Modern Education: Advanced Prediction Techniques for Student Achievement Data

Xi LU
Hubei Institute of Fine Arts, Wuhan 430060, Hubei, China

***Abstract*—Enhancing educational outcomes across varied institutions like universities, schools, and training centers necessitates accurately predicting student performance. These systems aggregates the data from multiple sources—exam centers, virtual courses, registration departments, and e-learning platforms. Analyzing this complex and diverse educational data is a challenge, thus necessitating the application of machine learning techniques. Utilizing machine learning algorithms for dimensionality reduction simplifies intricate datasets, enabling more comprehensive analysis. Through machine learning, educational data is refined, uncovering valuable patterns and forecasts by simplifying complexities via feature selection and dimensionality reduction methods. This refinement significantly amplifies the efficacy of student performance prediction systems, empowering educators and institutions with data-driven insights and thereby enriching the overall educational landscape. In this particular research, the Decision Tree Classification (DTC) model is used for forecasting student performance. DTC stands out as a potent machine-learning method for classification purposes. Two optimization algorithms, namely the Fox Optimization (FO) and the Black Widow Optimization (BWO), are integrated to heighten the model's accuracy and efficiency further. The amalgamation of DTC with these pioneering optimization techniques underscores the study's dedication to harnessing the forefront of machine learning and bio-inspired algorithms, ensuring more precise and resilient predictions of student performance, ultimately culminating in improved educational outcomes. From the results garnered for G1 and G3, it is evident that the DTBW model demonstrated the most exceptional performance in both predicting and categorizing G1, achieving an Accuracy and Precision value of 93.7 percent. Conversely, the DTFO model emerged as the most precise predictor for G3, achieving an Accuracy and Precision of 93.4 and 93.5 percent, respectively, in the prediction task.**

Keywords—Student performance; classification; decision tree classification; fox optimization; black widow optimization

I. INTRODUCTION

The expansion of educational data sourced from admission systems, academic information systems, and e-learning platforms is substantial. Nonetheless, a significant portion of this data remains untapped due to its intricate nature and sheer volume. The analysis of this data holds pivotal importance in forecasting student performance. Data mining, known as knowledge discovery in databases (KDD), has proven to be efficacious across diverse domains, including education, paving the way for the emergence of Educational Data Mining (EDM) [1, 2].

Forecasting student outcomes in education significantly relies on EDM, allowing the anticipation of various results like passing, failing, and grading. A core focus involves establishing an early alert system to reduce costs, save time, and optimize available resources. Enhanced educational techniques are vital in refining student performance, enabling educators to tailor teaching methods and provide extra support where needed. These predictions empower students to gauge their potential academic progress and take necessary actions. Long-term institutional goals are centered on fortifying student retention, ultimately enhancing the institution's standing, rankings, and the career prospects of its graduates [3]–[6].

Educational establishments utilize data mining, commonly referred to as EDM, to thoroughly analyze the available data. Machine learning algorithms serve as pivotal tools for uncovering essential knowledge. Accurate performance prediction is instrumental in early identification of struggling students [7, 8]. EDM supports institutions in refining and developing novel learning methods by examining educational data. However, predicting academic performance presents challenges due to the diverse factors influencing it [9, 10]. Technological progress has facilitated the development of effective machine-learning methods [11–16]. Recent scholarly research emphasizes the efficacy of machine learning techniques in advancing the field of education.

Predicting student performance through machine learning (ML) is crucial for enhancing education in several ways. It enables early identification of academic struggles, allowing for timely interventions and personalized learning plans. By optimizing resource allocation and addressing factors influencing dropout rates, institutions can improve retention and graduation rates. Machine learning facilitates data-driven decision-making, adaptive assessments, and efficient educational planning. Continuous monitoring supports quality assurance, accountability, and a competitive advantage for institutions. Overall, it empowers educators to provide targeted support, leading to improved student outcomes and a more responsive education system.

II. RELATED WORK

Ajay et al. [17] investigated the influence of the "CAT" social factor in predicting student performance among Indians. They employed four classifiers and found that the IB1 model exhibited the highest accuracy at 82%. This factor categorized individuals based on social status, directly impacting educational outcomes. Dorina et al. [18] developed a predictive model for student success using various classification algorithms. While the MLP model achieved the highest accuracy for identifying successful students, it encountered challenges in handling high-dimensional data and class imbalances. Carlos used machine learning to create a student failure prediction model, achieving a high accuracy of 92.7% with the ICRM classifier. However, due to varying student characteristics, their study did not encompass testing across different educational levels. Edin Osmanbegovic et al. [19] devised a model to predict student academic success while tackling data dimensionality issues. Despite Naïve Bayes achieving the highest accuracy at 76.65%, the model did not effectively address the class imbalance problem.

A study [20] utilized various data mining methods to predict course dropouts in the context of EDM challenges. The support vector machine, with specific predictors, offered the most accurate classifications. However, including earned grades from prerequisite courses posed a limitation due to potential improvements in student knowledge during the course. Another study [21] aimed to enhance the ID3 model for predicting student academic performance, overcoming its inefficiencies in selecting attributes with numerous values. The proposed model significantly improved performance, achieving a high accuracy of 93% with the wID3 classifier. A study [22] introduced an early identification model for student failures, exploring multiple data mining methods and preprocessing techniques. Although the support vector machines outperformed other models, the study did not address reducing classification errors. Introducing an ensemble model, a study [23] aimed to identify underperforming students by combining classifiers. The ensemble model, incorporating standard-based grading assessments, outperformed individual classifiers, achieving an accuracy of 85%.

Suggesting a predictive system for online student learning performance, another study [24] found that methods considering time-dependent variables achieved higher accuracy. However, the model was not tested in an offline mode, potentially affecting its performance. Thammasiri et al. [25] proposed a model to predict poor academic performance among freshmen. The combination of support vector machines with SMOTE achieved the highest accuracy of 90.24%, addressing class imbalance issues. Challenging assumptions, a study [26] emphasized the applicability of data mining in small datasets for predicting student success. Although achieving over 90% accuracy with Reptree, the model did not effectively handle high data dimensionality or class balancing challenges. Addressing multiclass classification issues, a study [27] proposed a multi-level model to improve overall accuracy. This model, involving resampling and two levels of classification, achieved over 90% accuracy for both overall model and individual class predictions, using J48 as a key classifier.

III. Objective

The core aim of this research was to establish a robust machine-learning model designed for predicting Student Performance, drawing on data from credible sources. The study focused on leveraging the Decision Tree Classification (DTC) technique. An innovative approach was introduced by seamlessly integrating two optimization algorithms: Fox Optimization (FO) and Black Widow Optimization (BWO). This unique amalgamation of techniques was intended to significantly boost the Accuracy and Precision of the predictive model, thereby offering more effective forecasts of student performance within an educational setting. The DTC model is instrumental in predicting student performance in Mathematics due to its ability to comprehend and represent intricate relationships within data. Specifically tailored for educational contexts, the DTC method efficiently delineates critical factors influencing math performance. Its hierarchical structure allows for identifying significant decision paths, highlighting key determinants such as study habits, prior academic achievements, and socio-economic backgrounds. By comprehensively mapping these interdependencies, the DTC model predicts outcomes accurately and unveils pivotal insights essential for targeted interventions and tailored academic support, thereby enhancing student performance in Mathematics.

This study underscores the vital role of data-driven predictive models in education, advocating for a comprehensive approach to evaluate students' academic performance. Demonstrating the effectiveness of data mining techniques, including clustering and classification, the research innovatively integrates the DTC model with FO and BWO. This integration highlights the potential of combining machine learning and optimization algorithms to enhance precision, providing a robust toolkit for addressing challenges in students' academic journeys. The thorough evaluation process reveals the significant potential of these hybrid models to improve the DTC model's classification accuracy and precision, contributing to advancements in academic performance prediction.

IV. Materials and Methodology

A. Data Preparation

The primary aim of this study revolves around constructing a robust method to accurately evaluate students' academic performance while considering various contextual factors that influence it. To accomplish this objective, the initial dataset necessitates crucial preprocessing steps. The first essential step involves converting textual data into numerical values, a foundational requirement for conducting machine learning tasks. This conversion is pivotal as it facilitates effective data analysis and enables the application of advanced statistical techniques. The dataset encompasses a diverse range of variables that potentially impact students' academic outcomes, encompassing factors such as sex, school, urban or rural residency (address), age, family size (famsize), parental cohabitation status (Pstatus), parental education and occupations (Medu, Fedu, Mjob, and Fjob), school choice motivation (reason), weekly study time (studytime), guardian, home-to-school travel time (traveltime), current health status, past class failures (failures), participation in supplementary education (schoolsup), family educational support (famsup), engagement in extra paid classes, involvement in extracurricular activities, attendance at nursery school, aspirations for higher education, access to the internet, student absences, weekday (Dalc), and weekend (Walc) alcohol consumption, involvement in romantic relationships, quality of family relationships, free time, and frequency of socializing.

This research aims to predict and categorize students' academic performance, utilizing the G1 and G3 variables. G3 represents final grades obtained from school reports, ranging from zero (indicating the lowest grade) to 20 (representing the highest grade). These grades are segmented into four distinct levels: Poor (0–12), Acceptable (12–14), Good (14–16), and Excellent (16–20), allowing for a more nuanced evaluation of student achievement. This methodology seeks to establish a comprehensive framework for comprehending and assessing academic performance within a myriad of contextual factors, ultimately contributing to improvements in educational practices and developing policies in the academic sphere.

Fig. 1 displays a correlation matrix detailing the relationships among input and output variables within this study. It notably highlights the positive influence of parental education, particularly maternal education, on students' academic performance. Moreover, factors such as daily and weekly alcohol consumption, prior academic failures, and student age demonstrate discernible impacts on school grades. Ultimately, the matrix underscores the critical importance of both study time and parental education as pivotal factors contributing to academic success. Notably, there is a strong positive correlation (0.8264) between grades in the first period ("G1") and final grades ("G3"), indicating that students who perform well in the initial period tend to have higher final grades. Additionally, some demographic and lifestyle factors exhibit correlations. For instance, parental education levels ("Medu" and "Fedu") show moderate positive correlations, implying a potential influence on academic performance. The variable "sex" demonstrates a weak negative correlation with "age" (-0.0437), suggesting a slight tendency for younger students to be male.

The correlation matrix provides a snapshot of associations between different variables, offering insights into potential patterns and relationships. However, it is important to approach these correlations cautiously, as correlation does not imply causation and other factors may contribute to the observed relationships.

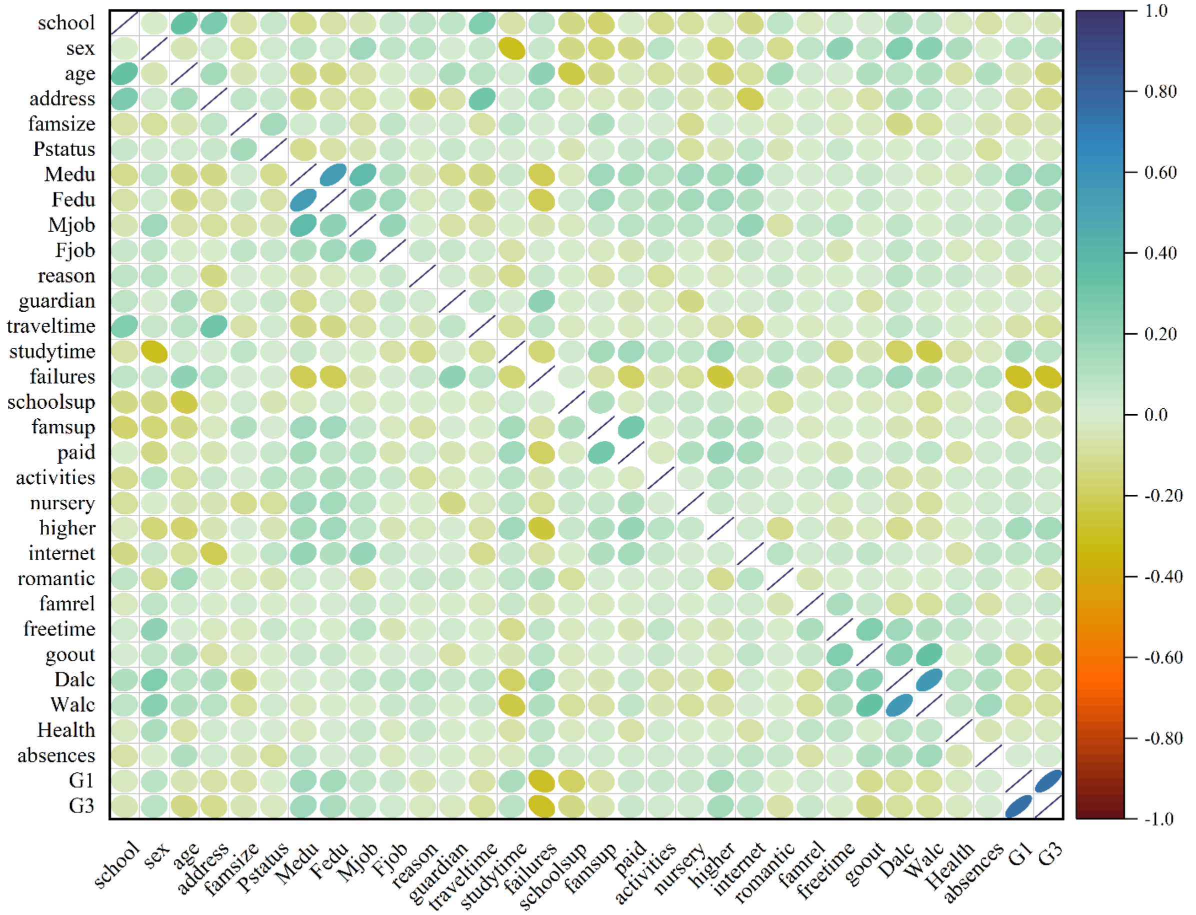


Fig. 1. Correlation matrix for the input and output variables.

B. *Evaluation of Models' Applicability*

In academic studies focused on classification problems, Accuracy is a widely employed metric used to evaluate the overall performance of a model. It relies on four fundamental components: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). TP signifies accurate predictions, TN represents correct negative predictions, FP indicates incorrect positive predictions, and FN denotes inaccurate negative predictions. However, Accuracy tends to favor the majority class, offering limited insights in situations where data is imbalanced. Three additional evaluation metrics—Recall, Precision, and F1-Score—are utilized to overcome this limitation. Recall evaluates the model's capability to correctly identify all relevant instances within a specific class, which is crucial in reducing False Negatives. Precision measures the accuracy of positive predictions, aiming to minimize False Positives, instances predicted as positive but not belonging to the class. F1-Score, combining Precision and Recall, provides a balanced assessment of model performance, particularly valuable in scenarios with imbalanced data, considering both minority and majority classes. Defined by mathematical equations, these metrics collectively provide a deeper understanding of a classification model's effectiveness. They are especially beneficial in challenging situations involving imbalanced data, where the interpretation of Accuracy might be misleading. The utilization of these metrics empowers researchers and data analysts to make more informed decisions and adjustments to enhance model performance in such intricate scenarios [28].

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

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

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

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

C. Decision Tree Classification (DTC)

A decision tree takes the form of a structure resembling a flowchart, where each internal node conducts a test based on an attribute, with each branch signaling the outcome of that particular test. Meanwhile, every leaf node, also termed a terminal node, denotes a distinct class label. Making predictions with a decision tree involves assessing the attribute values of a given data point, typically referred to as a tuple, by following a path from the root of the tree to a leaf node containing the projected class label for that specific data point. The strength of decision trees lies in their ease of conversion into classification rules. They serve as predictive models in decision tree learning, enabling the translation of observations about an object into conclusions about its intended value. These models have diverse applications in statistics, data mining, and machine learning, particularly in classification trees, which specifically handle finite class values. Compared to other classification methods, decision tree construction is commonly recognized as a swift process [29].

The decision tree relies on three key parameters:

1) D (Data Partition): D represents the initial dataset containing training examples and their respective class labels.

2) Attribute list: This parameter comprises attributes that detail the features of the data.

3) Attribute selection method: This parameter defines the strategy used to select the most suitable attribute for creating divisions or branches in the decision tree. Common methods involve measures like information gain or the Gini index.

Here is an overview of how the algorithm operates:

- It initiates by establishing a node labeled "A."
- If all the examples in the present dataset share the same class, "A" becomes a leaf node designated with that common class label.
- When the attribute list is empty, node "A" transforms into a leaf node, now tagged with the class that most frequently appears among the data samples.
- The algorithm then selects the attribute to split the data in a way that generates the purest subsets.
- Node "A" is assigned this selected attribute as the decision criterion.
- If the chosen attribute is discrete, it is removed from the attribute list.
- The data is segregated into subsets based on the outcomes of the selected attribute.
- If any of these subsets are empty, a leaf node is linked to node "A," labeled with the majority class of the original dataset.
- For non-empty subsets, the process repeats recursively, commencing with the creation of a new node until all data partitions have been addressed.
- Ultimately, the algorithm returns the resulting decision tree structure.

This algorithm is a foundational process for constructing decision trees, commonly applied in tasks involving data classification and predictive modeling within machine learning and data analysis contexts.

DTC is a preferred method for predicting student performance due to its interpretability, ability to handle non-linear relationships, versatility with mixed data types, ease of implementation, and avoidance of overfitting through pruning. DTC is suitable for educational datasets with both categorical and numerical variables, making it applicable to real-world scenarios. Additionally, decision trees can be part of ensemble methods, offering improved predictive accuracy. The transparency of decision tree models is valuable in educational contexts, enabling stakeholders to understand and discuss predictions.

D. Fox optimization (FO)

The Fox Optimization Algorithm (FO) draws inspiration from the hunting behavior of red foxes and is structured around two primary phases: exploitation and exploration. The exploitation phase mimics a fox closing in on its prey, utilizing strategies to optimize the immediate vicinity. Conversely, the exploration phase is influenced by the relative distance between the fox and its target. This algorithm functions with a consistent population of foxes, maintaining a set structure as detailed below [30]:

$$\bar{x} = (x_0, x_1, \dots, x_{n-1}) \quad (5)$$

In the identification of each fox $\bar{x}^t$ within the t-th iteration, a notation $\left(\bar{x}_j^i\right)^t$ is introduced. In this context, i represents the count of foxes, while j denotes the specific coordinates within the solution space, delineated by the dimensions. $(\bar{x})^{(i)} = \left[(x_0)^{(i)}, (x_1)^{(i)}, (x_2)^{(i)}, \dots, (x_{n-1})^{(i)}\right]$ is employed to denote each point within the solution space $< a, b >^n$, where $a, b \in \mathbb{R}$. Furthermore, with regard to the solution space, a function

$f \in \mathbb{R}^n$ is regarded as the standard function of n variables. If the value of this function, $f\left((\bar{x})^{(i)}\right)$, represents a global maximum or minimum within the interval $< a, b >$, then $\left((\bar{x})^{(i)}\right)$ is deemed the optimal solution.

When foxes struggle to find prey, family members embark on the quest for food. When a more promising area is discovered, they share and communicate this location within the population, effectively supporting it and considering the associated cost. The metric utilized for this dissemination relies on the Euclidean squared distance.

$$D((\bar{x}^i)^t, (\bar{x}^b)^t) = \sqrt{\|(\bar{x}^i)^t - (\bar{x}^b)^t\|}\,, \qquad (6)$$

$(\bar{x}^b)$ represents the individuals within the population shifting their positions towards the direction of the best performer.

$$(\bar{x}^i)^t = (\bar{x}^i)^t + \alpha * S * ((\bar{x}^b)^t - (\bar{x}^i)^t), \qquad (7)$$

Here, α is randomly chosen from the range $\left(0, d((\bar{x}^i)^t, (\bar{x}^b)^t)\right)$, while S signifies the 'sign' word. The random value β, ranging between 0 and 1, remains consistent for all individuals in the population. This value embodies the behavior of the fox as:

$$\begin{cases} Stay\ and\ masquerade & if\ \ \beta \leq 0.75 \\ Move\ closer & if\ \beta > 0.75 \end{cases} \qquad (8)$$

An advanced Cochleoid equation elucidates the behavior of individuals when β influences the movement of the population in a given iteration. Two components determine the fox radius: $\phi_0 \in < 0{,}2\pi >$ representing the initial observation angle, and $\alpha \in < 0{,}0.2 >$ as a scaling parameter. This value is preset for all individuals in the population, symbolizing random alterations in distance as the fox approaches the target.

$$r = \begin{cases} a\,\frac{sin\phi_0}{\phi_0} & if\ \phi_0 \neq 0 \\ \delta & if\ \phi_0 = 0 \end{cases} \qquad (9)$$

$$\begin{cases} x_0^{new} = ar * cos(\phi_1) + x_0^{ac} \\ x_1^{new} = ar * sin(\phi_1) + ar * cos(\phi_2) + x_1^{ac} \\ x_2^{new} = ar * sin(\phi_1) + ar * sin(\phi_2) + ar * cos(\phi_3) + x_2^{ac} \\ \cdots \\ x_{n-2}^{new} = ar * \sum_{q=1}^{n-2} sin(\phi_q) + ar * cos(\phi_{n-1}) + x_{n-2}^{ac} \\ x_{n-1}^{new} = ar * sin(\phi_1) + ar * cos(\phi_2) + \cdots + ar * sin(\phi_{n-1}) + x_{n-1}^{ac} \end{cases} \qquad (10)$$

In this context, δ, fluctuating between 0 and 1, stands as a random value set at the beginning of the algorithm, contingent upon prevailing weather conditions. The movement pattern for the population of individuals is articulated as follows:

Where "ac" in x_0^{ac} signifies "actual," and ϕ_1, ϕ_2, ϕ_3, and so on, up to ϕ_{n-1}, all exist within the range of $< 0{,}2\pi >$.

5% of the least successful candidates are selected based on the criterion function to replicate this action in each iteration. This selection is a subjective assumption aimed at introducing slight variations within the group. In iteration t, the two top-performing individuals are chosen for an alpha couple.

The pair comprises $\left(\bar{x}^{(1)}\right)^t$ & $\left(\bar{x}^{(2)}\right)^t$, while the center of the habitat is calculated using a specific equation. The square of the individual Euclidean distance between the couple determines the habitat range.

$$(H^{cntr})^t = \frac{\left(\bar{x}^{(1)}\right)^t + \left(\bar{x}^{(2)}\right)^t}{2} \qquad (11)$$

$$(H^{diamtr})^t = \sqrt{\|(\bar{x}^{(1)})^t - (\bar{x}^{(2)})^t\|} \qquad (12)$$

In this context, 'H' denotes the Habitat. Each iteration involves the selection of a random parameter 'q' ranging from 0 to 1, governing the substitutions conducted throughout the repetition in the following manner:

$$\begin{cases} Reproduction\ Of\ The\ Alpha\ Couple \\ \qquad if\ \ q < 0.45 \\ New\ Nomadic\ Individual \\ \qquad if\ q\ \geq 0.45 \end{cases} \qquad (13)$$

The top two candidates indicated as $\left(\bar{x}^{(1)}\right)^t$ and $\left(\bar{x}^{(2)}\right)^t$, are amalgamated to generate a new candidate, denoted as $\left(\bar{x}^{(rep)}\right)^t$, where "rep" signifies reproduction. This fusion takes place in the following manner:

$$\left(\bar{x}^{(rep)}\right)^t = q\ \frac{\left(\bar{x}^{(1)}\right)^t + \left(\bar{x}^{(2)}\right)^t}{2} \qquad (14)$$

The Steps of the Fox Optimization algorithm is represented as Algorithm 1.

ALGORITHM. 1. PSEUDO-CODE OF FO

```
Commence,
Establish the algorithm's parameters: the fitness functions $f(0)$, the number of iterations $T$, the initial fox observation angle $\phi_0$, the maximum population size $n$, weather conditions $\theta$, and the solution space range $< a, b >$,
Create a population of $n$ foxes randomly distributed within the solution space.
t= 0
while $t \leq T$ do
Define iteration coefficients: fox proximity change ($\alpha$), scaling parameter ($\alpha$).
For every fox within the current population,
Organize individuals based on their fitness function values,
Select $(\bar{x}^b)^t$
Compute the repositioning of individuals
If the new position is superior to the previous one, then
Relocate the fox to the new position,
else
Revert the fox to its previous location,
end if
Determine the parameter β to define the fox's hunting awareness,
If the fox remains unnoticed, then
Calculate the fox's observation radius ($\boldsymbol{r}$)
Compute the repositioning
else
The fox maintains its current position to remain concealed,
end if
end for
Arrange the population following the fitness function,
Eliminate the poorest-performing foxes from the group, or they fall victim to hunters,
Introduce new foxes into the population as nomadic foxes outside the habitat or through reproduction from the alpha couple within the herd
t + +,
end while
Return the fittest fox ($\overline{\boldsymbol{x}}$)$^{\boldsymbol{b}}$,
Stop.
```

E. *Black Widow Optimization (BWO)*

The BWO is a recent and intriguing meta-heuristic approach for tackling complex numerical optimization challenges [31]. BWO incorporates operators commonly found in evolutionary algorithms, akin to Genetic Algorithms (GAs) [31]. Like other evolutionary algorithms, BWO employs criteria resembling natural evolutionary processes, such as selection, reproduction, and mutation, which vary and distinguish it from other evolutionary methods. However, what sets BWO apart is its simulation of the unique mating behavior of black widow spiders. Furthermore, BWO exhibits distinctions from traditional evolutionary algorithms, contributing to its strong performance in solving complex problems. This algorithm draws inspiration from Darwin's theory of natural selection, characterized by species evolving and the emergence of new ones. BWO is known for its rapid convergence and ability to evade local optima, making it well-suited for solving various optimization problems with multiple local optima. This success is attributed to BWO's balanced approach, maintaining harmony between the exploration and exploitation phases. For a visual representation of the BWO process (see Fig. 2).

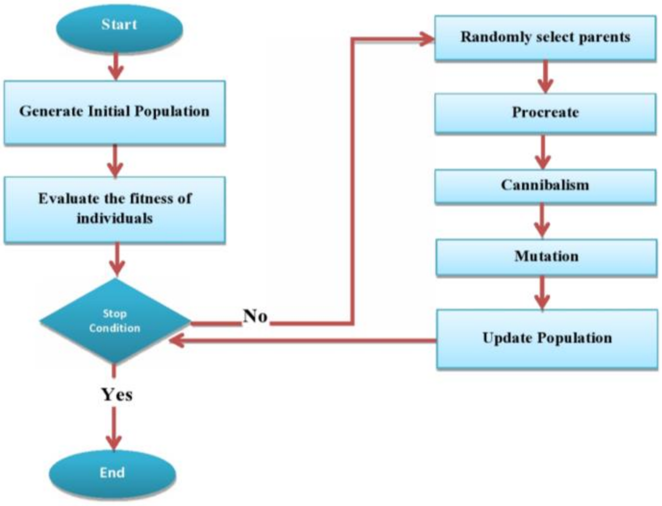


Fig. 2. Flowchart of the BWO.

The primary steps of the BWO can be summarized as follows:

1) *Step one: Initialization*

During this step, the population consists of a specific number of widows, denoted as N, where each widow is represented as an array of size $1 \times N_{var}$, signifying a potential solution to the problem. This array can be described as follows: $widow = \left(x_1, x_2, \ldots\ldots, x_{N_{var}}\right)$, where N_{var} corresponds to the dimensionality of the optimization problem. N_{var} can also be understood as the count of threshold values the algorithm aims to determine. Here, x_i represents the $i - th$ candidate solution within the array.

The fitness of each widow is determined by evaluating the fitness function, denoted as f, for every widow in the set $\left(x_1, x_2, \ldots\ldots, x_{N_{var}}\right)$. This fitness value can be expressed as follows: fitness = f(widow), which is equivalent to $fitness = f\left(x_1, x_2, \ldots\ldots, x_{N_{var}}\right)$. The optimization procedure commences by initializing a population of spiders randomly in a matrix of dimensions $N_{pop} \times N_{var}$. Subsequently, pairs of parents are selected randomly to engage in the reproduction step, which is followed by the mating process. During or after mating, the male black widow is consumed by the female.

2) *Step two: Procreate*

During the procreation step, an alpha (α) array is generated. This alpha array has the same length as a widow array and is filled with random numbers. Subsequently, offspring is generated using alpha (α) and Eq. (14), where x_1 and x_2 represent the parents and y_1 and y_2 denote the offspring. The outcome of the crossover operation is assessed and then stored for further processing.

$$y_1 = \alpha \times x_1 + (1-\alpha) \times x_2 \text{ and } y_2 = \alpha \times x_2 + (1-\alpha) \times x_1 \quad (15)$$

3) Step three: Cannibalism

The cannibalism process can be classified into various categories, including sexual cannibalism, sibling cannibalism, and a commonly observed form in which baby spiders consume their mother. Following the implementation of the cannibalism mechanism, the resulting new population is assessed and saved in a variable referred to as $pop2$.

4) Step four: Mutation

The mutation process involves randomly selecting a number of individuals, denoted as $Mutepop$, from the population to undergo mutation. Each selected solution has two elements within their array randomly exchanged in this mutation operation. After applying mutation, the resulting new population is evaluated and stored in a new population variable, typically named $pop3$. Finally, the new population is obtained by combining (or migrating) the individuals from $pop3$ and $pop2$. Subsequently, this combined population is sorted, aiming to identify the best widow with N_{var} dimensions in terms of threshold values. Algorithm 2 provides the pseudo-code for the BWO algorithm.

ALGORITHM 2: PSEUDO-CODE OF BWO ALGORITHM

```
Initialize: Maximum number of iterations, rate of procreating, rate of Cannibalism, rate of mutation;
while $\boldsymbol{Stopconditionnotmet}$ do
    for $\boldsymbol{i = 1}$ to $\boldsymbol{nr}$ do
        Randomly select two solutions as parents from $\boldsymbol{pop1}$.
        Generate D children
        Destroy father.
        Based on the cannibalism rate, destroy some of the children (newly achieved solutions).
        Save the remaining solutions into $\boldsymbol{pop2}$.
    end for
    Based on the mutation rate, calculate the number of mutation children $\boldsymbol{nm}$.
    for $\boldsymbol{i = 1}$ to $\boldsymbol{nr}$ do
        Select a solution from $\boldsymbol{pop1}$.
        Mutate randomly one chromosome of the solution and generate a new solution.
        Save the new one into $\boldsymbol{pop2}$.
    end for
    Update $\boldsymbol{pop = pop2 + pop3}$.
    Returning the best solution.
    Return the best solution from pop.
end while
```

V. RESULTS AND DISCUSSION

A. Convergence Results

In this study, two powerful metaheuristic optimization algorithms, the FO and BWO, were employed to fine-tune and optimize the DTC model's hyperparameters, particularly the DTFO and DTBW hybrid models. The primary aim was to enhance the predictive accuracy of these models. To evaluate the convergence of these optimization methods, two convergence curves (one related to G1 and the other related to G3) were utilized (see Fig. 3), tracking accuracy over 200 iterations. This curve visually demonstrated the evolution of Accuracy with each iteration, enabling the assessment of convergence progress and rate. In the case of G1 values, both models initially showed similar convergence rates of nearly 0.8, but the DTFO model ultimately achieved higher accuracy (almost 0.94). Notably, a linear pattern in the trend line around the 160-iteration mark indicated the optimal computational efficiency point for the DTFO model. On the other hand, regarding the G3 values, the DTFO model registered a lower convergence value at the beginning and a higher convergence value at the final iteration; it achieved a high convergence value of 0.92 at the final stage.

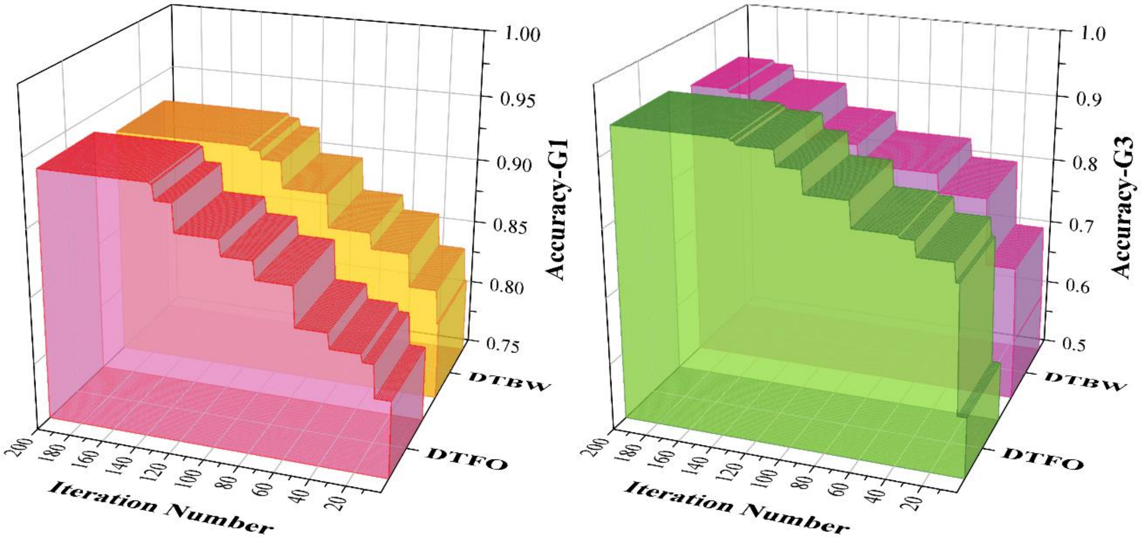


Fig. 3. Convergence of hybrid models.

B. Hyperparameter

Table I displays the results of hyperparameter tuning for four different decision tree models, each associated with a specific target variable (G1 or G3). The hyperparameters include `max_depth` (maximum depth of the tree), `min_samples_split` (minimum samples required to split an internal node), `min_samples_leaf` (minimum samples required at a leaf node), and `max_leaf_nodes` (maximum number of leaf nodes). The values in each cell represent the chosen hyperparameter settings for the corresponding model and target variable. The hyperparameter tuning process aims to optimize the performance of the decision tree models in predicting student outcomes (G1 or G3).

The overall influence involves balancing model complexity and generalization. Higher values tend to lead to more complex models prone to overfitting, while lower values result in simpler models that generalize better. Hyperparameter tuning aims to find the optimal combination for the effective prediction of student outcomes.

TABLE I. RESULT OF HYPERPARAMETER

Hyperparameter	Model (Target)			
	DTFO (G1)	DTBW (G1)	DTFO (G3)	DTBW (G3)
max_depth	71	661	106	467
min_samples_split	0.001	0.209	0.001	0.116
min_samples_leaf	0.0005	0.0038	0.0005	0.0415
max_leaf_nodes	580	5	1270	4

C. Comparing results of predictive models

This study focused on constructing three prediction models employing a classification approach to forecast students' exam performance in Mathematics and systematically improve their forthcoming grades. The models comprised a single Decision Tree Classification (DTC) and two optimized models using the Fox Optimization (FO) and the Black Widow Optimization (BWO). The dataset was split, allocating 70% for training and 30% for testing to assess their predictive performance. Table II and Fig. 4 illustrate the Accuracy, Precision, Recall, and F1-score for training, testing, and all phases across all models in predicting G1 and G3 scores.

- G1 Scores

Among the three models, the DTBW model exhibited superior training performance compared to the others, as evidenced by higher metric values during training than in the testing phase. The maximum metric values achieved by DTBW were 0.937 for all four metrics (Accuracy, Precision, Recall, and F1-Score). On the contrary, the DTC model obtained the lowest values, with 0.822 for Accuracy and Recall, 0.818 for Precision, and 0.82 for F1-Score.

- G3 Scores

Considering the mentioned models (DTC, DTFO, and DTBW), DTFO exhibited superior performance compared to the others, evident from its higher metric values. The maximum metric values achieved by DTFO were 0.934 for Accuracy and Recall and 0.935 for Precision and F1-Score. In contrast, the DTBW model obtained the lowest values, with 0.822 for Accuracy and Recall, 0.825 for Precision, and 0.823 for F1-Score.

TABLE II. RESULT OF PRESENTED MODELS

	Model	**Phase**	**Index values**			
			Accuracy	***Precision***	***Recall***	***F1 _core***
G1	***DTC***	Train	0.914	0.914	0.914	0.912
		Test	0.822	0.818	0.822	0.820
		All	0.914	0.914	0.914	0.912
	DTFO	Train	0.927	0.927	0.927	0.925
		Test	0.831	0.826	0.831	0.825
		All	0.927	0.927	0.927	0.925
	DTBW	Train	0.937	0.937	0.937	0.937
		Test	0.881	0.884	0.881	0.881
		All	0.937	0.937	0.937	0.937
G3	***DTC***	Train	0.916	0.916	0.917	0.915
		Test	0.856	0.854	0.856	0.852
		All	0.916	0.916	0.917	0.917
	DTFO	Train	0.932	0.932	0.932	0.932
		Test	0.864	0.869	0.864	0.866
		All	0.934	0.935	0.934	0.935
	DTBW	Train	0.924	0.924	0.924	0.924
		Test	0.822	0.825	0.822	0.823
		All	0.924	0.924	0.924	0.924

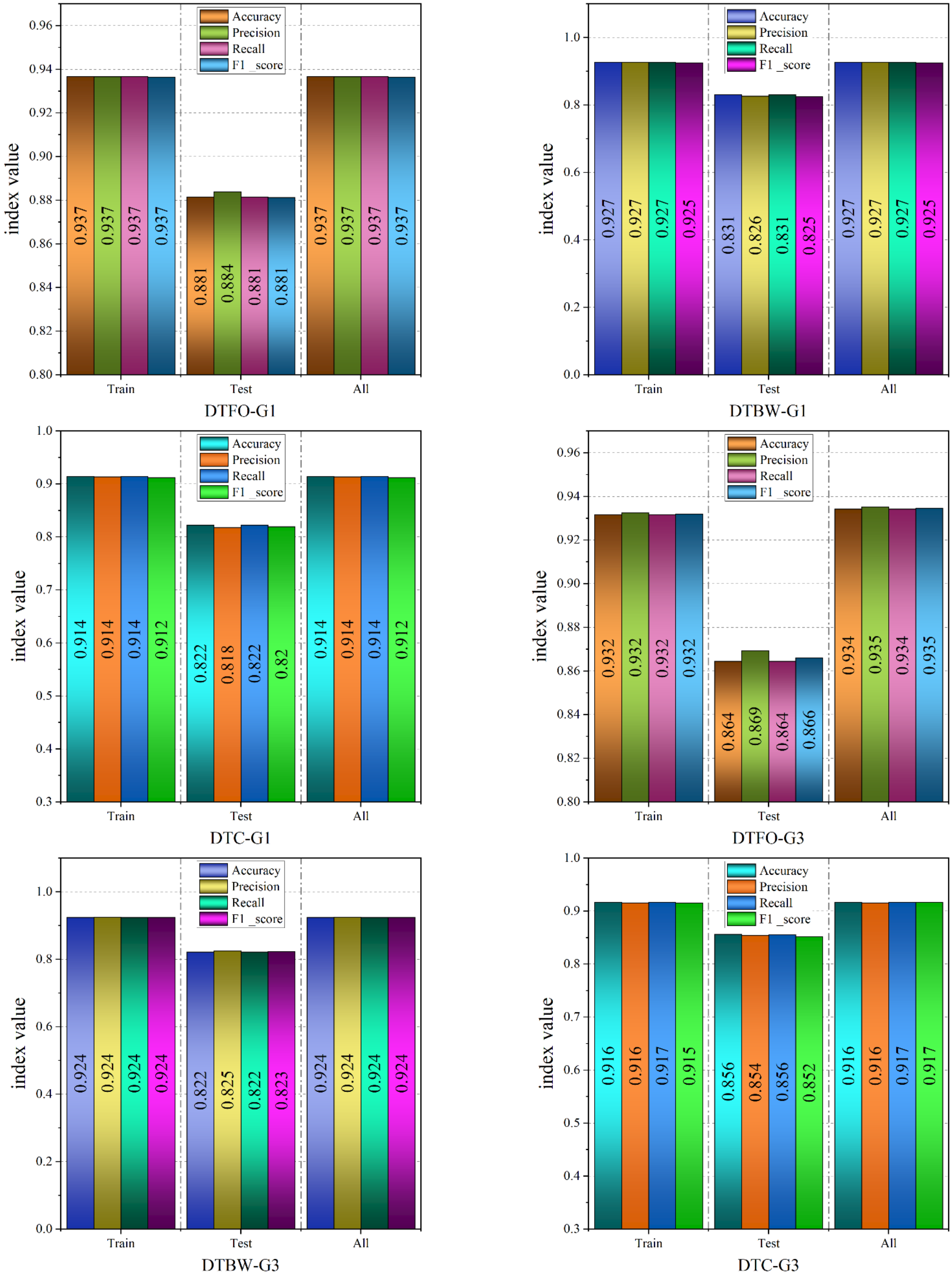


Fig. 4. Column plot for the evaluation of developed models.

Following data processing and a comprehensive evaluation of the models' classification capabilities during the training and testing phases, 395 students were extensively examined based on their test results (G1 and G3 values). These students were categorized into four distinct groups: Poor (comprising students with scores ranging from 0 to 12), Acceptable (encompassing those with scores ranging from 12 to 14), Good (enrolling students with scores ranging from 14 to 20), and Excellent (comprising students with scores ranging from 16 to 20). The Index values for Precision, Recall, and F1-score are presented in Table III for G1 and Table IV for G3, which are used as evaluation metrics for assessing the classification performance of the developed models across the various student categories. A comparative analysis has been conducted in the subsequent section, considering each of these three Index values. As a result of this categorization, in the case of G1, 41 (10.38%) students were identified within the Excellent category, 54 (13.67%) within the Good category, 68 (17.21%) within the Acceptable category, and 232 (58.73%) within the Poor category. On the other hand, regarding G3 values, 40 (10.13%) students were identified within the Excellent category, 60 (15.19%) within the Good category, 62 (15.7%) within the Acceptable category, and 232 (58.73%) within the Poor category.

D. *Precision*

- G1 Scores

The DTFO model demonstrated the highest values in the Good and Poor groups, achieving precision scores of 0.942 and 0.945, respectively. Conversely, the DTBW model obtained a maximum precision value of 0.947 for the Acceptable group. As for the excellent group, the DTC model outperformed others, attaining a precision score of 0.925.

- G3 Scores

The DTC model demonstrated the highest values in the Excellent and Acceptable categories, achieving precision scores of 0.922 and 0.898, respectively. On the other hand, the DTFO model obtained a maximum precision value of 0.974 for the Poor group. As for the Good group, the DTBW model outperformed others, attaining a precision score of 0.9.

E. *Recall*

- G1 Scores

The DTFO model displayed the highest scores in the Excellent, Good, and Acceptable groups, reaching 0.902, 0.907, and 0.897, respectively. When it comes to the Poor group, the DTBW model delivered the top performance with a recall score of 0.978.

- G3 Scores

In the Excellent and Good categories, the DTBW model demonstrated the highest values, achieving Recall values of 0.95 and 0.90, respectively. Furthermore, the DTC model obtained a maximum Recall value of 0.97 for the Poor group. While for the Acceptable group, the DTFO model outperformed others, attaining a score of 0.887.

F. *F1-score*

- G1 Scores

A superior F1-score reflects the model's ability to balance precisely identifying positive cases (Precision) and encompassing all genuine positive cases (Recall). Upon considering all student categories, it becomes evident that the DTFO model demonstrated the highest values in the Good and Acceptable groups, achieving precision scores of 0.925 and 0.91, respectively. In addition, the DTBW model obtained a maximum F1-Score value of 0.956 for the Poor group. Finally, in the case of the Excellent group, the DTC model outperformed others, attaining an F1-Score of 0.914.

- G3 Scores

In the Excellent and Good categories, the DTBW model demonstrated the highest values, achieving F1-Score values of 0.927 and 0.90, respectively. Furthermore, the DTFO model outperformed others in the Poor category, attaining a score of 0.965. While for the Acceptable group, the DTC model obtained a maximum F1-Score value of 0.876.

TABLE III. EVALUATION INDEXES OF THE DEVELOPED MODELS' PERFORMANCE BASED ON G1

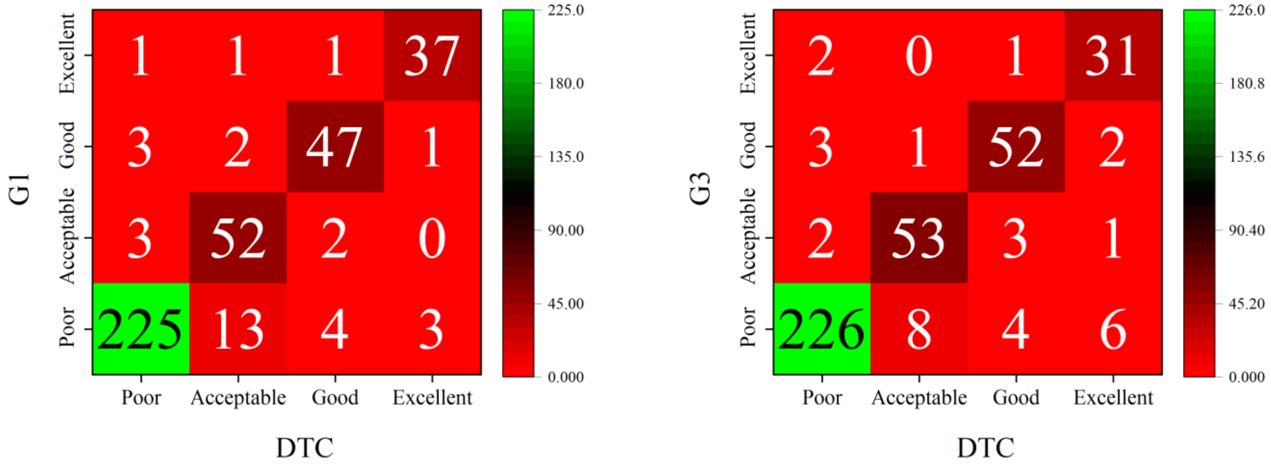
Model	**Grade**	**Index values**		
		Precision	***Recall***	***F1-score***
DTC	***Excellent***	0.925	0.902	0.914
	Good	0.887	0.870	0.879
	Acceptable	0.912	0.765	0.832
	Poor	0.918	0.970	0.943
DTFO	***Excellent***	0.902	0.902	0.902
	Good	0.942	0.907	0.925
	Acceptable	0.922	0.897	0.910
	Poor	0.945	0.961	0.953
DTBW	***Excellent***	0.881	0.902	0.892
	Good	0.906	0.889	0.897
	Acceptable	0.947	0.794	0.864
	Poor	0.934	0.978	0.956

TABLE IV. EVALUATION INDEXES OF THE DEVELOPED MODELS' PERFORMANCE BASED ON G3

Model	Grade	Index values		
		Precision	***Recall***	***F1-score***
DTC	***Excellent***	0.912	0.775	0.838
	Good	0.897	0.867	0.881
	Acceptable	0.898	0.855	0.876
	Poor	0.926	0.970	0.948
DTFO	***Excellent***	0.884	0.950	0.916
	Good	0.898	0.883	0.891
	Acceptable	0.859	0.887	0.873
	Poor	0.974	0.957	0.965
DTBW	***Excellent***	0.905	0.950	0.927
	Good	0.900	0.900	0.900
	Acceptable	0.855	0.855	0.855
	Poor	0.952	0.944	0.948

The confusion matrix illustrated in Fig. 5 provides insights into accurately categorizing students into their respective grades and the misclassification into incorrect categories. In the case of G1 values, the DTFO model correctly categorized 37, 49, 61, and 223 students into Excellent, Good, Acceptable, and Poor classes, respectively, with only 25 students being misclassified. On the other hand, the DTBW and DTC models misclassified 29 and 34 students, respectively. Notably, misclassifications in the two optimized models primarily occurred between neighboring categories, such as 6 and 10 students for DTFO and DTBW, who were mistakenly placed in the Acceptable category instead of the Poor category. According to G3 values, the DTC model correctly categorized 31, 52, 53, and 223 students into Excellent, Good, Acceptable, and Poor classes, respectively, with 33 misclassified students. On the other hand, the DTBW and DTFO models misclassified 30 and 26 students, respectively. In the case of the single DTBW model, 9 students were inaccurately positioned in the Acceptable category instead of the Poor category.

The actual number of students falling into the Poor, Acceptable, Good, and Excellent categories was 232, 68, 54, and 41, respectively, for G1, while 233, 62, 60, and 40 for G3 values. Fig. 6 provides a visual representation of the student distribution across these categories based on measurement and classification model outcomes, facilitating a visual comparison. In the case of G1, the DTFO model exhibited the highest accuracy in correctly classifying students in the Acceptable, Good, and Excellent groups, identifying 61, 49, and 37 students accurately, respectively. In the case of the Poor category, the DTBW model outperformed the other models, correctly classifying 227 students. Regarding the G3 values, the DTFO model exhibited the highest accuracy in correctly classifying students into Acceptable and Excellent groups, identifying 55 and 38 students accurately. When considering the Poor category, the DTC model outperformed the other models, correctly classifying 226 students. Furthermore, according to the Good category, the DTBW model performed best, identifying 54 students correctly.



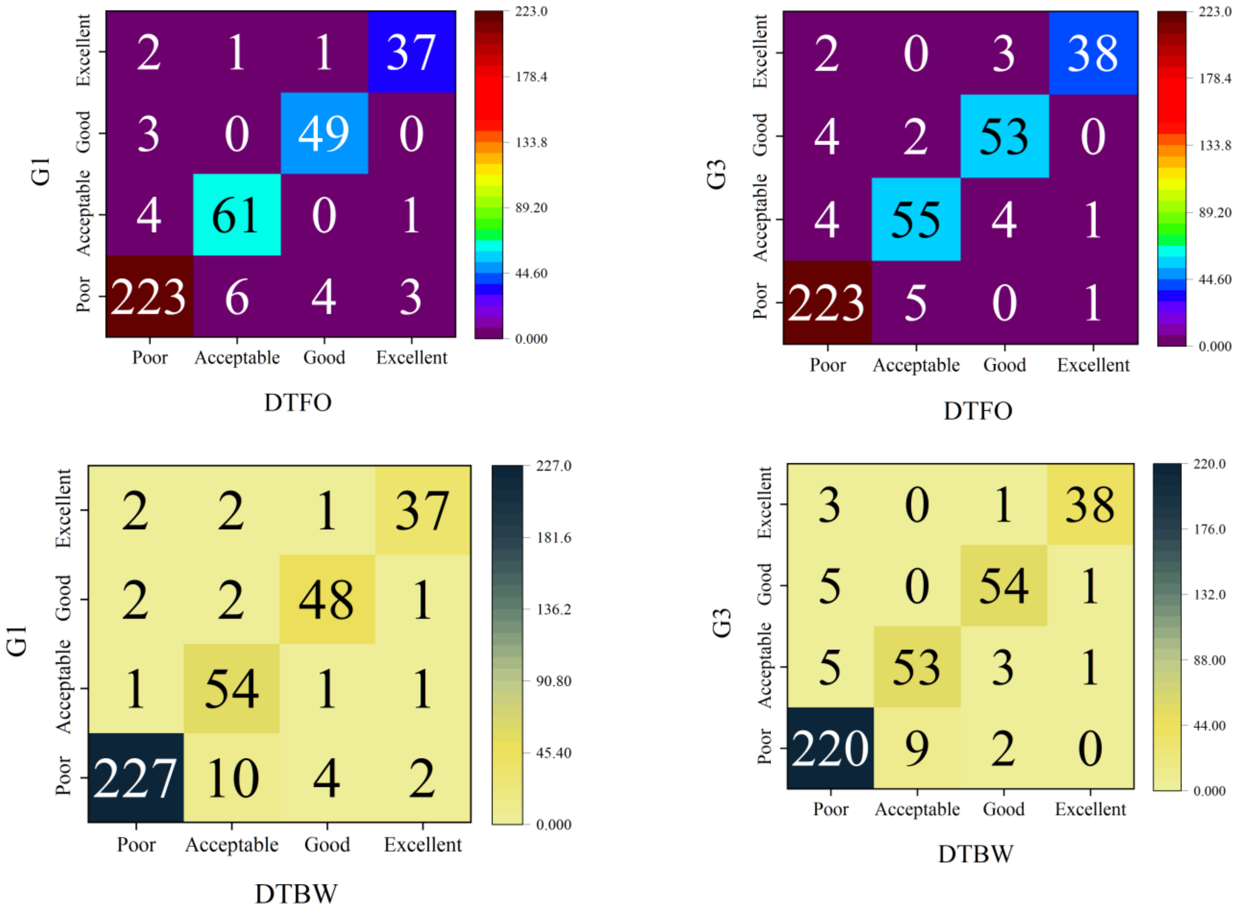
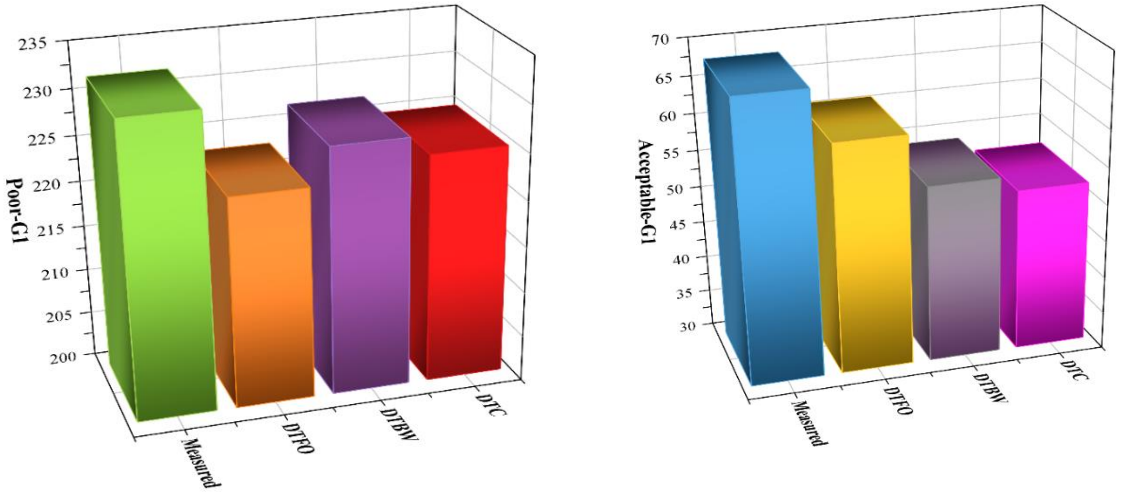


Fig. 5. Confusion matrix for each model's accuracy.



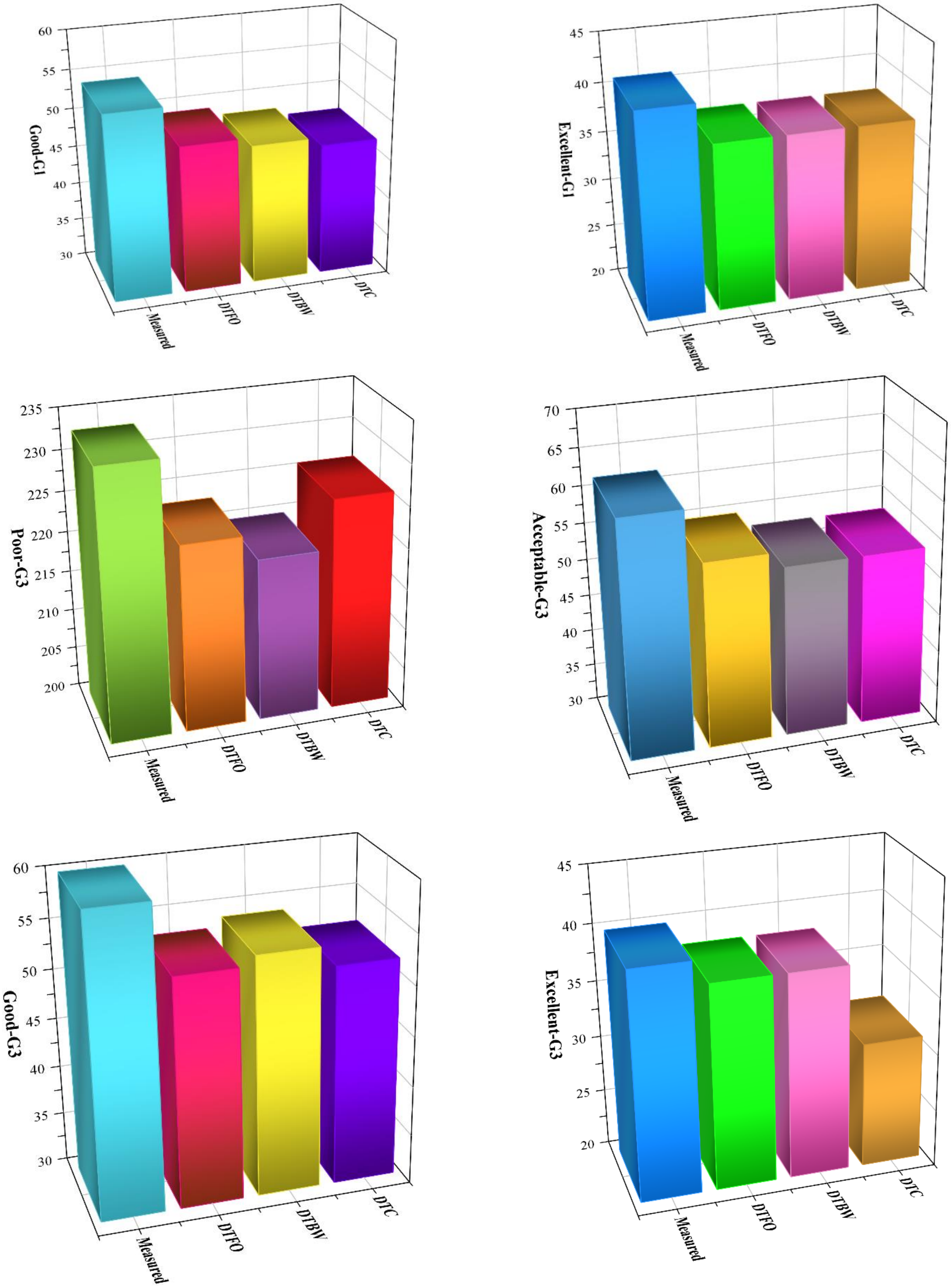


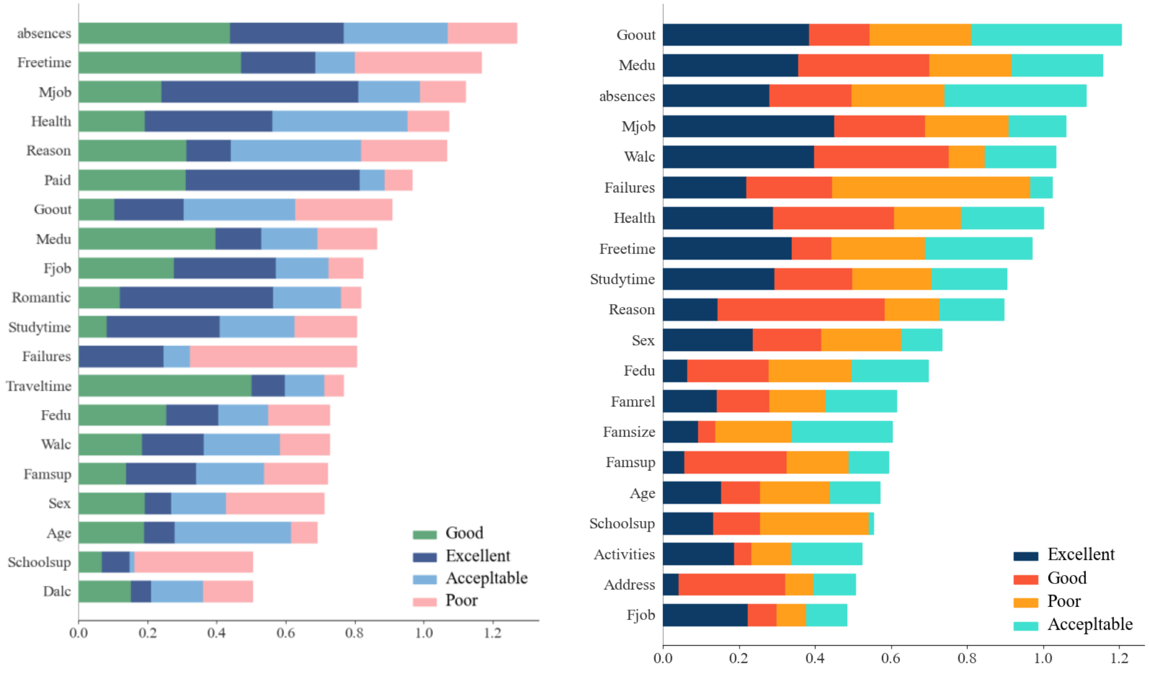
Fig. 6. 3D column plot for the developed models' accuracy compared to measured value.

G. *Sensitivity analyzes*

SHAP (SHapley Additive exPlanations) values, derived from cooperative game theory, allocate feature contributions in ML models. They assess the impact of each feature on a model's prediction for a specific input, providing nuanced and interpretable insights. Adapted for use in ML, SHAP values offer a fair distribution of feature importance, aiding the interpretation of complex models by attributing predictions to individual features.

Fig. 7(a) reveals that "absences," "Freetime," "mother's job," and "Health" stand out as pivotal elements for anticipating G1 performance. Additionally, the plot highlights the fluctuating significance of these features across the four grade levels, indicating that the determinants influencing G1 scores are not uniform for all students. This underscores the variability in the impact of these factors across different grade levels.

On the other hand, for G3 in Fig. 7(b), it was observed that "absences," "Goout," "mother's education," and "mother's job " had the greatest impact on the model's output.



(a)

(b)

Fig. 7. SHAP value for the impact of inputs on model's output a) G1 and b) G3.

VI. Conclusion

This research underscores the crucial significance of predictive models based on data in education. It stresses the need to consider qualitative and quantitative elements for predicting and evaluating students' academic performance. The findings offer valuable guidance for policymakers, educational institutions, and students, aiming to enhance future academic outcomes. The study demonstrates the effectiveness of data mining techniques such as clustering, classification, and regression in understanding and proactively tackling the diverse challenges encountered by undergraduate students. Furthermore, the research introduces an innovative approach by combining the Decision Tree Classification (DTC) model with optimization algorithms such as Fox Optimization (FO) and Black Widow Optimization (BWO). This advanced methodology illustrates how integrating machine learning techniques and optimization algorithms can elevate the Precision and effectiveness of predictive models. It provides a robust toolkit for addressing the evolving challenges in students' academic journeys. The study's thorough evaluation process, which included dividing the models into training and testing sets, reveals that these hybrid models have the potential to enhance the classification capabilities of the DTC model significantly. This enhancement is reflected in notable improvements in Accuracy and Precision. Upon analyzing the results, it has been observed that the potential to significantly enhance the classification capabilities of the DTC model by these hybrid models is increasingly recognized. Based on the results, it can be concluded that:

- In the case of G1 values, a marked improvement in Accuracy was achieved by applying FO and BWO optimization algorithms to the DTC model, with an increase of 1.42% and 2.51%, respectively. When the 395 students were categorized based on their final grades, the exceptional ability of the BWO to augment classification Accuracy became evident. Specifically, the DTBW model displayed an impressive Accuracy rate of 93.7%, accurately classifying the majority of students, whereas the DTFO and DTC models misclassified 6.33% and 8.6% of all students, respectively.
- With respect to G3 values, the improvement of Accuracy through the application of FO and BWO optimization algorithms to the DTC model was 1.96% for the application of FO and 0.87% for BWO. The DTFO model displayed an impressive Accuracy rate of 93.4%, accurately classifying the majority of students, whereas the DTBW and DTC models experienced misclassification rates of 7.59% and 8.35%, respectively.

The study sought to revolutionize academic performance prediction in education, assuming that predictive models significantly influence outcomes. Recognizing the holistic nature of student evaluation, it justified the importance of both qualitative and quantitative elements. Integration of machine learning with optimization algorithms was assumed to enhance predictive models, supported by literature. Standard practices in machine learning, such as thorough evaluation using training and testing sets, were assumed to reflect model effectiveness. The assumption that misclassification rates indicate their direct measurement of prediction accuracy justified model performance. The study assumed that an increase in accuracy corresponded to improved classification capabilities, signifying enhanced predictions of students' final grades. Additionally, the assumption was made that optimization algorithms, specifically FO and BWO, led to marked improvements by fine-tuning decision tree models. Moreover, the research aimed to transform academic performance prediction in education, aligning with its overarching goal. Assumptions were strategically made to support this objective, including the significant influence of predictive models on academic outcomes, the importance of both qualitative and quantitative elements in predictions, and the enhancement of models through the integration of machine learning and optimization algorithms. Standard machine learning practices were assumed to reflect model effectiveness, with chosen metrics aligning with the study's goal of accurate predictions. The research assumed that improvements in accuracy corresponded to enhanced classification capabilities and that optimization algorithms led to marked improvements.

FUNDING

2021 Annual Major Scientific Research and Cultivation Funding Project of Hubei Institute of Fine Arts (Project No. 21ZD11); National University Foreign Language Teaching and Research Key Project Supported by Shanghai Foreign Language Education Press (Project No. 2023HB0038).

REFERENCES

[1] S. Natek, M. Zwilling, Student data mining solution–knowledge management system related to higher education institutions, Expert Syst Appl 41 (2014) 6400–6407.

[2] Y. Zhao, C. Zhang, Y. Zhang, Z. Wang, J. Li, A review of data mining technologies in building energy systems: Load prediction, pattern identification, fault detection and diagnosis, Energy and Built Environment 1 (2020) 149–164.

[3] D. Kabakchieva, K. Stefanova, V. Kisimov, Analyzing university data for determining student profiles and predicting performance, in: Educational Data Mining 2011, 2010.

[4] C. Romero, S. Ventura, Educational data mining: A survey from 1995 to 2005, Expert Syst Appl 33 (2007) 135–146.

[5] C. Romero, S. Ventura, M. Pechenizkiy, R.Sj. Baker, Handbook of educational data mining, CRC press, 2010.

[6] R.S.J.D. Baker, K. Yacef, The state of educational data mining in 2009: A review and future visions, Journal of Educational Data Mining 1 (2009) 3–17.

[7] A. Ahmed, I.S. Elaraby, Data mining: A prediction for student's performance using classification method, World Journal of Computer Application and Technology 2 (2014) 43–47.

[8] E. Chandra, K. Nandhini, Knowledge mining from student data, European Journal of Scientific Research 47 (2010) 156–163.

[9] H.A.A. Hamza, P. Kommers, A review of educational data mining tools & techniques, International Journal of Educational Technology and Learning 3 (2018) 17–23.

[10] M.M.A. Tair, A.M. El-Halees, Mining educational data to improve students' performance: a case study, International Journal of Information 2 (2012).

[11] F. Ünal, Data mining for student performance prediction in education, Data Mining-Methods, Applications and Systems 28 (2020) 423–432.

[12] J.-M. Trujillo-Torres, H. Hossein-Mohand, M. Gómez-García, H. Hossein-Mohand, F.-J. Hinojo-Lucena, Estimating the academic performance of secondary education mathematics students: A gain lift predictive model, Mathematics 8 (2020) 2101.

[13] M.R. Apriyadi, D.P. Rini, Hyperparameter Optimization of Support Vector Regression Algorithm using Metaheuristic Algorithm for Student Performance Prediction, International Journal of Advanced Computer Science and Applications 14 (2023).

[14] C. Márquez-Vera, A. Cano, C. Romero, S. Ventura, Predicting student failure at school using genetic programming and different data mining approaches with high dimensional and imbalanced data, Applied Intelligence 38 (2013) 315–330.

[15] C. Romero, 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) 40 (2010) 601–618.

[16] B. Sekeroglu, K. Dimililer, K. Tuncal, Student performance prediction and classification using machine learning algorithms, in: Proceedings of the 2019 8th International Conference on Educational and Information Technology, 2019: pp. 7–11.

[17] A.K. Pal, S. Pal, Data mining techniques in EDM for predicting the performance of students, International Journal of Computer and Information Technology 2 (2013) 2279–2764.

[18] D. Kabakchieva, Student performance prediction by using data mining classification algorithms, International Journal of Computer Science and Management Research 1 (2012) 686–690.

[19] E. Osmanbegovic, M. Suljic, Data mining approach for predicting student performance, Economic Review: Journal of Economics and Business 10 (2012) 3–12.

[20] S. Huang, N. Fang, Predicting student academic performance in an engineering dynamics course: A comparison of four types of predictive mathematical models, Comput Educ 61 (2013) 133–145.

[21] L. Ramanathan, S. Dhanda, D.S. Kumar, Predicting students' performance using modified ID3 algorithm, International Journal of Engineering and Technology 5 (2013) 2491–2497.

[22] E.B. Costa, B. Fonseca, M.A. Santana, F.F. de Araújo, J. Rego, Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses, Comput Human Behav 73 (2017) 247–256.

[23] F. Marbouti, H.A. Diefes-Dux, K. Madhavan, Models for early prediction of at-risk students in a course using standards-based grading, Comput Educ 103 (2016) 1–15.

[24] Y.-H. Hu, C.-L. Lo, S.-P. Shih, Developing early warning systems to predict students' online learning performance, Comput Human Behav 36 (2014) 469–478.

[25] D. Thammasiri, D. Delen, P. Meesad, N. Kasap, A critical assessment of imbalanced class distribution problem: The case of predicting freshmen student attrition, Expert Syst Appl 41 (2014) 321–330.

[26] M. Pandey, S. Taruna, A multi-level classification model pertaining to the student's academic performance prediction, Int J Adv Eng Technol 7 (2014) 1329.

[27] H. Sharma, S. Kumar, A survey on decision tree algorithms of classification in data mining, International Journal of Science and Research (IJSR) 5 (2016) 2094–2097.

[28] X. Luo, Efficient English text classification using selected machine learning techniques, Alexandria Engineering Journal 60 (2021) 3401–3409.

[29] D. Połap, M. Woźniak, Red fox optimization algorithm, Expert Syst Appl 166 (2021) 114107.

[30] V. Hayyolalam, A.A.P. Kazem, Black widow optimization algorithm: a novel meta-heuristic approach for solving engineering optimization problems, Eng Appl Artif Intell 87 (2020) 103249.

[31] J. Holland, Adaptation in natural and artificial systems, univ. of mich. press, Ann Arbor 7 (1975) 390–401.