Applied Intelligence (2022) 52:3352–3365
https://doi.org/10.1007/s10489-021-02613-x

Using Meta-Learning to predict student performance in virtual learning environments

Ángel Casado Hidalgo[1] · Pablo Moreno Ger[1] · Luis De La Fuente Valentín[1]

Accepted: 13 June 2021 / Published online: 6 July 2021

Abstract

Educational Data Science has meant an important advancement in the understanding and improvemen of learning models in recent years. One of the most relevant research topics is student performance prediction through click-stream activity in virtual learning environments, which provide abundant information about their behaviour during the course. This work explores the potential of Deep Learning and Meta-Learning in this field, which has thus far been explored very little, so that it can serve as a basis for future studies. We implemented a predictive model which is able to automatically optimise the architecture and hyperparameters of a deep neural network, taking as a use case an educational dataset that contains information from more than 500 students from an online university master's degree. The results show that the performance of the autonomous model was similar to the traditionally designed one, which offers significant benefits in terms of efficiency and scalability. This also opens up interesting areas of research related to Meta-Learning applied to educational Big Data.

Keywords Meta-learning · Deep neural networks · Educational data mining · Learning analytics · Student performance

1 Introduction

Educational Datat Science research has been relevant and profuse in recent years. This is mainly due to the success of online teaching in academic and professional environments, with blended or completely remote methods which make access to education greatly more flexible. At The Open University alone, (a pioneering university in online education) 2 million students from 157 countries have graduated since it was founded, making it the largest academic institution in the UK today [1]. Some of the most prestigious universities, such as the Massachusetts Institute of Technology (MIT), Harvard University, or Stanford University, have launched online educational platforms that offer courses in a massive and totally free way. To date, almost 150 million students have enrolled in their courses since they were created [2, 3].

It is logical to think that an intensive use of these online platforms can generate valuable information to improve teaching quality. Academic records, course syllabuses, learning resources, or the actions carried out by the student on a daily basis on the website serve as a database which can provide meaningful feedback for predictive purposes [4]. For example, taking into account the behaviour of the student throughout the course and their interaction with the available resources, it is possible to define a digital footprint that allows us to study which factors determine their performance to a greater extent and, therefore, are related to student learning. This would help teachers to create courses which offer greater personal enrichment, as well as a more positive evaluation of the educational experience. In addition to this, based on the analysis of this information, educational strategies could be established on a larger scale, which would imply a true paradigm shift in how we perceive learning models [5]. In this sense, Artificial Intelligence becomes an essential tool to explain certain student behaviour patterns in the virtual classroom. The design of predictive models capable of deciphering the intrinsic relationship of the data can significantly help teachers and institutions to face educational challenges. Specifically, machine learning algorithms allow us to

✉ Ángel Casado Hidalgo
ach.casado@gmail.com

Pablo Moreno Ger
pablo.moreno@unir.net

Luis De La Fuente Valentín
luis.delafuente@unir.net

[1] Universidad Internacional de La Rioja, Avda. de la Paz, 137, Logroño, La Rioja, Spain

identify these relationships and determine the student's grade based on them. This means that knowing certain representative information of the student during the course, we can estimate their performance.

Although there is numerous research about student performance prediction, especially in the field of Educational Data Mining [6], there are still interesting areas of research which could be potentially explored.

On the one hand, most of the studies which analyse student results in online courses use classical learning models. These tend to focus on the evaluation and subsequent comparison of supervised learning algorithms, especially using support vector machines (SVM) and decision trees. However, currently there are very few studies that use deep neural networks to build a predictive model, despite its capabilities for handling massive data. This is possibly due to the structured nature of the data and the difficulty in interpretating the results in Deep Learning models [7].

On the other hand, there are new and exciting areas of research that could be applied to this topic, such as Meta-Learning. To the best of our knowledge, this has not yet been used in the disciplines of EDM or LA. This is probably related to the limited use of neural networks in this type of problem. In a nutshell, meta-learning techniques allow the predictive model to learn to learn [8], i.e., instead of carrying out a manual trial and error process in order to define the neural network, meta-learning algorithms provide some autonomy and intelligence to the predictive model for its self-management. This helps to automatically optimise the hyperparameters, find suitable architecture in a search space defined by the specialist [9], or make predictions taking into account only a few examples, in a similar way to how people are able to identify a new object without seeing thousands of images of it previously [10].

This research aims to explore the potential of Deep Learning and Meta-Learning applied to student performance prediction based on their activity in virtual environments. To achieve this goal, we have defined the following specific objectives:

- Identifying and extracting the characteristics that represent the student's interaction with the virtual platform.
- Determining the most relevant characteristics for the prediction of the student's academic performance.
- Designing and implementing a predictive model that learns to optimise itself autonomously.
- Evaluating the performance of the autonomously optimised model compared with a manually optimised model.

To carry out our experiment, we used a dataset provided by the International University of La Rioja: 4 academic courses of a university master's degree extracted from the Sakai educational platform. We selected features related to the student's online activity, and which make up their digital footprint during the course, such as the most visited resources, participation in discussion forums, or lectures visualisation.

The contents of the rest of the work are organised as follows: Section 2 describes the main studies and works related to student performance prediction, as well as a brief introduction to meta-learning and its main applications today. Section 3 introduces the methods, Section 4 the results, and Section 5 the discussion of the experiment. Finally, Section 6 details the conclusions from the study, as well as indicating possible future areas of research on meta-learning applied to Learning Analytics.

2 Related work

Teaching in virtual environments is a major educational and technological challenge. How to create effective online learning environments which are able to meet students' expectations is a field in continuous evolution. Although, such a challenge leads to significant benefits. Mainly based on the Learning Analytics (LA) and Educational Data Mining (EDM) communities, virtual platforms are used as knowledge generators about learning models. On one side, EDM is focused on developing methods which explore the unique types of data that come from learning environments. It can be defined as the application of Data Mining techniques in education, with the aim of responding to relevant questions about the sector [11]. On the other side, LA is defined as the measurement, collection, analysis, and reporting of student data and its context, with the purpose of understanding and optimising the learning experience and the environment in which it happens. There are three important topics in LA: data, analysis, and action.

Obviously, educational Big Data provides abundant information ready to be analysed, using powerful predictive algorithms, among which those based on Deep Learning stand out. The enormous possibilities that Deep Learning provides, and the current state-of-the-art, encourage the research of such models. In fact, its use is quite widespread in Data Mining, and some recent papers apply deep neural networks to educational datasets, such as OULAD (Open University Learning Analytics dataset) [6]. This dataset includes student interaction with the virtual environment, classified into 20 types of events or relevant categories of the platform, for a total of 32,000 students during 9 months of the course [12]. Given the richness and volume of the data offered, this has become one of the reference models for experimentation, especially in the field of EDM and the prediction of academic performance.

In this scenario, research can be approached from different perspectives: early dropout rate, analysis of factors that influence performance, validation of student assessment methodologies, etc. However, much of the current work, especially in EDM, focuses on evaluating the accuracy of different algorithms. Techniques such as SVM, decision trees, random forest, or artificial neural networks, have been widely used to predict the student's grade as a classification problem. In this regard, it is worth mentioning how little adoption is made of Deep Learning techniques, despite its ability to learn more complex relationships in the data than classical algorithms [13]. This makes it to be considered as the most advanced technique to perform classification tasks in Data Mining: pattern extraction in large volumes of data, labeling, semantic indexing, etc. In addition, the use of Deep Learning would open new and interesting avenues of research in rating prediction, such as the use of meta-learning models, discussed in the next section. In a recent research, the authors used a deep neural network to perform predictions on the OULA dataset [6]. For this purpose, more than 50 input variables were extracted, mainly based on the student's activity in the virtual environment, as well as their personal (age, gender, disability), academic (level of studies before entering the course) and sociodemographic (place of residence and its associated level of wealth) information. After applying a selection of the 30 most influential variables, the authors used a network composed of three hidden layers of 50, 20 and 10 nodes to build the model, with which they obtained predictions of over 90%. Another interesting approach in the use of cognitive systems is the use of Recurrent Neural Networks. In another publication [14], the authors applied a model based on recurrent networks on the OULA dataset, also obtaining an accuracy higher than 90%. Such a model, comparatively, achieved better results than logistic regression and artificial neural networks. For the construction of the dataset, the study relied on a temporal sampling strategy: the student's activity in the environment was extracted on a weekly basis, and the model was trained with the accumulated data from each time unit, until a total of 25 weeks was completed. A similar study also employed RNN with data obtained in two MOOC format courses provided by the edX educational platform [15]. However, the authors extracted course information only up to the middle of the semester, with the aim of obtaining a model that was able to predict dropout sufficiently in advance. In the section on simple neural networks (a single hidden layer), promising results have also been obtained. For instance, a prediction was performed using a model composed of a layer of 6 neurons, taking as reference a set of five academic courses in a school of Computer Engineering [16]. The results showed an accuracy close to 80%, and identified the student's memorization capacity and attention level as good predictors of academic performance. However, these types of models have not shown superior performance to other classical algorithms in some experimental cases, such as SVM and decision trees [17].

However, before training the predictive model, it is essential to manage the available data properly. To achieve this, the features or input variables must be analysed to identify those that do not add value to solving the problem. The benefit is twofold: on the one hand, dimensionality is reduced, making the detection of data patterns easier and, on the other, the algorithm efficiency is increased, so it consumes less computational resources. In recent years, several studies have focused on identifying which feature selection techniques can be most useful in predicting academic performance. The authors [18] established a comparison between six different techniques: CFS, Chi-Square, filter feature selection, Gain Ratio, Principal Component analysis and Relief algorithm. Nevertheless, there are more advanced techniques that have not yet been explored in EDM and LA literature, and which may be equally interesting for solving supervised learning problems. Specifically, the ANOVA and RFECV techniques tend to be very effective when dealing with continuous input variables and categorical variables, as in our case. ANOVA (Analysis of Variance) is a statistical method that checks if the mean of two or more groups show a significant difference between them. As for RFECV (Recursive Feature Elimination with Cross-validation), this technique recursively explores the set of features considering smaller groups of variables each time [19]. This is considered an embedded method, since it requires the use of a classifier to evaluate the accuracy obtained with each set of variables.

Other Artificial Intelligence domains, such as Meta-Learning, are being disseminated extraordinarily in the technological field, which is why they represent an important ally in the creation of predictive models and their application for the industry. Meta-learning, also known as learning to learn, is the science of systematically observing how different machine learning approaches work on a wide range of learning tasks, in order to learn new tasks much faster through this experience [8]. That is, just as we are able to learn and develop learning strategies (we learn to learn), machine learning algorithms can also use similar forms of self-management. For example, a predictive model based on neural networks could learn to optimise itself autonomously, without the help of a technical expert who supervises the learning stage by manual comparison of different hyperparameters - also called Hyperparameter Optimisation (HPO). Furthermore, it could search and find the best network topology in a specific search space, avoiding human intervention - Neural Architecture Search (NAS) [20].

HPO can be carried out using several techniques and algorithms, such as Naive Evolution or Tree-structured Parzen Estimator (TPE). In the case of TPE, it performs sequential model-based optimisation (SMBO). SMBO methods sequentially build models in order to approximate hyperparameter performance based on historical measurements, and then it chooses new hyperparameters to test based on that model [21]. Currently, there are numerous solutions which already incorporate hyperparameter optimisation automatically, among many other functionalities. This field, which is also clearly gaining popularity, is known as AutoML (or Automatic Machine Learning).

NAS has become one of the most important research focuses in the area of Meta-Learning. NAS aims to avoid the costly process of finding the right neural network architecture manually [20]. Several algorithms, such as DARTS (Differentiable Architecture Search) or ENAS (Efficient Neural Architecture Search), have achieved an excellent performance, surpassing even those obtained by architecture designed by relevant Deep Learning experts [22, 23]. However, its use is limited to image classification problems. Despite this, NAS advances are leading to new algorithms, such as Auto-Keras, a framework built on top of a tree search algorithm which efficiently explores the search space [24]. Although Auto-Keras has a lower performance when compared to previous algorithms, it can be applied to structured data, which makes it ideal for our case study. As far as it is known by the authors after reviewing the state-of-art, meta-learning has not yet been applied to educational data science. Its use could open new areas of research in EDM and, additionally, help researchers to share methodologies in this and other related disciplines, such as LA. Table 1 shows the main studies related to our use case:

3 Methods

In the experiment carried out, we incorporated advanced Deep Learning techniques into the challenge of grade prediction, with the aim of knowing its applicability to a real use case. In addition, we analysed the limitations which can be found during the feature extraction from the virtual platform, with special emphasis on those which make up the student's digital footprints during the course.

For this, an end-to-end methodology is applied on an educational dataset provided by the International University of La Rioja, corresponding to 4 academic courses from the University's master's degree in Computer Security (from 2015 to 2018).

The dataset provided for the experiment consists of 11 files in CSV format, which were extracted from the Sakai platform. Table 2 lists the files provided and a brief description of their contents.

The course grades (Califications) are defined on the basis of the Evaluate Elements of each subject (Rooms). Within these elements are the activities (Tasks, Task Send) sent by the students registered in the educational platform (Users). In addition, the files reflect the messages (Messages), topics (Topics) and discussion forums (Forums) maintained by the users during the course. Finally, each time a user connects to the online platform, a login (Sessions) is recorded, which is accompanied by all the actions or events

Table 1 Summary of main findings related to the experimental case

Study	Goal	Application
Kuzilek et al. [12]	Defining a virtual environment-based educational dataset (OULAD)	Feature extraction - can be used as a reference model to identify the most relevant variables offered by the Sakai platform
Jin et al. [24]	Measuring the effectiveness of different feature selection methods	Feature selection - ANOVA and RFECV techniques are useful for handling structured data
Waheed et al. [6]	Predicting academic performance in virtual environments using Deep Learning	Feature extraction, predictive model definition - very similar to the use case, although it uses OULAD as dataset
Finn et al. [10]	Predicting academic performance using various Machine Learning techniques	Definition of the predictive model - although it does not apply the desired techniques, it serves as a reference regarding the expected results to be applied on an experimental dataset
Jin et al. [24]	Designing an efficient algorithm capable of searching neural network architectures	Architectural Search (NAS) - AutoKeras is an algorithm applied to structured data
Real et al. [25]	Applying evolutionary algorithms to discover neural network architecture automatically	Hyperparameter optimisation - TPE technique can prove effective in case studies

Table 2 CSV files provided for the experiment

File	Records	Attributes	Description
Califications	49513	12	Subject Grading
Evaluate Elements	682	24	Grading elements of the course
Events	3350557	7	Online user activity
Forums	197	27	Discussion forums
Messages	19230	28	User messages
Rooms	65	4	Course subjects
Sessions	675064	9	User sessions on the platform
Task Send	30318	7	Activities submitted by the student
Tasks	315	3	Activities associated with the subject
Topics	530	35	Discussion forum topics
Users	699	1	Registered users on the platform

(Events) that the user performs in the virtual environment, leaving the clickstream activity through the course contents. Since one of the objectives of this study is to identify and extract the characteristics that represent the student's interaction with the virtual platform, the Events file was used as the main source of information for our study, although numerous transformations were necessary during the preprocessing phase. In addition, the target variable was also extracted from other combinations of the Califications file with the scoring items, assignments and course subjects.

It is worth noting that all the subjects of the academic year followed the same evaluation criteria, with the final grade being the result of the grades of the assignments during the course (40%) and the final exam (60%). This allowed an objective prediction to be made, although the difficulty of the subjects, the number of credits and the teaching staff varied among the different courses and subjects, which are limiting factors for the good performance of the algorithm.

Finally, we performed a classification of the most relevant variables captured by the Sakai platform, and the use of advanced meta-learning techniques which facilitate the design of the deep neural network, among others. The main stages of the method are as follows:

1. Dataset generation. First, we create an educational dataset enriched with detailed information about the student's activity in the virtual environment, inspired by the Open University Learning Analytics Dataset (OULAD). This involves redefining the dataset extracted from the Sakai platform, feeding the model with more abstract information, and is therefore less dependent on the partial qualifications obtained. In addition to this, taking the OULAD standard as a reference helps to share the results with the scientific community, since multiple researchers use it as the basis for their experiments.
2. Baseline model design using Deep Neural Networks. We follow a traditional approach of neural architecture search and parameter optimisation, carried out manually, with the aim of establishing a baseline model that serves as a reference to verify the effectiveness of the subsequent techniques used. All the input variables provided in the initial stage are used, regardless of their correlation degree, which is evaluated at the next stage.
3. Feature selection. Once the reference model is obtained in the previous stage, we evaluate the impact on feature reduction after applying the ANOVA and RFECV filter methods, as described in [19]. Depending on the results obtained, the most representative variables are used as the input dataset for optimizing the model through meta-learning.
4. Neural architecture search. Using meta-learning techniques that are suitable to structured data, we look for the best neural network topologies, using two different search algorithms: Greedy and Bayesian. The most efficient architecture found by both methods are then evaluated taking the rest of the hyperparameters used in the baseline model, so that we can compare their performance and choose the most promising neural network.
5. Hyperparameter optimisation. Once the final architecture is known, we use an upper layer of abstraction so the model can learn to learn which hyperparameters values are optimal to solve the proposed problem. For this, we use one of the well-known techniques: The Tree-structured Parzen Estimator.
6. Final model and results analysis. We use the hyperparameters obtained in the previous stage to build the final model and analyse its precision with respect to the baseline model created on the second stage.

4 Results

The results obtained in each of the steps of the method are described below.

4.1 Educational dataset

The feature extraction procedure based on the student's online activity resulted in a total of 18 input variables. The final relationship is shown in Table 3, including the partial and total grades obtained during the course.

Statistical analysis of the final dataset yielded some interesting results. In total, we obtained 5,066 records, each of which represents the student's online activity, as well as his or her grade, in a single subject of the UNIR's university master's degree in Computer Security. In addition, we identified almost 500 students, distributed among the 4 academic courses that were analysed, with an average final score of 6.55 out of 10. As for online activity, an average of 400 clicks were made per subject and student. Anecdotally, it should be noted that those who exceeded 3000 clicks, failed the subject. In return, most of the students who obtained an outstanding grade did not post any messages in the discussion forum.

4.2 Baseline model

We obtained the reference model following a manual optimisation process, after testing multiple combinations of hyperparameters and neural architecture. Table 4 shows the range of values used for the different hyperparameters, including the number of hidden layers, nodes, learning rate, optimiser, etc. It is important to note that not all possible combinations were tested, since it is a highly time-consuming task, although all values were used by some models. The implementation was carried out using Google Colab and Keras framework. The division of training and test data ratio were 80-20, respectively.

The accuracy of the predictive models for the validation set ranged between 55 and 67%, while, without a regularisation layer, the neural network was capable of achieving an accuracy of 90% on the training set. This is logically due to the model's ability to 'memorise' the input data, as it only has 5,000 records.

Table 3 List of features obtained after processing the educational dataset

Feature	Description
student_id	Student identifier
site_id	Subject identifier
site_name	Name of the subject
grade_cont	Grade obtained in continuous assessment
grade_exam	Grade obtained in the exam
grade_final	Final grade obtained in the subject
grade	Classification of the final grade for the subject (0: fail, 1: pass, 2: remarkable, 3: outstanding)
activities	Total number of evaluated works
calendar	Total number of calendar views
clicks	Total number of actions performed on the platform
content_new	Amount of content uploaded by the student (attachments)
content_read	Total number of clicks on resources related to the subject (PDF)
ects	Number of credits of the subject
forums_new	Number of messages sent by the student in the forum
forums_read	Number of messages read by the student in the forum
homepage	Number of views of the subject's home page
lessons	Number of accesses to virtual classes
messages_new	Number of messages sent by the student through the internal mail of the subject
messages_read	Number of internal mail messages read
pages	Total number of clicks on different pages of the subject
quiz	Total number of virtual quiz attempts
site_time	Total estimated time of the student in the contents of the subject
task_read	Number of visualizations of assignments of the subject
task_sent	Total number of submitted tasks
webcontent_read	Total number of views in web content (subject links)

Table 4 Range of values used for manual hyperparameter optimisation

Hyperparameter	Range of values
Hidden layers	1 – 10
Number of nodes	16, 64, 128, 256, 512, 1024
Number of classes	4
Activation function	ReLU, Tanh for hidden layers; Softmax for output layer
Cost function	Categorical Cross-Entropy
Initializer	Glorot Uniform, Random Normal
Optimizer	SGD, Adam
Learning rate	0.001, 0.01, 0.1
Batch size	16, 64, 128
Epochs	20, 50, 100, 300, 500

The best results were obtained with a model of 5 hidden layers, an initial one of 128 neurons and the rest of 64, plus an output layer of 4 classes, one for each possible value of the final grade (failed, passed, remarkable and outstanding). We used ReLU as the activation function in all layers, except for the output layer, that was set up as a softmax. Regarding the rest of the hyperparameters, we chose Adam as optimiser and Glorot Uniform as initialiser, which had a slightly higher performance than SGD and Random Normal, respectively. The learning rate was set at the default value defined by Keras for Adam, 0.001.

Figure 1 shows the evolution of the model over the training and validation sets during 20 time periods. The final accuracy, over the test set, was 67.1%.

4.3 Most relevant features

We used ANOVA and RFECV techniques for the input variables set to determine those that were most relevant in our experimental case. First, we evaluated the RFECV method, which automatically figures out the optimal number of variables to be used in the model. As can be seen in Fig. 2, the best results were obtained for 13 features, although from 4 features onwards, similar performances were observed in terms of accuracy.

This optimal number of variables was used as an input parameter in the implementation of the ANOVA method, so that both techniques could be evaluated under equal conditions. The results show that 9 of the 13 variables were common in both cases, while there was some discrepancy when discarding variables of little relevance. Only one of these (the attachments uploaded by the student to the platform (content_new)), was excluded by ANOVA and RFECV (Table 5).

Finally, using the baseline model created in the previous stage, the two new datasets (selection filters of the original dataset) were evaluated and compared, in order to choose the one with the best performance. Both accuracy and loss were slightly higher in the set of features selected by RFECV, with differences of around 2% over the validation set (Figs. 3 and 4). Accuracy for the test set was 63.3% using ANOVA and 66.7% using RFECV.

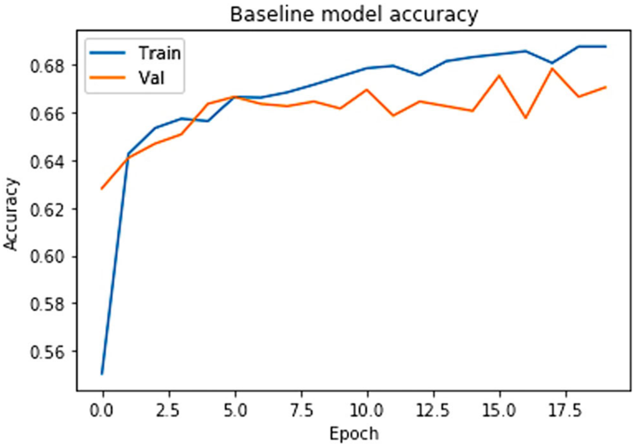


Fig. 1 Baseline model accuracy

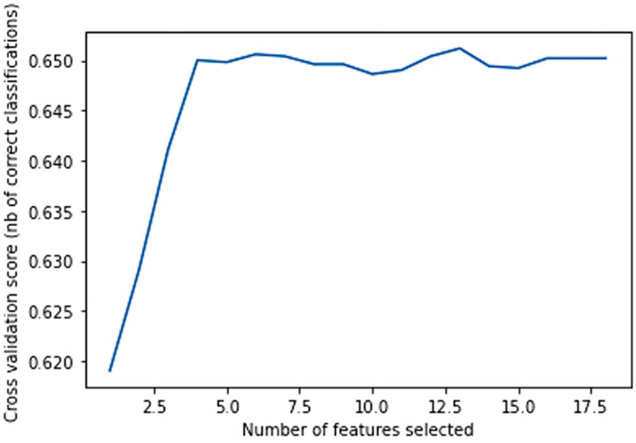


Fig. 2 Optimal number of features using RFECV

Table 5 Most relevant features using ANOVA and RFECV techniques

Feature	ANOVA	RFECV
activities	X	X
calendar		X
clicks	X	
content_new	X	
content_read	X	X
ects		X
forums_new	X	X
forums_read	X	
homepage	X	
lessons	X	X
messages_new		X
messages_read	X	X
pages	X	X
quiz	X	X
site_time	X	
task_read	X	X
task_sent	X	X
webcontent_read		X

4.4 Neural architectures found

We performed the neural architecture search by AutoKeras using two different algorithms: Greedy and Bayesian. During the experiment, the search space was configured to limit it into blocks of dense layers, suitable for the treatment of structured data. After 20 attempts testing different combinations of layers and nodes, these were the ones that reported the best results in each case (Table 6).

Subsequently, the performance of these architectures was evaluated and compared using Keras. Although in both cases we reported very similar results in terms of accuracy and loss reduction, the architecture found by the Greedy algorithm, composed of 16, 1024 and 512 neurons, respectively, showed a slightly better performance on the test set, with 66.3% of accuracy, compared to 65.4% of accuracy for the design suggested by the Bayesian algorithm. Figures 5 and 6 show the evolution of both models by time period during training.

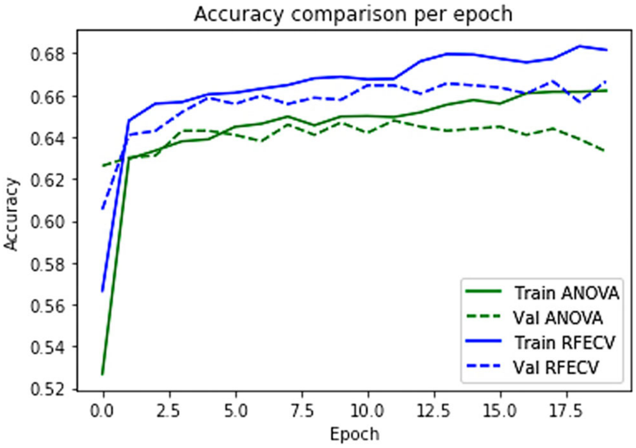


Fig. 3 Accuracy comparison after feature selection using ANOVA and RFECV techniques

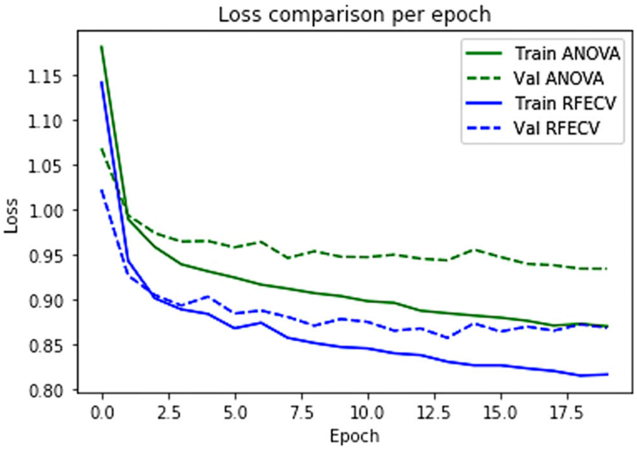


Fig. 4 Loss comparison after feature selection using ANOVA and RFECV techniques

4.5 Optimal hyperparameters

Once we found the best architecture suggested by the Neural Architecture Search algorithms, it was used to configure the hyperparameter search space. For this, we used the Tree-structured Parzen Estimator algorithm so that, in a maximum of 20 attempts, we found the best combination of optimiser, learning rate, dropout rate, and batch size. Furthermore, thanks to the graphical interface provided by the Neural Network Intelligence toolkit, it was possible to analyse the results according to various categories, which greatly facilitated the understanding of the experiments carried out autonomously by the algorithm. The following images (Fig. 7 and 8) show the accuracy obtained in each attempt, which were also stored in the programme log based on a unique identifier. The 10 best models obtained an accuracy of between 65% and 67%, while, in the global calculation, there were several models with accuracies of 63%, reaching a minimum of 50%.

Regarding the optimal combinations of hyperparameters, the 4 best models used Adam as optimiser, although the learning rate ranged between 0.001 and 0.01 (Fig. 9). Regarding batch size and dropout rate, these took values of 16 and 64, and around 0.2, respectively. For its part, the best model achieved an accuracy of 66.9% over the test set, using, in addition to the Adam optimiser, a learning rate of 0.005, batch size of 16 elements, and a dropout rate slightly higher than 0.14 (Fig. 10).

Table 6 Neural architecture suggested by AutoKeras using Greedy and Bayesian

Hyperparameter	Greedy	Bayesian
Hidden layers	3	3
Number of nodes	16 – 1024 – 512	64 – 1024 – 16
Number of classes	4	4
Activation function	ReLU; Softmax in output layer	ReLU; Softmax in output layer
Cost function	Categorical Cross-Entropy	Categorical Cross-Entropy
Trials	20	20

4.6 Final model

Finally, we used the previous results to build the final model, whose architecture and hyperparameters were automatically optimised thanks to the algorithms already mentioned. As a summary, Table 7 shows the values used in the configuration of the final model.

After 50 time periods of training, the model obtained an accuracy over the test set of 67.2%. As in previous sections, Fig. 11 shows the evolution by time period in terms of accuracy.

5 Discussion

The results obtained during the experimentation phase yield interesting observations.

During Stage 1, we were able to identify the features related to the virtual activity of the students on the educational platform, in addition to the partial and final grades on each subject. In this regard, it is important to highlight the limited information that could be obtained through the files provided, which did not contain a descriptive file which would have allowed for an understanding of the relationships between entities, or a formal description of the 157 attributes included. This required, in the first instance, a detailed analysis of the information, as well as a systematic processing to group, filter and transform the files. In addition, the final course grades were not available on the platform, so they had to be calculated during the extraction process.

On the other hand, the final dataset volume, composed of 18 input variables and 5,066 records, was scarce if we compared our case study with others carried out with OULAD, The Open University dataset. The nature of Deep Learning, based on an understanding of the complex relationships in large datasets, suggests that the information from the experiment may be insufficient to achieve a high level of accuracy.

However, in Stage 2, during the creation of the neural network baseline model, we obtained a maximum accuracy over the test set of 67.1%. Although achieving high precision was not among the objectives in this research, having an enriched database with more information would have allowed us to appreciate relevant differences between several architecture and hyperparameter settings. As other studies show, other relevant data, such as the student's sociodemographic information, or the history of previous grades, would have made a valuable contribution to the prediction results.

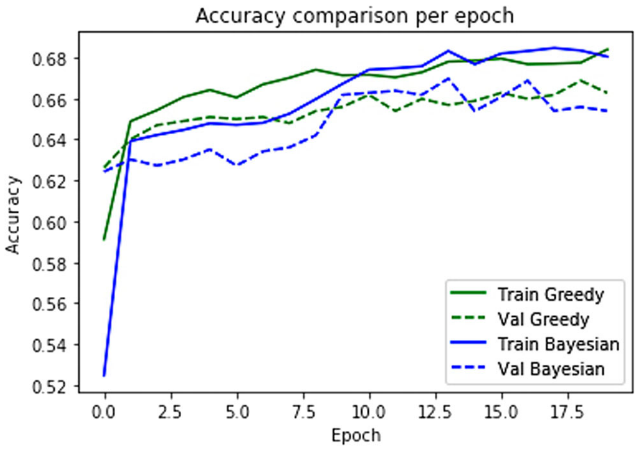


Fig. 5 Model accuracy using the architecture found by Greedy and Bayesian algorithms

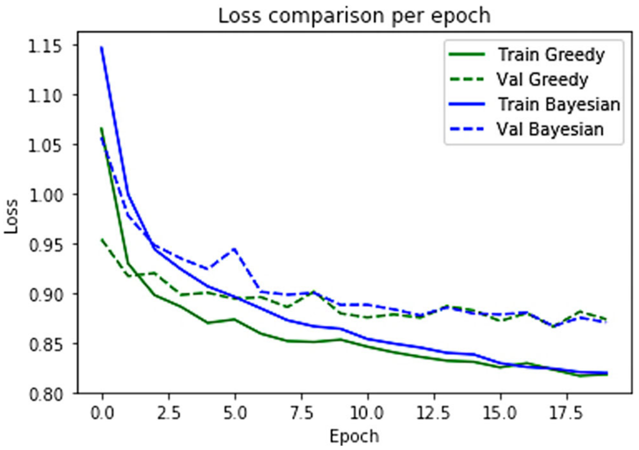
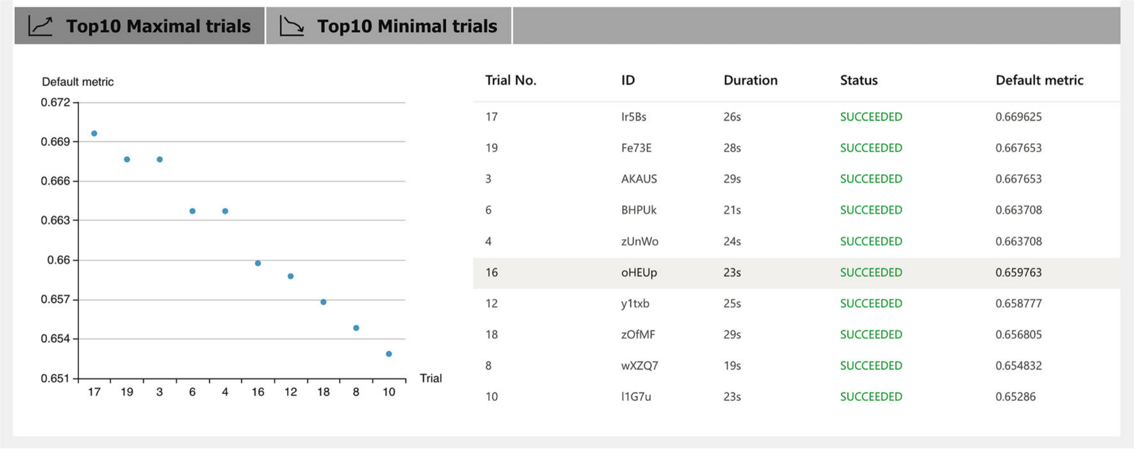


Fig. 6 Model for loss using the architecture found by Greedy and Bayesian algorithms



Trial No.	ID	Duration	Status	Default metric
17	Ir5Bs	26s	SUCCEEDED	0.669625
19	Fe73E	28s	SUCCEEDED	0.667653
3	AKAUS	29s	SUCCEEDED	0.667653
6	BHPUk	21s	SUCCEEDED	0.663708
4	zUnWo	24s	SUCCEEDED	0.663708
16	oHEUp	23s	SUCCEEDED	0.659763
12	y1txb	25s	SUCCEEDED	0.658777
18	zOfMF	29s	SUCCEEDED	0.656805
8	wXZQ7	19s	SUCCEEDED	0.654832
10	l1G7u	23s	SUCCEEDED	0.65286

Fig. 7 Experiment classification based on its accuracy (NNI)

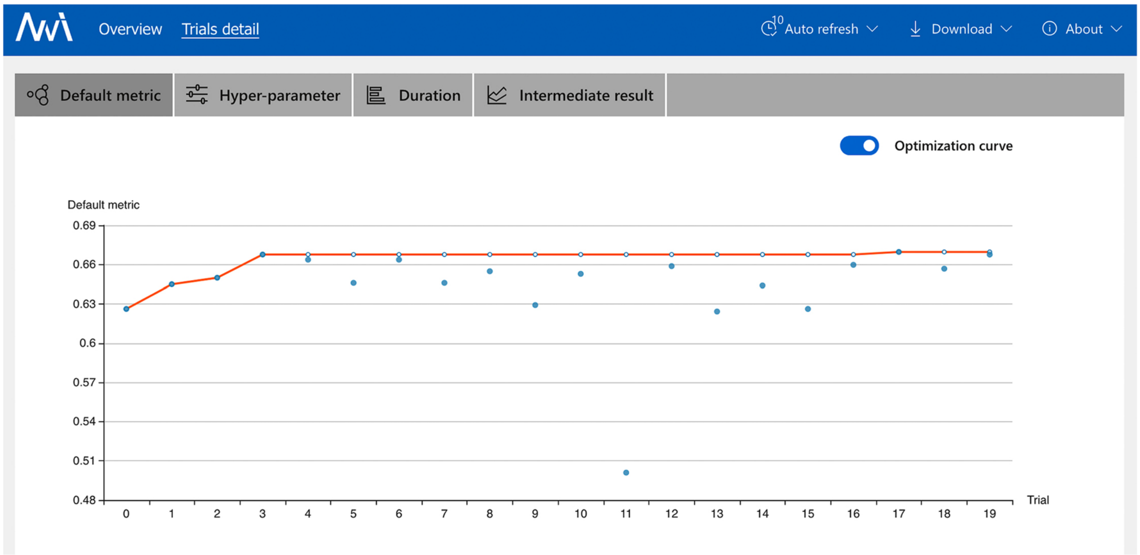
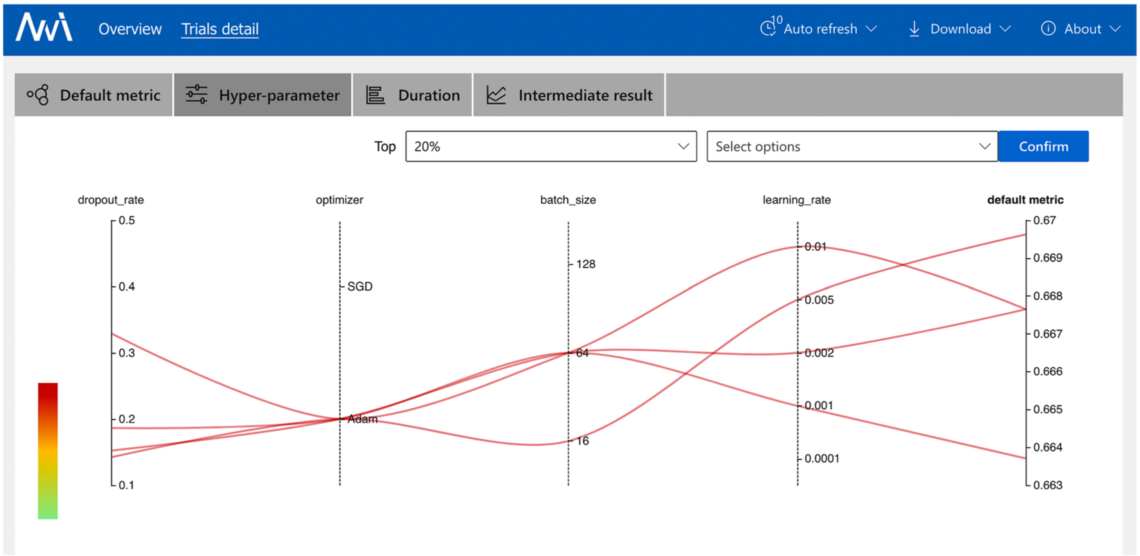
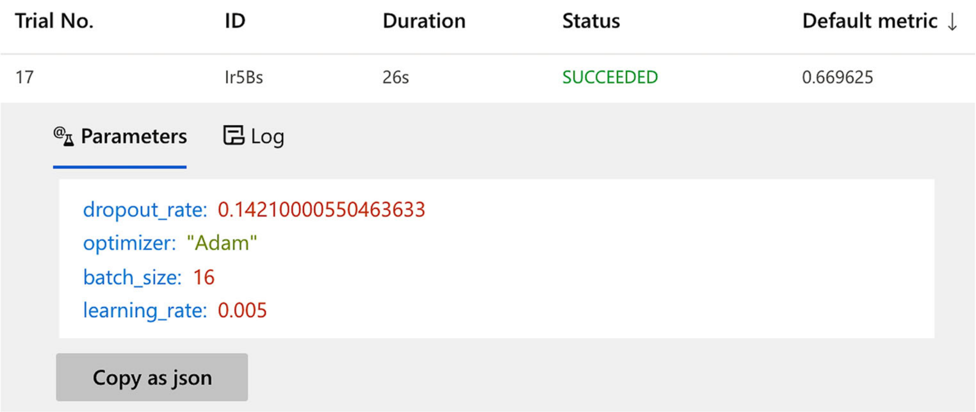


Fig. 8 Experiment accuracy (NNI)



Top 20%

Fig. 9 Hyperparameters obtained on the best 4 models (NNI)



Trial No.	ID	Duration	Status	Default metric ↓
17	Ir5Bs	26s	SUCCEEDED	0.669625

dropout_rate: 0.14210000550463633
optimizer: "Adam"
batch_size: 16
learning_rate: 0.005

Fig. 10 Best hyperparameter combination (NNI)

In Stage 3, the ANOVA and REFCV techniques did not provide conclusive results on the most relevant features, although there were similarities in 9 of the 13 selected variables. Data such as the number of activities delivered, lectures visualisation or number of test attempts completed during the course were useful to predict academic performance. In terms of accuracy, reducing the number of features from 18 to 13 barely impacted on the precision of the baseline model, which came from 67.1% to 66.7% in the case of RFECV. In a larger dataset, the dimensionality reduction would have favoured the computational speed of the model. In our case, as there were only 5,000 records, the improvement was not particularly relevant.

During Stage 4, we tried several Neural Architecture Search algorithms, with the aim of designing neural network architecture autonomously, that is, without manual intervention by a Deep Learning specialist. The structured nature of the data, composed of independent numerical variables, limited the possibilities during the meta-learning phase. Algorithms such as ENAS and DARTS, which offer a superior performance and greater efficiency than other methods, are designed to solve image classification problems and therefore could not be used in this research. Failing that, we performed the architecture search in AutoKeras, using Greedy and Bayesian algorithms. Both techniques identified topologies composed by 3 hidden layers in the search space, with an intermediate layer of 1,024 neurons. Comparatively, the proposed architecture had two layers less than the one designed manually, although the number of nodes increased considerably: 128 x 64 x 64 x 64 x 64 on the traditional model versus 16 x 1024 x 512 on the standalone model. The latter turned out to be more efficient, since it contained a much fewer parameters than the Stage 2 model.

The next step in the design of the autonomous model was carried out in Stage 5, with a hyperparameter search that would optimise the architecture suggested by the AutoKeras Greedy algorithm: optimiser, learning rate, dropout rate and batch size. We completed a total of 20 attempts using the Tree-structured Parzen Estimator method, obtaining accuracies which generally ranged from 63% to 67%, although one of the cases fell to a precision of 50%. This highlights the impact that hyperparameter optimisation can have on a classification problem, even in those studies where the accuracy is limited by the input dataset, as discussed above. Regarding the manual design,

Table 7 Final values obtained by the autonomous model

Hyperparameter	Value
Hidden layers	3
Number of nodes	16 – 1024 – 512
Number of classes	4
Activation function	ReLU; Softmax in output layer
Cost function	Categorical Cross-Entropy
Initializer	Glorot Uniform
Optimizer	Adam
Learning rate	0.005
Dropout rate	0.14
Batch size	16
Epochs	50

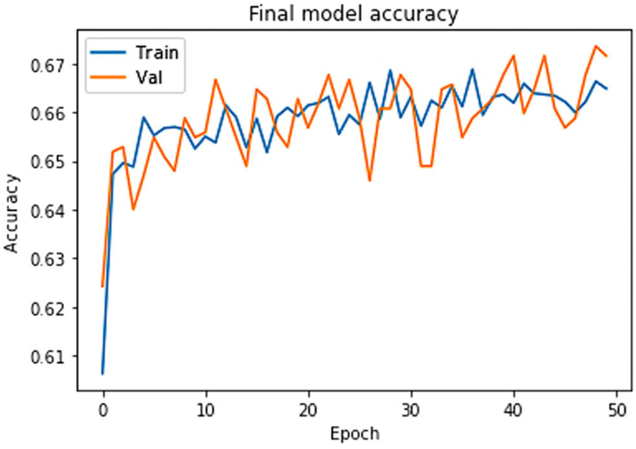


Fig. 11 Final model accuracy

the optimiser used was the same, Adam, although the rest of hyperparameters were different. In this section, it is relevant to point out that the algorithm is capable of even suggesting the dropout rate, which helps to stabilise the training of the model by reducing the overfitting.

Finally, during Stage 6, the model resulting from previous stages was implemented with Keras, so that its performance was compared with the manual model created in Stage 2. The accuracy of the autonomous model over the test set was 67.2%, which is very similar to that obtained with the manually designed model, of 66.7%, comparing the two of them using the filtered dataset of 13 input variables. These results corroborate that it is possible to design efficient models which are optimised automatically, with the help of meta-learning techniques, reducing the amount of time invested in optimisation tasks.

6 Conclusions

In this paper we have carried out the study of the state-of-art in Deep Learning and Meta-Learning and have applied some of these advances to an experimental case, a set of educational data provided by the International University of La Rioja, corresponding to 4 academic courses of an online master's degree. To do this, we implemented a methodology consisting of 6 stages: analysis and processing of the data provided for the generation of an educational dataset (Stage 1), creation of a manually optimised neural network baseline model (Stage 2), selection of most relevant features for predicting grades (Stage 3), search for neuronal architecture and hyperparameters autonomously (Stages 4 and 5, respectively), and final evaluation of the autonomous model (Stage 6), which can be compared with the one generated manually.

After conducting the experiment, the results yielded some interesting conclusions. It is clear that there was not enough information available in the case study. We could only extract 18 features, while similar studies carried out on OULAD extracted up to 50 variables. Moreover, the volume of the data provided by UNIR contained 5,066 unique subject grades, while OULAD has more than 32,000 records. This limited the possibilities of the experiment and impacted the performance of the predictive model. Therefore, the deep neural network, designed manually, achieved a precision of only 67% over the test set. Although obtaining high accuracy was not one of the objectives of our research, this was a challenge during the experiment, since it was difficult to detect the improvements caused by certain modifications in the learning model.

Regarding the feature selection, we identified those which had the greatest impact on the model's accuracy. The techniques of analysis of variance (ANOVA) and recursive elimination (RFECV), showed similar results on 70% of the cases (9 of the 13 chosen variables). Some expected features, such as the number of activities carried out by the student or the number of test attempts, turned out to be relevant in the prediction. Other less obvious variables also had a positive influence, such as the number of pages visited in the virtual classroom or the number of messages read by the student. On the contrary, the number of clicks made, the messages read in the discussion forum, or the content created by the student (number of attachments) were not particularly relevant. In fact, the elimination of such variables did not affect the precision of the manually created model, going from 67.1% to 66.7%.

In the Meta-Learning section, we designed and implemented a learning model that was able to be optimised autonomously. To do this, we approached the problem from two different perspectives. On the one hand, we analysed and tested different search algorithms for neural architecture, although only AutoKeras was applied in the case study. The structured nature of the data prevented the use of other Neural Architecture Search techniques, such as ENAS or DARTS, since they are designed to deal with image classification problems. The architecture suggested by AutoKeras turned out to be more efficient than the one designed manually, as it had fewer layers and parameters. On the other hand, we carried out an automatic search for hyperparameters which optimised the previous architecture. Using the Neural Network Intelligence toolkit and the Tree-structured Parzen Estimator algorithm, we ran up to 20 experiments to find the best combination of optimiser, learning rate, dropout rate, and batch size.

The final model, which was designed entirely based on the combination of the previous results, obtained an accuracy of 67.2% over the test set. From this, it can therefore be concluded that an autonomously optimised model using meta-learning techniques is able to obtain similar predictions to those achieved by manually designed models. In other words, the algorithm is capable of learning to learn the parameters that best adapt to a classification problem, specifically in the educational field. This opens up new application possibilities for the disciplines of Learning Analytics and Educational Data Mining. The latter, especially focused on the development of techniques which facilitate the analysis of Big Data in education, can benefit from the use of predictive models that incorporate meta-learning to automate experimentation, save design time, or even improve results.

Contributions The contributions of this study are described as follows:

Use of Deep Learning models in the prediction of academic performance. Throughout the experiment, we used deep neural networks to build our predictive model.

Although it has been used in Educational Data Mining on a few studies previously, majority of research is focused on classic algorithms such as SVM, Random Forest or decision trees.

Creation of an autonomously optimised model through meta-learning. We built a predictive model based on the combination of meta-learning algorithms for architecture search and hyperparameters optimisation. The results show that the autonomous model obtains the same accuracy compared to the one which is manually designed.

There was an improvement on model efficiency and scalability, even using more complex data sets. We found that it is possible to obtain models of equal performance with less effort, which produces important benefits in terms of efficiency and scalability. Meta-Learning models are not only time-saving in the design phase, but these can also be adapted to problems with more complex datasets, i.e., those which contain a greater number of input variables.

Future areas of work The work carried out opens up interesting areas for future research in the fields of Learning Analytics and Educational Data Mining in general, and in the prediction of academic performance in virtual environments in particular. Some of them are highlighted below:

- Application of Meta-Learning techniques for other educational datasets, such as OULAD.
- Adaptation of algorithms such as MAML (Model Agnostic Meta-Learning) to solve few-shot learning problems in LA and EDM.
- Grade prediction using only partial data of the activity recorded by the student.
- Use of other Artificial Intelligence techniques, such as Sentiment Analysis, to predict the student's academic performance.

Author Contributions Ángel Casado Hidalgo carried out the experiment and wrote the manuscript with support from Pablo Moreno Ger. Luis De La Fuente Valentín helped to supervise the project. All authors read and approved the final manuscript.

Funding This work has been partially funded by the PLeNTaS project, "Proyectos I+D+i 2019", PID2019-111430RB-I00 and by Universidad Internacional de La Rioja (UNIR, http://www.unir.net) through the IBM-UNIR Chair on Data Science in Education.

Availability of data and material The data which supports the findings of this study are available from the International University of La Rioja but restrictions apply to the availability of this data, which was used under license for the current study, and so are not publicly available. However, data is available from the authors upon reasonable request and with the permission of the International University of La Rioja.

Code Availability All code for data analysis and model training associated with the current submission is available at https://colab.research.google.com/drive/1pQNm36nEgA83FGqno-HvJH8AJRpoFmlj?usp=sharing

Declarations

Conflict of Interests The authors declare that they have no conflicts of interest.

References

1. The Open University (2020) OU50. www2.open.ac.uk. http://www2.open.ac.uk/about/annual-report-2018-19/#highlights.. Accessed 8 October 2020
2. Guerrero M, Heaton S, Urbano D (2021) Building universities' intrapreneurial capabilities in the digital era: The role and impacts of massive open online courses (MOOCs). Technovation 99:102139. https://doi.org/10.1016/j.technovation.2020.102139
3. Class Central (2020) The second year of the MOOC: a review of MOOC stats and trends in 2020. https://www.classcentral.com/report/the-second-year-of-the-mooc/. Accessed 8 October 2020
4. Liñán LC, Pérez ÁAJ (2015) Minería de datos educativos y análisis de datos sobre aprendizaje: Diferencias, parecidos y evolución en el tiempo. RUSC Univ Knowl Soc J 12(3):98–112. https://doi.org/10.7238/rusc.v12i3.2515
5. Jovanović J., Gašević D., Dawson S, Pardo A, Mirriahi N (2017) Learning analytics to unveil learning strategies in a flipped classroom. Internet High Educ 33:74–85. https://doi.org/10.1016/j.iheduc.2017.02.001
6. Waheed H, Hassan SU, Aljohani NR, Hardman J, Alelyani S, Nawaz R (2020) Predicting academic performance of students from VLE big data using deep learning models. Comput Hum Behav 104. https://doi.org/10.1016/j.chb.2019.106189
7. Montavon G, Samek W, Müller KR (2018) Methods for interpreting and understanding deep neural networks, vol 73, pp 1–15. Elsevier Inc. https://doi.org/10.1016/j.dsp.2017.10.011
8. Vanschoren J (2019) Meta-learning. In: Hutter F, Kotthoff L, Vanschoren J (eds) Automated machine learning. The springer series on challenges in machine learning. Springer, Cham. https://doi.org/10.1007/978-3-030-05318-5_2
9. Zoph B, Vasudevan V, Shlens J, Le Q (2018) Learning transferable architectures for scalable image recognition. In: Proceedings of the IEEE computer society conference on computer vision and pattern recognition, pp 8697–8710. https://doi.org/10.1109/CVPR.2018.00907
10. Finn C, Abbeel P, Levine S (2017) Model-agnostic meta-learning for fast adaptation of deep networks. In: Proceedings of the 34th international conference on machine learning - volume 70 (ICML'17). JMLR.org, pp 1126–1135
11. Romero C, Ventura S (2020) Educational data mining and learning analytics: An updated survey. WIREs Data Min Knowl Discov 10(3). https://doi.org/10.1002/widm.1355
12. Kuzilek J, Hlosta M, Zdrahal Z (2017) Data descriptor: open university learning analytics dataset. Sci Data 4:1–8. https://doi.org/10.1038/sdata.2017.171
13. Coelho OB, Silveira I (2017) Deep learning applied to learning analytics and educational data mining: a systematic literature review. In: Anais Do XXVIII Simpó,sio Brasileiro de Informática Na Educação (SBIE 2017), p 143. https://doi.org/10.5753/cbie.sbie.2017.143
14. Hassan SU, Waheed H, Aljohani NR, Ali M, Ventura S, Herrera F (2019) Virtual learning environment to predict withdrawal by leveraging deep learning. Int J Intell Syst 34(8):1935–1952. https://doi.org/10.1002/int.22129
15. Fei M, Yeung DY (2016) Temporal models for predicting student dropout in massive open online courses. In: Proceedings - 15th IEEE International conference on data mining workshop,

ICDMW 2015, pp 256–263. https://doi.org/10.1109/ICDMW.2015.174

16. Naser SA, Zaqout I, Ghosh MA, Atallah R, Alajrami E (2015) Predicting student performance using artificial neural network: in the faculty of engineering and information technology. Int J Hybrid Inf Technol 8(2):221–228. https://doi.org/10.14257/ijhit.2015.8.2.20
17. Strecht P, Cruz L, Soares C, Mendes-Moreira J, Abreu R (2015) A comparative study of classification and regression algorithms for modelling students' academic performance. In: 8th international conference on educational data mining
18. Zaffar Maryam, Ahmed Hashmani Manzoor, Savita KS, Rizvi SyedSajjadHussain (2018) A study of feature selection algorithms for predicting students academic performance. Int J Adv Comput Sci Appl (IJACSA) 9(5). https://doi.org/10.14569/IJACSA.2018.090569
19. Yeoh TW, Daolio F, Aguirre HE, Tanaka K (2017) On the effectiveness of feature selection methods for gait classification under different covariate factors. Appl Soft Comput J 61:42–57. https://doi.org/10.1016/j.asoc.2017.07.041
20. Elsken T, Metzen JH, Hutter F (2019) Neural architecture search: a survey. J Mach Learn Res 20. http://jmlr.org/papers/v20/18-598.html
21. Bergstra J, Bardenet R, Bengio Y, Kégl B. (2011) Algorithms for hyper-parameter optimization. In: Advances in neural information processing systems 24: 25th annual conference on neural information processing systems 2011, NIPS 2011, pp 1–9. https://doi.org/10.1007/978-3-030-05318-5_1
22. Liu H, Simonyan K, Yang Y (2019) DARTS: Differentiable architecture search. In: 7th international conference on learning representations, ICLR 2019. arXiv:1806.09055
23. Pham H, Guan MY, Zoph B, Le Qv, Dean J (2018) Efficient neural architecture search via parameters sharing. In: Proceedings of the 35th international conference on machine learning, PMLR, vol 80. pp 4095–4104
24. Jin H, Song Q, Hu X (2018) Auto-Keras: An efficient neural architecture search system. In: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (KDD '19). Association for Computing Machinery, New York, pp. 1946-1956. https://doi.org/10.1145/3292500.3330648
25. Real E, Moore S, Selle A, Saxena S, Suematsu YL, Tan J, Le Qv, Kurakin A (2017) Large-scale evolution of image classifiers. In: Proceedings of the 34th international conference on machine learning - Volume 70 (ICML'17). JMLR.org, pp 2902–2911.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Ángel Casado Hidalgo is a Predoctoral Fellow at Universidad Internacional de La Rioja (UNIR), where he obtained a MSc in Computer Science - Artificial Intelligence (First Class Honours). He is also a member of the Data Driven Science research group and a visiting student at The Open University (UK), where he is studying Meta-Learning applications in Learning Analytics.

Dr. Pablo Moreno-Ger got his PhD from Universidad Complutense de Madrid, and is an Associate Professor at Universidad Internacional de La Rioja (UNIR), where holds de IBM Chair on Data Science in Education. He is the Dean of the School of Engineering and Technology at UNIR, and he has a wide international research experience including visiting scholarships at different universities (Harvard University, the Open University of the Netherlands and Universidade de Coimbra) and participation in several national and international research projects. He has authored more than 150 publications in journals and international conferences.

Luis De La Fuente Valentín is a full-time associate professor at Universidad Internacional de La Rioja, UNIR. He got his PhD at Universidad Carlos III de Madrid, in 2011. He has authored more than 40 papers and participated in several national and European public funded projects, one of them as investigator in charge. His current research interest is on machine learning tools applied to the educational field.