**Identifying at-risk students across different stages of distance learning courses and identifying their most relevant predictors at each stage.**

Paul Duncker, Utrecht University

**Abstract**

**MOOCs have become commonplace in distance learning over the past decade, but they are facing high student dropout rates. A warning system that could identify at-risk students could decrease the withdrawal rate. Many prior studies investigated student dropout prediction models in MOOCs, but most of these studies used a single course or non-dynamic dataset. These models tend to overfit on dynamic real-world data since they trained on a small specific dataset. This paper contributes to the body of research by investigating student dropout performance of the XGBoost algorithm and evaluating the most important predictors at each stage on a diverse and dynamic dataset. For the analysis the OULA dataset is used, the dataset is pre-processed into three main predictors: demographics, assessment and log data (VLE interaction data). It is split up in eleven intervals wherein each interval the data gets richer. Two analyses are performed on the data. The first analysis is done on the dataset where the withdrawal data isn’t removed, this resulted in a prediction accuracy between 0.76 and 0.86 and the most important predictors are log and assessment data. The second analysis is done on the dynamic dataset, the data of dropped out students are removed after they withdrew from a course, this resulted in a performance that couldn’t beat the accuracy threshold. This implicates that XGBoost isn’t able to predict dropouts on a dynamic dataset.**

*Keywords*: Distance Learning, Machine Learning, MOOCs, OULAD, Student dropout, XGBoost

**Introduction**

Distance education has gained a lot of popularity since its first introduction. American universities and colleges that started online courses between 2002 and 2010 maintained an annual growth of registered students of 10 to 20% (Allen, I. E., & Seaman, J. (2011).) In figure 1 is shown how rapidly the number of MOOCs (Massive Open Online Courses), have grown. By the end of 2018, over 900 universities provided more than 11400 online courses in which over 101 million students enrolled (Shah, D, (2018)). It is highly acceptable to state that distance education has become commonplace.

Distance education differs from traditional education on several aspects. The biggest difference between the two is that distance education has a very free structure and is therefore very accessible and not limited by geographical boundaries, time or wealth. It has low (or even no) tuition fees, its content is accessible anywhere at any time, previous education isn't necessarily required, and it has unlimited enrollment. Where, on the other hand, traditional education charge (higher) tuition, have their content restricted to students only, require previous education and have limited enrollment.

However, there is a downside to the free structure of distance education, drop-out rates are extremely high. For example, at the Open University (UK), 78% of the students failed to complete their courses (Simpson, O. (2010)). And even worse, in 2012 only 5% of students that enrolled in a Coursera (one of



Figure 1, MOOCs growth over the year

the largest providers of MOOCs) MOOC officially completed the course (Piech, C., Huang, J., Chen, Z., Do, C., Ng, A., & Koller, D. (2013)). According to Hlosta, M., Zdrahal, Z., & Zendulka, J. (2017) most

drop-outs happen before or in the first 30 days of the module. The main reason for this high withdrawal rate is likely due to low student engagement and the lack of face-to-face contact between student and teacher.

A potential remedy for this problem is a warning system that identifies at-risk students with a high probability of premature withdrawal from courses. The course tutor can draw attention to the at-risk students that they might drop-out and the necessary help can be provided to these students in time. Alternatively, the prediction can be emailed directly to the student to make them aware.
The goal of this paper is to contribute to the development of early warning systems for distance education. I have investigated the XGBoost machine learning algorithm to predict withdrawals on different intervals of the Open University Learning Analytics Dataset (OULAD) (Kuzilek, J., Hlosta, M., & Zdrahal, Z. (2017)). The performance and the most relevant predictors of each interval dataset will be evaluated. The reason the OULAD is used, is that it is not thoroughly researched yet and it is publicly available as a CSV file. OULAD contains data from 32,592 students and 7 different course modules, and various data such as assessment scores and dates, log data (i.e. activity streams as a result of interacting with the virtual learning environment) and student demographics.

This paper is organized as follows: Section 2 presents related work that focuses on the prediction of drop-out rates in distance learning. Section 3 describes the dataset and presents the research methodology. Section 4 presents the evaluation results. Section 5 presents the conclusion and proposes additional research.

**Related Work**

This section reviews literature with a focus on predicting dropouts in distance learning using the OULAD.  In the field of learning analytics, multiple MOOCs datasets are researched. I decided to keep the discussion to the papers to those who use the OULAD, since it is researched in this work as well. To predict withdrawals, different machine learning techniques are applied to the OULAD, using predictors like demographics, log data, previous study results. The most important findings of each

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Citation | Algorithms | Ìnput Variables | Output variables |  Highest Accuracy |
| Heuer, H., & Breiter, A. (2018). | DT, RF, LR, SVM | Demographics, log data  | Fail |  0.91 |
| Hussain, M., Zhu, W., Zhang, W., & Abidi, S. M. R. (2018). | DT, JRIP, J48, GBT, CART, NBC | Log data, ast results, highest education, assessment scores | Student engagement |  - |
| Hassan, S. U., Waheed, H., Aljohani, N. R., Ali, M., Ventura, S., & Herrera, F. (2019). | LSTM, ANN, LR | Log data  | Early withdrawals | 0.97 |
| Jha, N., Ghergulescu, I., & Moldovan, A. N. (2019). | DRF, GBM, DL, GLM | Log data, demographics, Assessment scores | Withdrawal |  0.91 |
| Score |  0.93 |
| Hlosta, M., Zdrahal, Z., & Zendulka, J. (2017, March). | LR, SVM, NBC, RF, XGB | Assessment 1 | Withdrawal date |  - |
| Rizvi, S., Rienties, B., & Khoja, S. A. (2019) | DT | Demographics | Result |  0.83 |
| Alshabandar, R., Hussain, A., Keight, R., Laws, A., & Baker, T. (2018, July). | EDDA, LR, kNN | Log data | Withdrawal |  0.93 |
| Haiyang, L., Wang, Z., Benachour, P., & Tubman, P. (2018, July) | TSF | Log data | Withdrawal |  0.94 |

Table 1, Legend: ANN: Artificial Neural Network, CART: Classification And Regression Tree, DL: Deep Learning, DT: Decision Tree, DRF: Distributed Random Forest, RF: Random Forest, EDDA: Eigenvalue Decomposition Discriminant Analysis, GBM: Gradient Boosting Machine, GBT: Gradient Boosting Tree, GLM: Generalized Learning Model, JRIP: JRIP Decision Rules, J48: J48 Decision Tree, kNN: k-Nearest Neighbours,  LR: Logistic Regression, LSTM: Long Short Term Memory, NBC: Naïve Bayes Classifier, SVM: Support Vector Machine, TSF: Time Series Forest , XGB: Extreme Gradient Boosting.

study are summarized. Table 2 provides an overview of the existing studies, which algorithms they used and their input variables, their output variables, and their highest accuracy.

Rizvi, S., Rienties, B., & Khoja, S. A. (2019) only considered demographics as the input variable for their model. They showed that neighborhood poverty level and prior education are the most important predictors within the demographic dataset.

Heuer, H., & Breiter, A. (2018) used three different representations of the log data for their model. A binarized representation (i.e. whether the student was active on the VLE that day), a normalized representation and the total number of clicks that day. Interestingly, they found that the binarized representation of the log data yielded the best performance. Furthermore, they showed that demographics have very little influence on the performance of the model in combination with the log data.
Jha, N., Ghergulescu, I., & Moldovan, A. N. (2019) have shown also that demographics have little influence on the performance. Their models are based on all attributes (i.e., demographic, assessments and log data), only achieved about 0.01 higher than models based on log data only.

Hassan, S. U., Waheed, H., Aljohani, N. R., Ali, M., Ventura, S., & Herrera, F. (2019) have divided the log data into time intervals and applied the advanced Long Short Term Memory algorithm to predict (early) drop-outs. At week 25 the model predicted the drop-out rate with 97% accuracy, solely based on clickstream data. The model had an accuracy of around 84% in the first ten weeks. They didn't issue demographics or assessments as an input variable.

Hlosta, M., Zdrahal, Z., & Zendulka, J. (2017, March) have shown that the submission of the first assessment of a module is a very strong predictor of whether a student will pass or fail the module.

Alshabandar, R., Hussain, A., Keight, R., Laws, A., & Baker, T. (2018, July) investigated the influence of important contextual components, like assignment deadlines on student withdrawal. The features used in the research are log data organized in six different time intervals, corresponding to assignment submission dates. They only considered data from a single course module.

Haiyang, L., Wang, Z., Benachour, P., & Tubman, P. (2018, July) showed a time series based approach. They separated the dataset in smaller datasets based on a single course module. As the input variables, they considered log activity on three different learning environments: “forum”, “oucontent” and “resource”. They trained their model with the time series forest algorithm for each dataset. They processed an accuracy between 0.74% and 0.94% for each dataset. On the module called:  “AAA2013J” they reached an accuracy of 0.84% with only 5% of the data processed.

Research gap

Considerable research has been conducted to predict withdrawals in the OULAD. However, a research gap still exists, previous work has a high tendency to overfit on new and real-world data. Alshabandar, R., Hussain, A., Keight, R., Laws, A., & Baker, T. (2018, July) & Haiyang, L., Wang, Z., Benachour, P., & Tubman, P. (2018) narrowed their dataset down to where a model is trained on a single course module. The model will perform well on all the data from that course but if used to predict withdrawals from a new course it will perform poorly.  Hassan, S. U., Waheed, H., Aljohani, N. R., Ali, M., Ventura, S., & Herrera, F. (2019) dropped all the fail instances of the dataset their model was trained on. This makes the data very heterogeneous since fail instances generally have similar data values as the withdrawal data; low engagement and low assessments. This results in a low accuracy on new (real-world) data where fail instances are added.

Secondly, previous work didn't consider that in the real-world withdrawal data is dynamic, this means the dataset becomes smaller overtime when students drop out in real-time. In the OULAD a total of 10,156 out of 32,593 students are classified as withdrawn in the dataset, but of these 10,156 students, 3,097 withdrew before the start of the course. The data of all these students have zero log data and zero assessment results, this homogenous data is easy to classify for a predictive algorithm. The model is trained on predicting something that already happened, which results in poor performance of real-world data. This study complements previous work by predicting dropouts on a non-specified dataset, all variables and all different modules are kept in the dataset, and real-world dataset, where students will be removed from the dataset after they unregistered from a course.

**Method**

In this section, I'll go in-depth on what the OULAD consists of, how it has been pre-processed and what machine learning algorithm is used to conduct the experiment.

OULAD

In this study, the dataset is collected from the Open University, UK. The OULA dataset contains data from seven different course modules given between the years 2013 and 2014. It has three main predictors: student demographics, current study results, and log data. The dataset consists of multiple tables that are connected. The main table is the student information table, this table contains all the demographic data of the student. The features of this demographic data are gender, imd-band (poverty level of the region), age-band, region, disability, studied credits. The student information table is linked to the student assessment table and the student log data table, the other two predictors in our model.

The log data contains the information of every click of every student in a certain virtual learning environment and it gives information on which date this click has been made. The assessment data contains all the grades students received for their assessment and the date the assessment took place.

Data preparation

The data preparation and model training are done with R in RStudios. In my model I chose to simplify the log and assessment data. I processed the three main predictors as follows:

* The log data is processed as all the cumulative clicks a student had made in four weeks disregarding wherein the learning environment the clicks were made. One variable is all the clicks a student before the start of the course made, named week\_<0. The rest of the variables are called week\_0-4 up until week\_36-40 with steps of 4 weeks.
* The student assessment data was processed into two features. One was the average grade of all the assessments a student received, disregarding its weight, named unweighted\_average. The other one was the cumulative grade multiplied by its weight factor, named weighted\_average.
* In the demographic data all variables are kept but one. Because the IMD-Band is only a small portion of the demographic data and it had missing values it is regarded as negligible and therefore dropped from the dataset.

The data was split into eleven different time periods. The first time period consisted of the data that was available before the start of the course, this is all the demographic data and the log activity before the start of the course module. From there the next time period is the data that is available after the first four weeks of the course, this includes new log data and a new unweighted\_average grade and weighted\_average. This continues with steps of four weeks, where every time log data is added and the unweighted\_average grade and weighted\_average are updated with new grades they received in those four weeks.

I’m considering two different types of datasets one dataset is the non-dynamic dataset where the drop-out instances are not removed from the dataset after they drop-out from a course. The second dataset is the dynamic dataset where the drop-out instances are removed. Both datasets are split into 11 different time intervals.

The feature that is predicted, the dependent variable, is whether a student withdraws from the course or not. Initially, the final\_result variable of the Student Information table is categorized into four different values: Withdrawn, Pass, Distinction and Fail. The latter three are renamed as “Not Withdrawn”.

Experiment

The contribution of this research is twofold:

* To predict whether a student will drop out from the course
* Evaluate the most important predictors of each dataset interval

The XGBoost algorithm is used to predict student dropout and to evaluate the most important predictors. The algorithm performs well on large datasets with both continuous and numerical features. It recently gained a lot of popularity during the Kaggle competition[[1]](#footnote-1). In the KDD-CUP15, a competition where you have to predict student dropout, the top ten best solutions used XGBoost as their algorithm[[2]](#footnote-2). XGBoost performs well because it is an ensemble learning method, it combines multiple ‘weak’ learners into a stronger learner to offer a solution. The algorithm is an advanced version of decision trees. Since the goal of this paper isn't a comprehensive evaluation of the machine learning system, I will only shortly explain the two concepts of bagging and boosting on which the algorithm is based. Bagging (from **bo**otstrap **agg**regat**ing**), improves stability and accuracy, reduces variance and helps to avoid overfitting. The technique works as follows: Consider the training data set *D* with *n* elements. Bagging generates randomly with replacement *m* subsets (or bags) of *D*, each with *n’* elements. Random with replacement means the instances are randomly picked and there could be duplicates in the subsets. Each subset is trained, and their mean output is the output of the algorithm.
Boosting is used to reduce bias and variance. It is based on the principle that all the trees are built sequentially and that each tree learns from its predecessors to minimize the residual errors: $F\_{m+1}\left(x\right)=F\_{m}\left(x\right)+h(x)$, where $F\_{m}$ is an imperfect model and *h* is an estimator fitted to the residual of $F(m)$. The general idea is that the instances which are hard to predict, will be focused on in successor trees to minimalize residuals.

In this study caret implementation of XGBoost is implemented. In the experiment, I used four-fold cross-validation and a tunegrid for the algorithm to find the optimal results for each of the 11 different datasets. The following parameters were chosen in the tunegrid:

* Eta, step size shrinkage, A low value makes the process conservative: ( 0.05, 0.1 )
* Nrounds, the amount of trees that are built: (50, 100).
* Max\_depth, the maximum depth of a tree: (4,5,6)
* Min\_chid weight, minimum sum of instance weight: 2.
* Colsample\_bytree, specifies ratio of the chosen subsample of columns : 0.5.
* Gamma, minimum loss reduction: 0.
* Subsample, subsample ratio of the training instances: 1.

To evaluate the most important predictors we use the build-in varImp() function from caret. The function permutes each predictor variable. The difference between the two accuracies is then averaged over all trees and normalized by the standard error.[[3]](#footnote-3)

Performance evaluation

For the total 32,592 students of the non-dynamic dataset 22,437 did not withdraw from the course. If we have an algorithm that classifies every single student as 'did not withdraw', we would achieve an accuracy of 0.69. This accuracy is our threshold. To evaluate the performance I'm going to measure the following metrics: Accuracy, Recall, Precision, and F1.

In the dynamic dataset, the accuracy threshold is dynamic and changes over each dataset interval, the different interval thresholds are presented in the graph. The same metrics are measured.

**Results**

This section evaluates the two experiments I conducted. In the first experiment, I test the performance of the model on the 11 non-dynamic dataset intervals. The results are shown in Figure 2.

The performance of the algorithm increases when the log and the assessment scores become richer. The biggest improvement in prediction performance is received in the data from the first four weeks of the course. Another notable result is the recall decreases from week 4 onward. This means the amount of classified false negatives keeps getting smaller over time. From week 32 the recall has the same value as the accuracy, the algorithm is not classifying any student as a withdraw when they did not withdraw.

The parameter grid and the top 3 most important variables of each interval are shown in table 2. The most important findings in the variable importance are that the numerical features, log data, and assessment data, have the most predicting power. Secondly, the log data of the most recent 8 weeks of each interval have the biggest contribution to the accuracy. The weighted average comes as third. The log data after week 32 becomes less relevant in the largest intervals. This could be because some courses have already finished and therefore have no predictive power.

In the second experiment performance of the model on the dynamic dataset is tested. The results are shown in Figure 2.

The most remarkable result is that the prediction accuracy is very poor. On all intervals, except the week\_0-4 interval, does the accuracy not beat the threshold. The recall performance is close to 1 and the precision performance is almost equal to the accuracy, this means the algorithm classifies every instance as did not withdraw. In other words, the algorithm couldn’t learn from any deviating

withdrawal data and be able to classify drop-outs. A possible reason for this is that students who failed course but didn’t withdraw are processed as ‘did not withdraw’ in the dataset. These fail instances probably caused a lot of false negatives. Failing and withdrawing students generally have low grades and low activity, so their log and assessment data would look-alike and would therefore be difficult to classify.

Only the interval week\_0-4 barely beats the threshold, the threshold is 0.81 and the accuracy is 0.83. The most important variables in the week\_0-4 interval. These are: Weighted\_Average with 0.43, Unweighted\_Average with 0.2 and studied\_credits with 0.1. The other intervals don’t beat the accuracy threshold, because of this variable importance can’t be measured.

Every specific combination of the tune grid parameters on any given interval resulted in either no difference of accuracy or the difference in accuracy was smaller than 0.002 and therefore negligible.



Figure 2, Legend: Dark green: recall, Purple: precision, Red: F1, Blue: accuracy

Figure 3, Legend: Dark Green: recall, Purple: precision, Red: F1, Blue: accuracy, Black: baseline 

|  |  |  |
| --- | --- | --- |
| Interval | Tunegrid | Variable importance |
| Eta | Max depth | Nrounds | Top 1variable | Value | Top 2Variable  | Value | Top 3variable | Value |
| Week\_0 | 0.1 | 4 | 100 | Week\_<0 | 0.46 | Studied credits | 0.31 | Highest education lower than A | 0.04 |
| Week\_0-4 | 0.1 |  5 | 50 | Week\_0-4 | 0.33 | Weighted average | 0.23 | Studied credits | 0.15 |
| Week\_0-8 | 0.1 | 6 | 100 | Week\_4-8 | 0.28 | Week\_0-4 | 0.22 | Weighted average | 0.18 |
| Week\_0-12 | 0.1 | 5 | 100 | Week\_8-12 | 0.29 | Week\_4-8 | 0.22 | Weighted average | 0.17 |
| Week\_0-16 | 0.1 | 5 | 100 | Week\_12-6 | 0.32 | Week\_8-12 | 0.19 | Weighted average | 0.18 |
| Week\_0-20 | 0.1 | 5 | 100 | Week\_16-20 | 0.46 | Week\_12-6 | 0.13 | Weighted average | 0.12 |
| Week\_0-24 | 0.1 | 6 | 50 | Week\_20-24 | 0.31 | Week\_16-20 | 0.21 | Weighted average | 0.16 |
| Week\_0-28 | 0.1 | 5 | 100 | Week\_24-28 | 0.35 | Week\_20-24 | 0.31 | Weighted average | 0.11 |
| Week\_0-32 | 0.1 | 4 | 100 | Week\_28-32 | 0.65 | Weighted Average | 0.07 | Week\_24-28 | 0.05 |
| Week\_0-36 | 0.1 | 5 | 50 | Week\_28-32 | 0.35 | Week\_32-36 | 0.17 | Week\_24-28 | 0.11 |
| Week\_0-40 | 0.1 | 4 | 100 | Week\_28-32 | 0.34 | Week\_32-36 | 0.21 | Week\_24-28 | 0.11 |

Table 2, variable importance

**Discussion**

The student dropout rate in distance learning and specifically MOOCs, is widely researched. This research investigated the performance of XGBoost and the most important predicting variables in the OULA dataset to contribute to the development of an early warning system. We processed students’ engagement and performance using the OULAD’s VLE interaction and assessment table, respectively. The data is split into eleven intervals, the data contains demographic, log and assessment data. One series of intervals is dynamic, here the withdrawal instances are removed after their unregistration, for the non-dynamic series this doesn’t apply. Experimental results show that the non-dynamic dataset achieves a good performance with accuracies between 0.76 and 0.86. This accuracy is a bit lower compared to previous studies, where most achieve an accuracy of over 0.9. But if you take into consideration the model has learned from al the seven different courses together instead of just a singular course it performs quite well. The most important variables are the most recent log data variables and the weighted average of the assessment scores. The XGBoost algorithm can positively classify student dropouts even on a diverse dataset consisting of seven different courses. Future work could investigate the algorithm with different parameters and new features based on the more important variables, that better represent the underlying problem, to achieve an even higher accuracy.

The non-dynamic dataset, on the other hand, can’t beat the performance threshold. This is possibly since in the ‘did not withdraw’ partition of the data, the instances that failed the course are quite similar of nature with the drop-out data, both have low assessment scores and low log data values. This could be analyzed in future research and if this claim is true, a model could be trained to classify between the dropout and fail instances together against the pass instances.

**Literature**

Allen, I. E., & Seaman, J. (2011). *Going the distance: Online education in the United States, 2011*. Sloan Consortium. PO Box 1238, Newburyport, MA 01950.

Alshabandar, R., Hussain, A., Keight, R., Laws, A., & Baker, T. (2018, July). The application of gaussian mixture models for the identification of at-risk learners in massive open online courses. In *2018 IEEE Congress on Evolutionary Computation (CEC)* (pp. 1-8). IEEE.

Haiyang, L., Wang, Z., Benachour, P., & Tubman, P. (2018, July). A Time Series Classification Method for Behaviour-Based Dropout Prediction. In *2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT)* (pp. 191-195). IEEE.

Hassan, S. U., Waheed, H., Aljohani, N. R., Ali, M., Ventura, S., & Herrera, F. (2019). Virtual learning environment to predict withdrawal by leveraging deep learning. *International Journal of Intelligent Systems*, *34*(8), 1935-1952.

Heuer, H., & Breiter, A. (2018). Student Success Prediction and the Trade-Off between Big Data and Data Minimization. *DeLFI 2018-Die 16. E-Learning Fachtagung Informatik*

Hussain, M., Zhu, W., Zhang, W., & Abidi, S. M. R. (2018). Student Engagement Predictions in an e-Learning System and Their Impact on Student Course Assessment Scores. *Computational intelligence and neuroscience*, *2018*.

Jha, N., Ghergulescu, I., & Moldovan, A. N. (2019). OULAD MOOC Dropout and Result Prediction using Ensemble, Deep Learning and Regression Techniques.

Kuzilek, J., Hlosta, M., & Zdrahal, Z. (2017). Open university learning analytics dataset. *Scientific data*, *4*, 170171.

Piech, C., Huang, J., Chen, Z., Do, C., Ng, A., & Koller, D. (2013). Tuned models of peer assessment in MOOCs. *arXiv preprint arXiv:1307.2579*.

Rizvi, S., Rienties, B., & Khoja, S. A. (2019). The role of demographics in online learning; A decision tree based approach. *Computers & Education*, *137*, 32-47.

Shah, D. (2018) *Class Central: By The Numbers: MOOCs in 2018.* https://www.classcentral.com/report/mooc-stats-2018/

Simpson, O. (2010). 22%-can we do better?. *The CWP Retention Literature Review*, *47*.

1. www.kaggle.com [↑](#footnote-ref-1)
2. https://www.linkedin.com/pulse/present-future-kdd-cup-competition-outsiders-ron-bekkerman [↑](#footnote-ref-2)
3. https://cran.r-project.org/web/packages/caret/vignettes/caret.html [↑](#footnote-ref-3)