Interpretable Feature Engineering from Moodle Logs for Student Performance Prediction Using Focused Learning Windows and Environmental Context

Dino Vlahek *, Marija Pokos Lukinec, and Dijana Oreški

Faculty of Organization and Informatics, University of Zagreb, Varaždin, Croatia
Email: dvlahek@foi.hr (D.V.); mapokos@foi.hr (M.P.L.); dijana.oreski@foi.hr (D.O.)
*Corresponding author
Manuscript received May 11, 2026; revised July 9, 2026; accepted August 20, 2026; published November 24, 2026

***Abstract*—This study presents a reproducible feature-engineering framework for predicting student performance from Moodle learning management system logs. The framework estimates the beginning of each student's focused learning period using a rolling-window procedure and derives behavioral, temporal, session-based, assignment, quiz, environmental, and interaction features. Weather and air-quality variables are aligned with the estimated preparation interval to examine whether external context provides incremental predictive information beyond Learning Management System (LMS) behavior alone. The framework was evaluated on 342 students across two academic years using raw-score regression and pass/fail classification. All data-dependent pruning was fitted exclusively on the training portion of each validation fold. With the primary five-day focused-learning window, the best Random Forest regression model reached Coefficient of Determination $R^2 = 0.149$ and Root Mean Square Error (RMSE) = 3.327, while pass/fail Receiver Operating Characteristic curve-Area under the receiver operating Characteristic curve (ROC-AUC) reached 0.684. The numerical improvements obtained by adding environmental and interaction features were small and were not statistically significant in paired five-fold comparisons. A leakage-safe environmental-alignment permutation test yielded a right-tail empirical $p = 0.047$, which is interpreted as marginal evidence of contextual predictive information rather than evidence of a causal environmental effect. Sensitivity analysis for $k \in \{3, 5, 7, 10\}$ showed similar performance and highly stable feature rankings, with Spearman correlations of approximately 0.971–0.990 relative to $k = 5$. Cross-year classification reached ROC-AUC = 0.707, whereas raw-score regression did not transfer well across cohorts. This moderate performance supports use as a low-stakes risk-prioritization and decision-support aid rather than for precise individual grade prediction. The present evaluation uses information available up to the assessment time and should therefore be interpreted as a pre-assessment risk-identification study.**

***Keywords*—learning analytics, Moodle, feature engineering, student performance prediction, focused learning window, environmental context, pre-assessment risk identification**

I. Introduction

Machine learning is increasingly used to support educational analysis and decision-making [1, 2]. A central step in this process is feature engineering, through which raw data are transformed into representations that can be used effectively by predictive models [3]. Traditionally, feature engineering includes feature extraction, feature evaluation, and feature selection. Although these steps are well established in broader machine learning applications, including speech recognition, computer vision, and data analytics [4], their role is especially important in education because the resulting features must remain interpretable to teachers, students, and institutional stakeholders.

Educational data analytics includes the specialized area of learning analytics, which focuses on the collection, analysis, interpretation, and communication of data about learners and their learning in order to support actionable insights for teaching and learning [1, 5]. Learning analytics is commonly divided into descriptive, prescriptive, and predictive approaches [2]. Descriptive analytics summarizes historical learning behavior, while prescriptive analytics recommends possible interventions. Predictive learning analytics, in contrast, uses machine learning to identify patterns associated with future outcomes. In this setting, the extraction of meaningful features from Learning Management System (LMS) data is a crucial prerequisite for reliable and interpretable prediction.

Several studies have demonstrated the usefulness of LMS data for predicting academic outcomes. For example, Helal *et al.* [6] used traditional classifiers trained on Moodle data and described activity-frequency features obtained by discretizing, counting, summing, and aggregating raw records according to predefined groups such as gender, age, attendance type, and attendance mode. Kotsiantis *et al.* [7] applied decision trees to socio-economic and tutor-record data. Although decision trees provide transparent decisions, the transformation from raw records to predictive features is not described in sufficient detail. Moreover, simple tree models may overfit or become biased toward the majority class, whereas stronger models such as random forests and neural networks can capture more complex relationships at the cost of lower interpretability [4]. In Ref. [8], random forests were used for student-success prediction, with features based on categorized LMS interactions, study sessions, time gaps, and proportions of different activities. By contrast, Aydoğdu [9] proposed a neural-network model, but the feature preparation was limited mainly to one-hot encoding, scaling, and normalization.

Although many studies employ machine learning in educational contexts [6–11], the feature-extraction process is often described only briefly. This limits reproducibility and makes it difficult to understand why a model produces a particular prediction. The need for more explicit and reusable feature-engineering procedures in learning analytics has therefore been emphasized recently [12]. This is not only a technical issue, since feature definitions affect the transparency, bias, interpretability, and practical acceptance of predictive models in education.

To address this limitation, this study defines a reproducible

procedure for transforming Moodle activity logs into structured and interpretable features. In addition to standard behavioral and temporal descriptors, we introduce a focused-learning-window procedure that estimates the beginning of each student's intensive preparation period before the first colloquium. We further align weather and air-quality variables with this estimated interval, and construct interaction terms between learning behavior and environmental context. Environmental variables are treated as contextual descriptors, and their contribution is evaluated through their direct and interaction-based use with LMS behavior. Based on these aims, the study is guided by the following research questions:

RQ1: How can raw Moodle activity logs be systematically transformed into interpretable temporal and behavioral features for predicting student performance?

RQ2: Can the estimated focused learning start time provide useful features for predicting student performance?

RQ3: Do weather and air-quality variables improve prediction when aligned with each student's focused learning interval?

RQ4: Do interaction features between learning behavior and environmental conditions contribute to predictive performance?

The results indicate that the proposed representation is more suitable for interpretable pre-assessment risk identification than for accurate grade prediction. Behavioral variables provide the strongest signal, while environmental variables and interaction terms add only modest numerical improvements. The leakage-safe environmental permutation analysis provides marginal evidence for student-specific environmental alignment ($p = 0.047$), but the observational design does not support causal conclusions. The complementary pass/fail task transfers better across academic cohorts than exact score prediction. Compared with previous LMS-based prediction studies, the novelty of this work lies not only in the use of Moodle activity logs but also in the explicit formalization of the feature-engineering process. The proposed framework advances prior work by combining mathematically defined behavioral and session-based features, an estimated focused learning window, environmental alignment with the individual preparation interval, interaction features between learning behavior and context, and robustness analyses that evaluate regression, pass/fail classification, cross-cohort transfer, statistical uncertainty, and focused-window sensitivity. The contributions of this research are as follows:

(1) An interpretable and reproducible feature-engineering framework for transforming raw Moodle activity logs into behavioral, temporal, session-based, assessment-related, and contextual features.
(2) A focused-learning-window procedure for estimating the beginning of each student's intensive preparation period before assessment.
(3) Integration of weather and air-quality variables aligned with the estimated focused learning interval.
(4) Construction of interaction features between learning behavior and environmental context.
(5) An empirical robustness analysis including feature-set ablation, paired statistical comparisons with confidence intervals, pass/fail classification, cross-year validation, leakage-safe permutation testing of environmental alignment, held-out permutation importance, and focused-window sensitivity for $k \in \{3, 5, 7, 10\}$.

The rest of the paper is structured as follows. Section II reviews related work on feature extraction, feature evaluation, and LMS-based learning analytics. Section III describes the proposed feature-engineering methodology. Section IV presents the evaluation protocol, while Section V reports and discusses the empirical results. The paper concludes with limitations and directions for future work.

II. Related Work

Because the proposed framework combines feature extraction, feature pruning, and LMS-based learning analytics, the related work is organized into three parts. Section II.A reviews general feature-extraction approaches. Section II.B summarizes feature evaluation and selection, which is relevant because the proposed pipeline applies pruning after extraction. Section II.C then focuses on feature extraction from LMS data.

A. Feature Extraction

Feature extraction transforms raw data into features by applying transformations or combining variables with mathematical operators [3, 4]. For example, transforming numerical features to a common range or standard normal distribution ensures that all features contribute equally to the model and that the model is not biased toward features with larger numeric ranges [13, 14]. In addition, combining raw data with mathematical operators can result in informative features that improve model performance. The most commonly used operators are logical [15], arithmetic [16], trigonometric [17], and aggregation operators [18]. For instance, iterative feature recombination involves systematically applying a set of mathematical operators to pairs of raw features to generate new ones. Such approaches were proposed in [19, 20].

However, increasing the number of features not only increases dataset dimensionality but may also introduce redundant features, thereby increasing computational costs without adding valuable information. While higher dimensionality can theoretically improve model performance, in practice it may lead to the "curse of dimensionality", in which data points become increasingly sparse in high-dimensional space, making it more difficult for machine learning models to generalize [4]. When working with high dimensions and highly correlated raw data, techniques such as Principal Component Analysis (PCA) and Independent Component Analysis (ICA) can be used in order to combine correlated variables into a smaller set of uncorrelated or independent components. PCA accomplishes this by transforming the original correlated raw data variables into orthogonal principal components that capture the maximum variance in the data, effectively eliminating linear correlations. In contrast, ICA aims to uncover underlying independent sources by maximizing the statistical independence among components [21]. However, both methods can be computationally intensive, particularly with large datasets, due to the complex calculations involved. For example, PCA requires the computation of covariance matrices and their eigenvalues, tasks that become

increasingly resource-demanding as the dataset size grows. Similarly, the iterative optimization processes of ICA are usually slow when dealing with extensive data. On the other hand, in time series analysis, efficiently representing cyclical variables, such as hours of the day or months of the year, is essential to maintain their inherent periodicity. A mathematically robust approach involves transforming raw time series using sine and cosine functions, effectively mapping them onto the unit circle [22]. Additionally, Wavelet and Fourier transforms decompose time series data into frequency domains, allowing the extraction of new features. By representing the original signal in the frequency domain, these transformations extract recurring structures that manifest at regular intervals, i.e. trends and seasonality [23].

B. Feature Evaluation and Selection

Feature extraction is usually followed by feature evaluation and feature selection. These steps are important because extracted features may contain redundant, irrelevant, or highly correlated information, which can increase model complexity and reduce interpretability [12]. Feature evaluation estimates the relevance of individual features for a predictive task. Common approaches include statistical criteria such as the Fisher criterion [24] and Laplace estimation [25], as well as correlation-based measures such as Pearson and Spearman coefficients [26]. Although correlation-based approaches are simple and interpretable, they mainly capture linear relationships. Information-based and entropy-based approaches can capture more complex dependencies, but they may also be sensitive to features with many distinct values [27, 28].

Feature selection aims to retain informative features while removing irrelevant or redundant ones. The main strategies include filter, wrapper, and embedded methods [29, 30]. Filter methods evaluate features independently of the final learning algorithm and are computationally efficient, but they may ignore interactions between features. Correlation-based feature selection is one example of this approach, selecting features that are related to the target while reducing redundancy among predictors [31]. Wrapper methods evaluate subsets of features using a predictive model, often achieving strong performance but requiring higher computational cost. Embedded methods perform feature selection during model training, for example through tree-based feature importance or regularization.

C. Feature Extraction in Learning Analytics

Early methods for extracting meaningful information from raw LMS data primarily focused on student-aggregated summary and statistical features. These approaches aimed to distill large volumes of activity data into features, such as frequency of interactions, time spent on various learning activities, activity counts, average session duration, most frequently accessed resources, and activity distribution over time [10]. Similarly, the study of Kadoić and Oreški [32] examined related approaches for converting raw log data into meaningful activity metrics. It involved calculating login frequency and total time spent online, as well as counting interactions such as forum topics read and created, lessons accessed and downloaded, quizzes attempted, crosswords completed, and assignments submitted. Additionally, the study incorporated average grades obtained across activities, providing a comprehensive representation of student engagement and performance within the LMS environment. However, both studies describe their methodologies only verbally and lack clear, structured definitions of their feature-extraction processes. Although Leppänen *et al.* [33] uses fine-grained interaction data from learning materials, its transformation of raw data is limited to basic preprocessing steps, such as counting element-level events and normalization, without explicit feature engineering or the construction of higher-level behavioral features.

Beyond traditional counting and averaging methods, more advanced feature extraction techniques were explored by Taylor *et al.* [34]. This study introduced several methods, including ratio and difference calculations, variance analysis, and basic time-difference computations, to derive sophisticated features from student activity logs. Unlike the previous approaches, this study specifically demonstrates the use of lag features, which involve shifting the target variable or other feature values by a specified number of time steps to capture temporal dependencies. By leveraging these transformations, the study provides a comprehensive set of extracted features, each accompanied by detailed descriptions, enhancing the model's ability to recognize patterns and make more accurate predictions over time. Balogh and Kuchárik [35] introduced a feature extraction approach for Moodle sessions, focusing on summarizing student activity patterns using three key statistical measures: activity frequency, mean activity time, and the standard deviation of activity time. To enhance the accuracy of behavioral pattern recognition, the authors also incorporated K-means clustering, enabling the identification of distinct student behavior patterns based on the extracted features. Similarly, the study by Chen and Cui [36] includes an explicit feature-extraction section. However, the extracted features are primarily descriptive in nature, relying on aggregate statistics such as activity frequencies and time-based usage measures, without incorporating higher-level behavioral or temporal abstractions.

The importance of data preprocessing techniques is emphasized in Ref. [11]. While feature extraction is only outlined in general terms, the study applies one-hot encoding and scaling techniques to a combination of demographic profile data and student LMS activities. These preprocessing steps help structure the data for more effective machine learning model training. Furthermore, the preprocessing step for feature extraction from LMS activities, utilizing Discrete Fourier Transform (DFT) and Principal Component Analysis (PCA) on a time series of student activity logs, is presented in Ref. [37]. This approach not only identifies periodic patterns in student behavior but also reduces data dimensionality, which improves machine learning performance by isolating key components that contribute to students' engagement and performance.

Recent research on explainable artificial intelligence in learning analytics increasingly applies post-hoc methods to identify the variables associated with model predictions. However, predictive accuracy alone does not establish that an explanation is stable or educationally useful. Tiukhova *et al.* [38] examined the stability of feature importance across student cohorts and model configurations,

showing that explanation consistency is an important requirement for educational interpretation. Other studies have examined the balance between predictive performance and explainability in student-risk models [39]. Moreover, explanations of incorrect predictions may still produce unsuitable recommendations, indicating that model outputs require human interpretation before they are translated into educational interventions [40]. A recent systematic review confirms that explainability in learning analytics and educational data mining includes both model-level and instance-level approaches, but that their evaluation remains methodologically heterogeneous [41].

The present study occupies a complementary position within this literature. Instead of relying primarily on post-hoc explanations, interpretability is introduced through explicitly defined features linked to recognizable behavioral, temporal, assessment-related, and contextual constructs. Permutation importance is subsequently used as a global assessment of feature use by the trained models.

III. Methodology

The methodology is designed to make each transformation from raw Moodle logs to predictive variables explicit. In contrast to studies that describe LMS feature extraction only at a high level [12], all main features in this work are defined mathematically. The mathematical definitions are intentionally retained in the main methodology because the principal contribution of this work is a reproducible and interpretable feature-engineering framework rather than only the predictive modeling itself. This reduces ambiguity, supports reproducibility, and allows the framework to be extended or reimplemented in other educational settings. The pipeline begins with preprocessing and filtering of raw log records, continues with the estimation of a focused learning start time, and then derives behavioral, temporal, session-based, assessment-related, environmental, and interaction features. The extracted variables are finally aggregated and pruned to form a compact feature vector used for student performance prediction. A compact overview of all engineered features is provided in Appendix A, while the complete mathematical definitions are retained in this section to facilitate reproducibility and independent implementation of the proposed framework.

A. Data Collection and Preprocessing

The data were collected from Moodle in the form of standard CSV log files. Each record contains a timestamp, an anonymized student identifier, the LMS component, the activity context, and the event description. The logs cover interactions with learning materials, assignments, quizzes, and other course resources.

The dataset consists of activity records from mandatory undergraduate courses at the Faculty of Organization and Informatics, University of Zagreb. For each academic year, the observation period begins at the start of the semester and ends at the first colloquium. The two academic cohorts were processed separately and then combined for the main analysis. The analyzed courses were delivered primarily in a face-to-face format. Moodle was used as a complementary LMS for distributing learning materials, administering assignments and quizzes, and recording student learning activities. Therefore, the extracted Moodle features represent students' interaction with the digital learning environment that supported classroom teaching, rather than activity in a fully online course. The final analytic sample comprised 110 students in the 2023 cohort and 232 students in the 2024 cohort, for a total of 342 students. The cohort labels correspond to the assessment years used in the cross-year validation.

The first colloquium assessed the core intended learning outcomes of the analyzed courses, including understanding of course concepts, application of theoretical knowledge, and completion of course-related assignments and quizzes. Although the specific learning outcomes differed between courses and academic years, the proposed feature-engineering framework does not depend on a particular course syllabus. Instead, it requires LMS activity records that can be mapped to learning materials, assignments, quizzes, and assessment outcomes. The first colloquium was scored on a scale from 0 to 25 points. For the classification analysis, a score of at least 12 points was treated as passing.

All personally identifiable information was removed or anonymized before analysis. For each student s, the raw LMS records were transformed into a structured feature vector. The prediction target is the number of points obtained on the first colloquium:

$$y^s = \text{colloquium score of student } s.$$

Before feature extraction, timestamps were converted to datetime format and invalid timestamps were removed. Numeric values written with decimal commas were standardized by replacing commas with decimal points. Students without relevant learning-material activity before the colloquium were excluded, since their focused learning interval could not be estimated reliably.

B. Selection of Learning-material Activities

To estimate focused learning behavior, only LMS activities directly related to course learning materials were considered. These activities represent interactions with instructional resources relevant to the assessed course content, such as lecture materials, course documents, or other study resources. Note that, in the present implementation, learning-material activity was operationalized as access to the PDF instructional resources corresponding to the first eight course materials. This mapping was specific to the analyzed courses. When applying the framework to another course, the corresponding instructional-resource mapping must be defined before feature extraction.

Let t_{end} denote the timestamp of the colloquium, while T_L^s denotes the ordered set of learning-material activity timestamps for student s before the colloquium:

$$T_L^s = \{t_i \mid t_i \le t_{end} \wedge t_i \text{ is associated with a learning-material activity}\}.$$

This filtering step ensures that the estimation of focused preparation is based on study-related behavior rather than general LMS navigation.

C. Estimation of Focused Learning Start Time

The goal of this step is to estimate the beginning of the

student's focused preparation period before the first colloquium. Let T_L^s denote the sorted set of relevant learning-material timestamps of student s.

Focused preparation was estimated using a rolling window approach. For each timestamp $t_i \in T_L^s$, a candidate window of length k days was defined as:

$$w_i = [t_i, t_i + k).$$

The set of activities belonging to this window is:

$$T_{w_i}^s = \{t_j \in T_L^s | t_i \leq t_j < t_i + k\}.$$

In this study, k = 5. The window length k defines the temporal scale at which focused learning behavior is detected. The window size was set to k = 5 days based on the structure of academic preparation cycles and expected behavioral patterns before assessment. In a higher-education context, a five-day window approximates a continuous working week of preparation and is long enough to capture sustained exam-oriented engagement rather than isolated, single-day LMS logins. This choice is also consistent with learning analytics studies showing that student activity often becomes temporally concentrated before assessments, rather than being uniformly distributed across the semester [36, 42]. From the perspective of self-regulated learning, such temporal concentration can be interpreted as part of students' time-management and preparation strategies [43]. Methodologically, very short windows may be overly sensitive to high-frequency noise caused by isolated downloads or brief access events, while substantially longer windows may over-smooth the learning signal and mask short-term activation patterns typical of intensive pre-exam preparation. Thus, k = 5 was selected as a practical compromise between sensitivity to focused study bursts and robustness to sporadic LMS activity. The most active learning window was selected as:

$$w^* = \arg\max_{w_i} |T_{w_i}^s|$$

The number of learning-material activities contained in the selected most-active window was also retained as a behavioral feature:

$$A_{w^*}^s = |T_{w^*}^s|$$

This feature represents the concentration of learning-material activity within the most active k-day preparation interval.

After identifying the most active learning window, the most active calendar day inside this window was selected. For each day d, the set of activities on that day was defined as:

$$T^{s,d} = \{t_i \in T_{w^*}^s \mid day(t_i) = d\}$$

where $day(t_i)$ denotes the calendar day corresponding to timestamp t_i. Let $D(w^*)$ denote the set of calendar days contained in the window w^*. Thus, the most active day is defined as:

$$d_{\max} = \arg\max_{d \in D(w^*)} |T^{s,d}|$$

Finally, the estimated focused learning start time was defined as the earliest activity timestamp on the most active day:

$$t_\alpha^s = \min(T^{s,d_{\max}})$$

This value represents an approximation of the start of the student's most intensive preparation phase, not the exact first moment of learning. The five-day value was retained as the primary setting because it was specified conceptually rather than selected by predictive performance. To test the robustness of this choice, the complete feature-generation and evaluation workflow was additionally repeated for $k \in \{3, 7, 10\}$ days, with k = 5 retained as the reference configuration.

D. Feature Extraction

After estimating t_α^s, several groups of features were extracted from the LMS logs to capture different aspects of student behavior. Let T_{all}^s denote the set of all LMS activity timestamps for student s before the colloquium.

The extracted features can be broadly categorized into the following groups:

- Activity-based features, describing the overall volume and temporal distribution of LMS interactions across the entire observation period,
- Session-based features, capturing how student activity is organized into sessions and how regularly these sessions occur over time,
- Learning-window features, characterizing the timing, intensity, and temporal distribution of activity within the estimated focused learning interval $[t_\alpha^s, t_{end}]$,
- Assignment features, reflecting student engagement with assignments through completion time and post-submission interaction,
- Quiz features, capturing patterns of quiz-related activity and their temporal relationship to the focused learning phase,
- Temporal context features, describing activity patterns with respect to calendar structure, such as weekend engagement,
- Weather and environmental features, incorporating external contextual conditions during the focused learning period,
- Derived features, capturing higher-level properties such as learning intensity, efficiency, and regularity,
- Interaction features, modeling relationships between behavioral and environmental variables.

Each group is formally defined in the following subsections.

1) General activity features

The total number of LMS activities before the colloquium was computed as:

$$A^s = |T_{all}^s|$$

This feature reflects the overall level of LMS engagement before the assessment. Activities were also grouped by part of day. The following four parts of day were defined:

$Night$ = [00:00, 06:00),

$Morning$ = [06:00, 12:00),

$A\,f\,ternoon$ = [12:00, 18:00),

$Evening$ = [18:00, 24:00).

For each part of day p, the number of activities was calculated as:

$$A_p^s = |\{t_i \in T_{all}^s | hour(t_i) \in p\}|,$$

where $hour(t_i)$ returns the hour of the day of timestamp t_i. These quantities describe the temporal distribution of LMS usage throughout the day, capturing differences in study habits and daily learning routines. Prior work in learning analytics suggests that the timing of student activity can reflect underlying behavioral patterns such as self-regulation and procrastination, which are associated with academic performance [42].

In addition, weekend activity was defined as:

$$W^s = |\{t_i \in T_{all}^s | weekday(t_i) \in \{Saturday, Sunday\}\}|,$$

where $weekday(t_i)$ returns the day of the week corresponding to timestamp t_i. This quantity captures the extent to which students engage with the LMS during weekends, reflecting differences in study scheduling and the use of non-working days for learning.

2) Learning-window features

The time difference between the estimated focused learning start time and the colloquium was calculated in days and hours:

$$D_\alpha^s = \frac{t_{end} - t_\alpha^s}{24 \cdot 3600},$$

$$H_\alpha^s = \frac{t_{end} - t_\alpha^s}{3600}.$$

These quantities indicate how early the student enters the focused preparation phase. The number of learning-material activities during the focused learning interval was computed as:

$$A_\alpha^s = |\{t_i \in T_L^s | t_\alpha^s \le t_i \le t_{end}\}|.$$

This measure captures the intensity of study-related activity during the preparation period. The activity distribution by part of day was also computed for the focused learning interval:

$$A_{\alpha,p}^s = |\{t_i \in T_L^s | t_\alpha^s \le t_i \le t_{end} \wedge hour(t_i) \in p\}|.$$

These quantities describe when focused learning activity occurs.

3) Session-based features

Student sessions were defined using the 50-minute (δ = 50) inactivity threshold configured in the institutional Moodle environment. Consecutive activities separated by no more than 50 min were therefore treated as belonging to the same session. We did not select or optimize this threshold in the present study. It reflects the session configuration of the analyzed LMS environment. A session λ_j^s is a maximal sequence of consecutive activities satisfying:

$$t_i - t_{i-1} \le \delta.$$

Timestamps in T_{all}^s are ordered such that t_{i-1} denotes the immediately preceding activity of t_i. The set of sessions for student s is:

$$\Lambda^s = \{\lambda_1^s, \lambda_2^s, \dots, \lambda_J^s\}.$$

The total number of sessions is:

$$S^s = |\Lambda^s|.$$

This reflects how frequently a student returns to the LMS. Sessions were also assigned to parts of day according to the timestamp of the first activity in the session:

$$S_p^s = \left|\{\lambda_j^s \in \Lambda^s | hour\left(start\left(\lambda_j^s\right)\right) \in p\}\right|.$$

These quantities describe when students typically begin their learning sessions. The time gap between two consecutive sessions was calculated as:

$$g_j^s = start\left(\lambda_j^s\right) - end\left(\lambda_{j-1}^s\right), j = 2, \dots, J.$$

Here $start\left(\lambda_j^s\right)$ and $end\left(\lambda_j^s\right)$ denote the first and last timestamps of session λ_j^s, respectively. From these gaps, mean, median, and mode session periodicity were computed:

$$\mu_g^s = mean\left(g_j^s\right),$$

$$\tilde{g}^s = median\left(g_j^s\right),$$

$$m_g^s = mode\left(g_j^s\right),$$

where $mean(\cdot)$, $median(\cdot)$, and $mode(\cdot)$ denote standard statistical operators. These measures capture not only the frequency of LMS usage but also its temporal structure. In contrast to aggregate activity counts, session-based features provide insight into how learning is distributed over time, distinguishing between regular, repeated engagement and irregular or concentrated study patterns.

4) Assignment and quiz features

Let Z^s denote the set of assignments associated with student s. For each assignment $z \in Z^s$, completion time was calculated as:

$$C^{s,z} = t_{submit}^{z,s} - t_{start}^{z,s},$$

where $t_{start}^{z,s}$ denotes the timestamp of the first assignment-related activity. If no explicit submission event was found, the last assignment-related activity was used as the end of the assignment interaction interval. The total assignment completion time was computed as:

$$C^s = \sum_{z \in Z^s} C^{s,z}.$$

This captures the total duration of assignment-related work. Let $T^{z,s}$ denote the set of timestamps associated with assignment z for student s. For each assignment, the number of post-submission review activities was calculated as:

$$V^{s,z} = \left|\left\{t_i \in T^{z,s} \mid t_i > t^{z,s}_{submit},\ t_i \in \mathcal{R}^{z,s}\right\}\right|.$$

Here, $\mathcal{R}^{z,s}$ denotes the set of post-submission status-view events associated with assignment z for student s, while $t^{z,s}_{submit}$ denotes the timestamp of assignment submission. The total post-submission assignment review count was computed as:

$$V^s = \sum_{z \in Z^s} V^{s,z}.$$

These measures capture both the temporal extent of assignment engagement and post-submission behavior. While completion time reflects how students distribute their effort during assignment work, the post-submission count captures subsequent checking of the assignment submission status.

Regarding the quiz features, let T^s_Q denote the set of quiz-related activity timestamps for student s:

$$T^s_Q = \{t_i \in T^s_{all} \mid t_i \text{ corresponds to quiz activity}\}.$$

The total number of quiz-related activities was calculated as:

$$Q^s = \left|T^s_Q\right|.$$

This quantity reflects overall quiz engagement.

The number of quiz-related activities after the estimated focused learning start time was calculated as:

$$Q^s_\alpha = \left|\left\{t_i \in T^s_Q \mid t_i \geq t^s_\alpha\right\}\right|.$$

This captures quiz activity during the preparation phase.

The time difference between the first quiz-related activity and the colloquium was calculated as:

$$H^s_Q = \frac{t_{end} - \min\left(T^s_Q\right)}{3600}.$$

This quantity reflects how early the student engaged with quizzes. While total quiz activity reflects overall engagement with self-assessment, the timing of quiz interactions provides insight into study strategies, distinguishing between early preparation and last-minute practice.

An additional exercise-related course activity descriptor, denoted P^s_{ex}, was obtained from the course-record field Vjezbe. This variable represented exercise-related activity recorded before the first-colloquium timestamp and was available consistently across the analyzed cohorts. It was not derived from the first-colloquium score or from post-assessment information. Only this explicitly defined auxiliary predictor and the target score were imported from the course record. We excluded all other gradebook fields using an explicit whitelist.

5) Weather and air-quality features

Weather and air-quality data were collected for the Varaždin area, Croatia, corresponding to the local study environment of the Faculty of Organization and Informatics, University of Zagreb. Since the analyzed courses were delivered at this institution, the same local environmental context was assigned to all students for a given calendar day. Weather and air-quality data were integrated with LMS features as contextual descriptors that may contain incremental predictive information alongside student learning behavior. Let $E_m(d)$ denote the value of environmental variable m on date d. The set of environmental variables considered in this study is defined as:

$$m \in \mathcal{M} = \begin{Bmatrix} pm10, pm2.5, tavg, tmin, tmax, prcp, \\ \text{snow,wdir,wspd,wpgt,pres} \end{Bmatrix}.$$

These variables include air-quality indicators, temperature-related measures, precipitation, and wind-related factors. For each student, the average value of each environmental variable during the focused learning interval was calculated as:

$$\bar{E}^s_m = mean(\{E_m(d) \mid date(t^s_\alpha) \leq d \leq date(t_{end})\}).$$

where $date(t)$ extracts the calendar date from timestamp t. These quantities summarize the local environmental context during the student's estimated focused preparation interval. Prior studies have reported associations between environmental conditions, cognitive performance, and daily behavior [44, 45]. In the present observational setting, however, these variables are not interpreted as individual exposures or causal determinants. Their role is limited to testing whether local contextual conditions provide incremental predictive information when aligned with student-specific learning intervals.

6) Learning intensity and regularity features

Learning intensity per day was calculated as:

$$I^s_D = \frac{A^s_\alpha}{D^s_\alpha}$$

The average number of focused-learning activities per session was calculated as:

$$I^s_S = \frac{A^s_\alpha}{S^s}$$

The proportion of all LMS activity during the focused learning interval was:

$$R^s_L = \frac{A^s_\alpha}{A^s}$$

Learning regularity was represented as:

$$R^s_\mu = \frac{1}{\mu^s_g + 1},$$

$$R_{\tilde{g}}^{s} = \frac{1}{\tilde{g}^{s} + 1}$$

These quantities provide normalized measures of student behavior that go beyond raw activity counts. While activity-based features describe the overall volume of LMS interaction, these derived features capture how learning effort is distributed across time and sessions. They distinguish between sustained, regularly distributed learning and concentrated or irregular study patterns, thereby reflecting differences in learning strategies, efficiency, and self-regulated behavior [43]. These derived behavioral features form the basis for modeling how learning patterns interact with external contextual factors, as described in the following subsection.

7) Learning-environment interaction features

To represent context-dependent associations between learning behavior and environmental conditions, interaction features were defined by combining behavioral measures with contextual variables.

$$Z_{m}^{s} = A_{\alpha}^{s} \cdot \bar{E}_{m}^{s},$$

$$Z_{D,m}^{s} = I_{D}^{s} \cdot \bar{E}_{m}^{s},$$

$$Z_{S,m}^{s} = S^{s} \cdot \bar{E}_{m}^{s},$$

$$Z_{R,m}^{s} = R_{\mu}^{s} \cdot \bar{E}_{m}^{s}$$

Each interaction term represents a joint behavioral-contextual descriptor rather than a causal effect. The term Z_{m}^{s} combines focused learning activity with the corresponding environmental average. The term $Z_{D,m}^{s}$ combines daily learning intensity with environmental context, $Z_{S,m}^{s}$ combines session frequency with environmental context, and $Z_{R,m}^{s}$ combines learning regularity with environmental context. Together, these features allow the predictive models to represent context-dependent associations in which otherwise similar learning patterns may carry different predictive information under different local conditions.

E. Feature Aggregation and Pruning

After feature extraction, deterministic aggregation steps were applied to reduce sparsity and improve interpretability. Assignment-specific variables were aggregated into total assignment completion time C^{s} and total post-submission review count V^{s}. Weekend activity was represented by the total number of LMS activities during weekends W^{s}, and raw datetime variables were removed after numerical features had been derived. Environmental features were restricted to averages over the focused learning interval, while minimum, maximum, and fixed pre-exam aggregates were excluded to reduce redundancy. All feature-construction steps preceding pruning are deterministic student-level transformations based only on each student's pre-assessment records. They do not use statistics estimated from other students, test-fold information, or held-out target values. Data-dependent pruning was performed inside the validation procedure rather than once on the complete dataset. For every cross-validation split, zero-variance predictors and predictors exceeding the absolute pairwise correlation threshold of 0.95 were identified from the training fold only. The held-out fold was then reindexed to the columns retained from its corresponding training fold. The same principle was used in cross-year validation, where pruning was fitted on the training cohort and applied to the held-out cohort. The final feature vector for student s is defined as:

$$\vec{x}^{s} = \langle f_{1}^{s}, f_{2}^{s}, \ldots, f_{K}^{s} \rangle.$$

The resulting supervised learning problem is defined as:

$$\vec{x}^{s} \rightarrow y^{s},$$

where y^{s} is the number of points obtained on the first colloquium.

IV. Evaluation Protocol

The extracted features were evaluated in two complementary prediction settings. The first setting is regression, where the target variable y^{s} is the number of points obtained by student s on the first colloquium. The second setting is pass/fail classification, which provides a practically oriented pre-assessment risk-identification formulation.

A. Feature Subsets

Table 1 distinguishes the feature space produced by deterministic feature construction from the dimensionality retained after data-dependent pruning. Before pruning, the three candidate sets contained 33, 44, and 88 predictors, respectively. Zero-variance and correlation-based pruning was fitted independently within each training fold, so the retained dimensionality could vary slightly between folds. A compact overview of the main feature groups is provided in Appendix A.

To analyze the contribution of different feature groups, three feature sets were constructed. These sets progressively incorporate behavioral, environmental, and interaction-based information, enabling a controlled analysis of their individual and combined contributions. The evaluated feature sets are formally defined in Table 1.

Table 1. Candidate feature sets before fold-wise pruning and dimensionality retained after training-fold-only zero-variance and correlation pruning

Feature set	Candidate definition	Candidates	Retained, regression	Retained, classification
Behavior only	All behavioral, temporal, learning-window, session, assignment, quiz, exercise, and derived features defined in Sections III.C–III.D, including the four day-part activity variables A_{p}^{s}, four session day-part variables S_{p}^{s}, four focused-learning day-part variables $A_{\alpha,p}^{s}$,the best-window activity count $A_{w^{*}}^{s}$, and P_{ex}^{s}.	33	27	27
Behavior + weather	Behavior-only candidates augmented with the 11 environmental averages $\bar{E}_{m}^{s}$, $m \in \mathcal{M}$.	44	34–35	35
Behavior + weather + interactions	Behavior + weather candidates augmented with the four interaction families Z_{m}^{s}, $Z_{D,m}^{s}$, $Z_{S,m}^{s}$, and $Z_{R,m}^{s}$ for each $m \in \mathcal{M}$.	88	49–50	50

- Behavior only: includes temporal, behavioral, session-based, assignment, quiz, exercise-related, and derived features, excluding environmental and interaction features.
- Behavior + weather: extends behavioral features with environmental variables averaged over the focused learning interval.
- Behavior + weather + interactions: further extends the feature set with interaction terms between behavioral and environmental variables.

B. Regression Evaluation

The primary prediction task was formulated as a regression problem, where the target variable y^s represents the number of points obtained by student s on the first colloquium. The regression task evaluates how accurately the extracted features can predict continuous student performance.

Three regression approaches were evaluated. A mean baseline model was used as a reference and predicted the average target value estimated from the training data. In addition, two nonlinear machine learning models were evaluated: Random Forest [46] and XGBoost [47]. Both models were implemented using standard machine learning libraries [48]. The Random Forest models used 800 trees, a maximum tree depth of 10, a minimum of four samples per leaf, and a fixed random seed of 42. For classification, balanced class weighting was used. The XGBoost regression model used 800 trees, a learning rate of 0.02, a maximum depth of 3, subsample and column-subsample ratios of 0.8, the squared-error regression objective, and random seed 42. The XGBoost classification model used 500 trees, a learning rate of 0.03, a maximum depth of 3, subsample and column-subsample ratios of 0.8, the binary logistic objective, log-loss evaluation, and random seed 42. Its class-imbalance weight (scale_pos_weight) was recalculated from the class ratio in each training fold so that no held-out labels contributed to model configuration.

Random Forest and XGBoost were selected because the analyzed dataset consists of 342 students represented by a relatively small set of structured tabular features. Both methods can capture nonlinear relationships and interactions without requiring feature scaling or a large sequential training dataset. They also support global inspection through permutation-based feature importance. Recent educational data mining studies have considered deep neural networks and transformer-based architectures [39, 49]. However, these approaches generally involve higher data requirements, additional model complexity, and a less direct correspondence between the engineered variables and the resulting predictions. Since the primary objective of this study is to evaluate an interpretable feature engineering framework rather than to provide an exhaustive benchmark of predictive architectures, deep learning and transformer models were not included. This absence remains a limitation, and future work should compare the proposed representation across a broader range of conventional and deep learning models.

Model evaluation was performed using five-fold cross-validation. In each fold, zero-variance and correlation-based pruning were fitted exclusively on the training data, after which the selected feature columns were applied to the held-out fold. The models were then trained on the pruned training fold and evaluated on its held-out fold. For XGBoost classification, the class-weight ratio was also calculated from the training-fold labels only. Results are reported as the mean and descriptive population standard deviation across the five fold-level values. For the principal Random Forest comparisons, 95% confidence intervals were calculated separately from the fold-level values using the t distribution and the sample standard error. The same folds were used across feature sets to permit paired statistical comparisons.

The following evaluation metrics were used:

$$R^2 = 1 - \frac{\sum_i (y_i - \hat{y}_i)^2}{\sum_i (y_i - \bar{y})^2},$$

$$RMSE = \sqrt{\frac{1}{n}\sum_i (y_i - \hat{y}_i)^2},$$

$$MAE = \frac{1}{n}\sum_i |y_i - \hat{y}_i|$$

C. Classification Evaluation

In addition to regression, a complementary binary classification task was evaluated to assess the usefulness of the extracted features for pre-assessment risk identification. The same machine learning algorithms, Random Forest and XGBoost, were applied in the classification setting using their classification variants. The continuous colloquium score was transformed into a binary pass/fail target using the passing threshold of 12 points:

$$y^s_{\text{pass}} = \begin{cases} 1, & y^s \geq 12, \\ 0, & y^s < 12 \end{cases}$$

The classification dataset contained 342 students, of whom 198 failed and 144 passed. This corresponds to 57.89% failed students and 42.11% passed students, indicating moderate class imbalance. Therefore, model performance was evaluated using accuracy, balanced accuracy, F1-score, precision, recall, and ROC-AUC. As other confusion-matrix-based metrics are well known, we define only balanced accuracy, which is used to account for class imbalance. It is defined as the average recall across both classes:

$$\text{Balanced Accuracy} = \frac{1}{2}\left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\right),$$

where TP, TN, FP, and FN denote true positives, true negatives, false positives, and false negatives, respectively. Classification models were evaluated using stratified 5-fold cross-validation to preserve the class distribution across folds. The same feature sets used in the regression task were evaluated in the classification setting.

D. Prediction Timing and Intended Use

The present experiments use LMS and environmental records from the beginning of the observation period up to the timestamp of the first colloquium. Therefore, the reported results do not constitute a validation of a fixed operational

early-warning horizon such as 14, 7, or 3 days before assessment. They quantify how much pre-assessment behavioral and contextual information is identifiable when the complete pre-colloquium observation period is available. In a prospective deployment, a cutoff $t_{cut} = t_{end} - h$ should be imposed and every LMS, focused-window, and environmental feature should be recomputed using records available no later than t_{cut}. Evaluating such fixed horizons for $h \in \{14, 7, 3\}$ days is an important next step for establishing intervention lead time.

E. Cross-cohort Validation

To assess generalizability across academic cohorts, an additional cross-year validation was conducted. Models were trained on one academic year and tested on the other, and vice versa. This procedure evaluates the robustness of the learned relationships under distributional shifts between cohorts.

Cross-cohort validation was evaluated using the raw colloquium score and pass/fail target without including the academic year as a predictive feature. No transformation based on the held-out cohort's target distribution was used. For each train–test direction, zero-variance and correlation pruning were fitted on the training cohort only, and the resulting feature columns were then applied to the held-out cohort. This design avoids using information from the test cohort when defining the predictor space and provides a stricter estimate of transfer across academic years.

F. Robustness, Permutation, and Feature-importance Analysis

To further evaluate the robustness of the proposed feature-engineering framework, five complementary analyses were conducted. First, the three progressively enriched feature sets were compared for both regression and pass/fail classification using identical five-fold splits. Second, 95% confidence intervals and paired Wilcoxon signed-rank and paired t tests were calculated from fold-level performance to quantify uncertainty in the incremental differences between feature sets. Third, environmental alignment was evaluated with 300 permutations. For every permuted evaluation, data-dependent pruning was again fitted within the corresponding training fold only. Fourth, Random Forest permutation importance was calculated on held-out folds and averaged across folds, avoiding importance estimates based on the same observations used to fit the model. Fifth, the complete feature-generation and evaluation pipeline was repeated for focused-window lengths $k \in \{3, 5, 7, 10\}$, and performance together with feature-ranking stability was compared against the primary $k = 5$ configuration.

For the environmental-alignment permutation test, the right-tail empirical p-value was calculated as:

$$p = \frac{1 + \sum_{b=1}^{B} \mathbf{1}\left(\mathrm{AUC}_{\mathrm{perm}}^{(b)} \geq \mathrm{AUC}_{\mathrm{obs}}\right)}{B + 1},$$

where $B = 300$ is the number of permutations and $\mathbf{1}(\cdot)$ is the indicator function. The Random Forest configuration, full feature set, five-fold validation splits, and random seed of 42 were fixed across the observed and permuted evaluations. For every observed and permuted evaluation, zero-variance and correlation pruning was independently fitted using the corresponding training fold only.

V. Results and Discussion

The proposed feature-engineering framework was implemented in Python using the scikit-learn and XGBoost libraries. Experiments were conducted on a Microsoft Windows 11 system equipped with an AMD Ryzen 7 CPU and 32 GB of RAM. The revised evaluation uses leakage-safe fold-wise preprocessing and reports regression and classification performance, statistical uncertainty, focused-window sensitivity, cross-year generalization, environmental permutation testing, and held-out feature-importance analysis.

A. Regression Performance

Table 2 and Fig. 1 show that Random Forest produced the strongest regression results. Using behavioral features alone, it achieved $R^2 = 0.117$. Adding environmental averages increased this value to $R^2 = 0.136$, while the full interaction set reached $R^2 = 0.149$ with RMSE = 3.327. These differences are small relative to fold-to-fold variability and, as shown below, are not statistically significant in paired fold comparisons. The absolute R^2 also remains modest, so the result should not be interpreted as high-precision individual grade prediction. Instead, it indicates limited but measurable predictive structure in the engineered pre-assessment variables.

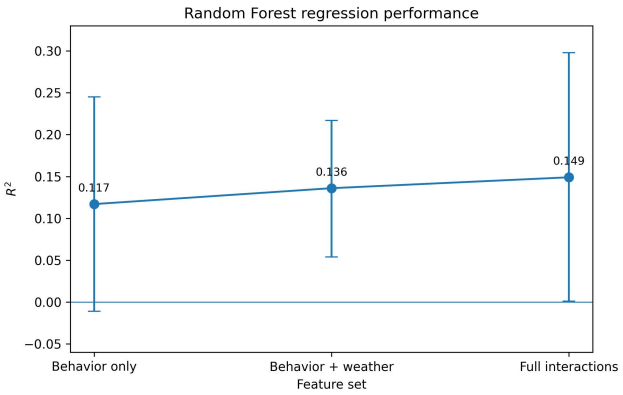

Fig. 1. Random Forest regression performance across feature sets. Points show mean R^2 across five folds and error bars show 95% confidence intervals.

Table 2. Regression performance using leakage-safe five-fold cross-validation. Values are mean ± descriptive population standard deviation across folds. Feature counts are fold-dependent because pruning is fitted on the training fold only

Feature set	Model	R^2	RMSE	MAE	Retained features
Behavior only	Baseline mean	−0.005 ± 0.002	3.623 ± 0.291	2.909 ± 0.198	–
Behavior only	Random Forest	0.117 ± 0.092	3.382 ± 0.217	2.656 ± 0.161	27
Behavior only	XGBoost	0.025 ± 0.103	3.555 ± 0.249	2.812 ± 0.193	27
Behavior + weather	Random Forest	0.136 ± 0.059	3.355 ± 0.267	2.649 ± 0.190	34–35
Behavior + weather	XGBoost	0.038 ± 0.068	3.539 ± 0.275	2.763 ± 0.163	34–35
Behavior + weather + interactions	Random Forest	**0.149 ± 0.107**	**3.327 ± 0.347**	**2.609 ± 0.258**	49–50
Behavior + weather + interactions	XGBoost	0.047 ± 0.092	3.520 ± 0.281	2.763 ± 0.203	49–50

B. Pass/Fail Classification Performance

Using the pass/fail target and class distribution defined in Section IV.C, Table 3 summarizes the classification results.

As shown in Table 3 and Fig. 2, all trained models exceeded the majority-class baseline in balanced discrimination. The highest ROC-AUC was obtained by Random Forest with the full feature set (AUC = 0.684), while XGBoost with the full feature set achieved the highest balanced accuracy and F1-score. The difference between the behavioral Random Forest AUC (0.676) and the full-model AUC (0.684) is small. An AUC of 0.684 represents moderate rather than strong discrimination, and the regression result of R^2 = 0.149 likewise indicates limited continuous-score predictability. The models are therefore not appropriate for automated grading or high-stakes decisions. Their intended role is low-stakes risk prioritization and decision support, with any intervention requiring instructor judgment and other academic information not represented in the LMS logs.

Table 3. Pass/fail classification performance using leakage-safe stratified five-fold cross-validation. Values are mean ± descriptive population standard deviation across folds

Feature set	Model	Acc.	Bal. Acc.	F1	AUC	Retained features
Behavior only	Baseline majority	0.579 ± 0.005	0.500 ± 0.000	0.000 ± 0.000	0.500 ± 0.000	–
Behavior only	Random Forest	0.643 ± 0.031	0.624 ± 0.035	0.539 ± 0.055	0.676 ± 0.041	27
Behavior only	XGBoost	0.631 ± 0.047	0.620 ± 0.051	0.554 ± 0.072	0.670 ± 0.048	27
Behavior + weather	Random Forest	0.635 ± 0.042	0.616 ± 0.043	0.534 ± 0.056	0.683 ± 0.046	35
Behavior + weather	XGBoost	0.629 ± 0.032	0.618 ± 0.034	0.553 ± 0.045	0.676 ± 0.040	35
Behavior + weather + interactions	Random Forest	0.623 ± 0.049	0.608 ± 0.053	0.533 ± 0.067	**0.684 ± 0.046**	50
Behavior + weather + interactions	XGBoost	**0.643 ± 0.027**	**0.633 ± 0.029**	**0.573 ± 0.039**	0.682 ± 0.037	50

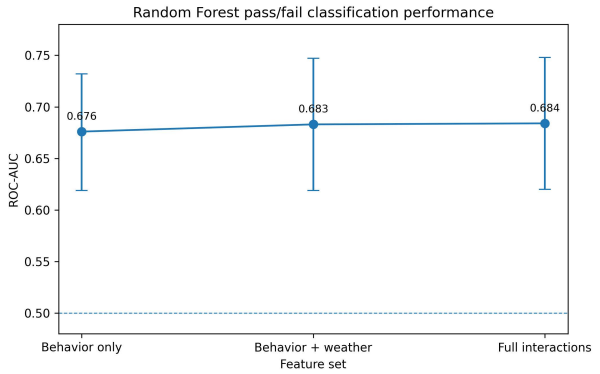

Fig. 2. Random Forest pass/fail classification performance across feature sets. Points show mean ROC-AUC across five folds and error bars show 95% confidence intervals. The dashed horizontal line indicates chance-level ROC-AUC.

C. Statistical Uncertainty and Paired Feature-set Comparisons

For the Random Forest regression model, the 95% confidence intervals for R^2 were [−0.011, 0.245] for Behavior only, [0.054, 0.217] for Behavior + weather, and [0.001, 0.298] for the full feature set. For classification AUC, the corresponding intervals were [0.619, 0.732], [0.619, 0.747], and [0.620, 0.748].

As shown in Table 4, none of the incremental feature-set comparisons reached statistical significance. Paired t tests led to the same qualitative conclusion. Because only five folds are available, these tests have low power and the Wilcoxon test has a coarse attainable p-value resolution. They are therefore interpreted as uncertainty analyses rather than definitive hypothesis tests.

Table 4. Paired Random Forest comparisons across identical five-fold splits. Δ denotes the second feature set minus the first

Task	Comparison	Metric	Mean Δ	95% CI of Δ	t-test p	Wilcoxon p
Regression	Behavior → Weather	R^2	0.019	[−0.048, 0.085]	0.484	0.625
Regression	Behavior → Full	R^2	0.032	[−0.072, 0.137]	0.438	0.625
Regression	Weather → Full	R^2	0.014	[−0.072, 0.100]	0.678	0.625
Classification	Behavior → Weather	AUC	0.007	[−0.034, 0.049]	0.649	0.813
Classification	Behavior → Full	AUC	0.009	[−0.047, 0.064]	0.694	0.625
Classification	Weather → Full	AUC	0.001	[−0.026, 0.028]	0.909	0.813

D. Focused-window Sensitivity

Table 5. Sensitivity of the full Random Forest model to focused-window length k. Values are mean ± descriptive population standard deviation across folds

k (days)	R^2	RMSE	AUC	Balanced accuracy
3	0.168 ± 0.087	3.292 ± 0.325	0.706 ± 0.046	0.641 ± 0.044
5	0.149 ± 0.107	3.327 ± 0.347	0.684 ± 0.046	0.608 ± 0.053
7	0.159 ± 0.094	3.309 ± 0.329	0.690 ± 0.039	0.630 ± 0.061
10	0.142 ± 0.107	3.341 ± 0.347	0.688 ± 0.046	0.617 ± 0.040

As shown in Table 5, the three-day window was numerically strongest, but performance differences across the tested window lengths were modest and did not indicate a sharp optimum. More importantly, Random Forest feature rankings were highly stable relative to the primary five-day configuration: Spearman rank correlations across regression and classification comparisons ranged from 0.971 to 0.990, and 8–10 of the top 10 features were shared. These results support the robustness of the representation to reasonable changes in temporal scale. The primary k = 5 setting is therefore retained for its conceptual interpretation as approximately one working week, not because it maximized predictive performance.

E. Cross-year Generalization

Cross-year validation was used to evaluate whether the extracted features transfer across academic cohorts. Models were trained on one academic year and tested on the other, and vice versa. Table 6 reports the best pass/fail classification results for each train–test direction.

Cross-year classification retained moderate discrimination under the stricter training-cohort-only preprocessing. Training on 2023 and testing on 2024 produced a best AUC of 0.681 with XGBoost and the full interaction set. In the

opposite direction, XGBoost with behavioral and environmental features reached AUC = 0.707. The asymmetry across directions indicates cohort dependence, but pass/fail discrimination transfers more successfully than exact raw-score prediction.

As shown in Table 7, all reported cross-year raw-score regression configurations produced negative R^2 values. Exact score prediction is therefore highly sensitive to cohort-level differences such as assessment difficulty, grading distributions, and course-specific conditions. The more defensible interpretation is that the features contain some transferable information for broad risk discrimination but are not reliable for precise score prediction across cohorts.

Table 6. Selected leakage-safe cross-year pass/fail classification results

Train → Test	Model / feature set	Acc.	Bal. Acc.	F1	AUC
2023 → 2024	XGBoost, Full interactions	0.621	0.638	0.564	0.681
2023 → 2024	Random Forest, Behavior only	0.526	0.597	0.556	0.676
2023 → 2024	Random Forest, Full interactions	0.586	0.597	0.520	0.668
2024 → 2023	XGBoost, Behavior + weather	0.627	0.646	0.602	0.707
2024 → 2023	XGBoost, Behavior only	0.600	0.622	0.560	0.686
2024 → 2023	XGBoost, Full interactions	0.591	0.614	0.545	0.683

Table 7. Selected leakage-safe cross-year regression results

Train → Test	Model / feature set	R^2	RMSE	MAE
2023 → 2024	XGBoost, Full interactions	−0.402	4.176	3.255
2023 → 2024	Baseline mean	−0.438	4.229	3.271
2024 → 2023	Random Forest, Full interactions	−0.206	3.653	2.859
2024 → 2023	XGBoost, Full interactions	−0.219	3.673	2.797

F. Practical Implications for Pre-assessment Risk Identification

The results support low-stakes decision support rather than automated prediction of individual academic outcomes. Instructors could use interpretable behavioral indicators to prioritize students who may warrant additional attention near an assessment. However, because the current evaluation uses the full observation period up to the colloquium, it does not establish how early such a signal becomes actionable. A prospective early-warning system should first be validated at fixed lead times, for example 14, 7, and 3 days before assessment, before claims about timely intervention are made.

Because the features are interpretable, the model output can be connected to observable learning behavior. For example, a student identified as being at risk due to low focused-learning activity or delayed quiz engagement can receive a different intervention than a student with frequent but irregular sessions. This makes the framework more suitable for educational decision support than black-box prediction alone.

The proposed framework is intended to complement, rather than replace, instructor judgment, since academic performance depends on numerous cognitive, pedagogical, and social factors that cannot be fully captured through LMS interaction logs alone. Consequently, the generated predictions should be interpreted as decision-support information that may assist instructors in identifying students who could benefit from timely educational interventions, rather than as a substitute for comprehensive academic evaluation.

G. Environmental Permutation Test

To examine whether environmental and interaction features carried information beyond arbitrary feature expansion, an alignment permutation test was performed. Environmental and interaction-related columns were permuted across students, breaking their student-specific alignment while preserving marginal distributions. The complete cross-validation pipeline, including training-fold-only pruning, was repeated for each permutation.

As shown in Table 8, the observed AUC of the full Random Forest model was 0.684, compared with a mean permuted AUC of 0.653. The right-tail empirical p-value was 0.047. Because this value lies close to the conventional 0.05 threshold, the result is interpreted as marginal evidence that student-specific alignment between environmental context and learning behavior contains predictive information. It is not interpreted as evidence that environmental conditions cause changes in learning behavior or academic performance. The modest predictive increment, observational design, shared local environmental measurements, and limited sample require cautious interpretation and independent validation.

Table 8. Leakage-safe permutation test for environmental and interaction feature alignment

Quantity	Value
Observed AUC	0.684
Mean permuted AUC	0.653
Std. permuted AUC	0.018
Minimum permuted AUC	0.603
Maximum permuted AUC	0.721
Number of permutations	300
Right-tail empirical p-value	0.047

The permutation test and the paired feature-set comparisons address different questions. The permutation test evaluates whether student-specific alignment between environmental/contextual features and individual learning behavior carries information beyond arbitrary feature expansion, whereas the paired comparisons evaluate whether the incremental performance gain of the enriched feature sets is large relative to fold-to-fold variability. A marginal permutation result together with non-significant paired feature-set gains is therefore internally consistent.

H. Permutation-based Feature Importance

As summarized in Table 9 and Fig. 3, held-out permutation importance provides a more conservative view than importance calculated on the training sample. Classification is dominated by direct behavioral indicators, including quiz activity after the focused start, homework completion time, afternoon activity, exercise-related activity, and focused afternoon activity. In regression, the activity–precipitation interaction has the largest mean held-out importance, followed by homework completion time and the learning activity ratio. The contextual interaction therefore appears more prominently for continuous-score variation than for pass/fail discrimination, while the overall importance magnitudes remain modest.

Table 9. Top held-out permutation-importance features for Random Forest models using the full feature set. Importance is evaluated on held-out folds and summarized across folds

Task	Feature	Importance mean	Importance std.
Classification	Quiz activity after focused start (Q_α^s)	0.0061	0.0039
Classification	Homework completion time (C^s)	0.0058	0.0137
Classification	Afternoon activity count ($A_{afternoon}^s$)	0.0050	0.0093
Classification	Exercise-related activity (P_{ex}^s)	0.0049	0.0041
Classification	Afternoon focused-learning activity ($A_{\alpha,afternoon}^s$)	0.0042	0.0071
Regression	Activity–precipitation interaction (Z_{prcp}^s)	0.0555	0.0183
Regression	Homework completion time (C^s)	0.0226	0.0127
Regression	Learning activity ratio (R_L^s)	0.0189	0.0091
Regression	Night activity count (A_{night}^s)	0.0109	0.0039
Regression	Afternoon focused-learning activity ($A_{\alpha,afternoon}^s$)	0.0104	0.0113

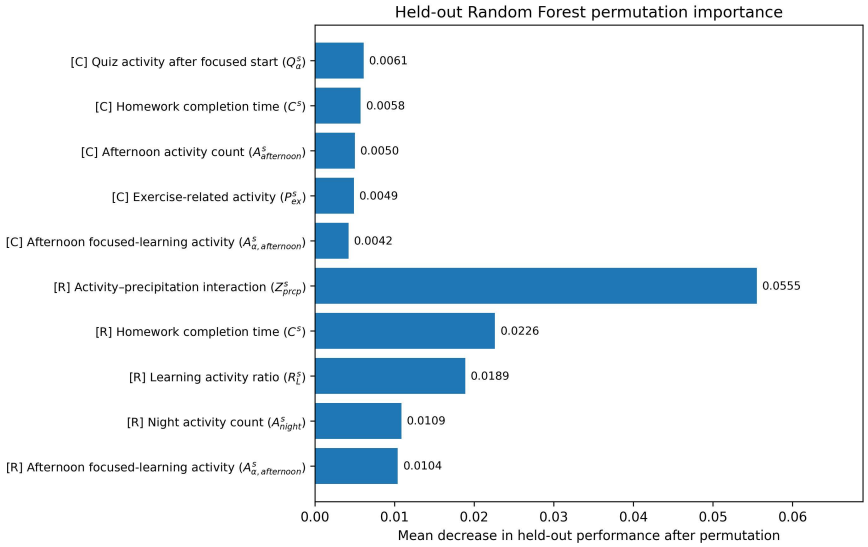

Fig. 3. Top held-out permutation-importance features for the Random Forest classification and regression models. Bars show the mean decrease in held-out predictive performance after feature permutation, and error bars show the standard deviation across cross-validation folds.

I. Theoretical Interpretation and Discussion

The extracted features can be interpreted within established learning theory frameworks. Behavioral features such as total activity A^s, focused learning activity A_α^s, and learning intensity I_D^s reflect the level of student engagement. In learning analytics, engagement is commonly treated as an important indicator of academic performance because it captures the extent to which students interact with learning materials and assessment-related resources [1, 2]. Session-based features and periodicity measures describe the temporal organization of learning and can be related to distributed practice and self-regulated learning [43, 50]. Assignment-related and quiz-related variables, such as C^s, V^s, Q^s, and Q_α^s, further capture active learning and formative assessment behavior [51, 52].

The environmental results require particular caution. Average environmental conditions $\bar{E}_m^s$ are contextual measurements shared by students in the same local setting and should not be interpreted as individual exposure. The leakage-safe permutation test provides marginal evidence for student-specific alignment ($p = 0.047$), and the activity–precipitation interaction has the largest mean held-out permutation importance in regression. Neither result establishes a causal environmental effect. A more limited interpretation is that the predictive meaning of focused LMS activity may vary with measured local context.

For example, the precipitation-related interaction $Z_{\text{prcp}}^s = A_\alpha^s \bar{E}_{\text{prcp}}^s$ combines the amount of focused learning activity with average precipitation over the same estimated interval. Its importance indicates a predictive association conditional on the behavioral measure, not that precipitation improves or worsens learning. Plausible mechanisms involving time use, commuting, indoor activity, comfort, or concentration remain hypotheses because the available data do not directly measure students' physical location, motivation, commuting behavior, or study environment. Consequently, the environmental component is best viewed as exploratory contextual enrichment that motivates further study with individual-level exposure measurements and larger datasets.

Overall, the results support three main interpretations. First, behavioral LMS features remain the primary source of predictive information. Second, environmental variables alone provide limited additional value. Third, interaction terms between behavioral and environmental variables may capture context-dependent learning patterns that are not visible from behavioral indicators alone. This supports the broader view that learning analytics models can benefit from contextual enrichment, provided that such effects are interpreted cautiously and validated on larger datasets.

J. Answers to the Research Questions

RQ1 is addressed by the proposed pipeline, which converts raw Moodle logs into interpretable temporal, behavioral, session-based, assessment-related, environmental, and interaction features while fitting data-dependent pruning only on training data. RQ2 is supported by the focused-learning

variables and by the sensitivity analysis: performance remains similar across $k \in \{3, 5, 7, 10\}$ and feature rankings are highly stable. RQ3 receives cautious support because environmental alignment yields a small numerical predictive increment and a marginal permutation result ($p = 0.047$), but the evidence is observational and does not imply causality. RQ4 is also supported only in a predictive and exploratory sense: interaction features contribute to the best regression configuration and the activity-precipitation interaction has the highest mean held-out regression importance, while paired feature-set gains are not statistically significant.

V. Limitations

This study is limited to one institution, one LMS platform, and two academic cohorts. Although the feature definitions and extraction procedure can be reimplemented in other LMS settings, the present feature rankings, model parameters, thresholds, and performance estimates should not be transferred to other institutions without local re-estimation and validation. Institutional teaching practices, course structure, assessment design, Moodle configuration, event-logging conventions, and student composition can all change the meaning and distribution of the engineered variables.

The environmental variables were measured for the local study environment at daily resolution and do not capture individual study locations or indoor conditions. The permutation result of $p = 0.047$ is close to the conventional threshold and should be treated as marginal, exploratory evidence rather than causal confirmation. The focused-window sensitivity analysis reduces concern about the specific choice of $k = 5$, since $k = 3, 5, 7, 10$ produced similar performance and stable rankings, but other courses may exhibit different temporal scales.

A further limitation is prediction timing. The current analysis uses records available up to the first colloquium and therefore does not validate a fixed intervention horizon. Prospective work should impose cutoffs such as 14, 7, and 3 days before assessment and recompute all features using only information available by each cutoff. The present study also lacks consistent measures of prior knowledge, motivation, guided and collaborative learning, individual study location, assignment quality, and academic-integrity indicators. These variables may provide complementary information if they can be collected consistently and ethically.

VI. Conclusion

This study presented an interpretable and reproducible framework for transforming raw Moodle activity logs into features for student performance prediction. The revised evaluation explicitly prevents test-fold information from entering zero-variance or correlation-based pruning. Behavioral features remain the primary source of predictive information, while environmental variables and interactions provide smaller contextual contributions. Paired five-fold comparisons do not show statistically significant incremental gains from the enriched feature sets. The environmental-alignment permutation test yields marginal evidence ($p = 0.047$), which is interpreted as a predictive association rather than a causal environmental effect.

With the primary five-day focused-learning window, the best Random Forest regression result reaches $R^2 = 0.149$ and RMSE = 3.327, while pass/fail ROC-AUC reaches 0.684. Cross-year classification reaches AUC = 0.707, whereas cross-year raw-score regression remains below the mean-baseline level with negative R^2 values. Sensitivity analysis across $k = 3, 5, 7, 10$ shows similar performance and feature-ranking correlations of approximately 0.971–0.990, indicating that the main representation is not dependent on a sharply tuned five-day value.

The practical interpretation remains deliberately limited. The observed predictive performance is moderate and does not justify automated grading or high-stakes decisions. Moreover, because the current design uses information through the assessment timestamp, it should be viewed as pre-assessment risk-identification evidence rather than a validated fixed-horizon early-warning system. Future work should evaluate explicit intervention lead times, larger multi-course and multi-institutional samples, additional LMS platforms, individual-level contextual measurements, and broader model families while retaining the leakage-safe and interpretable evaluation protocol introduced here.

Appendix A: Overview of Extracted Features

Table A1 provides a compact overview of the main extracted feature groups used in the proposed framework.

Table A1. Overview of extracted feature groups and representative variables

Feature group	Representative variables	Interpretation
General activity	A^s, A^s_{night}, $A^s_{morning}$, $A^s_{afternoon}$, $A^s_{evening}$	Overall LMS engagement and daily activity distribution
Learning window	t^s_α, D^s_α, H^s_α, A^s_α, $A^s_{w^*}$	Timing, intensity, and activity concentration of focused preparation
Session activity	S^s, S^s_p, μ^s_g, $\tilde{g}^s$, m^s_g	Frequency and regularity of LMS sessions
Assignment engagement	C^s, V^s	Assignment completion duration and post-submission review behavior
Quiz and exercise engagement	Q^s, Q^s_α, H^s_Q, P^s_{ex}	Overall quiz activity, quiz timing, and exercise-related course activity
Weekend activity	W^s	Learning activity during weekends
Environmental context	$\bar{E}^s_m$, $m \in \mathcal{M}$	Average weather and air-quality conditions during focused learning
Learning intensity	I^s_D, I^s_S, R^s_L	Normalized measures of learning intensity and activity concentration
Learning regularity	R^s_μ, $R^s_{\tilde{g}}$	Regularity of learning sessions
Interaction features	Z^s_m, $Z^s_{D,m}$, $Z^s_{S,m}$, $Z^s_{R,m}$	Joint behavioral-contextual predictive descriptors

Data and Code Availability

The implementation code, privacy-safe synthetic example data, and instructions for executing the complete analysis workflow are publicly available at https://github.com/dvlahek/moodle-focused-learning-feature-engineering. The repository includes synthetic Moodle event logs, synthetic assessment scores, synthetic environmental records, an extracted feature dataset derived

from those synthetic inputs, and an automated workflow test. The original student-level Moodle logs and assessment records cannot be made publicly available because of student privacy and institutional restrictions. The synthetic data contain no real participant records and are provided for executable workflow verification rather than reproduction of the protected study results. Further methodological details are available from the corresponding author upon reasonable request.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Dino Vlahek conducted the feature-engineering implementation, modeling, analysis, and manuscript preparation. Marija Pokos Lukinec and Dijana Oreški contributed to the study design, interpretation of the educational context, and manuscript revision. All authors approved the final version.

ACKNOWLEDGMENT

During the preparation of this work, ChatGPT (OpenAI, GPT-5.6 Sol) was used in selected parts of the manuscript to improve language and clarity. The authors reviewed and edited the content after using this tool and take full responsibility for the content of the publication. The authors acknowledge institutional support from the Faculty of Organization and Informatics, University of Zagreb. The broader research environment was supported by the Institutional Research Project through the National Recovery and Resilience Plan 2021–2026 under the project PiPi: Data infrastructure and augmented intelligence for sustainable development.

REFERENCES

[1] P. Leitner, M. Khalil, and M. Ebner, *Learning Analytics in Higher Education—A Literature Review*, Cham: Springer International Publishing, 2017, pp. 1–23.

[2] T. Susnjak, G. S. Ramaswami, and A. Mathrani, "Learning analytics dashboard: A tool for providing actionable insights to learners," *International Journal of Educational Technology in Higher Education*, vol. 19, no. 1, 12, 2022.

[3] H. Liu and H. Motoda, *Feature Extraction, Construction and Selection: A Data Mining Perspective*, Springer Science & Business Media, 1998.

[4] C. M. Bishop and N. M. Nasrabadi, *Pattern Recognition and Machine Learning*, Springer, 2006.

[5] H. Khosravi, A. Shibani, J. Jovanovic, Z. A. Pardos, and L. Yan, "Generative AI and learning analytics: Pushing boundaries, preserving principles," *Journal of Learning Analytics*, vol. 12, no. 1, pp. 1–11, Mar. 2025.

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

[7] S. Kotsiantis, C. Pierrakeas, and P. Pintelas, "Predicting students' performance in distance learning using machine learning techniques," *Applied Artificial Intelligence*, vol. 18, no. 5, pp. 411–426, 2004.

[8] S. Van Goidsenhoven, D. Bogdanova, G. Deeva, S. Broucke, J. Weerdt, and M. Snoeck, "Predicting student success in a blended learning environment," in *Proc. 10th International Conference on Learning Analytics & Knowledge*, 2020, pp. 17–25.

[9] Ş. Aydoğdu, "Predicting student final performance using artificial neural networks in online learning environments," *Education and Information Technologies*, vol. 25, no. 3, pp. 1913–1927, 2020.

[10] Y. Psaromiligkos, M. Orfanidou, C. Kytagias, and E. Zafeiri, "Mining log data for the analysis of learners' behaviour in web-based learning management systems," *Operational Research*, vol. 11, pp. 187–200, Aug. 2011.

[11] T. Purwoningsih, H. B. Santoso, and Z. A. Hasibuan, "Data analytics of students' profiles and activities in a full online learning context," in *Proc. 2020 Fifth International Conference on Informatics and Computing (ICIC)*, 2020, pp. 1–8.

[12] D. Vlahek, D. Oreški, and M. Pokos Lukinec, "Learning from the features we ignore: A critical perspective on feature engineering and the role of feature learning in learning analytics," *IEEE Access*, vol. 13, pp. 198252–198265, 2025.

[13] L. B. de Amorim, G. D. Cavalcanti, and R. M. Cruz, "The choice of scaling technique matters for classification performance," *Applied Soft Computing*, vol. 133, 109924, 2023.

[14] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Applied Soft Computing*, vol. 97, 105524, 2020.

[15] G. Pagallo, "Learning dnf by decision trees," in *Proc. 11th International Joint Conference on Artificial Intelligence*, vol. 1, 1989, pp. 639–644.

[16] M. Muharram and G. D. Smith, "Evolutionary constructive induction," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 11, pp. 1518–1528, 2005.

[17] H. Guo and A. K. Nandi, "Breast cancer diagnosis using genetic programming generated feature," in *Proc. IEEE Workshop on Machine Learning for Signal Processing*, 2005, pp. 215–220.

[18] C. Perlich and F. Provost, "Distribution-based aggregation for relational learning from identifier attributes," *Machine Learning*, vol. 62, pp. 62–105, 2005.

[19] S. Piramuthu and R. T. Sikora, "Iterative feature construction for improving inductive learning algorithms," *Expert Systems with Applications*, vol. 36, no. 2, pp. 3401–3406, 2009.

[20] D. Vlahek and D. Mongus, "An efficient iterative approach to explainable feature learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 5, pp. 2606–2618, 2023.

[21] M. S. Reza and J. Ma, "Ica and pca integrated feature extraction for classification," in *Proc. 2016 IEEE 13th International Conference on Signal Processing (ICSP)*, 2016, pp. 1083–1088.

[22] A. Bansal, K. Balaji, and Z. Lalani, "Temporal encoding strategies for energy time series prediction," arXiv preprint, arxiv:2503.15456, 2025.

[23] A. S. Al-Fahoum and A. A. Al-Fraihat, "Methods of eeg signal features extraction using linear analysis in frequency and time-frequency domains," *International Scholarly Research Notices*, vol. 2014, no. 1, 730218, 2014.

[24] Q. Gu, Z. Li, and J. Han, "Generalized fisher score for feature selection," in *Proc. 27th Conference on Uncertainty in Artificial Intelligence (UAI)*, 2012, pp. 266–273.

[25] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Proc. 18th International Conference on Neural Information Processing Systems*, 2005, pp. 507–514.

[26] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *Journal of Machine Learning Research*, vol. 3, pp. 1289–1305, 2003.

[27] M. Verleysen, F. Rossi, and D. François, "Advances in feature selection with mutual information," in *Similarity-Based Clustering: Recent Developments and Biomedical Applications*, M. Biehl, B. Hammer, M. Verleysen, and T. Villmann, Eds. Springer, 2009, pp. 52–69.

[28] L. Raileanu and K. Stoffel, "Theoretical comparison between the Gini index and information gain criteria," *Annals of Mathematics and Artificial Intelligence*, vol. 41, pp. 77–93, 2004.

[29] J. Gu, Z. Wang, J. Kuen, L. Ma, A. Shahroudy, B. Shuai, T. Liu, X. Wang, G. Wang, J. Cai, and T. Chen, "Recent advances in convolutional neural networks," *Pattern Recognition*, vol. 77, pp. 354–377, 2018.

[30] J. Li, K. Cheng, S. Wang, F. Morstatter, R. P. Trevino, J. Tang, and H. Liu, "Feature selection: A data perspective," *ACM Computing Surveys*, vol. 50, no. 6, pp. 1–45, 2017.

[31] M. A. Hall, "Correlation-based feature selection for machine learning," Ph.D. dissertation, University of Waikato, 1999.

[32] N. Kadoić and D. Oreški, "Analysis of student behavior and success based on logs in moodle," in *Proc. 2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2018, pp. 0654–0659.

[33] L. Leppänen, J. Leinonen, P. Ihantola, and A. Hellas, "Predicting academic success based on learning material usage," in *Proc. 18th Annual Conference on Information Technology Education (SIGITE)*, 2017, pp. 13–18.

[34] C. Taylor, K. Veeramachaneni, and U.-M. O'Reilly, "Likely to stop? predicting stopout in massive open online courses," arXiv preprint, arXiv:1408.3382, 2014.

[35] Z. Balogh and M. Kuchárik, "Predicting student grades based on their usage of LMS Moodle using petri nets," *Applied Sciences*, vol. 9, no. 20, 2019.
[36] F. Chen and Y. Cui, "Utilizing student time series behaviour in learning management systems for early prediction of course performance," *Journal of Learning Analytics*, vol. 7, no. 2, pp. 1–17, 2020.
[37] J. Gamulin, O. Gamulin, and D. Kermek, "Using fourier coefficients in time series analysis for student performance prediction in blended learning environments," *Expert Systems*, vol. 33, no. 2, pp. 189–200, 2016.
[38] E. Tiukhova, P. Vemuri, N. G. L. Flores, A. S. Islind, M. Óskarsdóttir, S. Poelmans, B. Baesens, and M. Snoeck, "Explainable learning analytics: Assessing the stability of student success prediction models by means of explainable AI," *Decision Support Systems*, vol. 182, 114229, 2024.
[39] A. Zanellati, S. P. Zingaro, and M. Gabbrielli, "Balancing performance and explainability in academic dropout prediction," *IEEE Transactions on Learning Technologies*, vol. 17, pp. 2086–2099, 2024.
[40] M. Saqr and S. López-Pernas, "Why explainable ai may not be enough: Predictions and mispredictions in decision making in education," *Smart Learning Environments*, vol. 11, 52, 2024.
[41] S. V. S. Gunasekara and M. Saarela, "Systematic literature review on explainable learning analytics and educational data mining," *IEEE Access*, vol. 13, pp. 214387–214408, 2025.
[42] L. P. Macfadyen and S. Dawson, "Mining lms data to develop an 'early warning system' for educators: A proof of concept," *Computers & Education*, vol. 54, no. 2, pp. 588–599, 2010.
[43] B. J. Zimmerman, "Becoming a self-regulated learner: An overview," *Theory into Practice*, vol. 41, no. 2, pp. 64–70, 2002.
[44] X. Zhang, X. Chen, and X. Zhang, "The impact of exposure to air pollution on cognitive performance," *Proc. Natl. Acad. Sci.*, vol. 115, no. 37, pp. 9193–9197, 2018.
[45] K. G. Low and J. M. Feissner, "Seasonal affective disorder in college students: Prevalence and latitude," *Journal of American College Health*, vol. 47, no. 3, pp. 135–137, 1998.
[46] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
[47] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, 2016, pp. 785–794.
[48] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.
[49] B. Alnasyan, M. Basheri, and M. Alassafi, "The power of deep learning techniques for predicting student performance in virtual learning environments: A systematic literature review," *Computers and Education: Artificial Intelligence*, vol. 6, 100231, 2024.
[50] N. J. Cepeda, H. Pashler, E. Vul, J. T. Wixted, and D. Rohrer, "Distributed practice in verbal recall tasks: A review and quantitative synthesis," *Psychological Bulletin*, vol. 132, no. 3, 354, 2006.
[51] P. Black and D. Wiliam, "Assessment and classroom learning," *Assessment in Education: Principles, Policy & Practice*, vol. 5, no. 1, pp. 7–74, 1998.
[52] S. Freeman, S. L. Eddy, M. McDonough, M. K. Smith, N. Okoroafor, H. Jordt, and M. P. Wenderoth, "Active learning increases student performance in science, engineering, and mathematics," *Proc. Natl. Acad. Sci.*, vol. 111, no. 23, pp. 8410–8415, 2014.

Copyright © 2026 by the authors. This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited (CC BY 4.0).