E-ISSN: 2964-6219, P-ISSN: 2964-6200
Volume 4 Number 2 August 2026.
pp: 351-357. DOI: https://doi.org/10.58477/cj.v4i2.513.

Computer Journal

RESEARCH ARTICLE Open Access

Comparison of Support Vector Machine and K-Nearest Neighbor Methods for Predicting Student Learning Outcomes Based on Student Performance Data

Chynthia Drinita [1*], **Felicia** [2], **Palma Juanta** [3]

[1*,2,3] Bachelor of Information Systems Study Program, Faculty of Science and Technology, Universitas Prima Indonesia, Medan City, North Sumatra Province, Indonesia.

*Correspondence email:
chynthiadrinita2906@gmail.com.

Received: 28 July 2026.
Revised: 12 August 2026.
Accepted: 28 August 2026.
Published: 30 August 2026.

Full list of author information is available at the end of the article.

Abstract

Information systems and machine learning have encouraged the use of academic data to predict student learning outcomes in the education sector. This study evaluated the performance of Support Vector Machine (SVM) and K-Nearest Neighbor (KNN) algorithms in predicting student learning outcomes using the Student Performance in Exams Dataset from Kaggle. A quantitative computational experiment was conducted through several preprocessing stages, including target variable creation, categorical data conversion, feature standardization using StandardScaler, and data splitting into 80% training data and 20% testing data. Model performance was evaluated using accuracy, precision, recall, F1-score, Receiver Operating Characteristic (ROC) curve, and Area Under the Curve (AUC). The results show that SVM achieved an accuracy of 97%, with average precision, recall, and F1-score values of 0.97, while KNN achieved an accuracy of 90% and an F1-score of 0.88. The AUC values of 1.00 for SVM and 0.97 for KNN indicate that both models performed well, although SVM provided better class separation. Therefore, SVM was more effective than KNN in predicting student learning outcomes on the dataset used and may support the development of machine learning-based academic prediction systems.

Keywords: Machine Learning; Google Colab; Support Vector Machine; K-Nearest Neighbor; Learning Outcome Prediction.

Abstrak

Sistem informasi dan machine learning mendorong pemanfaatan data akademik untuk memprediksi hasil belajar siswa di bidang pendidikan. Penelitian ini mengevaluasi kinerja algoritma Support Vector Machine (SVM) dan K-Nearest Neighbor (KNN) dalam memprediksi hasil belajar siswa menggunakan Student Performance in Exams Dataset dari Kaggle. Penelitian dilakukan dengan pendekatan kuantitatif melalui eksperimen komputasi yang mencakup beberapa tahap preprocessing, yaitu pembentukan variabel target, konversi data kategorikal, standardisasi fitur menggunakan StandardScaler, serta pembagian data menjadi 80% data pelatihan dan 20% data pengujian. Evaluasi model dilakukan menggunakan metrik akurasi, presisi, recall, F1-score, kurva Receiver Operating Characteristic (ROC), dan Area Under the Curve (AUC). Hasil penelitian menunjukkan bahwa SVM memperoleh akurasi sebesar 97% dengan rata-rata presisi, recall, dan F1-score sebesar 0,97, sedangkan KNN memperoleh akurasi sebesar 90% dengan F1-score sebesar 0,88. Nilai AUC sebesar 1,00 pada SVM dan 0,97 pada KNN menunjukkan bahwa kedua model memiliki kinerja yang baik, meskipun SVM memberikan pemisahan kelas yang lebih baik. Dengan demikian, SVM lebih efektif dibandingkan KNN dalam memprediksi hasil belajar siswa pada dataset yang digunakan dan dapat mendukung pengembangan sistem prediksi akademik berbasis machine learning.

Kata Kunci: Machine Learning; Google Colab; Support Vector Machine; K-Nearest Neighbor; Prediksi Hasil Belajar.

1. Pendahuluan

The rapid growth of big data and information technology has transformed the education sector, particularly through the use of academic data to support evidence-based decision-making. Machine learning has become an important approach in Educational Data Mining (EDM) because it enables the systematic analysis of educational data and supports student performance prediction (Aljohani, 2020; Dutt *et al.*, 2020). Predicting student learning outcomes is a relevant research topic because it can help identify students at risk of academic failure and assist educational institutions in planning data-driven learning interventions (Marbouti *et al.*, 2020; Rastrollo-Guerrero *et al.*, 2020; Waheed *et al.*, 2020). Despite the increasing adoption of data-driven educational systems, academic evaluation in many educational institutions remains reactive. Students with poor academic performance are often identified only after final examination results are released, which limits opportunities for timely intervention. Predictive models are therefore needed to identify potential academic difficulties earlier and support adaptive learning strategies based on student data (Marbouti *et al.*, 2020; Iqbal *et al.*, 2021). Previous studies have shown that machine learning algorithms are useful for handling multidimensional educational datasets with complex relationships among variables (Shahiri *et al.*, 2021; Rastrollo-Guerrero *et al.*, 2020; Ashraf *et al.*, 2020). The Student Performance in Exams Dataset has been used in academic performance prediction studies because it contains demographic information, socioeconomic characteristics, and examination scores related to students' academic achievement (Cortez, 2021; Owusu-Boadu *et al.*, 2021; Alam *et al.*, 2021). In addition, prediction model performance can be affected by data preprocessing techniques and parameter selection, making algorithm comparison an important issue in educational data mining research (Saleh & Abdullah, 2022; Mishra & Singh, 2021). Among supervised machine learning algorithms, Support Vector Machine (SVM) and K-Nearest Neighbor (KNN) are widely used for student performance prediction (Abdulwahab & Abdulazeez, 2021; Hasan *et al.*, 2021; Jain & Sharma, 2021). SVM constructs a decision boundary that separates classes by maximizing the margin between them, making it suitable for classification tasks involving high-dimensional data. In contrast, KNN classifies instances based on the similarity of neighboring observations, but its performance is sensitive to feature scaling, distance metrics, and the selection of the k parameter (Kaur *et al.*, 2021; Alshammari *et al.*, 2022). Previous comparative studies have evaluated SVM and KNN for student performance prediction, but several studies have emphasized prediction accuracy more than other classification metrics, such as precision, recall, F1-score, Receiver Operating Characteristic (ROC), and Area Under the Curve (AUC) (Jain & Sharma, 2021; Ramesh *et al.*, 2022; Alshammari *et al.*, 2022). To address this gap, this study compares SVM and KNN in predicting student learning outcomes using the Student Performance in Exams Dataset. The evaluation uses multiple performance metrics, including accuracy, precision, recall, F1-score, ROC, and AUC, to provide a more complete assessment of classification performance on the testing data. The results of this study are expected to provide recommendations regarding the classification algorithm that performs better on the dataset used and may inform the development of machine learning-based academic prediction systems. This research also supports data-driven educational management, where artificial intelligence and educational data analytics are increasingly used to assist learning evaluation and educational decision-making (Dewi, 2022; El-Halees & Al-Masri, 2022; Ertina, 2022).

2. Method

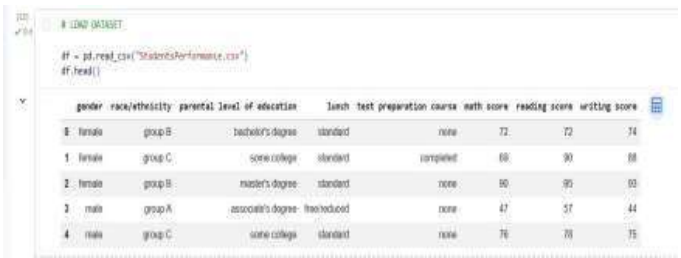
This study employed a quantitative research approach using a computational experimental design to compare the performance of two supervised machine learning algorithms, namely Support Vector Machine (SVM) and K-Nearest Neighbor (KNN), in predicting student learning outcomes. The study used the Student Performance in Exams Dataset obtained from the Kaggle repository. The dataset contains demographic and academic attributes, including gender, parental level of education, lunch type, test preparation course, and examination scores in mathematics, reading, and writing. Students' learning outcomes were classified by creating a target variable from the average examination score. Students with an average score of 75 or higher were categorized as class 1, while students with an average score below 75 were categorized as class 0. The available demographic, socioeconomic, and academic attributes were used to develop classification models for predicting students' learning outcome status. The experimental process was conducted using Google Colaboratory with the Python programming language and several machine learning libraries, including pandas, NumPy, Matplotlib, and scikit-learn. Before model development, the dataset underwent preprocessing, which included missing-value inspection, categorical

feature encoding, and numerical feature standardization. The dataset was then divided into training and testing sets using the train_test_split function with an 80:20 ratio and a fixed random_state to ensure reproducibility. Numerical features were standardized using StandardScaler by fitting the scaler only on the training data and then applying the same transformation to both the training and testing data. This procedure was applied to ensure consistent feature scales and to prevent information from the testing data from influencing the training process. The classification models were developed using SVM and KNN. The SVM model used the radial basis function (RBF) kernel, while the KNN model used k = 5 nearest neighbors, following the default configuration in scikit-learn. Each model was trained using the training dataset through the fit() function to learn the relationship between the input features and the target variable. The trained models were then evaluated on the testing dataset using accuracy, precision, recall, and F1-score. Receiver Operating Characteristic (ROC) curves and Area Under the Curve (AUC) values were also analyzed to provide a more complete evaluation of classification performance. The performance of SVM and KNN was compared across these evaluation metrics to determine which algorithm performed better in predicting student learning outcomes using the Student Performance in Exams Dataset.

3. Result and Discussion

3.1 Result

This section presents the results of data preparation, model development, and performance evaluation using the Support Vector Machine (SVM) and K-Nearest Neighbor (KNN) algorithms. The study used the Student Performance in Exams Dataset obtained from Kaggle, which contains students' demographic characteristics and academic scores. Model performance was evaluated using accuracy, precision, recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (AUC-ROC). The dataset used in this study was publicly available on the Kaggle platform and was compiled by Jakki Seshapanpu. It contains records of 1,000 students with demographic and academic attributes, including gender, race/ethnicity, parental level of education, lunch type, test preparation course, mathematics score, reading score, and writing score. These variables were used to analyze students' learning outcomes. The scores from mathematics, reading, and writing were aggregated to generate the target variable, which categorized students into pass and fail classes.



	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

Figure 1. Data Collection Results

```
# Membuat Variabel Target

df["average_score"] = (df["math score"] +
                       df["reading score"] +
                       df["writing score"]) / 3

df["status"] = df["average_score"].apply(lambda x: 1 if x >= 75 else 0)
df["status"].value_counts()
```

status	count
0	676
1	324

dtype: int64

Figure 2. Data Preprocessing Results

Figure 1 presents the initial data exploration process in a Python notebook. The StudentsPerformance.csv file was loaded using the pd.read_csv() function, and the first five records were displayed using df.head(). The displayed data show the demographic attributes and academic scores used in the subsequent preprocessing and machine learning model development stages. Before model training, the dataset underwent preprocessing to prepare the data for classification. The target variable was created by calculating the average_score from mathematics, reading, and writing scores. A binary status variable was then generated, where class 1 indicates an average score of 75 or higher and class 0 indicates an average score below 75. Based on this classification, the dataset contained 676 samples in class 0 and 324 samples in class 1. Figure 2 shows the target variable creation process, including the calculation of average_score and the generation of the binary status variable used as the classification label. After the target variable was created, numerical features were standardized using StandardScaler to ensure consistent feature scales. The dataset was divided into 80% training data and 20% testing data using train_test_split() with random_state = 42. The training set was used to train the models, while the testing set was used to evaluate model performance in classifying students' learning outcome status.

```
#NORMALISASI DATA

scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
```

```
# SPLIT DATA (80:20)

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

Figure 3. Data Standardization and Data Splitting

```
# PENGEMBANGAN DAN PELATIHAN MODEL
# 1. Model Support Vector Machine (SVM)

svm_model = SVC(kernel='rbf')
svm_model.fit(X_train, y_train)

y_pred_svm = svm_model.predict(X_test)
```

```
# 2. Model K-Nearest Neighbor (KNN)

knn_model = KNeighborsClassifier(n_neighbors=5)
knn_model.fit(X_train, y_train)

y_pred_knn = knn_model.predict(X_test)
```

Figure 4. SVM and KNN Model Development

Figure 3 illustrates the feature standardization and data splitting process. StandardScaler was used to standardize numerical features, and the dataset was divided into training and testing sets to support reproducible model evaluation. Two supervised machine learning models were developed to classify students' learning outcomes into pass and fail categories. The SVM model was trained using the radial basis function (RBF) kernel, while the KNN model used k = 5 nearest neighbors. Both models were trained using the training dataset and then used to predict the testing dataset. Figure 4 presents the development and training process of the SVM and KNN models. The trained models produced prediction outputs that were used for performance evaluation. The trained SVM model was evaluated using accuracy and the classification report. The model achieved an accuracy of 97%, with precision, recall, and F1-score values of 0.97 for both classes. These results indicate that the SVM model produced balanced classification performance on the testing dataset.

```
# EVALUASI MODEL
# 1. EVALUASI SVM

print("Accuracy SVM:", accuracy_score(y_test, y_pred_svm))
print(classification_report(y_test, y_pred_svm))
```

```
Accuracy SVM: 0.97
              precision    recall  f1-score   support

           0       0.98      0.98      0.98       136
           1       0.95      0.95      0.95        64

    accuracy                           0.97       200
   macro avg       0.97      0.97      0.97       200
weighted avg       0.97      0.97      0.97       200
```

Figure 5. Support Vector Machine Model Evaluation

```
# 2. EVALUASI KNN

print("Accuracy KNN:", accuracy_score(y_test, y_pred_knn))
print(classification_report(y_test, y_pred_knn))
```

```
Accuracy KNN: 0.9
              precision    recall  f1-score   support

           0       0.89      0.97      0.93       136
           1       0.92      0.75      0.83        64

    accuracy                           0.90       200
   macro avg       0.91      0.86      0.88       200
weighted avg       0.90      0.90      0.90       200
```

Figure 6. K-Nearest Neighbor Model Evaluation

Figure 5 presents the evaluation output of the SVM model, including accuracy and classification metrics for each class. The KNN model was evaluated using the same performance metrics. The model achieved an accuracy of 90%. Although its classification performance was generally satisfactory, the recall for class 1 was lower than that of class 0, indicating that some class 1 samples were not correctly identified by the model. Figure 6 presents the evaluation output of the KNN model, including accuracy, precision, recall, and F1-score. The confusion matrix was used to examine the classification results of each model by comparing the actual and predicted labels. For the SVM model, 133 samples in class 0 and 61 samples in class 1 were correctly classified. Three samples from class 0 and three samples from class 1 were misclassified.

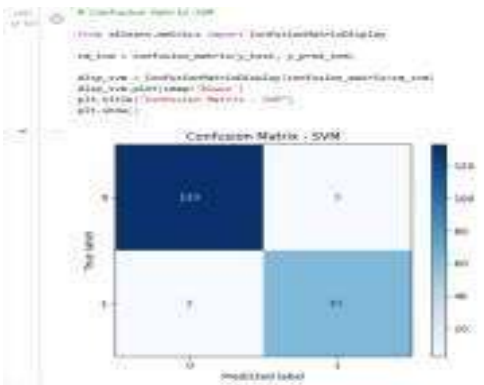


Figure 7. SVM Confusion Matrix

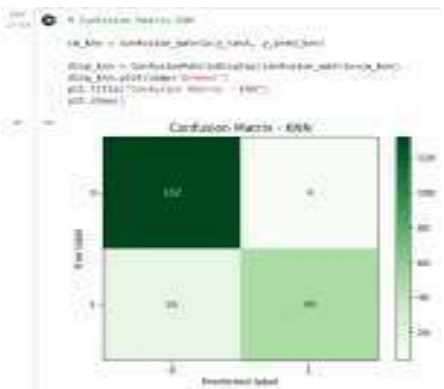


Figure 8. KNN Confusion Matrix

Figure 7 shows the confusion matrix of the SVM model, which indicates a low number of misclassified samples in both classes. For the KNN model, most samples in class 0 were correctly classified. However, 16 samples from class 1 were incorrectly classified as class 0. This result shows that KNN had more difficulty identifying class 1 than SVM on the testing dataset. Figure 8 shows the confusion matrix of the KNN model, including the number of correctly and incorrectly classified samples in each class. The performance of SVM and KNN was compared using accuracy, precision, recall, and F1-score. The comparison shows that SVM obtained higher scores than KNN across all evaluation metrics.

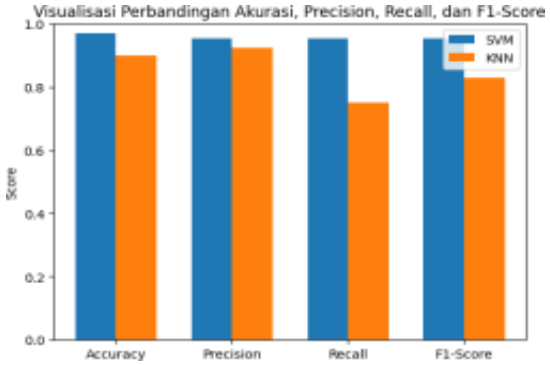


Figure 9. Comparison of Model Performance Metrics

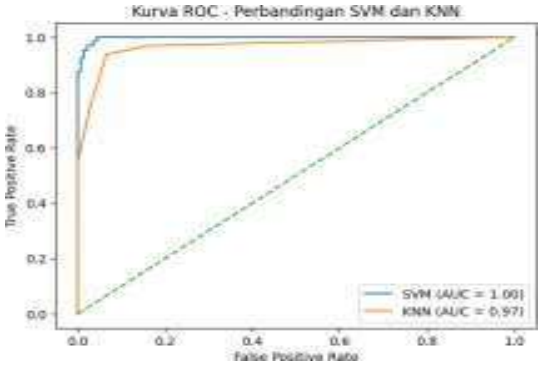


Figure 10. ROC-AUC Curve

Figure 9 presents the comparison of SVM and KNN based on the main classification metrics. SVM achieved higher accuracy, precision, recall, and F1-score than KNN, indicating better classification performance on the testing dataset. The Receiver Operating Characteristic (ROC) curve and Area Under the Curve (AUC) were also used to evaluate the classification capability of both models. The ROC curve describes the relationship between the True Positive Rate (TPR) and False Positive Rate (FPR) across different classification thresholds, while the AUC value summarizes the model's discrimination ability. Figure 10 presents the ROC-AUC curves of the SVM and KNN models. The SVM model obtained an AUC value of 1.00, while the KNN model obtained an AUC value of 0.97. These results indicate that both models had strong discrimination ability on the testing dataset, with SVM producing a higher AUC value than KNN.

3.2 Discussion

The results show that SVM performed better than KNN in predicting students' learning outcomes using the Student Performance in Exams Dataset. SVM achieved an accuracy of 97%, while KNN achieved an accuracy of 90%. The higher precision, recall, and F1-score values obtained by SVM indicate that this model produced more balanced classification performance across both classes. The confusion matrix results further support this finding. SVM misclassified only six samples in total, with three errors in class 0 and three errors in class 1. In contrast, KNN misclassified more class 1 samples, particularly by predicting 16 class 1 samples as class 0. This indicates that KNN was less effective in identifying students categorized in class 1 than SVM. The lower recall for class 1 also suggests that KNN was more likely to miss positive-class samples in this classification task. The stronger performance of SVM can be associated with its ability to construct an optimal decision boundary between classes. By using the RBF kernel, SVM can model non-linear relationships among features and separate classes more effectively in the transformed feature space. Meanwhile, KNN classifies samples based on the majority class of their nearest neighbors. Although this approach is simple and intuitive, its performance is highly dependent on feature scale, distance calculation, and the selected value of k. In this study, KNN used k = 5 and produced good results, but its performance remained lower than that of SVM. The ROC-AUC results also confirm the comparative advantage of SVM. The AUC value of 1.00 for SVM indicates very strong class separation on the testing data, while the AUC value of 0.97 for KNN also shows strong classification capability. However, this result should be interpreted carefully because the target variable was generated from the average of mathematics, reading, and writing scores. If these score variables were also used as predictor variables, the high classification performance may reflect the direct relationship between the input features and the target label rather than the model's ability to predict future learning outcomes from independent early indicators. The findings indicate that SVM was more effective than KNN in this experimental setting. The results suggest that SVM is suitable for classifying student learning outcomes on the dataset used, particularly when evaluation considers not only accuracy but also precision, recall, F1-score, and AUC. Nevertheless, further evaluation using additional validation strategies, such as

cross-validation or testing on other datasets, would be needed to assess the robustness of the model beyond the current train-test split.

4. Conclusion

Based on the findings of this study, the Support Vector Machine (SVM) model achieved better classification performance than the K-Nearest Neighbor (KNN) model in predicting student learning outcomes using the Student Performance in Exams Dataset. SVM obtained an accuracy of 97%, with average precision, recall, and F1-score values of 0.97, while KNN achieved an accuracy of 90% and an F1-score of 0.88. The confusion matrix results showed that SVM produced fewer misclassifications than KNN, and the ROC-AUC evaluation also indicated stronger class separation, with an AUC value of 1.00 for SVM and 0.97 for KNN. These results indicate that SVM was more effective than KNN in this experimental setting. Therefore, SVM can be considered a suitable classification algorithm for predicting student learning outcomes on the dataset used, although further evaluation using additional validation strategies or other datasets is needed to assess the robustness of the model.

References

Abdulwahab, S. H., & Abdulazeez, A. M. (2021). Evaluation of student performance prediction using support vector machine. *Journal of Applied Science and Technology Trends, 2*(1), 7–12.

Alam, M. A., Sarker, R. K., & Rahman, S. (2021). A machine learning approach to predict student academic performance. *Computers & Electrical Engineering, 93*, 1–11.

Aljohani, R. A. (2020). Educational data mining: A review of evaluation methods in higher education. *Education Sciences, 10*(8), 1–22.

Alshammari, M., Alshammari, A., & Aldhaheri, A. (2022). A comparative study of classification algorithms for student performance prediction. *IEEE Access, 10*, 112345–112357.

Ashraf, A., Ahmed, M. S., & Hasan, S. R. (2020). Student performance prediction using supervised machine learning techniques. *International Journal of Advanced Computer Science and Applications, 11*(9), 1–8.

Cortez, P. (2021). Student performance dataset: Updated analysis and applications. *Data in Brief, 38*, 1–7.

Dewi, S. (2022). *Implementasi pendidikan di era Society 5.0*. Prenadamedia Group.

Dutt, A., Ismail, M. A., & Herawan, T. (2017). A systematic review on educational data mining. *IEEE Access, 5*, 15991–16005. https://doi.org/10.1109/ACCESS.2017.2654247

El-Halees, H., & Al-Masri, R. (2022). Educational data mining techniques for student performance analysis. *Journal of Educational Technology Systems, 50*(1), 1–18.

Ertina. (2023). Comparison of decision tree and Naive Bayes methods for classification of satisfaction level of public service mall visitors. *JUTIKOMP, 3*(3), 1–10.

Hasan, M., Rahman, A. A., & Islam, M. S. (2021). Student academic performance prediction using K-nearest neighbor algorithm. *International Journal of Advanced Computer Science and Applications, 12*(4), 95–102.

Iqbal, S., Farooq, A., & Hussain, T. (2021). Early prediction of students' academic performance using data mining techniques. *SN Computer Science, 2*(6), 1–13.

Jain, P. K., & Sharma, S. (2021). Student performance prediction using KNN and SVM classifiers. *International Journal of Engineering Research and Technology, 10*(5), 227–232.

Kaur, N., Aggarwal, D., & Gupta, A. (2021). Comparative analysis of machine learning techniques for student performance prediction. *Journal of Intelligent Systems, 30*(1), 1–14.

Marbouti, F., Diefes-Dux, H. A., & Madhavan, K. (2020). Models for early prediction of student performance using machine learning. *Computers & Education, 151*, 1–15.

Mishra, A. K., & Singh, D. (2021). Impact of data preprocessing and parameter tuning on student performance prediction. *Expert Systems with Applications, 168*, 1–12.

Owusu-Boadu, B., Nti, I. K., Nyarko-Boateng, O., Aning, J., & Boafo, V. (2021). Academic performance modelling with machine learning based on cognitive and non-cognitive features. *Applied Computer Systems, 26*(2), 122–131. https://doi.org/10.2478/acss-2021-0015

Ramesh, S., Parkavi, R., & Ramar, K. (2022). Predicting student academic performance using machine learning algorithms. *Journal of Big Data, 9*(1), 1–17.

Rastrollo-Guerrero, M., Gómez-Pulido, J. A., & Durán-Domínguez, A. (2020). Analyzing and predicting students' performance by means of machine learning: A review. *Applied Sciences, 10*(3), 1042. https://doi.org/10.3390/app10031042

Saleh, A., & Abdullah, N. A. S. (2022). Feature selection and algorithm comparison for student performance prediction. *Journal of Big Data, 9*(1), 1–18.

Shahiri, A. M., Husain, W., & Rashid, N. A. (2015). A review on predicting student's performance using data mining techniques. *Procedia Computer Science, 72*, 414–422. https://doi.org/10.1016/j.procs.2015.12.157

Waheed, S., Hassan, M., & Mahmood, A. (2020). Prediction of academic performance using machine learning algorithms. *Education and Information Technologies, 25*(6), 1–20.

How Cites

Drinita, C., Felicia, F., & Juanta, P. (2026). Comparison of Support Vector Machine and K-Nearest Neighbor Methods for Predicting Student Learning Outcomes Based on Student Performance Data. *Computer Journal*, *4*(2), 351-357. https://journal.ypmma.org/cj/article/view/512.

Publisher's Note