Personalized Multi-Regression Models for Predicting Students' Performance in Course Activities

Asmaa Elbadrawy
Computer Science
University of Minnesota
Twin Cities, MN
asmaa@cs.umn.com

R. Scott Studham
University of Minnesota
Twin Cities, MN
studham@umn.edu

George Karypis
Computer Science
University of Minnesota
Twin Cities, MN
karypis@cs.umn.com

ABSTRACT

Methods that accurately predict the grade of a student at a given activity and/or course can identify students that are at risk in failing a course and allow their educational institution to take corrective actions. Though a number of approaches have been developed for building such performance prediction models, they either estimate a single model for all students based on their past course performance and interactions with learning management systems (LMS), or estimate student-specific models that do not take into account LMS interactions; thus, failing to exploit fine-grain information related to a student's engagement and effort in a course. In this work we present a class of linear multi-regression models that are designed to produce models that are personalized to each student and also take into account a large number of features that relate to a student's past performance, course characteristics, and student's engagement and effort. These models estimate a small number of regression models that are shared across the different students along with student-specific linear combination functions to facilitate personalization. Our experimental evaluation on a large set of students, courses, and activities shows that these models are capable of improving the performance prediction accuracy by over 20%. In addition, we show that by analyzing the estimated models along with the student-specific combination functions we can gain insights on the effectiveness of the educational material that is made available at the courses of different departments.

Keywords

Regression with multi-regression models, Analyzing student behavior, Predicting student performance

1. INTRODUCTION

Motivated by the ongoing desire to improve the quality of education and address the ever increasing cost of higher education by ensuring that students graduate within four years, data mining techniques have been increasingly deployed to analyze the vast amounts of historical data being collected at various Colleges and Universities that pertain to students' academic performance. One of the problems that these techniques are trying to solve is to identify the students that are at risk of failing a course and thus allow the institution to take corrective actions by providing additional services and resources to the students and/or instructors.

Two general classes of approaches have been developed for solving this problem, both of which rely on supervised learning. The first uses a set of features related to the student's past course performance and interactions with online learning management systems (LMS) to estimate a single regression model [3]. This model is estimated to predict the student's course grade as a function of these features. The second uses factorization models, initially developed in the context of recommender systems, to predict students' grades for different course activities. Specifically, multi-relational models were used to predict students' performance by learning latent factors that satisfy student-task and task-skill relations [5] while approaches based on tensor factorization were used to take temporal aspects into consideration by modeling the fact that, over time, students acquire knowledge and develop expertise [7]. Unlike the models based on a single regression, factorization models can achieve better prediction accuracy as their prediction models are *personalized* to each student. However, these approaches entirely ignore the various features associated with how the students interact with the material/information provided in the LMS, which can potentially be used to improve the overall prediction accuracy.

In this work we investigate the effectiveness of a class of linear multi-regression models for predicting the students' performance at various course activities (e.g., quizzes and assignments). These models, that were inspired by previously developed approaches in the area of recommender systems [8, 2, 6], estimate a small number of linear regression models along with a student-specific linear function to combine them. The advantage of this approach is that the regression models are estimated by taking into account the historical information of all students that allows for cross-student information sharing and thus overcome issues related to data sparsity while providing accurate modeling of each student's unique characteristics via the user-specific linear combination function. The regression models utilize a wide-range of features that include the students' past course performance, their interactions with the LMS, and information related to the type of course and activity. We experimentally evaluated the performance of these models on a large dataset extracted from the University of Minnesota's Moodle installation [1] that contains 832 courses, 11,556 students, and 189,641 graded activities. The multi-regression models were able to achieve an RMSE of 0.147 whereas the RMSE of the corresponding single regression model was 0.177.

An advantage of the multi-regression model is that by clustering the students based on their combination weights we can segment them into groups whose prediction models are quite similar. By analyzing these groups we can gain insights on the factors that determine the students' performance, discover systematic differences across the groups, and identify areas for further analysis. Towards this end, we analyzed the combination weights for the multi-regression model consisting of just two regression models and identified three groups of students. The underlying regression models for two of these groups were different from each other in how much they rely on the LMS interaction features. In addition, some of the Departments in which the courses that these student took showed a high specificity to one of these groups. These results may suggest that the type of information that is provided in the LMS for certain departments may not be beneficial in improving the grades of the students and as such accessing it does not lead to better understanding and thus grades.

The rest of the paper is organized as follows. Section 2 describes the multi-regression model that we used. Section 3 describes the dataset that we used along with the various features that we extracted. Section 4 provides the experimental evaluation and analysis of the results. Finally, Section 5 provides some concluding remarks.

2. MODEL DESCRIPTION

We wish to learn a model that predicts the student grades within the different course activities, like assignments and quizzes, given some input features. To achieve this, we developed a linear multi-regression model inspired by [8] and [2]. In this model, the grade $\hat{g}_{s,a}$ for student s in activity a is estimated as

$$\begin{aligned}\hat{g}_{s,a} &= b_s + b_c + \boldsymbol{p}_s^t W \boldsymbol{f}_{sa} \\ &= b_s + b_c + \sum_{d=1}^{l} \Big(p_{s,d} \sum_{k=1}^{n_F} f_{sa,k} w_{d,k} \Big),\end{aligned} \tag{1}$$

where b_s and b_c are student and course bias terms, respectively, $\boldsymbol{f}_{sa}$ is a vector of length n_F that holds the input features (the predictors), l is the number of linear regression models, W is a matrix of dimensions $l \times n_F$ that holds the coefficients of the l linear regression models, and $\boldsymbol{p}_s$ is a vector of length l that holds the memberships of student s within the l different regression models. The term $w_{d,k}$ represents the weight of feature k under the d^{th} regression model, whereas the term $p_{s,d}$ represents the membership of student s in the d^{th} regression model; that is, how much the d^{th} regression model contributes to the grade estimation for student s. Throughout the rest of the paper, we will refer to the model parameters as the bias terms, the regression models' feature weights (referred to by the vectors $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_l$) and the students' memberships within the different regression models (referred to as $\boldsymbol{p}_s$ for each student s).

This multi-regression model has an advantage over learning a single linear regression model for all students since the multi-regression model is more personalized. Personalization is achieved through the student-specific membership weights which determine how much each linear model contributes to the grade estimation for the target student. It is also achieved through learning student-specific bias terms that reflect upon individual student differences. The course bias terms capture the grade patterns within the different courses. This multi-regression model also has an advantage over learning a different model for each student as it learns a smaller number of models that capture the performance patterns of the different student groups. Accordingly, it makes use of the similarities among the students (with respect to performance) and can better handle the data sparsity issue.

The parameters of the multi-regression model are estimated by solving a minimization process of the form

$$\underset{(W,P,B)}{\text{minimize}} \;\; \mathcal{L}(W,P,B) + \lambda(||P||_F^2 + ||W||_F^2), \tag{2}$$

where the loss function $\mathcal{L}(.)$ is the Root Mean Squared Error (RMSE) and W, P and B are the feature weights, students memberships and bias terms, respectively. The term $\lambda(||P||_F^2 + ||W||_F^2)$ controls the magnitude of the feature weights and the student memberships and thus prevents over-fitting. The scalar λ is fine-tuned in the process of estimating the model parameters.

In addition to accurately predicting students performance, the multi-regression model can be used to analyze how the different features contribute to the predicted grades and thus gain some insights about the students behavior. For proper analysis of the estimated model parameters, it is more convenient that all the returned feature weights, student memberships, and bias terms have non-negative values which will make all the model's components to contribute additively to the predicted grades. Accordingly, the optimization problem for learning the model parameters takes the form

$$\begin{aligned}&\underset{(W,P,B)}{\text{minimize}} \;\; \mathcal{L}(W,P,B) + \lambda(||P||_F^2 + ||W||_F^2), \; s.t. \\ &\qquad w_{d,c} \geq 0, \; 1 \leq d \leq l, \; 1 \leq c \leq n_F, \\ &\qquad p_{s,d} \geq 0, \; 1 \leq d \leq l, \forall s, \\ &\qquad b_c \geq 0, \; \forall c, \\ &\qquad b_s \geq 0, \; \forall s.\end{aligned} \tag{3}$$

The two minimization problems are solved using stochastic coordinate descent.

3. DATASET AND EVALUATION

We used a dataset extracted from the University of Minnesota's Moodle installation; which is one of the largest Moodle installations world wide. The dataset spans two semesters and it contains 832 course instances and 11,556 students. The courses belong to 157 different departments, each student has registered in at least 4 courses, and the total of number of assignment and quiz submissions are 114,498 and 75,143, respectively. The dataset also contained a total of 251,348 forum posts. We will refer to the assignments and quizzes as *activities*. The activity grades are normalized to be in the range $[0, 1]$.

3.1 Feature Description

For each student-activity pair (s, a), we constructed a feature vector $\boldsymbol{f}_{sa}$ whose features $\boldsymbol{f}_{sa}$ fall into three categories: student-specific features, activity-specific features and Moodle-interaction features. Each of these features are described next.

3.1.1 Student-specific features

These are features related to the student and they mainly describe the student's previous grade history. We use two student-specific features:

- *cumGPA*: The GPA accumulated over the courses previously taken by the student.
- *cumGrade*: The average grade achieved over all of the pervious activities in the course. For the first activity in the course, cumGrade is set to the cumGPA.

3.1.2 Activity-specific features

These are features that relate to the activity and the course that this activity belongs to. We use three activity-specific features:

- *activity type:* This can either be quiz or assignment. The activity type is handled by having two indicator values, one for quiz and one for assignment.
- *course level:* The course level takes an integer value of 1, 2, 3 or 4 and describes the course's level of difficulty with 4 being the most difficult. These levels are derived from the numeric designation of the courses.
- *department:* The department to which the course belongs to. Departments are handled by having one indicator feature per department. The feature corresponding to the department of the training instance is set to 1 and the rest are set to 0.

3.1.3 Moodle interaction features

These features describe the student's interaction with Moodle prior to the due date of the activity. These features were extracted from Moodle's log files and are the following:

- *n-init-disc*: The number of discussions initiated by the student.
- *n-engaged-disc*: The number of times that the student posted to an open discussion.
- *n-read-posts*: The number of forum discussions that are read by the student.
- *n-viewed-mater*: The number of times the student viewed some course material.
- *n-add-contrib*: The number of times the student contributed to the course by adding something to the course page (e.g., a wiki-page).
- *n-other-accesses*: The number of times the student made any other kind of access to the course pages. This feature is concerned with the student's interaction with the other Moodle modules (e.g., surveys).

For each of the above Moodle interaction features, we created five different features that measured the specified interaction at various time intervals prior to the due date. Four of them measure the interaction at $[0, 1)$, $[1, 2)$, $[2, 4)$, and $[4, 7)$ days prior to the due date, whereas the fifth measures the interaction up to the due date of the previous assignment. These features will be denoted by appending "-x" to the feature name, where x is the intervals' upper bound (e.g., *n-init-disc-1*, *n-init-disc-2*, *n-init-disc-4*, and *n-init-disc-7*), and the fifth will be denoted without the "-x" suffix. Note that for the forum interaction features, the collected numbers were normalized with respect to the total number of available forum discussions/posts.

3.2 Evaluation

The dataset was randomly split into training and test subsets containing 80% and 20% of the student-activity pairs, respectively. The model was trained on the training set and evaluated on the test set. This process was repeated 4 times and the obtained results on the test set were averaged and reported. The model is evaluated in terms of the root mean squared error (RMSE) between the actual and predicted grades on the test set.

3.3 Baseline Approach

We compare the performance of a multi-regression model against the performance of a linear regression model. The linear regression model estimates the student grades as

$$\hat{g}_{sa} = w_0 + \sum_{k=1}^{n_F} w_k f_k, \tag{4}$$

where f_k is the value of feature k and the w_k's are the regression coefficients of the linear regression model. Note that this linear regression model is different from a multi-regression model with one linear model since the latter estimates the student grade as

$$\hat{g}_{sa} = b_s + b_c + m_{s,1} \sum_{k=1}^{n_F} w_{1,k} f_k,$$

where b_s and $m_{s,1}$ are the student-specific bias and membership terms and b_c is the course-specific bias term.

4. RESULTS AND ANALYSIS

The results and analysis are presented in three parts. In the first part, we explore how the multi-regression model performs given different features and how it performs against a single linear regression model. In the second part we show how the different bias terms affect the performance of the multi-regression model. In the third part we analyze the estimated model parameters for the cases in which we learn one, two and three linear models in order to gain insights about the different student populations.

4.1 Multi-Regression Models Prediction Accuracy

In order to see the importance of the Moodle interaction features and how much they influence the accuracy of the predicted grades, we trained the multi-regression models and the baseline model twice; once using the student and activity features and once using the student, activity and the Moodle-interaction features.

Figure 1 shows the performances of the linear and the multi-regression models with and without using Moodle-interaction features[1]. The figure also shows the change in RMSE obtained by the multi-regression model as the number of linear models increases.

These results show that using a multi-regression model with one linear model gives an RMSE of 0.168 whereas using the linear regression model described by Equation 4 gives an RMSE of 0.223. This is due to the student-specific membership and bias terms which enable the multi-regression model to better capture individual student performances. Moreover, the course-specific bias terms can capture the grade distribution within the different courses.

Figure 1 also shows that the RMSE obtained by the multi-regression model decreases with increasing number of linear models. A larger number of linear models with student-specific memberships allow for more personalization. Using ten regression models, the obtained RMSE falls to 0.145. The incremental gains in RMSE saturate with increasing number of linear models.

Comparing the performance of the two multi-regression models in Figure 1, we can see that the model that uses the Moodle features performs better than the one that does not use them. A multi-regression model with ten linear models gives and an RMSE of 0.168 without using the Moodle features and gives an RMSE of 0.145 using the Moodle features. It can also be seen from the RMSE curves of the two multi-regression models that as the number of linear models increases, the incremental gains achieved by the model that does not use the Moodle features saturate faster than the incremental gains achieved by the other model. The use of Moodle features lead to more drop in RMSE with increasing number of regression models. We believe this is because the model that uses the Moodle features have more student-Moodle interaction information to learn from as the number of regression models increase.

4.2 Effect of the Bias Terms on the Prediction Accuracy of the Multi-Regression Models

In order to understand how the different bias terms contribute to the prediction accuracy, we trained the multi-regression model using each of the student and course bias terms separetely. Figure 2 shows the performance of multi-regression models that use different bias terms and different number of linear models. The plots show that the course bias contributes to the model accuracy more than the student bias.

4.3 Analyzing Feature Weights

For analyzing how the different features contribute to the predicted grades, we learn the multi-regression model using Equation 3 that has the non-negativity constrains. As mentioned in Section 2, and according to Equation 3, all the parameters returned by the model are non-negative. This

[1]These results were generated by learning the model without the non-negativity constrains according to Equation 2. Recall that these constrains are only used when we wish to analyze the model parameters and not when we predict the student grades.

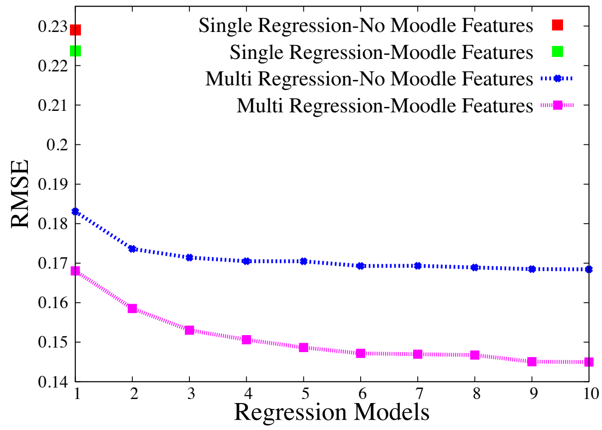

Figure 1: Change in RMSE as the number of regression models increases.

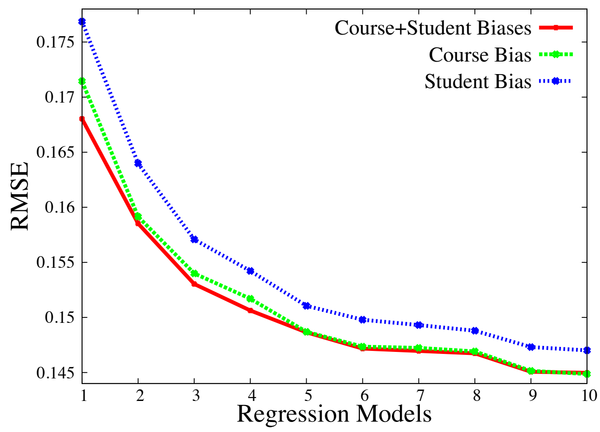

Figure 2: Effect of the different bias terms on the model performance.

way it is easy to compare the importance of the different features among the different estimated models. Note that the non-negativity constrains did not degrade the RMSE in a significant way. For the case of two linear models, the RMSE obtained with and without the non-negativity constrains are 0.165 and 0.160 respectively.

4.3.1 Determining Importance of Model Parameters

For each feature weight $w_{d,k}$, we would like to estimate how much it contributed to all the estimated grades. Given a grade $g_{s,a}$, and according to Equation 2, the weight $w_{d,k}$ contributes to $\hat{g}_{s,a}$ by $(m_{s,l} w_{d,k} f_{sa,k})$. Accordingly, the importance $i_{d,k}$ of the feature weight $w_{d,k}$ is accumulated using all the estimated grades as

$$i_{d,k} = \frac{\sum_{g_{s,a} \in \mathcal{G}} (m_{s,d} w_{d,k} f_{sa,k}) / \hat{g}_{s,a}}{|\mathcal{G}|},$$

where $\mathcal{G}$ is the set of all grades in the test set and $|\mathcal{G}|$ is the size of $\mathcal{G}$.

Similarly, we estimate the importance of the student and course biases by how much they contribute to all the pre-

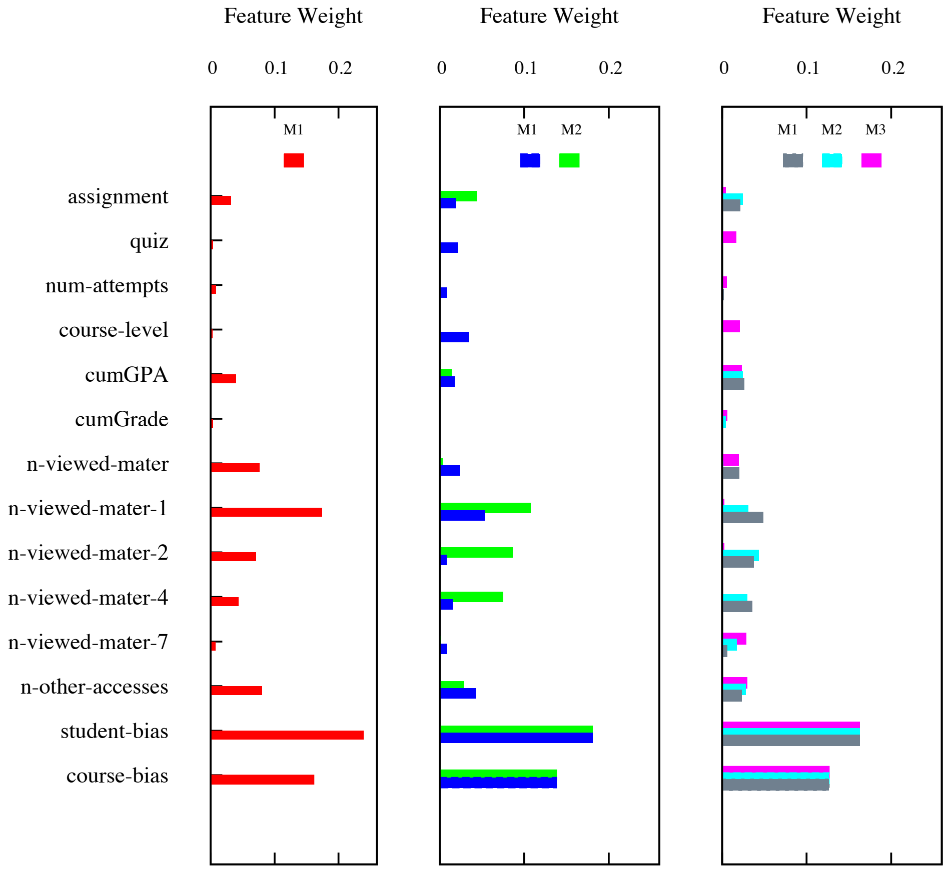

Figure 3: Feature weights for learning a multi-regression model with one linear model (left), two linear models (center) and three linear models (right).

dicted grades. The importance of a student bias term is estimate as

$$i_S = \frac{\sum_{g_{s,a} \in \mathcal{G}} b_s / \hat{g}_{s,a}}{|\mathcal{G}|},$$

and the importance of a course bias term is estimate as

$$i_C = \frac{\sum_{g_{s,a} \in \mathcal{G}} b_c / \hat{g}_{s,a}}{|\mathcal{G}|}.$$

4.3.2 Results

We analyze the estimated feature weights for learning a multi-regression model with one, two and three linear models. Figure 3 shows the estimated feature weights for one (left), two (center) and three regression models (right). The binary features representing the departments were omitted as well as the features with zero or very low importance values. The features related to the forum activities had very low importance values and thus were omitted from the figure. These features have very low importances as they only appear in a small fraction of the training data (between 10% and 25% of the training instances), whereas the features related to viewing the course material appeared in almost all the training instances.

In all three cases shown in Figure 3, the student bias, course bias and the features related to viewing the course material contribute the most to the predicted grades. In the case of two regression models, which we will refer to as M1 and M2, the quiz, number of attempts and course level are important under M1 and not M2. Another interesting point is that the features related to viewing the course material have higher importance under M2. In the case of three regression models, which we will refer to as M1, M2 and M3, the quiz, number of attempts and course level are important under M1 but not under M2 or M3, whereas the assignment and most of the features related to viewing the course material have higher importance under M2 and M3. The fact that we have some models concerned with assignments and others concerned with quizzes and their number of attempts reflects that the type and properties of the activity have an impact on predicting student performance.

4.4 Analyzing Student Memberships

Analyzing student memberships can give insights about the different student populations. We focus on the case in which we learn a multi-regression model with two linear models since this case is easy to visualize, and moreover, we have seen from Figure 3 that the features related to viewing the course material is more important under one of the two mod-

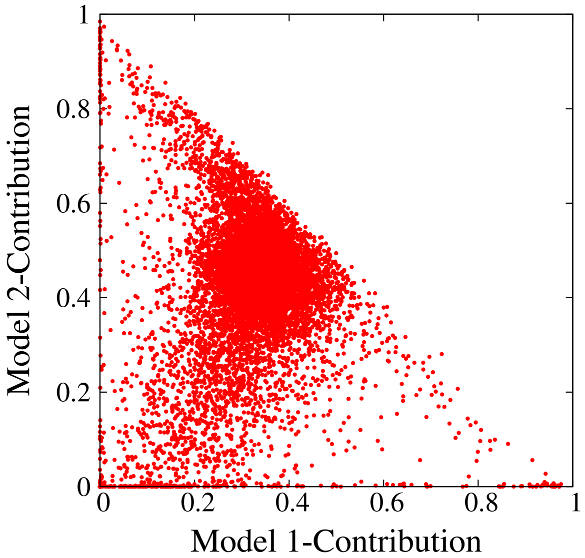

Figure 4: Student memberships for a multi-regression model with two linear models. Every point represents a student.

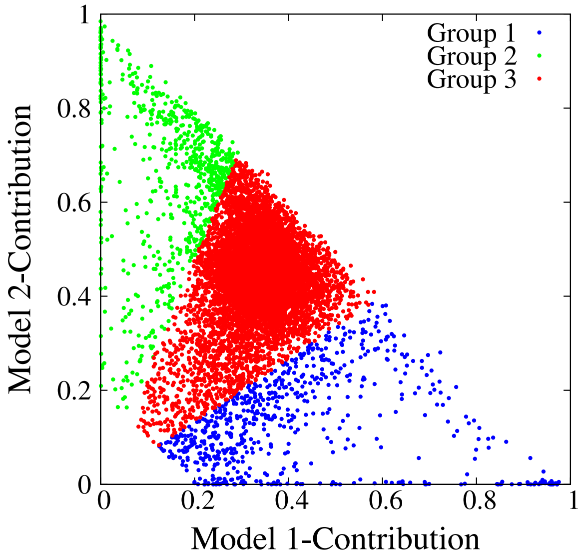

Figure 5: Student memberships for a multi-regression model with two linear models. Every point represents a student. See Section 4.4 for a description of the point coloring.

els. The latter point indicates that viewing the course material does not have the same impact on all students.

4.4.1 Determining Models Contributions to Student Grades

Given a multi-regression model with two linear models M1 and M2, we would like to estimate for each student s how much each of the two models contributes to the grades predicted for s. Given a grade $g_{s,a}$, and according to Equation 2, model d, where $d \in \{1,2\}$, contributes to $\hat{g}_{s,a}$ by $(m_{s,d}\sum_{k=1}^{n_F} w_{d,k} f_{sa,k})$. Accordingly, the contribution of model d to the grades of student s is estimated as

$$j_{s,d} = \frac{\sum_{g_{s,a}\in\mathcal{G}_s} m_{s,d}\sum_{k=1}^{n_F} w_{d,k} f_{sa,k}/\hat{g}_{s,a}}{|\mathcal{G}_s|},$$

where $\mathcal{G}_s$ is the set of all grades of student s, and $|\mathcal{G}_s|$ is the size of $\mathcal{G}_s$. The value $j_{s,d}$ estimates how much model d, where $d \in \{1,2\}$, contributes to the grades predicted for student s, taking into account the membership of s in d, $m_{s,d}$. The value $j_{s,d}$ lies in the range $[0,1]$, where $j_{s,d} = 0$ means model d does not contribute at all to the grades predicted for s, and $j_{s,d} = 1$ means that the grades of s are only estimated via model d. Accordingly, in our case we have $0 \le (j_{s,1} + j_{s,2}) \le 1$.

We have plotted $j_{s,1}$ against $j_{s,2}$ for each student s as shown in Figure 4. Each point corresponds to a student, and the x- and y-axis represent $j_{s,1}$ and $j_{s,2}$, respectively. Some students have almost the same contributions by the two models, whereas other students have a higher contribution by one of the two models. Since the two models differ in how much viewing course material influences the predicted grades, this can indicate that students with high contribution by one model can be different from students with high contribution by the other model.

In order to explore for student differences, we divided the students into three different groups based on their $(j_{s,1}, j_{s,2})$ values using the following procedure: First, we normalized the model contributions $j_{s,1}$ and $j_{s,2}$ for each student to have a one-norm of 1. Second, we estimated the mean and standard deviation of $j_{s,1}$ for all students, and we got $(\mu,\sigma) = (0.453, 0.142)$. Finally, we clustered the students into three different groups as follows:

- Student (Group 1) contains all students with $j_{s,1} > \mu + \sigma$.
- Student (Group 2) contains all students with $j_{s,1} < \mu - \sigma$.
- Student (Group 3) contains all students with $\mu - \sigma \le j_{s,1} \le \mu + \sigma$.

Student Groups 1, 2 and 3 are shown in Figure 5 in blue, green and red colors, respectively. We have omitted all students with $(j_{s,1} + j_{s,2}) < 0.2$, that is, students with both models contributing to their grades by less than 0.2. Groups 1, 2 and 3 contain 847, 797 and 7824 students, respectively.

4.4.2 Analysis of Student Groups

The distribution of the GPA for the three student groups is shown in Figure 6. Group 1 has a GPA average that is lower than Groups 2 and 3, and a higher GPA standard deviation than the other two groups.

We have finally investigated the relative appearance of the different departments within the three student groups. For each (department, group) pair (d, g), we have computed the appearance as

$$q_{d,g} = \frac{n_{d,g}/n_g}{\sum_{h=1}^{3} n_{d,h}/n_h},$$

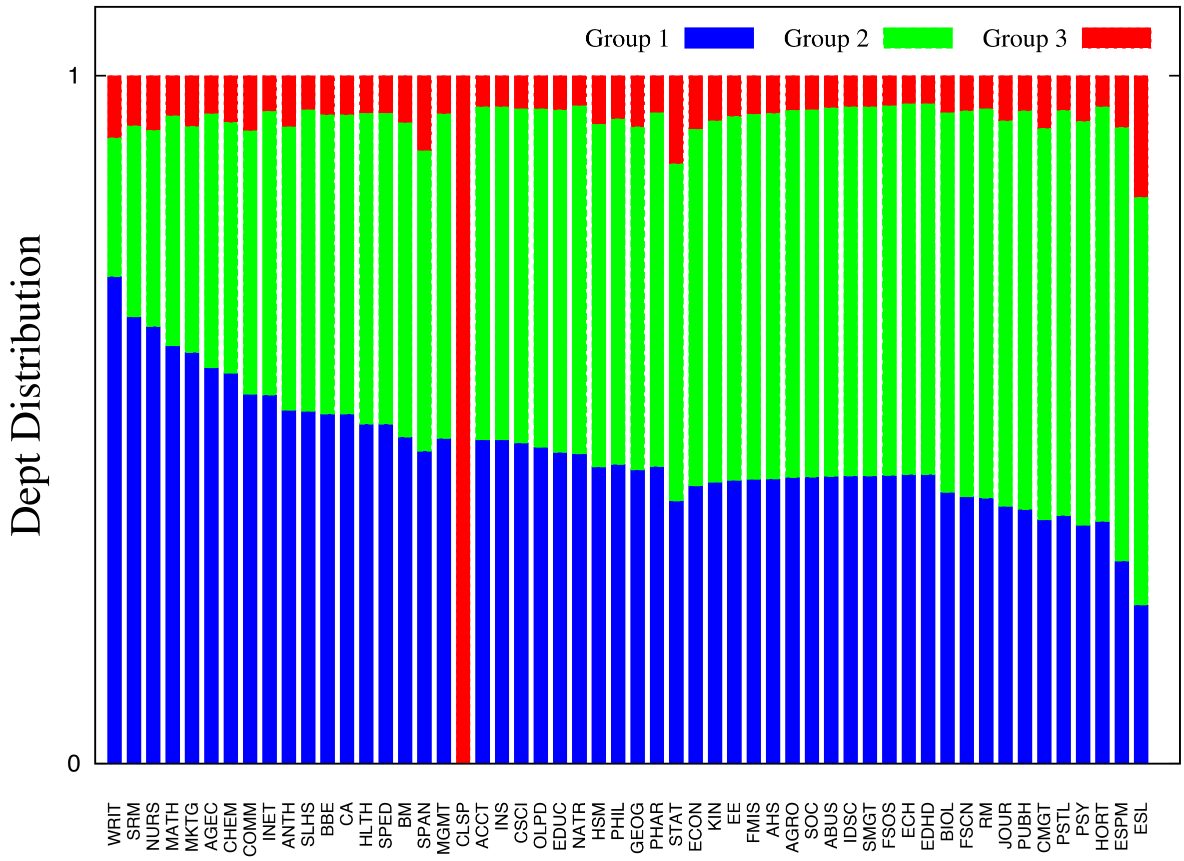

Figure 7: Relative appearance for the different departments within the three student groups.

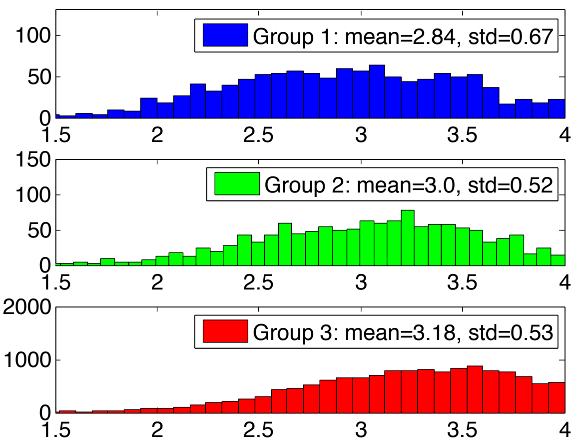

Figure 6: GPA distribution for the three student groups.

where $n_{d,g}$ is the number of students belonging to group g and are enrolled in courses belonging to department d, and n_g is the number of students belonging to gtoup g. The $q_{d,g}$ metric can be interpreted as the probability of group g given department d. Figure 7 shows how the different departments tend to appear within each group. Each department is represented by a vertical line. The vertical line has a red, a blue and a green part corresponding to $q_{d,G1}$, $q_{d,G2}$ and $q_{d,G3}$, that is the department's appearance within student Groups 1, 2 and 3, respectively. An interesting finding is that some departments primarily appear under Group 1 and almost never appear under Group 2 (the departments towards the left of the figure like Writing, Nursing and Math). These departments tend to have students whose access to course material is less influencing of their predicted grades. This finding may suggest that the type of material provided in the LMS for the departments that intensively appear with Group 1 may not be addressing the right student needs and therefore are not beneficial in improving the grades of the students. Accordingly, students' access to such non-beneficial material does not lead to better understanding and thus does not lead to better grades.

5. CONCLUSIONS

In this work, we have used a multi-regression model to predict student performance in course activities and analyze the resulting student populations. We have shown that a multi-regression model performs better than single linear regression as it captures personal student differences through the student-specific membership weights. We have also shown that the RMSE tends to decrease with increasing the number of linear regression models and thus allowing room for more personalized predictions. We have also shown that using the Moodle interaction features lead to an improved prediction accuracy.

Analyzing the estimated parameters of the multi-regression model showed that the student bias, course bias and features related to viewing the course material are the factors that mostly contribute to the predicted grades. The analysis also showed that the activity-specific features had different contributions within the different linear models. Moreover, the analysis of the different student populations showed that the features relating to viewing of course material contribute to the predictions of a certain student subpopulation higher than other students. It also appeared that some departments tend to have students whose viewing of course material contribute to their predicted grades less than other students. This might indicate that the material provided in the LMS for these departments may not be addressing the right student needs and thus are not helping students achieving better grades.

6. REFERENCES

[1] http://moodle2.umn.edu.

[2] D. Agarwal and B.-C. Chen. Regression-based latent factor models. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD, 2009.

[3] R. Barber and M. Sharkey. Course correction: Using analytics to predict course success. In *International Conference on Learning Analytics and Knolwedge*, LAK. ACM, 2012.

[4] L. Bottou. Online Algorithms and Stochastic Approximations. In *Online Learning and Neural Networks*. Cambridge University Press, 1998.

[5] T. H. Nguyen Thai-Nghe, Lucas Drumond and L. Schmidt-Thieme. Multi-relational factorization models for predicting student performance. In *KDDinED Workshop*, KDD. ACM, 2011.

[6] S. Rendle. Factorization machines with libFM. *ACM Trans. Intell. Syst. Technol.*, 3(3), May 2012.

[7] N. Thai-Nghe, T. Horváth, and L. Schmidt-Thieme. Factorization models for forecasting student performance. In *International Conference on Educational Data Mining*, pages 11–20, 2011.

[8] L. Zhang and Y. Zhang. Discriminative factored prior models for personalized content-based recommendation. In *ACM International Conference on Information and Knowledge Management*, CIKM, 2010.