Acad-GPS:
Academic Grade Prediction System

By
Muhammad Farooq Mazhar
NUST201463991MSEECS61214F

Supervisor
Dr. Hassan Aqeel Khan
Department of Electrical Engineering

A thesis submitted in partial fulfillment of the requirements for the degree of Masters of Science in Electrical Engineering (MS EE)

In
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Islamabad, Pakistan.

(July 2018)

Approval

It is certified that the contents and form of the thesis entitled "**Acad-GPS: Academic Grade Prediction System**" submitted by **Muhammad Farooq Mazhar** have been found satisfactory for the requirement of the degree.

Advisor: **Dr. Hassan Aqeel Khan**
Signature: ____________________

Date: ____________________

Committee Member 1: **Dr. Khawar Khurshid**

Signature: ____________________
Date: ____________________

Committee Member 2: **Dr. Sajid Saleem**

Signature: ____________________
Date: ____________________

Committee Member 3: **Dr. Ahmad Salman**

Signature: ____________________
Date: ____________________

Abstract

Students that fail courses have to retake those courses, often during regular semesters, sometimes during summer semesters. Every summer, NUST has to keep their departments open to offer courses to allow students to graduate from their programs without incurring excessive delays. The cost of engaging faculty to stay back for the summer is an additional financial burden on universities' budgets. There is a cost to universities even when students retake courses during regular semesters. These losses and delays are a result of students failing courses because they underestimate the effort they need to put in. In case of elective courses, students sometimes select courses that are not aligned with their inherent talents and abilities. In this research we propose to develop an academic grade prediction system (Acad-GPS), which predicts a student's future grades based on his/her academic history. This will allow students to prepare themselves for the academic rigour of upcoming courses. We have formulated this problem as a recommender system problem. The successful development of Acad-GPS will provide better guidance for university students, lead to fewer students failing courses, which will not only result in immediate cost savings to universities and the national exchequer, but also reduce average graduation times for students, avoid unnecessary delays of new entrants into the job market.

Dedication

This dissertation is dedicated with reverence, love and affection to my parents, whose love and prayers always accompanied and guided me like a light whenever I was in darkness and enabled me to reach this stage and to Dr. Hassan Aqeel, Dr. Usman Ilyas, Dr. Khawar Khurshid and all GEC members without their support and guidance it would not have been possible to achieve the desired results.

Certificate of Originality

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at NUST SEECS or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at NUST SEECS or elsewhere, is explicitly acknowledged in the thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics which has been acknowledged.

Author Name: **Muhammad Farooq Mazhar**

Signature: ____________________

Acknowledgment

I am extremely grateful to Dr. Hassan Aqeel for his valuable supervision and support to perform such a valuable research in the field of data mining and machine learning. I truly admire his dedication to my thesis and for all help and suggestions and guidance that he has offered. A great deal of my thesis has been possible because of him.
Thanks to my loving family and my caring friends who have helped me a lot, encouraged me in every despair moment and enabled me to face the challenges of my masters thesis.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Counseling and advising college students is a routine task of faculty members. Students are assigned faculty members in their departments as advisers to mentor and track their academic progress and identify foreseeable problems. Students choosing elective courses must base their decisions on limited information, usually information gleaned about other students that already took those courses before. Often time's students select electives which are not in line with their inherent strengths and talents which results in poor performance and low grades in that course. Although students could solicit advice from their academic advisers when it comes to choosing their electives, which is rarely the case. Guidance on making the right choice could be solicited from an academic adviser, however, even the most experienced academic advisors can only go off their own personal experience of student performance. Even then, it is rarely the case that academic advisers go so far as to consult the student's transcript / grade history to help him with such a decision.

Ideally, students' decisions about picking their electives should be informed by the history of all students before them. In technical terms, what is needed is a system that is able to consider the academic performance and histories of all students that took the same course prior to that, and predict the most likely performance of a student given his personal academic track record. Armed with this information, students will be in a better position to navigate the rigorous landscape of their college degree programs.

1.1 Overview

We propose to develop an artificial intelligence (AI) application that we call the Academic Grade Prediction System (Acad-GPS), which operates on the history of all available student transcripts. The information of the predicted grade provided by Acad-GPS will allow them to better negotiate their way through college, similar to the GPS information we rely on to navigate physical landscapes (hence the acronym Acad-GPS). We propose to explore two previously unexplored approaches to this problem: 1) A matrix factorization based recommender system, and 2) User Based Cumulative Filtering recommender system.

The first approach proposes to formulate the problem of predicting academic grades as a recommendation problem. A recommender system or a recommendation system is a subclass of information filtering system that seeks to predict the "rating" or "preference" that a user would give to an item [12] [15]. Recommender systems have found use in a wide variety of areas including movies, music, news, books, research articles, search queries, social tags, and products in general. There are also recommender systems for experts [7], collaborators [6], jokes, restaurants, garments, financial services [10], life insurance, romantic partners (online dating), and Twitter pages [16].

The second approach proposes to use a technique, called cumulative filtering, which represents the clustering base solution provided information of academic performance fo students in past. Although cumulative filtering requires extensive and less sparse datasets, we will be able to use the same software tools and libraries to develop both, the matrix factorization and cumulative filtering based solutions.

There are several tangible benefits to the deployment of Acad-GPS. When students make fewer missteps in their academic programs, they can graduate from their programs with fewer delays. Graduation delays due to failed courses that need to be repeated are doubly expensive to the Saudi government; A) Every semester's delay in graduation is a delay in a young person's entry into the workforce and an equivalent loss of productivity to the country, i.e., over the course of his lifetime that student will spend 4 months less time being a productive and active member of society. B) Universities are often times compelled to open departments during the summer to offer courses critical to meeting the graduation requirements of final year students, so they may clear their F grades and graduate and join the workforce. To do that they have to retain faculty members, pay them an additional two month salary, and also bear the additional cost of facility maintenance, i.e. cost of support staff, utilities, etc.

A key design consideration in the development of Acad-GPS will be its usability. To this end, Acad-GPS will be designed to use predictive variables that are already available to NUST. In other words, Acad-GPS will not depend on the output of any custom surveys that the university must then ensure students have to take in order to function. By relying only on predictive variables that the university already collects in its learning management system, we can ensure that Acad-GPS can be designed as a supplemental component to National University of Science and Technology present learning management system, called CMS and LMS. As part of Acad-GPS we will develop a data import module that retrieves updated student transcript information at the conclusion of every semester for use by its predictive engine. This way, Acad-GPS will not interfere with the functioning of CMS/LMS. The CMS/LMS system is used university wide, which makes the extension of its use to other departments within the university a simple task.

1.2 Problem Statement

We are given a data set of previous academic record of students. By past performance, our task is to predict the performance of the student in the University.

1.3 Objectives

The most immediate beneficiary of the Academic-GPS will be the Electrical Engineering department of SEECS - NUST. By the end of the dissertation, we intend to complete an in-house pilot deployment of Academic-GPS. This will require access to the database of grades of all / a large number of students. We will use the pilot deployment phase to identify bugs, problems and shortcomings in the system that may emerge in a wider deployment.

Entry of new grades into Acad-GPS is needed because the system learns from new data, and adding new grades of each student at the end of each semester will improve the accuracy of the system's predictions with every passing semester. Clearly, it will be preferable to have the data imported into Acad-GPS without the need for manual entry.

After the system emerges from the pilot phase we will present the system to the Rector NUST and offer to make it available for use by other colleges and departments of the university.

Most importantly perhaps, it the deployment of Acad-GPS will allow students to graduate with fewer missteps and frustration, and better enable them to discover their own professional strengths and interest areas.

Another key output of the deployment of this system will be immediate cost savings from all departments of the university that choose to deploy Acad-GPS, by reducing or eliminating the demand for courses that have to be offered during the summer semester to allow students to graduate early.

1.4 Contributions

With our research work we provide following contributions:

1. We systematically reviewed the literature about grade/GPA prediction and comprehensively presented them.
2. We analyzed a real-world data collected from undergraduate students of Electrical Engineering Department at NUST.
3. We evaluated state of the art machine learning techniques (CF and NMF) in predicting the performance of NUST students.
4. We proposed a feedback generated from course average GPA and domain average GPA to calculate the student's knowledge for particular course domain and provide feedback if the student needs to put more effort in that course based on the predicted GPA.

1.5 Limitations

The findings of this study have been based on NUST LMS dataset and considers broader generalizations in many cases. Consequently, datasets from other universities or campuses with same course guidelines could create a more robust model. In this way, an efficient grade prediction criterion can be developed which could be applied to all the universities of Pakistan.

This study takes into account a limited number of predictors mainly the students' performance in the University which provides us ground to predict the GPA of course which are yet to be studied. However, socioeconomic, motivational and environmental factors also play a vital role in the prediction of student success coupled with the scores of matric, high school and admission results which can be considered in future study. To predict students' grades using CF (UBCF) and MF (SVD and NMF) techniques on the dataset, we can see that the RMSE for MF technique is lower compared to the RMSE of CF techniques. RMSE can be estimated with more precise results if more information of the students' GPAs is available.

Moreover, there is a need to improve the prediction results by dealing with the cold start problems. Also, models based on Restricted Boltzmann Methods, neural networks and tensor factorization can be investigated to take the temporal effect into account in the student performance prediction. Despite these limitations, our research findings have important practical implications for the universities and institutes in enhancing their students' retention rate.

1.6 Thesis Outline

We organize the rest of this thesis as follows. Chapter 2 contains a review of the literature regarding academic performance among students. Chapter 3 presents the methodology of the study. Chapter 4 provides results of the study. Finally, in Chapter 5, the thesis is concluded, and we provide the directions of further research.

Chapter 2

Literature Review

Grade prediction has been a problem of interest to education researchers for decades. Grade prediction has been attempted at various educational levels, ranging from pre-school to college, with a variety of input data, and prediction techniques.

2.1 Educational and psychological measurement

Lloyd [21] studied a group of 788 third grade boys and 774 third-grade girls to predict whether they would go on to successfully graduate from high school or drop out. They were able to predict correctly for 7 out of 10 cases.

2.2 Correlation of grade prediction and self evaluation

Goda et al. [14] evaluated the validity of self-evaluation comments by students in predicting their academic grades on a six-grade scale for a particular course. They used a machine learning solution using a support vector machine classifier for prediction.

Another formulation of the grade prediction problem is as a classification problem. Classifiers like support vector machines used by Goda et al. [14] essentially categorize students into one of a number of categories (in this case those categories are the predicted grades) based on their academic histories and profile information. Like the regression based approach, this approach does not handle missing values very well either. Moreover, training algorithms for classifiers do not distinguish between the magnitude of misclassification errors, i.e., erroneously predicting an “A” grade as an “F” is considered as bad as predicting it as a “B”. In other words, predicted labels are treated as nominal variables, whereas letter grades are ordinal variables.

2.3 Personality and intelligence as indicators

Chamorro-Premuzic and Furnham [5] used the NEO–PI-R measure of the Big Five personality traits, which is a Study Process Questionnaire that measures approaches to learning, two measures of cognitive ability, the Wonderlic IQ Test and the Baddeley Reasoning Test of fluid intelligence. A year later they completed comprehensive essay-based exams and received a mean score based on six examinations. They demonstrated that academic performance correlated with ability, achieving and learning approaches. However, this approach depends on the administration of a lengthy questionnaire to incoming students to assess these personality traits to predict their performance.

2.4 College freshman grades over time

Sawyer and Maxey [26] considered a very large data set of college freshman from 260 colleges and evaluated the validity of grade prediction equations from ACT scores over the years. They concluded that grade prediction equations for each college held up over time and remained remarkably stable, in spite of many changing factors from one academic year to the next.

2.5 Academic early warning system

Beck and Davidson [3] also developed one such “academic early warning system,” as is being proposed by us. However, theirs depended on surveys of the academic orientation of incoming freshman and was only used to predict first semester grades.

2.6 SAT score based prediction

Chissom and Lanier [8] worked on a similar problem of predicting first quarter freshman GPA using SAT scores and high school GPA and CGPA. Several mathematical methods and techniques have been used to address the college grade prediction problem.

One of the most straightforward prediction techniques is regression [24]. Baron and Norman [2], Bridgeman and McCamley-Jenkins [4] and Wainer et al [31]. are some examples from among dozens of studies that develop predictive models of first semester GPAs of college students based on their high school GPAs, CGPAs, scores on various standardized college admissions tests (mostly SAT scores), as well as other personal information including ethnicity, gender and mother tongue. All of them, however, are united in the common dependence on linear regression-based models. The survey report by Young and Kobrin from 2001 reviewed 49 different studies over a period of more than 25 years that all used linear regression based models, with variations in the make up of the surveyed populations, number of colleges and choice of predictive variables.

2.7 Regression based model

Although the grades being predicted are letter grades, an ordinal variable, they can be alternatively represented as as their equivalent grade points, usually mapped to a linear scale of either 0 to 4 or 0 to 5. While regression models are easy to implement and develop, they are difficult to use in situation where some of the input variables may be missing, as will be the case in the scenario we are considering. College students often times pick their own electives in the course of completing their program, i.e. not all students at the same stage will necessarily have taken the same courses in their previous semesters. In this sense, if we were to use a regression model we may have to develop separate models for each unique combination of course histories. This requires sufficient training data for each combination, which can become difficult to attain. It should be noted that much of the work on college grade prediction focuses on the first semester GPA, because the predictive variables used that are used for them (high school GPA, SAT scores) are commonly available for all students. Grade prediction of individual college courses further into college programs become less predictable because of the sheer variety and sparseness of available predictive variables due to each students individual selections of prior courses.

2.8 K-nearest neighbor clustering

Other classification algorithms include clustering algorithms like k-nearest neighbor, which suffers from the same disadvantages as classification based approaches (i.e., it treats predicted variables as nominal) and regression based approaches (i.e., it requires workarounds to deal with missing and sparse predictive variables). Moreover, k-nearest neighbor, like all clustering algorithms suffers from a high space and computational overhead, although there are some workarounds available.

2.9 Correlation Between GPA and Entry Test

Shulruf and Hattie [27] investigated the predictive correlations between the New Zealand National Certificate of Educational Achievement (NCEA) with the student's first-year grade point averages (GPA) in the university. Evaluating different models for university entry criteria, they found that if excellence and merit are given greater weight in NCEA results, then there would be potentially increased in the merit-based admissions system. This model improved the student's success rate during the first year study at university. Our study investigates the predictive performance in subsequent semesters in the university.

2.10 Personalized Multi-Linear Regression Models (PLMR)

Grade prediction accuracy using Matrix Factorization (MF) method degrades when dealing with small sample sizes. Elbadrawy et al. [9] investigated different recommender system techniques to accurately predict the students' next term course grades as well as within the class assessment performance of George Mason University (GMU), University of Minnesota (UMN) and Stanford University (SU) . Their study revealed that both Personalized Multi-Linear Regression models (PLMR) and advance Matrix Factorization (MF) techniques could predict next term grades with lower error rate than traditional methods. PLMR was also useful for predicting grades on assessments within a regular class or online course by incorporating features captured through students' interaction with LMS and MOOC server logs.

2.11 Regression and Classification Models

The final grade prediction based on the limited initial data of students and courses is a challenging task because, at the beginning of undergraduate studies, most of the students are motivated and perform well in the first semester but as the time passed there might be a decrease in motivation and performance of the students. Meier et al. [22] proposed an algorithm to predict the final grade of an individual student when the expected accuracy of the prediction is sufficient. The algorithm can be used in both regression and classification settings to predict students' performance in a course and classify them into two groups (the student who perform well and the student who perform poorly). Their study showed that in-class exams were better predictors of the overall performance of a student than the homework assignment. The study also demonstrated that timely prediction of the performance of each student would allow instructors to intervene accordingly. Zimmermann et al. [33] considered regression models in combination with variable selection and variable aggregation approach to predict the performance of graduate students and their aggregates. They have used a dataset of 171 students from Eidgenössische Technische Hochschule (ETH) Zürich, Switzerland. According to their findings, the undergraduate performance of the students could explain 54% of the variance in graduate-level performance. By analyzing the structure of the undergraduate program, they assessed a set of students' abilities. Their results can be used as a methodological basis for deriving principle guidelines for admissions committees.

Morsy and Karypis [23], proposed a cumulative knowledge-based regression model based on the historical students' course grade data as well as the information available about the courses. They obtained a large dataset of the student-course grades from the College of Science and Engineering at UMN and depicted the relationships between the courses in terms of the knowledge components being taught in the university. Their approach showed significant improvement in predicting the next term grade of the students. Kapur et al [18]. compared a number of marks prediction algorithms based on data mining and classification such as Naïve Bayes, Naïve Bayes Multiple Nominal, decision tree, K-star, and Random Forest to predict the potential of the students. They collected data for various factors including the previous performances of the students, their background, the curriculum designed in their universities, and the method of teaching in each institute and compared the performance of each algorithm to suggest an optimal method for predicting the performance of the students.

2.12 Learning Management System (LMS)

Learning Management Systems (LMSs), for example, Moodle provide students with online access to course content and to communicate and collaborate with instructors and peers. The data collected from LMS can be used to track students' involvement in the studies and to predict their future academic performance. Currently, Massive Open Online Courses (MOOCs) are the popular low-cost technological solution to deliver distance learning education to students across the world. Many approaches have been used to forecast the performance of the students. Elbadrawy and Karypis described how the student and course academic features determined the enrollment patterns by using academic features to determine student and course groups at various levels of granularity. Their study also showed that incorporating the features-based groups into the various methods lead to better grade predictions and course rankings.

2.13 Multilayer Perceptron Neural Network

Educational Data Mining utilizes data mining techniques to discover novel knowledge originating in educational settings [1]. EDM can be used for decision making in refining repetitive curricula and admission criteria of educational institutions. Saarela and Kärkkäinen [25] applied the EDM approach to analyze the effects of core Computer Science courses and provide novel information for refining repetitive curricula to enhance the success rate of the students. They utilized the historical log file of all the students of the Department of Mathematical Information Technology (DMIT) at the University of Jyväskylä in Finland. They analyzed patterns observed in the historical log file from the student database for enhanced profiling of the core courses and the indication of study skills that support timely and successful graduation. They trained multilayer perceptron neural network model with cross-validation to demonstrate the constructed nonlinear regression model. In their study, they found that the general learning capabilities can better predict the students' success than specific IT skills.

2.14 Factorization Machines (FM)

Next term grade prediction methods are developed to predict the grades that a student will obtain in the courses for the next term. Sweeney et al. [28] developed a system for predicting students' grades using simple baselines and MF-based methods for the dataset of George Mason University (GMU). Their study showed that Factorization Machines (FM) model achieved the lowest prediction error and can be used to predict both cold-start and non-cold-start predictions accurately. In subsequent studies, Sweeney et al. [29] explored a variety of methods that leverage content features. They used FM, Random Forests (RF), and the Personalized Multi-Linear Regression (PMLR) models to learn patterns from historical transcript data of students along with additional information about the courses and the instructors teaching them. Their study showed that hybrid FM-RF and the PMLR models achieved the lowest prediction error and could be used to predict grades for both new and returning students.

2.15 Dropout Early Warning System (DEWS)

Dropout early warning systems help higher education institutions to identify students at risk, and to identify interventions that may help to increase the student retention rate of the institutes. Knowles utilized the Wisconsin DEWS approach to predict the student dropout risk [19]. They introduced flexible series of DEWS software modules that can adapt to new data, new algorithms, and new outcome variables to predict the dropout risk as well as impute key predictors. In subsequent studies, Xu et al. [**?**] developed a novel machine learning method for predicting student performance in degree programs. Their proposed method addresses the diversity of students' backgrounds and selected courses to make accurate predictions. They further developed an ensemble based progressive prediction architecture that incorporates students' evolving performance into the prediction.

2.16 Hidden Markov Model and Bayesian Knowledge Tracing

Hidden Markov model has been used widely to model student learning. Van De Sande investigated solutions of hidden Markov model and concluded that the utilization of a maximum likelihood test should be the preferred method for finding parameter values for the hidden Markov Model [**?**]. Hawkins et al. in a separate study developed and analyzed a new fitting procedure for Bayesian Knowledge Tracing and concluded that empirical probabilities had the comparable predictive accuracy to that of expectation maximization [17].

2.17 Way forward

Interestingly though, although the problem lends itself to formulating it as a recommendation problem, we are not aware of any significant study that has chosen to approach this problem from that angle.

Although treating grade prediction as a recommendation problem is mathematically more complex, such a formulation is inherently able to deal with the problem of missing variables / sparse predictive variables, which is a challenge in all foregoing approaches.

In our study, the approach is to use machine learning techniques to predict course grades of students. We used the state of the art techniques that are described and implemented in this section to do a comparative analysis of different techniques that can predict students' GPA in registered courses.

Chapter 3

Methodology

Machine Learning with Educational Data Mining (EDM) has gained much more attention in the last few years. Many machine learning techniques, such as collaborative filtering, matrix factorization [30], and artificial neural networks [32] are being used to predict students' GPA or grades. In this section, we will describe these machine learning techniques and how they are being used to predict students' GPA in registered courses within the context of education.

A natural question is whether we can somehow use some aspects of a student's past academic record to drive the recommendations, so we have some set of features for the user and the courses under consideration. However, here we'd like to be able to learn the inherent dimensions of learning required by courses from the data. That will help us cope with this problem where we do not have these features explicitly available.

In addition, we would like to take into account interactions between students and courses. In this application, the data consists of a matrix of academic records, where we have a large number of students (rows) and the grades they earned in courses (columns) they took.

However, many student have only taken a subset of available courses. We will transform this data matrix into a big students x courses matrix of "ratings". Depending on how many elective courses and pre-graduation students are included, the matrix can be sparse. If a student u has already taken a certain course v the entry in the u-th row and v-th column will contain the grade point of the grade earned. If student has not taken course v yet that entry will be empty (meaning it is unknown) which is distinct from a zero entry. In the latter case, our Acad-GPS will be expected to predict a grade for it.

Our goal here is to fill in all the empty cells of this matrix, while taking into consideration all the course grades available for a student, and every other student that has taken this course before.

3.1 Data Pre-processing and Selection

A real-world student data is collected from Electrical Engineering Department at SEECS NUST across students of the graduated batches. The dataset contains data of undergraduate students enrolled in the Electrical Engineering program. The data of each student contains the students pre-university traits (secondary school percentage, high school percentage, entry test scores and interview), the course credits and the obtained grades of multiple different courses that the students take in different semesters. We consider only letter-grade courses but not fail courses. The information of courses and their domain is shown in Table 3.1, which was obtained from the curriculum for Electrical Engineering designed for Pakistani Universities.

Table 3.1: Course Domain Table

Course Domain	Courses
Humanities	Communication Skills I, Communication Skills II, Islamic Studies
Management Sciences	Industrial Chemistry, Entrepreneurship, D Lab
Natural Sciences	Linear Algebra, Calculus and Analytical Geometry, Complex Variables and Transforms, Probability & Statistics
Computing	Object Oriented Programming, Computing Fundamentals and Programming
Electrical Engineering Foundation	Linear Circuit Analysis, Electricity and Magnetism, Electronics Workbench, Electronic Devices and Circuits,Digital Logic Design, Electrical Network Analysis, Electronic Circuit and Design, Signals & Systems
Electrical Engineering Core	Solid State Electronics, Microcontrollers and Interfacing, Electrical Machines, Power Electronics

3.2 Collaborative Filtering

Collaborative filtering (CF) is one of the most traditional recommender system technique to date. In the educational context, the CF algorithms make predictions of GPA by identifying similar students in the dataset. In this method, predictions are made by selecting and aggregating the grades of other students. In particular, there is a list of m students $\mathbf{S} = \{\mathbf{s_1}, \mathbf{s_2}, ..., \mathbf{s_m}\}$ and a list of n courses $\mathbf{C} = \{\mathbf{c_1}, \mathbf{c_2}, ..., \mathbf{c_n}\}$. Each student si has a list of courses Csi, which represents student GPA in a course. The task of CF algorithm is to find a student whose GPAs are similar to some other student. CF technique can be further split into item-based CF and user-based CF.

3.2.1 Item-based Collaborative Filtering

In Item-based Collaborative Filtering (IBCF) technique, if a student wants to know what GPA he/she would achieve in an upcoming course, the algorithm considers his/her history of GPAs in courses and predict the GPA of the new course to be the same as the most similar course the student has taken. The main steps are:

1. For each two courses, measure how similar they are regarding having received similar gradings by same students.
2. For each course, identify the courses that are most similar using k-nearest neighbors.
3. For the course, the student is going to take, find the most similar courses within the courses the student has taken and use those courses to predict the GPA of the course in the query.

In the first step, algorithm tends to identify similar courses from the data inside the user-item matrix to calculate the similarity matrix by using the distance between each pair of the course.

3.2.2 User-based Collaborative Filtering

In User-based Collaborative Filtering (UBCF), the algorithm considers similar students that have similar GPA in same courses. The main steps are:

1. The algorithm measures how similar each student in the database to the activestudent by calculating the similarity matrix.
2. Identify the most similar students by using k nearest neighbors.

3. Predict the GPA of the course of the active user by aggregating the GPA of that course taken by the most similar students. The aggregation can be a simple mean or weighted average by taking similarity between students into account.

The k nearest neighbour technique is used to select the neighbourhood for the active user $\mathbf{N}(\mathbf{a}) \subset \mathbf{U}$. The average rating of the neighbourhood users is calculated using the equation Equation 3.1, which becomes the predicted rating for the active use. The grade prediction becomes extremely challenging for the student with a few courses attended which is a well-known drawback of CF technique over the sparse dataset.

$$\hat{r}_{aj} = \frac{1}{\mid N(a) \mid} \sum_{i \in N} (a)\, r_{ij} \tag{3.1}$$

3.3 Matrix Factorization

Matrix factorization is a decomposition of a matrix into two or more matrices. Matrix factorization techniques are used to discover hidden latent factors and to predict missing values of the matrix. In our study, we formulated the problem of predicting student performance as a recommender system problem and used matrix factorization methods (SVD and NMF) which are the most effective approaches in recommender systems.

3.3.1 Singular Value Decomposition

Singular Value Decomposition (SVD) is a matrix factorization technique that decomposes students-courses matrix R into

$$R = U \Sigma V^T \tag{3.2}$$

where;

- $\mathbf{U}$ is an $\mathbf{m} \times \mathbf{r}$ orthogonal matrix, where $\mathbf{m}$ represents number of users and $\mathbf{r}$ represents the rank of the matrix $\mathbf{R}$,
- $\mathbf{\Sigma}$ is an $\mathbf{r} \times \mathbf{r}$ diagonal matrix with singular values along the main diagonal entries and zero everywhere else,
- $\mathbf{V}$ is an $\mathbf{r} \times \mathbf{n}$ orthogonal matrix where n represents the number of courses.

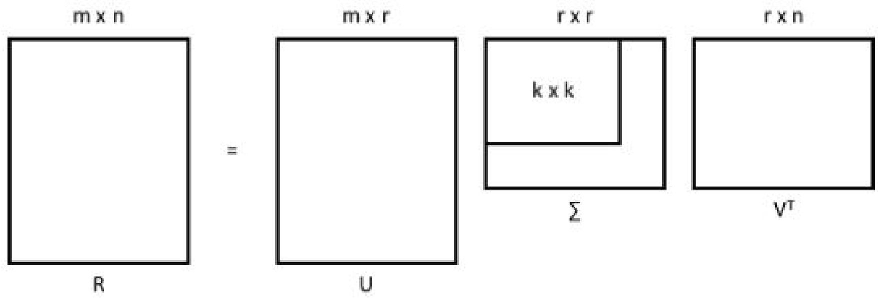

Figure 3.1: Decomposition of Matrix $\mathbf{R}$ by SVD

The graphical representation of SVD is shown in Figure 3.1. In newly constructed matrices, $\mathbf{r}$ represents the rank of the matrix $\mathbf{R}$. The values in the matrix $\mathbf{\Sigma}$ are known as singular values $\sigma_{\mathbf{i}}$, and they are stored in decreasing order of their magnitude. Each singular value $\sigma_{\mathbf{i}}$ of the matrix $\mathbf{\Sigma}$ represents hidden latent features, and their weights have variance on the values of matrix $\mathbf{R}$. The sum of all elements represents the total variance of matrix $\mathbf{R}$.

SVD is widely being used to find the best $\mathbf{k}$-rank approximation for the matrix $\mathbf{R}$. The rank $\mathbf{r}$ can be reduced to $\mathbf{k}$, where $\mathbf{k} < \mathbf{r}$, by taking only the largest singular value $\mathbf{k}$ which is the first diagonal value of the matrix $\mathbf{\Sigma}$ and then reduce both $\mathbf{U}$ and $\mathbf{V}$ accordingly. The obtained result is a $\mathbf{k}$-rank approximation $\mathbf{R_k} = \mathbf{U_k \Sigma_k V_k^T}$ of the matrix $\mathbf{R}$, in such a way that the Frobenius norm of $\mathbf{R} - \mathbf{R_k}$ is minimized. The Frobenius norm $(\|\mathbf{R} - \mathbf{R_k}\|\,\mathbf{F})$ is defined as simply the sum of squares of elements in$\mathbf{R} - \mathbf{R_k}$[52]. To predict the GPA in a course, SVD assumes that each student grade is composed of the sum of preferences of the various latent factors of the courses. To predict the grade of a student $\mathbf{i}$ for course $\mathbf{j}$ is as simple as taking the dot product of vector $\mathbf{i}$ in the student feature matrix and the vector $\mathbf{j}$ in the course feature matrix.

The problem with SVD is that it is not useful on big and sparse datasets. Simon Funk proposed to use a Stochastic Gradient Descent (SGD) algorithm to compute the best rank-$\mathbf{k}$ matrix approximation using only the known ratings of original matrix. Stochastic Gradient Descent (SGD) is a convex optimization technique that gets the most accurate values of those two featured matrices that are obtained during the decomposition of the original matrix in the method of SVD. SGD has following steps:

1. Re-construct the target students-courses matrix by multiplying the two lower ranked matrices.
2. Get the difference between the target matrix and the generated matrix.
3. Adjust the values of the two lower-ranked matrices by distributing the difference to each matrix according to their contribution to the product target matrix.

Above is a repeated process till the difference is lower than a preset threshold. By reducing the dimensionality of the students-courses matrix, the execution speed is reduced, and the accuracy of the prediction is increased because of considering only the courses that contribute to the reduced data. Dimensionality reduction leads to the reduction of noise and over-fitting. This method is also used in recommender systems for the Netflix challenge [13].

3.3.2 Non-Negative Matrix Factorization

Non-negative matrix factorization (NMF) is a matrix factorization technique that decomposes a matrix **V** into two non-negative factor matrices **W** and **H** such that

$$V \approx WH \tag{3.3}$$

where;

- **W** is a $\mathbf{u} \times \mathbf{k}$ orthogonal matrix,
- **H** is a $\mathbf{k} \times \mathbf{v}$ orthogonal matrix.

Graphical representation of NMF is shown in Figure 3.2. NMF is a powerful technique that uncovers the latent hidden features in a dataset and provides a non-negative representation of data [20]. The problem with NMF is to find **W** and **H** when the dataset is large and sparse. A sequential coordinate-wise descent (SCD) algorithm can be used with NMF to impute the missing values [11]. NMF imputation using SCD takes all entries into account when imputing a single missing entry.

3.4 Methods

We used CF (UBCF) and MF (SVD and NMF) techniques to predict GPA of the student for the courses. A feedback model can be further developed based on the predicted GPA of the student in a course.

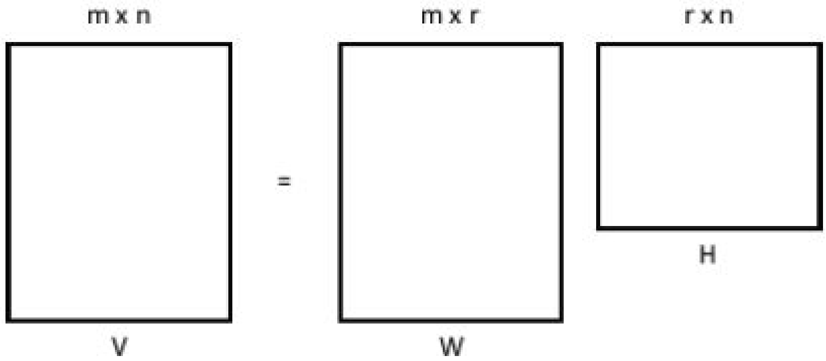

Figure 3.2: Decomposition of Matrix **V** by NMF

3.5 Problem Formulation

For this study, we would like to predict student GPA from the scale 0.0 - 4.0. The given data we have is *(Student,Course,GPA)* triplet and we need to predict GPA for each student for the courses he/she will enroll in the future. In general, we have $mathbfn$ students and $mathbfm$ courses, comprising an $mathbfn \times m$ sparse GPA matrix G, where $\{\mathbf{G_{ij}} \in \mathbf{R} | \mathbf{G_{ij}} \leq \mathbf{4}\}$ is the grade student i earned in course j.

For training machine learning models, students grades need to be converted to GPA. These grades are converted to numerical GPA values using the NUST grading policy on a 4 point GPA scale with respect to the letter grades A=4, B+=3.5, B=3.0, C+=2.5, C=2.0, D+=1.5, D=1.0 and F=0.0.

A prediction algorithm works best with centering predictor variables, so all the data were transformed by centering (average GPA of a course is subtracted from all GPAs of that course).

3.6 Prediction of Student Grades

As our objective is to predict students GPA in the courses for which he/she needs to enroll in the future, we used CF (UBCF) and MF (SVD and NMF) techniques to predict courses GPA of students. We take the data into a matrix in the form of *(Student,Course,GPA)* triplet. For illustration, here we have taken a few students and courses to display their grades. In the

Table 3.3 we can see that a student with Id. SB145 have a GPA 3.5 in the course Electronic Circuit and Design and have a GPA of 4.0 in the D-Lab course. While this student needs to enroll into Linear Circuit Analysis, Islamic Studies, and Signals and System. A student with Id. SB185 have similar GPA in Electronic Circuit and Design course like the student with Id. SB145 and this student need to enroll into Linear Circuit Analysis, Islamic Studies, Signals and Systems, and D-Lab courses.

Table 3.2: Students course and GPA in particular courses

Student ID	LCA	ECD	IS	SS	DL
NUST201304501BSEECS60413F		3.5			4.0
NUST201304531BSEECS60413F		4.0			3.5
NUST201304614BSEECS60413F		3.5			
NUST201304724BSEECS60413F					
NUST201304780BSEECS60413F	2.0		2.5		

Linear Circuit Analysis (LCA) Electronic Circuit Design (ECD) Islamic Studies (IS) Signals& Systems(SS) D-lab (DL)

Collaborative Filtering: We have used UBCF to predict the students' grades in courses. UBCF provides us with grade prediction of a student s in a course c by identifying student grades in same courses as s. For prediction of grades, the neighborhood students ns similar to student s are selected that have taken at least nc courses that were taken by student s. To apply UBCF model we first converted the students-courses matrix R into a real-valued rating matrix having student GPA from 0 to 4. To measure the accuracy of this model we have split the data into 70% trainset and 30% testset. In UBCF model The similarity between students and courses is calculated using k nearest neighbors.

Matrix Factorization: Matrix factorization is the decomposition of a matrix V into the product of two matrices W and H, i.e. $\mathbf{V} \approx \mathbf{WH_T}$. In this study, we have used SVD and NMF matrix factorization techniques to predict the student GPA. The main issue of MF techniques is to find out the optimized value of matrix cells for W and H.

In SVD approach, the students' dataset is converted into real-valued rating matrix having student grades from 0 to 4. The dataset is split into 70% for training the model and 30% for testing the model accuracy. We used Funk SVD to predict GPA in the courses for which the students have not yet taken the courses. The largest ten singular values are 191.8012, 18.8545, 14.7946, 13.8048, 12.4328, 11.8258, 11.1058, 10.2583, 9.5020 and 9.1835. It can be observed from that the distribution of the singular values of students-courses matrix diminishes quite fast suggesting that a low-rank matrix can approximate the matrix with high accuracy. This encourages the adoption of low-rank matrix completion methods for solving our grade/GPA prediction problem.

By applying Funk's proposed heuristic search technique called Stochastic Gradient Descent (SGD) gradient to the matrix G we obtained two matrices student and courses dimensional spaces (with the number of hidden features set to two, to ease the task of visualizing the data). The stochastic gradient descent technique estimates the best approximation matrix of the problem using greedy improvement approach.

Table 3.3 represents the students' features dimensional space, and Table 3.4 represents courses' features dimensional space. With the dot product of these features dimensional space we can predict GPA in the courses for which the students are shown in Table 3.2 needs to enroll. Please note that we usually do not know the exact meaning of the values of these two-dimensional space, we are just interested in finding the correlation between the vectors in that dimensional space. For understanding, take an example of a movie recommender system. After matrix factorization, each user and each movie are represented by two-dimensional space. The values of the dimensional space represent the genre, amount of action involved, quality of performers or any other concept. Even if we do not know what these values represent, but we can find the correlation between users and movies using the values of dimensional space.

Table 3.3: Students feature dimensional space

Student ID	V1	V2
NUST201200538BSEECS60412F	0.39	0.18
NUST201304614BSEECS60413F	0.45	0.20
NUST201304486BSEECS60413F	0.42	0.20
NUST201305641BSEECS60413F	-0.31	0.02
NUST201201281BSEECS60412F	0.09	0.12

Table 3.4: Courses feature dimensional space

Course	V1	V2
Linear Circuit Analysis	1.19	-0.04
Electronic Circuit and Design	0.94	0.10
Islamic Studies	1.77	-0.03
Signals and Systems	0.34	0.20
D-Lab	0.46	0.18

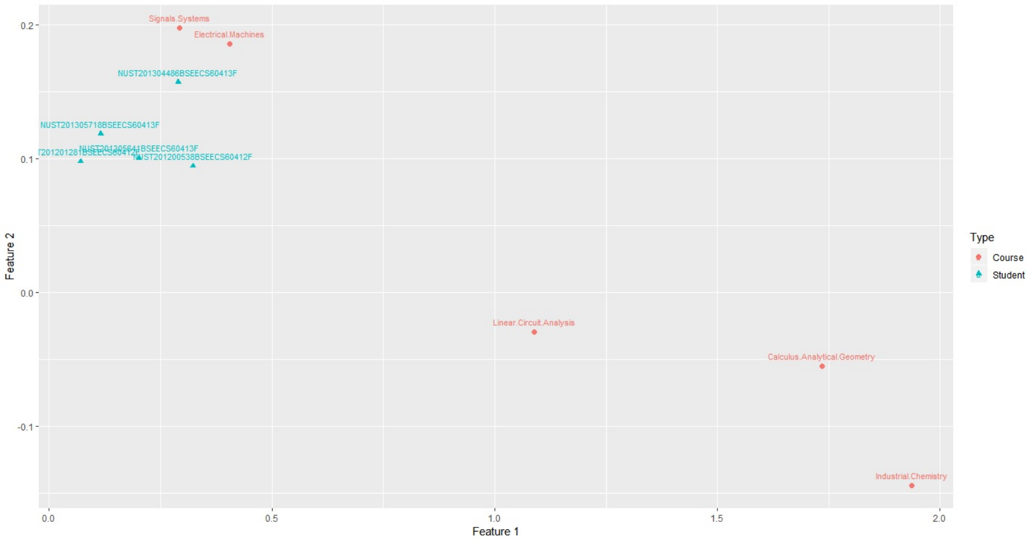

Figure 3.3: Decomposition of Students and Courses feature set in SVD

In NMF approach, we have a $\mathbf{u} \times \mathbf{v}$ matrix V with non-negative entries of student grades from 0 - 4 that decomposes into two non-negative, rank-k matrices W $(\mathbf{u} \times \mathbf{k})$ and H $(\mathbf{k} \times \mathbf{v})$ such that $mathbfV \approx WH$. Before decomposing a matrix into two matrices first, we need to choose a rank-k for NMF that gives the smallest error for grade predictions of the students-courses matrix. In our experiments with NMF, the rank-k 1 gives the minimum Mean Squared Error (MSE). So, we have used one as rank-k (due to missing or sparse data) value and decomposed the matrix into W and H. The Figure 3.4 shows the plot of error values and rank of the matrix

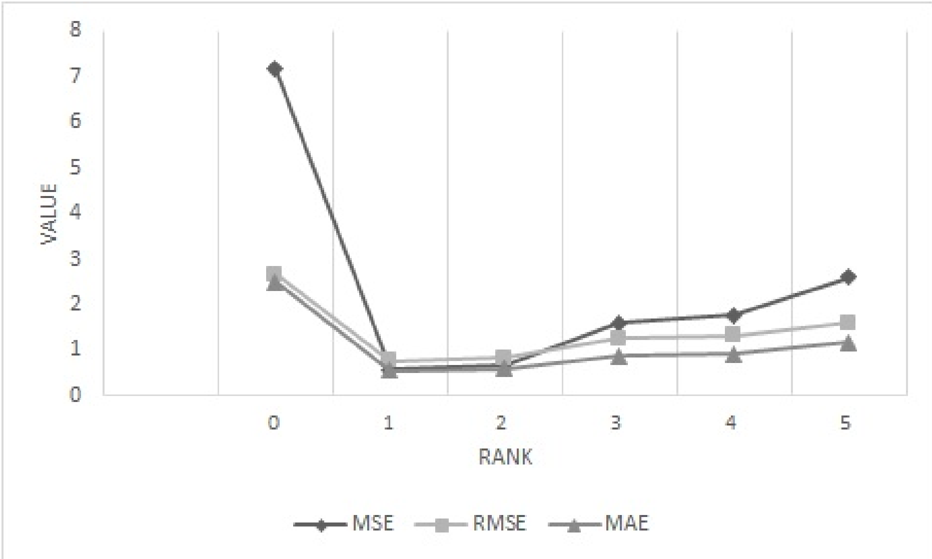

Figure 3.4: Rank selection of matrix - **k** based on error values in NMF

3.7 Feedback

Machine learning techniques can be utilized to identify the weak students who need appropriate counseling/advising in the courses, by early predicting the courses grades. Students can receive their feedbacks based on the student's knowledge in the particular course domain based on the average GPA of the student in a particular domain achieved against the predicted grade and course average providing feedback to the instructor about the courses in which a student is weak.

Chapter 4

Experimental Results

We have performed different kinds of experiments with our data using different machine learning techniques to develop an interactive graphical representation to explore students' performance in universities. We have done some exploratory analysis and experiments to achieve our goal and make a reasonable contribution in research using machine learning techniques.

4.0.1 Grade Prediction

For students, GPA prediction, students-courses matrix G is constructed. The data were transformed by centering the predictor variables by taking average GPA of a course and subtracted it from all GPAs of that course. 70% of the dataset is used for training the CF and MF models. Student GPAs for the courses has been predicted.

4.0.2 Evaluation on Model Performance

There are several types of measures for evaluating the success of models. However, the evaluation of each model depends heavily on the domain and system's goals. For our system, our goal is to predict students' GPA and make decisions if a student needs to work hard to complete the course. These decisions work well when our predictions are accurate. To achieve it, we have to compare the prediction GPA against the actual GPA for the students-courses pair. Some of the most used metrics for evaluation of the models are the Root Mean Squared Error (RMSE), Mean Squared Error (MSE) and Mean Absolute Error (MAE). We evaluated model predictions by repeated random subsample cross-validation. We performed multiple repetitions. In each run, we choose randomly 70% of students data into the train set and 30% of students data into the test set. We have computed RMSE, MSE, and MAE for each model. From 4.1 the results show that the NMF model provides a clear improvement over the CF and SVD models. Please note we are not performing student-level cross-validation of predicted results on newly registered students in this study but the currently enrolled students.

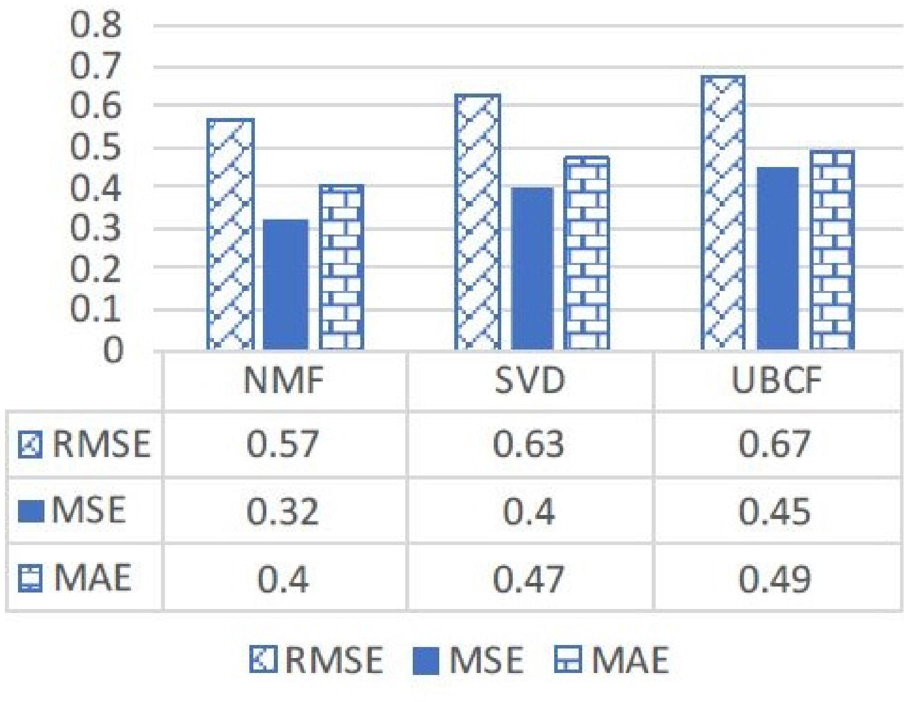

Figure 4.1: Evaluation of grade prediction models

4.1 Insights

In this study, we have used CF (UBCF) and MF (SVD and NMF) techniques to predict the students' performance in the courses. CF is a popular method to predict the students' performance due to its simplicity. In this technique, the students' performance is analyzed by using the previous data. It provides feedback to enhance the students' learning process based on the outcome of the analysis. However, this method has several disadvantages: since it depends upon the historical data of users or items for predicting the results. It shows poor performance when there is too much sparsity in the data, due to which we are not able to predict the students' performance accurately.

Comparatively, in SVD technique, the data matrix R is decomposed into users-features space and items-features space. When SVD technique is used with gradient descent algorithm to compute the best rank-k matrix approximation using only the known ratings of R, the accuracy of predicting the students' performance enhances but it may contain negative values which are hard to interpret. NMF technique enhances the meaningful interpretations of the possible hidden features that are obtained during matrix factorization.

From the above discussion, it is clear that the NMF technique outperforms CF and MF techniques with lesser chances of error. The overall result obtained in this study also shows that NMF surpasses other techniques in predicting the student's performance.

Chapter 5

Conclusions and Future Work

Early GPA predictions are a valuable source for determining student performance in the university. In this study, we discussed CF, SVD and NMF prediction methods for predicting student GPA.We proposed NMF recommender system techniques for predicting student performance in the courses. Results show that the proposed technique is predicting satisfactory results. In a recommender system approach, we measure student knowledge in course domain, which provides appropriate counseling to them about different courses in a particular course domain by estimating the performance of other students in that course. This recommender system leads to student motivation and provides them early recommendation if they need to improve their knowledge in the courses. It also helps a teacher to determine weak students in the class and to give them the advice to improve their performance. In this way rate of the student, retention can be increased.

Students need to make careful and informed career decisions, while at the same time training them and facilitating their transition between different educational pathways. This can only be achieved by meeting the variety of educational indicators, including average time to graduation, that are used for those rankings. It requires that we track student progress and publish a sophisticated range of education outcomes, showing year-on-year improvements.

The benefits of Acad-GPS proposed here are numerous. It will allow university students to identify courses that are likely to derail their progress in their respective programs. It leads to immediate cost-savings to the universities deploying it by reducing the need to offer remedial courses during the summer, which requires additional funds to pay faculty members to retain them for the period of the summer. It also cuts down on delays and the average graduation times due to lingering failed courses. It reduces student frustration by giving them a heads-up at the beginning of each semester in all courses in which they will have to put in more effort. In addition, continuing difficulties in early semesters of a program can be indicative of a student having chosen a program that is not in line with his or her innate strengths and abilities.

5.0.1 Future Work Directions

Here in this research work, we used various techniques for student grade prediction including CF, SVD and NMF out of which, we concluded that the NMF technique outperformed CF and MF techniques in terms of predicting the students' GPA more accurately. This technique can be further explored to enhance the features of grade prediction algorithms by using Convoluted Neural Networks, Artificial Neural Networks, Restricted Boltzmann Methods and Genetic Programing techniques. These tools can be used to create generative models for feature extraction such as identifying the attributes (Matric, High School, Entry Test, and Interview Score) that play important role in grade prediction.

Appendix A

UBCF code in R

```
rm(list = ls())
# Package Recommenderlab: Provides a research infrastructure to test
and develop recommender algorithms
# including UBCF, IBCF, FunkSVD and association rule-based algorithms.
require(recommenderlab)

setwd("/Users/Farooq/Downloads/Archive")

# Read the prepared data set of course grades
Data <- read.csv("Student Courses Data Set.csv")

raw_data <- Data

# Save the student number
id <- raw_data$Registration.No

# Convert to a matrix, dropping the 1st column (IDs) along the way
raw_data <- as.matrix(raw_data[,-1])

# Set the row name of the matrix to be the id
dimnames(raw_data)[[1]] <- id

# Replace multiple dots in course name by one dot
# Dots are occurring due to space and use of symbol "&"
dimnames(raw_data)[[2]] <- gsub(pattern = "\\.+", replacement = "\\.",
dimnames(raw_data)[[2]])

# Center the data
```

```
# Centering is done by subtracting the column means (omitting NAs)
of data from their corresponding columns
centered <- scale(raw_data, scale = FALSE, center = TRUE)

# Get the means so we can convert the predictions back to meaning
full GPAs
means <- colMeans(raw_data, na.rm = TRUE)

# Transform data to 'realRatingMatrix' data type
# RealRatingMatrix A matrix containing ratings (typically 1-5 stars,
etc.)
data <- as(centered, "realRatingMatrix")

# Train recommenders object using UBCF method
r_ubcf <- Recommender(data, method = "UBCF", param=list(normalize
= NULL)) # User-based collaborative filtering

# Make predictions using the recommender objects
gpa_ubcf <- predict(r_ubcf, data, type = "ratings")
gpa_ubcf <- as(gpa_ubcf, "matrix") + means

# For evaluation of the methods on the data set
value_count <- apply(raw_data, 1, FUN = function(x) length(x[!is.na(x)]))

having_more_than_5 <- which(value_count > 5)

inTrain <- sample(having_more_than_5, 80, replace = FALSE)
training <- data[inTrain, ]
testing <- data[-inTrain, ]

# Evaluation model performance
# given: how many items were given to create the predictions.
# goodRating: threshold for determining what rating is a good rating.

e <- evaluationScheme(training, method = "split", train = 0.7, goodRating
= 4, given = 5)
r1 <- Recommender(getData(e, "train"), "UBCF")

p1 <- predict(r1, getData(e, "known"), type="ratings")

error <- round(rbind(UBCF = calcPredictionAccuracy(p1, getData(e,
```

```
"unknown"))),2)
error

# Save all predictions to files
write.csv(gpa_ubcf, file = "UBCF-Predictions.csv")

# Function to convert gpa to letter grade
gpa2grade <- function(gpa){
if (gpa < 1) return("F")
r <- gpa %% (1/3)
f <- round((gpa %/% (1/3))*(1/3),2)
c <- f + (1/3)
g <- 0
if (f + r >= c - (1/6)) g <- c else g <- f
g <- round(g,1)
if (g <= 1.0) return("D")
if (g == 1.5) return("D+")
if (g == 2.0) return("C")
if (g == 2.5) return("C+")
if (g == 3.0) return("B")
if (g == 3.5) return("B+")
if (g >= 4.0) return("A")

}

# Wrapper function to predict grades for a student
# Predict any courses the student hasn't had a grade

predict_gpa <- function(student_record, save = FALSE){
row <- which(dimnames(raw_data)[[1]] == student_record)
if (length(row) == 0) row <- student_record # student_record can
be the student id, or the row number
if (!is.numeric(row)) stop("Student not found")

# Create a data frame containings the grade and GPA.

GPA_UBCF <- data.frame(Course = names(round(gpa_ubcf[row,][!is.na(gpa_ubcf[row,]
Grade.Predicted = sapply(round(gpa_ubcf[row,][!is.na(gpa_ubcf[row,])],2),
gpa2grade),
GPA.Predicted = round(gpa_ubcf[row,][!is.na(gpa_ubcf[row,])],2))
if (save) {
```

```
fname <- paste0("Prediction for ", dimnames(raw_data)[[1]][row],
".csv")
write.csv(GPA_UBCF, fname, row.names = FALSE)
}
return(GPA_UBCF)
}
# Using the prediction function

predict_gpa("SB300") # This will output the prediction results using
UBCF
predict_gpa("SB300", save = TRUE) # save = TRUE will save the prediction
to a csv file
```

Appendix B

SVD code in R

```
content...rm(list = ls())
# Package Recommenderlab: Provides a research infrastructure to test
and develop recommender algorithms
# including UBCF, IBCF, FunkSVD and association rule-based algorithms.
require(recommenderlab)
# Package ggplot2: A system for 'declaratively' creating graphics,
based on "The Grammar of Graphics".
require(ggplot2)

setwd("/Users/Rohan/Downloads/Archive")

# Read the prepared data set of course grades
Data <- read.csv("Student Courses Data Set.csv")

raw_data <- Data

# Save the student number
id <- raw_data$Registration.No

# Convert to a matrix, dropping the 1st column (IDs) along the way
raw_data <- as.matrix(raw_data[,-1])

# Set the row name of the matrix to be the id
dimnames(raw_data)[[1]] <- id

# Replace multiple dots in course name by one dot
# Dots are occurring due to space and use of symbol "&"
dimnames(raw_data)[[2]] <- gsub(pattern = "\\.+", replacement = "\\.",
```

```
dimnames(raw_data)[[2]])

# Center the data
# Centering is done by subtracting the column means (omitting NAs)
of data from their corresponding columns
centered <- scale(raw_data, scale = FALSE, center = TRUE)

# Get the means so we can convert the predictions back to meaning
full GPAs
means <- colMeans(raw_data, na.rm = TRUE)

# Transform data to 'realRatingMatrix' data type
# RealRatingMatrix A matrix containing ratings (typically 1-5 stars,
etc.)
data <- as(centered, "realRatingMatrix")

# Train recommenders object using SVDF methods
r_svdf <- Recommender(data, method = "SVDF", param=list(normalize
= NULL)) # Funk SVD

# For illustration with chart perform SDVF with rank = 2
r1 <- Recommender(data, method = "SVDF", param=list(normalize = NULL,
k = 2))

# Extract the 2 feature space matrices
students <- as.data.frame(getModel(r1)$svd$U)
courses <- as.data.frame(getModel(r1)$svd$V)
students
courses

# Assign student ID and course name to each data point
students$Type <- "Student"
students$Name <- rownames(raw_data)
courses$Type <- "Course"
courses$Name <- dimnames(raw_data)[[2]]

# Build a mixed matrix with students and courses
students_and_courses <- data.frame(rbind(students, courses))
names(students_and_courses) <- c("Feature 1", "Feature 2", "Type",
"Name")
```

```
# Subset to plot - Get 5 random students and 5 random courses
set.seed(4) # Change this seed number to get another subset

chart_data <- students_and_courses[c(sample(1:nrow(students),5,replace
= FALSE),
sample((nrow(students)+1):(nrow(students)+nrow(courses)),5,replace
= FALSE))
,]
chart_data

# Plot the matrix
g <- ggplot(chart_data, aes(x = ‘Feature 1‘,
y = ‘Feature 2‘,
shape = Type,
color = Type,
label = Name)) +
geom_point(stat = "identity", size = 2) +
geom_text(aes(label=Name),hjust=0.5, vjust=-1., size = 3)

g

# Make predictions using the recommender objects
gpa_svdf <- predict(r_svdf, data, type = "ratings")
gpa_svdf
gpa_svdf <- as(gpa_svdf, "matrix") + means

# For evaluation of the methods on the data set
value_count <- apply(raw_data, 1, FUN = function(x) length(x[!is.na(x)]))
having_more_than_5 <- which(value_count > 5)

inTrain <- sample(having_more_than_5, 80, replace = FALSE)
training <- data[inTrain, ]
testing <- data[-inTrain, ]

# Evaluation model performance
# given: how many items were given to create the predictions.
# goodRating: threshold for determining what rating is a good rating.
e <- evaluationScheme(training, method = "split", train = 0.7, goodRating
= 4, given = 5)
r1 <- Recommender(getData(e, "train"), "UBCF")
```

```
p1 <- predict(r1, getData(e, "known"), type="ratings")

error <- round(rbind(SVDF = calcPredictionAccuracy(p1, getData(e,
"unknown"))),2)
error

# Save all predictions to files
write.csv(gpa_svdf, file = "SVDF-Predictions2.csv")

# Function to convert gpa to letter grade
gpa2grade <- function(gpa){
if (gpa < 1) return("F")
r <- gpa %% (1/2)
f <- round((gpa %/% (1/2))*(1/2),2)
c <- f + (1/2)
g <- 0
if (f + r >= c - (1/4)) g <- c else g <- f
g <- round(g,1)
if (g <= 1.0) return("D")
if (g == 1.5) return("D+")
if (g == 2.0) return("C")
if (g == 2.5) return("C+")
if (g == 3.0) return("B")
if (g == 3.5) return("B+")
if (g >= 4.0) return("A")

}

# Wrapper function to predict grades for a student
# Predict any courses the student hasn't had a grade

predict_gpa <- function(student_record, save = FALSE){
row <- which(dimnames(raw_data)[[1]] == student_record)
if (length(row) == 0) row <- student_record # student_record can
be the student id, or the row number
if (!is.numeric(row)) stop("Student not found")

# For each algorithm, create a data frame containings the grade and
GPA.

GPA_SVDF <- data.frame(Course = names(round(gpa_svdf[row,][!is.na(gpa_svdf[row,]
```

```
Grade.Predicted = sapply(round(gpa_svdf[row,][!is.na(gpa_svdf[row,])],2),
gpa2grade),
GPA.Predicted = round(gpa_svdf[row,][!is.na(gpa_svdf[row,])],2))

if (save) {
fname <- paste0("Prediction for ", dimnames(raw_data)[[1]][row],
".csv")
write.csv(GPA_SVDF, fname, row.names = FALSE)
}
return(GPA_SVDF)
}

# Using the prediction function

predict_gpa("NUST201201346BSEECS60412F") # This will output the prediction
results using UBCF

predict_gpa("NUST201201346BSEECS60412F", save = TRUE) # save = TRUE
will save the prediction to a csv file
```

Appendix C

NMF code in R

```
rm(list = ls())
# Package NNLM: This is a package for Non-Negative Linear Models
(NNLM). It implements
# fast sequential coordinate descent algorithms for non-negative
linear regression
# and non-negative matrix factorization (NMF).
require(NNLM)
# Package Knitr: A General-Purpose Package for Dynamic Report Generation
in R
require(knitr);

setwd("/Users/Rohan/Downloads/Archive")

# Read the prepared data set of course grades
Data <- read.csv("Student Courses Data Set.csv")

raw_data <- Data

# Save the student number
id <- raw_data$Registration.No

# Convert to a matrix, dropping the 1st column (IDs) along the way
raw_data <- as.matrix(raw_data[,-1])

# Set the row name of the matrix to be the id
dimnames(raw_data)[[1]] <- id

# Replace multiple dots in course name by one dot
```

```
# Dots are occurring due to space and use of symbol "&"
dimnames(raw_data)[[2]] <- gsub(pattern = "\\.+", replacement = "\\.",
dimnames(raw_data)[[2]])

# Transformed data ready for NMF
data <- raw_data
true_data = data.matrix(data, rownames.force = NA)
? nnmf
# Choose the rank of NMF by minimising MSE.
MSE_medians = list()
RMSE_medians = list()
MAE_medians = list()
for (k in 0:5){
MSE = list()
RMSE = list()
MAE = list()
for (i in 1:100){
data_ = data.matrix(data, rownames.force = NA)
indexes = which(!is.na(data_))
zeroed_indexes = sample(length(indexes), length(indexes) * 0.01)
index = indexes[zeroed_indexes]

data_[indexes[zeroed_indexes]] <- NA

# NMF imputation
data_.nmf <- nnmf(data_, k, check.k = FALSE, method = "scd" );
data_.hat.nmf <- with(data_.nmf, W %*% H);
data_.hat.nmf <- replace(data_.hat.nmf, is.na(data_.hat.nmf), 0)
MSE[[i]] <- sum(((data_.hat.nmf[index]-true_data[index])^2))/length(index)
RMSE[[i]] <- sqrt(sum(((data_.hat.nmf[index]-true_data[index])^2))/length(index)
MAE[[i]] <- sum((abs(data_.hat.nmf[index]-true_data[index])))/length(index)
}
MSE_medians[[k+1]] <- median(unlist(MSE))
RMSE_medians[[k+1]] <- median(unlist(RMSE))
MAE_medians[[k+1]] <- median(unlist(MAE))
}
C <- c(0,1,2,3,4,5)
df <- data.frame(C,c(unlist(MSE_medians)),c(unlist(RMSE_medians)),
c(unlist(MAE_medians)))
colnames(df) <- c('rank','MSE','RMSE','MAE')
df
```

```
# Evaluation model performance
g_range <- range(0, MSE_medians, RMSE_medians, MAE_medians)

plot(unlist(MSE_medians), type="o", col="blue", ann=FALSE, axes=FALSE)
axis(1, at=1:6, lab=c(0,1,2,3,4,5))
axis(2, las=1, at=4*0:g_range[2])
lines(unlist(RMSE_medians), type="o", col="green")
lines(unlist(MAE_medians), type="o", col="red")
box()
title(xlab="Rank")
title(ylab="Value")

legend(5, g_range[2], c("MSE","RMSE", "MAE"),
col=c("blue","green","red"), pch=20:20);

# NMF imputation with chosen rank 1
data = data.matrix(data, rownames.force = NA)
data.nmf <- nnmf(data, 1, check.k = FALSE);
data.hat.nmf <- with(data.nmf, W %*% H);

data.hat.nmf[data.hat.nmf > 4] = 4
cM <- colMeans(data.hat.nmf, na.rm = TRUE)
indx <- which(is.na(data.hat.nmf), arr.ind=TRUE)
data.hat.nmf[indx] <- cM[indx[,2]]


# Save all predictions to files
gpa_nmf = data.frame(data.hat.nmf)
write.csv(gpa_nmf, file = "NMF-Predictions.csv")

# Function to convert gpa to letter grade
gpa2grade <- function(gpa){
if (gpa < 1) return("F")
r <- gpa %% (1/2)
f <- round((gpa %/% (1/2))*(1/2),2)
c <- f + (1/2)
g <- 0
if (f + r >= c - (1/4)) g <- c else g <- f
g <- round(g,1)
if (g <= 1.0) return("D")
```

```
if (g == 1.5) return("D+")
if (g == 2.0) return("C")
if (g == 2.5) return("C+")
if (g == 3.0) return("B")
if (g == 3.5) return("B+")
if (g >= 4.0) return("A")

}

# Wrapper function to predict grades for a student
# Predict any courses the student hasn't had a grade

predict_gpa <- function(student_record, save = FALSE){
row <- which(dimnames(raw_data)[[1]] == student_record)
if (length(row) == 0) row <- student_record # student_record can
be the student id, or the row number
if (!is.numeric(row)) stop("Student not found")

# For each algorithm, create a data frame containings the grade and
GPA.

GPA_NMF <- data.frame(Course = names(round(data.hat.nmf[row,][!is.na(data.hat.nm
Grade.Predicted = sapply(round(data.hat.nmf[row,][!is.na(data.hat.nmf[row,])],2)
gpa2grade),
GPA.Predicted = round(data.hat.nmf[row,][!is.na(data.hat.nmf[row,])],2))

if (save) {
fname <- paste0("Prediction for ", dimnames(raw_data)[[1]][row],
".csv")
write.csv(GPA_NMF, fname, row.names = FALSE)
}
return(GPA_NMF)
}

# Using the prediction function

predict_gpa("NUST201201346BSEECS60412F") # This will output the prediction
results using UBCF

predict_gpa("NUST201201346BSEECS60412F", save = TRUE) # save = TRUE
will save the prediction to a csv file
```

Bibliography

[1] Ryan SJD Baker and Kalina 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.

[2] Jonathan Baron and M Frank Norman. Sats, achievement tests, and high-school class rank as predictors of college performance. *Educational and Psychological Measurement*, 52(4):1047–1055, 1992.

[3] Hall P Beck and William D Davidson. Establishing an early warning system: Predicting low grades in college students from survey of academic orientations scores. *Research in Higher Education*, 42(6):709–723, 2001.

[4] Brent Bridgeman, Laura McCamley-Jenkins, and Nancy Ervin. Predictions of freshman grade-point average from the revised and recentered sat® i: Reasoning test. *ETS Research Report Series*, 2000(1):i–16, 2000.

[5] Tomas Chamorro-Premuzic and Adrian Furnham. Personality, intelligence and approaches to learning as predictors of academic performance. *Personality and individual differences*, 44(7):1596–1603, 2008.

[6] Hung-Hsuan Chen, Liang Gou, Xiaolong Zhang, and Clyde Lee Giles. Collabseer: a search engine for collaboration discovery. In *Proceedings of the 11th annual international ACM/IEEE joint conference on Digital libraries*, pages 231–240. ACM, 2011.

[7] Hung-Hsuan Chen, II Ororbia, G Alexander, and C Lee Giles. Expertseer: A keyphrase based expert recommender for digital libraries. *arXiv preprint arXiv:1511.02058*, 2015.

[8] Brad S Chissom and Doris Lanier. Prediction of first quarter freshman gpa using sat scores and high school grades. *Educational and Psychological Measurement*, 35(2):461–463, 1975.

[9] Asmaa Elbadrawy, Agoritsa Polyzou, Zhiyun Ren, Mackenzie Sweeney, George Karypis, and Huzefa Rangwala. Predicting student performance using personalized analytics. *Computer*, 49(4):61–69, 2016.

[10] Alexander Felfernig, Klaus Isak, Kalman Szabo, and Peter Zachar. The vita financial services sales support environment. In *Proceedings of the National Conference on Artificial Intelligence*, volume 22, page 1692. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2007.

[11] Vojtěch Franc, Václav Hlaváč, and Mirko Navara. Sequential coordinate-wise algorithm for the non-negative least squares problem. In *International Conference on Computer Analysis of Images and Patterns*, pages 407–414. Springer, 2005.

[12] R Francesco, R Lior, and S Bracha. Introduction to recommender systems handbook, recommendersystems handbook, 2011.

[13] Simon Funk. Netflix update: Try this at home, 2006.

[14] Kazumasa Goda, Sachio Hirokawa, and Tsunenori Mine. Correlation of grade prediction performance and validity of self-evaluation comments. In *Proceedings of the 14th annual ACM SIGITE conference on Information technology education*, pages 35–42. ACM, 2013.

[15] Lev Grossman. How computers know what we want—before we do. *Time Magazine*, 27, 2010.

[16] Pankaj Gupta, Ashish Goel, Jimmy Lin, Aneesh Sharma, Dong Wang, and Reza Bosagh Zadeh. Wtf: The who-to-follow system at twitter. In *Proceedings of the 22nd international conference on World Wide Web WWW*, 2013.

[17] William J Hawkins, Neil T Heffernan, and Ryan SJD Baker. Learning bayesian knowledge tracing parameters with a knowledge heuristic and empirical probabilities. In *International Conference on Intelligent Tutoring Systems*, pages 150–155. Springer, 2014.

[18] Bhrigu Kapur, Nakin Ahluwalia, and R Sathyaraj. Comparative study on marks prediction using data mining and classification algorithms. *International Journal of Advanced Research in Computer Science*, 8(3), 2017.

[19] Jared E Knowles. Of needles and haystacks: Building an accurate statewide dropout early warning system in wisconsin. *JEDM— Journal of Educational Data Mining*, 7(3):18–67, 2015.

[20] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, (8):30–37, 2009.

[21] Dee Norman Lloyd. Prediction of school failure from third-grade data. *Educational and Psychological Measurement*, 38(4):1193–1200, 1978.

[22] Yannick Meier, Jie Xu, Onur Atan, and Mihaela Van der Schaar. Predicting grades. *IEEE Transactions on Signal Processing*, 64(4):959–972, 2016.

[23] Sara Morsy and George Karypis. Cumulative knowledge-based regression models for next-term grade prediction. In *Proceedings of the 2017 SIAM International Conference on Data Mining*, pages 552–560. SIAM, 2017.

[24] J Neter, MH Kutner, CJ Nachtsheim, and W Wasserman. Applied linear statistical models (vol. 4). chicago: Irwin, 1996.

[25] Mirka Saarela and Tommi Kärkkäinen. Analysing student performance using sparse data of core bachelor courses. *Journal of educational data mining*, 7(1), 2015.

[26] Richard Sawyer and E James Maxey. The validity over time of college freshman grade prediction equations. 1979.

[27] Boaz Shulruf, John Hattie, and Sarah Tumen. The predictability of enrolment and first-year university results from secondary school performance: the new zealand national certificate of educational achievement. *Studies in Higher Education*, 33(6):685–698, 2008.

[28] Mack Sweeney, Jaime Lester, and Huzefa Rangwala. Next-term student grade prediction. In *Big data (Big Data), 2015 IEEE international conference on*, pages 970–975. IEEE, 2015.

[29] Mack Sweeney, Huzefa Rangwala, Jaime Lester, and Aditya Johri. Next-term student performance prediction: a recommender systems approach. *arXiv preprint arXiv:1604.01840*, 2016.

[30] Nguyen Thai-Nghe, Lucas Drumond, Tomáš Horváth, Lars Schmidt-Thieme, et al. Multi-relational factorization models for predicting student performance. In *Proc. of the KDD Workshop on Knowledge Discovery in Educational Data*, pages 27–40. Citeseer, 2011.

[31] Howard Wainer, Thomas Saka, and John R Donoghue. The validity of the sat at the university of hawaii: A riddle wrapped in an enigma. *Educational Evaluation and Policy Analysis*, 15(1):91–98, 1993.

[32] Ya-huei Wang and Hung-Chang Liao. Data mining for adaptive learning in a tesl-based e-learning system. *Expert Systems with Applications*, 38(6):6480–6485, 2011.

[33] Judith Zimmermann, Kay H Brodersen, Hans R Heinimann, and Joachim M Buhmann. A model-based approach to predicting graduate-level performance using indicators of undergraduate-level performance. *Journal of Educational Data Mining*, 7(3):151–176, 2015.