Research Article
Journal of Intelligent Systems: Theory and Applications 7(2) (2024) 43-55
DOI: 10.38016/jista.1200225

Assessment of effective factors on student performance based on machine learning methods

Hasan Yıldırım[1*]

[1] Department of Mathematics, Karamanoğlu Mehmetbey University, Karaman, Türkiye

hasanyildirim@kmu.edu.tr

Abstract

Machine learning methods have gained increasing attention in the field of education due to advancing technological tools and rapidly growing data. The general focus of this attention is on identifying the best method, but it is also critical to determine the extent to which the methods under consideration differ statistically and to correctly identify variable importance metrics. In this study, we benchmarked the performance of twenty-three machine learning algorithms on real educational data via cross-validation based on criteria such as accuracy, AUC and F1-score. Besides, the methods were statistically compared using DeLong and McNemar tests. The findings showed that the LightGBM method appeared to be the best method and presented the most important factors determining student achievement according to this method. The systematic process followed in the study is considered to yield valuable insights for data-driven studies as well as the field of education.

Keywords: Student performance, Machine learning, Artificial intelligence, Feature selection, Statistical analysis

1. Introduction

Artificial intelligence is increasingly deeply integrated into real life and is enhancing human beings' ability to predict routines, capacities, and behaviors. With the technological facilities that are being developed to achieve these goals, the size of the data collected is also expanding proportionally to the ability to collect and process data. The field of education is arguably one of the fields generating the highest amount of valuable knowledge from the data gathered. Artificial intelligence models can be effectively used for primary purposes such as students' performance evaluations (in-term and end-of-term), dropout status, and identification of individuals at risk (Albreiki et al., 2021). The beneficial results provided by artificial intelligence models have expanded their usages in education and enabled them to prepare personalized (Li et al., 2020), updated (Guan et al., 2020) course content, developed effective course selection tools (Tilahun ve Sekeroglu, 2020), and prepared exam formats (Wu et al., 2020).

Educational research is conducted not only to improve decisions locally, but also on the results of several examinations (such as program for international student assessment (PISA), trends in international mathematics and science study (TIMSS), etc.) carried out globally. The motivation underlying these studies is to improve the socio-economic status and quality of life of both individuals and the countries in which they live through improving the quality of education (Sağlam and Aydoğmuş, 2016). Machine learning, as the most important sub-field of artificial intelligence, contributes significantly to realizing this motivation and providing accurate recommendations to decision (policy) makers.

The usage of machine learning models in the field of education is particularly focused on the supervised learning. Supervised learning is based on assuming that the quantitative (e.g., student grade: 82/100, student attendance percentage: 73%) or qualitative (e.g., student achievement status: failed or success, student grade: AA or FF) variable that is the focus of the study is known and accurately predicted by a set of variables that are expected to affect it. The most widely used algorithms in the literature for this type of learning (Sekeroglu et al., 2021) are logistic regression (LR), naive bayes (NB), k-nearest neighbor (KNN), classification and regression trees (CART), linear regression (LIN), random forests (RF), bagging (BG), gradient boosting machine (GBM), extreme gradient boosting (Xgboost), artificial neural networks (ANN), support vector machines (SVMs), extreme learning machine (ELM), long short-term memory (LSTM), deep neural networks (DNN). In the literature, there are numerous studies involving such

[*] Corresponding Author
E-mail: hasanyildirim@kmu.edu.tr

Recieved : 31 Oct 2023
Revision : 14 Feb 2024
Accepted : 2 Jul 2024

models, some of the prominent studies can be given as follows:

Gamuling et al. (2016) studied student performance prediction in blended learning environments using discrete Fourier transforms (DFT) and various machine learning methods (including KNN, SVM, ANN and NB). Elbadrawy et al. (2016) proposed to utilize random forests, personalized multi-regression, and matrix factorization approaches to predict students' grades and assessments in future courses. Tran et al. (2017) proposed a unified system that connects classical machine learning methods (LR, CART and SVM) and recommender systems to predict student performance.) Like Tran et al. (2017), but not including the outputs of recommender systems, Adejo and Connolly (2018) presented an ensemble model incorporating cart, ann and svm methods to predict student performance. Hussain et al. (2019) employed various methods such as CART, LR, ANN, SVM and NB to identify the difficulties encountered by students during the term and to improve their performance. Yousafzai et al. (2020) employed genetic algorithm, CART and KNN models through both classification and regression models to predict student performance. Deo et al. (2020) have proposed models such as ELM, RF and Volterra to predict student performance in engineering mathematics courses and presented the results comparatively. Assellman et al. (2021) utilized RF and some boosting-based algorithms (including Adaboost and Xgboost) to accurately predict student performance. Suleiman and Anane (2022) have applied LR, CART, SVM and RF algorithms to predict the cumulative grade of students based on their performance in different years. Pallathadka et al. (2023) have comparatively presented the results of Naive Bayes, ID3, C4.5, and SVM models for predicting student performance. Chen and Zhai (2023) have compared the results of KNN, CART, RF, LR, SVM, NB, and ANN models in different application scenarios using several different datasets. Extensive studies on this topic are currently ongoing and comprehensive listings of these studies categorized according to aims, methods and outcomes can be obtained from the reviews by Albreiki et al. (2021), Sekeroglu et al. (2021) and Alalawi et al. (2023).

1.1. Study Aims and Motivation

The use of machine learning models in the literature is beneficial to a certain extent, however, some aspects have been relatively often disregarded:

i. It is critical in data-driven education studies to realize this motivation by not only estimating the value of the target variable that is the focus of the study, but also identifying the important factors that affect it. The variable importance measures can lead to more compact and scalable models.

ii. The statistical significance in performance comparisons of machine learning models can provide additional insights in model selection. The principle that the best model is the simplest model can be followed unless there is a significant difference.

This study focuses on these two mentioned perspectives and presents a comprehensive comparison of best machine learning algorithms. The content of the study is summarized as follows: The methods evaluated in the study are given in Section 2. Section 3 provides details about the experimental process. Model training results are presented in Section 4. Finally, the discussion and summary comments on the results of the study are reported in Section 5.

Table 1. List of models (algorithms) evaluated in the study

Type	Abbrevation	Model (Authors)	Brief Explanation
Instance-based	KNN	K-nearest neigbors (Cover and Hart, 1967)	The versatile algorithm employed in machine learning for both classification and regression tasks, and operates on the principle that similar data points are generally close in feature space. Its applications range from recommender systems to pattern recognition and anomaly detection, making it invaluable in academic and industrial contexts.
Statistical	NB	Naive bayes (Domingos and Pazzani, 1997)	The probabilistic machine learning algorithm based on Bayes' Theorem, which assumes strong (naive) independence between features. It is particularly effective for classification tasks including spam detection and sentiment analysis due to its simplicity, efficiency and ability to handle large datasets.
	LR	Logistic regression (Cox, 1958)	The statistical model widely utilized for binary classification tasks, such as predicting whether an event will occur or not. It estimates probabilities using a logistic function, making it ideal for scenarios where outcomes are categorical and decisions are probabilistic.
	PLS	Partial least squares (Wold,	An extension of the partial least squares algorithm that particularly addresses the prediction of continuous dependent variables is partial least squares regression. By finding the directions of

		1982; Wold et al., 1984)	greatest variance that closely relate independent variables to the dependent variable, it constructs predictive models. It is particularly effective when there are multicollinearity problem in data set or high dimensional settings. Therefore, it is highly applicable in both research and practical problem solving.
Tree and rule-based	CART	Classification and Regression Trees (Breiman et al., 1984)	A nonparametric decision tree learning technique that is suitable for both classification and regression tasks. It forms binary trees by partitioning the dataset into subsets based on feature values maximizing the separation of data in terms of the purity of the target variable. The CART is notorious for its interpretability and flexibility, which makes it a practical solution and a popular choice in areas where clear decision rules are required.
	C5.0	C5.0 (Quinlan, 1992; Quinlan, 1993)	An advanced decision tree algorithm that builds on predecessors like ID3 and C4.5, enhancing accuracy through boosting, winnowing, and pruning. It is widely used for classification and adapted for regression, excelling in handling large datasets and is popular for its robust performance and interpretability.
	C5.0-Rules	C5.0-rules (Quinlan, 1992; Quinlan, 1993)	C5.0-rules is a variation of the C5.0 algorithm that generates a set of decision rules rather than a tree structure, tailored for classification and adaptable for regression tasks. This approach simplifies the decision-making process by extracting the most significant rules from data, enhancing interpretability and accuracy.
	RuleFit	RuleFit (Friedman and Popescu, 2008)	A machine learning algorithm that combines decision tree-like rules with linear regression models to predict outcomes. It generates rules from an ensemble of trees and uses them as features in a linear model, effectively capturing both linear and interaction effects among variables. It is particularly valued for its interpretability and precision, making it suitable for applications in fields like healthcare and finance where understanding the model's decision process is crucial.
	BAT	Bayesian additive trees (Chipman et al., 2010)	A statistical model that uses Bayesian methods to combine multiple decision tree models for more reliable predictions. It estimates complex functions by averaging over many trees, improving accuracy and robustness while providing credible intervals for predictions. It is particularly effective in scenarios requiring careful uncertainty estimation, such as in medical prognosis and economic forecasting.
Neural network-based	MLP	Multilayer perceptron (Hornik et al., 1989)	A form of deep learning where an MLP, a type of artificial neural network, is used to classify data into distinct categories. It features multiple layers of neurons with non-linear activation functions, enabling it to capture complex patterns and relationships in data.
Spline-based	MARS	Multivariate adaptive regression spline (Friedman, 1991)	A non-parametric technique that models relationships within data by fitting piecewise linear splines, which are flexible enough to capture complex patterns. It's particularly useful in scenarios where the relationship between variables is non-linear and intricate, adjusting automatically to changes in data trends.
Kernel-based	SVM	Support Vector Machines (Vapnik et al., 1996; Schölkopf and Smola, 2002)	A powerful class of supervised learning models used for classification and regression tasks. They work by finding the hyperplane that best separates different classes in the feature space, maximizing the margin between data points of different categories. This capability to handle both linear and non-linear boundaries makes SVMs highly effective in diverse applications such as image recognition, bioinformatics, and text categorization.
Ensembles	Bag	Bagging (Breiman, 1996)	An ensemble machine learning technique used to improve the stability and accuracy of classification algorithms. It involves creating multiple versions of a predictor model by training them on different subsets of the original dataset, then aggregating their predictions to form a final verdict. It is particularly effective in reducing variance and avoiding overfitting, making it widely used

			in decision tree algorithms and complex classification tasks across various domains.
	RF	Random forests (Breiman, 2001)	An ensemble learning method that builds upon the concept of bagging by creating a multitude of decision trees at training time and outputting the class that is the mode of the classes or mean prediction of the individual trees. It enhances prediction accuracy and controls over-fitting by introducing randomness in the tree generation process through feature and data sampling.
	Boosting	Boosting (Schapire, 1990; Freund and Schapire; 1996)	An ensemble technique that aims to create a strong classifier from a number of weak classifiers. It works by sequentially applying weak models to progressively modified versions of the data, increasing the weight of misclassified instances so that subsequent models focus more on difficult cases. It has widely used variants include AdaBoost and Gradient Boosting, which are effective in reducing bias and variance in complex datasets.
Base reference	Null	Null	A simple model that provides a baseline by using no predictive information to make forecasts in statistics and machine learning. It typically predicts the most frequent category in classification tasks or the mean/median in regression tasks. This model is important for performance benchmarking, as it sets the minimal threshold that any other more complex model should exceed to be considered effective.

2. Methods

The study includes twenty-three algorithms, covering the most widely employed algorithms in the literature. The algorithms can be categorized as instance-based (KNN), statistical (Naive Bayes, Logistic Regression, Partial Least Squares), tree and rule-based (CART, C5.0, C5.0-rules, RuleFit, Bayesian Additive Trees), neural network-based (multilayer perceptron), spline-based (MARS), kernel-based (SVM) and ensemble approaches (Bagging, Boosting, Random Forests). Besides, the Null model is included in the study serving as a benchmark (base) reference as a simple and non-informative model that can be obtained without building any model. It should be noted that different base learner models are utilized in the training process of ensemble models. The C5.0 algorithm, for instance, was not only included in the study on standalone basis but was also considered as a base learner for the bagging algorithm. A similar approach has been carried out for CART, Mars, Mlp algorithms. These algorithms were used as base learner in both bagging and boosting ensemble models. The list of these algorithms and comprehensive explanations are presented in Table 1.

3. Experimental Design and Settings

3.1. Data Description and Source

The dataset was retrieved from a data science platform Kaggle (2023) which is an open source machine learning and data sharing platform. The data set includes thirty variable measurements of one hundred and forty-five students. The sequential grades of the students are considered as the target variable in the study. Variables and their characteristics can be seen in Table 2.

Due to the data set consisting almost completely of categorical data, low-frequency categories were merged to make the results more generalizable and not negatively affect the model estimation. The categories having a frequency of about ten or less were joined with the closest category. Since the target variable is multi-level and the frequency variation between levels is quite volatile (e.g., only seven students failed), we have treated grades below CC, which are defined as failing and conditionally passing, as Fail, and the remaining grades as Success. Therefore, the problem is treated as a binary classification problem. Details of these merging processes are presented in Table 2.

Table 2. Characteristics of the data set

Type	Question	Possible Answers
Personal	Age	(1: 18-21, 2: 22-25, 3: 26+)
	Sex	(1: Female, 2: Male)
	Graduated High School Type	(1: Private, 2: State, 3: Other)
	Scholarship Type	(1: None, 2: 25%, 3: 50%, 4: 75%, 5: Full) **Preprocess**: (None + 25% + 50%) as 50% and lower

	Additional Work	(1: Yes, 2: No)
	Regular Artictic or Sports Activity	(1: Yes, 2: No)
	Do you have a partner?	(1: Yes, 2: No)
	Total salary if available	(1: $135-200, 2: $201-270, 3: $271-340, 4: $341-410, 5: Above $410) **Preprocess:** ($341-410 + Above $410) as $341 and above
	Transportation to the university	(1: Bus, 2: Private Car/Taxi, 3: Bicycle, 4: Other) **Preprocess:** (Bicycle + Other) as Other
	Accommodation type in Cyprus	(1: Rental, 2: Dormitory, 3: With Family)
Family	Mother's education	(1: Primary School, 2: Secondary School, 3: High School, 4: University, 5: Msc., 6: Ph.D.) **Preprocess:** (University + Msc. + Ph.D.) as University
	Father's education	
	Number of sisters/brothers (If available)	(1: 1, 2: 2, 3: 3, 4: 4, 5: 5 or above)
	Parental status	(1: Married, 2: Divorced, 3: Died - One of Them or Both)
	Mother's occupation	(1: Retired, 2: Housewife, 3: Government Officer, 4: Private Sector Employee, 5: Self-Employment, 6: Other) **Preprocess:** (Self-Employment + Other) as Other
	Father's occupation	
Education Habits	Weekly study hours	(1: None, 2: <5 Hours, 3: 6-10 Hours, 4: 11-20 Hours, 5: More Than 20 Hours) **Preprocess:** (11-20 hours + More than 20 hours) as More than 11 hours
	Reading frequency (non-scientific books/journals)	(1: None, 2: Sometimes, 3: Often)
	Reading frequency (Scientific books/journals)	(1: None, 2: Sometimes, 3: Often)
	Attendance to the seminars/conferences related to the department	(1: Yes, 2: No)
	Impact of your projects/activities on your success	(1: Positive, 2: Negative, 3: Neutral)
	Attendance to classes	(1: Always, 2: Sometimes, 3: Never)
	Preparation to midterm exams 1	(1: Alone, 2: With Friends, 3: Not Applicable)
	Preparation to midterm exams 2	(1: Closest Date to The Exam, 2: Regularly During the Semester, 3: Never)
	Taking notes in classes	(1: Never, 2: Sometimes, 3: Always)
	Listening in classes	(1: Never, 2: Sometimes, 3: Always)
	Discussion improves my interest and success in the course	(1: Never, 2: Sometimes, 3: Always)
	Flip-classroom	(1: Not Useful, 2: Useful, 3: Not Applicable)
	Cumulative grade point average in the last semester	(1: <2.00, 2: 2.00-2.49, 3: 2.50-2.99, 4: 3.00-3.49, 5: above 3.49)
	Expected Cumulative grade point average in the graduation	
Output	Grade	(0: Fail, 1: DD, 2: DC, 3: CC, 4: CB, 5: BB, 6: BA, 7: AA) **Preprocess:** (Fail + DD + DC) as Fail; the rest of grades as Success

3.2. Preprocessing and Parameter Tuning

The preprocessing approach and experimental settings applied to the dataset before applying machine learning models can be summarized as follows:

i. The dataset was processed with one-hot encoding and label encoding for nominal and ordinal variables, respectively.

ii. Numerical variables have been standardized.

iii. The dataset is split 75% as training data and 25% as test data. As cross validation approach, the 10-fold CV method was utilized. The models were trained with the data obtained with cross-validation on the training data and their generalization performance (i.e., testing) was evaluated with the test data.

iv. The grid space approach was adopted as the model tuning parametrization. The optimal parameters were derived by using a parameter space consisting of thirty different possible values of the unique parameters of each model. The ranges and optimum values of the tuning parameters for each model are provided in detail in Table 4.

v. The test performance was extracted for each model based on the optimal parameters found by cross-validation.

For the best model among all models in the test performances, confusion matrix, roc curve and variable importance results are presented.

3.3. Performance Criteria

In classification models, depending on whether the target variable is binary or multilevel, performance criteria are primarily defined based on the confusion matrix. A classical confusion matrix can be presented as the following structure given in Table 3.

Table 3. A general representation of a confusion matrix

		Predicted	
		Positive	**Negative**
Actual (Truth)	**Positive**	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

In this study, the accuracy, area under the roc curve (AUC) and F-score, which are the most widely used measures in the literature, can be defined based on confusion matrix as follows.

- **Accuracy**:

  The percentage of correctly classified cases (including true positives and true negatives) relative to the total number of cases is defined as accuracy.

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)}$$

The model's predictive skill increases as the accuracy value converges to one.

- **Area Under the Curve (AUC)**:

  In binary classification problems, the Area Under the Receiver Operating Characteristic Curve (AUC-ROC) statistical measure is utilized to assess a model's inherent capacity to differentiate between the positive and negative classes across a range of thresholds for classification. For different threshold values, the true positive rate (sensitivity) is displayed against the false positive rate (1-specificity) through the ROC curve. An AUC value of 1.0 indicates a perfect classifier, while a value of 0.5 indicates a model that performs no better than random chance at classifying true positives and true negatives. The AUC measures the model's overall ability in performing effectively.

- **F-score (or F1-score)**: The F1-score, also known as the F-score or F-measure, is a robust metric for assessing the accuracy of a binary classification model, especially in contexts in which false positives and false negatives have different costs or when class imbalances are present. It is a harmonic mean of precision and recall. The harmonic mean, in contrast to the arithmetic mean, tends to be the lower of the two values, providing that both precision and recall are at an appropriate level. In particular, the F1-score approaches its least accurate value at 0, while reaching its best value at 1, corresponding to perfect precision and recall. The F1-score is defined by using the confusion matrix components as follows:

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

where

$$\text{Precision} = \frac{TP}{(TP+FP)}$$

$$\text{Recall} = \frac{TP}{(TP+FN)}$$

Table 4. The ranges of parameters corresponding to the each model

Model	Range	Best
Bag (C5.0)	min_n: [2, 15]	min_n: 6
Bag (CART)	tree_depth: [1, 15], min_n: [2, 15], cost_complexity: [0, 1]	tree_depth: 13, min_n: 6, cost_complexity: 3.2x10^-8
Bag (MARS)	num_terms [0, min(200, max(20, 2 * #variables)) + 1], prod_degree: [1, 2], prune_method: [backward, none, exhaustive, forward, seqrep, cv]	num_terms: 4, prod_degree: 2, prune_method: backward
Bag (MLP)	hidden_units: [2, 20] penalty: [0, 1]	hidden_units: 4 penalty: 0.00000218
BAT	trees: [10, 200] prior_terminal_node_coef: [0.01, 1] prior_terminal_node_expo: [0.01, 2]	trees: 106 prior_terminal_node_coef: 0.0928 prior_terminal_node_expo: 1.70

Boosting (C5.0)	trees: [1, 100] min_n: [2, 15] sample_size: [0.1, 1]	trees: 6 min_n: 7 sample_size: 0.969
Boosting (LightGBM)	mtry: [1, #variables] trees: [1, 2000] min_n: [2, 40] tree_depth: [1, 15] learn_rate: [-3, -0.5] loss_reduction: [-10, 1.5]	mtry: 10 trees: 1080 min_n: 2 tree_depth: 11 learn_rate: 0.00115 loss_reduction: 0.0486
Boosting (XGBoost)	mtry: [1, #variables] trees: [1, 2000] min_n: [2, 40] tree_depth: [1, 15] learn_rate: [-3, -0.5] loss_reduction: [-10, 1.5] sample_size: [0.1, 1]	mtry: 2 trees: 1212 min_n: 2 tree_depth: 5 learn_rate: 0.00990 loss_reduction: 2.62x10^-8 sample_size: 0.706
C5.0	min_n: [2, 15]	min_n: 3
C5.0 Rules	trees: [1, 100] min_n: [2, 15]	trees: 85 min_n: 3
CART	tree_depth: [1, 15], min_n: [2, 15], cost_complexity: [0, 1]	tree_depth: 5, min_n: 7, cost_complexity: 0.0000307
KNN	neighbors: [1, 20] weight_function: [cosine] dist_power: [0.1, 2]	neighbors: 14 weight_function: cosine dist_power: 1.37
Logistic Regression	none	none
MARS	num_terms [0, min(200, max(20, 2 * #variables)) + 1], prod_degree: [1, 2], prune_method: [backward, none, exhaustive, forward, seqrep, cv]	num_terms: 5, prod_degree: 2, prune_method: backward
MLP	hidden_units: [1, 10] penalty: [-10, 0] epochs: [10, 1000]	hidden_units: 2 penalty: 0.00392 epochs: 706
Naive Bayes	smoothness: [0.01, 2], Laplace: [0, 1]	smoothness: 1.33, Laplace: 0.0493
NULL	none	none
PLS	predictor_prop: [0, 1] num_comp: [2, 20]	predictor_prop: 0.0295 num_comp: 4
Random Forests	mtry: [1, 100] trees: [1, 2000] min_n: [2, 40]	mtry: 70 trees: 1648 min_n: 36
RuleFit	mtry: [0, 1] trees: [1, 100] min_n: [1, 20] tree_depth: [1, 20] learn_rate: [0, 1] loss_reduction: [0, 20] sample_size: [0, 2] penalty: [0, 1]	mtry: 0.453 trees: 8 min_n: 6 tree_depth: 6 learn_rate: 5.99*10^-8 loss_reduction: 7.92 sample_size: 0.799 penalty: 0.000883
SVM (Linear)	cost: [0, 30] margin: [0, 1]	cost: 0.244 margin: 0.177
SVM (Polynomial)	cost: [0, 30] degree: [1, 3] scale_factor: [0, 1]	cost: 5.84 degree: 2 scale_factor: 0.000605
SVM (Radial)	cost: [0, 30] rbf_sigma: [0, 1] margin: [0, 1]	cost: 20.4 rbf_sigma: 0.000467 margin: 0.178

4. Results and Discussion

This section presents the performance comparisons of the twenty-three different machine learning methods evaluated in this study. Initially, the performance of these methods on the test data according to measures such as accuracy, AUC and F-score are given in Table 5.

Table 5. The comparative test performance results of machine learning methods

Model	Accuracy	AUC	F-Score
Bag (C5.0)	0.7027	0.7500	0.7027
Bag (CART)	0.6757	0.7794	0.7143
Bag (MARS)	0.6757	0.6824	0.7273
Bagging (MLP)	0.6486	0.7676	0.6977
BAT	0.7027	0.7515	0.7442
Boosting (C5.0)	0.6757	0.7088	0.6842
Boosting (LightGBM)	**0.7568**	**0.7941**	0.7805
Boosting (XGBoost)	0.7027	0.7676	0.7442
C5.0	0.6486	0.6956	0.6667
C5.0 (Rules)	0.7297	0.7676	0.7619
CART	0.7027	0.7324	0.7556
KNN	0.6216	0.7441	0.6667
LR	0.5946	0.5676	0.6512
MARS	0.6486	0.7250	0.6486
MLP	0.6486	0.6941	0.6977
NB	0.5676	0.7500	0.7037
Null	0.5405	0.5000	0.7018
PLS	0.6216	0.7279	0.6818
RF	0.6757	0.7765	0.6842
RuleFit	0.7568	0.7500	**0.7907**
SVM (Linear)	0.5946	0.7176	0.6667
SVM (Polynomial)	0.7027	0.7412	0.7442
SVM (Radial)	0.6486	0.7088	0.6977

According to the results given in Table 5, LightGBM as a boosting algorithm provided the best results in the accuracy (0.7568) and AUC (0.7941) criteria, while RuleFit algorithm dominated in the F1-score (0.7907). It is worth to note that RuleFit algorithm yields slightly higher F1-score than LightGBM algorithm and LightGBM is the second-best algorithm in terms of this criterion. By combining these findings, it can be said that the LightGBM algorithm achieves the most generalizable and superior performance than any other algorithm. A visual interpretation of the AUC values, which are often favored in studies, is also given in Figure 1.

The null model is also included in the study to represent a reference and to clarify the necessity of complex models. In order to assess whether each model is statistically significantly different from each other, especially the null model, DeLong (Delong et al., 1988) and McNemar (McNemar, 1947) tests were performed. The DeLong test relies on AUC values to compare machine learning models, whereas the McNemar test is based on model predictions. The statistical significance value for both tests was set at 0.05 and the results are reported in Table 6.

According to the DeLong test results, all models are statistically different from the Null model, while two bagging models (with CART and MLP learners), LightGBM, XgBoost, RuleFit and SVM (Linear kernel) models have statistically different AUC values with logistic regression. Regarding the McNemar test, LightGBM, as the best model, provided statistically different predictions from MARS, RF and Bagging (C5.0 learner) models, while all model predictions were different from Null and NB models.

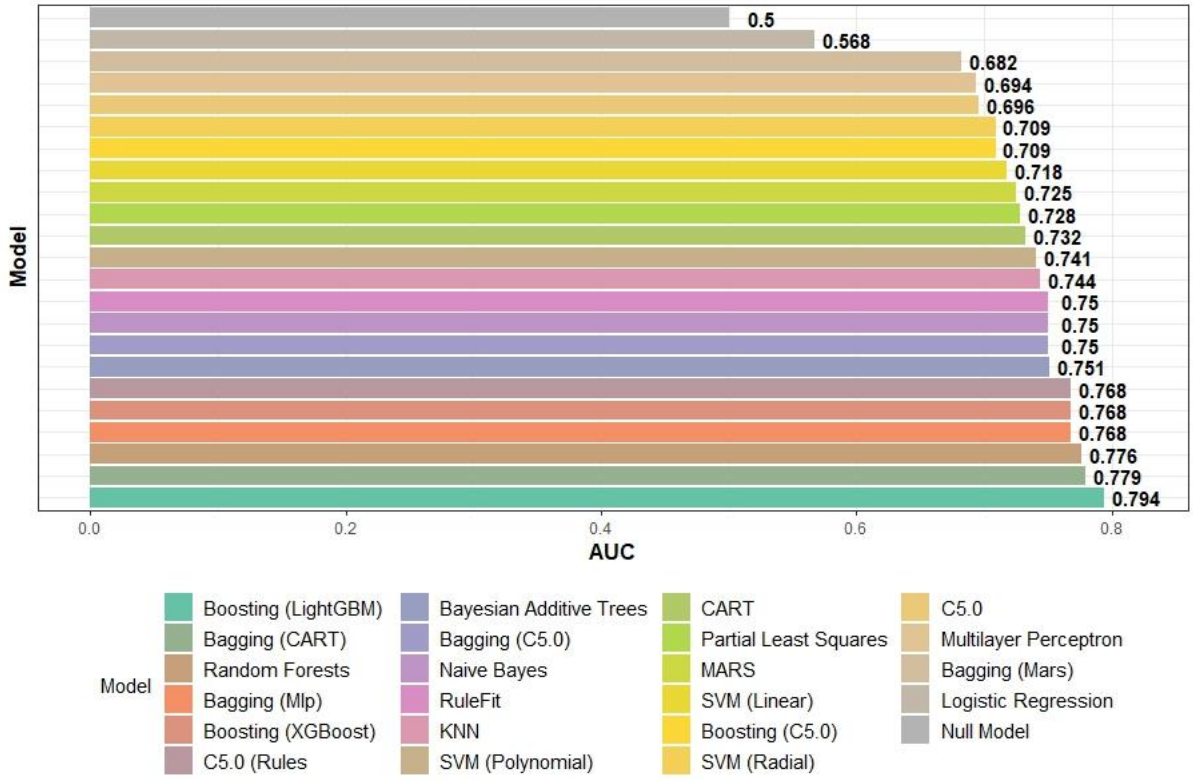

Figure 1. Visual comparison of performance results according to the AUC criterion

Table 6. The statistical comparison of each model based on AUC values and predicted categories

Model	DeLong Test	McNemar Test
Bag (C5.0)	(Null: 0.0019)	(Null: <0.001; NB: 0.0001; SVM (Linear): 0.0433; BAT: 0.0412; Bag (MARS): 0.0233; PLS: 0.0455; Boosting (XGBoost): 0.0133)
Bag (CART)	(Null: 0.0002; LR: 0.038)	(Null: 0.0003; NB: 0.0015)
Bag (Mars)	(Null: 0.0423)	(Null: 0.0009; NB: 0.0094; Bag (C5.0): 0.0233)
Bag (Mlp)	(Null: 0.0012; LR: 0.033)	(Null: 0.0005; NB: 0.0026)
BAT	(Null: 0.0019)	(Null: 0.0005; NB: 0.0026; MARS: 0.0412; Bag (C5.0): 0.0412)
Boosting (C5.0)	(Null: 0.0180)	(Null: <0.001; NB: 0.0002)
Boosting (LightGBM)	(Null: 0.0006; LR: 0.027)	(Null: 0.0015; NB: 0.0077; MARS: 0.0133; RF: 0.0233; Bag (C5.0): 0.0133)
Boosting (XGBoost)	(Null: 0.0001; LR: 0.038)	(Null: 0.0012; NB: 0.0026)
C5.0	(Null: 0.0295)	(Null: 0.0001; NB: 0.0003)
C5.0 (Rules)	(Null: 0.0007)	(Null: 0.0003; NB: 0.0015)
CART	(Null: 0.0101)	(Null: 0.0002; NB: 0.0009)
KNN	(Null: 0.0026)	(Null: 0.0003; NB: 0.0033)
LR	(Null: 0.4552)	(Null: 0.0005; NB: 0.0098)
MARS	(Null: 0.0070)	BAT: 0.0412, PLS: 0.0455; Boosting (LightGBM): 0.0133)
MLP	(Null: 0.0410)	(Null: 0.0005; NB: 0.0026)
NB	(Null: 0.0024)	(Null: 0.2482; LR: 0.0098)
Null	None	None
PLS	(Null: 0.0076)	(Null: 0.0009; NB: 0.0044; MARS: 0.0455; RF: 0.0412; Bag (C5.0): 0.0455)

RF	(Null: 0.0004)	(Null: <0.001; NB: 0.0002; PLS: 0.0412; Boosting (LightGBM): 0.0233)
RuleFit	(Null: 0.0050; LR: 0.042)	(Null: 0.0005; NB: 0.0055)
SVM (Linear)	(Null: 0.0156; LR: 0.048)	(Null: 0.0015; NB: 0.0159; Bagging (C5.0): 0.0433)
SVM (Polynomial)	(Null: 0.0047)	(Null: 0.0005; NB: 0.0026)
SVM (Radial)	(Null: 0.0228)	(Null: 0.0005; NB: 0.0026)

It is important that the models are statistically different from each other, especially from the Null model, for the generalizability and usability of the results. In this context, we focus on the predictions of the LightGBM model, which is found to be the best model, and the confusion matrix and ROC cuver derived from these predictions is given in Figure 2 and 3, respectively.

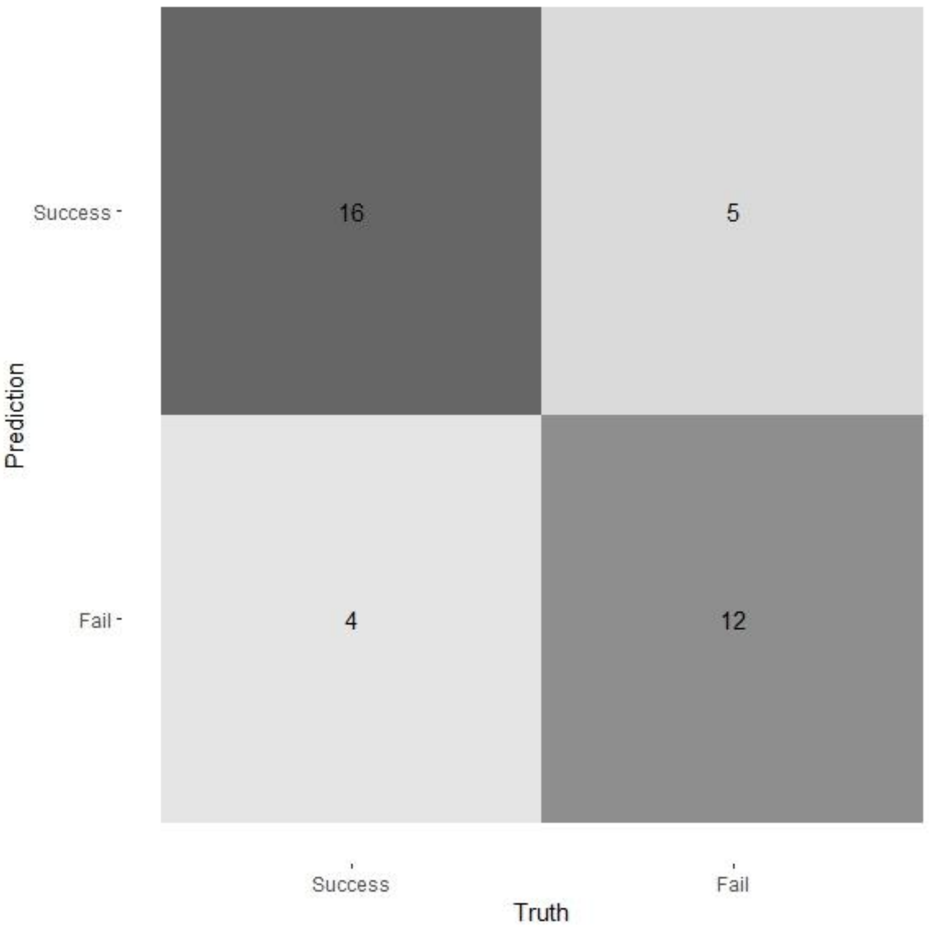

Figure 2. Confusion matrix for the best model (Boosting (LightGBM))

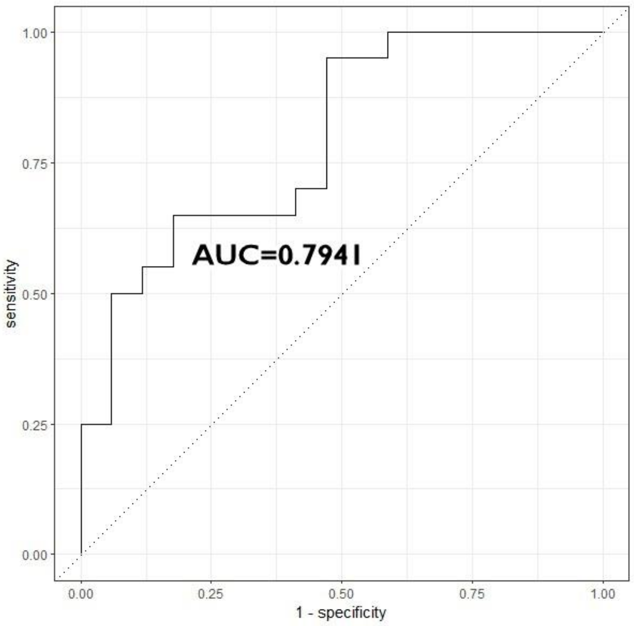

Figure 3. Roc curve for the best model (Boosting (LightGBM))

The confusion matrix and ROC curve suggest that the LightGBM model provides promising results in predicting student performance. It is critical to identify the most important factors for the performance of the model, i.e., for discriminating between success and failure. Therefore, the variable importance plot computed by using the intrinsic variable importance scores of the LightGBM model is displayed in Figure 4.

In Figure 4, the fifteen most important variables are ranked on a scale of 0-100. According to this graph, variables such as last semester GPA between 2-2.50, average income between $135-200, expected GPA between 3-3.50, gender of the student being male, number of brothers or sisters appear to be the most important variables in affecting success. Likewise, variable levels such as attending seminars etc. and not having a partner were also found to be important in model performance. The results and particularly variable importance scores presented in this study may provide a more valuable set of sociological and academic insights for researchers studying in the field of education.

It can be said that the findings of the study provide better performance by using a broader method compared to the studies in the literature such as Yılmaz and Sekeroglu (2020), Chen and Zhai (2023). In Asselman et al. (2023), the XGBoost algorithm, one of the ensemble approaches, stands out as an ensemble method and demonstrates similar performance to our study, but its shortcomings are notable in terms of variable importance and statistical significance tests. In Adejo and Connoly's (2018) study, the hybrid machine learning model also produced a competitive result and concluded that university support had a significant impact on success. In this context, it can be said that it is compatible with the scholarship status in our study.

The finding that students' performance in previous semesters has a significant effect on their future achievement is consistent with the literature (Pallathadka et al., 2021). Similarly, the findings that income and family education have a significant effect on achievement supports the results of Filho et al. (2023).

The gender variable, which was found to be relatively significant in the study, stands out as a different finding from the study of Karaboğa and Demir (2023). On the other hand, Suleiman and Anane (2022) reported that gender was a significant but low contributing variable on student achievement. As in our study, last semester GPA was considered significant in this study as well.

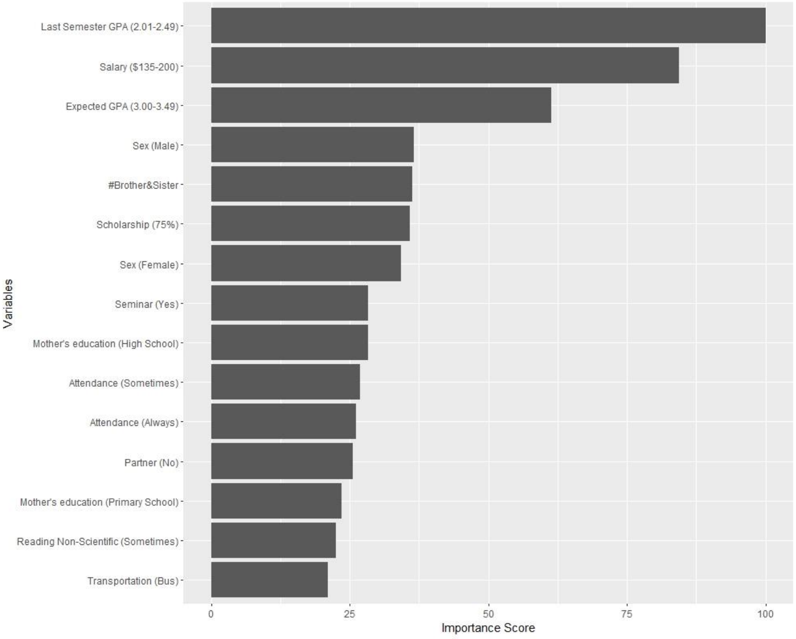

Figure 4. Variable importance plot based on Boosting (LightGBM) model.

5. Conclusion

In this study, a comprehensive comparison of the performance of machine learning methods is presented both in terms of classical metrics and statistically. Machine learning algorithms, which are widely used in the field of education as in every field, have been shared to determine the extent to which they differ statistically and the ways to determine the importance of variables rather than their performance alone. The LightGBM algorithm was ranked as the best algorithm by cross-validating twenty-three algorithms using a real dataset based on comparison them on accuracy, AUC, and F-score criteria. The results were statistically compared with two different tests to investigate the extent to which the best method differs, and it was found that LightGBM provided favorable results in this respect as well. In addition, the confusion matrix, ROC curve and variable importance plots indicated that the LightGBM algorithm offers generalizable performancealong with identifying the relative importance of the most important factors affecting student achievement.

The study is not free of limitations. First, the implementation of deep learning algorithms in such studies may provide useful insights. Furthermore, a field-based analysis of student achievement performances and the factors affecting them may provide more effective results in different subgroups. In future work, we would like to address these two limitations, and we aim to specialize the most advanced deep learning models to narrower focused educational groups.

References

Adejo, O. W., & Connolly, T. (2018). Predicting student academic performance using multi-model heterogeneous ensemble approach. Journal of Applied Research in Higher Education, 10(1), 61–75. https://doi.org/10.1108/JARHE-09-2017-0113

Alalawi, K., Athauda, R., & Chiong, R. (2023). Contextualizing the current state of research on the use of machine learning for student performance prediction: A systematic literature review. Engineering Reports, 5(12), e12699. https://doi.org/10.1002/eng2.1269

Albreiki, B., Zaki, N., & Alashwal, H. (2021). A Systematic Literature Review of Student' Performance Prediction Using Machine Learning Techniques. Education Sciences, 11(9), Article 9. https://doi.org/10.3390/educsci11090552

Asselman, A., Khaldi, M., & Aammou, S. (2023). Enhancing the prediction of student performance based on the machine learning XGBoost algorithm. Interactive Learning Environments, 31(6), 3360–3379. https://doi.org/10.1080/10494820.2021.1928235

Breiman, L. (1996). Bagging predictors. Machine learning, 24, 123-140.

Breiman, L. (2001). Random forests. Machine learning, 45, 5-32.

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. Classification and Regression Trees (CART). 1984. Belmont, CA, USA: Wadsworth International Group.

Chen, Y., & Zhai, L. (2023). A comparative study on student performance prediction using machine learning. Education and Information Technologies, 28(9), 12039–12057. https://doi.org/10.1007/s10639-023-11672-1

Chipman, H. A., George, E. I., & McCulloch, R. E. (2010). BART: Bayesian additive regression trees.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. IEEE transactions on information theory, 13(1), 21-27.

Cox, D. R. (1958). The regression analysis of binary sequences. Journal of the Royal Statistical Society: Series B (Methodological), 20(2), 215–232.

DeLong, E. R., DeLong, D. M., & Clarke-Pearson, D. L. (1988). Comparing the areas under two or more correlated receiver operating characteristic curves: a nonparametric approach. Biometrics, 837-845.

Deo, R. C., Yaseen, Z. M., Al-Ansari, N., Nguyen-Huy, T., Langlands, T. A. M., & Galligan, L. (2020). Modern Artificial Intelligence Model Development for Undergraduate Student Performance Prediction: An Investigation on Engineering Mathematics Courses. IEEE Access, 8, 136697–136724. https://doi.org/10.1109/ACCESS.2020.3010938

Domingos, P., & Pazzani, M. (1997). On the optimality of the simple Bayesian classifier under zero-one loss. Machine learning, 29, 103-130.

Elbadrawy, A., Polyzou, A., Ren, Z., Sweeney, M., Karypis, G., & Rangwala, H. (2016). Predicting Student Performance Using Personalized Analytics. Computer, 49(4), 61–69. https://doi.org/10.1109/MC.2016.119

Filho S., , R. L. C., Brito, K., & Adeodato, P. J. L. (2023). A data mining framework for reporting trends in the predictive contribution of factors related to educational achievement. Expert Systems with Applications, 221, 119729.

Freund, Y., & Schapire, R. E. (1996, July). Experiments with a new boosting algorithm. In icml (Vol. 96, pp. 148-156).

Friedman, J. H. (1991). Multivariate adaptive regression splines. The annals of statistics, 19(1), 1-67.

Friedman, J. H., & Popescu, B. E. (2008). Predictive learning via rule ensembles.

Gamulin, J., Gamulin, O., & Kermek, D. (2016). Using Fourier coefficients in time series analysis for student performance prediction in blended learning environments. Expert Systems, 33(2), 189–200. https://doi.org/10.1111/exsy.12142

Guan, C., Mou, J., & Jiang, Z. (2020). Artificial intelligence innovation in education: A twenty-year data-driven historical analysis. International Journal of Innovation Studies, 4(4), 134–147. https://doi.org/10.1016/j.ijis.2020.09.001

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. Neural networks, 2(5), 359-366.

Hussain, M., Zhu, W., Zhang, W., Abidi, S. M. R., & Ali, S. (2019). Using machine learning to predict student difficulties from learning session data. Artificial Intelligence Review, 52(1), 381–407. https://doi.org/10.1007/s10462-018-9620-8

Karaboğa, H. A., & Demir, I. (2023). Examining the factors affecting students' science success with Bayesian networks. International Journal of Assessment Tools in Education, 10(3), 413-433.

Liu, J., Loh, L., Ng, E., Chen, Y., Wood, K. L., & Lim, K. H. (2020). Self-Evolving Adaptive Learning for Personalized Education. Conference Companion Publication of the 2020 on Computer Supported Cooperative Work and Social Computing, 317–321. https://doi.org/10.1145/3406865.3418326

McNemar, Q. (1947). Note on the sampling error of the difference between correlated proportions or percentages. Psychometrika, 12(2), 153-157.

Pallathadka, H., Wenda, A., Ramirez-Asís, E., Asís-López, M., Flores-Albornoz, J., & Phasinam, K. (2023). Classification and prediction of student performance data using various machine learning algorithms. Materials Today: Proceedings, 80, 3782–3785. https://doi.org/10.1016/j.matpr.2021.07.382

Quinlan, J. R. (1992). Learning with continuous classes. In 5th Australian joint conference on artificial intelligence (Vol. 92, pp. 343-348).

Quinlan, J. R. (1993). Combining instance-based and model-based learning. In Proceedings of the tenth international conference on machine learning (pp. 236-243).

Sağlam, A. Ç., & Aydoğmuş, M. (2016). Gelişmiş ve Gelişmekte Olan Ülkelerin Eğitim Sistemlerinin Denetim Yapıları Karşılaştırıldığında Türkiye Eğitim Sisteminin Denetimi Ne Durumdadır? Uşak Üniversitesi Sosyal Bilimler Dergisi, 9(1), 17–38. https://dergipark.org.tr/en/pub/usaksosbil/issue/21662/232993

Schapire, R. E. (1990). The strength of weak learnability. Machine learning, 5, 197-227.

Schölkopf, B., & Smola, A. J. (2002). Learning with kernels: support vector machines, regularization, optimization, and beyond. MIT press.

Sekeroglu, B., Abiyev, R., Ilhan, A., Arslan, M., & Idoko, J. B. (2021). Systematic Literature Review on Machine Learning and Student Performance Prediction: Critical Gaps and Possible Remedies. Applied Sciences, 11(22), Article 22. https://doi.org/10.3390/app112210907

Students Performance. (2023) Retrieved 25 September 2023, from https://www.kaggle.com/datasets/joebeachcapital/students-performance

Suleiman, R., & Anane, R. (2022). Institutional Data Analysis and Machine Learning Prediction of Student Performance. 2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design (CSCWD), 1480–1485. https://doi.org/10.1109/CSCWD54268.2022.9776102

Tilahun, L. A., & Sekeroglu, B. (2020). An intelligent and personalized course advising model for higher educational institutes. SN Applied Sciences, 2(10), 1635. https://doi.org/10.1007/s42452-020-03440-4

Tran, T.-O., Dang, H.-T., Dinh, V.-T., Truong, T.-M.-N., Vuong, T.-P.-T., & Phan, X.-H. (2017). Performance Prediction for Students: A Multi-Strategy Approach. Cybernetics and Information Technologies, 17(2), 164–182. https://doi.org/10.1515/cait-2017-0024

Vapnik, V., Golowich, S., & Smola, A. (1996). Support vector method for function approximation, regression estimation and signal processing. Advances in neural information processing systems, 9.

Wold, H. (1982). Soft modelling: the basic design and some extensions. Systems under indirect observation, Part II, 36-37.

Wold, S., Ruhe, A., Wold, H., & Dunn, Iii, W. J. (1984). The collinearity problem in linear regression. The partial least squares (PLS) approach to generalized inverses. SIAM Journal on Scientific and Statistical Computing, 5(3), 735-743.

Wu, Z., He, T., Mao, C., & Huang, C. (2020). Exam paper generation based on performance prediction of student group. Information Sciences, 532, 72–90. https://doi.org/10.1016/j.ins.2020.04.043

Yousafzai, B. K., Hayat, M., & Afzal, S. (2020). Application of machine learning and data mining in predicting the performance of intermediate and secondary education level student. Education and Information Technologies, 25(6), 4677–4697. https://doi.org/10.1007/s10639-020-10189-1