \in \mathbb{R}^J$, and $\mathbf{c}_f \in \mathbb{R}^K$ [1]. A CP solution is usually expressed in terms of the factor matrices $\mathbf{A} \in \mathbb{R}^{I \times F}$, $\mathbf{B} \in \mathbb{R}^{J \times F}$, and $\mathbf{C} \in \mathbb{R}^{K \times F}$, which have the vectors $\mathbf{a}_f$, $\mathbf{b}_f$ and $\mathbf{c}_f$ as columns, respectively, i.e., $\mathbf{A} = [\mathbf{a}_1 \ \ \mathbf{a}_2 \ \ ... \ \ \mathbf{a}_F]$ and likewise for $\mathbf{B}$ and $\mathbf{C}$. Now, let $\mathbf{X}_k$ denote the kth slice (frontal 'slab') of $\underline{\mathbf{X}}$. Then (4.1) can be written as:

$$\mathbf{X}_k \approx \mathbf{A}\mathbf{C}_k\mathbf{B}^T \tag{4.2}$$

Where $\mathbf{C}_k$ is a diagonal matrix holding the kth row of $\mathbf{C}$ on its diagonal [31]. Denote the kth row of $\mathbf{C}$ by $\mathbf{c}_k$. Then $\mathbf{C}_k = Diag(\mathbf{c}_k)$. Notations presented in this section will be used for the formulation and algorithm of LRTF model.

4.2 LRTF Model Formulation

As a first step, we model the grades matrix $\mathbf{G}$ which includes the observed and missing grades as defined in (3.2) and its context $\mathbf{T}$ as a tensor with two frontal slices – $\mathbf{G}$ in front, and the context matrix we wish to use behind it [1] , i.e.,

$$\mathbf{X}_1 = \mathbf{G}, \quad \mathbf{X}_2 = \mathbf{T} \tag{4.3}$$

Therefore $\underline{\mathbf{X}} \in \mathbb{R}^{n \times m \times 2}$ in the case where only one context is added behind $\mathbf{G}$, where the first mode describes n students, the second mode describes m courses, and the third

[1] If we desire to exploit more than one contextual information, a third slice of context can be added, etc.

mode describes the number of contexts added to the grade matrix. Modeling data in frontal slices is illustrated in figure 4.1. The advantage of this modeling is that adding a context doesn't increase sparsity, it maintains exactly the same sparsity pattern as in $\mathbf{G}_o$.

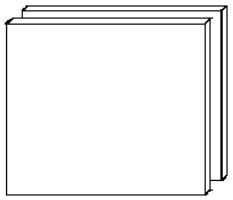

Figure 4.1: Illustration of modeling data in LRTF [1]

In LRTF model, we typically use an alternating optimization algorithm to estimate the factor matrices $\mathbf{A} \in \mathbb{R}^{n \times F}$, $\mathbf{B} \in \mathbb{R}^{m \times F}$ and $\mathbf{C} \in \mathbb{R}^{2 \times F}$ of the CP decomposition of $\underline{\mathbf{X}}$. After every update of $\mathbf{A}$, $\mathbf{B}$ and $\mathbf{C}$ we use them to impute for missing grades and context. Overall, LRTF and imputation can be formulated as follows:

$$\min_{\mathbf{G}_m, \mathbf{T}_m, \mathbf{A}, \mathbf{B}, \mathbf{C}} \quad \|\mathbf{W} \circledast \mathbf{G}_o + \overline{\mathbf{W}} \circledast \mathbf{G}_m - \mathbf{A}\mathbf{C}_1\mathbf{B}^T\|_F^2 + \|\mathbf{W} \circledast \mathbf{T}_o + \overline{\mathbf{W}} \circledast \mathbf{T}_m - \mathbf{A}\mathbf{C}_2\mathbf{B}^T\|_F^2 \tag{4.4}$$

Clearly, grade matrix and context share the same $\mathbf{A}$ and $\mathbf{B}$ factors in the above LRTF formulation. Therefore, in order for two students who take the same classes to be more predictive for each other, they must share similar time profiles (or any other context) as well. Solving problem (4.4) requires small rank F due to the imputation that occurs within every iteration (the update of $\mathbf{G}_m$ and $\mathbf{T}_m$) and the fact that sparsity isn't affected by adding context slabs. In recommender systems, accounting for user and item biases as variables improves the accuracy of rating prediction [18], [28]. This is the case for grade prediction as well as in [6], [3], [7]. Similar to CMFS, and CMFC, in our experience, adding student and course biases always improves the accuracy of grades prediction for LRTF as well. After accounting for the student and course biases vectors for the grade matrix and its context, equation 4.4 becomes:

$$\begin{aligned} \min_{\mathbf{G}_m, \mathbf{T}_m, \mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{b}_s, \mathbf{b}_c, \mathbf{t}_s, \mathbf{t}_c} \quad & \|\mathbf{W} \circledast \mathbf{G}_o + \overline{\mathbf{W}} \circledast \mathbf{G}_m - \mathbf{A}\mathbf{C}_1\mathbf{B}^T - \mathbf{b}_s\mathbf{1}^T - \mathbf{1}\mathbf{b}_c^T\|_F^2 \\ & + \|\mathbf{W} \circledast \mathbf{T}_o + \overline{\mathbf{W}} \circledast \mathbf{T}_m - \mathbf{A}\mathbf{C}_2\mathbf{B}^T - \mathbf{t}_s\mathbf{1}^T - \mathbf{1}\mathbf{t}_c^T\|_F^2 \end{aligned} \tag{4.5}$$

4.3 LRTF Algorithm

We present the algorithm that solves the LRTF formulation to estimate the CP factor matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$, which are used to predict the grades and their context as follows:

$$\widehat{g}_{i,j} = b_s(i) + b_c(i) + \mathbf{A}(i,:)\mathbf{C}_1\mathbf{B}^T(:,j), \qquad \widehat{t}_{i,j} = t_s(i) + t_c(i) + A(i,:)\mathbf{C}_2\mathbf{B}^T(:,j) \quad (4.6)$$

Recall $\mathbf{X}_b$ defined in (3.14) and define:

$$\mathbf{Y}_b := \left[\begin{array}{c:c} \mathbf{G}_b^T & \mathbf{T}_b^T \end{array} \right]. \quad (4.7)$$

Algorithm 2 LRTF equation (4.5)

1: Scaling: scale the context matrix $\mathbf{T}_o$ with ν – important for accurate prediction.
2: Initialization: impute missing entries in $\mathbf{G}_m$ with the average of the observed grades in $\mathbf{G}_o$; same for $\mathbf{T}_m$; $\mathbf{b}_s$=$\mathbf{b}_c$=$\mathbf{t}_s$=$\mathbf{t}_c$=$\mathbf{0}$.
3: Initialize for $\mathbf{A}, \mathbf{B}, \mathbf{C}$ using *N-way Toolbox* to provide a better initialization as they use algebraic methods (direct trilinear decomposition) for initialization.
4: **Repeat**
5: Update $\mathbf{G}_b$ and $\mathbf{T}_b$ using (3.13)
6: Update $\mathbf{A} \leftarrow \mathbf{X}_b([\mathbf{C}_1\mathbf{B}^T, \mathbf{C}_2\mathbf{B}^T])^\dagger$
7: Update $\mathbf{B} \leftarrow \mathbf{Y}_b([\mathbf{C}_1\mathbf{A}^T, \mathbf{C}_2\mathbf{A}^T])^\dagger$
8: Update $\mathbf{c}_1 \leftarrow \text{vec}(\mathbf{G}_b)(\mathbf{B} \odot \mathbf{A})^\dagger$
9: Update $\mathbf{c}_2 \leftarrow \text{vec}(\mathbf{T}_b)(\mathbf{B} \odot \mathbf{A})^\dagger$
10: Update grade bias vectors $\mathbf{b}_s = \frac{(\mathbf{G}-\mathbf{A}\mathbf{C}_1\mathbf{B}^T-\mathbf{1}\mathbf{b}_c^T)\mathbf{1}}{m}$, and $\mathbf{b}_c = \frac{(\mathbf{G}-\mathbf{A}\mathbf{C}_1\mathbf{B}^T-\mathbf{b}_s\mathbf{1}^T)^T\mathbf{1}}{n}$,
11: Update context bias vectors $\mathbf{t}_s = \frac{(\mathbf{G}-\mathbf{A}\mathbf{C}_2\mathbf{B}^T-\mathbf{1}\mathbf{t}_c^T)\mathbf{1}}{m}$, and $\mathbf{t}_c = \frac{(\mathbf{G}-\mathbf{A}\mathbf{C}_2\mathbf{B}^T-\mathbf{t}_s\mathbf{1}^T)^T\mathbf{1}}{n}$
12: Impute missing values of $\mathbf{G}$ by updating $\mathbf{G}_m = \mathbf{A}\mathbf{C}_1\mathbf{B}^T + \mathbf{b}_s\mathbf{1}^T + \mathbf{1}\mathbf{b}_c^T$
13: Impute missing values of $\mathbf{T}$ by updating $\mathbf{T}_m = \mathbf{A}\mathbf{C}_2\mathbf{B}^T + \mathbf{t}_s\mathbf{1}^T + \mathbf{1}\mathbf{t}_c^T$
14: **until convergence** (the normalized difference of the cost function of two successive iteration $< \epsilon$)

Where $(\mathbf{Z})^\dagger$ is Moore-Penrose inverse of a matrix $\mathbf{Z}$

Chapter 5

Experimental Design

In this chapter, we explain the setup of the experiments we performed to test the three proposed models, CMFS, CMFC, and LRTF. We describe the features of grade datasets and explain the construction of the context we used as a side information in section 5.1. Our baseline where no context is added and other methods used for comparison are summarized in section 5.2. In section 5.3, we explain the different test sets used for testing alongside with the metrics used for evaluation. Finally, in the last section 5.4, we clarify the values of parameters and strategies used for model selection for all methods.

5.1 Dataset and Context

The experimental results were obtained using actual historical grade data for College of Science and Engineering (CSE) students at the University of Minnesota. Results are shown for experimentation on two different datasets. Dataset 1 contains all the grades of students of CSE for any course they have taken between Fall 2002 until Fall 2013, including courses offered by other colleges. This dataset includes $n = 10,245$ students and $m = 3712$ courses. There are $244,086$ observed grades in total, hence the percentage of known entries in $\mathbf{G}_o$ using this data is only 0.642%. Dataset 2 includes grades of students of CSE strictly for courses under one of the following departments: Aerospace

Engineering, Biomedical Engineering, Chemical Engineering, Chemistry, Civil Engineering, Computer science, Electrical Engineering, Material Science, Mathematics, Mechanical Engineering, Physics, and Statistics. Thus, it has less courses $m = 941$ but more students $n = 12,938$ as it extends over longer time, from Fall 2002 to Spring 2015. This dataset is less sparse as it has $269,073$ observed entries in $\mathbf{G}_o$, which leads to 2.21% density. For both datasets, we create the students $\times$ courses matrix of observed grades $\mathbf{G}_o$ encoded by mapping $[F, D, D+, C-, C, C+, B-, B, B+, A-, A]$ to numeric grades $[0, 1, 1.33, 1.67, 2, 2, 33, 2, 67, 3, 3.33, 3.67, 4]$, respectively. The $\mathbf{G}_o$ matrices for the two datasets are visualized in figure 5.1 through which we can view their sparsity. Table 5.1 summarizes the two datasets and their features.

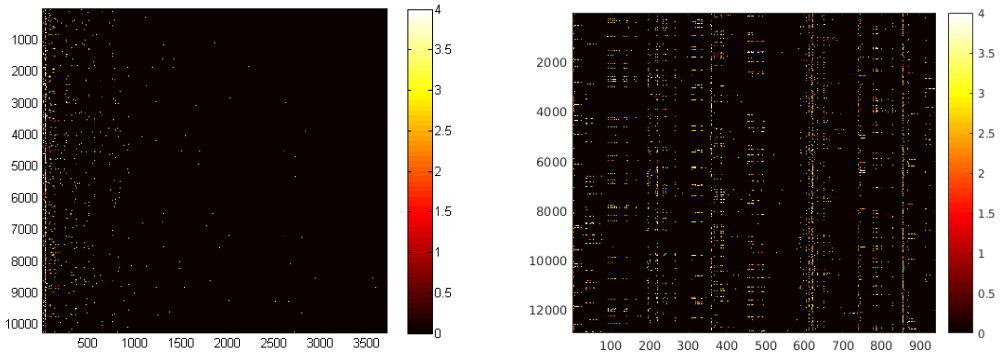

Figure 5.1: $\mathbf{G}_o$ of Dataset 1 (left) and $\mathbf{G}_o$ of Dataset 2 (right)

.

Table 5.1: Grade Datasets

Features	Dataset 1	Dataset 2
# of students (n)	10,245	12,938
# of courses (m)	3712	941
# of observations	244,086	269,073
sparsity	99.34%	97.79%
period	Fall 2002 - Fall 2013	Fall 2002 - Spring 2015

5.1.1 Construction of Context Matrix

For each observed dataset, we experimented using CMFS, CMFC, and LRTF with two different types of contextual information: absolute time $\mathbf{T}_o$ indicating the semester in which a grade was earned; and student experience $\mathbf{E}_o$, reflecting seniority when taking a course. Throughout our simulations, we used these two contexts individually and together with each one of the proposed models. We found out that using the time context individually is the most informative as it provides the best grade prediction. We think that adding the experience context $\mathbf{E}_o$ besides the time $\mathbf{T}_o$ didn't improve the prediction due to the correlation between these two context – student experience can be inferred from (is implicit in) the absolute time information. Therefor, we show results in the following chapter for simulations conducted using the time context $\mathbf{T}_o$ alongside with observed grades $\mathbf{G}_o$.

Each professor has her/his own way of grading even when they teach the same course. Moreover, year-to-year student cohort variation may cause a given professor to grade the same material differently through the years. By encoding every semester in $\mathbf{T}_o$ with a distinct digit, the various offerings of the same course can be distinguished. The details of how each model exploits this context can be found in chapters 3 and 4. $\mathbf{T}_o$ is the student $\times$ course time matrix, constructed in exactly the same way as the grade matrix. We map semesters to consecutive integer numbers starting from 1. For instance, we encode Fall 2002 as 1, Spring 2003 as 2, Summer 2003 as 3 and so on. Hence, the maximum value in $\mathbf{T}_o$ is 34 for Dataset 1 and 38 for Dataset 2. An illustrative example of this mapping is shown in figure 5.2.

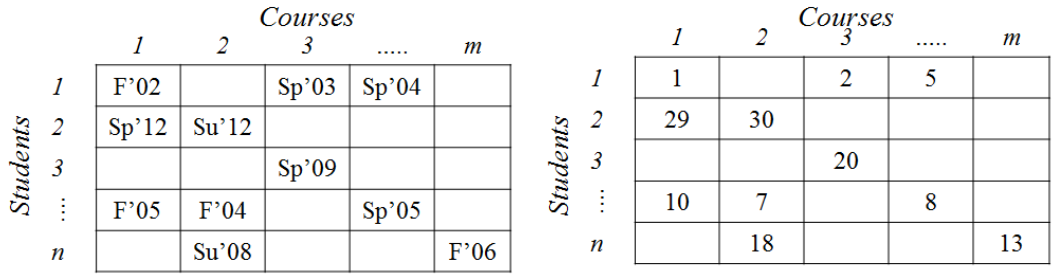

Figure 5.2: Semesters in a matrix (left) and illustrative example of $\mathbf{T}_o$ (right)

.

5.2 Comparison with Other Methods

We compare the performance of CMFS, CMFC and LRTF with our baseline where no context is used and the following methods:

Baseline: Factoring the grade matrix $\mathbf{G}$ as defined in (3.2) with iterative imputation and grade bias terms in the same way as the proposed models, but without including any context. The baseline is formulated as follows:

$$\min_{\mathbf{G}_m,\mathbf{A},\mathbf{B},\mathbf{b}_s,\mathbf{b}_c} \quad \|\mathbf{W} \circledast \mathbf{G}_o + \overline{\mathbf{W}} \circledast \mathbf{G}_m - \mathbf{A}\mathbf{B}^T - \mathbf{b}_s\mathbf{1}^T - \mathbf{1}\mathbf{b}_c^T\|_F^2 \tag{5.1}$$

Where $\mathbf{A}$ and $\mathbf{B}$ are the low-rank factors of $\mathbf{G}$.

Matrix Factorization: The MF approach described in [6]. To match the MF model with our notation, we will write its formulation as follows:

$$\min_{\mu,\mathbf{b}_s,\mathbf{b}_c,\mathbf{A},\mathbf{B}} \quad \sum_{g_{i,j}\in\mathbf{G}_o} (g_{i,j} - \mu - b_s(i) - b_c(j) - \mathbf{a}_i\mathbf{b}_j^T)^2 + \lambda(\|\mathbf{A}\|_F^2 + \|\mathbf{B}\|_F^2 + \|\mathbf{b}_s\|_2^2 + \|\mathbf{b}_c\|_2^2) \tag{5.2}$$

Where μ is a global bias, $\mathbf{b}_s \in \mathbb{R}^n$ and $\mathbf{b}_c \in \mathbb{R}^m$ are the student and course bias vectors, respectively, $\mathbf{A} \in \mathbb{R}^{n\times F}$ and $\mathbf{B} \in \mathbb{R}^{m\times F}$ are the latent representations, and F is the model rank.

BiasOnly: As described in [6], only global and local bias terms are considered in equation (5.2). Biases are estimated using MF formulation above with $F = 0$.

5.3 Test Sets and Evaluation Metrics

To assess the proposed models, we test their prediction of grades in a test set, $\mathbf{G}_{test}$, which is either: 1) the last semester or 2) randomly selected 10% of the observed grades in $\mathbf{G}_o$. Note that when we test on the last semester for Dataset 1, we discard Fall 2013 and Summer 2013 and test on Spring 2013, as Summer 2013 and Fall 2013 don't have enough observed grades (Dataset 1 was recorded before the end of the Fall 2013 semester). The cardinality (N) of the test set for theses two cases for Dataset 1 and Dataset 2 is shown in table 5.2. We denote the time context associated with $\mathbf{G}_{test}$ as $\mathbf{T}_{test}$. Grades in $\mathbf{G}_{test}$ and their corresponding context $\mathbf{T}_{test}$ are excluded from $\mathbf{G}_o$ and $\mathbf{T}_o$, respectively.

The grade prediction accuracy is evaluated using the *Root Mean Squared Error* (RMSE) and *Mean Absolute Error* (MAE) of $\mathbf{G}_{test}$ as defined in (5.3). Same metrics are used to calculate the accuracy of predicting context in $\mathbf{T}_{test}$.

$$RMSE = \sqrt{\frac{1}{N} \sum_{g_{s,c} \in \mathbf{G}_{test}} (g_{s,c} - \widehat{g_{s,c}})^2}, \qquad MAE = \frac{\sum_{g_{s,c} \in \mathbf{G}_{test}} |g_{s,c} - \widehat{g_{s,c}}|}{N} \tag{5.3}$$

Table 5.2: Cardinality of Test Sets (N)

$\lvert\mathbf{G}_{test}\rvert$	Dataset 1	Dataset 2
last semester	14,723	9,176
random 10%	24,407	26,908

5.4 Model Selection and Training

Parameters in CMFS, CMFC, LRTF, and methods in section 5.2 are selected based on the performance of these models on a validation set $\mathbf{G}_{val}$, which is randomly selected 10% of observed grade data $\mathbf{G}_o$ not including the test set $\mathbf{G}_{test}$.

For all models, we perform a greedy search on the best model rank F. For our models and baseline, we show results in the next chapter for the best three model ranks for each dataset. For MF method (as described in section 5.2) we show results associated with the best rank.

Similarly, we search for the best value of ν in our modeling which is used to scale the context matrix. Best ν varies depending on the type of test set $\mathbf{G}_{test}$ we are testing on as described in section 5.3. We found that $\nu = \frac{1}{2}$ gives the best prediction when $\mathbf{G}_{test}$ is the last semester, while $\nu = \frac{1}{10}$ gives the best prediction in the case of testing on $\mathbf{G}_{test}$ as randomly selected 10% of observed data. In MF and BiasOnly methods, for each dataset and $\mathbf{G}_{test}$, we search for the best λ which is the regularization parameter.

While the cost function of our modeling is monotonically improving with iterations, the prediction RMSE/MAE is not. We found that the RMSE is a convex (U-shaped) function of the number of iterations when we solve for the proposed models. The stopping criterion used to terminate model fitting iterations is based on the cost function

as explained in algorithms 1 and 2. We monitor the prediction RMSE on $\mathbf{G}_{val}$ and terminate when it starts rising again.

Once we fine-tuned parameters based on $\mathbf{G}_{val}$, we train our models using the provided algorithmic procedures on the grade matrix $\mathbf{G}_o$ and its context $\mathbf{T}_o$. Then, grades in the test set $\mathbf{G}_{test}$ and their context $\mathbf{T}_{test}$ are predicted using the 'last model' fitted before we terminate the iterations by equations (3.11), (3.12) and (4.6) for CMFS, CMFC, and LRTF, respectively. The prediction error metrics are then calculated using the formulas provided in section 5.3.

Chapter 6

Experimental Results

In this chapter we show the performance of CMFS, CMFC, and LRTF models which exploit the time context $\mathbf{T}_o$ alongside with $\mathbf{G}_o$ and compare them to our baseline, Matrix Factorization, and BiasOnly models which use only the observed grade matrix $\mathbf{G}_o$ as described in section 5.2. The performance is measured in terms of the accuracy of predicting grades in the test set $\mathbf{G}_{test}$. For each dataset, for our models and the baseline, we show the results with the best three model ranks and compare them with BiasOnly and Matrix Factorization (MF) with its the best rank. We also show how well our methods predict the time context in $\mathbf{T}_{test}$. As we test on two different test sets, last semester and randomly missing grades, we dedicate one section for each test set type.

6.1 Prediction of Last Semester

The performance results shown in this section are calculated on the last semester test set. Table 6.1 shows the prediction error of students' final grades evaluated by RMSE and MAE for our models and the methods in comparison. Amongst our models, CMFC with rank 1 works best for Dataset 1, while CMFS with rank 1 gives the smallest error for Dataset 2. The results on the last semester were unexpected, as our methods with their best rank outperform BiasOnly, but they don't improve the baseline or MF with rank 1. This might be due to the nature of the dataset we are using (the best models have very low rank, and student and course BiasOnly predicts almost as well as any other method, which indicates either very simple or very challenging 'extreme' data); or the

fact that we resort to alternating optimization as we can't fit the models to optimality. Another explanation for these results is that we tune model parameters for randomly missing validation set and test on the last semester test set which have different natures. To resolve this, we used the semester before the last one to select models and it didn't work well as different courses usually offered in the fall and spring semesters. It is worth mentioning here that imputing for missing grades while fitting the model helps, as this is the main difference between our baseline and MF – note the results of baseline vs. MF in table 6.1. For each dataset, we underlined the smallest error produced by our models and the baseline to make it easier for the reader to compare.

Table 6.1: Prediction Error of $\mathbf{G}_{test}$ (last semester)

	Dataset 1			Dataset 2		
Method	Latent Factors	RMSE	MAE	Latent Factors	RMSE	MAE
Baseline	1	<u>0.6659</u>	0.4728	1	<u>0.6628</u>	0.4778
	2	0.6706	0.4784	2	0.6709	0.4880
	3	0.6884	0.4964	3	0.6732	0.4919
MF	1	0.6702	0.4667	1	0.6638	0.4737
BiasOnly	—	0.6824	0.4800	—	0.6712	0.4843
CMFS	1	0.6818	0.4865	1	<u>0.6672</u>	0.4831
	2	0.6877	0.4904	2	0.6694	0.4858
	3	0.6925	0.4972	3	0.6701	0.4859
CMFC	1	<u>0.6716</u>	0.4747	1	0.6674	0.4816
	2	0.6781	0.4855	2	0.6699	0.4863
	3	0.6879	0.4945	3	0.6756	0.4919
LRTF	7	0.6797	0.4820	8	0.6680	0.4809
	8	0.6780	0.4816	9	0.6689	0.4816
	9	0.6778	0.4825	10	0.6698	0.4830

In table 6.2, we show the prediction error of our modeling for the time at which grades in the last semester test set were obtained. Note that the correct time values in $\mathbf{T}_{test}$ which we are predicting are the same which correspond to last semester. Recall that we scale $\mathbf{T}_o$ by $\nu = \frac{1}{2}$, hence every semester adds 0.5 including summers. CMFS

with rank 1 can predict time with less than three semesters error – roughly within one year error as we account for summer semesters. We should mention that although the smallest error in table 6.2 is RMSE = 1.3329, this is not the best time prediction that can be obtained. The reason is that we fine-tune the model based on the prediction of grades as it is our main interest and then predict for the context.

Table 6.2: Prediction Error of $\mathbf{T}_{test}$ (last semester)

	Dataset 1			Dataset 2		
Method	Latent Factors	RMSE	MAE	Latent Factors	RMSE	MAE
CMFS	1	1.7599	1.2662	1	<u>1.3329</u>	0.8273
	2	2.1210	1.3918	2	2.0686	1.4342
	3	2.2021	1.4660	3	2.1386	1.4946
CMFC	1	<u>1.6435</u>	1.1009	1	1.3374	0.8335
	2	2.1820	1.4429	2	2.0975	1.4699
	3	2.3025	1.5728	3	2.2731	1.6980
LRTF	7	2.4888	1.6791	8	2.1249	1.2882
	8	2.5199	1.7098	9	2.1698	1.3402
	9	2.6359	1.7910	10	2.2195	1.3857

6.2 Prediction of Randomly missing 10% of Data

In this section we show the prediction error of testing on randomly missing 10% of data in the same manner as the previous section 6.1. Table 6.3 shows the prediction error for the proposed models, baseline, MF, and BiasOnly. For this test set, the best of our models outperform MF and BiasOnly methods and improve the baseline prediction. CMFC gives the smallest error among our models for Dataset 1, while LRTF provides the best prediction for Dataset 2. We underlined the smallest error provided by the proposed models and the best of the other methods to highlight the improvement.

For this test set, our model predict the time with smaller error than in the case of last semester as clear in table 6.4. We scale $\mathbf{T}_o$ by $\nu = \frac{1}{10}$ in this case, hence every semester encoded with 0.1 including summers. For Dataset 2, the time was predicted using CMFS with less than two semesters error, RMSE = 0.1941.

Table 6.3: Prediction Error of $\mathbf{G}_{test}$ (randomly missing)

	Dataset 1			Dataset 2		
Method	Latent Factors	RMSE	MAE	Latent Factors	RMSE	MAE
Baseline	1	0.6150	0.4460	1	0.5848	0.4355
	2	<u>0.6108</u>	0.4399	2	0.5866	0.4369
	3	0.6227	0.4468	3	0.5856	0.4344
MF	1	0.6130	0.4431	1	<u>0.5812</u>	0.4351
BiasOnly	—	0.6257	0.4603	—	0.5876	0.4424
CMFS	1	0.6170	0.4474	1	0.5840	0.4353
	3	0.6119	0.4415	2	0.5843	0.4358
	4	0.6199	0.4475	4	0.5845	0.4352
CMFC	1	<u>0.6093</u>	0.4414	1	0.5819	0.4331
	2	0.6101	0.4399	2	0.5819	0.4327
	3	0.6187	0.4455	3	0.5844	0.4342
LRTF	3	0.6118	0.4428	6	<u>0.5787</u>	0.4305
	4	0.6101	0.4412	7	0.5803	0.4321
	5	0.6117	0.4429	8	0.5814	0.4323

Table 6.4: Prediction Error of $\mathbf{T}_{test}$ (randomly missing)

	Dataset 1			Dataset 2		
Method	Latent Factors	RMSE	MAE	Latent Factors	RMSE	MAE
CMFS	1	<u>0.2157</u>	0.1489	1	<u>0.1941</u>	0.1295
	3	0.2401	0.1620	2	0.2043	0.1371
	4	0.2581	0.1754	4	0.2347	0.1610
CMFC	1	0.2264	0.1622	1	0.2051	0.1421
	2	0.2409	0.1724	2	0.2114	0.1457
	3	0.2505	0.1741	3	0.2161	0.1470
LRTF	3	0.2369	0.1632	6	0.2197	0.1445
	4	0.2442	0.1705	7	0.2275	0.1493
	5	0.2529	0.1731	8	0.2300	0.1511

Chapter 7

Conclusion

In this work, we extended the traditional MF method in the context of CF to incorporate contextual information in a different way than is commonly done in the recommender system literature. Utilizing *matrix factorization* and *tensor factorization* we proposed models that allow for flexible integration of side information, *without amplifying sparsity / increasing the needed model rank*. They also can handle the "cold start" problem when predicting for a certain context, such as next semester. These models share the feature of factoring the grade matrix and the context with a common factor to exploit the context in which a grade was earned. Algorithms to solve the presented formulations were provided.

These model were tested on actual grade data obtained from the University of Minnesota with time context measured in semesters. We conducted a careful comparison with our baseline and existing methods that are used in the setting of predicting student performance. Comparisons were made on the prediction of grades in two types of test set, last semester and randomly missing grades. Although the results were surprising for last semester testing as our modeling didn't improve the baseline or MF, they showed a promising prediction improvement for the case of testing with randomly missing grades. In the case of predicting randomly missing grades, the smallest error is provided by one of the proposed models, CMFC for Dataset 1 and LRTF for Dataset 2.

Another aspect that the presented modeling could be used for is context prediction, e.g., course enrollment forecasting. For the randomly missing grade prediction, the time was predicted using CMFS with less than two semesters error, RMSE = 0.1941.

References

[1] T. G. Kolda and B. W. Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.

[2] R. S. Baker and K. Yacef. The state of educational data mining in 2009: A review and future visions. *JEDM-Journal of Educational Data Mining*, 1(1):3–17, 2009.

[3] A. Elbadrawy, R.S. Studham, and G. Karypis. Collaborative multi-regression models for predicting students' performance in course activities. In *Proceeding of the 5th International Conference on Learning Analytics and Knowledge*, LAK 2015, pages 103–107, Poughkeepsie, NY, Mar 2015.

[4] M. Sweeney, J. Lester, and H. Rangwala. Next-term student grade prediction. In *IEEE International Conference on Big Data*, pages 970–975, Santa Clara, CA, Nov 2015. IEEE.

[5] R. Barber and M. Sharkey. Course correction: using analytics to predict course success. In *Proceedings of the 2nd international conference on learning analytics and knowledge*, pages 259–262, Vancouver, BC, Apr 2012. ACM.

[6] A. Polyzou and G. Karypis. Grade prediction with course and student specific models. In *Proceeding of the 20th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining*, pages 89–101, Auckland, New Zealand, Apr 2016. Springer.

[7] N. Thai-Nghe, T. Horvth, and L. Schmidt-Thieme. Factorization models for forecasting student performance. In *Proceeding of the 4th International Conference on Educational Data Mining*, pages 11–20, Eindhoven, The Netherlands, Jul 2011.

[8] A Toscher and Michael Jahrer. Collaborative filtering applied to educational data mining. *KDD cup*, 2010.

[9] N. Thai-Nghe, A. L. Drumond Krohn-Grimberghe, and L. Schmidt-Thieme. Recommender system for predicting student performance. In *Proceeding of the 1st Workshop on Recommender Systems for Technology Enhanced Learning*, pages 2811–2819, Barcelona, Spain, Sep 2010.

[10] M.A. Al-Barrak and M. Al-Razgan. Predicting students final gpa using decision trees: a case study. *International Journal of Information and Education Technology*, 6(7):528, 2016.

[11] H. Bydžovská. *Are Collaborative Filtering Methods Suitable for Student Performance Prediction?* Springer International Publishing, Coimbra, Portugal, Sep 2015.

[12] K. E. Arnold and M. D. Pistilli. Course signals at purdue: Using learning analytics to increase student success. In *Proceedings of the 2Nd International Conference on Learning Analytics and Knowledge*, LAK '12, pages 267–270, Vancouver, BC, 2012. ACM.

[13] N. Thai-Nghe, L. Drumond, T. Horváth, and L. Schmidt-Thieme. Using factorization machines for student modeling. In *UMAP Workshops*, 2012.

[14] T. Denley. Course recommendation system and method, January 2013. US Patent App. 13/441,063.

[15] S. Ray and A. Sharma. A collaborative filtering based approach for recommending elective courses. In *International Conference on Information Intelligence, Systems, Technology and Management*, pages 330–339, Gurgaon, India, Mar 2011. Springer.

[16] A. Elbadrawy and G. Karypis. Domain-aware grade prediction and top-n course recommendation. Boston, MA, Sep 2016. ACM.

[17] H. Cen, K. R. Koedinger, and B. Junker. Learning factors analysis–a general method for cognitive model evaluation and improvement. In *International Conference on Intelligent Tutoring Systems*, pages 164–175. Springer, 2006.

[18] P. Melville and V. Sindhwani. Recommender systems. In *Encyclopedia of machine learning*, pages 829–838. Springer, 2011.

[19] E. J. Candès and B. Recht. Exact matrix completion via convex optimization. *Foundations of Computational mathematics*, 9(6):717–772, 2009.

[20] A. Paterek. Improving regularized singular value decomposition for collaborative filtering. In *Proceedings of KDD Cup and Workshop*, pages 5–8, San Jose, CA, Aug 2007.

[21] A. Karatzoglou, X. Amatriain, L. Baltrunas, and N. Oliver. Multiverse recommendation: N-dimensional tensor factorization for context-aware collaborative filtering. In *Proceeding of the 4th ACM Conference on Recommender Systems*, RecSys '10, pages 79–86, Barcelona, Spain, Sep 2010. ACM.

[22] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin. Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Transactions on Information Systems (TOIS)*, 23(1):103–145, 2005.

[23] K. Oku, S. Nakajima, J. Miyazaki, and S. Uemura. Context-aware svm for context-dependent information recommendation. In *Proceedings of the 7th international Conference on Mobile Data Management*, page 109, Nara, Japan, May 2006. IEEE.

[24] L. Baltrunas, B. Ludwig, and F. Ricci. Matrix factorization techniques for context aware recommendation. In *Proceedings of the fifth ACM conference on Recommender systems*, pages 301–304. ACM, 2011.

[25] G. Adomavicius and A. Tuzhilin. Context-aware recommender systems. In F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, editors, *Recommender Systems Handbook*. Springer, 2010.

[26] L. Xiong, X. Chen, T. Huang, J. Schneider, and J. G. Carbonell. Temporal collaborative filtering with bayesian probabilistic tensor factorization. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, Columbus, OH, Apr 2010. SIAM.

[27] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor. *Recommender Systems Handbook*. Springer, 2010.

[28] R. M. Bell, Y. Koren, and C. Volinsky. The BellKor 2008 Solution to the Netflix Prize, year = 2008. Technical report.

[29] J. D. Carroll and J. Chang. Analysis of individual differences in multidimensional scaling via an n-way generalization of.

[30] R.A. Harshman. Foundations of the parafac procedure: models and conditions for an explanatory multimodal factor analysis. *UCLA Working Papers in Phonetics 16*, pages 1–84, 1970.

[31] A. Stegeman and N. D Sidiropoulos. On kruskals uniqueness condition for the candecomp/parafac decomposition. *Linear Algebra and its applications*, 420(2):540–552, 2007.