



Machine Learning and Deep Learning-Based Students' Grade Prediction

Adil Korchi¹ · Fayçal Messaoudi² · Ahmed Abatal³ · Youness Manzali⁴

Received: 21 July 2023 / Accepted: 13 October 2023 / Published online: 31 October 2023
© The Author(s), under exclusive licence to Springer Nature Switzerland AG 2023

Abstract

Predicting student performance in a curriculum or program offers the prospect of improving academic outcomes. By using effective performance prediction methods, instructional leaders can allocate adequate resources and instruction more accurately. This paper aims to identify machine learning algorithm features for predicting student grades as an early intervention. Predictive models spot at-risk students early, allowing educators to provide timely support. Educators can customize teaching methods, and these models assess program success, helping institutions refine or expand them through data-driven decisions. But the problem definition of student grade prediction is to develop predictive models or algorithms that can forecast or estimate the future academic performance or grades of students based on various input features and historical data, and to do so, we utilized a student dataset comprising personal information and grades, employing various regression algorithms, including decision tree, random forest, linear regression, k-nearest neighbor, XGBoost, and deep neural network. We chose these algorithms for their suitability and distinct strengths. We assessed their performance using determination coefficient, mean average error, mean squared error, and root mean squared error. The results showed that the deep neural network outperformed others with a determination coefficient of 99.97%, confirming its reliability in predicting student grades and assessing performance, and this will certainly help to develop predictive models that can accurately forecast or estimate students' academic performance based on various input features and enable teaching staff to provide timely assistance in addressing these issues.

Keywords Machine learning · Predicting students' grades · Deep neural network · Regression · Supervised learning

1 Introduction

Predicting students' marks is a common problem in educational data mining, with applications in areas such as student assessment, course design, and academic advising [1]. Machine learning techniques, such as linear regression and neural

networks, can be used to build predictive models that can estimate students' marks based on various factors, such as past grades, attendance, and test scores. These models can help educators to identify at-risk students, design personalized learning interventions, and provide feedback to students on their progress [2].

The primary goal of this endeavor is to harness the predictive capabilities of machine learning to improve educational outcomes. By utilizing historical data and the inherent patterns within it, the machine learning algorithms enable educators and institutions to identify students who may be at risk of underperforming or dropping out. This early intervention can significantly impact a student's academic journey, providing timely support and personalized learning experiences.

One of the key advantages of using machine learning for predicting students' marks is its ability to analyze and consider a multitude of variables simultaneously. In traditional assessment methods, educators might rely solely on a student's performance in a single exam or assignment. However, machine learning models can take into account a wide range of factors that influence academic performance, including study habits, socioeconomic background, and even extracurricular activities. This holistic approach to prediction can offer a more comprehensive view of the educational path of a student.

Moreover, these predictive models can adapt and improve over time as more data becomes available. This adaptability is particularly valuable in the dynamic field of education, where student demographics, teaching methods, and curricula can change from year to year. By continuously training and refining these models, educators can stay ahead of the curve and make data-driven decisions to enhance the learning experience for their students, and there are various approaches that can be used to predict students' performance with machine learning algorithms, but they generally involve the following steps:

- **Collect and prepare the data:** this involves collecting the relevant data from students' records, such as their past grades, attendance, and test scores. To ensure that the data is in an acceptable format for the machine learning method, it should be cleaned and preprocessed.
- **Choose a machine learning model:** for this purpose, a variety of machine learning models, including neural networks, decision trees, and linear regression, can be utilized. The choice of model will depend on the characteristics of the data and the specific goals of the prediction.
- **Train the model:** once the model has been chosen, it needs to be trained on the data. This involves feeding the model a large number of examples and changing the model's parameters to reduce the difference in scores between predictions and actual results.
- **Evaluate the model:** after the model has been trained, it is important to evaluate its performance to determine how well it is able to predict students' marks. This can be done using techniques such as cross-validation, where the model is tested on a portion of the data that was not used for training.
- **Make predictions:** the model can be used to make predictions on new data after it has been trained and assessed.

In this paper, we will explore different methodologies for predicting students' grades through the application of machine learning. These methodologies

encompass aspects such as data collection and preprocessing, model selection, model training, evaluation, and prediction.

The organization of this paper is as follows: Sect. 2 delves into pertinent research in the field, Sect. 3 enumerates the machine learning techniques utilized, Sect. 4 outlines the adopted methodology, Sect. 5 presents the results of the predictive models employed, Sect. 6 contains a discussion regarding the rationale behind our selection of the mentioned algorithms and the distinct qualities that set these algorithms apart from others, influencing our choice for the proposed work. Section 7 presents the final thoughts and conclusions of the research.

2 Related Work

Several studies have explored the use of machine learning for predicting student grades. One widely used approach is linear regression, which is a statistical method for finding the linear relationship between a dependent variable (in this case, student grades) and one or more independent variables (such as attendance, test scores, and other factors). Linear regression has been shown to be effective in predicting student grades in a number of studies [3–6].

Another popular approach for predicting student grades is the use of decision tree algorithms, which build a tree-like model of decisions based on the data. Decision trees have been used to predict student grades in a number of studies [7–9] and have demonstrated their effectiveness in performing this specific task.

In addition to linear regression and decision trees, other machine learning algorithms that have been used for predicting student grades include k-nearest neighbor (k-NN) [10, 11] and random forests [12]. These approaches have also been shown to be effective for this task, although they may have different strengths and limitations depending on the specific characteristics of the data and the goals of the prediction.

To predict students' performance based on the use of the internet as a learning resource and the impact of the time spent by students on social networks, the authors of a study [13] used a variety of machine learning algorithms, including decision trees, naïve Bayes, artificial neural networks (ANN), and logistic regression. They discovered that the ANN model, which had an accuracy of about 80%, performed the best.

The BiLSTM deep neural network model was employed by the authors in [14] coupled with an attention mechanism model, to predict students' grades from historical data. The results showed that the BiLSTM combined with the attention mechanism yielded a better accuracy of 90.16%.

In another study [15], the authors applied a deep learning model to predict students' academic performance. They employed a data set containing different variables such as demographic, social, educational, and student grades. They used the synthetic minority oversampling (SMOTE) technique to overcome the data imbalance problem. Their proposed solution resulted in approximately 96% accuracy for grade predictions across courses.

Sekeroglu et al. [16] looked into two data sets to predict and categorize student performance using several machine learning techniques, such as backpropagation,

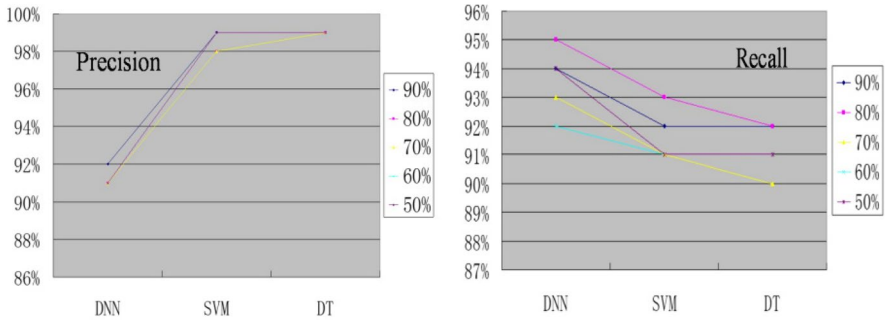


Fig. 1 The precision and recall of the DT, SVM, and DNN algorithms in [19]

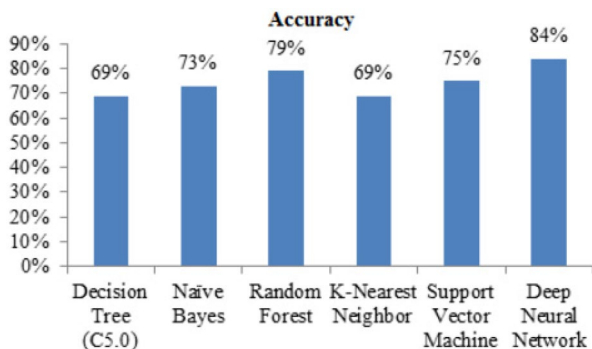
long-short term memory, support vector regression, and for classification, gradient boosting classifier. As a result, the support vector regression model outperformed the other algorithms at the R-squared score of 83% in grade prediction, and for classification, the backpropagation model performed the best with an accuracy equal to 87%.

The purpose of [19] is to enhance online teaching quality by predicting student pass rates, improving academic performance, and strengthening online education management. Researchers have used machine learning to forecast pass rates and identify key student factors impacting learning. However, they have not developed an online education-specific pass rate prediction model or introduced deep neural network (DNN) algorithms.

The study establishes a pass rate prediction feature model for online education, optimizing decision tree (DT) and support vector machine (SVM) algorithms using grid search. It compares these with DNN and finds DNN more complex with lower interpretability, while DT and SVM are simpler. Figure 1 illustrates that all three algorithms perform well at different feature model partition ratios, but DNN excels.

In [22], the author introduced a student performance prediction system based on deep neural network (DNN). They conducted training and testing on a Kaggle dataset, employing various algorithms including decision tree, naïve Bayes, random forest, support vector machine, k-nearest neighbor, and DNN using R Programming. The comparison of algorithm accuracies revealed that DNN achieved the highest accuracy at 84%, as depicted in Fig. 2.

Fig. 2 Accuracy comparison graph of 6 machine learning algorithms



2.1 Comparison of Some Similar Works

The following table compares some similar works and provides information on the methodologies used as well as the advantages and disadvantages of this research.

3 Machine Learning Model Used

The choice of the best algorithm depends on the specific dataset and problem at hand. It is often a good practice to experiment with multiple algorithms, tune their hyperparameters, and evaluate their performance using appropriate metrics to determine which one works best for your particular use case. Additionally, feature engineering and data preprocessing play a significant role in improving prediction accuracy.

In the context of predicting student grades, the selection of machine learning algorithms is driven by the nature of educational datasets and the objectives of the prediction task. In general, there are not specific algorithms that you absolutely cannot use, but there are some that may not be well-suited or are generally not recommended due to various reasons.

Decision trees and random forests are often preferred for their ability to handle diverse data types, capture non-linear relationships, and mitigate overfitting. Linear regression is chosen when the assumption of a linear relationship between features and grades is reasonable and when interpretability is paramount. K-nearest neighbors can be effective in identifying students with similar characteristics who tend to achieve similar grades. XGBoost is a favored choice for large and high-dimensional datasets, offering robustness and the ability to capture complex interactions. Deep neural networks come into play when complex, non-linear relationships or unstructured data are involved. The ultimate selection depends on factors such as the prediction goal, dataset characteristics, and the trade-offs between interpretability and predictive accuracy. Hence, the algorithms we have opted for and due to their demonstrated effectiveness in numerous research investigations in the field of student grade prediction are as follows:

- Decision tree regressor

A decision tree is a machine learning method used to categorize data or make predictions based on the answers provided to a series of previous questions. This model is a type of supervised learning, which means that it is trained and tested on a data set with the required categorization. It is a graphical representation that provides all possible solutions to a problem from given conditions.

- Random forest regressor

A supervised learning technique [17, 18] called a random forest employs an ensemble learning approach for regression. It is a meta-estimator that employs the mean to increase prediction accuracy and reduce overfitting. It does this by fitting a number of classification decision trees to different subsamples of the data set [19].

– Linear regression

Linear regression is a popular statistical method used for modeling the relationship between a dependent variable and one or more independent variables. It is a simple but powerful technique that assumes a linear relationship between the variables; its purpose is to solve regression problems. Regression builds a target prediction value on a set of independent variables. Linear regression is principally employed to find the relationship between variables and predictions.

– K-nearest neighbors regressor

K-nearest neighbors (k-NN) regressor is a type of supervised learning algorithm used for regression tasks. It works on the principle of finding the k-nearest data points to a new, unseen data point and using their target values (dependent variable) to predict the value for the new data point. Here is how the k-NN regressor algorithm works:

- Training: during the training phase, the algorithm stores the feature vectors and their corresponding target values (dependent variable) from the training dataset.
- Prediction: when given a new, unseen data point for which we want to predict the target value.

– XGBoost regressor

XGBoost is a supervised machine learning algorithm used on large data sets. It is an accurate implementation of gradient boosting which can be applied to predictive modeling by regression.

– Deep neural network

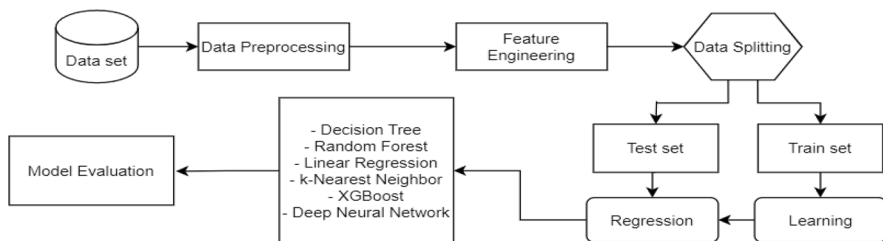


Fig. 3 The study's pipeline

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score
0	female	group B	bachelor's degree	standard	none	72	72	74
1	female	group C	some college	standard	completed	69	90	88
2	female	group B	master's degree	standard	none	90	95	93
3	male	group A	associate's degree	free/reduced	none	47	57	44
4	male	group C	some college	standard	none	76	78	75
5	female	group B	associate's degree	standard	none	71	83	78
6	female	group B	some college	standard	completed	88	95	92
7	male	group B	some college	free/reduced	none	40	43	39
8	male	group D	high school	free/reduced	completed	64	64	67
9	female	group B	high school	free/reduced	none	38	60	50
10	male	group C	associate's degree	standard	none	58	54	52

Fig. 4 Data set used

A deep neural network is characterized by a particularity that is composed of an input layer, an output layer, and at least 3 layers in between of interconnected nodes, or “neurons.” This allows it to process data in a complex way, using advanced mathematical models. Each of these layers performs different types of sorting and specific categorization in a process called feature hierarchy.

4 Methodology

The methodology of this study includes the following steps, which are summarized in the figure below (Fig. 3).

4.1 Data Set Description

The goal of this study is to predict students’ total scores using techniques of machine learning and deep learning. To achieve this, we used a data set containing information on various 1000 student characteristics, including gender of education level of parents, lunch, and exam preparation courses. In addition, the data set included scores on math, reading, and writing exams as shown in Figs. 4 and 5.

Fig. 5 Data set information

	math score	reading score	writing score
count	1000.00000	1000.000000	1000.000000
mean	66.08900	69.169000	68.054000
std	15.16308	14.600192	15.195657
min	0.00000	17.000000	10.000000
25%	57.00000	59.000000	57.750000
50%	66.00000	70.000000	69.000000
75%	77.00000	79.000000	79.000000
max	100.00000	100.000000	100.000000

	0	1	2
gender_male	0	0	0
race/ethnicity_group B	1	0	1
race/ethnicity_group C	0	1	0
race/ethnicity_group D	0	0	0
race/ethnicity_group E	0	0	0
parental level of education_bachelor's degree	1	0	0
parental level of education_high school	0	0	0
parental level of education_master's degree	0	0	1
parental level of education_some college	0	1	0
parental level of education_some high school	0	0	0
lunch_standard	1	1	1
test preparation course_none	1	0	1

Fig. 6 Categorical variables conversion

4.2 Data Cleaning and Preprocessing

The first step in our analysis was to clean and preprocess the data. This included handling missing values, converting categorical variables to numeric form (Fig. 6), and scaling the data to ensure that all variables were on the same scale.

4.3 Feature Engineering

To improve the performance of the machine learning algorithms, we performed feature engineering on the data set. This involved selecting the most relevant features and creating new feature: total score, the target variable, by combining the existing ones (math, reading, and writing scores) as shown below (Fig. 7). Feature engineering seeks to produce more robust and predictive data set for the machine learning algorithms (Fig. 8).

	gender	race/ethnicity	parental level of education	lunch	test preparation course	math score	reading score	writing score	Total score
0	female	group B	bachelor's degree	standard	none	72	72	74	218
1	female	group C	some college	standard	completed	69	90	88	247
2	female	group B	master's degree	standard	none	90	95	93	278
3	male	group A	associate's degree	free/reduced	none	47	57	44	148
4	male	group C	some college	standard	none	76	78	75	229
5	female	group B	associate's degree	standard	none	71	83	78	232
6	female	group B	some college	standard	completed	88	95	92	275
7	male	group B	some college	free/reduced	none	40	43	39	122
8	male	group D	high school	free/reduced	completed	64	64	67	195
9	female	group B	high school	free/reduced	none	38	60	50	148
10	male	group C	associate's degree	standard	none	58	54	52	164

Fig. 7 Creation of the target variable "total score"

	0	1	2
gender_male	0	0	0
race/ethnicity_group B	1	0	1
race/ethnicity_group C	0	1	0
race/ethnicity_group D	0	0	0
race/ethnicity_group E	0	0	0
parental level of education_bachelor's degree	1	0	0
parental level of education_high school	0	0	0
parental level of education_master's degree	0	0	1
parental level of education_some college	0	1	0
parental level of education_some high school	0	0	0
lunch_standard	1	1	1
test preparation course_none	1	0	1
math score	72	69	90
reading score	72	90	95
writing score	74	88	93
Total score	218	247	278

Fig. 8 Final data set

4.4 Data Visualization

To enhance comprehension and interpretation of information, we opt for representing our processed data through graphical elements such as charts and visual displays. Utilizing visual representations like graphs is an effective means of unveiling patterns and trends within complex data, ultimately aiding in simplifying information and facilitating more informed decision-making.

We proceeded to visualize the dataset, aiming to gain a deeper understanding of its contents and explore relationships among various variables. Our goal was to detect any discernible patterns or trends within the data. This helped us to identify the most important features for predicting students' total scores. Figures 9, 10, 11, and 12 represent graphically the different variables.

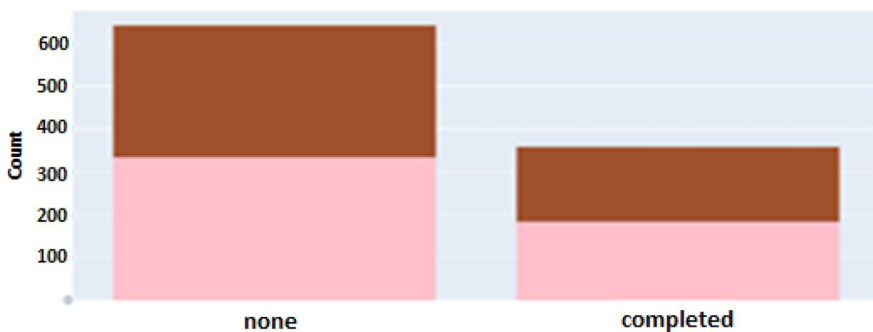
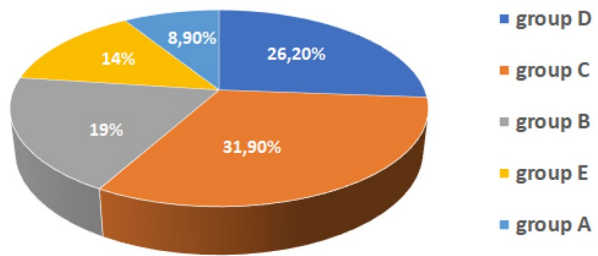


Fig. 9 Distribution of the variable “gender” according to “test preparation course” (completed or not)

Fig. 10 Distribution of the variable “race/ethnicity”



A correlation study is a statistical analysis that measures the relationship between two or more variables. It is used to understand how the values of one variable are affected by changes in the values of another variable. The degree and direction of the linear link between variables is measured by the Pearson correlation coefficient. The Pearson correlation coefficient, often denoted as “ r ” or the Pearson’s r , is a statistical measure that quantifies the strength and direction of a linear relationship between two continuous variables. It assesses how closely the data points in two datasets or variables cluster around a straight line. Its value falls between -1 and 1 , with -1 denoting a high negative correlation, 0 denoting no correlation, and 1 denoting a significant positive correlation. To perform a correlation study in Python, the `corr()` method of a Pandas DataFrame or the `pearsonr()` function from the `scipy.stats` module can be used.

Upon data processing, we acquire the correlation depicted in Fig. 13. It becomes evident that a robust relationship exists among the variables: math score, reading score, writing score, and the target variable total score.

4.5 Data Splitting

To ensure the validity of our results, we used a 30/70 ratio to divide the data set into training and testing sets. The testing set was used to gauge how well the models performed, while the training set was used to train the machine learning algorithms.

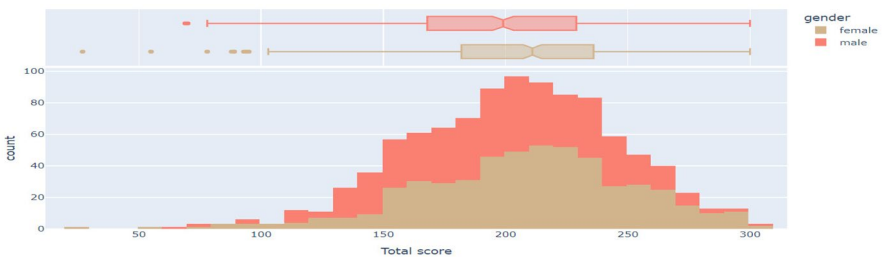


Fig. 11 Distribution by gender of the total score

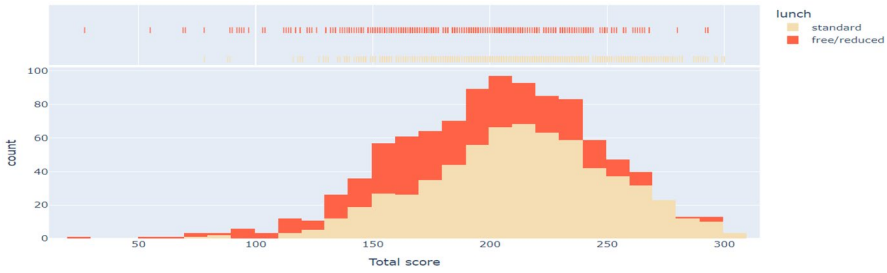


Fig. 12 Distribution of total scores by lunch type

4.6 Motivation

We are motivated to use regression models to predict student performance since there are valuable tools in education for the following reasons:

- Identifying influential factors: regression helps pinpoint factors like attendance and socioeconomic status that affect student performance.
- Data-driven decisions: schools use regression to analyze vast data sets, enabling informed decisions to enhance teaching methods and support systems.
- Early intervention: regression identifies at-risk students, allowing early intervention and support, preventing long-term academic struggles.
- Efficient resource allocation: schools optimize limited resources by focusing on areas identified by regression models, ensuring maximum impact.

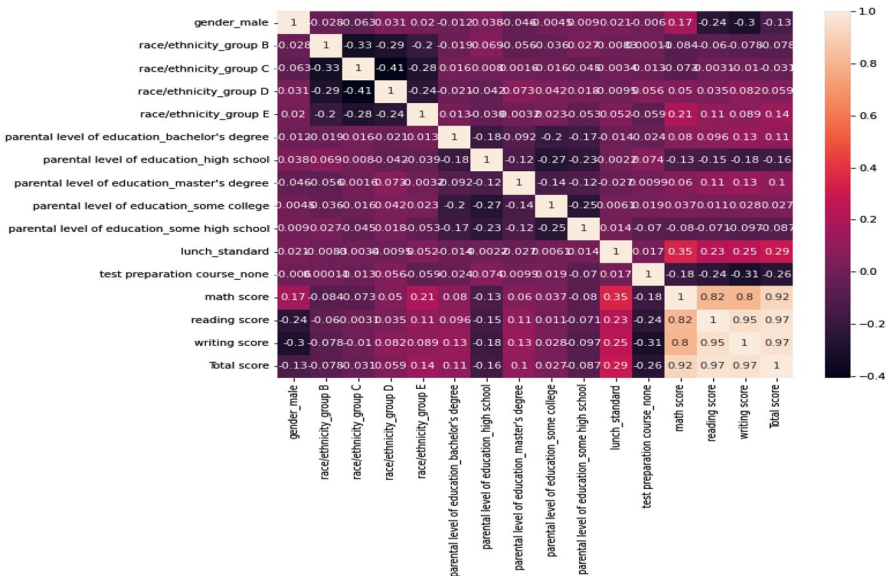


Fig. 13 Correlation study

- Policy assessment: policymakers assess existing policies' effectiveness using regression, guiding adjustments for improved education systems.
- Personalized learning: regression tailors teaching methods based on individual student factors, enhancing engagement and achievement.
- Continuous improvement: regression supports ongoing research, enabling the testing of new hypotheses and contributing to the evolution of effective educational practices.

4.7 Machine Learning Algorithm Implementation

The implementation of machine learning algorithms is driven by the desire to leverage data to automate tasks, make predictions, gain insights, and solve complex problems across a wide range of domains and applications. It empowers organizations and individuals to extract value from data and make data-informed decisions and that is why we implemented a range of machine learning algorithms for predicting student grades, including decision trees, random forests, linear regression, k-nearest neighbor, XGBoost, and deep neural networks [20].

The procedures listed below were used to implement these algorithms using the scikit-learn library and the Python programming language.

- Create an instance of the algorithm class: each algorithm is implemented using a corresponding class from a library such as scikit-learn. To create an instance of the class, we need to call the class with any desired hyperparameters (the `random_state` parameter is set to ensure that the results are reproducible).
- Fit the model to the training data: the `fit()` method is used to fit the model after it has been generated to the training set of data. This method takes the training data and target variables as inputs and adjusts the model's internal parameters to fit the data.
- Repeat the process for each algorithm: the process of creating and fitting the model is repeated for each algorithm. After all the algorithms are trained, they can be used to make predictions on the testing data.

4.8 Deep Neural Network Implementation

The DNN used was built using the Keras library in Python using the following steps:

The first step is to create a model object using the `sequential` class. This creates a model that is a linear stack of layers, where the input goes through each layer sequentially and the output of one layer is the input of the next layer.

Next, the layers are added to the model using the `model.add()` method. The model has 15 layers, each with a specified number of neurons and an activation function. The activation function determines the output of a neuron given an input or set of inputs. In this case, the `relu` activation function is used for all but the output layer, which uses a linear activation function.

After the layers are added, the model is compiled using the `model.compile()` method. This step specifies the optimizer, loss function, and metrics that will

be used to train the model. The Adam optimizer is used, and the loss function is the mean squared error (MSE). The MAE, MSE, and RMSE metrics are also used to evaluate the model's performance.

Finally, the model is trained using the `model.fit()` method, which takes the training data and target variables as inputs and trains the model for the given number of epochs. The model is trained for 100 epochs. The model is also evaluated on the testing data using the validation data parameter.

5 Model Evaluation and Results

To assess and contrast how well the machine learning algorithms work, we used regression plots of the models as well as several metrics, including the R-squared score, mean absolute error (MAE), mean squared error (MSE), and root mean squared error (RMSE).

5.1 Regression Plots

A regression plot is a valuable tool in statistical analysis and modeling. It is a scatter plot that illustrates the connection between two variables and the fitted line or curve that represents the model's predictions. It is a useful tool for visualizing the performance of a machine learning model and identifying trends and patterns in the data.

We used regression plots to visualize the predictions made by each model based on the test set (`y_test`). These plots showed the relationship between expected and actual values, allowing us to see how well each model was able to accurately predict the total scores of students as shown in the following

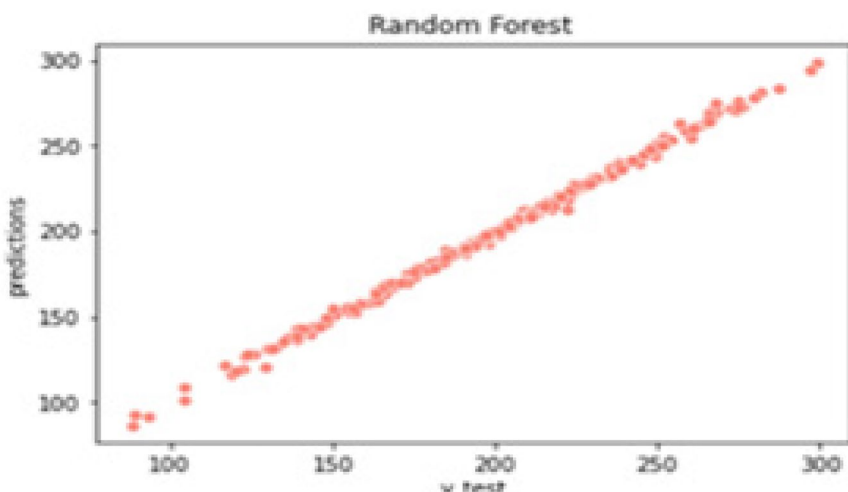


Fig. 14 Random forest regression plot

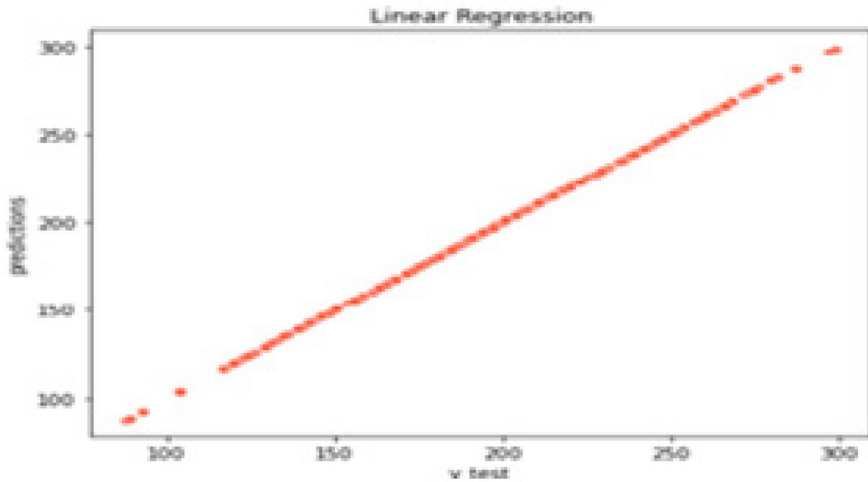


Fig. 15 Linear regression regression plot

figures (Figs. 14, 15, 16, 17, 18, 19). The regression graphs can be created in Python by the `regplot()` function of the Seaborn library.

Based on the regression plots of the models used, we can see that the DNN model fits the data better compared to the other machine learning algorithm used.

5.2 Performance Metrics

Performance metrics play a fundamental role in assessing, improving, and optimizing performance across a wide range of domains. They provide a structured and measurable way to evaluate, compare, and make decisions, ultimately leading to better outcomes, increased efficiency, and informed actions.

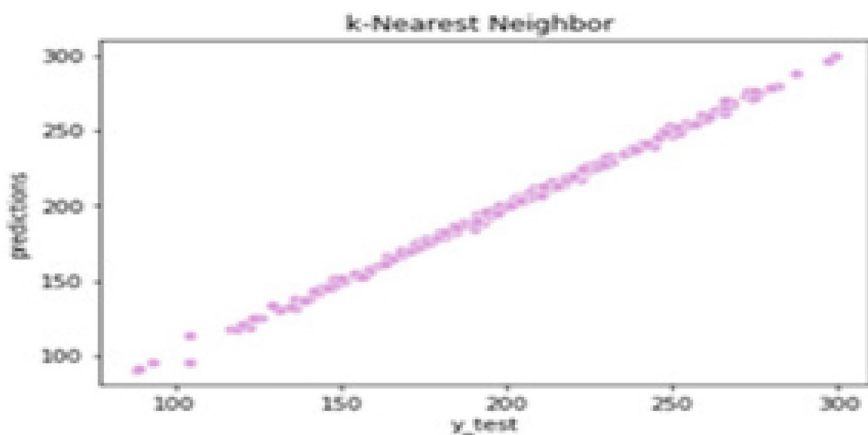


Fig. 16 k-nearest neighbors regression plot

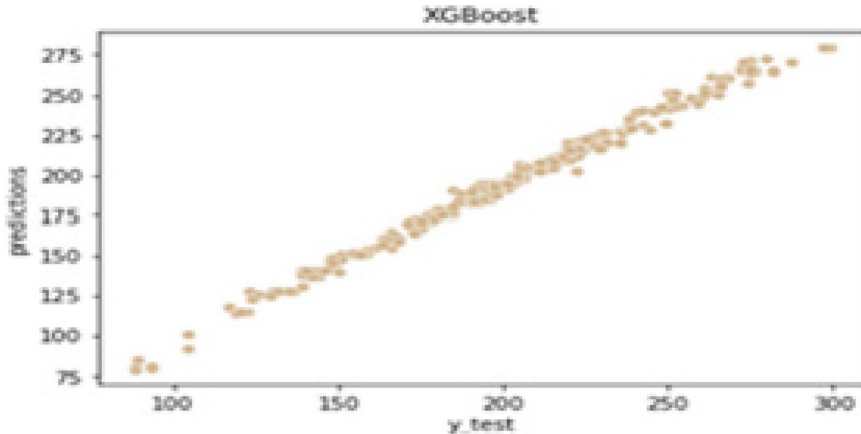


Fig. 17 XGBoost regression plot

We selected the following metrics to determine which of the models exhibits superior performance.

- R-squared score: this metric gauges how much of the target variable's variance the model is able to account for. An improved fit is indicated by a higher R-squared value.
- Mean absolute error (MAE): it is the average absolute difference between the values that were predicted and the actual values measured by the mean absolute error (MAE). Better fit is indicated by a lower MAE.
- Mean squared error (MSE): the average squared difference between the anticipated values and the actual values measured by the mean squared error (MSE). Better fit is indicated by a lower MSE.

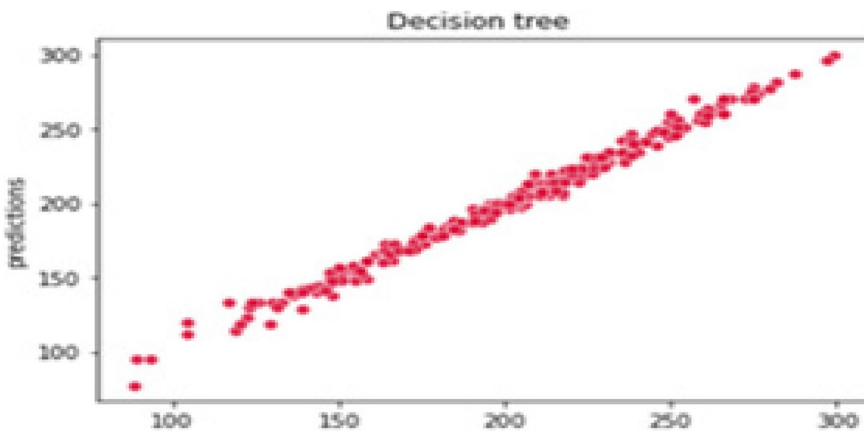


Fig. 18 Decision tree regression plot

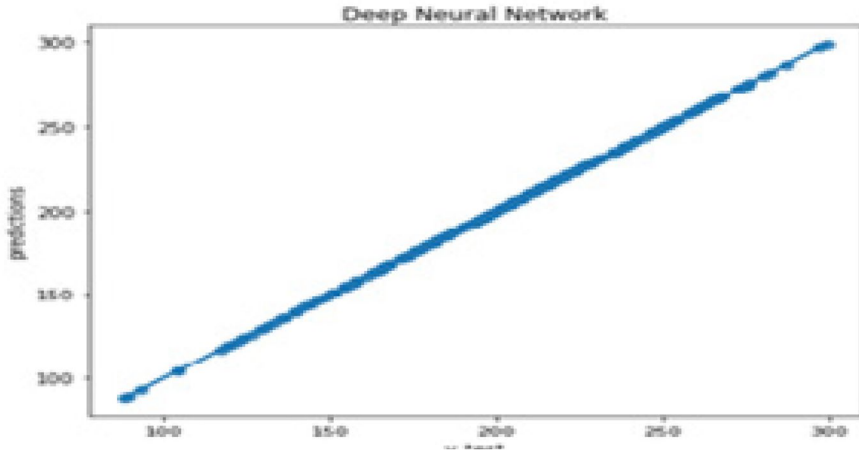


Fig. 19 Deep neural network regression plot

- **Root mean squared error (RMSE):** this measure is used to quantify the error in the same units as the target variable and is the square root of the MSE. Better fit is indicated by a lower RMSE.

The table below shows the R-squared, MAE, MSE, and RMSE values of the different models used (Table 1).

Based on the results of the regression plots and the different evaluation metrics shown in Table 2, the DNN model was found to be the best-performing model with a determination coefficient equal to 99.97% and MAE=0.45, MSE=0.05, RMSE=1.13, followed by the LR model with an R-squared equal to 99.10% and relatively high errors. In the third position, there is the k-NN, followed by the RF model, then the DT, and the XGB in the last position.

6 Discussion

In the research papers we have examined, particularly in [5, 19], and [22], deep neural networks (DNN) demonstrate superior performance in predicting students' grades compared with DT, LR, RF, k-NN, and XGB. This is attributed to its capability to autonomously learn and extract pertinent information from raw data, reducing the necessity for labor-intensive feature engineering. Additionally, DNN can be customized for specific tasks, thereby conserving both time and resources. Its ongoing advancement in the realms of artificial intelligence (AI) and machine learning contributes significantly to innovation, particularly in domains such as enhancing learning processes. This does not imply that the other algorithms do not achieve superior performance; quite the opposite, they do so in situations dissimilar to our specific problem.

Table 1 Analyzing various research papers within the domain of student grade prediction through the application of machine learning techniques

Research	Methodology	Advantage	Disadvantages
[19]	The study employed various machine learning algorithms, including logistic regression, decision trees, and neural networks, to predict student pass rates in online education. Data were collected from online learning platforms, encompassing factors such as time spent on course materials, engagement with online discussions, quiz scores, and previous educational background. These features were used as inputs to the machine learning models, and the pass/fail outcomes were predicted based on these inputs.	Improved prediction accuracy: machine learning algorithm process in this article trained a vast amount of data and identify complex patterns, leading to more accurate predictions of student pass rates. This paper adapts in real-time as new available data, ensuring predictions remain relevant and up-to-date. Identified predictive features: by analyzing the importance of different features, the study identified which aspects of student behavior and engagement were most indicative of pass rates, providing valuable insights for educators. Scalability: machine learning models used handle large datasets efficiently, making them suitable for online education platforms with a substantial number of students.	Data quality: the accuracy of predictions used a completeness of the data. This fact leaded to inaccurate predictions. The interpretability of the methods especially complex ones like neural networks makes the interpretation and understanding of the reasons behind specific predictions not complete. There was an overfitting: in machine learning models and as a result, the data was overfitting and gives a poor generalization to new unseen data.
[21]	The paper proposes a novel hybrid machine learning approach for demand forecasting, which may contribute to advancements in the field. The performances of five regression techniques of machine learning, random forest (RF), extreme gradient boosting (XGBoost), gradient boosting, adaptive boosting (AdaBoost), and artificial neural network (ANN) algorithms, are compared with a proposed hybrid (RF-XGBoost-LR) model for sales forecasting of a retail chain considering the various parameters of forecasting accuracy.	In this paper, we notice a Comparative analysis in order to understand the strengths and weaknesses of different approaches. A practical relevance that can have significant real-world applications. Insights for decision for companies looking to improve their demand forecasting strategies and inventory management. The methodological contribution is the development of a hybrid machine learning approach that could contribute to the broader field of machine learning and forecasting methodologies.	The use of data and assumptions affected the quality of data used and the assumptions made. The paper does not sufficiently detail the methodology and parameters used in the hybrid machine learning approach. The results of the study may be specific to the particular multi-channel retail company and dataset used, limiting the generalizability of findings to other contexts. The paper did not ensure proper statistical rigor in its analysis and comparisons.

Table 1 (continued)

Research	Methodology	Advantage	Disadvantages
[22]	The paper proposed the student performance prediction system then trained the model and tested with Kaggle dataset using different machine learning algorithms such as decision tree, naïve Bayes, random forest, support vector machine, k-nearest neighbor, and deep neural network in RProgramming. Finally, it compares the results of six algorithms out of which deep neural network outperformed with 84% as accuracy.	<p>Providing accurate predictions of student performance and enabling early identification of at-risk students.</p> <p>Handle large datasets, which is essential for educational institutions with a substantial number of students.</p> <p>The comparison graph of accuracy for all the six different machine learning algorithms shows that deep neural network produced high accuracy.</p>	<p>Limitations in terms of the choice of algorithms, parameter tuning, or the representativeness of the dataset.</p> <p>The quality and completeness of the data used in this study significantly impact the validity of the results.</p>

Table 2 Model evaluation based on R-squared, MAE, MSE, and RMSE

Model	R-squared (%)	MAE	MSE	RMSE
DT	97.32	3.17	17.24	4.15
LR	99.10	3.42	1.81	4.26
RF	98.59	1.44	4.00	2.00
k-NN	98.72	1.39	3.44	1.85
XGB	96.04	6.02	49.84	7.06
DNN	99.97	0.45	0.05	1.13

To draw comparisons between our study and previous research in the same domain and in order to avoid falling into the errors already made by other researchers, we concentrated on addressing the shortcomings observed in some prior works. Specifically, we made enhancements in the following aspects:

- Completeness of the data that lead to inaccuracies.
- The interpretability of the methods, especially complex ones like neural networks, hindered a complete understanding of the reasons behind specific predictions.
- The overfitting problem.
- The quality of data and underlying assumptions.
- Lack of sufficient detail regarding the methodology and parameters utilized in the hybrid machine learning approach.
- The dataset's limitations restricted the generalizability of findings to other contexts.
- Limitations concerning the choice of algorithms, parameter tuning, and the representativeness of the dataset.
- The quality and completeness of the data used.

7 Conclusion

The objective of this paper is to apply machine learning algorithms for the prediction of student scores. After implementing and evaluating a range of machine learning algorithms, our findings demonstrated that the deep neural network model performed better than the competing algorithms in terms of determination coefficient and error metrics. With a determination coefficient of 99.97% and negligible errors, the deep neural network demonstrated the highest level of accuracy in predicting students' grades.

These results have important implications for educators and administrators looking to use machine learning to improve student outcomes and support student success. By identifying the most effective algorithms for predicting student grades, we can better understand the factors that contribute to student performance and tailor teaching approaches and support to the specific needs of individual students.

Overall, this study highlights the potential of machine learning to revolutionize the way we approach education by providing personalized and targeted support to students. By continuing to explore and refine these techniques, we can continue to make progress in helping students achieve their full potential.

Finally, we can say that the application of machine learning techniques to predict students' marks has the potential to revolutionize the field of education. These models can offer a data-driven approach to identifying at-risk students, personalizing education, and improving overall academic outcomes. However, their implementation should be guided by a strong commitment to ethics and privacy to ensure that the benefits of these technologies are realized while safeguarding students' rights and well-being.

Author Contribution The authors confirm their contribution to the paper as follows: Study conception and design: KA, MF, MY, AA. Data collection: KA, MF, MY, AA. Analysis and interpretation of results: KA, MF, MY, AA. Draft manuscript preparation: KA, MF, MY, AA. All authors reviewed the results and approved the final version of the manuscript.

Funding The authors did not receive support from any organization for the submitted work.

Data Availability The datasets used during the current study are freely available in the UCI repository.

Code Availability The code will be available upon request to reviewers.

Declarations

Ethical Approval Not applicable

Consent to Participate Not applicable

Consent for Publication All authors of the manuscript have agreed for authorship, read and approved the manuscript, and given consent for the submission of the manuscript.

Conflict of Interest The authors declare no competing interests.

References

1. Okewu E, Adewole P, Misra S, Maskeliunas R, Damasevicius R (2021) Artificial neural networks for educational data mining in higher education: a systematic literature review. *Appl Artif Intell* 35(13):983–1021
2. Bañeres D, Rodríguez ME, Guerrero-Roldán AE, Karadeniz A (2020) An early warning system to detect at-risk students in online higher education. *Appl Sci* 10(13):4427
3. Yang SJ, Lu OH, Huang AY, Huang JC, Ogata H, Lin AJ (2018) Predicting students' academic performance using multiple linear regression and principal component analysis. *J Inf Process* 26:170–176
4. Huang S, Fang N (2013) Predicting student academic performance in an engineering dynamics course: a comparison of four types of predictive mathematical models. *Comput Educ* 61:133–145
5. Gorr WL, Nagin D, Szczypula J (1994) Comparative study of artificial neural network and statistical models for predicting student grade point averages. *Int J Forecast* 10(1):17–34
6. Gadhavi M, Patel C (2017) Student final grade prediction based on linear regression. *Indian J Comput Sci Eng* 8(3):274–279
7. Al-Barrak MA, Al-Razgan M (2016) Predicting students final GPA using decision trees: a case study. *Int J Inf Educ Technol* 6(7):528
8. Kolo DK, Adepoju SA (2015) A decision tree approach for predicting students academic performance
9. Hamoud A, Hashim AS, Awadh WA (2018) Predicting student performance in higher education institutions using decision tree analysis. *Int J Interact Multimed Artif Intell* 5:26–31
10. Amra IAA, Maghari AY (2017) Students performance prediction using KNN and Naïve Bayesian. In: 2017 8th Int Conf Inf Technol (ICIT). pp 909–913
11. Maghari A (2018) Prediction of student's performance using modified KNN classifiers. In Alfere SS, Maghari AY (2018) Prediction of Student's Performance Using Modified KNN Classifiers. In: The First International Conference on Engineering and Future Technology (ICEFT 2018). pp 143–150

12. Batool S, Rashid J, Nisar MW, Kim J, Mahmood T, Hussain A (2021) A random forest students' performance prediction (rfspp) model based on students' demographic features. In: 2021 Mohammad Ali Jinnah University International Conference on Computing (MAJICC). pp 1–4
13. Altabrauee H, Ali OAJ, Ajmi SQ (2019) Predicting students' performance using machine learning techniques. *J Univ Babylon Pure Appl Sci* 27(1):194–205
14. Yousafzai BK, Khan SA, Rahman T, Khan I, Ullah I, Ur Rehman A, Cheikhrouhou O (2021) Student-performulator: student academic performance using hybrid deep neural network. *Sustainability* 13(17):9775
15. Aslam N, Khan I, Alamri L, Almuslim R (2021) An Improved Early Student's Academic Performance Prediction Using Deep Learning. *Int J Emerg Technol Learn (iJET)* 16(12):108–122
16. Sekeroglu B, Dimililer K, Tuncal K (2019) Student performance prediction and classification using machine learning algorithms. In: *Proceedings of the 2019 8th International Conference on Educational and Information Technology*. pp 7–11
17. Burdakov O (2020) Ioannis C. Demetriou and Panos M. Pardalos (eds): *Approximation and optimization: algorithms, complexity and applications*. In: *SN Oper Res Forum*, vol 1. Springer International Publishing, pp 1–5
18. Korani W, Mouhoub M (2021) Review on nature-inspired algorithms. In: *Oper Res Forum*, vol 2. Springer International Publishing, pp 1–26
19. Ma X, Yang Y, Zhou Z (2018) Using machine learning algorithm to predict student pass rates in online education. In: *Proceedings of the 3rd International Conference on Multimedia Systems and Signal Processing*. pp 156–161
20. Manzali Y, Elfar M (2023) Random forest pruning techniques: a recent review. In: *Oper Res Forum*, vol 4, no 2. Springer International Publishing, pp 1–14
21. Mitra A, Jain A, Kishore A, Kumar P (2022) A comparative study of demand forecasting models for a multi-channel retail company: a novel hybrid machine learning approach. In: *Oper Res Forum*, vol 3, no 4. Cham: Springer International Publishing, p 58
22. Vijayalakshmi V, Venkatachalapathy K (2019) Comparison of predicting student's performance using machine learning algorithms. *Int J Intell Sys Appl* 11(12):34

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

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

Authors and Affiliations

Adil Korchi¹ · Fayçal Messaoudi² · Ahmed Abatal³ · Youness Manzali⁴

✉ Adil Korchi
korchi.a@ucd.ac.ma

Fayçal Messaoudi
faycal.messaoudi@usmba.ac.ma

Ahmed Abatal
a.abatal@uhp.ac.ma

Youness Manzali
younes.manzali@usmba.ac.ma

¹ Faculty of Juridical, Economic and Social Sciences, Chouaib Doukkali University, El Jadida, Morocco

² National School of Commerce and Management, Sidi Mohamed Ben Abdellah University, Fez, Morocco

³ Faculty of Sciences and Techniques, Hassan Premier University, Settat, Morocco

⁴ Faculty of Science Dhar El Mahraz, Sidi Mohamed Ben Abdellah University, Fez, Morocco