FUDMA Journal of Sciences (FJS)
ISSN online: 2616-1370
ISSN print: 2645 - 2944
Vol. 8 No. 6, December, (Special Issue) 2024, pp 288 - 298
DOI: https://doi.org/10.33003/fjs-2024-0806-2927

PREDICTION ACCURACY ANALYSIS OF MACHINE LEARNING CLASSIFIERS ON STUDENT COURSE ASSESSMENT METHODS

***[1]Godwin A. Otu, [2]Oludele Awodele, [3]Sola A. Adeniji, [1]Henry O. Mafua, [4]Kehinde A. Olayanju, [5]Adeniyi U. Adedayo, [5]Suleiman A. Usman, [5]Samson Adeyinka, [1]Aisha Ramalan and [5]Maryam Masari**

[1]Department of Computer Science, Air Force Institute of Technology Kaduna, Nigeria.
[2]Department of Computer Science, Babcock University, Nigeria.
[3]Department of Computer Science, National Open University Abuja, Nigeria.
[4]Department of Computer Science, Federal College of Education (Technical), Akoka, Lagos-Nigeria.
[5]Department of Computer Security, Air Force Institute of Technology Kaduna, Nigeria.

*Corresponding authors' email: goddy2fine@gmail.com

ABSTRACT

There is growing need to improve the quality of education through an effective service delivery from educators. Also, educational institutions are searching for ways to reduce student failure rate. The rapid growth in size and availability of student data and robust algorithms to generate machine learning models, more accurate predictions and tailored learning interventions can be factored. The research investigates the prediction accuracy of machine learning algorithms, including Logistic Regression, Random Forest (RF), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Naive Bayes, XGBoost, and Gradient Boosting, applied to student learning attributes and course assessment. The aim is to evaluate the effectiveness of these algorithms in predicting student performance on various metrics. A dataset encompassing student learning attributes and assessment modes were analyzed. Each algorithm's predictive capabilities was assessed using accuracy, precision, recall and F1-score metrics. Logistic regression had the highest accuracy score of 0.93, SVM and XGBoost both achieved an accuracy 0f 0.90 while Random Forest, KNN and Naive Bayes achieved same accuracy score of 0.88 while Gradient Boosting achieved an accuracy score of 0.85 each which was the lowest. RF, SVM, KNN got the same F-score, recall and precision of 0.93, 0.97 and 0.90 respectively while Naive Bayes, XGBoost, and Gradient Boosting achieved the same recall of 0.94 while KNN had a recall of 0.97. Gradient Boosting had a precision of 0.89, and an F-score of 0.92, the F-score of Naïve Bayes was 0.93. This research underscores the potential of advanced machine learning techniques in enhancing educational outcomes.

Keywords: Prediction Accuracy, Machine Learning Algorithms, Student Performance, Course Assessment, Educational Data Mining

INTRODUCTION

The introduction of machine learning (ML) into the educational sector has provided new avenues for understanding and strengthening student learning output (Xie and Yanxin, 2024; Liang *et al*., 2014). With the large volume of data generated by educational institutions, leveraging predictive analytics can assist educators identify significant factors that influence student performance and inform intervention techniques (Al-Shabandar *et al*., 2017). This research focuses on the prediction accuracy of several prominent machine learning algorithms—Logistic Regression, Random Forest, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Naive Bayes, XGBoost, and Gradient Boosting—when applied to student learning attributes and course assessment data. The comparison of these classifiers in these research aim to determine which classifier yields the most accurate predictions, which in turn supports data-driven decision-making in educational domain. As educational institutions mostly depend on data to tailor learning experiences, the need for efficient predictive models becomes a necessity. Understanding how different ML algorithms perform in this domain can assist educators in selecting the most appropriate tools for analyzing student data and improving academic output. This research will contribute to the area of educational data mining by holistically evaluating how effective these ML algorithms are in the prediction of student success relying on different student learning characteristics. Yang et al., 2024) employed deep learning technique to effectively identify course characteristics from reviews, assesses learners' satisfaction, attention, and cost of improvement, and consequently formulated a prioritized list of course attributes for improvement.

The use of machine learning algorithms a tool in solving myriad of challenges in the education sector has been the focus of many researches, each delving into different areas of student performance prediction. (Chen *et al*, 2021; Chen *et al.* 2019) converted student dropout task into the learning behaviour and classification model using logical regression, support vector machine and decision tree. The model was used to track the registration characteristics and learning behaviour characteristics. Zihan *et al.* (2023) built the dropout prediction model by using enhanced decision tree, so as to predict the dropout probability of learners, and then compute the ratio in various degrees in order to derive varied intervention measures. Logistic Regression has been used mostly in researches because it is easy to implement and in all its simplicity and less difficulty in result interpretation, most profoundly in binary classification tasks which pertains to student success (Tang *et al*., 2018). In contrast, Random Forest, with its ensemble technique, has demonstrated robust performance in dealing with complex datasets, reducing overfitting, thereby guaranteeing an improve prediction accuracy (Xia and Qi, 2024; Xia and Qi, 2023). Support Vector Machines (SVM) are also prevalent in educational data mining due to their effectiveness in high-dimensional

spaces, making them suitable for datasets with numerous learning attributes. K-Nearest Neighbors (KNN), while straightforward, offers a unique perspective on local data structure and can be effective for identifying patterns based on proximity (Liang *et al*., 2014). Naive Bayes classifiers which are probabilistic in solving problems in research have been deployed successfully in different educational contexts, especially in areas involving object type data (Chao, 2024). On the more advanced scale, XGBoost and Gradient Boosting have garnered traction for being able to handle huge datasets with different characteristics, which results in most cases outperforming conventional algorithms in terms of accuracy and computational efficiency (Burgos *et al*., 2018).

Recent literature points out the significance of comparing these algorithms to identify the best solutions for specific educational datasets. Findings have shown that ensemble methods, such as XGBoost and Random Forest, most times yield more accurate and superior results compared to simpler models (LIU *et al*., 2017). In all these findings, there remains a need for comprehensive comparisons that involves different array of ML algorithms, particularly when considering student learning characteristics and course assessment dataset.

This study aims to fill the research gap by providing a systematic evaluation of the prediction accuracy of these seven machine learning algorithms. The outcome of the research will not only strengthen the understanding of their applicability in educational parlance but also contribute to the wider area of educational data mining by discovering the most effective techniques for predicting student academic success (Alturki *et al*., 2022).

MATERIALS AND METHODS

Data Collection

Data is collected from students of Air Force Institute of Technology Kaduna, Nigeria using a questionnaire designed from Google form. The data collected includes student learning attributes which consists of attendance in classes, participation of students, student grades, and demographics like age and learning groups while the course assessment data comprises of assignment scores, examination results, total course grades and preferred assessment type. Student identification information (matriculation number and name) will not be used in the research in order to protect student privacy.

Data Preprocessing

Data preprocessing is a fundamental step in the preparation of data before it is inputted in the ML algorithm (Ortiz *et al*., 2024). It involves cleaning and transforming the collected data to make it suitable for analysis. This step includes: Handling Missing Values with methods such as imputation using KNNimputer from sklearn library and the use of fillna function from pandas library. Also null values and incomplete records will be dropped using dropna function from pandas library. Secondly, encoding categorical variables using label encoding and python get_dummies function in order to facilitate numerical analysis. Then followed by feature scaling. This involves the standardization and normalization will be performed on numerical features to ensure uniformity across the dataset using standard scalar from Sklearn library particularly for algorithms sensitive to the scale of input data, such as KNN and SVM. Lastly, feature selection phase. Feature selection aims to identify the most relevant attributes for predicting student outcomes (Tri *et al*., 2024). The correlation analysis involving the use of heatmap and describe function all from pandas library will be employed to select a subset of features that do not have negative correlation so as to contribute significantly to model performance. This step helps to reduce dimensionality and also aid in the easy interpretability of the model (Kaur and Sarmadi, 2024).

Model Training

The dataset is divided into x-train, y-train, x-test and y-test set. The x-train also known as the independent variable is a set of input features and y-train is the target variable or the dependent variable. Both x-training and y-train are used for training the model. The x-test is used to make predictions, the predicted values are the compared with the y-test data. The train-test-split function of sklearn is used to split the data randomly into training and testing sets using an 80-20 split. That is 80% for training and 20% for testing. Cross-validation will also be applied to ensure robustness and cut-off overfitting which might result in the model not being generative.

Model Evaluation

The performance of each model is evaluated based on the following metrics: accuracy, precision, recall, and F-score (Prakash and Kalaiarasan, 2024). After model evaluation, a comparative analysis is carried-out to identify which algorithm performs best in terms of prediction accuracy and other evaluation metrics. Confusion matrices will be used to present the results elaborately (Dobson, 2024).). The performance metrics can be measured using the following index (Swaminathan *et al.,* 2024).

True Positive (TP): It is the case when both actual class & predicted class of data point is 1.

True Negative (TN): It is the case when both actual class & predicted class of data point is 0.

False Positive (FP): It is the case when actual class of data point is 0 & predicted class of data point is 1.

False Negative (FN): It is the case when actual class of data point is 1 & predicted class of data point is 0.

Accuracy is the ratio of correct predictions among the total predictions

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Precision is defined as the ratio of true positive predictions to the total predicted positives, indicating the accuracy of positive prediction. It is computed using the formula:

$$precission = \frac{TP}{TP+Fp} \quad (2)$$

Recall is the proportion of true positive predictions among to the total predicted positives, indicating the accuracy of positive predictions. It is calculated using the formula:

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

F1 Score is the harmonic mean of precision and recall, providing a balance between the two metrics (Sharma *et al*., 2023). It is calculated using the formula:

$$F1 - Score = \frac{2*Recall*Precision}{Recall+Precision} \quad (4)$$

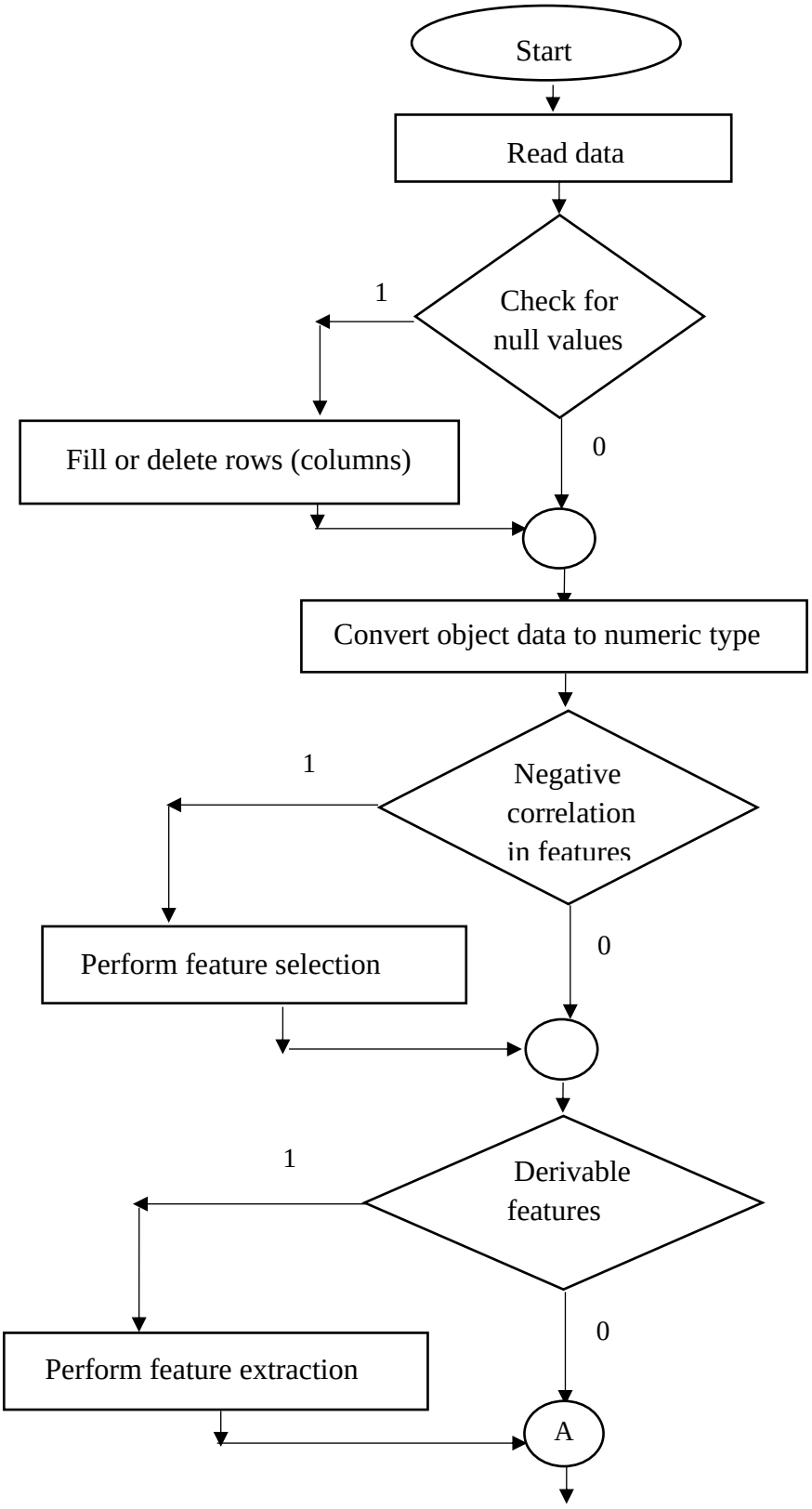

Figure 1a: Program Flowchart (Otu *et al*., 2023)

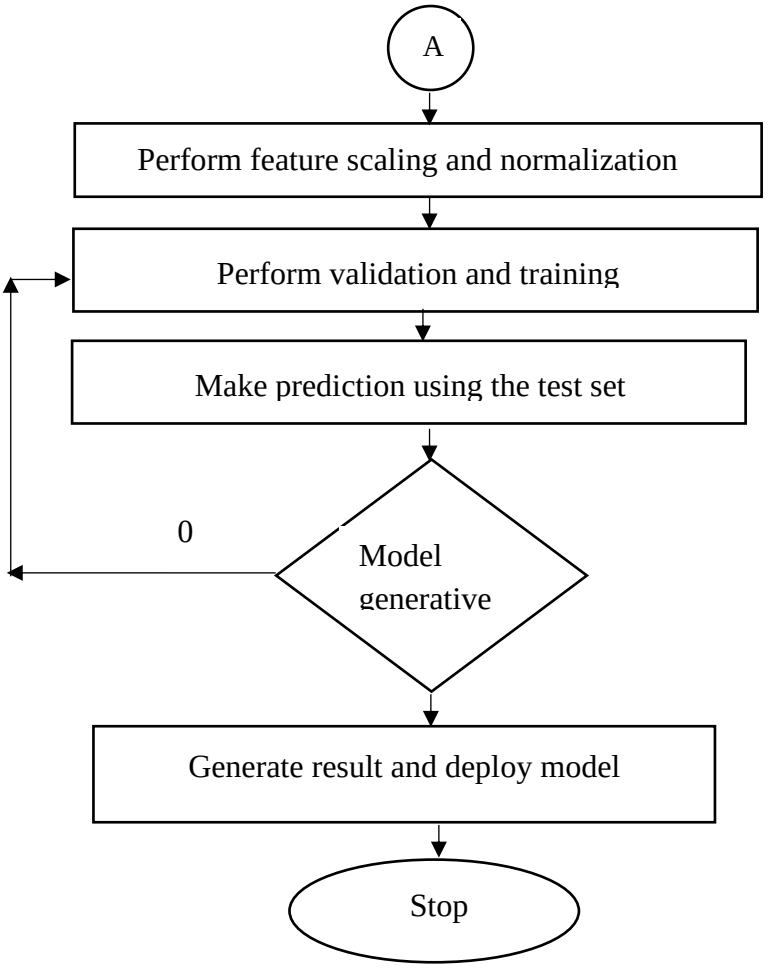

Figure 1b: Program Flowchart (Otu *et al*., 2023)

RESULTS AND DISCUSSION

Logistic Regression

Logistic Regression serves as a fundamental baseline for binary classification problems. In the analysis, it demonstrated a high level of accuracy but often struggled with complex relationships in the data. While it is interpretable and computationally efficient, its performance is generally lower than more complex models (Lalitha *et al.*, 2021). This aligns with existing literature indicating that Logistic Regression may not capture nonlinear patterns effectively (Elvira *et al*., 2024). Table 1 shows the classification report for the logistic regression model.

Table 1: Showing the result from the logistic regression algorithm

Accuracy: 0.93	**precision**	**Recall**	**F1-score**	**Support**
0	0.92	1.00	0.96	36
1	1.00	0.25	0.40	4
Accuracy			0.93	40
Macro avg	0.96	0.62	0.68	40
Weighted avg	0.93	0.93	0.90	40

The classification report for Table 1 is generated from model evaluation metrics using equation 1, 2, 3 and 4 as follows:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{36+1}{36+1+0+3} = \frac{37}{40} = 0.925 \simeq 0.93$$

$$precission = \frac{TP}{TP+Fp} = \frac{36}{36+3} = \frac{36}{39} = 0.92$$

$$Recall = \frac{TP}{TP+FN} = \frac{36}{36+0} = \frac{36}{36} = 1.00$$

$$F1-Score = \frac{2*Recall*Precision}{Recall+Precision} = \frac{2*0.93*1.00}{0.93+1.00} = 0.96$$

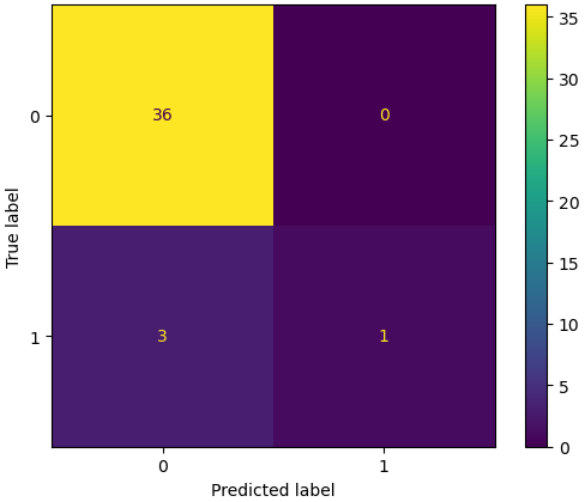

Figure 2: Confusion matrix showing predicted result using logistic regression algorithm

The confusion matrix depicts that the logistic regression algorithm was able to correctly predict 37 items and predicted 3 items incorrectly. The zero in the upper right corner of the matrix which stands for false positive (FP) occurs when actual class of data point is 0 and predicted class of data point is 1. While the lower left corner which designates false negative (FN) shows 3 items incorrectly predicted by the algorithm occurs in a case when actual class of data point is 1 and predicted class of data point is 0.

Random Forest

The Random Forest algorithm exhibited superior performance due to its ensemble (combination of a family of classifier algorithms) nature, which solves the problem of overfitting by averaging many decision trees (Balakrishnan and Coetzee, 2013). The use of the model produced a high accuracy, displaying its robustness against noise and its ability to handle huge datasets with many attributes. Attributes significance analysis showed that certain learning attributes were particularly influential, validating educational theories about student success factors. Table 2 shows the classification report of random forest model.

Table 2: Showing the result from the random forest algorithm

Accuracy: 0.88	**precision**	**Recall**	**F1-score**	**Support**
0	0.90	0.97	0.93	36
1	0.00	0.00	0.00	4
Accuracy			0.88	40
Macro avg	0.45	0.49	0.47	40
Weighted avg	0.81	0.88	0.84	40

The classification report for Table 2 is generated from model evaluation metrics using equation 1, 2, 3 and 4 as follows:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{35+0}{35+0+1+4} = \frac{35}{40} = 0.875 \simeq 0.88$$

$$precission = \frac{TP}{TP+Fp} = \frac{35}{35+4} = \frac{35}{36} \simeq 0.90$$

$$Recall = \frac{TP}{TP+FN} = \frac{35}{35+1} = \frac{35}{36} \simeq 0.97$$

$$F1-Score = \frac{2*Recall*Precision}{Recall+Precision} = \frac{2*0.90*0.97}{0.90+0.97} = 0.93$$

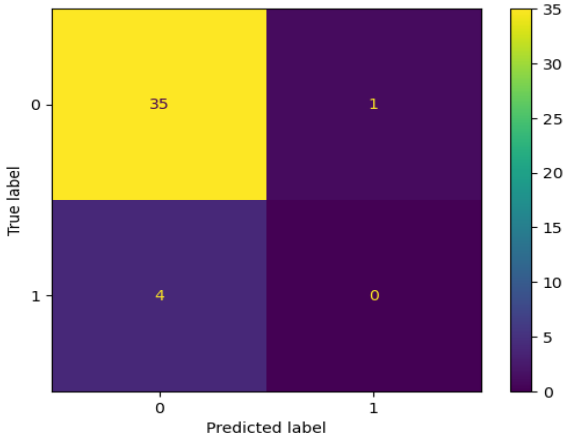

Figure 3: Confusion matrix showing predicted result using random forest algorithm

The confusion matrix shows that the random forest algorithm was able to correctly predict 35 items and predicted 4 items incorrectly. The 1 in the upper right corner of the matrix which stands for false positive (FP) occurs when actual class of data point is 0 and predicted class of data point is 1. While the lower left corner which designates false negative (FN) shows 4 items incorrectly predicted by the algorithm occurs in a case when actual class of data point is 1 and predicted class of data point is 0.

Support Vector Machine (SVM)

SVM displayed promising results, especially when making use of the kernel functions to capture nonlinear relationships. Its accuracy was competitive, but training time was notably higher, especially with large datasets (Hadyaoui and Cheniti-Belcadhi, 2023). The choice of kernel most importantly affected the results, prioritizing the significance of parameter tuning in SVM applications. Table 3 shows the classification report of support vector machine model.

Table 3: Showing the result from the support vector machine algorithm

Accuracy: 0.90	precision	Recall	F1-score	Support
0	0.90	1.00	0.95	36
1	0.00	0.00	0.00	4
Accuracy			0.90	40
Macro avg	0.45	0.50	0.47	40
Weighted avg	0.81	0.90	0.85	40

The classification report for Table 3 is generated from model evaluation metrics using equation 1, 2, 3 and 4 as follows:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{36+0}{36+0+0+4} = \frac{36}{40} = 0.9 \simeq 0.90$$

$$precission = \frac{TP}{TP+Fp} = \frac{36}{36+4} = \frac{36}{40} = 0.90$$

$$Recall = \frac{TP}{TP+FN} = \frac{36}{36+0} = \frac{36}{36} \simeq 1.00$$

$$F1-Score = \frac{2*Recall*Precision}{Recall+Precision} = \frac{2*0.90*1.00}{0.90+1.00} = 0.947 \simeq 0.95$$

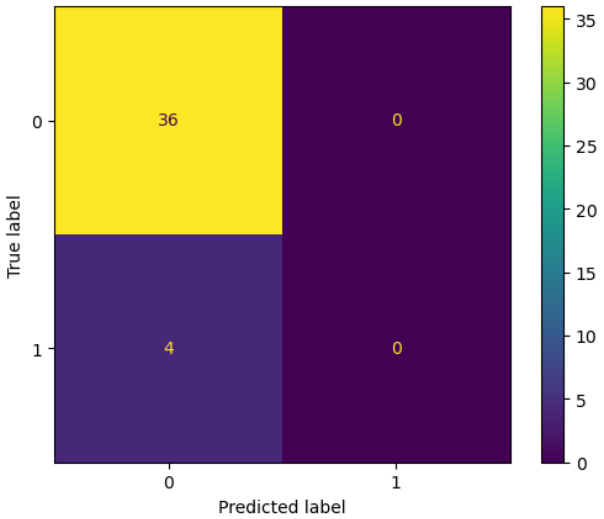

Figure 4: Confusion matrix showing predicted result using support vector machine algorithm

The confusion matrix shows that the support vector machine (SVC) algorithm was able to correctly predict 36 items and predicted 4 items incorrectly. The 0 in the upper right corner of the matrix which stands for false positive (FP) occurs when actual class of data point is 0 and predicted class of data point is 1. While the lower left corner which designates false negative (FN) shows 4 items incorrectly predicted by the algorithm occurs in a case when actual class of data point is 1 and predicted class of data point is 0.

K-Nearest Neighbors (KNN)

KNN's output variable, is a function of the choice of distance metric and the number of neighbors (k). Although it was easy to implement, KNN manage to perform with high-dimensional data due to the curse of dimensionality, resulting in a decreased accuracy compared to ensemble techniques. Its sensitivity to irrelevant attributes also highlighted the relevance of proper feature selection (Kloft *et al*., 2014). Table 4 shows the classification report of K-Nearest Neighbors (KNN) model.

Table 4: Showing the result from the K-Nearest Neighbors (KNN) algorithm

Accuracy: 0.88	precision	Recall	F1-score	Support
0	0.90	0.97	0.93	36
1	0.00	0.00	0.00	4
Accuracy			0.88	40
Macro avg	0.45	0.49	0.47	40
Weighted avg	0.81	0.88	0.84	40

The classification report for table 4 is generated from model evaluation metrics using equation 1, 2, 3 and 4 as follows:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{35+0}{35+0+1+4} = \frac{35}{40} = 0.875 \simeq 0.88$$

$$precission = \frac{TP}{TP+Fp} = \frac{35}{35+4} = \frac{35}{39} = 0.90$$

$$Recall = \frac{TP}{TP+FN} = \frac{35}{35+1} = \frac{35}{36} \simeq 0.97$$

$$F1-Score = \frac{2*Recall*Precision}{Recall+Precision} = \frac{2*0.92*0.97}{0.92+0.97} = 0.93$$

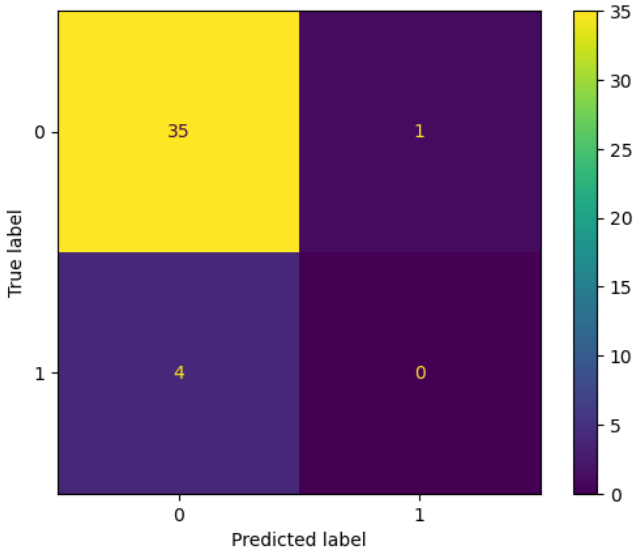

Figure 5: Confusion matrix showing predicted result using K-Nearest Neighbors (KNN) algorithm

The confusion matrix shows that the K-Nearest Neighbors (KNN) algorithm was able to correctly predict 35 items and predicted 5 items incorrectly. The 1 in the upper right corner of the matrix which stands for false positive (FP) occurs when actual class of data point is 0 and predicted class of data point is 1. While the lower left corner which designates false negative (FN) shows 4 items incorrectly predicted by the algorithm occurs in a case when actual class of data point is 1 and predicted class of data point is 0.

Naive Bayes

Naive Bayes produced a surprisingly effective performance given its simplicity (Qiu *et al*., 2016). This model is particularly useful for object (string, date) data and demonstrated resilience against attribute independence assumptions. While it did not achieve the highest accuracy, it performed very well in terms of speed and efficiency, making it a valuable choice for real-time applications. Table 5 shows the classification report of Naïve Bayes model.

Table 5: Showing the result from the Naïve Bayes algorithm

Accuracy: 0.88	**precision**	**Recall**	**F1-score**	**Support**
0	0.92	0.94	0.93	36
1	0.33	0.25	0.29	4
Accuracy			0.93	40
Macro avg	0.96	0.62	0.68	40
Weighted avg	0.93	0.93	0.90	40

The classification report for Table 5 is generated from model evaluation metrics using equation 1, 2, 3 and 4 as follows:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{34+1}{34+1+2+3} = \frac{35}{40} = 0.875 \simeq 0.88$$

$$precission = \frac{TP}{TP+Fp} = \frac{34}{34+3} = \frac{34}{37} = 0.92$$

$$Recall = \frac{TP}{TP+FN} = \frac{34}{34+2} = \frac{34}{36} \simeq 0.94$$

$$F1-Score = \frac{2*Recall*Precision}{Recall+Precision} = \frac{2*0.92*0.94}{0.92+0.94} = 0.93$$

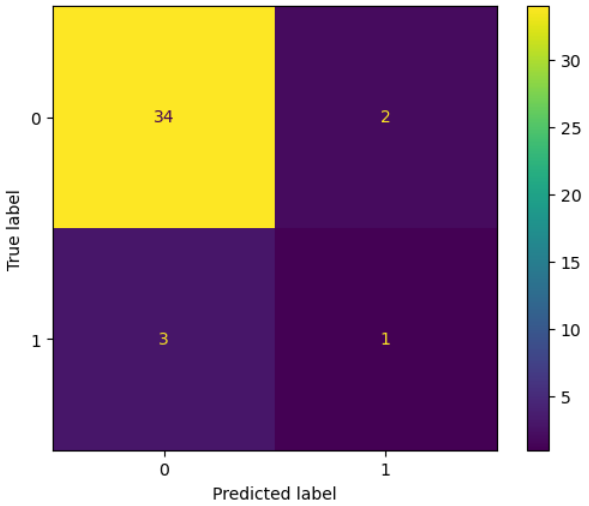

Figure 6: Confusion matrix showing predicted result using Naïve Bayes algorithm

The confusion matrix shows that naïve Bayes algorithm was able to correctly predict 35 items and predicted 5 items incorrectly. The 2 in the upper right corner of the matrix which stands for false positive (FP) occurs when actual class of data point is 0 and predicted class of data point is 1. While the lower left corner which designates false negative (FN) shows 3 items incorrectly predicted by the algorithm occurs in a case when actual class of data point is 1 and predicted class of data point is 0.

XGBoost

XGBoost outperformed many other algorithms, showcasing its strength in handling structured data and its ability to model intricate interactions. Its gradient boosting framework and built-in regularization led to high accuracy and robustness, inferring its status as a good choice in many prediction tasks (Song, 2022). The application of cross-validation also strengthened its reliability. Table 6 shows the classification report of XGBoost model.

Table 6: Showing the result from the XGBoost algorithm

Accuracy: 0.90	**precision**	**Recall**	**F1-score**	**Support**
0	0.90	1.00	0.95	36
1	0.00	0.00	0.00	4
Accuracy			0.90	40
Macro avg	0.45	0.50	0.47	40
Weighted avg	0.81	0.90	0.85	40

The classification report for Table 6 is generated from model evaluation metrics using equation 1, 2, 3 and 4 as follows:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{36+0}{36+0+0+4} = \frac{36}{40} = 0.90$$

$$precission = \frac{TP}{TP+Fp} = \frac{36}{36+4} = \frac{36}{40} = 0.90$$

$$Recall = \frac{TP}{TP+FN} = \frac{36}{36+0} = \frac{36}{36} \simeq 1.00$$

$$F1-Score = \frac{2*Recall*Precision}{Recall+Precision} = \frac{2*0.90*1.00}{0.90+1,00} = 0.947 \simeq 0.95$$

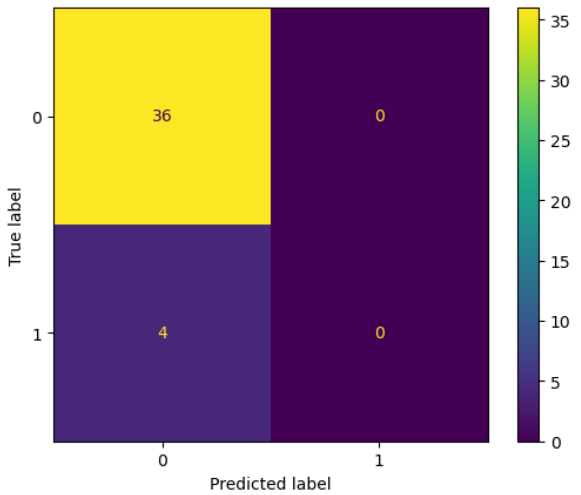

Figure 7: Confusion matrix showing predicted result using XGBoost algorithm

The confusion matrix shows that the XGBoost algorithm was able to correctly predict 36 items and predicted 4 items incorrectly. The 0 in the upper right corner of the matrix which stands for false positive (FP) occurs when actual class of data point is 0 and predicted class of data point is 1. While the lower left corner which designates false negative (FN) shows 4 items incorrectly predicted by the algorithm occurs in a case when actual class of data point is 1 and predicted class of data point is 0.

Gradient Boosting

Similar to XGBoost, the Gradient Boosting model showed a capable predictive power but was generally slower and more susceptible to overfitting without proper tuning. The loop nature of boosting allows for improved performance on various tasks, but it requires meticulous weight optimization (Wang *et al*., 2020). Table 7 shows the classification report of Gradient Boosting model.

Table 7: Showing the result from the Gradient Boosting algorithm

Accuracy: 0.85	precision	Recall	F1-score	Support
0	0.89	0.94	0.92	36
1	0.00	0.00	0.00	4
Accuracy			0.85	40
Macro avg	0.45	0.47	0.46	40
Weighted avg	0.81	0.85	0.83	40

The classification report for table 7 is generated from model evaluation metrics using equation 1, 2, 3 and 4 as follows:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{34+0}{34+0+2+4} = \frac{34}{40} = 0.85$$

$$precission = \frac{TP}{TP+Fp} = \frac{34}{34+4} = \frac{34}{38} = 0.89$$

$$Recall = \frac{TP}{TP+FN} = \frac{34}{34+2} = \frac{34}{36} = 0.94$$

$$F1-Score = \frac{2*Recall*Precision}{Recall+Precision} = \frac{2*0.90*0.94}{0.90+0.94} = 0.9195 \simeq 0.92$$

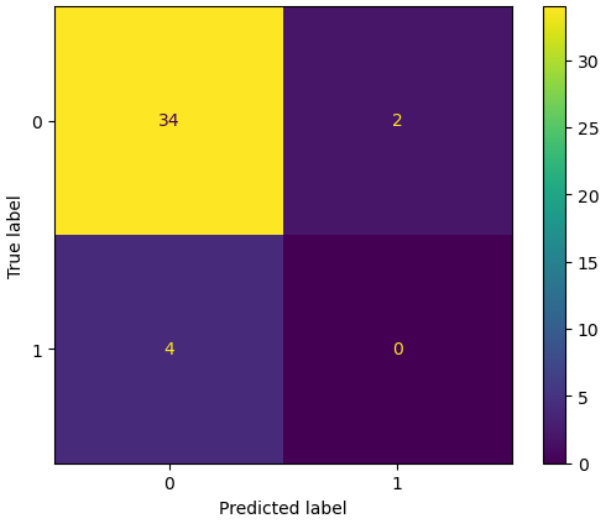

Figure 8: Confusion matrix showing predicted result using Gradient Boosting algorithm

The confusion matrix shows that the Gradient Boosting algorithm was able to correctly predict 34 items and predicted 6 items incorrectly. The 2 in the upper right corner of the matrix which stands for false positive (FP) occurs when actual class of data point is 0 and predicted class of data point is 1. While the lower left corner which designates false negative (FN) shows 4 items incorrectly predicted by the algorithm occurs in a case when actual class of data point is 1 and predicted class of data point is 0.

CONCLUSION

The comparative analysis of these machine learning algorithms showed important variations in their predictive accuracies concerning student learning characteristics and course assessment methods. Logistic Regression, support vector and XGBoost emerged as the most effective algorithms for this dataset, indicating their superiority in handling intricate educational data. Future research should explore hybrid models or ensemble technique which employs the combination of the strengths of these algorithms to further strengthen prediction accuracy.

REFERENCES

Al-Shabandar R, Hussain A, Laws A, et al. (2017). Machine learning approaches to predict learningoutcomes in Massive open online courses[C]//2017 International Joint Conference on NeuralNetworks (IJCNN). IEEE, 2017: 713-720.

Alturki, S., Cohausz, L., & Stuckenschmidt, H. (2022). Predicting Master's students' academic performance: An empirical study in germany. *Smart Learning Environments*, *9*(1), 38. doi:https://doi.org/10.1186/s40561-022-00220-y

Balakrishnan G, Coetzee D. (2013). Predicting student retention in massive open online courses using hidden markov

models [J]. Electrical Engineering and Computer Sciences University of California at Berkeley, 2013, 53: 57-58.

Burgos C, Campanario M L, de la Pena D, et al. (2018). Data mining for modeling students' performance: A tutoring action plan to prevent academic dropout [J]. Computers & ElectricalEngineering, 2018, 66: 541-556.

Chao Ma. (2024). Improving the prediction of student performance by integrating a random forest classifier with meta-heuristic optimization algorithms. *International Journal of Advanced Computer Science and Applications*, *15*(6) https://doi.org/10.14569/IJACSA.2024.01506106

Chen J, Feng J, Sun X, et al. (2019). MOOC dropout prediction using a hybrid algorithm based on decision tree and extreme learning machine [J]. Mathematical Problems in Engineering, 2019, 2019

Chen, Min & Wu, Lifa. (2021). A dropout prediction method based on time series model in MOOCs. Journal of Physics: Conference Series. 1774. 012065. https://doi.org/10.1088/1742-6596/1774/1/012065 .

Dobson, J. E. (2024). On the confusion matrix. *Configurations*, *32*(4), 331-350. https://doi.org/10.1353/con.2024.a942087

Elvira Vidya Berliana, Mardhani Riasetiawan. (2024). Comparative analysis of naïve bayes classifier, support vector machine and decision tree in rainfall classification using confusion matrix. *International Journal of Advanced Computer Science and Applications*, *15*(7) https://doi.org/10.14569/IJACSA.2024.0150755

Hadyaoui, A., & Cheniti-Belcadhi, L. (2023). Ontology-based group assessment analytics framework for performances prediction in project-based collaborative learning. *Smart Learning Environments*, *10*(1), 43. https://doi.org/10.1186/s40561-023-00262-w

Kaur, A., & Sarmadi, M. (2024). Comparative analysis of data preprocessing methods, feature selection techniques and machine learning models for improved classification and regression performance on imbalanced genetic data. Ithaca: https://doi.org/10.1007/s40745-024-00575-8

Kloft M, Stiehler F, Zheng Z, et al. (2014). Predicting MOOC dropout over weeks using machine learning methods[C]//Proceedings of the EMNLP 2014 workshop on analysis of large scale social interaction in MOOCs. 2014: 60-65.

Lalitha, R., J, D. L., Kavitha, K., RamaReddy, T., Srinivas, R., & Sujana, C. (2021). Prediction and analysis of corona virus disease (COVID-19) using cubist and OneR. *IOP Conference Series. Materials Science and Engineering*, *1074*(1) https://doi.org/10.1088/1757-899X/1074/1/012022

Liang J, Li C, Zheng L. (2014). Machine learning application in MOOCs: Dropout prediction[C]//2016 11th International Conference on Computer Science & Education (ICCSE). IEEE, 2016: 52-57.

LIU T, Xiu L I. (2017). Finding out reasons for low completion in MOOC environment: an explicable approach using hybrid data mining methods [J]. DEStech Transactions on Social Science,Education and Human Science, 2017 (meit).

Ortiz, B. L., Gupta, V., Kumar, R., Jalin, A., Cao, X., Ziegenbein, C., . . . Choi, S. W. (2024). Data preprocessing techniques for AI and machine learning readiness: Scoping review of wearable sensor data in cancer care. *JMIR mHealth and uHealth*, *12* https://doi.org/10.2196/59587

Otu, Godwin & Usman, Suleiman & Ugbe, Raphael & Iheagwara, Stephen & Okafor, Akudo & Okonkwo, Fredrick & Shukurah, Oyebanji & Adeniyi, Usman. (2023). COMPARATIVE ANALYSIS OF AGGREGATION AND INHERITANCE STRATEGIES IN INCREMENTAL PROGRAM DEVELOPMENT. FUDMA JOURNAL OF SCIENCES. 7. 57-64. https://doi.org/10.33003/fjs-2023-0702-1710 .

Prakash, K., & Kalaiarasan, C. (2024). Web services performance prediction with confusion matrix and K-fold cross validation to provide prior service quality characteristics. *Journal of Electrical Systems*, *20*(2), 284-292. Retrieved from https://www.proquest.com/scholarly-journals/web-services-performance-prediction-with/docview/3078054634/se-2

Qiu J, Tang J, Liu T X, et al. (2016). Modeling and predicting learning behavior in MOOCs[C]//Proceedings of the ninth ACM international conference on web search and data mining. ACM, 2016: 93-102.

Sharma, N., Appukutti, S., Garg, U., Mukherjee, J., & Mishra, S. (2023). Analysis of Student's academic performance based on their time spent on extra-curricular activities using machine learning techniques. *International Journal of Modern Education and Computer Science*, *15*(1), 46. doi:https://doi.org/10.5815/ijmecs.2023.01.04

Song, Q. (2022). Generation and research of online english course learning evaluation model based on genetic algorithm improved neural set network. *Computational Intelligence and Neuroscience : CIN*, *2022* https://doi.org/10.1155/2022/7281892

Swaminathan, Sathyanarayanan & Tantri, B Roopashri. (2024). Confusion Matrix-Based Performance Evaluation Metrics. African Journal of Biomedical Research. 27. 4023-4031. https://doi.org/10.53555/AJBR.v27i4S.4345 .

Tang C, Ouyang Y, Rong W, et al. (2018). Time series model for predicting dropout in massive openonline courses[C]//International Conference on Artificial Intelligence in Education. Springer, Cham, 2018: 353-357.

Tri, N. Q., Tung, N. T., Anh, D. L., Huong Pham, T. V., & Doanh, S. C. (2024). Classification of glucose-level in deionized water using machine learning models and data pre-processing technique. *PLoS One*, *19*(12) https://doi.org/10.1371/journal.pone.0311482

Wang W, Yu H, Miao C. (2020). Deep model for dropout prediction in MOOCs [C]//Proceedings of the ICDPAM 2020 Journal of Physics: Conference Series 1774 (2021) 012065

IOP Publishing https://doi.org/10.1088/1742-6596/1774/1/0120657

Xia, X., & Qi, W. (2023). Dropout prediction and decision feedback supported by multi temporal sequences of learning behavior in MOOCs: Revista de universidad y sociedad del conocimiento. *International Journal of Educational Technology in Higher Education*, *20*(1), 32. https://doi.org/10.1186/s41239-023-00400-x

Xia, X., & Qi, W. (2024). Driving STEM learning effectiveness: Dropout prediction and intervention in MOOCs based on one novel behavioral data analysis approach. *Humanities & Social Sciences Communications*, *11*(1), 430. https://doi.org/10.1057/s41599-024-02882-0

Xie, Yanxin. (2024). Efficiency of Hybrid Decision Tree Algorithms in Evaluating the Academic Performance of Students. International Journal of Advanced Computer Science and Applications. 15. https://doi.org/10.14569/IJACSA.2024.0150251 .

Yang, P., Liu, Y., Luo, Y., Wang, Z., & Cai, X. (2024). Text mining and multi-attribute decision-making-based course improvement in massive open online courses. *Applied Sciences*, *14*(9), 3654. https://doi.org/10.3390/app14093654

Zihan, Song & Sung, Sang-Ha & Park, Do-Myung & Park, Byung-Kwon. (2023). All-Year Dropout Prediction Modeling and Analysis for University Students. Applied Sciences. 13. 1143. https://doi.org/10.3390/app13021143 .