Selecting the Best Approach for Predicting Student Dropout in Full Online Private Higher Education

José Manuel Porras [1], Antonio Porras [1], José Alberto Fernández [2], Cristóbal Romero [1] and Sebastián Ventura [1]

[1] *University of Cordoba, Department of Computer Science, Cordoba, Spain*
[2] *Nucleoo, 21 Gran Via de Colón, Granada, Spain*

Abstract

This paper describes a project carried out between the University and a course provider company, where an early dropout prediction system has been developed in fully online private higher education. The aim is to be able to predict students at risk of dropping out as soon as possible in order to help them and put them back on success track again. A classical Cross-Industry Standard Process for Data Mining development methodology has been employed over anonymized and unbalanced data from 16,673 students enrolled in 517 fully online courses. The project objective is to determine both the optimal approach for grouping the data and to identify the most accurate classification algorithm and balancing method for predicting dropout. The experiments conducted showed that grouping the data into cumulative periods/quartiles and utilizing the XGBoost machine learning algorithm with SMOTE data balancing method obtained the best results with the highest AUC and True Positive prediction values. However, the approach of grouping the data on a weekly basis and employing the LSTM deep learning algorithm with weights obtained the highest values in F-1 measure and True Negative indicator. The latter approach was finally the selected one for being implemented in the early prediction system.

Keywords

Dropout prediction, full online private learning, deep learning, explainable artificial intelligence

1. Introduction

Nowadays, the attendance of official, private and free full online courses has increased enormously since the COVID-19 pandemic [1]. However, one of the main problems with these types of courses is their high dropout rate. It occurs when there is a high number of students who start courses but then they do not complete them [2]. It is important to note that the problem of detecting students dropping out of online courses affects both private companies and all public educational institutions offering such courses, which makes it a problem of great scope and impact. One way to help solve this problem is by using data mining and early warning detection systems [3]. The aim of such systems is to be able to detect students who are at risk of dropping out as early as possible so that intervention actions can be taken to try to prevent them from dropping out [4]. This paper describes the development of an early dropout prediction system for full online private higher education students. Starting from students' interaction information with courses (labelled as either successful completers or dropouts) a dropout risk prediction model was generated to predict students enrolled in new courses [5]. The final objective is to be able to use the prediction obtained and then call by phone the student detected as at high risk of drop out. The course provider company has a Call Center Service for supporting students registered in their courses. They intend to phone students for asking them about what problems they may find for continuing with the course. The objective is to increase the retention rate in their online courses.

Learning Analytics Summer Institute Spain (LASI Spain) 2023, June 29–30, 2023, Madrid, Spain

EMAIL: i82poroj@uco.es (A. 1); i12popea@uco.es (A. 2); josealberto.fernandez@nucleoo.com (A. 3); cromero@uco.es (A. 4); sventura@uco.es (A. 5)

2. Background

Predicting student dropout in online learning is an important and widely studied educational problem [3]. Educational Data Mining (EDM) techniques have been successfully applied [4] to solve this problem as a binary classification task (0 or 1). In this task, there are a set of training data already labelled (students who have already completed the course) and therefore it is known whether they have dropped out (labelled with the label or class 1) or whether they have successfully completed the course (labelled with class 0). The aim of the task is to predict the label or class of new students taking a new course as early as possible.

Traditionally, a wide range of classical machine learning and data mining algorithms have been used to solve this classification problem [3] [5], such as: Decision trees, Bayesian networks, Support Vector Machines and Neural networks. But in recent years, new and more powerful classification algorithms have appeared, such as advanced ensembles and Deep Learning algorithms. They have obtained significantly superior predictive capabilities compared to classical algorithms [6]. Nevertheless, these advanced learning algorithms have the drawback of generating black box models. They do not generate a prediction model that is easily interpretable by humans, unlike white-box models such as decision trees or rule-based models. To address this issue, Explainable Artificial Intelligence or XAI (eXplainable Artificial Intelligence) techniques have emerged. Some examples are [7] permutation feature importance, SHAP (SHapley additive explanations), or LIME (Local Interpretable Model-agnostic Explanations). Their aim is to enhance the interpretability of machine learning models and enables humans to comprehend the models and use them in educational decision-making processes.

Another important issue when dealing with student dropout data is the class imbalance problem. It happens when there is an unequal distribution between minority and majority classes (i.e., the number of dropouts and completers). Two different approaches have been traditionally employed in the bibliography to address this problem [8]: (i) cost-sensitive learning, which assigns a higher cost or weight to misclassifying the minority class; and (ii) sampling or balancing techniques, which involve creating a dataset to achieve a more balanced class distribution. One widely used algorithm for this last purpose is SMOTE (Synthetic Minority Oversampling Technique), which generates synthetic data points based on k-nearest neighbors method [9].

3. Methodology

In this project, the CRISP-DM (Cross Industry Standard Process for Data Mining) methodology has been used as the basis for a data science process [10] and, in our case, consisted of the following stages:

- **Business and Data understanding**: In our case, all the anonymized data has been provided by an online course provider company about students who have enrolled at least in one course and whether they have finished or dropped out. Dropout students are those who, after enrolling in a course, choose not to continue in the course and cease their payment commitment. This information was stored in 4 tables: Courses (course information), Enrollments (student enrolment information in courses), PeriodActivity (information about students' activity by periods) and WeeklyActivity (information about students' weekly activity in courses).
- **Data preparation**: Firstly, we have selected the best attributes for predicting student dropout starting from the all the variables available in the 4 previous tables for creating a summarization dataset. Then, we have preprocessed data by cleaning (null values) and normalizing of all the attributes’ values to the same 0-1 scale. Finally, a label 1 (Dropout) or 0 (No dropout) was added to each student as class to predict.
- **Modeling and Evaluation**: In this stage, the best data grouping approach, balancing method and classification algorithm were determined. To do this, the pre-processed data were grouped in different ways and balanced before being used for training different classification algorithms in order to find which ones obtained the best results in predicting dropout.
- **Deployment**: Finally, the best grouping approach and selected prediction model were applied for real-time prediction of the label or class of new students. New students are those who are

currently taking a course and for whom we intend to predict whether or not they are at risk of dropping out. The Call Center Service will call only to student predicted at high risk of dropout.

3.1. Data

The anonymized data used in this project came from 16,673 students who attended 517 online courses in Moodle Learning Management Systems. Of this group, 10,650 students successfully completed their courses (92.11%), while 912 students, who had been active in their courses at some point, dropped out before completing the course (7.89%). Therefore, the distribution of classes (dropouts and non-dropouts) was unbalanced among the students who participated in the study. Starting from four database tables (Courses, Enrolments, PeriodActivity and WeeklyActivity), we have selected 14 attributes or variables provided by the course company provider as a summarization dataset (see Table 1). We have only selected 3 attributes related to the course that we think that provide us useful information for predicting dropout out. And we have obtained 11 traditional summarization attributes [4] related to the students' activity or interaction with the Learning Management System (LMS).

Table 1
Description of used attributes

Attribute Name	Description
course_type	Type of course
international	Whether the course is international or not
price	Course fees
com_tutor	Number of communications students with tutor
htm_completed	Number of HTML pages completed
vid_completed	Number of completed video views
exa_completed	Number of examinations completed
autoeva_completed	Number of self-assessments completed
n_posts	Number of posts written
n_discussions	Number of discussions the student has participated
assignment_submited	Number of tasks performed
discussions_viewed	Number of discussions the student has seen
course_visits	Number of visits to the course
hours_in_course	Number of hours of activity in the course

Then, all the values of these attributes have been preprocessed as follows:

- **Data cleaning**: If a null or unknown value is found, it is replaced by the value of 0.
- **Data normalisation/scaling**: For course_type (the only categorical value) we used a label encoder to convert this attribute into a numerical one. Then, we rescaled the values of all the attributes to the range [0-1] using the standard min-max scaler normalization.

3.2. Approaches for grouping data

In this project, we have proposed three different classical approaches of grouping the students' interaction data in order to determine the most useful to solve our problem:

- **Independent periods/quartiles.** This approach consists of grouping the data of the student's interaction with the courses in four specific periods or moments of time (25%, 50%, 75% and 100% of the duration of each course). So, four tabular data files are generated (see Figure 1). In these files, the columns represent the attributes, and the rows represent the students. That is, the interaction information of a student during each time period is collected in a single row of the dataset together with its label or class value at the end (1 or 0). Thus, 4 individual datasets are obtained, one for each time period from which a different prediction model will be generated.

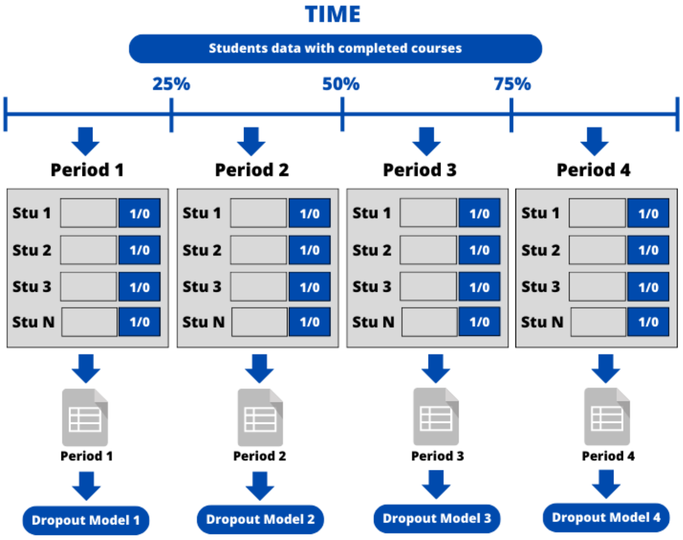

Figure 1: Approximation of independent periods

- **Cumulative periods/quartiles.** This approach is very similar to the previous one, with the only difference that the student interaction data are accumulated between periods. That is, in each new period the values of the current attributes are accumulated or added to the values of the preceding periods. As in the independent approach, 4 tabular datasets are also generated and so, dropout can be predicted at four points in time using the corresponding prediction models.

- **Weeks.** In this approach the student interaction data are grouped sequentially by weeks. So, for each student there is a sequence of weeks or rows in the data file where their interaction activity with the course is stored. The length of the sequence (number of rows) for each student is variable depending on the number of weeks where each student has interacted with the course (see Figure 2). It is also important to note that a student can start and drop out a course at any moment or week of a course. Finally, a single sequential dataset and a single prediction model are obtained.

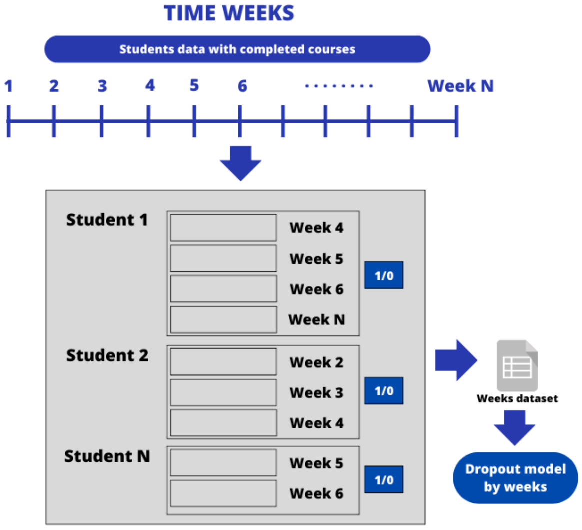

Figure 2: Approximation by weeks

4. Experimentation and Results

In the experimentation carried out, two different types of classification algorithms have been used. On the one hand, we used both classical Machine Learning (ML) and a more advanced ensemble algorithms for the two approaches of grouping data by periods. In this case the information for each student comes in a single row of data in a tabular dataset. The specific classical ML algorithms used are [11]:

- **Decision Tree (DT):** It has internal nodes that divide the data into smaller subsets based on rules and a series of leaf nodes that assign a class to this data.
- **Neural Network (NN):** It is a multilayer neural network where each neuron processes its inputs and transmits its output value to the neurons in the next layer. Each of these connections between neurons has a weight assigned during the training and the neurons in the output layer provide the class value.
- **KNearest Neighbours (KNN):** It is a ranking method based on the nearest neighbours method that returns the majority vote of its k nearest or similar data points.
- **Random Forest (RF):** It is a combination of a group of predictor trees where the classification is done by the most voted class among all of them.

The advanced ensemble algorithm is the popular Kaggle winner algorithm:

- **XgBoost (XgB):** It is a more advanced ensemble algorithm composed of a set of decision trees that improves predictive accuracy through incremental error minimization.

On the other hand, we have used deep learning classification algorithms for the approach of grouping data by weeks, where the information of each student comes in several rows of data in a sequential dataset. The specific deep learning algorithms used in this work are [12]:

- **LSTM (Long Short-Term Memory):** It is a recurrent neural network with memory, which has the ability to maintain and update long-term information.
- **biLSTM**: It is a variant of LSTM that processes sequences in both end-to-end and end-to-beginning directions to better capture long-term dependencies in sequential data.
- **CNN1D**: It is a convolutional neural network that applies and slide a one-dimension filter over the time series. The filter can also be seen as a generic non-linear transformation of a time series.
- **Cnn1D-LSTM**: It is a combination of a one-dimensional convolutional layer for extracting main features, followed by a LSTM layer.
- **Cnn2D-LSTM**: It is a combination of a two-dimensional convolutional layer for extracting main features, followed by a LSTM layer.

All the above algorithms were implemented in Phyton language using the libraries: pandas, numpy, scickit-learn, xgboost, tensorflow and keras; and their default parameters. To compare their classification performance, a 5-fold Cross-Validation has been performed over the datasets of the 3 proposed approaches. Cross-validation divides the dataset into 5 subsets of similar size. The model is then trained on 4 of the subsets and evaluated on the remaining subset. This process is repeated 5 times, so that each subset is used as a validation set exactly once. As evaluation metrics, the traditional Accuracy measure was not used as it is not a suitable metric to evaluate unbalanced datasets. Instead, we have used the four low level evaluation performance metrics provided by the confusion matrix (True Positives, True Negatives, False Positives and False Negatives) and two well-known general metrics traditionally used with unbalance datasets [13]:

- **AUC:** It is the Area Under the ROC curve, which it is one of the most important metrics used to represent the expected performance of a classifier.
- **F-measure** or **F-score**: It is the harmonic mean of the values of the precision and recall measures, and it is a popular metric for unbalanced datasets. There are several versions of F-measure depending on what is more important in our problem. In our case, as both False Negatives and False Positives are equally important, then we used F1-Score. In our problem, our main objective is to correctly predict the student dropout (to increase TP) but without failing too much in predicting TN (student who finished correctly the course without

dropping out) because there is also a cost associated to this fail (in money necessary to call too many students that will not actually drop out).

4.1. Experiment 1

In the first experiment, classical ML and ensemble algorithms have been applied over the four datasets of the two period-based approaches. SMOTE algorithm [9] has been previously applied over the dataset for solving the problem of class unbalance. In both approaches, the XgBoost algorithm always obtained the best results for all evaluation measures with much higher values than the other algorithms. As an example, Table 2 shows the results (% of evaluation metrics) obtained by all classification algorithms when applied over the period 1 dataset. In this specific case, the first period is exactly the same dataset for both independent and accumulated period approaches.

Table 2
Results of classical ML and ensemble algorithms with period/quartile 1

Metrics (%)	DT	RF	NN	XgB	KNN
AUC	69.77	72.58	75.94	**86.51**	66.31
F1	70.97	72.87	74.37	**86.49**	69.3
TP	71	83	82	**90**	74
TN	60	51	57	**83**	53
FP	40	49	43	**17**	47
FN	29	17	18	**10**	26

As it can be seen in Table 2, the AUC and F1 values for the XGboost algorithm are about 86%, that is a large improvement of more than 10% with respect to the rest of the algorithms. Regarding the values of the confusion matrix, the highest value is obtained with the TP metric (% of students who drop out and are correctly classified). It is exactly what we want to predict in our problem, with a 90% success rate. And for the students who do not drop out, the percentage of correctly classified TN is 83%. It should be noted that, for the FP and FN measures, the best results correspond with the lowest values, as these are the percentages of incorrectly classified students.

All the results obtained by the XGBoost algorithm with SMOTE applied to each one of the 4 datasets of the independent and cumulative period approximations are shown below (see Table 3 and Table 4, respectively).

Table 3
XgBoost results with the independent periods approach

Metrics (%)	Quartil 1 (25%)	Quartil 2 (50%)	Quartil 3 (75%)	Quartil 4 (100%)
AUC	**86.51**	83.92	82.88	81.41
F1-score	**86.49**	83.86	82.79	81.31
TP	**90**	90	90	89
TN	**83**	78	76	74
FP	**17**	22	24	26
FN	**10**	**10**	**10**	11

As it can be seen in Table 3, the best evaluation values with the independent period approach are obtained in the first period. One possible reason for the better prediction in the first period may be due to the fact that the highest number of students' dropouts occurs at the beginning of the course (in the first period).

Table 4
XgBoost results with the cumulative period approach

Metrics (%)	Quartil 1 (25%)	Quartil 2 (50%)	Quartil 3 (75%)	Quartil 4 (100%)
AUC	86.51	87.84	88.92	**90.54**
F1-score	86.49	87.82	88.90	**90.52**
TP	90	92	93	**95**
TN	83	84	85	**87**
FP	17	16	15	**13**
FN	10	8	6	**5**

In Table 4, we can see how the XGBoost algorithm obtains better results as the periods progress and there is more available accumulated information of the student interaction with the course. It is precisely in the last period 4 where the highest values are obtained, with 90% in AUC and F1, and TP values of 95% and TN 87%, which are very high percentages indicating a very good prediction. In fact, the TP values (students who drop out and are correctly classified) are the highest values obtained in all the experiments.

Finally, we want to obtain some comprehensible information about the black-box prediction model generated by the XGBoost algorithm. In order to do it, we have used a XAI technique called permutation feature importance [7] to discover the attributes that most influence in the classification. This technique measures the decrease in the score of a predictor model when a single attribute value is randomly mixed. Figure 3 shows the most important or influential attributes in the XGBoost ranking with the period 4 dataset of cumulative periods.

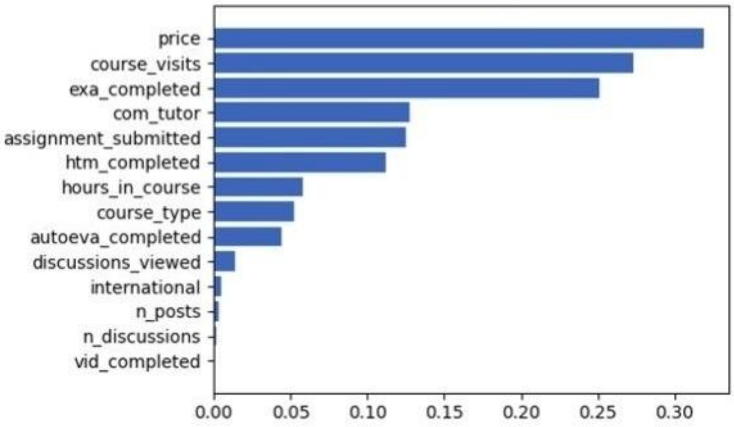

Figure 3: Importance of the characteristics of the Xgboost prediction model

It can be seen in Figure 4 that the price of the course is the most influence attribute. Then, other top attributes are well-known features about the interaction and engagement of the students with online courses such a: the number of visits to the course, the number of exams completed, the number of assignments submitted, and the number of html completed. The number of times contacted with tutor is another top attribute, but this is a feature directly related with this type of private online courses. In this line, we think that one possible explanation about why the price is the most influence attribute can be due to the courses not being free. So, may be that there is an inverse relationship between price and dropout, that is, the higher the price of the course, the less likely the students are to drop out.

4.2. Experiment 2

In the second experiment, different deep learning algorithms have been applied over the single dataset of the approach of grouping data by weeks. We set class weights to solve the problem of class imbalance since there is no version of SMOTE algorithm for sequential data. Specifically, we have set a higher weight to the minority class by using the compute_class_weight function of sckikit-learn library. Table 5 shows the results obtained by the deep learning algorithms.

Table 5
Results of the deep learning algorithms for the weekly approach

Metrics (%)	LSTM	BiLSTM	CNN1D	CNN2D	CNN1d LSTM	CNN2d LSTM
AUC	**89.93**	89.47	89.43	88.89	85.39	83.95
F1	**93.72**	92.67	92.88	92.66	91.86	90.47
TP	**87**	**87**	86	86	79	78
TN	**94**	92	92	92	92	90
FP	**6**	8	8	8	8	10
FN	**13**	**13**	14	14	21	22

As it can be seen in Table 5, all the algorithms obtained very similar results in all metrics. The best results were obtained by the LSTM neural network, with 90% AUC and 93% F1 which is the highest values obtained in all the experiments. With respect to the values of the confusion matrix, the percentage of TP or students who drop out and are correctly classified is 87%, which is a good value. And the percentage of TN or students who finish and are correctly classified is 94%, which is the highest value achieved in all the experiments.

Finally, we depict the permutation features importance for showing some comprehensible information about the black-box prediction model obtained by the LSTM algorithm. Figure 5 shows the ranking of the most important attributes obtained by LSTM prediction model.

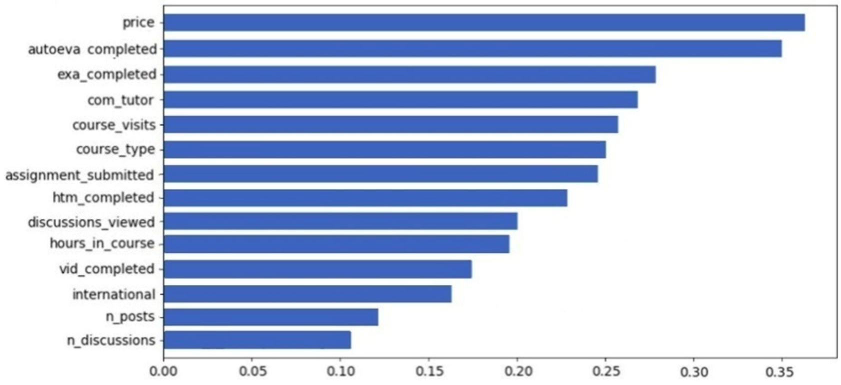

Figure 5: Importance of the characteristics of the LSTM prediction model

As we can see in Figure 5, the price was again the most important feature followed by: the number of auto-evaluations completed, the number of exams completed, the number of times communicated with the tutor and the number of course visits, the type of the course, and the number of assignments submitted.

5. Conclusions and Future Work

The two experiments conducted showed that grouping the data into cumulative periods and using the XGBoost machine learning algorithm with SMOTE data balancing method obtained the best results with the highest prediction values of AUC and TP. However, the approach of grouping the data by weeks and employing the LSTM deep learning algorithm with weights obtained the highest prediction values of F1 and TN and was the selected for implementation in the early prediction system deployed to the course provider company. This decision was motivated by four factors. Firstly, F-1 measure is better than AUC in our problem for unbalanced dataset where False Negatives and False Positives are equally important. Secondly, this approach is easier of maintenance in production due to the fact that it has only one prediction model versus four prediction models of the other approach. Thirdly, it uses only real student data without needing to use synthetic data. And finally, it has the advantage of making predictions earlier, specifically each week, without no need to wait for other longer specific time periods.

In addition, it is crucial to note that our prediction model will provide three different outputs about the classification of student dropout: binary 0 or 1 values, the raw probabilities between 0-1 and three thresholds (high, medium and low). By default, the output of a LSTM prediction model as any neural network is a probability (between 0 and 1), so in order to be able to use it for binary classification (0 or 1) we had to set a threshold, in our case 0 (<0.5) is considered class No Dropout and 1 (>=0.5) is considered class Dropout (in case of sigmoid). We have used these values (0 or 1) in the experiments for obtaining all classification evaluations metrics. However, for using it in real-time for predicting new students we have also defined three intervals: students with dropout rates from 0 to 0.2, from 0.2 to 0.8, and from 0.8 to 1. That is, students with a low dropout rate (lower than 0.2), students with a medium dropout rate (between 0.2 and 0.8), and students with a high dropout rate (higher than 0.8). This way, the Call Center Service can phone only the students predicted as high dropout rate and not to all students predicted as drop out in the binary classification.

Finally, permutation feature importance has been used as XAI technique to show comprehensible information about the black box prediction model obtained by XGBoost and LSTM algorithms. We can see that the main difference between the obtained permutation feature importance in both models (see Figures 4 and 5) is the size of the bars. In Figure 5 all the attributes, even the last ones, provide some importance values to the LSTM prediction model, but in Figure 4 only some attributes provide importance values to the XGBoost prediction model. With respect to the specific attributes, the results obtained showed that in both cases the course price was the most influential attribute, as it is already noticed in other studies [14], as well as the communication with the tutor and some attributes related to engagement and participation in the courses. This may suggest that other alternative variables (such as paid method, motivation for taking the course, etc.) than those that traditionally appear in the literature of online student dropout prediction could also be used in this type of full online private courses. In conclusion, it is very useful to apply XAI techniques to black-box prediction models to make them understandable for later use in educational decision-making processes to prevent students from dropping out. Specifically, this is the main benefit of implementing this type of early prediction model, as it would allow detecting and taking evasive actions to help students at risk of dropping out before it actually happens.

As future work, we intend to carry out more experiments by using much more data from more courses and a higher number of students. We also want to implement other approaches such as meta-modeling and algorithm voting approaches trying to improve the accuracy of the models.

6. Acknowledgements

This work has been funded by the Project CTA-22/1085 entitled Learning Analytics for Improving the Quality of Teaching in Educational Settings, by the Project PID2020-115832GB-I00 project of Spanish Ministry of Science and Innovation and the European Regional Development Fund and the ProyExcel-0069 project of the Andalusian University

7. References

[1] W. Wang, Y. Zhao, Y. J. Wu, M. Goh, Factors of dropout from MOOCs: a bibliometric review. Library Hi Tech, Vol. ahead-of-print, 2022.

[2] B. Prenkaj, P. Velardi, G. Stilo, D. Distante, S. Faralli, A survey of machine learning approaches for student dropout prediction in online courses. ACM Computing Surveys (CSUR), Vol. 53, No. 3, 2020, pp. 1-34.

[3] S. Donoso-Díaz, T. N. Iturrieta, G. D. Traverso, Sistemas de Alerta Temprana para estudiantes en riesgo de abandono de la Educación Superior. Ensaio: Avaliação e Políticas Públicas em Educação, Vol. 26, No. 100, 2018, pp. 944-967.

[4] C. Romero, S. Ventura, Educational data mining and learning analytics: An updated survey. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, Vol. 10, No. 3, 2020.

[5] C. Márquez-Vera, A. Cano, C. Romero, A. Y. M. Noaman, H. Mousa Fardoun, S. Ventura, Early dropout prediction using data mining: a case study with high school students. Expert Systems, Vol. 33, No. 1, 2016, pp. 107-124.

[6] D. A. Shafiq, M. Marjani, R. A. A. Habeeb, D. Asirvatham, Student Retention Using Educational Data Mining and Predictive Analytics: A Systematic Literature Review. IEEE Access, 2022.

[7] E. Melo, I. Silva, D. G. Costa, C. M. Viegas, T. M. Barros, On the Use of eXplainable Artificial Intelligence to Evaluate School Dropout. Education Sciences, Vol. 12, No. 12, 2022, pp. 845.

[8] T.M. Barros, P.A Souza Neto, I. Silva, L.A. Guedes, Predictive Models for Imbalanced Data: A School Dropout Perspective. Education Sciences. Vol. 9, No. 4, 2019, pp. 275.

[9] A. Fernández, S. Garcia, F. Herrera, N. V. Chawla, SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. Journal of artificial intelligence research, Vol. 61, 2018, pp. 863-905.

[10] C. Schröer, F. Kruse, J. M. Gómez, A systematic literature review on applying CRISP-DM process model. Procedia Computer Science, Vol. 181, 2021, pp. 526-534.

[11] Delen, D. A comparative analysis of machine learning techniques for student retention management. Decision Support Systems, Vol. 49, No. 4, 2010, pp. 498-506.

[12] A. Hernández-Blanco, B. Herrera-Flores, D. Tomás, B. Navarro-Colorado, A systematic review of deep learning approaches to educational data mining. Complexity, 2019.

[13] M. Hossin, M. N. Sulaiman, A review on evaluation metrics for data classification evaluations. Journal of data mining & knowledge management process, Vol. 5, No. 2, 2015, pp. 1-11.

[14] J. Martínez-Carrascal, T. Sancho-Vinuesa. Exploring the impact of time between consecutive assessments on course withdrawal: a survival analysis approach. LASI. 2023.