Identifying At-Risk Students in Massive Open Online Courses

Jiazhen He†⋆ James Bailey†⋆ Benjamin I. P. Rubinstein† Rui Zhang†
†Department of Computing and Information Systems, The University of Melbourne, Australia
⋆Victoria Research Laboratory, National ICT Australia
jiazhenh@student.unimelb.edu.au, {bailey, brubinstein, rui.zhang}@unimelb.edu.au

Abstract
Massive Open Online Courses (MOOCs) have received widespread attention for their potential to scale higher education, with multiple platforms such as Coursera, edX and Udacity recently appearing. Despite their successes, a major problem faced by MOOCs is low completion rates. In this paper, we explore the accurate early identification of students who are at risk of not completing courses. We build predictive models weekly, over multiple offerings of a course. Furthermore, we envision student interventions that present meaningful probabilities of failure, enacted only for marginal students. To be effective, predicted probabilities must be both well-calibrated and smoothed across weeks. Based on logistic regression, we propose two transfer learning algorithms to trade-off smoothness and accuracy by adding a regularization term to minimize the difference of failure probabilities between consecutive weeks. Experimental results on two offerings of a Coursera MOOC establish the effectiveness of our algorithms.

Introduction
With the booming popularity of Massive Open Online Courses (MOOCs), such as Coursera, edX and Udacity, MOOCs have attracted the attention of educators, computer scientists and the general public. MOOCs aim to make higher education accessible to the world, by offering online courses from universities for free, and have attracted a diverse population of students from a variety of age groups, educational backgrounds and nationalities. Despite these successes, MOOCs face a major problem: low completion rates. For example, Table 1 shows the student participation in the first offering of a Coursera MOOC Discrete Optimization (DisOpt) launched in 2013; actions are measured in terms of viewing/downloading lectures and completing quizzes/assignments.

<table>
<thead>
<tr>
<th></th>
<th>DisOpt MOOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of students enrolled</td>
<td>51,306</td>
</tr>
<tr>
<td>Number of students with actions</td>
<td>27,679</td>
</tr>
<tr>
<td>Number of students completed</td>
<td>795</td>
</tr>
</tbody>
</table>

Early prediction can help instructors design interventions to encourage course completion before a student falls too far behind. We focus on Coursera MOOCs, which often last for several weeks with students engaging in activities such as watching/downloading lectures, attempting assignments/quizzes, and posting to/viewing discussion forums. To obtain early predictions, we build models weekly and leverage multiple offerings of a course to obtain ground truth to supervise the training of our models. Exploration of predictive analysis on MOOCs across multiple offerings has been limited thus far, but is nonetheless important, since data distributions across offerings is likely non-stationary: e.g., different cohorts of students enroll in offerings, and course materials (lectures and assignments) are refined over time. It is not clear a priori whether a model trained on previous offerings will serve a new offering well.

A key aspect of our approach is a plan for interventions that involve presenting at-risk students with meaningful probabilities of failure. We hypothesize that such carefully crafted interventions could help students become aware of their progress and potentially persist. However a necessary condition for such an approach to be effective, is to have probabilities that are well calibrated. By focusing on intervening with only those students near the pass/fail borderline, we aim for students who could be motivated by being ‘nearly there’ in succeeding in the class. Our intervention plan expressly avoids displaying failure probabilities for high-risk students, for fear of discouraging them from further participation in the course. Therefore calibration is not necessary across the entire unit interval, only near 0.5.

In this paper, we explore the accurate and early identification of students who are at risk of not completing courses.

Copyright © 2015, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.
ing students or undermining credibility of the intervention system. Therefore, we impose a requirement of smoothed probabilities across consecutive weeks. Towards this end, we propose two transfer learning algorithms—Sequentially Smoothed Logistic Regression (LR-SEQ) and Simultaneously Smoothed Logistic Regression (LR-SIM)—to balance accuracy with smoothness. These algorithms add a regularization term, which takes the probabilities in consecutive weeks into account, so as to minimize their difference. While LR-SEQ uses knowledge from the previous week to smooth the current week in a sequential fashion, LR-SIM learns across weeks simultaneously.

Contributions. The main contributions of this paper are:

- The first exploration of early and accurate prediction of students at risk of not completing a MOOC, with evaluation on multiple offerings, under potentially non-stationary data;
- An intervention that presents marginal students with meaningful failure probabilities: to the best of our knowledge a novel approach to completion rates;
- Two transfer learning logistic regression algorithms which would be practical for deployment in MOOCs, for balancing accuracy & inter-week smoothness. Training converges quickly to a global optimum in both cases; and
- Experiments on two offerings of a Coursera MOOC that establish the effectiveness of our algorithms in terms of accuracy, inter-week smoothness and calibration.

Related Work

Low completion rates is a major problem in MOOCs. One way to address this problem is to identify at-risk students early and deliver timely intervention. A few studies have focused on predicting students’ success/failure in MOOCs. Jiang et al. (2014) use students’ Week 1 assignment performance and social interaction to predict their final performance in the course. Ramesh et al. (2013) analyze students’ online behavior and identify two types of engagement, which is then used as a latent feature to help predict final performance. The same methodology is then used to predict a similar task for whether students submitted their final quizzes/assignments (Ramesh et al. 2014). However, these predictions are not studied for intervention. Instead, we propose to intervene students by presenting meaningful failure probabilities: to the best of our knowledge a novel approach to completion rates.

Problem Statement

We explore the accurate and early prediction of students who are at risk of failing, which we cast as a supervised binary classification task where possible class labels are whether or not a student will fail a course.

Predicted probabilities can serve a dual purpose, both for the identification of at-risk students and within subsequent intervention. We hypothesize that carefully employing the predicted probabilities as part of an intervention message could incentivize students to invest further in the course. Specifically, we propose to intervene with those who are on the pass/fail borderline rather than high-risk students. For example, given a 0.45 predicted probability, a hypothetical intervention message might resemble the following.

Great work on your efforts so far—you’re nearly there! In fact our statistical models suggest your profile matches students with a 55% chance of passing. This is based mainly on your lecture downloads this week. We’d like to encourage you to watch lecture 4 and post to the board. Doing just these 2 activities have greatly improved outcomes for students like you!
By targeting only those students near the pass/fail border, we are focusing on the part of the cohort that with an incremental investment could most personally benefit and increase the course pass rate.

Our application motivates 4 requirements of the learner:

- **Early & accurate predictions** enable timely interventions for at-risk students, with minimal unfounded and missing interventions;
- **Well-calibrated probabilities** allow proper targeting of interventions to those students who are truly near the classifier’s decision boundary and to supply meaningful interventions—e.g., approximately 60% of students with a risk prediction of 0.6 should eventually fail the course;
- **Smoothed probabilities** across consecutive weeks mitigate large fluctuations from slight changes in activities. Such fluctuations (cf. Figure 1) may undermine the credibility of intervention messages—we opt for consistent feedback. Moreover smoothing admits a principled approach to learning from the entire course when distributions change and even feature spaces change (i.e., a form of regularization through transfer learning); and
- **Interpretable models** suggest additions to intervention messages such as explanations for the current prediction and possible areas for improvement. Moreover such models can be useful in informing instructors on the profiles of successful vs. struggling students.

![Figure 1: Failure-probability trajectories for three students across nine weeks produced by logistic regression with cross-validation performed weekly on DisOpt launched in 2014.](image)

**Algorithms**

In initial experiments we explored a variety of supervised binary classifiers for predicting failure weekly: regularized logistic regression, SVM (LibSVM), random forest, decision tree (J48), naïve Bayes, and BayesNet (in weka with default parameters used). Results (omitted due to space) indicate that regularized logistic regression performs best in terms of Area Under the ROC Curve (AUC), followed by BayesNet, naïve Bayes, random forest, decision tree and SVM. Only BayesNet is comparable to logistic regression, whilst SVM performs worst. In addition to the advantage of outperforming other classifiers, logistic regression: produces interpretable linear classifiers with weights indicating relative importance (under certain assumptions); is naturally well-calibrated (Niculescu-Mizil and Caruana 2005b); and is a technique widely appreciated by researchers in the education community. Therefore in the sequel we focus our attention on approaches based on logistic regression.

To address smoothness, we propose two adaptations to logistic regression. To aid their development, we first review basic regularized logistic regression. A glossary of symbols used in this paper is given in Table 2.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$n$</td>
<td>The number of weeks</td>
</tr>
<tr>
<td>$n_i$</td>
<td>The number of students by week $i$</td>
</tr>
<tr>
<td>$n_{i,i-1}$</td>
<td>The number of extant students by both week $i$ and week $i-1$</td>
</tr>
<tr>
<td>$x_i$</td>
<td>The set of students by week $i$</td>
</tr>
<tr>
<td>$x_{ij}$</td>
<td>The $j$th student by week $i$</td>
</tr>
<tr>
<td>$d_i$</td>
<td>The number of features for student $x_{ij}$</td>
</tr>
<tr>
<td>$x_i^{(-1,i)}$</td>
<td>The set of students in week $i$ also existing in week $i-1$</td>
</tr>
<tr>
<td>$x_i^{(i-1,i)}$</td>
<td>The set of students with extended feature space by week $i$</td>
</tr>
<tr>
<td>$w_i$</td>
<td>The weight vector for week $i$</td>
</tr>
<tr>
<td>$w$</td>
<td>The weight vector for all weeks</td>
</tr>
<tr>
<td>$y_i$</td>
<td>The set of labels for students by week $i$</td>
</tr>
<tr>
<td>$y_{ij}$</td>
<td>The label of $j$th student by week $i$</td>
</tr>
<tr>
<td>$\lambda_1$</td>
<td>Regularization parameter for overfitting</td>
</tr>
<tr>
<td>$\lambda_2$</td>
<td>Regularization parameter for smoothness</td>
</tr>
</tbody>
</table>

**Logistic Regression (LR)**

Let $n$ be the number of weeks that a course lasts for. We have $n_i$ students by the end of week $i$ ($1 \leq i \leq n$). $x_i = \{x_{i1}, x_{i2}, \ldots, x_{in_i}\}$ is the set of students in week $i$. Each student $x_{ij}$ is described by $d_i$ features. Note that the number of students by the end of each week $i$ can be different, since students can enter a course at any time while it is running.

Logistic regression predicts label $y$ (fail for $y=1$ and pass for $y=1$) for input vector $x_{ij}$ (a student) according to,

$$p(y|x_{ij}, w_i) = \sigma(yw_i^T x_{ij}) = \frac{1}{1 + \exp(-yw_i^T x_{ij})}$$

(1)

where $w_i = [w_{i1}, w_{i2}, \ldots, w_{iH}]^T$ is the weight vector to be learned.

From a data set by week $i$, given by $(x_i, y_i) = [(x_{i1}, y_{i1}), (x_{i2}, y_{i2}), \ldots, (x_{in_i}, y_{in_i})]$, we wish to find $w_i$ by $L_2$-regularized maximum likelihood estimation: minimizing with regularization parameter $\lambda_1 > 0$,

$$\mathcal{L}(w_i) = \sum_{j=1}^{n_i} \log(1 + \exp(-y_{ij}w_i^T x_{ij})) + \frac{\lambda_1}{2} ||w_i||^2$$

(2)
can be non-stationary as engagement varies and prescribed
add a regularization term minimizing the difference between
ing approaches to linear classifiers (Ando and Zhang 2005):
In our setting, the previous week’s knowledge is used to help
knowledge learned in related tasks to better learn a new task.
In our setting, the previous week’s knowledge is used to help
learn smoothed probabilities for the current week.
A natural approach is to follow existing transfer learning
approaches to linear classifiers (Ando and Zhang 2005):
add a regularization term minimizing the difference between
w i and w i−1. However, the data distribution across weeks
can be non-stationary as engagement varies and prescribed
activities evolve. Moreover the number of features might change (d i ̸= d i−1). Instead we seek to minimize the differ-
ence between predicted probabilities between two weeks
directly. Unfortunately this leads to a non-convex objective.
Therefore we minimize a surrogate: the difference between
w i x i−1,i and w i−1 x i−1,i , where x i−1,i denotes the set of
students in week i that also exist in week i − 1, and similarly
x i−1,i denotes the set of students in week i − 1 that also exist in week i. The objective function1 for week i is
\[ \mathcal{L}(w_i) = \sum_{j=1}^{n_i} \log(1 + \exp(-y_j w_i^T x_{ij})) + \frac{\lambda_1}{2} \|w_i\|^2 + \lambda_2 \sum_{j=1}^{n_i-1} \|w_i^T x_{ij} - w_{i-1}^T x_{i-1,j}\|^2 \]  
where parameter \( \lambda_2 > 0 \) controls smoothness and the level
of transfer. This surrogate objective function is convex there-
for efficiently solved by Newton-Raphson to a guaran-
teed global optimum. To recap: n weekly logistic regression
models are learned sequentially such that week i’s model
cannot be built until model for week i − 1 is obtained.

Sequentially Smoothed LR (LR-SEQ)
In order to smooth probabilities across weeks, we propose
a transfer learning algorithm, *Sequentially Smoothed Log-
istic Regression (LR-SEQ)*. Transfer learning leverages
the knowledge learned in related tasks to better learn a new task.
In our setting, the previous week’s knowledge is used to help
learn smoothed probabilities for the current week.

Simultaneously Smoothed LR (LR-SIM)
The drawback of LR-SEQ is that early inaccurate predic-
tions cannot benefit from the knowledge learned in later
weeks (where data is closer to the end of the course), in-turn
undermining models learned later. To combat this effect, we
propose *Simultaneously Smoothed Logistic Regression (LR-
SIM)* that simultaneously learns models for all weeks. LR-
SIM allows early and later prediction to be correlated and
to influence each other, which we expect should yield im-
proved prediction due to inter-task regularization but also
good smoothness.

We first extend the feature space for each student x i,j to
a new space with n components. The student x i,j with new
feature space has \( \sum_{i=1}^{n} d_i \) dimensions, with the ith compo-
nent having \( d_i \) features corresponding to the features in
the original feature space by the end of week i, and others zero.

For example, for a student at the end of week 2, x 2,j, we ex-
tend to a new point x ′ 2,j, where the 2nd component with \( d_2 \)
features are actually the same as x 2,j, and others being zero.
Hence we encode the same information by the end of week
2 that contributes to the outcome. We must learn a single \( w \),
which also has \( \sum_{i=1}^{n} d_i \) dimensions corresponding to x i,j.
But only the \( i \)th component—the \( i \)th model—contributes to
the prediction by the end of week i, due to the zero values of
other dimensions of x i,j.

\[ \begin{align*}
   x^1_{ij} & \quad 1 \\
   x^2_{ij} & \quad x^1_{ij} \quad [0, \ldots, 0] \quad \cdots \quad [0, \ldots, 0] \\
   x^n_{ij} & \quad [0, \ldots, 0] \quad x^2_{ij} \quad \cdots \quad [0, \ldots, 0] \\
   \vdots & \quad \vdots \quad \vdots \quad \ddots \quad \vdots \\
   x^n_{nj} & \quad [0, \ldots, 0] \quad [0, \ldots, 0] \quad \cdots \quad x^n_{nj}
\end{align*} \]

Based on the extended x i,j and w, we can minimize the
difference of probabilities predicted for week i and week
i − 1 for i (i \geq 2) together, via a simple expression, as
shown in Eq. (4). Again the objective function is convex and
efficiently minimized.

\[ \begin{align*}
   \mathcal{L}(w) = & \sum_{i=1}^{n} \sum_{j=1}^{n_i} \log(1 + \exp(-y_j w^T x_i,j)) + \frac{\lambda_1}{2} \|w\|^2 \\
   & + \lambda_2 \sum_{i=2}^{n} \sum_{j=1}^{n_{i-1}} \|w^T x_i^{-1,i,j} - w^T x_{i-1,j}\|^2 \quad (4)
\end{align*} \]

Our algorithms can operate for tasks with differing fea-
ture spaces and feature dimensions. For example, one might
use individual-level features for each lecture and assign-
ment, which might be released weekly, to help understand
and interpret student performance.

**Experimental Results**
We conduct experiments to evaluate the effectiveness of our
algorithms on real MOOCs.

**Dataset Preparation**

**Discrete Optimization Dataset**  The first offering of *Dis-
crete Optimization (DisOpt1)* launched in 2013 by The Uni-
versity of Melbourne lasted for nine weeks, with 51,306 stu-
dents enrolled, of which 795 students received a certificate
of completion for the course. This course has an open course
curriculum with all the videos and assignments released at
the beginning of the course, enabling students to study at
their own pace. There are 57 video lectures and 7 assign-
ments in total. Students can watch/download video lectures,
and attempt assignments multiple times. Their final grade is
assessed by the total score on 7 assignments.

The second offering of *Discrete Optimization (DisOpt2)*
launched in 2014 also lasted for nine weeks, attracting
33,975 students to enroll, of which 322 students completed.
There are 4 fewer video lectures compared to DisOpt1, with
43 video lectures. The number of assignments remain but some
of the assignment contents differ to those of DisOpt1. The
total score of all assignments differs between offerings.
An overview of the two offerings is shown in Table 3.
Table 3: Overview on two offerings for DisOpt

<table>
<thead>
<tr>
<th>Feature</th>
<th>DisOpt1</th>
<th>DisOpt2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duration</td>
<td>9 weeks</td>
<td>9 weeks</td>
</tr>
<tr>
<td>Number of students enrolled</td>
<td>51,306</td>
<td>33,975</td>
</tr>
<tr>
<td>Number of students completed</td>
<td>795</td>
<td>322</td>
</tr>
<tr>
<td>Number of video lectures</td>
<td>57</td>
<td>53</td>
</tr>
<tr>
<td>Number of assignments</td>
<td>7</td>
<td>7</td>
</tr>
<tr>
<td>Total score of all assignments</td>
<td>396</td>
<td>382</td>
</tr>
</tbody>
</table>

Table 4: Features for each week $i$ for DisOpt

<table>
<thead>
<tr>
<th>Feature</th>
<th>LR</th>
<th>LR-MOV</th>
<th>LR-SEQ</th>
<th>LR-SIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Percentage of lectures viewed/downloaded by week $i$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Percentage of lectures viewed/downloaded in week $i$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Percentage of assignments done by week $i$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Percentage of assignments done in week $i$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average attempts on each assignment done by week $i$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average attempts on each assignment done in week $i$</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Percentage of score on assignments done by week $i$, to total score on all assignments</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 2: Student participation in the first and second offering of Discrete Optimization

**Cohorts** Among all the students enrolled, only a tiny fraction complete, which makes the data extremely imbalanced. Figure 2 shows the number of students in different course activities. In DisOpt1, among all the students enrolled, only around 41%, 13% and 2% of the students watch/download videos, do assignments and complete the course respectively. The same thing happens in DisOpt2 with low completion rate, and DisOpt2 had fewer students enrolled. Students enroll for various reasons. For example, some treat MOOCs like traditional courses by taking lectures and assignments at a fixed pace, while others treat MOOCs as online references without doing any assignments (Anderson et al. 2014). In this paper, we are interested in helping those who intend to pass. Therefore, we focus on students who are active in assignments/quizzes, which indicates an intention to pass. In particular, at the end of each week, we retain the students who did at least one assignment by that week.

**Features Used** We extract features from student engagement with video lectures and assignments, and performance on assignments by the end of each week to predict their performance at the end of the course. The features are shown in Table 4. In order to easily apply the model trained on previous offerings to a new offering, we extract features present across offerings.

**Performance Measure**
To evaluate the effectiveness of our proposed algorithms, we train prediction models on DisOpt1, and test on DisOpt2. Due to the class imbalance where high proportion of students fail, we prefer area under the ROC curve (AUC), which is invariant to imbalance. To measure the smoothness for week $i$, we compute the difference of probabilities between week $i$ and week $i-1$ for each active student (in terms of our rule for maintaining students) in week $i$ and $i-1$, and obtain the averaged difference for all students, and standard deviation (stdev).

**Smoothness and AUC**
To evaluate the effectiveness of our proposed algorithms LR-SEQ and LR-SIM, we compare them with two baselines, LR and a simple method using moving averages, denoted LR-MOV. LR-MOV predicts as final probability for week $i$ an average of LR’s week $i$ and $i-1$ probabilities, ($i \geq 2$). The prediction for week 1 is the same as LR. We train models using the above four algorithms on DisOpt1, where $\lambda_1 = 10$ and $\lambda_2 = 1$, and apply them to DisOpt2. Figure 3 and Table 5 show the smoothness and AUC across weeks respectively.

Figure 3: Comparison of LR, LR-MOV, LR-SEQ and LR-SIM on smoothness across weeks. Mean difference of probabilities across students plus/minus standard deviation. Closer to zero difference is better.
smoothness as LR with reduced standard deviation, demonstrating the need for performing some kind of smoothing.

From Table 5, we can see that LR-SIM and LR-MOV are comparable to LR in terms of AUC, while LR-SEQ decreases slightly. (Note: LR-MOV cannot achieve better smoothness as shown in Figure 3.) LR-SIM does outperform LR in the first two weeks (in bold): one reason might be that the reduced model complexity due to transfer learning helps to mitigate overfitting; another reason might be that later, more accurate predictions improve early predictions via transfer learning in DisOptl and the data distributions over DisOptl and DisOpt2 do not significantly vary. On the other hand, LR-SEQ gets continually worse in the first three weeks: LR-SEQ only uses the previous week’s knowledge to constrain the present week, but early predictions might be inaccurate, which undermine models learned later (cf. week 3, with the worst AUC). Later, LR-SEQ catches up with LR as data closer to the end of the course becomes available.

Overall, LR-SIM and LR-SEQ outperform LR consistently in terms of smoothness. And LR-SIM maintains or even improves on LR’s AUC in early weeks, while LR-SEQ suffers slightly inferior AUC in the first few weeks, and is comparable to LR in the last few weeks. Notably, using the data collected by the end of early weeks we can achieve quite good AUC: about 0.87 by week 2 and 0.9 by week 3, establishing the efficacy of early identification of at-risk students. Furthermore, this demonstrates that a model trained on the first offering works well on the second offering.

Parameter Analysis

We compare the performance of LR-SIM, LR-SEQ and LR in terms of smoothness and AUC varying $\lambda_1$ and $\lambda_2$. Figure 4 shows results for week 2. We choose week 2 to emphasize early intervention. The curves from right to left show varying $\lambda_2$ from $10^{-4}$ to $10^{4}$. The smoothness is computed between week 2 and week 1, and AUC is for week 2. It can be seen that LR achieves good AUC but poor smoothness. LR-SIM dominates LR-SEQ. As $\lambda_2$ increases, LR-SEQ and LR-SIM get smoother. But LR-SIM can achieve better AUC while LR-SEQ gets worse. Overall, LR-SIM clearly outperforms LR-SEQ and LR.

Calibration

Given an instance, it is not possible to know what the true underlying probability is, therefore some approximations are often used. A common way is to group instances based on the ranked predicted probability into deciles of risk with approximately equal number of instances in each group, and compare the predicted probability with observed probability within each group. A reliability diagram plotting the predicted probability with observed probability, is commonly used for calibration (Niculescu-Mizil and Caruana 2005a; Zhong and Kwok 2013).

Figure 5 shows the reliability diagram using LR-SIM for week 2. Our predicted probabilities agree closely with the observed probability in the gray region of marginal at-risk students for whom we wish to intervene.

Conclusion

We have taken an initial step towards early and accurately identifying at-risk students, which can help instructors design interventions. We have compared different prediction models, with regularized logistic regression preferred due to its good performance, calibration and interpretability. Based on the predicted probabilities, we envision an intervention that presents students meaningful probabilities to help them realize their progress. We developed two novel transfer learning algorithms LR-SEQ and LR-SIM based on
regularized logistic regression. Our experiments on Coursera MOOC data indicate that LR-SEQ and LR-SIM can produce smoothed probabilities while maintaining AUC, with LR-SIM outperforming LR-SEQ. LR-SIM has exceptional AUC in the first few weeks, which is promising for early prediction. Our experiments leveraging the two offerings of a Coursera MOOC demonstrate that the prediction models trained on a first offering work well on a second offering.

Model interpretability is important in learning analytics, where detailed feedback may be favored over generic feedback like ‘how’s it going?’. Such specifics can shed light on why a student is failing, and also what strategies other students follow to succeed. In particular, within logistic regression, the learned weight vectors can be used for explaining the contribution of each feature—albeit under certain assumptions on feature correlation. In these cases, features are not only important for prediction, but also for interpretability.

In the future, we will collaborate with course instructors to deploy our identification models and subsequent interventions in a MOOC for A/B testing to determine efficacy.

References
Kloft, M.; Stiehler, F.; Zheng, Z.; and Pinkwart, N. 2014. Predicting MOOC dropout over weeks using machine learning methods. In Proceedings of the EMNLP Workshop on Model interpretability is important in learning analytics, where detailed feedback may be favored over generic feedback like ‘how’s it going?’. Such specifics can shed light on why a student is failing, and also what strategies other students follow to succeed. In particular, within logistic regression, the learned weight vectors can be used for explaining the contribution of each feature—albeit under certain assumptions on feature correlation. In these cases, features are not only important for prediction, but also for interpretability.

In the future, we will collaborate with course instructors to deploy our identification models and subsequent interventions in a MOOC for A/B testing to determine efficacy.

References