An Integrated Interpretable Performance Prediction Model with Dynamically Optimized Attributes

Meenakshi Devi
Department of Computer Science & Applications,
Kurukshetra University, Kurukshetra,
Haryana, India

Rakesh Kumar
Department of Computer Science & Applications,
Kurukshetra University, Kurukshetra,
Haryana, India

ABSTRACT
The increasing availability of educational data has opened new possibilities for applying machine learning. It helps improve student learning outcomes. This paper examines the application of an integrated machine learning approach to an educational dataset. It keeps the focus on key modeling attributes that influence student performance. The emphasis is on selecting only meaningful attributes to reduce irrelevant information. To ensure stable, reliable modeling, the model validation is incorporated in the integrated approach. The model evaluation using cross-validation establishes confidence in the results' reliability. An optimization technique follows cross-validation. It searches the relevant attributes, thereby simplifying the model and making it easier to interpret. In addition to evaluating predictive performance metrics, model interpretation is included for the selected attributes. The interpretation analysis enables a better understanding of the relevant attributes across different models. SHAP is used to illustrate the contribution of individual attributes to prediction outcomes. The findings report that combining validation, optimization, and interpretability enhances the model performance for the educational data.

General Terms
Data science, Educational Data Analysis, Explainable AI, Machine learning, Optimization, Model Interpretability.

Keywords
Classification, Cross-Validation, Explainable AI, Machine Learning, SHAP, Student Performance Prediction.

1. INTRODUCTION
In recent years, the use of Machine Learning (ML) has expanded significantly for educational data modeling. The most common applications range from basic to more domain-specific modeling. It includes (1) predicting student performance, (2) evaluating the impact of teaching practices and student-related factors on the learning process. These models help identify key contributors to student outcomes. It enables predictions of potential gains in specific subject areas. In addition, factors related to the learning environment are analyzed to understand their influence on academic performance. It includes teacher-student interactions, study time, teaching resources, and study aids [1]. The learning is not limited to a single classroom. With the advancement in educational technology, it has spread across multiple environments. It includes offline, online, and blended classrooms. In online and blended environments, datasets often include traditional offline attributes. These attributes are combined with log files, discussion forum activity, and other interaction data to provide richer analysis. The ML techniques offer the opportunity to develop and evaluate models using these educational datasets. It enables data modeling to support more personalized learning experiences and actionable insights. The predictions generated by these models are widely applied in practical educational tools. It includes recommendation systems, early warning systems, and interventions designed to improve student outcomes [2].

Recent studies have shown that the most widely explored application of ML in education is predicting students' academic performance. However, it has also contributed to improvements in student retention, overall quality of education, and learner experience and satisfaction [3]. In this context, data-driven predictive models are used to forecast students' marks and grades based on their historical academic data. It helps educators identify trends and make informed decisions [4]. The study has shown that the most commonly used algorithms for assessing student performance include Artificial Neural Networks, Naïve Bayes, Decision Trees, and Support Vector Machines [5].

In addition, data mining and ML techniques are widely used to model students' learning behavior. Educational data of students has been analyzed to build models that can be applied across various educational contexts. The patterns discovered from these models are then used to predict future performance, identify at-risk students, detect potential dropouts, and suggest targeted support strategies. Student-related data is typically categorized into demographic information, pre-academic records, virtual learning environment activity, and assessment data. The predictive modeling process involves preprocessing the data and selecting the most relevant features. It generated models that can provide actionable insights for educators and decision-makers [6]. However, ML techniques not only modeled the data but also measured the impact of specific factors on students' performance. They are applied across diverse learning environments to identify at-risk students and enable early interventions. The study findings support teachers in designing effective activities. Additionally, it provides guidance tailored to students' needs, ultimately enhancing learning outcomes and overall academic performance [7], [8].

Predictive modeling is further strengthened by analyzing student data at different stages of a course to identify at-risk students. Early predictions enable educators to implement timely interventions and support measures [9], [10]. Another systematic study reported that the primary applications of machine learning in education are predicting student performance and identifying potential dropouts. Supervised, unsupervised, and semi-supervised learning methods are commonly employed to model educational data [11]. Another enhanced modeling is achieved through a machine learning–based framework that develops course-specific models. This framework provides the necessary infrastructure for data collection. It also enables context-based prediction of student

risk levels and supports timely, effective feedback from relevant authorities [12].

ML-based predictive modeling provides valuable insights from student data. The models assist the early identification of at-risk students from their predictive performance. It helps the educational sector by providing a basis for timely interventions. By combining this data-driven modeling with interpretability and validation, this study provides more reliable, interpretable personalized educational strategies.

1.1 Objectives

To achieve the goals outlined in this study, the following research objectives have been defined:

1. Early prediction of learning outcomes using models developed with different machine learning techniques to capture diverse patterns in student data.
2. Evaluation of model performance through a robust validation strategy to ensure reliability, accuracy, and generalizability.
3. Prediction with a dynamically optimized feature set by integrating machine learning models with optimization algorithms to retain only the most relevant attributes.
4. Interpretation of model results using SHAP to understand the contribution and importance of each attribute on student performance.

1.2 Organization of the paper

This paper is structured into five sections. Section 2 reviews recent studies related to educational predictive modeling. Section 3 explains the materials and methods used, including a brief overview of the dataset and the main stages of the research methodology. Section 4 presents the results of all prediction models, accompanied by their interpretations using explainable AI techniques to identify key contributing attributes, improve model transparency, and informed decision-making. Finally, Section 5 concludes the study and highlights directions for future research.

2. RELATED WORK

The analysis of related work highlights that much of the research has focused on selecting significant attributes for model construction, thereby enhancing the performance of different classifiers. The reviewed studies show that predictive models are often used to evaluate the influence of individual attributes and attribute groups on student performance. To further improve the predictive performance of educational models, optimization techniques have been applied to identify the most relevant features. It led to more accurate predictions of student marks and grades [4]. However, the study's predictive accuracy is limited because it relies on basic training methods without proper tuning.

Moreover, the researchers aimed to reduce both the sample size and the number of attributes. The findings indicated that variations in sample sizes and feature sets had different effects on model accuracy. The predictive models trained on smaller samples achieved performance comparable to the baseline models. They exhibited differences in risk ranking and ordering [8]. This study's findings are constrained by its focus on a single community college system and specific graduation outcome, which may limit the generalizability of results to other institutions, populations, or alternative success measures.

In another educational data study, various machine learning models were evaluated for predicting student dropout risk. The dropout rate served as the primary performance metric and supported timely teacher interventions. By applying predictive models using selected attributes, the study reported notable improvements in overall model performance [10].

Additionally, one such study using students' data from offline learning environments focused on predicting final grades. It examined the key input variables that most strongly influence the predictive performance of both classification and regression models. The study further demonstrated the strong dependence of the target variable on prior academic performance and other relevant factors [13].

In the student performance prediction methodology, performance is estimated using multiple classifiers applied to datasets containing students' enrolment and activity information. To capture student heterogeneity, several sub-datasets are generated based on different groups and variables. The study revealed relationships between key factors influencing performance. It further reported that no single model performed equally well across all contexts; however, some models proved highly effective due to their strong interpretability. Moreover, the findings highlighted variations in learning motivation associated with specific features, which helped instructors design targeted student support strategies and more effective learning environments [14].

The Random Forest (RF) has been applied alongwith Support Vector Machines (SVM) for predicting student performance. The RF classifier incorporates a built-in feature selection mechanism. Unlike SVM, which relies on finding an optimal hyperplane to separate classes, RF builds an ensemble of decision trees to capture complex relationships in the data. The results demonstrated the RF's superior performance compared to SVM, particularly in terms of accuracy [15].

Moreover, classification models have been developed to evaluate student academic performance by incorporating insights from previous studies and the current context. A factor analysis was conducted using the chi-square test, which helps identify statistically significant relationships between categorical variables. Then, it selects the most relevant features for the predictive models. Among all the classifiers, the Support Vector Machine (SVM) demonstrated superior performance across key evaluation metrics [16].

Another machine learning framework was used to develop a performance predictive performance model based on students' prior academic records. The study identified the most significant variables influencing student outcomes. Then, various machine learning algorithms: Naïve Bayes, ID3, C4.5, and SVM were applied to the input dataset. This information enables teachers to focus on students who need the most support. It also provides institutions with insights to improve teaching quality [17].

A recent study introduced a tool designed to help learners choose their field of study by evaluating their abilities across various streams and specific subjects. Machine learning algorithms predicted learning aptitude. Random Forest was found to achieve the highest prediction accuracy. It predicted whether a learner's ability leaned toward scientific or literary streams. However, it is computationally expensive and memory-intensive with large datasets. The Support Vector Classifier performed best for predicting specific streams such as Literature, MPS, or Biology [18]. There is another research focused on using machine learning within specific educational domains. It analyzes students' learning styles and predicts their

success rates. In addition, the trained models offered meaningful insights into learning patterns to support educators. The findings indicated that neural networks and decision tree models exhibited superior predictive performance in their respective domains [19].

3. MATERIALS AND METHOD

In this paper, an integrated approach to predictive modeling is adopted. It includes data preprocessing, model validation, optimization, development, and interpretation. The overall process flow is illustrated in Figure 1. It shows the interconnections among the different stages of the modeling pipeline. First, data preprocessing is performed to separate and encode the target variable. Also, input data is prepared with transformations for classification modeling. Additionally, K-fold cross-validation is used to assess the predictive model's reliability. In the machine learning pipeline, an optimization algorithm selects only relevant attributes to enhance predictive performance. Finally, the model interpretation using explainable AI techniques provides important values for each selected attribute.

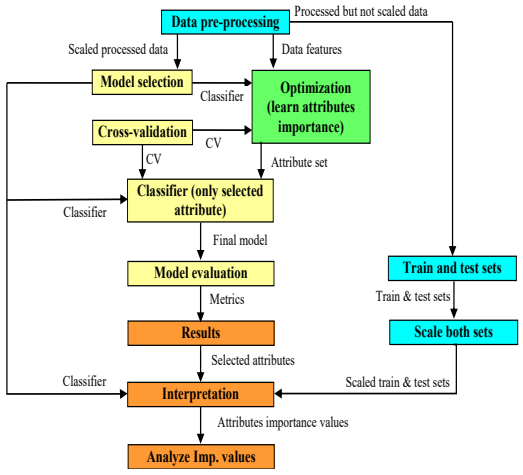

Fig 1: Steps followed in the methodology

3.1 Dataset

The real world dataset named Student performance[1] Cortez (2014) from UCI Machine Learning Repository[2] is used for model development. It consisted of the personal, historic, demographic and current academic performance details of the students in two subjects. It has 395 instances and 33 attributes including the target for Mathematics. A brief description of the data attributes is presented in Tables 1.

Table 1. Student Performance datasets (Mathematics)

Type	Attribute name
Binary	School, Sex, Address, Fam_size, Pstatus, Schoolsup, Famsup, Paid, Actvities, Nursery, Higher, Internet, Romantic
Numeric	Age, Medu, Fedu, Trave_time, Study_time, Failures, Famrel, Freetime, Gooout, Dalc, Walc, Health, Absences, G1, G2, G3
Nominal	Mjob, Fjob, reason, guardian

The target attribute, *G3*, is numerical in nature and is therefore suitable for regression models. For classification models, it is transformed into a binary categorical variable. The encoding process is illustrated in Figure 2.

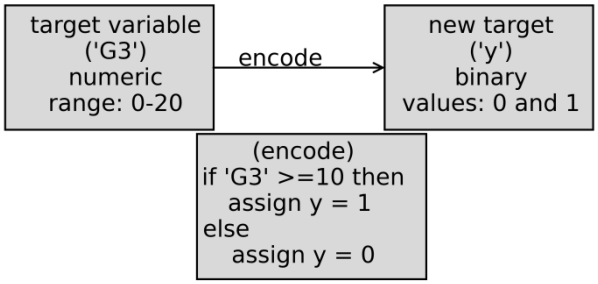

Fig 2: Target Encoding

3.2 Data transformations

The dataset contains categorical attributes, whereas machine learning models require numerical input. Therefore, all categorical attributes are encoded into numerical values. In addition, the data is standardized using z-score normalization with a standard scaler. The training and test datasets are standardized separately for model interpretation to prevent any form of data leakage.

3.3 Classification models

Several classifiers using multiple machine learning algorithms, including Logistic Regression (LR), Support Vector Classifier (SVC), Stochastic Gradient Descent Classifier (SGDC), K-Nearest Neighbors Classifier (KNC), Gaussian Naïve Bayes (GNB), and Decision Tree Classifier (DTC), are developed. Among all of them, three classifiers, LR, SVC, and SGDC model linear data. LR estimates the probability that a student will pass. SVC is a margin-based method that finds hyperplanes defined by support vectors to separate the pass and fail categories. SGDC optimizes a differentiable loss function. For non-linear modeling, k-NN and GNB are used. k-NN, a distance-based classifier, is used for similarity-based classification and does not require explicit training. GNB, another probabilistic classifier, based on maximum likelihood estimation, assumes a Gaussian distribution for the data. Finally, DTC, a tree-based model, is used to construct decision rules with its greedy learning strategy. All these models are chosen to represent different learning strategies. This enables a broad comparison of classification performance on the processed dataset. The model evaluates each selected attribute subset generated by the optimization algorithm. The same configured classifiers are used for the final model evaluation.

3.4 Model assessment

The classification models are evaluated and compared using standard performance metrics derived from the actual and predicted target values. Four performance metrics are used: accuracy, precision, recall, and F1 score, which are computed according to Equations (1)–(5).

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

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

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

$$F1\ score = \frac{2(Precision \times Recall)}{Precision + Recall} \quad (4)$$

[1] https://archive.ics.uci.edu/dataset/320/student+performance

[2] https://archive.ics.uci.edu/

$$F1_{weighted} = \sum_{i=1}^{C} \frac{n_i}{N} \times F1_i \quad (5)$$

where TP denotes true positives, TN denotes true negatives, FP denotes false positives, and FN denotes false negatives. In Equation (5), C is the total number of classes, n_i is the number of samples in class i, N is the total number of samples, and $F1_i$ is the F1-score of class i.

Here, accuracy indicates the general level of correct predictions; however, it can give an inflated view of performance when educational data are imbalanced. Precision measures how many students who were predicted to be at risk actually are. Recall measures the proportion of truly at-risk students correctly identified by the classifier. The weighted F1-score provides a balanced evaluation of the classifier in the binary student performance prediction task. The mean value of each metric across all the folds is calculated to produce the final assessment results. In addition, the size of the selected attribute subset is reported to demonstrate effective feature reduction while maintaining or even improving predictive performance.

3.5 Optimization Algorithm

The study employs Genetic Algorithms (GAs) [20], [21], a metaheuristic approach, to search for optimized attribute subsets. GAs are population-based algorithms that aim to identify a global optimum subset of attributes. Each candidate is represented as a binary chromosome, where 1 indicates that an attribute is selected and 0 indicates that it is excluded. The length of the chromosome is the total number of attributes. Its fitness is evaluated based on the predictive performance of the respective classifier trained only on the contributing attributes. The objective is to maximize the F1-score obtained through K-fold cross-validation. Then, genetic operators are applied to evolve the population across generations. The best-performing candidate in the final population is selected as the optimal attribute subset for the final model.

Genetic Algorithm (GA) is applied using the Distributed Evolutionary Algorithms in Python (DEAP) framework, with a population size of 50 and 50 generations. A two-point crossover is used as the crossover operator with a crossover probability of 0.8. At the same time, bit-flip mutation is applied with a mutation probability of 0.05 and an individual-bit mutation probability of 0.05. For parent selection, a tournament size of 2 is used. A Hall of Fame of size 5 is maintained to preserve the elitist solutions. At the end of optimization, the GA produced the best-performing subset of attributes, which is then used in the subsequent evaluation stage.

3.6 Explainable AI

Explainable AI is employed to understand the decision-making process of predictive models. SHAP (SHapley Additive exPlanations) is used to interpret the co-ntribution of individual attributes to the model output. It assigns an importance value to each attribute for every prediction. Since different machine learning models require different SHAP explainers, the explainer is chosen according to the nature and configuration of each classifier. LinearExplainer is used for Logistic Regression (LR), Support Vector Classifier with linear kernel (SVC-linear), and Stochastic Gradient Descent Classifier (SGDC). At the same time, KernelExplainer for the K-Nearest Neighbors Classifier (KNC) and the Gaussian Naive Bayes (GNB) classifier. TreeExplainer is used for the Decision Tree Classifier (DTC) because it is specifically designed for tree-based models. SHAP bar plots are generated from the SHAP values of these explainer. It helps compare the selected attributes by their contribution across classifiers. As a result, the most influential attributes are identified. Thus, SHAP reduces the black-box nature of machine learning models and enables informed and accountable decision-making.

4. RESULTS AND DISCUSSION

The assessment is carried out for both the full attribute set and the GA-selected attribute subset in order to compare the effect of attribute selection on classification performance. These comparisons are summarized in tables and graphs. It provides a clear visualization of changes in model performance, accuracy, and complexity due to attribute reduction. In addition to performance metrics, model interpretation should also be considered to understand how the selected attributes already influence each classifier's decision-making process.

4.1 Model evaluation

The performance metrics of each classifier are presented in Table 2. SVC and GNB selected 18 attributes each, reducing the feature set by 14 (approximately 44%). It indicates a moderate reduction and models dependency on the full attribute set. SGDC selected 16 attributes, reducing the feature set by 16 attributes (50%). It is shown to have a balanced trade-off between model simplicity and predictive capability. DTC selected 15 attributes, reducing 17 features (53%) and thereby reducing unnecessary complexity. LR used only 11 attributes, a reduction of 21 features (66%). It shows that smaller subsets are effective for classification and for interpretation. Finally, kNC selected only 7 attributes, reducing 25 features (78%). It indicates kNC may perform classification using only the most essential attributes. However, the most compact model may be achieved at the cost of losing some predictive information. From the results, it is clear that different classifiers have different levels of dependence on the data attribute space.

Table. 2 Model metrics

Classifier	Selected attribute	Accuracy score		F1 score (weighted)		precision		Recall	
		All	Selected	All	Selected	All	Selected	All	Selected
LR	11	0.9013	0.9367	0.9001	0.9359	0.9174	0.9425	0.9396	0.966
SVC	18	0.8861	0.9392	0.885	0.9388	0.9093	0.9456	0.9245	0.966
SGDC	16	0.881	0.9266	0.8797	0.9267	0.9063	0.9475	0.9208	0.9434
kNC	7	0.7899	0.9038	0.775	0.9018	0.803	0.9165	0.917	0.9472
GNB	18	0.8354	0.9089	0.8327	0.9083	0.8723	0.927	0.8906	0.9396
DTC	15	0.881	0.9291	0.8803	0.9292	0.9042	0.9511	0.9208	0.9434

The results in Table 2 show that better performance does not always depend on a larger attribute set. Here, SVC attains the highest accuracy (0.9392), but it uses 18 attributes. There is a performance improvement, but the model's complexity has increased as well. LR also achieves a high accuracy of 0.9367 with only 11 selected attributes. Here, SVC performance is better than LR. However, the improvement is marginal compared to the additional attributes required. The next classifier, SGDC, uses 16 attributes and achieves an accuracy of 0.9266. It suggests that selecting more attributes does not

necessarily improve predictive performance. kNC uses the smallest attribute set and achieves an accuracy of 0.9038. It shows that reasonable performance is achieved with very low attribute dependency. Despite using 18 attributes, GNB attains an accuracy of only 0.9089, which is lower than several models with fewer attributes. Here, the classifier does not benefit much from a larger attribute set and may be affected by less informative attributes. It means that additional attributes introduce noise. Finally, DTC uses 15 attributes and achieves an accuracy of 0.9291. Although it does not use the fewest attributes, it performs better without the full set. The achieved accuracy improvement indicates that these classifiers can effectively model the classification task with a compact attribute set.

The weighted F1-score values in Table 2 illustrate the same performance pattern as observed in accuracy. Classifiers that achieved higher accuracy have also got higher weighted F1-scores. LR, which previously showed strong accuracy, performs well again. It achieves a weighted F1-score of 0.9359 with only 11 attributes. This suggests that LR achieves good class-wise prediction with fewer attributes. SVC, which recorded the highest accuracy, also attains the highest weighted F1-score (0.9388). It shows a larger increase in score with fewer attributes than the full set. Similarly, DTC and SGDC exhibit moderately high weighted F1-scores (0.9292 and 0.9267) with 15 and 16 attributes, respectively. Their results align with their respective accuracy levels, with a moderate number of selected attributes. GNB also improves its weighted F1-score to 0.9083 after using 18 attributes. However, it shows that a larger attribute set does not always provide better performance. kNC, which had comparatively lower accuracy, shows a correspondingly lower F1 score (0.9018). Although with a minimum attribute set, it achieves a larger improvement. This shows that it works with a very small feature subset, though with lower prediction quality. Overall, the alignment between the accuracy and F1-score patterns confirms that classifiers that perform well in terms of accuracy also maintain reliable class-wise prediction performance. Also, the selected attributes are useful, but each classifier uses them differently.

The precision–recall (Table 2) values also align closely with the results for accuracy and F1 score, which is an improvement in score with fewer attributes. SVC and LR, which achieved higher accuracy and weighted F1-score earlier, also demonstrate excellent balance with very high recall (0.966) and strong precision. DTC gives the highest precision (0.9511) and maintains a solid recall. It means its positive predictions are more reliable, and it produces fewer false positives. SGDC shows a good balance between precision (0.9475) and recall (0.9434), with greater improvement than from the full set with fewer attributes. KNC has the lowest precision (0.9165) among the classifiers with the smallest attribute set, but a relatively better recall (0.9472) than GNB. It suggests that it identifies more positive cases, but also produces more false-positive predictions. It shows its alignment with its lower accuracy and weighted F1-score pattern. GNB shows moderate values for both precision (0.927) and recall (0.9396), which is consistent with its overall mid-level performance. In general, the classifiers with better accuracy and weighted F1-score also show a better balance between precision and recall.

4.2 Performance comparison

Figure 4 compares the accuracy of each classifier before and after attribute selection. In all cases, the selected attribute set gives better accuracy than the full set. This shows that the removed attributes did not help the models and likely introduced noise or redundancy.

The improvement is clear for kNC, where accuracy rises from 0.7899 to 0.9038, the largest gain among all classifiers. It shows that kNC is strongly affected by irrelevant attributes and performs much better with a compact set of attributes. SVC also shows a major increase, from 0.8861 to 0.9392. Here, attribute selection helps it form a clearer class boundary. LR improves from 0.9013 to 0.9367, indicating that the reduced attribute set is sufficient for strong prediction and supports a simpler model. GNB shows an increase from 0.8354 to 0.9089, which means it also benefits from removing less useful attributes. Finally, SGDC improves from 0.881 to 0.9266, and DTC rises from 0.881 to 0.9291. It shows that both classifiers gain stability and better predictive ability after selection.

Overall, the comparison shows that attribute selection does more than reduce dimensionality. It consistently improves model quality and enhances classifier accuracy. It implies that the selected attributes preserve useful information, while the removed ones degrade classification performance.

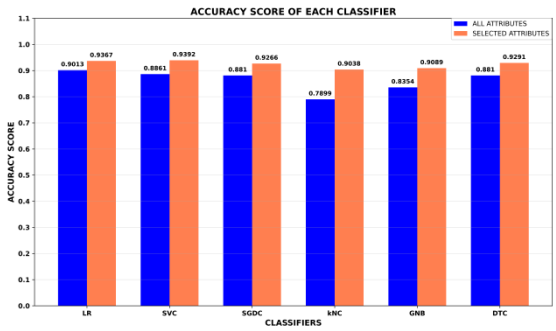

Fig 3: Accuracy score for all classifiers

Figure 5 shows the weighted F1-score of each classifier with all attributes and with the selected attributes. The improvement is visible across all classifiers. It means attribute selection improves the balance between precision and recall, not just the overall prediction rate.

The largest improvement appears in kNC. Its weighted F1-score increases from 0.7750 to 0.9018. It implies that this classifier is highly affected by irrelevant attributes and becomes much more reliable after attribute reduction. Similarly, SVC also shows a strong increase, from 0.8850 to 0.9388. It indicates better class-wise prediction after selecting only useful attributes. Also, the next-highest rise in the weighted F1-score is for GNB, from 0.8327 to 0.9083. A similar observation is made for SGDC and DTC. Here, the computed score improves from 0.8797 to 0.9267 for SGDC, and from 0.8803 to 0.9292 for DTC. In both classifiers, the selected attributes help produce more consistent classification results. Finally, LR improves from 0.9001 to 0.9359, indicating alignment with the accuracy results. It indicates that a smaller attribute set is sufficient to maintain balanced performance for all classifiers. Overall, the selected attributes strengthen all classifiers' performance.

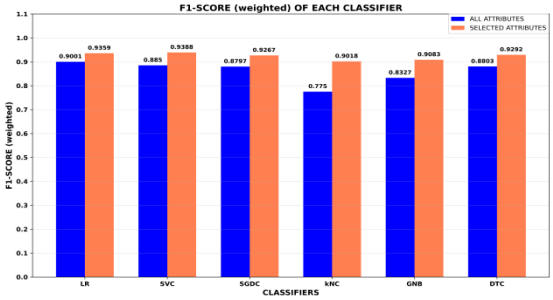

Fig 4: F1-score (weighted) for all classifiers

Figure 6 extends the analysis with an illustration of the precision increases for every classifier after attribute selection. It also shows the precision of each classifier with all attributes and with the selected attributes. It reports that the reduced attribute set helps classifiers make fewer false-positive predictions.

It is clear that kNC again achieves the maximum improvement in the precision. It goes from 0.8030 to 0.9165. The findings support the claim that attribute reduction improves its reliability. The next-highest increase is observed in GNB. Its precision changes from 0.8723 to 0.9270. Again, it indicates better predictive quality after removing weak attributes. Further, DTC shows improvement, from 0.9042 to 0.9511. The other two classifiers, SGDC and SVC, show approximately the same improvement in precision. It is from 0.9063 to 0.9475 for SGDC and from 0.9093 to 0.9456 for SVC. Finally, LR precision also rises from 0.9174 to 0.9425. These results show that the selected attributes help all classifiers produce more accurate positive predictions.

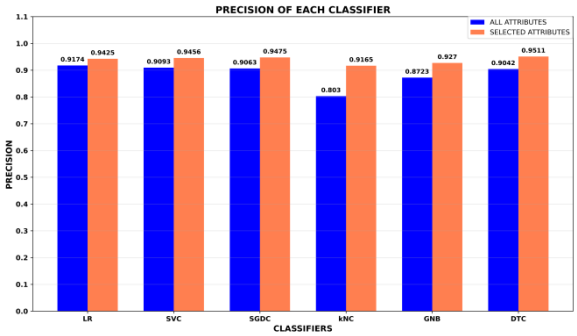

Fig 5: Precision for all classifiers

Figure 7 shows the recall values of each classifier before and after attribute selection. Recall improves for all classifiers when the selected attributes are used. This means that the reduced attribute set helps the classifiers detect more true positives.

Here, LR and SVC reach the highest recall after selection, both at 0.9660. This shows the respective classifiers have very strong detection ability with the reduced attribute set. The next-best-performing classifier is kNC. Its recall rises from 0.9170 to 0.9472. This indicates better model sensitivity after removing less useful attributes. The two classifiers, SGDC and DTC, also improve, achieving a recall of 0.9434. Finally, GNB recall improves from 0.8906 to 0.9396. It shows that identifying positive cases is now more reliable than before. The results confirm that attribute selection enhances recall across all classifiers.

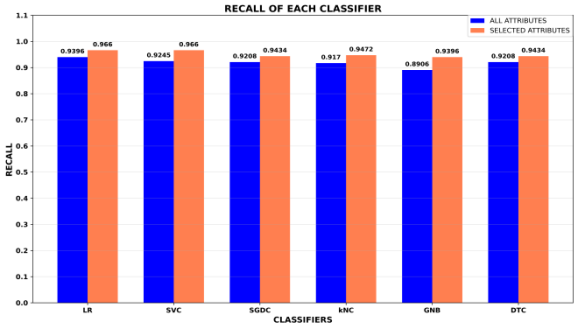

Fig 6: Recall for all classifiers

4.3 Model Interpretation

The SHAP summary plots for each selected attribute and its respective classifier are shown in Figures 8-13. It highlights the contribution of individual attributes to model predictions. A clear pattern appears across all plots. From the summary plots, G2 is found to be the most important attribute across all classifiers. This means the second grade has the strongest effect on the final prediction in all classifiers. The next-most important attribute is the Age. It is repeated across all six summary plots, but its rank varies across classifiers. This shows that Age remains relevant, but its effect is less stable than G2. Other important attributes are: Romantic, Famrel, and Guardian. All these attributes got selected in multiple classifiers. However, the SHAP importance values for these attributes vary across classifiers. It indicates that their contributions are moderate yet meaningful across classifiers. However, there are other attributes with higher SHAP importance values than these commonly occurring attributes. However, their occurrence frequency is limited to fewer classifiers. Finally, another grade-related attribute, G1, is also highly important in many classifiers, especially in GNB, kNC, and SGDC. These importance-value plots show that students' prior academic performance dominates the prediction task.

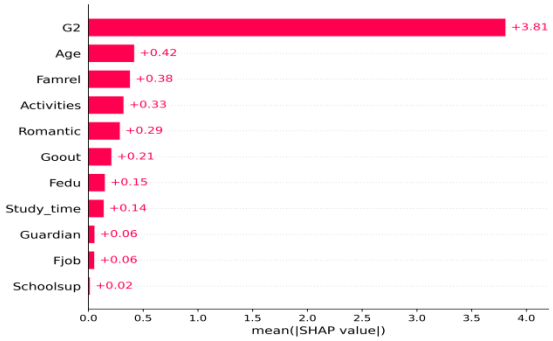

Fig 7: SHAP values for LR

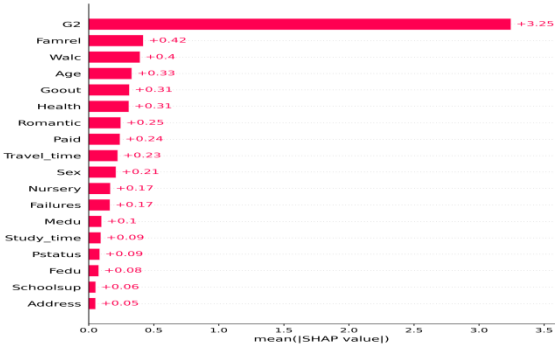

Fig 8: SHAP values for SVC

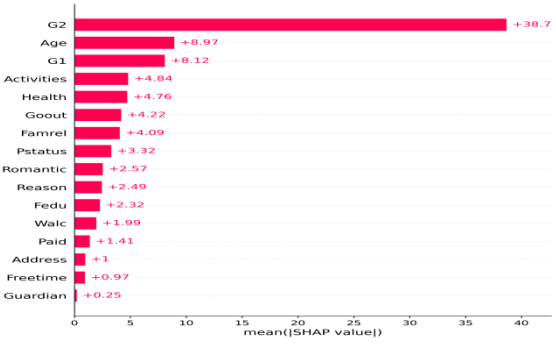

Fig 9: SHAP values for SGDC

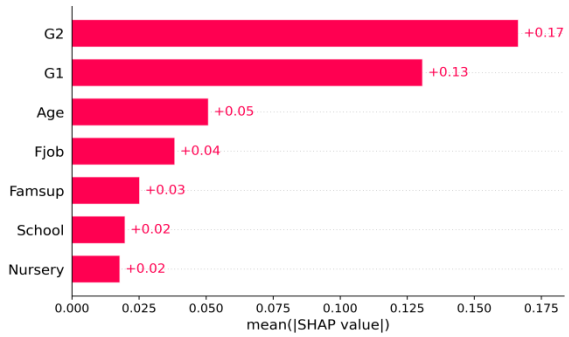

Fig 10: SHAP values for kNC

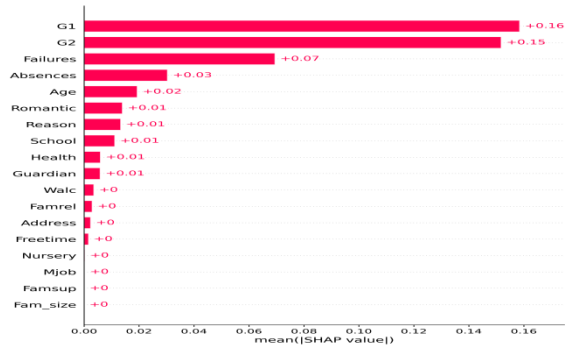

Fig 11: SHAP values for GNB

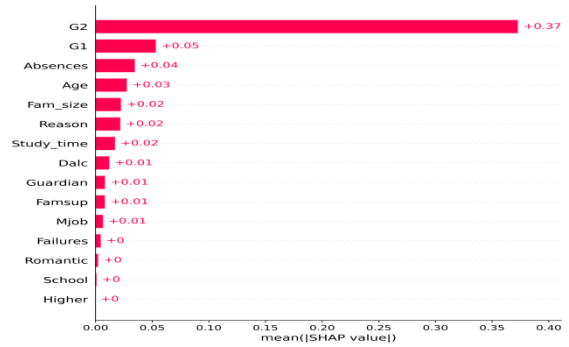

Fig 12: SHAP values for DTC

The role of the selected attributes changes by classifier. In Figures 8-13, the attribute positions indicate their relative influence on the classifier. In LR, G2 not only have highest importance value but its value is far ahead of all other attributes. It suggests that the classifier has one dominant predictor and that the others are supporting variables. In GNB, both previous grades, G1 and G2, are found to have high importance values. Also, Failures shows a clear next-high contribution for this classifier. So, this classifier uses multiple academic factors together. In kNC, the smallest attribute subset is selected, and again, G2 and G1 lead the plot. In the remaining classifiers: DTC, SGDC, and SVC, the influence is spread across more attributes. It includes: Famrel, Romantic, Guardian, Walc, Health, and Activities. This means these classifiers use not only academic history but also personal and family-related factors. Overall, the importance value plots show that some attributes, mainly G2, G1, and Age, drive the predictions across classifiers, while the other selected attributes provide model-specific support.

5. CONCLUSION

The paper presented an integrated educational data modeling approach with improved predictive performance. The approach combined data modeling with model validation and evaluation, optimization with a genetic algorithm, and model interpretation with SHAP. The results confirmed that the genetic algorithm provided a better attribute subset. The selected attributes consistently improve all the classifiers across all evaluated metrics. Moreover, the selected attribute subsets generalize better than the full set. It suggested that eliminating irrelevant attributes leads to more stable, reliable predictions. Additionally, attribute reduction using optimization techniques minimized the computational overhead of the full set. The interpretability analysis provided clear insights into each attribute's contribution to decision-making. It also focused on only a meaningful set of attributes selected with the optimization algorithm. It made the SHAP-based explanations more interpretable for real-world model deployment. Overall, the integrated approach provided classification models that are not only accurate and efficient but also interpretable and robust.

6. ACKNOWLEDGMENTS

The authors acknowledge the use of an AI-assisted tool to support language editing and manuscript preparation.

7. REFERENCES

[1] J.-M. Trujillo-Torres, H. Hossein-Mohand, M. Gómez-García, H. Hossein-Mohand, and F.-J. Hinojo-Lucena, “Estimating the academic performance of secondary education mathematics students: A gain lift predictive model,” Mathematics, vol. 8, no. 12, p. 2101, 2020.

[2] Y. Zhang, Y. Yun, R. An, J. Cui, H. Dai, and X. Shang, “Educational data mining techniques for student performance prediction: method review and comparison analysis,” Frontiers in psychology, vol. 12, p. 698490, 2021.

[3] A. S. Pinto, A. Abreu, E. Costa, and J. Paiva, “How machine learning (ml) is transforming higher education: A systematic literature review,” Journal of Information Systems Engineering and Management, vol. 8, no. 2, 2023.

[4] S. Hussain and M. Q. Khan, “Student-performulator: Predicting students academic performance at secondary and intermediate level using machine learning,” Annals of data science, vol. 10, no. 3, pp. 637–655, 2023.

[5] N. R. Yadav and S. S. Deshmukh, “Prediction of student performance using machine learning techniques: A review,” in International Conference on Applications of Machine Intelligence and Data Analytics (ICAMIDA 2022). Atlantis Press, 2023, pp. 735–741.

[6] R. Umer, T. Susnjak, A. Mathrani, and L. Suriadi, “Current stance on predictive analytics in higher education: opportunities, challenges and future directions,” Interactive Learning Environments, pp. 1–26, 2021.

[7] C. Herodotou, B. Rienties, A. Boroowa, Z. Zdrahal, and M. Hlosta, “A large-scale implementation of predictive learning analytics in higher education: the teachers role and perspective,” Educational Technology Research and Development, vol. 67, no. 5, pp. 1273–1306, 2019.

[8] K. A. Bird, B. L. Castleman, Z. Mabel, and Y. Song, “Bringing transparency to predictive analytics: A systematic comparison of predictive modeling methods in

higher education," AERA Open, vol. 7, p. 23328584211037630, 2021.

[9] M. Adnan, A. Habib, J. Ashraf, S. Mussadiq, A. A. Raza, M. Abid, M. Bashir, and S. U. Khan, "Predicting at-risk students at different percentages of course length for early intervention using machine learning models," Ieee Access, vol. 9, pp. 7519–7539, 2021.

[10] J. Kabathova and M. Drlik, "Towards predicting students dropout in university courses using different machine learning techniques," Applied Sciences, vol. 11, no. 7, p. 3130, 2021.

[11] K. Alalawi, R. Athauda, and R. Chiong, "Contextualizing the current state of research on the use of machine learning for student performance prediction: A systematic literature review," Engineering Reports, vol. 5, no. 12, p. e12699, 2023.

[12] K. Alalawi, R. Athauda, and R. Chiong, "An extended learning analytics framework integrating machine learning and pedagogical approaches for student performance prediction and intervention," International Journal of Artificial Intelligence in Education, pp. 1–49, 2024.

[13] P. Cortez, "Student Performance," UCI Machine Learning Repository, 2014, DOI: https://doi.org/10.24432/C5TG7T.

[14] S. Helal, J. Li, L. Liu, E. Ebrahimie, S. Dawson, D. J. Murray, and Q. Long, "Predicting academic performance by considering student heterogeneity," Knowledge-Based Systems, vol. 161, pp. 134–146, 2018.

[15] S. Rai, K. A. Shastry, S. Pratap, S. Kishore, P. Mishra, and H. Sanjay, "Machine learning approach for student academic performance prediction," in Evolution in Computational Intelligence: Frontiers in Intelligent Computing: Theory and Applications (FICTA 2020), Volume 1. Springer, 2021, pp. 611–618.

[16] D.Wang, D. Lian, Y. Xing, S. Dong, X. Sun, and J. Yu, "Analysis and prediction of influencing factors of college student achievement based on machine learning," Frontiers in Psychology, vol. 13, p. 881859, 2022.

[17] H. Pallathadka, A. Wenda, E. Ramirez-Asís, M. Asís-López, J. Flores-Albornoz, and K. Phasinam, "Classification and prediction of student performance data using various machine learning algorithms," Materials today: proceedings, vol. 80, pp. 3782–3785, 2023.

[18] P. Houngue, M. Hountondji, and T. Dagba, "An effective decision-making support for student academic path selection using machine learning," International Journal of Advanced Computer Science and Applications, vol. 13, no. 11, 2022.

[19] S. Fadili, M. Ertel, A. Mengad, and S. Amali, "Predicting optimal learning approaches for nursing students in morocco." International Journal of Advanced Computer Science & Applications, vol. 15, no. 4, 2024.

[20] J. H. Holland, "Genetic algorithms," Scientific american, vol. 267, no. 1, pp. 66–73, 1992.

[21] D. Beasley, D. R. Bull, and R. R. Martin, "An overview of genetic algorithms: Part 2, research topics," University computing, vol. 15, no. 4, pp. 170–181, 1993.