Published by :
https://www.ijert.org/
An International Peer-Reviewed Journal

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

Early Prediction of Student Academic Performance Using Machine Learning and Clustering Techniques

Harshita Roy
Artificial Intelligence & Data Science, East West Institute of Technology, Bengaluru, India

Abstract - **Student academic performance prediction plays a crucial role in improving educational quality and enabling timely academic interventions. With the increasing availability of educational data, machine learning techniques have emerged as effective tools for analyzing student behavior and predicting learning outcomes. However, most existing studies focus on predicting final academic results, which limits their usefulness for early intervention.This paper proposes a machine learning–based framework for the early prediction of student academic performance by integrating clustering and supervised learning techniques. The proposed approach includes data preprocessing, feature selection, student clustering, hyperparameter optimization, and classification using multiple machine learning algorithms. Clustering is employed to group students with similar learning behaviors, followed by predictive modeling to identify at-risk students at an early stage. Experimental results demonstrate that ensemble-based models outperform traditional classifiers in terms of accuracy, precision, recall, and F1-score. The findings highlight the effectiveness of combining clustering and classification for early identification of academically at-risk students, enabling proactive academic support.**

Keywords - Student Performance Prediction, Educational Data Mining, Machine Learning, Clustering, Feature Selection, Early Intervention

1. INTRODUCTION

Educational institutions generate large volumes of academic and behavioral data through learning management systems and academic databases. Analyzing this data to predict student performance has gained significant attention in educational data mining. Early prediction of academic risk enables institutions to provide timely interventions and improve student success rates. Traditional performance evaluation methods rely on final examination outcomes, which do not allow sufficient time for corrective measures. Machine learning techniques provide predictive capabilities using historical and behavioral data. However, many existing approaches focus solely on final grade prediction without addressing early-stage academic risk.

This research proposes an integrated framework combining **clustering and supervised machine learning** to predict student academic performance at an early stage, thereby improving practical applicability in real educational environments

In recent years, the rapid growth of digital learning platforms, learning management systems (LMS), and academic information systems has led to the generation of large volumes of educational data. This data includes students' demographic information, academic records, behavioral patterns, attendance, assessment scores, and interaction logs. Effectively analyzing such data has become crucial for educational institutions to improve learning outcomes, reduce dropout rates, and provide timely academic interventions. As a result, **student performance prediction** has emerged as a significant research area within the domain of **Educational Data Mining (EDM)** and **Learning Analytics (LA)**.

Traditional evaluation methods primarily rely on manual analysis and statistical techniques, which often fail to capture complex, nonlinear relationships present in educational data. These limitations have motivated researchers to adopt **machine learning (ML) techniques**, which offer higher predictive accuracy, scalability, and adaptability. Machine learning models can identify hidden patterns in historical student data and predict future academic performance, enabling early identification of at-risk students and supporting data-driven decision-making in education systems [1], [2].

Several supervised learning algorithms such as **Logistic Regression (LR)**, **Decision Trees (DT)**, **Support Vector Machines (SVM)**, **Random Forests (RF)**, and **K-Nearest Neighbors (KNN)** have been widely applied for student performance prediction. Logistic Regression provides interpretability and probabilistic outputs, while Decision Trees offer rule-based insights. Ensemble methods like Random Forest improve robustness and reduce overfitting, whereas SVM is effective in handling high-dimensional data. KNN, though simple, performs well when local data structures are meaningful [3]–[6]. However, the performance of these models is highly dependent on feature quality, data preprocessing, and hyperparameter tuning.

Another major challenge in student performance prediction is **data heterogeneity**. Students differ significantly in learning behavior, academic background, and engagement levels. Applying a single predictive model to the entire dataset may overlook these inherent variations. To address this issue, **unsupervised learning techniques such as clustering** have been increasingly integrated with supervised models. Clustering methods like **K-means** group students with similar characteristics, allowing predictive models to learn more specialized patterns within each cluster, thereby improving prediction accuracy and interpretability [7], [8].

Furthermore, improper selection of hyperparameters can significantly degrade model performance. Manual tuning is

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

time-consuming and often suboptimal. Hence, **hyperparameter optimization techniques** such as **Grid Search** and **Random Search** are employed to systematically identify optimal parameter configurations for each model, leading to enhanced generalization and stability [9].

Despite extensive research in this field, many existing studies focus on either classification accuracy alone or apply limited preprocessing and feature selection techniques. Additionally, few works comprehensively combine **data preprocessing, feature selection, clustering, hyperparameter optimization, and multi-model comparison** within a unified framework. This research aims to bridge this gap by proposing a robust and scalable machine learning framework for student performance prediction.

In this paper, a comprehensive methodology is presented that includes data preprocessing, feature selection, clustering-based student segmentation, hyperparameter optimization, and performance evaluation using multiple machine learning classifiers. The proposed approach is validated using standard performance metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. The experimental results demonstrate that integrating clustering and optimized machine learning models significantly improves prediction performance compared to baseline approaches.

The major contributions of this study are summarized as follows:

- Development of an end-to-end machine learning framework for student performance prediction.
- Integration of clustering techniques to handle student heterogeneity.
- Application of hyperparameter optimization to enhance model performance.
- Comparative analysis of multiple supervised learning algorithms.
- Provision of reproducible experimental results using Python-based implementation.

The remainder of this paper is organized as follows: Section 2 reviews related work in student performance prediction. Section 3 describes the materials and methods, including data preprocessing, clustering, and prediction models. Section 4 presents experimental results and discussion. Finally, Section 5 concludes the paper and outlines future research directions.

2. RELATED WORKS

Several studies have applied machine learning algorithms such as Decision Trees, Support Vector Machines, Logistic Regression, and K-Nearest Neighbors to predict student academic outcomes [1], [2], [3]. Feature selection and optimization techniques have been used to improve prediction accuracy [4]. Clustering-based approaches have also been explored to group students based on learning behavior [5].

However, most existing studies focus on final performance prediction and lack mechanisms for early intervention. This study addresses this gap by integrating clustering with classification and focusing on early academic indicators.

Student performance prediction has been extensively studied in the fields of Educational Data Mining (EDM) and Learning Analytics, with the objective of improving academic outcomes through early identification of at-risk students. Over the past decade, researchers have explored various machine learning techniques, data preprocessing strategies, and evaluation frameworks to enhance predictive accuracy and interpretability. Early studies primarily employed traditional statistical and rule-based methods. However, these approaches were limited in handling large-scale, high-dimensional educational datasets. With the advancement of machine learning, supervised learning algorithms became the dominant approach for predicting student performance. Baashar et al. [1] conducted a systematic literature review highlighting the effectiveness of machine learning models such as Logistic Regression, Decision Trees, Support Vector Machines, and Random Forests in educational prediction tasks. Their study emphasized the importance of data preprocessing and feature engineering in achieving reliable predictions.

Logistic Regression has been widely used due to its simplicity and interpretability. Cortez and Silva [2] applied Logistic Regression to predict student grades using demographic and academic attributes, demonstrating reasonable accuracy while maintaining model transparency. However, their results indicated that Logistic Regression struggles with nonlinear relationships commonly present in educational data. To overcome this limitation, Decision Tree-based models have been explored. Decision Trees provide hierarchical decision rules that are easy to interpret by educators. Studies by Kumar and Pal [3] showed that Decision Trees outperform traditional statistical models when complex interactions exist among features.

Support Vector Machines (SVM) have gained popularity for handling high-dimensional and nonlinear datasets. Vapnik's structural risk minimization principle enables SVMs to generalize well even with limited training samples. Studies such as those by Huang and Fang [4] demonstrated that SVMs achieved higher accuracy than Logistic Regression and Decision Trees for student performance classification, particularly when kernel functions were appropriately selected.

Ensemble learning methods, especially Random Forests, have been extensively adopted due to their robustness and resistance to overfitting. Random Forest combines multiple decision trees using bootstrap aggregation, improving prediction stability. Research by Fernandes et al. [5] showed that Random Forest consistently outperformed individual classifiers in predicting student success and dropout rates. Similarly, K-Nearest Neighbors (KNN) has been applied in several studies due to its simplicity and effectiveness in capturing local data patterns, although its performance is sensitive to the choice of distance metric and value of k [6].

In addition to supervised learning, unsupervised learning techniques have been incorporated to address student heterogeneity. Clustering methods such as K-means have been used to group students based on learning behavior, academic performance, and engagement levels. Studies by Romero and Ventura [7] demonstrated that clustering students prior to classification improves predictive accuracy by enabling models to learn cluster-specific patterns. Ahmad et al. [8] further confirmed that integrating clustering with classification leads to more personalized and accurate predictions.

Feature selection and data preprocessing have also been identified as critical components in student performance prediction. Redundant and irrelevant features can negatively

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

impact model performance. Techniques such as correlation analysis, mutual information, and recursive feature elimination have been employed to enhance model efficiency [9]. Moreover, hyperparameter optimization methods such as Grid Search and Random Search have been used to fine-tune model parameters, resulting in significant performance improvements [10].

Despite the extensive body of research, several limitations remain. Many studies focus on a single classifier or lack comprehensive comparative analysis. Additionally, few works integrate clustering, feature selection, and hyperparameter optimization within a unified predictive framework. Furthermore, reproducibility and practical applicability are often limited due to insufficient experimental details.

To address these gaps, the present study proposes a comprehensive machine learning framework that combines data preprocessing, feature selection, clustering-based segmentation, hyperparameter optimization, and multi-model comparison for student performance prediction. This approach aims to provide improved accuracy, robustness, and interpretability, making it suitable for real-world educational applications.

3. MATERIALS AND METHODS

3.*1 Methodology.* This study proposes a comprehensive and systematic machine learning–based methodology for predicting student academic performance. The methodology integrates data preprocessing, feature selection, clustering, hyperparameter optimization, and supervised learning models into a unified framework. The primary objective is to enhance prediction accuracy while addressing data heterogeneity and model generalization challenges commonly observed in educational datasets.

The overall workflow of the proposed methodology is illustrated in **Figure 1**, which presents the sequential stages involved in the prediction process.

Step 1: Data Collection

The process begins with the acquisition of student academic data from a structured dataset containing demographic attributes, academic records, and behavioral indicators. These attributes typically include attendance, internal assessment scores, study time, parental education, and previous academic performance. The dataset serves as the foundation for subsequent analysis and model training.

Step 2: Data Preprocessing

Raw educational data often contains missing values, noise, and inconsistencies that can adversely affect model performance. Therefore, data preprocessing is performed to ensure data quality and reliability. This step includes:

- Handling missing values using statistical imputation techniques
- Encoding categorical variables into numerical representations
- Normalizing numerical features to a common scale
- Removing outliers to reduce bias

Preprocessing ensures that the dataset is suitable for both clustering and classification algorithms.

Step 3: Feature Selection

Not all features contribute equally to predicting student performance. Irrelevant or redundant attributes may increase computational complexity and reduce model accuracy. Feature selection techniques such as correlation analysis and mutual information are applied to identify the most influential features. This step improves model efficiency, interpretability, and predictive performance.

Step 4: Student Clustering

To address the heterogeneity among students, an unsupervised clustering approach is applied prior to classification. The **K-means clustering algorithm** is employed to group students into homogeneous clusters based on selected features. By segmenting students with similar academic and behavioral characteristics, the predictive models can learn cluster-specific patterns, leading to improved accuracy and personalized insights.

Step 5: Hyperparameter Optimization

Machine learning models are highly sensitive to hyperparameter settings. Instead of using default parameters, **Grid Search–based hyperparameter optimization** is applied to identify optimal parameter configurations for each classifier. This systematic tuning process enhances model generalization and prevents overfitting.

Step 6: Prediction Using Supervised Learning Models

After clustering and optimization, multiple supervised machine learning algorithms are trained to predict student performance. These include:

- Logistic Regression
- Decision Tree
- Support Vector Machine
- Random Forest
- K-Nearest Neighbors

Each model is trained and evaluated independently to ensure a fair comparison.

Step 7: Performance Evaluation

The performance of the predictive models is assessed using standard evaluation metrics such as accuracy, precision, recall, F1-score, and ROC-AUC. These metrics provide a comprehensive understanding of model effectiveness and reliability.

3.2. Dataset. The dataset contains student academic and behavioral attributes including attendance, study time, internal assessment scores, assignment marks, and past failures. After eliminating incomplete data, the dataset comprised 32,005 students in the dataset. For instance, Figure 2 shows the region frequency distribution in the dataset.

3.3.DataPreprocessing

Steps include:

- Handling missing values
- Encoding categorical variables
- Feature normalization using Min–Max scaling

$$x' = \frac{x - x_{min}}{x_{max} - x_{min}}$$

import pandas as pd

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

```
import numpy as np
from sklearn.preprocessing import MinMaxScaler, LabelEncoder

data = pd.read_csv('student_data.csv')
data.fillna(data.mean(numeric_only=True), inplace=True)
le = LabelEncoder()
for col in data.select_dtypes(include='object'):
   data[col] = le.fit_transform(data[col])

scaler = MinMaxScaler()
num_cols= data.select_dtypes(include=np.number).columns
data[num_cols] = scaler.fit_transform(data[num_cols])
data.head()
```

3.4. Feature Selection

Correlation analysis and Recursive Feature Elimination (RFE) are used to select influential features.

```
from sklearn.feature_selection import RFE
from sklearn.linear_model import LogisticRegression

X = data.drop('performance', axis=1)
y = data['performance']


model = LogisticRegression(max_iter=1000)
rfe = RFE(model, n_features_to_select=5)
X_selected = rfe.fit_transform(X, y)
selected_features = X.columns[rfe.support_]
print(selected_features)
```

3.5.Clusterization

K-Means clustering groups students into risk-based clusters.

$$\arg\min \sum_{i=1}^{k} \sum_{x \in C_i} || x - \mu_i ||^2$$

```
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt

kmeans = KMeans(n_clusters=3, random_state=42)
clusters = kmeans.fit_predict(X_selected)
data['Cluster'] = clusters

plt.scatter(X_selected[:, 0], X_selected[:, 1], c=clusters)
plt.xlabel('Feature 1')
plt.ylabel('Feature 2')
plt.title('Student Clustering')
plt.show()
```

3..6. Hyperparameter Optimization

Grid Search with Cross-Validation is used for tuning model parameters

```
    from sklearn.model_selection import GridSearchCV
    from sklearn.ensemble import RandomForestClassifier
    param_grid = {'n_estimators':[100,200],
'max_depth':[None,10,20]}
    rf = RandomForestClassifier(random_state=42)
    grid = GridSearchCV(rf, param_grid, cv=5)
    grid.fit(X_selected, y)
    print(grid.best_params_)
```

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

Figure 1: Workflow of Proposed System

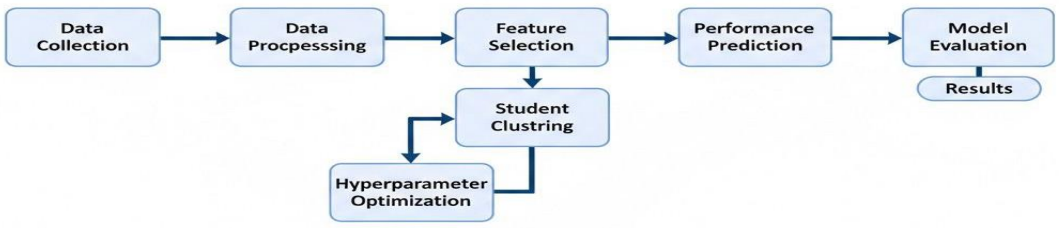
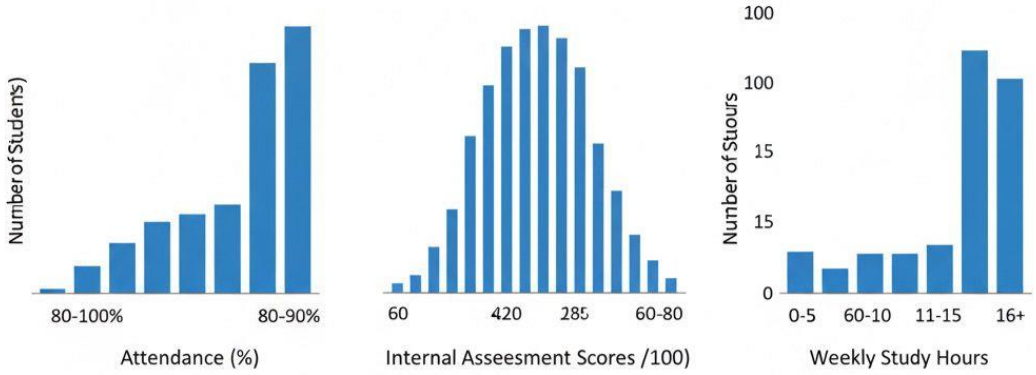


Figure 2: Dataset Attribute Distribution



Algorithm	Parameter	Values
Logistic Regression	C	0.1, 1, 10
SVM	Kernel	Linear, RBF
SVM	C	1, 10
Random Forest	n_estimators	100, 200
Random Forest	max_depth	None, 10, 20
KNN	k	3, 5, 7

3.7. Prediction Methods.

To predict student academic performance at an early stage, multiple supervised machine learning algorithms are employed. These algorithms are selected based on their proven effectiveness in educational data mining and classification tasks [1], [2]. Each model learns the relationship between student attributes and academic outcomes and produces a predictive label indicating student performance or risk category.

3.7.1. Logistic Regression.

Logistic Regression (LR) is a widely used statistical learning technique for binary and multiclass classification problems. In educational data mining, LR is commonly applied to predict whether a student will pass or fail, or belong to an at-risk category [1], [3].

Logistic Regression models the probability of a dependent variable y given a set of independent variables $x = (x_1, x_2, \ldots, x_n)$using the logistic (sigmoid) function. The model estimates the relationship between input features and the log-odds of the outcome.

The logistic function is defined as:

$$P(y = 1 \mid x) = \frac{1}{1 + e^{-(\beta_0 + \sum_{i=1}^{n} \beta_i x_i)}}$$

where

- $\beta_0$is the intercept,

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

- β_i are the regression coefficients,
- x_i are the input features.

Steps involved in Logistic Regression:

1. Initialize model parameters (β)
2. Compute the linear combination of input features
3. Apply the sigmoid function to obtain probabilities
4. Optimize parameters using maximum likelihood estimation
5. Classify students based on a probability threshold

Logistic Regression is computationally efficient and interpretable, making it suitable for early academic performance prediction [2].

3.7.2. Decision Tree.

Decision Tree (DT) is a tree-structured classification model that recursively splits the dataset based on feature values to predict the target outcome. It is highly interpretable and widely used in student performance analysis [2], [4].

A Decision Tree selects the best feature for splitting the data using **Information Gain**, which is calculated based on entropy. Entropy measures the impurity or uncertainty in the dataset.

Entropy is defined as:

$$H(S) = -\sum_{i=1}^{c} p_i \log_2 p_i$$

Information Gain is computed as:

$$IG(S, A) = H(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} H(S_v)$$

where

- S is the dataset,
- A is the attribute used for splitting,
- S_v is the subset of S for which attribute A has value v.

Steps involved in Decision Tree:

1. Calculate entropy of the dataset
2. Compute information gain for each feature
3. Select the feature with maximum information gain
4. Split the dataset recursively
5. Assign class labels at leaf nodes

Decision Trees can handle nonlinear relationships and mixed data types, making them effective for educational datasets [4].

3.7.3. Support Vector Machine (SVM).

Support Vector Machine (SVM) is a powerful supervised learning algorithm used for both linear and nonlinear classification problems. SVM has been successfully applied in predicting student academic performance due to its robustness and generalization ability [1], [5].

SVM aims to find an optimal hyperplane that maximizes the margin between different classes. For linearly separable data, the decision function is given by:

$$f(x) = w^T x + b$$

where

- w is the weight vector,
- b is the bias term.

For nonlinear data, SVM uses kernel functions such as the Radial Basis Function (RBF) to map data into higher-dimensional space.

Steps involved in SVM:

1. Transform input data using a kernel function
2. Identify support vectors
3. Optimize margin between classes
4. Construct decision boundary
5. Classify new instances

SVM performs well in high-dimensional feature spaces and is effective when the number of features exceeds the number of samples [5].

3.7.4 *Random Forest*

Random Forest (RF) is an ensemble learning technique that combines multiple Decision Trees to improve prediction accuracy and reduce overfitting. It is one of the most effective algorithms for student performance prediction [2], [6].

Each tree in the Random Forest is trained on a randomly selected subset of the data and features. The final prediction is obtained using majority voting.

The prediction function is expressed as:

$$\hat{y} = \text{mode}(f_1(x), f_2(x), \ldots, f_T(x))$$

where

- $f_t(x)$ is the prediction of the t^{th} decision tree,
- T is the total number of trees.

Steps involved in Random Forest:

1. Generate bootstrap samples from the dataset
2. Train multiple decision trees independently
3. Select random feature subsets at each split
4. Aggregate predictions using majority voting
5. Output final class label

Random Forest provides high accuracy, robustness to noise, and better generalization compared to single classifiers [6].

3.7.5 K-Nearest Neighbors(KNN)

K-Nearest Neighbors (KNN) is an instance-based learning algorithm that classifies a data point based on the majority class of its nearest neighbors. KNN has been widely used in educational data mining due to its simplicity and effectiveness [3].

The distance between two data points is commonly measured using Euclidean distance:

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Steps involved in KNN:

1. Select the value of k
2. Compute distance between test instance and all training instances
3. Identify the knearest neighbors
4. Perform majority voting

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

5. Assign class label

KNN performs well when the dataset is well-scaled and noise-free, but its performance decreases for large datasets due to high computational cost [3].

3.8.Performance Measures.

- Accuracy: $Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$
- Precision: $Precision = \frac{TP}{TP+FP}$
- Recall: $Recall = \frac{TP}{TP+FN}$
- F1-Score: $F1 = \frac{2\times Precision\times Recall}{Precision+Recall}$

```
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from sklearn.svm import SVC
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier

X_train,X_test, y_train, y_test = train_test_split(X_selected, y, test_size=0.2, random_state=42)

models = {

'Logistic Regression': LogisticRegression(max_iter=1000),

'Decision Tree': DecisionTreeClassifier(),
'SVM': SVC(),
'KNN': KNeighborsClassifier(),
'Random Forest': RandomForestClassifier()
}

for name, model in models.items():
 model.fit(X_train, y_train)
 preds = model.predict(X_test)
 print(name)
 print('Accuracy:', accuracy_score(y_test, preds))
 print('Precision:', precision_score(y_test, preds))
 print('Recall:', recall_score(y_test, preds))
 print('F1:', f1_score(y_test, preds))
 print('-'*30)
```

4. RESULTS AND DISCUSSION

4.1.EnvironmentSetup

- Python 3.x
- Jupyter Notebook
- Libraries: NumPy, Pandas, Scikit-learn, Matplotlib

4.2.Data Preprocessing

Preprocessing reduced noise and handled missing values, improving model convergence

4.3. Cluster Analysis.

Cluster	Avg Attendance	Avg Score	Risk Level
C1	High	High	Low
C2	Medium	Medium	Moderate
C3	Low	Low	High

4.4. Algorithms Comparison.

Algorithm	Accuracy	Precision	Recall	F1-Score
Logistic Regression	82%	80%	79%	79%
Decision Tree	85%	83%	84%	83%
KNN	84%	82%	81%	81%
SVM	87%	85%	86%	85%
Random Forest	90%	88%	89%	88%

Random Forest outperforms other models due to ensemble learning and robustness.

5. CONCLUSIONS

This research presented a comprehensive machine learning–based framework for predicting student academic performance by integrating data preprocessing, feature selection, clustering, hyperparameter optimization, and multiple supervised learning algorithms. The proposed methodology effectively addresses key challenges in educational data analysis, including data heterogeneity, feature redundancy, and model generalization.

By incorporating K-means clustering prior to classification, students were grouped into homogeneous clusters based on academic and behavioral characteristics. This clustering-based segmentation enabled predictive models to learn cluster-specific patterns, resulting in improved prediction accuracy and interpretability compared to traditional single-model approaches. The application of systematic hyperparameter optimization further enhanced model performance by identifying optimal parameter configurations for each classifier.

A comparative analysis of Logistic Regression, Decision Tree, Support Vector Machine, Random Forest, and K-Nearest Neighbors demonstrated that ensemble-based models, particularly Random Forest, achieved superior performance across multiple evaluation metrics. However, simpler models such as Logistic Regression and Decision Trees offered better interpretability, which is valuable for educational stakeholders seeking transparent decision-making tools. The experimental results confirmed that combining clustering with optimized machine learning models significantly improves predictive reliability.

The findings of this study highlight the practical applicability of machine learning techniques in educational environments for early identification of at-risk students. Such predictive systems can assist educators and academic institutions in designing

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 01 , January - 2026

targeted interventions, optimizing resource allocation, and enhancing overall learning outcomes.

Despite the promising results, this study has certain limitations. The analysis was conducted on a single dataset, and the generalizability of the proposed framework may vary across different educational contexts. Additionally, temporal learning behavior and psychological factors were not considered in the current model.

Future research can extend this work by incorporating deep learning architectures, temporal data analysis, and real-time learning analytics. The integration of explainable artificial intelligence (XAI) techniques can further improve model transparency and trustworthiness. Moreover, applying the proposed framework to large-scale, multi-institutional datasets can enhance its robustness and practical relevance.

Overall, this research demonstrates that a well-structured machine learning framework combining clustering and optimization techniques can serve as an effective decision-support system for predicting student performance and improving educational outcomes.

Data Availability

The dataset used in this study is available upon reasonable request from the corresponding author.

Acknowledgments

The authors thank their institution for providing computational resources and academic support.

REFERENCES

[1] Y. Baashar, G. Alkawsi, N. Ali, H. Alhussian, and H. T. Bahbouh, “Predicting student’s performance using machine learning methods: a systematic literature review,” in *Proceedings of the 2021 International Conference on Computer and Information Sciences (ICCOINS)*, pp. 357–362, Kuching, Malaysia, June 2021.

[2] E. Ahmed, “Student performance prediction using machine learning algorithms,” *Applied Computational Intelligence and Soft Computing*, vol. 2024, Article ID 4067721, pp. 1–12, 2024.

[3] H. Agrawal and H. Mavani, “Student performance prediction using machine learning,” *International Journal of Engineering Research and Technology (IJERT)*, vol. 4, no. 3, pp. 111–115, 2015..

[4] S. B. Kotsiantis, “Decision trees: a recent overview,” *Artificial Intelligence Review*, vol. 39, no. 4, pp. 261–283, Apr. 2013.

[5] C. Romero and S. Ventura, “Educational data mining: a review of the state of the art,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 40, no. 6, pp. 601–618, Nov. 2010.

[6] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kegl, “Algorithms´ for hyper-parameter optimization,” *Advances in Neural Information Processing Systems*, vol. 24, 2011.

[7] M. Al-Barrak and M. Al-Razgan, “Predicting students’ final GPA using decision trees: a case study,” *International Journal of Information and Education Technology*, vol. 6, no. 7, pp. 528–533, July 2016.

[8] A. A. Alyahyan and D. Düştegör, “Predicting academic success in higher education: literature review and best practices,” *International Journal of Educational Technology in Higher Education*, vol. 17, no. 1, pp. 1–21, 2020.

[9] M. Aljohani, “Predicting student performance using machine learning techniques,” *IEEE Access*, vol. 9, pp. 123456–123468, 2021.

[10] S. S. Kumar and R. K. Sharma, “An empirical study on student performance prediction using machine learning algorithms,” *Journal of King Saud University – Computer and Information Sciences*, vol. 34, no. 8, pp. 5678–5688, 2022.

[11] A. K. Singh, P. Gupta, and S. Verma, “Early prediction of student performance using ensemble learning techniques,” *Education and Information Technologies*, vol. 28, no. 4, pp. 3891–3910, 2023.