Computer Science and Information Systems 22(3):881–905 https://doi.org/10.2298/CSIS241022050S

Classification and Forecasting in Students' Progress using Multiple-Criteria Decision Making, K-Nearest Neighbors, and Multilayer Perceptron Methods

Slađana Spasić[1] and Violeta Tomašević[2]

[1] University of Belgrade – Institute for Multidisciplinary Research
Kneza Višeslava 1, 11030 Belgrade, Serbia
sladjana@imsi.bg.ac.rs
[2] Singidunum University, Faculty of Informatics and Computing
Danijelova 32, 11000 Belgrade, Serbia
vitomasevic@singidunum.ac.rs

Abstract. The research paper addresses students' performance in higher education. It proposes using the MCDM method - Promethee II to assess students' knowledge and the K-Nearest Neighbors (KNN) and Multilayer Perceptron (MLP) methods for grade classification. The main goals are tracking and diagnosing students' knowledge levels, predicting their outcomes, and providing tailored recommendations. It helps to identify students at risk of not passing the course and evaluates teaching methods. This encourages student engagement and progress during the course. The research demonstrates the suitability of Promethee II, MLP, and KNN methods for effectively monitoring, classifying, and predicting students' progress during the semester, enhancing the objectivity of the assessment process.

Keywords: Promethee II, MLP, KNN, student's grades mark classification, student's achievement forecasting, Matthews Correlation Coefficient, Class Balance Accuracy

1. Introduction

The primary objective of higher education is to provide students with academic and professional knowledge in specific areas, which is evaluated based on the grades they achieve in exams. Educational Data Mining is a new field that examines academic performance to improve educational effectiveness [8]. Predicting student academic performance has been a major focus in the field of education. In the past decade, there has been a growing interest in understanding student performance in learning management systems due to recent advancements in artificial intelligence, data mining, and the increasing influence of outcome-based theory in education. Developed models have generally analyzed student data to predict various forms of learning outcomes, such as student achievements, dropout and at-risk rates, and feedback and recommendations. Study [34] analyzed relevant research from this period 2010-2020, showed that learning outcomes were predominantly measured by class standings and achievement scores, and regression and supervised machine learning models were commonly used to categorize student performance. Among these models, Neural Networks and Random Forests (RF) exhibited the highest prediction performance, while linear regression models fared the worst [34]. In higher education, extensive research is conducted on student academic performance to address challenges such as underachievement and university dropout rates [14]. It is important to evaluate student performance in a course. This not only helps to determine their success, but also enables the identification of students who may be at risk of dropping out. Recent studies have shown that dropout rates in some European countries were between 14.7% to 34.1% in 2014 [9], while in Latin America, dropout rates are as high as 50%, leading to delayed completion of higher education for many students [2]. To address this, it is important to develop effective predictive models to identify at-risk students in a timely manner and provide them with personalized feedback and support.

Academicians measure student success in various ways, including final grades, grade point averages, and socio-economic aspects. Computational efforts, particularly those using data mining and learning analytics techniques, aim to improve student performance [6]. The timely prediction of student performance can help identify low-performing students and enable early interventions by educators, such as advising, progress monitoring, intelligent tutoring systems development, and policymaking [39]. The advanced methods utilized in learning analytics to predict student success are broadly categorized into supervised learning, unsupervised learning, data mining, and statistical approaches [21, 37]. Each category encompasses a plethora of intelligent algorithms, such as Artificial Neural Networks, Support Vector Machine, K-Nearest Neighbor (KNN), and RF. The factors influencing student performance are extensively researched, encompassing both academic (e.g., pre-admission scores and entry qualifications) and non-academic factors (gender, ethnicity, parents' socioeconomic status, emotional intelligence and resilience) [24, 15, 33]. With the increased use of distance, online, and hybrid learning, especially during the COVID-19 pandemic, it is important to develop fair assessment methods.

A recent study aimed to build a model using data mining techniques to test, predict, and understand the academic performance of IT students [23]. Students engage in planned activities to enhance their knowledge and achieve academic success. It is crucial to develop a method to predict overall performance and identify at-risk students early in the course and provide valuable feedback to teachers on the effectiveness of their teaching [30].

Until 2013, most studies used statistical methods and linear programming rather than neural network methods for academic achievement classification [24, 30]. The first study on predicting academic achievement compared four mathematical models, including multiple regression, multi-layer perceptual network, radial basis model, and support vector machines [22]. The focus of our research was on creating objective assessment methods and predicting learning outcomes to enhance teaching and learning techniques. To ensure that students are given appropriate feedback on time, it would be useful to monitor their progress throughout the semester. To address this problem, our proposed solution involves diagnosing a student's current level of knowledge, predicting their expected final outcome based on that state, and providing appropriate recommendations to the student. Additionally, it is crucial to continuously evaluate our teaching methods to ensure that they are appropriate and effective, and make any necessary changes to improve the learning experience for all participants. If a significant number of students are not achieving the expected level of progress, proactive steps will be taken to reevaluate our teaching methods and make any necessary changes to ensure that all participants receive the best possible education.

The primary objective of our research was to monitor the activities and progress of course participants throughout the semester to ensure they were on track to complete the course. If a student is not making sufficient progress, they should be advised to increase their efforts. To address this issue, we have employed a combination of Multiple-Criteria Decision-Making (MCDM) alongside classification methods, specifically KNN and Multi-Layer Perceptron (MLP). In previous research, this problem was addressed by applying various modern techniques, or a combination of these techniques, to data obtained through traditional methods of monitoring student success. These conventional methods typically summarize students' achievements in course activities in a scaled format. The innovative aspect of the proposed approach is introducing a more sophisticated way to monitor students' progress, utilizing multi-criteria analysis to gather quality input data for the prediction process. For this purpose, we employ the outranking-based Promethee II method [10]. This method was chosen from a wide range of MCDM methods because it gives an opportunity for a precise and detailed definition of the decision-maker's attitude towards different decision criteria.

The Promethee II method utilizes weighting coefficients and various types of preference functions assigned to the criteria. Unlike traditional approaches, this method enables lecturers to express their attitudes towards course activities in greater detail. By assigning weight coefficients, lecturers can favor or disfavor specific activities and establish their relative importance in the overall evaluation process. Additionally, by selecting the appropriate preference function and setting its thresholds, lecturers can accurately convey their personal views regarding specific activities.

Our proposal is to assess students' objective progress throughout the semester using the Promethee II method. At designated time points, we will monitor the results achieved by students on various course activities. The set of results obtained by a student at a specific moment represents their current state, meaning that throughout the semester, students pass through states that are time-dependent. The progress made by a student between two consecutive states can be quantified using the Promethee II method by treating states as alternatives and course activities as decision criteria. By comparing the net flows (which represent the output of the Promethee II method) of successive states, we can determine how much the next state is objectively better or worse than the previous one. This novel application of the Promethee II method is distinct because, in earlier applications, the alternatives were not time-dependent; instead, they represented a set of options that could address the problem. Although the temporal aspect was introduced into the Promethee II method in prior research [5], it was done in a different context, focusing on dynamic threshold settings to accommodate the decision-maker's temporal preferences.

Based on the results obtained using the Promethee II method, progression functions are generated that clearly illustrate the dynamics of student advancement throughout the semester. These functions are discrete and quantitatively describe the level of progress at specific time points and will be utilized as inputs for the MLP and KNN classifiers. The MLP and KNN classifiers will be compared to determine which one performs better. In practice, they can be used alternatively. Recent research [1, 40] has identified these classifiers as the most suitable options for this type of classification. Selecting the right algorithm is crucial for creating an effective predictive model. For example, authors of [4] found that logistic regression outperformed RF and KNN when predicting student dropout rates. Additionally, study [7] identified MLP, Logistic Regression, Support Vector Machines, and RF as the most accurate algorithms for various STEM programs at a Brazilian university. Despite various research reports discussing the adequacy and accuracy of different classification methods, we chose MLP due to its accurate predictions. Additionally, we selected the KNN model because it allows us to group students with similar performance levels in the course. This approach enables us to form groups of 3 to 20 students, allowing for personalized attention. As a result, weaker students can improve their progress, while more successful students can enhance their knowledge through more advanced lessons.

The generalizability of these models poses a significant challenge, as it is crucial to ensure that models trained on one group can also be effectively applied to others. While some researchers have experimented with ensemble methods [12], a one-size-fits-all approach is not practical. Differences in instructional context [18] and student demographics [31] can influence the effectiveness of a model. Therefore, it is essential to develop customized models for each degree program while recognizing that predictive performance may vary over time. Regularly assessing these models in their specific contexts is vital for effectively reducing dropout rates.

To implement the proposed solution in real-world educational settings, the lecturer needs to define several key components for the course:

1. Course activities – These are the decision-making criteria.
2. Lecturer's attitude – This involves determining the weight of each activity in the overall process and identifying which deviations in the earned points are significant, along with the degree to which they matter (preference functions).
3. Monitoring dynamics – This refers to the specific moments when progress will be tracked.

This information serves as the input data for software that utilizes the Promethee II method, a simplified and easily implementable version of the decision-making tool. The software then generates progress functions based on the results of multi-criteria analysis. Subsequently, these progress functions are input for another software application that implements various classification methods. For this research, data analysis and predictions were performed using Microsoft Excel and IBM SPSS Statistics 25 software (IBM, USA), which provided the environment for executing the proposed procedure.

The effectiveness of the approach improves as the volume of input data for the classifiers increases, leading to more reliable predictions. A larger set of input data is generated when there is a larger group of students in the course and when the system is used over a longer period, such as when the same lecturer teaches the course for several years. If needed, this data can be filtered by different time periods.

The progress of each student in the course is tracked using an appropriate progress function. When a new student enrolls, a new progress function is created, which contributes to the dataset used for predictions. This makes the system scalable in terms of the number of students, enhancing performance as the number increases. Additionally, the system can be expanded by adding new subjects. The input data can vary from subject to subject, depending on the lecturer, since the subjects are mutually independent. However, if this is the case, modifications will be necessary in the Excel implementation, although it is template-based.

The system is also scalable concerning the classification methods employed. The progress functions are generated independently of the classification method, allowing them to serve as input data for different classifiers.

To validate our approach experimentally, we should conduct a longitudinal study that follows at least 2-3 generations of students across one or more courses. Furthermore, we need to statistically test the hypothesis that the distribution of grades varies between students who experienced traditional teaching methods and those who were taught using new approach.

After the introduction in section 1, section 2 covers materials and methods, explaining the participants' data, problem statement, proposed solution, and implementation. Section 3 details the results of the Promethee II method application, classification, and statistical analysis. Section 4 contains the discussion and conclusion.

2. Material and Methods

2.1. Participants and Data Set

This research includes 400 anonymous students' data at Singidunum University in Belgrade, Serbia. The real data come from the learning activities of the students who attended the subject of Computer Architecture and Organization, which is performed in the first year of undergraduate studies at the Faculty of Informatics and Computing and the Faculty of Technical Sciences over three consecutive academic years (2021/22-2023/24). Only data related to teaching activities within the mentioned course were analyzed and all personal data was excluded. The students are registered under numerical codes to keep their identities anonymous. The data included in this analysis is represented with continuous numerical values, only the grade mark is discretized. The students' data has the following organization: student ID, attendance, activity, homework, test, and grade mark from 5 to 10. Data is collected for at five different time points (t_2–t_6). The grade mark is the output value based on the teacher's criterion. This study was conducted by the consensus on the design of the study of the Singidunum University.

2.2. Problem Statement

Let O be an object in the system S which at time t is in the state s_t. The object represents a student who attends the subject. The state s_t is determined by four parameters: attendance (p_1), activity (p_2), homework (p_3), and test (p_4). It can be represented as $p_1(s_t)$,. . . ,$p_4(s_t)$. Object state parameters are recorded in six times in moments t_j, $j = 1, \ldots, 6$ (moment t_1 represents the initial state in which all parameters are zero, i.e., $p_q(s_1) = 0$, $q = 1, \ldots, 4$), and thus the state matrix of the object $M_{6\times4}$ is obtained:

$$M = \begin{bmatrix} p_1(s_1) & p_2(s_1) & p_3(s_1) & p_4(s_1) \\ p_1(s_2) & p_2(s_2) & p_3(s_2) & p_4(s_2) \\ p_1(s_3) & p_2(s_3) & p_3(s_3) & p_4(s_3) \\ p_1(s_4) & p_2(s_4) & p_3(s_4) & p_4(s_4) \\ p_1(s_5) & p_2(s_5) & p_3(s_5) & p_4(s_5) \\ p_1(s_6) & p_2(s_6) & p_3(s_6) & p_4(s_6) \end{bmatrix} \tag{1}$$

The recording timing aligns with teaching activities and does not need to be evenly spaced. State s_6 indicates if the desired outcome is achieved. If not, it suggests ineffective teaching or insufficient learning. Early identification is crucial to avoid negative outcomes.

It is important to note that the selection of parameters p_1–p_4 was based on the structure of the subject for which the study was conducted. However, the proposed solution is flexible and can be easily adapted to meet the needs of any subject. The lecturer can define an arbitrary number of different parameters based on the content and scope of the material being studied, the group of students enrolled in the course, the assessment goals, and any personal or other requirements relevant to the course.

Our model can incorporate additional data to predict the final success or failure of a course. Personal factors, such as socioeconomic status, demographic details, psychological aspects, and academic performance, can be included in the Promethee II analysis as predictors of success. While many researchers have extensively examined the factors influencing student success, our university currently considers only a limited set of data during enrollment. These data include name, age, gender, place of birth, educational background, citizenship, pre-enrollment results, and entry qualifications. In this paper, it is theoretically feasible to use these data as predictors of success, with their inclusion at the additional time point (t_0). Also, it is possible to include the initial test at time point t_1. Specifically, we could utilize the input data collected during the enrollment process, which is available in the university's information system, similar to the approach taken by [1]. However, we refrained from using these data in this analysis due to personal data protection regulations. Importantly, if individual student consent were obtained, our model could potentially include these additional data. Including them might improve the accuracy of our predictions. According to the findings presented in the paper [25], variables related to grade point average, socioeconomic factors, and course completion rates could positively impact our model's effectiveness.

Additionally, the lecturer can establish the dynamics of monitoring student progress, including the timing and number of evaluations, in line with the teaching methods used in the course to achieve the desired outcomes.

2.3. Proposed Problem Solution

The proposed solution aims to identify the state of the object that indicates an unfavorable outcome based on the teacher's experiences with the teaching procedure over the past two years. Valid conclusions drawn from these experiences are applied to real student or group data to identify a critical state and make an appropriate decision. Fig. 1 shows the structure of the proposed system model.

Let S^{400} be the system of 400 objects which consists of two disjunctive subsystems S^{300} (students who have attended the subject in the previous three years) and S^{100} (new students from current academic year) of 300 and 100 objects, respectively. All of these systems are extensions of the system S (defined in Sec. 2.2). The teaching activities of the defined procedure are applied to the objects of the S^{400} system. The data subset of S^{300} provides training and the data subset of S^{100} test data with a ratio of 3:1.

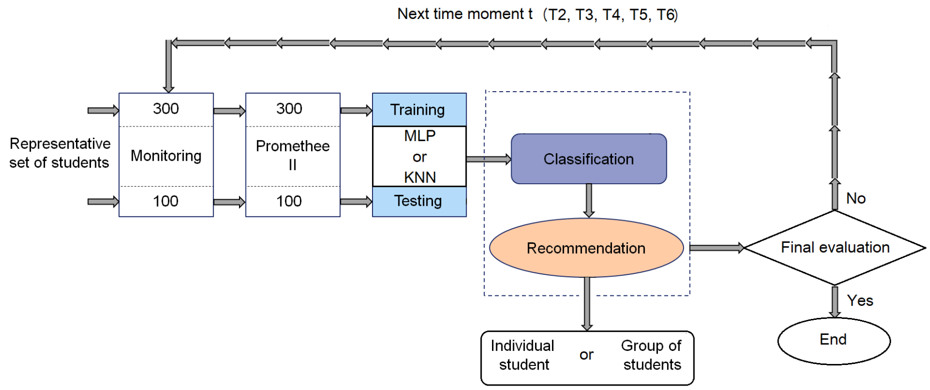

Fig. 1. The structure of the proposed system model

2.4. System Model Implementation

Monitoring. During the course, monitoring was done by recording the state matrices of all participants M_i, $i = 1, \dots, 400$. All state parameters (attendance, activity, homework and test) are cumulative. Therefore, each state parameter at time t is the sum of the value of that parameter at time $t-1$ and the result achieved between these time points. Attendance at lectures (p_1) was scored with a maximum of 5 points (1 point between every two recordings, i.e., $5 \cdot 1$). Class activity (p_2) was assessed by the lecturer in the range of 0–10 points ($5 \cdot 2$ points). Students could win a maximum of 35 points ($5 \cdot 7$ points) on homework (p_3), and up to 50 points ($5 \cdot 10$ points) on tests (p_4).

For example, the state matrix from (1) with the results achieved by the O_{83} student is given by (2). The state matrices have a first row of zeros because we assumed that all students start the course with the same level of knowledge. However, if we were to conduct an entry test, this row would be filled with numbers greater than or equal to zero.

$$M_{83} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0.5 & 0 & 4 & 7 \\ 1.5 & 1 & 8 & 14 \\ 2.5 & 2 & 11 & 21 \\ 3.5 & 3.5 & 16 & 29 \\ 4.5 & 5.5 & 20 & 35 \end{bmatrix} \tag{2}$$

Multi-criteria Analysis. Multi-criteria analysis was conducted on each state matrix M_i. The objective is to evaluate how much each state of the object O_i is better or worse than the previous one. Since every two consecutive states from M_i are compared, the total number of comparisons is 2000. The states are compared using the Promethee II method. Five evaluation tables $T_i(j)$, $j = 1, \dots, 5$, are created for the object O_i. The table $T_i(j)$ establishes a connection between the set of alternatives representing the two successive states through which the object O_i passed $A = \{s^i_{j+1}, s^i_j\}$ and the set of criteria representing the state parameters $C = \{p_1, p_2, p_3, p_4\}$ (Table 1).

Table 1. Evaluation table

$T_i(j)$	p_1	p_2	p_3	p_4
s^i_{j+1}	$p_1(s^i_{j+1})$	$p_2(s^i_{j+1})$	$p_3(s^i_{j+1})$	$p_4(s^i_{j+1})$
s^i_j	$p_1(s^i_j)$	$p_2(s^i_j)$	$p_3(s^i_j)$	$p_4(s^i_j)$

The nature of the introduced criteria is such that we strive to maximize them because they positively contribute to the desired goal. As all criteria are not equally important, they have been assigned priorities.The test (p_4) has the highest priority 5 because it directly reflects acquired knowledge. Homework (p_3) weights 2.5 because it reflects knowledge, but homework is not time-critical, and allows external assistance. Activity (p_2) has priority 1 as it reflects the lecturer's objective impression of the course participant, while attendance (p_1) weights 0.5 as it only reflects the physical presence in the classes. These priorities have been normalized to obtain non-negative relative weight coefficients (w_q, $q = 1, \ldots, 4$) for the criteria, with the sum of these coefficients equaling 1.

In general, different priorities can be assigned for various parameters depending on the academic discipline of the course, the method of assessing student's knowledge, and the set of criteria. For example, a teacher of foreign languages or art history history might prioritize attendance and participation more than a teacher of mathematical analysis would. In contrast, for students in a mathematical analysis course, greater emphasis may be placed on homework and tests.

Our approach allows lecturer the flexibility to define evaluation parameters according to their preferences. Based on their extensive teaching experience, the lecturer associates with each criterion one of the six preference functions recommended by the authors of the Promehtee II method (*Usual*, *U-Shape*, *V-Shape*, *Level*, *Linear*, and *Gaussian*), the one he considers most suitable for that parameter. Otherwise, the Promethee II method allows the addition of new preference functions, so their set can be expanded if necessary.

The preference function reflects the analyst's attitude towards the value difference between the two alternatives.

For the criterion p_q, $q = 1, \ldots, 4$, the difference for alternatives s^i_{j+1} and s^i_j is calculated as

$$d_q(s^i_{j+1}, s^i_j) = p_q(s^i_{j+1}) - p_q(s^i_j)\,. \tag{3}$$

The analyst uses thresholds to indicate the significance of the difference and to what degree it matters to him. This shows his preference for alternatives based on a specific criterion. This can be represented by the preference function in the form of a graphical dependence of the preference towards the alternative s^i_{j+1} in relation to the alternative s^i_j, denoted by $P = P_q(s^i_{j+1}, s^i_j)$, and the difference $d = d_q(s^i_{j+1}, s^i_j)$ from (3).

Among the six preference functions proposed by the author of the Promethee II method, three are associated with the criteria p_q, $q = 1, \ldots, 4$ (Fig. 2).

Criteria p_1 and p_2 are associated with *Level* preference function because the difference of up to 20% in attendance (0.2 out of 1 point) and up to 25% in activity (0.5 out of 2 points) is considered small enough to be neglected. Significant preference is given only at differences of 80% for presence (0.8 out of 1 point) and 75% for activity (1.5 out of 2 points). Criterion p_3 was assigned a *V-Shape* preference function with a threshold 7 that shows that any difference in the number of points on homework is important with linear growth. *Gaussian* preference function with a threshold 5 is associated with criterion p_4 as is often used to assess the success of results achieved in exams.

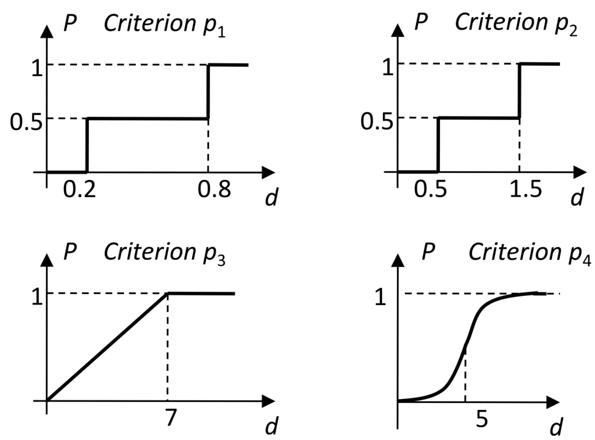

Fig. 2. Preference functions: p_1 – attendance, p_2 – activity, p_3 – homework, and p_4 - test

The Promethee II method is applied five times over the state matrix M_i, once for each evaluation table $T_i(j)$, $j = 1, \ldots, 5$. In general, the Promethee II method compares alternatives from a set of alternatives L based on m criteria by calculating:

aggregated preference index

$$\pi(a_i, a_l) = \sum_{j=1}^{m} P_j(a_i, a_l) \cdot w_j \text{ }, \forall a_i, a_l \in L \tag{4}$$

positive (or outgoing) outranking flow

$$\Phi^{+}(a_i) = \frac{1}{n-1} \sum_{x \in L} \pi(a_i, x) \tag{5}$$

negative (or ingoing) outranking flow

$$\Phi^{-}(a_i) = \frac{1}{n-1} \sum_{x \in L} \pi(x, a_i) \tag{6}$$

net outranking flow

$$\Phi(a_i) = \Phi^{+}(a_i) - \Phi^{-}(a_i) \tag{7}$$

and then ranks the alternatives in order of decreasing values of Φ.

Due to the specificity of the analyzed problem (comparison of only two alternatives), the implementation of the Promethee II method is reduced to the calculation of the aggregated preference index from (4):

$$\pi(s^i_{j+1}, s^i_j) = \sum_{q=1}^{4} P_q(s^i_{j+1}, s^i_j) \cdot w_q \tag{8}$$

Namely, in the considered case for $n = 2$ and $L = A$, from (5) it follows

$$\Phi^{+}(s^i_{j+1}) = \pi(s^i_{j+1}, s^i_j)\,. \tag{9}$$

As $d_q(s^i_j, s^i_{j+1}) \leq 0$ due to the cumulative nature of the criteria, it follows that $P_q(s^i_j, s^i_{j+1}) = 0$ and $\pi(s^i_j, s^i_{j+1}) = 0$. From (6) and (7), it follows that

$$\Phi(s^i_{j+1}) = 0 \text{ and } \Phi(s^i_{j+1}) = \Phi^{+}(s^i_{j+1}) = \pi(s^i_{j+1}, s^i_j)\,. \tag{10}$$

The index π shows how much the alternative s^i_{j+1} is better than the alternative s^i_j.

As a result of multi-criteria analysis performed on the state matrices of all participants, the M^{400}_{π} matrix was obtained. This matrix has the following appearance:

$$M^{400}_{\pi} = \begin{bmatrix} \pi(s^1_2, s^1_1) & \pi(s^1_3, s^1_2) & \pi(s^1_4, s^1_3) & \pi(s^1_5, s^1_4) & \pi(s^1_6, s^1_5) \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0.5336 & 0.6169 & 0.5772 & 0.7662 & 0.6105 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \pi(s^{400}_2, s^{400}_1) & \pi(s^{400}_3, s^{400}_2) & \pi(s^{400}_4, s^{400}_3) & \pi(s^{400}_5, s^{400}_4) & \pi(s^{400}_6, s^{400}_5) \end{bmatrix} \tag{11}$$

Numerical values in the matrix correspond to the O_{83} course participant. As expected, due to the cumulative nature of state parameters, elements of matrix M^{400}_{π} are in the range $[0, 1]$.

Progress Functions. Progress functions are generated based on the M^{400}_{π} matrix. For each object O_i, a discrete cumulative progress function f^i_p is formed, which shows how that object progresses towards the set goal over time. The function is defined as follows:

$$\begin{aligned} & f^i_p(t_1) = 0 \\ & f^i_p(t_j) = \sum_{r=1}^{j-1} \pi(s^i_{r+1}, s^i_r) = f^i_p(t_{j-1}) + \pi(s^i_j, s^i_{j-1}), \quad 1 < j \leq 6 \end{aligned} \tag{12}$$

For illustration, the graph of the f^{83}_p function is shown in Fig. 3.

Classification Models Application. We employed MLP and KNN classificators to draw conclusions from the input values provided for the training set and to predict the grade marks on the test set. The students' grades ranged from 5 to 10 (from poor to excellent), requiring the use of a multi-class model. Data analysis and predictions were carried out using IBM SPSS Statistics 25 software (IBM, USA).

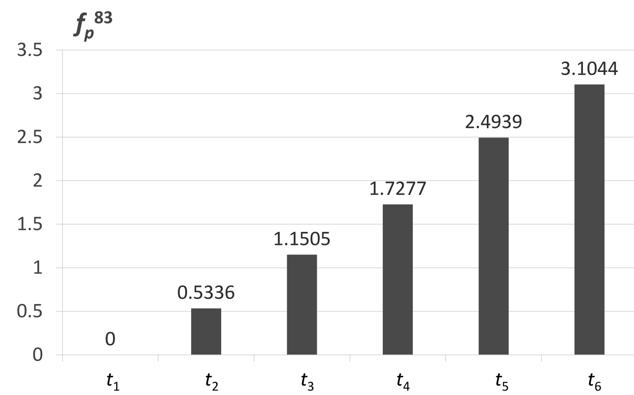

Fig. 3. O_{83} student progress function

Multi-layer Perceptron. The specific type of neural network called a Multi-layer Perceptron (MLP) was introduced by Frank Rosenblatt who is widely acknowledged as a pioneer in the training of neural networks, including multi-layer perceptron [38]. The MLP uses feedforward architecture and has one or more non-linear hidden layers between the input and output layers. A detailed explanation can be found in [35]. MLP can be used in predicting or classifying problems. It is a supervised method that means the results of prediction can be compared with the values of the target variables. MLP learns a function

$$f : R_m \to R_n \tag{13}$$

by training on a dataset, where m is the dimension of the input vector and n is the dimension for output vector. For a given set of m-dim input vectors and n-dim target variables, a nonlinear function for classification can be found.

In our investigation, the data set comprises student information from three consecutive academic years (2021/22 to 2023/24). We assigned cases to split the data into training (75%) and test (25%) sets. Before selecting hyperparameters, we conducted a series of classification and prediction tests that guided our decisions. We utilized predefined activation functions to evaluate the best options. Input variables were specified as covariates, and they are rescaled by default to improve network training. The automatic architecture of a network with a single hidden layer was selected with the best number of units in the hidden layer and the default activation functions for the hidden and output layers. We employed the hyperbolic tangent activation function in the hidden layer, while the output layer used the "Softmax" activation function.

Avoiding overfitting is essential in machine learning to ensure that a model generalizes well to unseen data. We used various techniques to achieve this. Our sufficiently large dataset captures the underlying patterns, and more data could enhance generalization. We split the dataset into training and testing sets, training the model on the former and evaluating it on the latter. The independent variable importance analysis was conducted to assess the significance of each predictor.

When learning neural network is completed, an MLP model is created that will be used to classify data. Classification is based on the values of progress function in milestones

$$f_p^i(t_j), \quad j = 2, 3, 4, 5, 6\,. \tag{14}$$

This used the input vector with variables T2 to T6 as features. Target variable was the student's grade mark ranged from 5 to 10. During training, we monitored the model's performance on the validation set. The algorithm used one consecutive step with no decrease in error as a stopping rule, a maximum of 15-minute training, and a maximum relative change in training error of 0.0001.

K-Nearest Neighbors. The K-Nearest Neighbors (KNN) model is widely used in machine learning for classification and regression. The initial concepts of the KNN model were introduced by [16] and further developed by [13]. KNN is a machine learning technique that can be used for supervised classification of cases based on their similarity to other cases. "Neighbors" are similar cases close to each other, while different cases are far from each other. The distance between the two cases is a measure of their difference and could be Euclidean or another distance. In this paper, Euclidean distance given in (15) is used to measure the similarity between neighbors:

$$d(x, y) = \sum_{i=0}^{N} \sqrt{(x_i^2 - y_i^2)} \tag{15}$$

The free parameter K is the number of nearest neighbors to be examined. The distance of the new case from each of the cases in the model is calculated, then the classifications of the most similar cases are added up and the new case is placed in the category that contains the largest number of nearest neighbors.

We implemented a KNN classification model, using variables T2 to T6 as features and the grade mark as the target variable. The value of the parameter K was selected based on the error rate indicated in the error log graph (Fig. 4), which demonstrated that the smallest error rate occurred when $K = 3$. Consequently, we chose $K = 3$ as the optimal parameter value for our KNN model.

Statistical Analysis and Measures for Multi-class Classification. Exploratory and data analyses were performed using the IBM SPSS Statistics 25 software (IBM, USA). The values of progress function in milestones (T2, T3, T4, T5, and T6) were used as input variables for the final grade prediction. There were 300 observations in the training group and 100 in the test group. Correlation analysis for paired samples was applied to test the correlation between the observed and predicted grade at the significance level $P < 0.05$.

Various metrics are available to assess the effectiveness of any multi-class classifier, making them valuable for comparing different models and analyzing a model's behavior when adjusting parameters [19]. These metrics are derived from the Confusion Matrix, providing essential information about algorithm and classification rule performance. We evaluated MLP and KNN methods for multi-class classification using a gold standard, measuring the level of agreement between observed and predicted grade marks.

Class Balance Accuracy (CBA) is a measure that aims to balance precision and recall for each input class [32]. Let is G the set of class labels with cardinality k, where $i, j =$

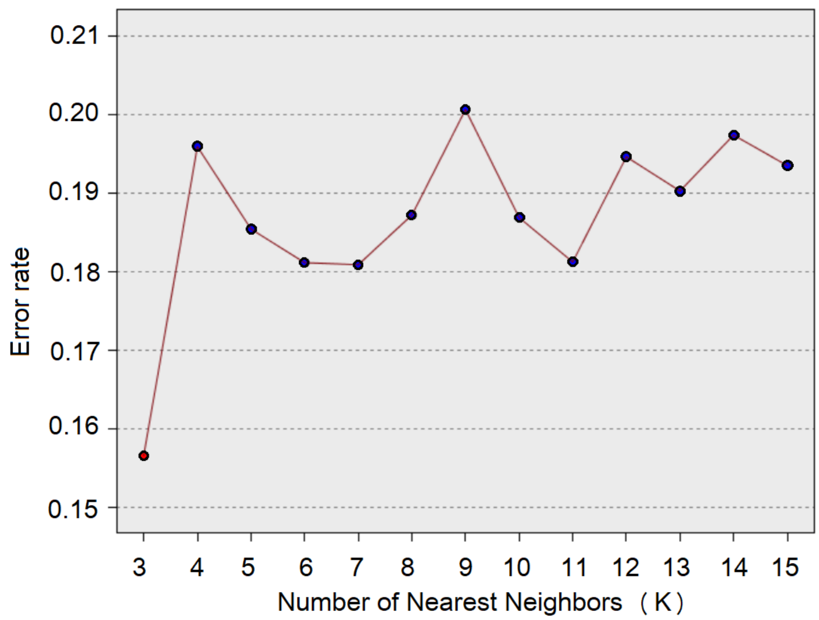

Fig. 4. K selection error log

$1, 2, \ldots, k$, and k is number of classes. The confusion matrix C^k is a k-dim square matrix or contingency table with elements c_{ij} representing the number of cases with true label i classified into group j. For confusion matrix C^k, $k > 2$, CBA is defined as

$$\mathrm{CBA} = \frac{\sum_{i=1}^{k} \frac{c_{ii}}{\max(t_i, p_i)}}{k}, \tag{16}$$

where

$$t_j = \sum_{i=1}^{k} c_{ij} \tag{17}$$

$$p_i = \sum_{j=1}^{k} c_{ij} \tag{18}$$

$$c = \sum_{i=1}^{k} c_{ii} \tag{19}$$

$$s = \sum_{i=1}^{k} \sum_{j=1}^{k} c_{ij} \tag{20}$$

t_j is the number of times class j truly occurred (column total),
p_i is the number of times class i was predicted (row total),

c is the total number of samples correctly predicted, and
s the total number of samples.

Brian W. Matthews introduced the Matthews Correlation Coefficient (MCC) [29, 28] that became widely used as a measure to evaluate the performance of Machine Learning techniques, with some adaptations for multi-class scenarios [11]. The MCC has a range of $[-1, 1]$. Values close to 1 indicate very accurate predictions, suggesting a strong positive correlation between the predicted values closely matching the actual classification. An MCC of 0 suggests no correlation between the variables, indicating that the classifier is randomly assigning units to classes without any connection to their true values [19]. MCC can also be negative, indicating an inverse relationship between true and predicted classes. While this is undesirable, it often occurs due to modeling errors. A strong inverse correlation suggests that the model has learned how to classify the data but consistently switches all the labels.

In the multi-class case, the MCC can be defined in terms of a confusion matrix C^k for k classes and according to (17)–(20):

$$\text{MCC} = \frac{c \cdot s - \sum_{j=1}^{k} p_j \cdot t_j}{\sqrt{\left(s^2 - \sum_{j=1}^{k} p_j^2\right)\left(s^2 - \sum_{j=1}^{k} t_j^2\right)}} \tag{21}$$

In the multi-class case, the calculation of Cohen's Kappa (Kappa) is similar to Matthews Correlation Coefficient [17]. Referring to multi-class confusion matrix C^k and according to (17)–(20):

$$\text{Kappa} = \frac{c \cdot s - \sum_{j=1}^{k} p_j \cdot t_j}{s^2 - \sum_{j=1}^{k} p_j \cdot t_j} \tag{22}$$

Kappa allows the comparison of two models with the same accuracy but different Cohen's Kappa values. The Kappa and MCC statistics both range from -1 to $+1$, and their interpretation is as follows:

$[0.00, 0.09]$ agreement equivalent to chance,
$[0.10, 0.20]$ slight agreement,
$[0.21, 0.40]$ fair agreement,
$[0.41, 0.60]$ moderate agreement,
$[0.61, 0.80]$ substantial agreement,
$[0.81, 0.99]$ near perfect agreement, and
1.00 perfect agreement.

Negative values can be understood in the context of the MCC statistic.

3. Results

The classification was based on the progress function values $f_p^i(t_j)$, $j = 2, 3, 4, 5, 6$, at milestones t_2, t_3, t_4, t_5, and t_6, denoted as T2, T3, T4, T5, and T6 input variables. The accuracy of MLP and KNN classification was validated using a confusion matrix and statistical measures.

The MLP classification results are given in Table 2.

Table 2. MLP classification

MLP	Training	Predicted grade						Correct	Test	Predicted grade						Correct
Input var.	Observed	5	6	7	8	9	10		Observed	5	6	7	8	9	10	
T2, T3	5	45	10	2	1	0	0	77.60%	5	11	1	0	0	0	0	91.70%
	6	25	21	14	2	0	0	33.90%	6	8	9	6	0	0	0	39.10%
	7	0	8	32	14	0	0	53.30%	7	0	4	13	9	0	0	50.00%
	8	0	1	12	37	3	2	67.30%	8	0	0	3	14	0	0	82.40%
	9	0	0	1	8	20	7	55.60%	9	0	0	0	6	6	0	50.00%
	10	0	0	0	0	3	26	89.70%	10	0	0	0	0	1	9	90.00%
	Overall %	25.30%	13.30%	20.30%	20.70%	8.70%	11.70%	60.30%	Overall %	19.00%	14.00%	22.00%	29.00%	7.00%	9.00%	62.00%
T2, T3, T4	5	45	13	0	0	0	0	77.60%	5	11	1	0	0	0	0	91.70%
	6	10	44	8	0	0	0	71.00%	6	4	16	3	0	0	0	69.60%
	7	0	11	38	11	0	0	63.30%	7	0	3	16	7	0	0	61.50%
	8	0	0	8	37	10	0	67.30%	8	0	0	1	15	1	0	88.20%
	9	0	0	0	8	24	4	66.70%	9	0	0	0	3	9	0	75.00%
	10	0	0	0	0	5	24	82.80%	10	0	0	0	0	1	9	90.00%
	Overall %	18.30%	22.70%	18.00%	18.70%	13.00%	9.30%	70.70%	Overall %	15.00%	20.00%	20.00%	25.00%	11.00%	9.00%	76.00%
T2,T3, T4, T5	5	54	4	0	0	0	0	91.30%	5	11	1	0	0	0	0	91.70%
	6	10	48	4	0	0	0	77.40%	6	1	22	0	0	0	0	95.70%
	7	0	4	51	5	0	0	85.00%	7	0	1	21	4	0	0	80.80%
	8	0	0	4	48	3	0	87.30%	8	0	0	2	13	2	0	76.50%
	9	0	0	0	2	31	3	86.10%	9	0	0	0	2	10	0	83.30%
	10	0	0	0	0	2	27	93.10%	10	0	0	0	0	1	9	90.00%
	Overall %	21.30%	18.70%	19.70%	18.30%	12.00%	10.00%	86.30%	Overall %	12.00%	24.00%	23.00%	19.00%	13.00%	9.00%	86.00%
T2,T3, T4, T5,T6	5	58	0	0	0	0	0	100.00%	5	12	0	0	0	0	0	100.00%
	6	0	62	0	0	0	0	100.00%	6	0	23	0	0	0	0	100.00%
	7	0	1	59	0	0	0	98.30%	7	0	0	26	0	0	0	100.00%
	8	0	0	0	55	0	0	100.00%	8	0	0	0	17	0	0	100.00%
	9	0	0	0	0	36	0	100.00%	9	0	0	0	0	12	0	100.00%
	10	0	0	0	0	0	29	100.00%	10	0	0	0	0	0	10	100.00%
	Overall %	19.30%	21.00%	19.70%	18.30%	12.00%	9.70%	99.70%	Overall %	12.00%	23.00%	26.00%	17.00%	12.00%	10.00%	100.00%

These results indicate that the accuracy achieved using the input variables T5 and T6 was between 93% and 94.3% (not shown). There was a small variation in accuracy when using the independent variables T4-T6 and T3–T6, with accuracies ranging from 92% to 94.3%. In all three cases, T6 was found to be the most important independent variable in the MLP classification. When using independent variables T2–T5, the accuracy was between 86.0% and 86.3%. However, when using T3–T5 variables, the accuracy slightly decreased to 84%–84.3%, with T5 being determined as the most important independent variable in the MLP classification (Table 2). Using T2–T4 or T3–T4 as independent variables resulted in a further decrease in classification accuracy to 70.3%–78.0%, with T4 being identified as the most important input variable. The highest correct classification rate for the training set was 99.7% when all five input variables were used, while the test set had a predicted grade of 100% (Table 2). Fig. 5A shows a chart of run example of one-hidden-layer MLP with three variables in the input layer and the student's grade in the output layer. The corresponding Receiver-operating characteristic (ROC) curves and values of the Area Under the Curve (AUC) are plotted for six classes (Fig. 5B). The dependent variable has six categories, so each curve treats the category at issue as the positive state versus the aggregate of all other categories. It can be seen that the best results are achieved for grades 5 and 10: AUC was approximately 1 which represents an ideal measure of separability. However, AUC values are also large for the other classes (0.918-0.963).

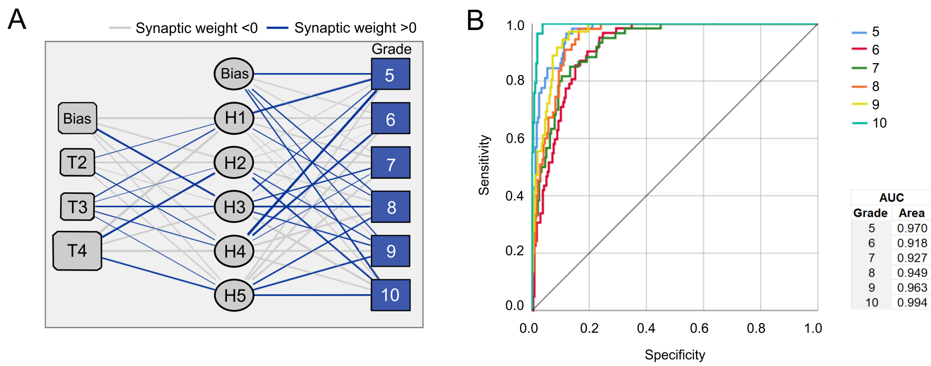

Fig. 5. A. MLP classifier: Classification is based on the values of progress function in milestones: T2, T3, and T4. A hidden layer activation function was hyperbolic tangent, while the output layer activation function was "Softmax". The target variable/class was the student's grade marks ranging from 5 (drop down) to 10 (excellent). B. ROC curves and corresponding AUC values

The results of the KNN classification (Table 3) indicated poorer performance compared to the MLP classifier. The highest accuracy was attained using the input variables T2–T6 (93.0%–94.0%). There was a slight variation in accuracy for the input variables T4–T6 (92.3%–94%), T5–T6 (91%–92%), and T3–T6 (90.7%–93%) (not displayed). When the independent variables were T2–T5, the accuracy ranged from 78.0%–79.0%, while for T3–T5, it ranged from 77.0%–83.0%.

It has been observed that in both MLP and KNN classifiers, the variable T6 has a significant impact on classification accuracy. However, while T6 is important for grading, it is not sufficient for predicting grades and guiding students towards their desired grade in a timely manner. Although the accuracy is slightly lower when T6 is not used as an input variable in classification, the results of the classification based on T2–T5 can be very influential in predicting student success in the final grade. When attempting to predict a student's success on the exam (at moment t_6) using input variables T2–T5, better results are obtained by using the MLP (86.0%) compared to the KNN classifier (79.0%). Although the difference of 7% accuracy is not negligible, the KNN classification offers the advantage of comparing the achievements of an individual student with those of their three to five nearest neighbors, allowing us to form groups of students with similar achievements and monitor their progress over time (Fig. 6B). Fig. 6 depicts how a focal case, student #83, would be classified using the three nearest neighbors ($K = 3$). KNN showed lower accuracy in general, except in cases of T2–T4 and T3–T4 input variables. This suggests a potential advantage of the MLP classifier.

Table 3. KNN classification

KNN	Training	Predicted grade						Correct	Test	Predicted grade						Correct
Input var.	Observed	5	6	7	8	9	10		Observed	5	6	7	8	9	10	
T2, T3	5	32	22	3	1	0	0	55.20%	5	8	4	0	0	0	0	66.70%
	6	15	33	11	3	0	0	53.20%	6	4	14	5	0	0	0	60.90%
	7	6	18	23	14	0	0	38.30%	7	0	4	14	8	0	0	53.80%
	8	0	2	19	24	6	4	43.60%	8	0	1	6	10	0	0	58.80%
	9	0	0	2	10	16	8	43.60%	9	0	0	2	3	7	0	58.30%
	10	0	0	0	0	9	20	69.00%	10	0	0	0	1	2	7	70.00%
	Overall %	17.70%	25.00%	19.30%	17.00%	10.30%	10.70%	49.30%	Overall %	12.00%	23.00%	27.00%	22.00%	9.00%	7.00%	60.00%
T2, T3, T4	5	46	12	0	0	0	0	79.30%	5	11	1	0	0	0	0	91.70%
	6	12	41	9	0	0	0	66.10%	6	3	18	2	0	0	0	78.30%
	7	1	16	29	14	0	0	48.30%	7	0	4	14	8	0	0	53.80%
	8	0	0	9	39	7	0	70.90%	8	0	0	1	16	0	0	94.10%
	9	0	0	0	8	22	6	61.60%	9	0	0	0	3	9	0	75.00%
	10	0	0	0	0	5	24	82.80%	10	0	0	0	0	0	10	100.00%
	Overall %	19.70%	23.00%	15.70%	20.30%	11.30%	10.00%	67.00%	Overall %	14.00%	23.00%	17.00%	27.00%	9.00%	10.00%	78.00%
T2,T3, T4, T5	5	47	11	0	0	0	0	81.00%	5	11	1	0	0	0	0	91.70%
	6	8	50	4	0	0	0	80.60%	6	2	20	1	0	0	0	87.00%
	7	0	9	43	8	0	0	71.70%	7	0	1	17	8	0	0	65.40%
	8	0	0	4	45	6	0	81.80%	8	0	0	3	14	0	0	82.40%
	9	0	0	0	7	23	6	63.90%	9	0	0	0	4	7	1	58.30%
	10	0	0	0	0	3	26	89.70%	10	0	0	0	0	0	10	100.00%
	Overall %	18.30%	23.30%	17.00%	20.00%	10.70%	10.70%	78.00%	Overall %	13.00%	22.00%	21.00%	26.00%	7.00%	11.00%	79.00%
T2, T3, T4, T5, T6	5	55	3	0	0	0	0	94.80%	5	12	0	0	0	0	0	100.00%
	6	4	56	2	0	0	0	90.30%	6	0	21	2	0	0	0	91.30%
	7	0	4	55	1	0	0	91.70%	7	0	0	24	2	0	0	92.30%
	8	0	0	1	54	0	0	98.20%	8	0	0	1	16	0	0	94.10%
	9	0	0	0	4	31	1	86.10%	9	0	0	0	1	11	0	91.70%
	10	0	0	0	0	1	28	96.60%	10	0	0	0	0	0	10	100.00%
	Overall %	19.30%	21.00%	19.30%	19.70%	10.70%	9.70%	93.00%	Overall %	12.00%	21.00%	27.00%	19.00%	11.00%	10.00%	94.00%

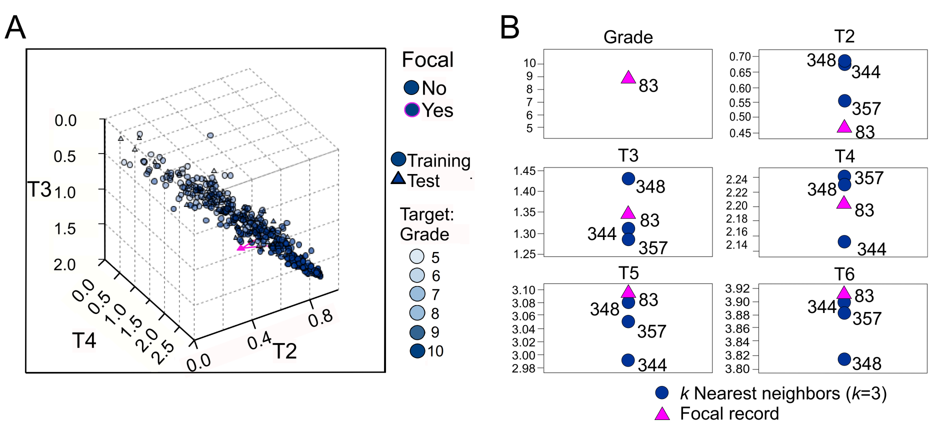

Fig. 6. KNN classification is based on the input values: T2, T3, T4, T5, and T6 with the target variable the student's grade mark. A. Three-dimensional projection of the five-dimensional predictor space representing training and test sets. The pink triangle represents the focal case (student #83) connected to its three nearest neighbors (pink lines); $K = 3$. The classification accuracy was from 93% to 94%; B. Peers chart of the focal record and its three nearest neighbors. The left upper panel shows the final grade (9) of student #83. Panels present the progress function values of the focal case and three nearest neighbors in milestones

Statistical parameters are shown in Tables 4 and 5.

Table 4. Measures for multi-class classification by MLP

Input variables	Set	Pearson's Correlation Coefficient	Cohen's Kappa statistics of agreement	CBA - Class Balance Accuracy	Matthews Correlation Coefficient	Classification quality (agreement)
T2, T3	Training	0.894	0.523	0.567	0.526	Fair
	Test	0.915	0.537	0.559	0.548	
T2, T3, T4	Training	0.941	**0.639**	**0.693**	**0.644**	Substantial
	Test	0.946	**0.707**	**0.716**	**0.712**	
T2, T3, T4, T5	Training	0.973	**0.834**	**0.850**	**0.834**	Near perfect
	Test	0.968	**0.828**	**0.832**	**0.829**	
T2, T3, T4, T5, T6	Training	0.999	**0.996**	**0.995**	**0.996**	Perfect
	Test	1.000	**1.000**	**1.000**	**1.000**	
T3, T4	Training	0.940	**0.640**	**0.689**	**0.640**	Substantial
	Test	0.950	**0.730**	**0.765**	**0.731**	
T3, T4, T5	Training	0.969	**0.810**	**0.834**	**0.810**	Near perfect
	Test	0.963	**0.803**	**0.840**	**0.803**	

Table 5. Measures for multi-class classification by KNN

Input variables	Set	Pearson's Correlation Coefficient	Cohen's Kappa statistics of agreement	CBA Class Balance Accuracy	Matthews Correlation Coefficient	Classification quality (agreement)
T2, T3	Training	0.853	0.299	0.479	0.299	Slight to fair
	Test	0.878	0.505	0.589	0.505	
T2, T3, T4	Training	0.933	0.447	**0.651**	0.447	Moderate to substantial
	Test	0.951	**0.732**	**0.742**	**0.732**	
T2,T3, T4, T5	Training	0.956	0.539	**0.740**	0.539	Moderate to substantial
	Test	0.952	**0.740**	**0.733**	**0.740**	
T2, T3, T4, T5, T6	Training	0.986	**0.915**	**0.913**	**0.915**	Near perfect
	Test	0.986	**0.926**	**0.927**	**0.926**	
T3, T4	Training	0.931	0.519	0.607	0.519	Moderate to substantial
	Test	0.951	**0.731**	**0.757**	**0.731**	
T3, T4, T5	Training	0.953	**0.720**	**0.734**	**0.720**	Substantial
	Test	0.961	**0.790**	**0.824**	**0.790**	

Pearson's Correlation Coefficients showed highly and significantly correlated predicted and observed grades using MLP and KNN methods. Most metric values for multi-class classification performed using MLP were higher than those obtained with KNN.

4. Discussion and Conclusion

Our research primarily focuses on classifying student achievement and its implications on teaching strategies and the learning environment. We emphasize student-centered support and data-driven strategies to provide personalized feedback to learners. Analyzing student groups helps identify deviations and signals the need to adapt teaching methods for better outcomes.

Based on the classification in [34], we have outlined some features of our research study. We collected performance data from blended learning environments with study focuses on STEM fields. We monitored the academic outcomes of 400 students, which is a common scale for this type of study. We utilized formative and summative assessments, employing predictive modeling and direct methods similar to the approaches used in studies that utilize statistical models and neural networks. This research discusses the classification of students' achievement, specifically predicting their final grades. Our initial assumption implies that all students start learning the subject without prior knowledge. This approach can be adjusted by dividing the class into smaller groups, where an entrance test can be used to assess prior knowledge and incorporate the test result as the value of the progress function at time t_1.

The teaching practice should be modified following the final success prediction of every single student or group. After each time point and prediction of the final grade, students can receive work instructions through direct conversation, which will help them master the course tasks more effectively. It is essential to identify the segment with the weakest results that requires improvement.

For example, let us consider student #83. His progress function value was the lowest among his three closest peers at the time moment t_2 (Fig. 6B). Analyzing his data reveals high performance in class attendance, engagement, and homework completion but poor tests performance. The student was advised to focus on improving this area, with the opportunity for individual consultations with the professor to aid his progress. Following this guidance, student #83 demonstrated better results in knowledge checks and success predictions at times t_3 and t_4 compared to his three closest neighbors. Further recommendations for student #83 were provided based on the prediction results at time t_4. The Fig. 6B indicates that the student embraced the teacher's recommendations, leading to his transformation from the lowest to the highest performer in his group at the control check time t_5 and final exam.

If the majority of the group shows poorer results at milestones t_2 or t_3, it is essential to adjust the teaching approach for the entire group. First, it should be analyzed which specific areas are yielding poor results and tailor the teaching strategies accordingly. For instance, if class attendance is low, the reasons behind this issue should be investigated. This is particularly important when employing a combination of in-person and online teaching methods.

If it becomes evident that students attending in-person classes perform better on tests, efforts should be made to encourage students to attend live classes. Additionally, the online learning experience should be enhanced by addressing the needs and preferences of the students.

If the overall engagement of the group is lacking, it could be considered increasing interaction during lessons to stimulate student participation and awarding for successfully completed homework. If a significant number of students are struggling with their test results, additional consultative sessions for extra support should be organized.

When using MLP and KNN, there was no difference of more than one grade between the observed and predicted grades, except for T2–T3, which means the error will not exceed one in the grade scale. The prediction accuracy for the input variables T2–T3 is fair because it is early to predict the outcome, given the timeline. For example, some students may perform well at the beginning but later their performance declines. Conversely, some students may not perform well initially but later show improvement. When the values of the progress function at subsequent time points are included, the accuracy of grading and predicting the outcome of the exam increases. Upon comparison, it is evident from Tables 4 and 5 that KNN results in one level lower classification quality than MLP. MLP performed better in prediction, while KNN facilitated the formation of smaller groups for comparative monitoring, despite being less accurate in one level of prediction.

The advantages and weakness of the proposed approach are as follows:

– By including the Promethee II method, the process of monitoring students' progress offers a high level of flexibility. Lecturers can customize various parameters, including course activities, activity priorities, types of preference functions for each activity, and thresholds for these preference functions.
– The data used for predictions is of high quality. It not only reflects students' success but also incorporates the lecturer's attitudes, which can significantly influence outcomes based on their expertise in the subject.
– Implementing the solution is straightforward because the Promethee II method only requires comparisons between two consecutive states, simplifying the overall approach.
– Progress functions can serve as input data for different classifiers.
– The solution is broadly applicable, as it can be adapted to various subjects.
– The proposed method's weakness is its lower accuracy during the initial phase of predicting the final student success.

4.1. Comparison with Existing Approaches

In a review paper [27] that covered publications from 2007 to 2016, it was found that Decision Tree, Rule-based, and Naive Bayes techniques were used in the majority of works to predict students' academic performance. Neural Network and KNN algorithms were used in fewer studies, which was an additional reason to apply these methods. The input data included academic and socioeconomic predictors of success. The meta-analysis determined that the average accuracy is higher when using KNN (87%) compared to using Neural Network (78.7%). Our results showed the opposite, MLP had better performance than KNN.

The authors in [1] used five different machine learning techniques (Naive Bayes, KNN, SVM, XG-boost, and MLP) to predict individual student results. MLP achieved the highest accuracy of 86.25% as in our study when using T2–T5 variables, while other classifiers achieved around 80% accuracy. In [40], they employed seven different classifiers (SVM, KNN, Logistic Regression, Decision Tree, AdaBoost, MLP, and Extra Tree Classifier) to classify students' final grades and they achieved a final accuracy of 81.73%.

Researchers developed a model to analyze IT students' academic performance using data mining techniques: WEKA software and a J48 decision tree to classify success grades [23]. Kappa statistics for this prediction ranged from 0.9070 to 0.9582, but other measures for multi-categorical classification were not assessed. According to Kappa statistics this classification was near perfect. Also in [8], data mining classification techniques such as J48 Decision Tree, KNN, and MLP were successfully used with the WEKA tool to identify patterns between students' initial grades upon entering the university and their grades at graduation.

The algorithm FlexNSLVOrd, developed by [20], predicts student performance in distance courses by analyzing their online interactions with e-learning platforms using fuzzy systems and ordinal classification. Despite being slower than other algorithms, it has shown better performance in studies.

The study [36] examined dropout rates in online learning environments using lasso and ridge logistic regression. They developed a predictive model based on data from 32,593 students across 22 courses, analyzing 173,912 assessment records. This study focused on early dropout rate predictions at intervals of 30, 45, 60, 90, and 120 days within a course lasting approximately 240 days. They found that the model's AUC improved from 0.549 and 0.661 in the early phase to 0.681 and 0.869 by mid-term. Initially, student demographics and course characteristics were significant predictors, but as the course progressed, student activity became more important. The primary difference between this study and ours is that the former analyzed data from multiple students across several courses over approximately 240 days. In contrast, our research focused on predictions made at five points during a shorter, intensive course. Despite methodological differences, both studies conclude that early predictions are significantly less accurate than those made later.

The article [25] presented a predictive model aimed at identifying students at risk of dropping out during the early stages of their university studies. The researchers analyzed data from 30,576 students enrolled in Higher Education Institutions between 2000 and 2020. They examined the significance of various factors related to dropouts, categorizing them by faculty, degree program, and semester across different predictive models. The findings indicate that variables such as Grade Point Average (GPA), socioeconomic factors, and course pass rates significantly influence the model, regardless of the semester, faculty, or program. Additionally, the study revealed a noteworthy difference in predictive power between Science, Technology, Engineering, and Mathematics (STEM) programs and humanistic programs. Our research was focusing specifically on STEM program.

A critical aspect of the model's accuracy lies in the predictor variables chosen for analysis. Typically, these variables are drawn from students' academic records. One widely used variable is the GPA, often analyzed alongside other grades. For example, the research [26] included both overall GPA and term GPA to predict student dropout rates, while also considering factors such as gender, ethnicity, enrollment status (full-time or part-time), academic classification (freshman or sophomore), and age. Their findings indicated that, although GPA is significantly associated with dropout rates, other variables can also yield strong predictive results.

Additionally, [3] investigated various features influencing student dropouts, including demographic information, family background (such as parents' educational levels), pre-enrollment attributes (like high school GPA and admission test scores), financial circumstances, enrollment details, and academic performance metrics. They specifically analyzed students' GPA, the percentage of credits passed, dropped, and failed, as well as the total credit hours attempted. Their findings indicated that the most significant variables affecting dropout rates were high school GPA, overall GPA, the percentage of failed credits, and various financial factors. In our prediction model, we have used only academic results in accordance with predefined preference functions, without other pre-enrollment attributes.

4.2. Future Work

The practical application of predictive models relies heavily on their accuracy and reliability. Since these models are based on statistical techniques, uncertainties are always a factor. Therefore, it is important to assess the robustness of the model, particularly regarding confidence in its recommendations, which can be enhanced through sensitivity analysis. Our future research will focus on this issue.

The goals of sensitivity analysis are to determine how the output of a system changes when input parameters are modified, as well as to identify which parameters are the most significant predictors. Numerous studies across various fields address sensitivity analysis for different classifiers, including KNN and MLP. However, our system is heterogeneous, comprising a component that prepares data for prediction and another that contains the classifiers, which adds complexity to the problem.

The input data set is diverse and extensive, making it challenging to identify the input variables for sensitivity analysis. Thus far, we have conducted some research on sensitivity related to preference functions, revealing it to be a very complex issue. Consequently, we may also explore the possibility of conducting a sensitivity analysis specifically for classifiers, using the values of the progress function as input parameters.

Future research could incorporate additional features like pre-enrollment attributes and an entrance test to enhance predictions of student progress.

Acknowledgments. Slađana Spasić was supported by the Ministry of Science, Technological Development and Innovation of Republic Serbia [contract number 451-03-136/2025-03/ 200053]. The authors would like to express their gratitude to Dejan Živković for his assistance in preparing the manuscript for publication.

References

1. Ahammad, K., Chakraborty, P., Akter, E., Fomey, U., Rahman, S.: A comparative study of different machine learning techniques to predict the result of an individual student using previous performances. International Journal of Computer Science and Information Security 19, 5–10 (2021)
2. Alvarado-Uribe, J., Mejía, P., A., Masetto, A., H., Molontay, R., Hilliger, I., Hegde, V., Gallegos, J., Díaz, Ceballos, R., H.: Student dataset from tecnologico de monterrey in mexico to predict dropout in higher education. Data 7, 119 (2022)
3. Ameri, S., Fard, M.J., Chinnam, R.B., Reddy, C.K.: Survival analysis based framework for early prediction of student dropouts. In: Proceedings of the 25th ACM International Conference on Information and Knowledge Management. pp. 903–912 (2016)
4. Aulck, L., Velagapudi, N., Blumenstock, J., West, J.: Predicting student dropout in higher education. arXiv preprint arXiv:1606.06364 (2016)
5. Banamar, I., Smet, D., Y.: An extension of promethee ii to temporal evaluations. International Journal of Multicriteria Decision Making 7, No. 3/4, 298–325 (2018)
6. Baradwaj, B., Pal, S.: Mining educational data to analyze students' performance. International Journal of Advanced Computer Science and Applications 2, 63–69 (2011)

7. Barbosa Manhaes, L.M., da Cruz, S.M.S., Zimbrao, G.: Towards automatic prediction of student performance in stem undergraduate degree programs. In: Proceedings of the 30th Annual ACM Symposium on Applied Computing. pp. 247–253 (2015)
8. Bawah, U., F., Ussiph, N.: Appraisal of the classification technique in data mining of student performance using j48 decision tree, k-nearest neighbor and multilayer perceptron algorithms. International Journal of Computer Applications 179, 39–46 (2018)
9. Behr, A., Giese, M., Teguim, H., Theune, K.: Motives for dropping out from higher education - an analysis of bachelor's degree students in germany. European Journal of Education 56 (2021)
10. Brans, J.P., Mareschal, B., Figueira, J., Greco, S., Ehrogott, M.: Promethee methods. In: Greco, S., Ehrgott, M., Figueira, J.R. (eds.) Multiple Criteria Decision Analysis: State to the Art Surveys, chap. 5, pp. 163–195. Springer, New York (2005)
11. Chicco, D., Jurman, G.: The advantages of the matthews correlation coefficient (mcc) over f1score and accuracy in binary classification evaluation. BMC Genomics 21, No. 6 (2020)
12. Chung, J.Y., Lee, S.: Dropout early warning systems for high school students using machine learning. Children and Youth Services Review 96, 346–353 (2019)
13. Cover, T.M., Hart, P.E.: Nearest neighbor pattern classification. IEEE Transactions on Information Theory 13, No. 1, 21–27 (1967)
14. Daniel, B.: Big data and analytics in higher education: Opportunities and challenges. British Journal of Educational Technology 46, No. 5 (2014)
15. Dixson, D., Worrell, F.: Formative and summative assessment in the classroom. Theory Into Practice 55, 14 (2016)
16. Fix, E., Hodges, J.L.: Discriminatory analysis, nonparametric discrimination: Consistency properties. Tech. Rep. Technical Report 4, USAF School of Aviation Medicine, Randolph Field (1951)
17. Fleiss, J., Cohen, J., Everitt, S., B.: Large sample standard errors of kappa and weighted kappa. Psychological Bulletin 72, 323–327 (1969)
18. Gasevic, D., Dawson, S., Rogers, T., Gasevic, D.: Learning analytics should not promote one size fits all: The effects of instructional conditions in predicting academic success. The Internet and Higher Education 28, No. 1. 68–84 (2016)
19. Grandini, M., Bagli, E., Visani, G.: Metrics for multi-class classification: an overview. arXiv preprint abs/2008.05756 (2020)
20. Gámez-Granados, J.C., Esteban, A., Rodriguez-Lozano, J., F., Zafra, A.: An algorithm based on fuzzy ordinal classification to predict students' academic performance. Applied Intelligence 53, 27537–27559 (2023)
21. Hellas, A., Liao, S., Ihantola, P., Petersen, A., Ajanovski, V., Gutica, M., Hynninen, T., Knutas, A., Leinonen, J., Messom, C.: Predicting academic performance: a systematic literature review. In: Proceedings of the Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education. pp. 175–199. Larnaca, Cyprus (2018)
22. Huang, S., Fang, N.: Predicting student academic performance in an engineering dynamics course: A comparison of four types of predictive mathematical models. Computers & Education 61, 133–145 (2013)
23. Ibrahim, W., Abdullaev, S., Alkattan, H., Oluwaseun, A., Alkattan, H., Alhumaima, A.: Development of a model using data mining technique to test, predict and obtain knowledge from the academics results of information technology students. Data 7, No. 5, 18 (2022)
24. Jesus, C., F., Castelli, M., Oliveira, T., Mendes, R., Nunes, C., Sa-Velho, M., Rosa-Louro, A.: Using artificial intelligence methods to assess academic achievement in public high schools of a european union country. Heliyon 6 (2020)
25. Jimenez-Macias, A., Moreno-Marcos, M., Merino, P., Ortiz, M., Delgado-Kloos, C.: Analyzing feature importance for a predictive undergraduate student dropout model. Computer Science and Information Systems 20, 50–50 (2022)
26. Kang, K., Wang, S.: Analyze and predict student dropout from online programs. In: Proceedings of the 2nd International Conference on Compute and Data Analysis. pp. 6–12 (2018)

27. Kumar, M., Singh, A., Handa, D.: Literature survey on student's performance prediction in education using data mining techniques. International Journal of Education and Management Engineering 6, 40–49 (2017)
28. Matthews, W., B.: Solvent content of protein crystals. Journal of Molecular Biology 33, No. 2, 491–497 (1968)
29. Matthews, W., B.: Comparison of the predicted and observed secondary structure of t4 phage lysozyme. Biochimica et Biophysica Acta (BBA) - Protein Structure 405, No. 2, 442–451 (1975)
30. Maura, E., A., P., Nazeeruddin, E., Nazeeruddin, M., Daqqa, I., Abdelsalam, H., Abdullah, M.: Is initial performance in a course informative? machine learning algorithms as aids for the early detection of at-risk students. Electronics 11, No. 13, 2057 (2022)
31. Moreno-Marcos, P.M., Laet, D., T., Munoz-Merino, P.J., Soom, V., C., Broos, T., Verbert, K., Kloos, D., C.: Generalizing predictive models of admission test success based on online interactions. Sustainability 11, No. 18, 4940 (2019)
32. Mosley, L.: A balanced approach to the multi-class imbalance problem. Phd dissertation, Iowa State University, USA (2013)
33. Mthimunye, K., Daniels, of academic performance, F.P., success andretention amongst undergraduate nursing students: A systematic review. South African Journal of Higher Education 33, 200–220 (2019)
34. Namoun, A., Alshanqiti, A.: Predicting student performance using data mining and learning analytics techniques: A systematic literature review. Applied Sciences 11, 1–28 (2020)
35. Popescu, M.C., Balas, V., Perescu-Popescu, L., Mastorakis, N.: Multilayer perceptron and neural networks. WSEAS Transactions on Circuits and Systems 8, No. 7 (2009)
36. Radovanović, S., Delibašić, B., Suknović, M.: Predicting dropout in online learning environments. Computer Science and Information Systems 18, No. 3, 957–978 (2021)
37. Rastrollo-Guerrero, J., Gomez-Pulido, A., J., Domínguez, A.: Analyzing and predicting students' performance by means of machine learning: A review. Applied Sciences 10, 1042 (2020)
38. Rosenblatt, F.: Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms. Spartan Books, Washington DC, 1st edn. (1961)
39. Zhang, L., Li, K.F.: Education analytics: Challenges and approaches. In: Proceedings of the 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA). pp. 193–198. IEEE, Krakow, Poland (2018)
40. Zulfiker, S., Kabir, N., Biswas, A., A., Chakraborty, P., Rahman, M.: Predicting students' performance of the private universities of bangladesh using machine learning approaches. International Journal of Advanced Computer Science and Applications 11, No. 3, 672–679 (2020)

Slađana Spasić completed her B.Sc. in Numerical Mathematics and Cybernetics in 1991, followed by a M.Sc. in Artificial Intelligence in 2003, and a Ph.D. in Applied Mathematics in 2007, all from the University of Belgrade, School of Mathematics, Serbia. From 2008 to 2013, she served as an assistant professor at the University of Belgrade - Institute for Multidisciplinary Research (IMSI) and Singidunum University. She was promoted to associate professor from 2013 to 2018, and she has been a full professor at both institutions since 2018. Starting in 2024, she will join IMSI. Her research interests include the structural aspects of biological signals and images, as well as the development of statistical and mathematical models in ecology, physiology, and other related fields. She published over 50 scientific papers on SCI list and 2 textbooks.

Violeta Tomašević received her B.Sc., M.Sc., and PhD degrees from the School of Electrical Engineering, University of Belgrade, Serbia, in 1988, 1994, and 2005, respectively.

From 1989 to 2007, she was with the "Mihajlo Pupin" Institute, Belgrade. In 2007, she joined the University Singidunum, Belgrade, where she is currently a Full Professor with the Informatics and Computing Department. Her research interests include cryptanalysis, expert systems, decision support systems, data retrieval and web programming. She published over 60 scientific papers and 3 textbooks.

Received: October 22, 2024; Accepted: April 10, 2025.