Student Dropout Predictor

Author: Ali Suhail
Student ID: 2605549
Date: 17th July 2025

July 23, 2025

Abstract :

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent porttitor arcu luctus, imperdiet urna iaculis, mattis eros. Pellentesque iaculis odio vel nisl ullamcorper, nec faucibus ipsum molestie. Sed dictum nisl non aliquet porttitor. Etiam vulputate arcu dignissim, finibus sem et, viverra nisl. Aenean luctus congue massa, ut laoreet metus ornare in. Nunc fermentum nisi imperdiet lectus tincidunt vestibulum at ac elit. Nulla mattis nisl eu malesuada suscipit. Aliquam arcu turpis, ultrices sed luctus ac, vehicula id metus. Morbi eu feugiat velit, et tempus augue. Proin ac mattis tortor. Donec tincidunt, ante rhoncus luctus semper, arcu lorem lobortis justo, nec convallis ante quam quis lectus. Aenean tincidunt sodales massa, et hendrerit tellus mattis ac. Sed non pretium nibh. Donec cursus maximus luctus. Vivamus lobortis eros et massa porta porttitor.

Index Terms: Keyword A, Keyword B, Keyword C.

1 Introduction (2)

Student dropout remains a significant issue in higher education, impacting not only students' academic success but also the financial health and reputation of universities. Early identification of students at risk of dropping out of their studies enables timely interventions, which can enhance retention and improve student outcomes. In recent years, ML has become a valuable approach for predicting at-risk students across various courses. These models utilise diverse data sources such as VLE interactions, continuous assessment results, and demographic details [1]. By analysing this information, predictive models can reveal which students are vulnerable and the reasons behind their struggles, enabling educators to offer targeted, personalised support [1]. Nonetheless, dropout prediction is challenging due to the complex interplay of demographic, academic, and behavioural factors. This project is motivated by the goal of creating a dependable and scalable ML-based dropout prediction system to assist educators and administrators in better understanding and mitigating dropout risks.

This project focuses on predicting student dropout by analysing data available up to a specified point within the duration of a module. Since the outcome is binary, indicating whether a student drops out or not, the task is framed as a classification problem. A variety of ML models will be developed and compared, with hyperparameter tuning applied to improve their performance. The model that demonstrates the highest effectiveness will be selected for final use.

A comprehensive ML pipeline will be developed specifically for predicting student dropout. The process will begin with exploratory data analysis to identify patterns and relationships within the data, including student demographics, academic history, and engagement with the VLE. Next, the data will be preprocessed by handling missing values, managing outliers, and encoding categorical variables. After preparation, the dataset will be split into training and testing sets to ensure an unbiased evaluation of model performance. Several machine learning models will then be implemented, covering traditional approaches such as LR, SVM, and RF, along with neural network models like the MLP Classifier. Each model will be optimised using hyperparameter tuning and evaluated using metrics such as accuracy, precision, recall, and F1-score. Based on this evaluation, the model with the best performance will be chosen to predict student dropout.

2 Background (5)

2.1 Literature Review

The rising dropout rates in higher education have become an increasing concern globally, with serious implications for students, educational institutions, and policymakers. Although greater access to university education has created a larger pool of graduates for the labour market, it has also resulted in a notable increase in the number of students leaving before completing their degrees [2]. According to the OECD (2019), dropout rates are increasing by an average of around 30% across many countries [3]. This highlights the need for effective strategies to identify and support students who are at risk of disengaging, while still maintaining academic standards despite growing enrolment figures.

In the United Kingdom, data from the Student Loans Company (SLC) highlights the issue, showing a 28% rise in university dropouts over five years. The number of students who took out loans but failed to complete their courses increased from 32,491 in 2018–19 to 41,630 in 2022–23 [4]. Mental health challenges have been identified as a major cause of early withdrawal [4]. These statistics emphasise the need for early intervention and predictive tools to help academic staff identify students who may require additional support. Previous research has shown that targeted academic measures, such as personalised emails and proactive tutor involvement, can reduce dropout rates by 11% in affected classes [5]. While the study acknowledged that factors like course design might also affect outcomes, it did not explore these in depth. Additionally, it pointed out that distance learning provides valuable opportunities to monitor and respond to student engagement.

Predicting student dropout is also important for managing academic resources and improving learning outcomes. Accurate predictions allow institutions to provide timely support, such as tutoring or customised learning pathways. They also enable better planning, such as adjusting teaching staff levels or identifying courses that may need revision. By implementing machine learning models that forecast dropout risk, universities can take data-

driven actions to reduce attrition, improve retention, and enhance the overall quality of education.

2.2 Objectives and Hypothesis

This project assumes that prediction accuracy improves as more data becomes available during the module, though dropout likelihood generally decreases over time. The focus is on early identification of at-risk students by uncovering key dropout-related features. The model will be designed for use around the module midpoint or earlier as an early warning system. By analysing dropout indicators at different stages, the project aims to find out when predictions are most accurate and understand performance variations. Beyond accuracy, it will identify major dropout factors to inform academic support. The chosen model will support institutions in improving retention and success. A table of additional assumptions and hypotheses will accompany the analysis.

In addition to building an accurate prediction system, this project aims to uncover the underlying factors driving student dropout at various points in the module. These insights will help educators implement targeted interventions earlier, where they are likely to be most effective. The goal is to support proactive, data-informed decision-making that improves retention and enhances student success.

3 Methods (10)

execution - i.e., what you did

3.1 Data Description

The dataset used in this project is the publicly available and anonymised Open University Learning Analytics Dataset (OULAD). It includes information on courses, students, and their interactions with the Virtual Learning Environment (VLE) across seven selected modules. These modules are delivered in two presentation periods: February and October, labelled as "B" and "J" respectively. The dataset is organised into multiple CSV files, each representing a table linked through unique identifiers. Further details can be found in [6].

After cleaning and preprocessing, the dataset comprises 27,984 students, with 19 selected features. These features are detailed in Table 1.

The following section outlines the feature engineering process used to derive aggregated variables such as `total_clicks`, `days_active_norm`, `weighted_score`, `banked_rate`, `late_rate`, and `fail_rate`.

3.2 Preprocessing and Feature Engineering

The initial stage involved cleaning and preparing the dataset. This process included handling missing values, eliminating duplicate entries, and standardising data formats. An exploratory analysis was conducted to examine the distribution of key features such as gender, age group, and final result, as well as to detect any class imbalance. These observations informed subsequent steps such as scaling, encoding, and transformation, ensuring that no feature disproportionately influenced the model. Following the cleaning process, feature engineering was carried out by merging relevant datasets and combining variables into a structured format appropriate for modelling. For example, the following time-series dataset represents a student's assessment activity:

Table 1
Description of dataset features used for student dropout prediction.

Feature	Description
`code_module`	A categorical variable and an abbreviated code identifying the module.
`code_presentation`	Also a categorical variable and an abbreviated code for the specific presentation of the module (e.g., "B" for February, "J" for October).
`date_registration`	A numerical feature for the student's registration date relative to the module start (in days).
module_presentation length	Numerical feature for the module presentation duration in days.
`gender`	Student's gender: Male = 1, Female = 0.
`region`	A Categorical variable for geographic region where the student resided during the module.
`highest_education`	A Categorical feature for the highest qualification held by the student at the time of enrollment.
`imd_band`	A categorical field for the socio-economic band based on the Index of Multiple Deprivation (IMD) of the student's residence.
`age_band`	Student's age group.
`num_of_prev_attempts`	Number of times the student has previously attempted the same module.
`studied_credits`	Total credits of all modules the student is enrolled in concurrently.
`disability`	Indicates whether the student has declared a disability.
`final_result`	Final outcome in the module: Distinction/Pass/Fail = 1, Withdrawal = 0. (Target variable)
`total_clicks`	Total number of interactions (clicks) with the VLE within the selected timeframe.
`days_active_norm`	Total number of days the student was active on the VLE, normalised for the selected timeframe.
`weighted_score`	Student's average weighted score for assessments submitted during the timeframe.
`banked_rate`	Proportion of assessment scores carried over from previous module presentations.
`late_rate`	Proportion of assessments or exams submitted after the deadline.
`fail_rate`	Proportion of assessments or exams the student failed.

In this example, `final_result` is the target variable, representing whether a student passed, failed, withdrew, or achieved distinction. According to the data specifications in [6] a score < 40 is a fail. Because the data is time-dependent, we need to aggregate it to make it usable for modelling. For instance, we can summarise a student's performance as:

As illustrated in Table 3, the `weighted_score` is calculated by multiplying each assessment score by its respective weight and then dividing the sum by the total weight (which is 100 in this case). The `late_rate` is determined by comparing the submission date with the due date; any submission made after the due date is considered late. The late rate represents the fraction of late submissions relative to the total number of submissions, ranging from 0 to 1. Lastly, the `fail_rate` is computed by counting the number of assessments with scores below 40 (including missing scores, which are treated as failures or non-submissions) divided by the

Table 2
Sample student assessment records showing assessment type, weight, score, and submission timing.

Code Module	Stu. ID	Assess ID	Assess Type	Weight	Score	Date Due	Date Subm.	Final Result
AAA	0	0	TMA	20	35	5	6	Pass
AAA	0	1	TMA	20	60	20	19	Pass
AAA	0	2	CMA	20	75	50	55	Pass
AAA	0	3	Exam	40	85	100	100	Pass

Table 3
Example of a feature-engineered student record with aggregated performance metrics.

Code Module	Student ID	Weighted Score	Late Rate	Fail Rate	Final Result
AAA	0	68	0.50	0.25	Pass

total assessments submitted. For the example in Table 2, the student failed one assessment (assessment ID 0 with a score of 35), resulting in a fail rate of 0.25 as shown in Table 3.

3.3 Time-Based Feature Limiting for Early Prediction

In real-world applications, full course histories are rarely available when attempting to predict student dropout early. Making predictions at the end of a course is typically too late for effective intervention. To address this, a timeline-based feature limitation approach is introduced. This involves restricting the available data to a specific point in time (e.g., the midpoint of a module) to emulate early-stage prediction. For instance, if the total module duration is 100 days, the dataset can be truncated to include only events occurring up to day 50. The aggregated data based on this restriction is shown in Table 4:

Table 4
Aggregated student performance metrics derived from data available up to the module midpoint.

Code Module	Student ID	Weighted Score	Late Rate	Fail Rate	Final Result
AAA	0	57	0.67	0.33	Pass

Table 4 illustrates student performance metrics based only on data available up to the midpoint of the module, in contrast to the full-course view over 100 days (Table 3). At the 50% mark, higher late and fail rates are observed, largely because the final exam has not yet occurred and is therefore not included in the data. This table represents the type of truncated data used for training ML models on thousands of student records to uncover patterns associated with dropout. Such intermediate datasets often reflect lower performance due to incomplete assessment coverage and can highlight early warning signs, such as high failure rates, frequent late submissions, or poor early engagement. These models are thus trained to recognise early behavioural indicators that correlate with eventual dropout risk.

3.4 Processing Dataset

Following feature engineering and the merging of relevant tables, the dataset undergoes final cleaning to prepare it for predictive modelling. This stage includes removing non-informative attributes, addressing missing values, and verifying that the dataset aligns with the intended prediction goals. For example, the `date_unregistration` feature is excluded, as it cannot be known during early-stage prediction. If used, it would introduce data leakage, providing the model with information that would not be available at prediction time, leading to overly optimistic and misleading performance.

3.4.1 Imputation Strategy for Missing Values Missing values in various features are handled through context-aware imputation. The following table summarises the imputation methods and the rationale behind them:

Table 5
Imputation strategies for selected features, based on student engagement and demographic relevance.

Feature	Imputation Method	Justification
`date_registration`	Median replacement	Most missing values are for withdrawn students. The median provides a reasonable neutral estimate.
`total_clicks`	Replace with 0	Students who fail or withdraw often have no VLE interaction. Zero engagement is a logical substitute.
`banked_rate`	Replace with 0	Missing values typically indicate withdrawn or failing students. The feature has low coverage and impact, so 0 is a practical default.
`weighted_score`	Replace with 0	Non-submission is represented by missing values. A 0 score reflects no submission, as per the dataset specification.
`late_rate`	Replace with 1	No submission implies full lateness. A value of 1 reflects total disengagement with deadlines.
`fail_rate`	Replace with 1	Missing values imply assessment failure. A value of 1 reflects complete non-completion.
`imd_band`	Bayesian Ridge Regression	IMD is a critical socio-demographic feature. Missing values are predicted using age, education, and region for contextual accuracy.

A total of 4,609 students who withdrew either before the module commenced or within the first 19 days were excluded. This cutoff was chosen because most modules start assessments after day 19, and these students generally exhibit no VLE activity or assessment records. Including them would introduce noise without contributing valuable information. Furthermore, early withdrawal data would not be available when predicting dropout during the early or mid-phase of the module, so retaining such records would reduce the model's realism and practical applicability.

3.4.2 Temporal Segmentation of Dataset for Dropout Prediction An automated data processing script was developed that allows the user to specify the portion of the module timeline to include. Using this, four datasets corresponding to different time points within the module were created for training, testing, and evaluation: Early, Midpoint, Late, and Full.

The Early dataset includes student data up to the first 25% of the module's duration. For instance, in a 100-day module, this would cover only the first 25 days. Data beyond this point is excluded to prevent leakage and compel the models to identify early indicators of potential dropout. The Midpoint dataset covers 50% of the module duration, Late covers 75%, and Full contains the complete data for the entire module. While the Late and Full datasets will be used primarily for exploratory data analysis and serve as benchmarks, the main emphasis is placed on enhancing dropout prediction performance using the Early and Midpoint datasets.

The choice of 25%, 50%, 75%, and 100% time points reflects a balanced progression through the module, providing meaningful intervals for prediction. Selecting a very early cutoff, such as

10%, would make dropout prediction challenging due to insufficient VLE interaction and assessment data. At such an early stage, the model would have to rely heavily on demographic information alone, which is less ideal. By using 25%, 50%, 75%, and 100%, the model has more opportunity to learn from a combination of demographic data, VLE activity, and assessment performance. The 25% mark is particularly suitable for early prediction since, by this point, most modules have already involved some assessments and VLE activities, offering a solid foundation for detecting early signs of potential dropout.

3.4.3 Train/Test Split To prevent data leakage and ensure unbiased evaluation, the dataset is split into training and testing sets before any detailed exploratory data analysis (EDA). An 80/20 split is applied, with 80% of the data used for training (including all EDA) and 20% reserved as a test set for final model evaluation. The split is stratified by course module to maintain proportional representation of each module in both subsets.

3.4.4 Exploratory Data Analysis (EDA) Following the split, comprehensive EDA is conducted exclusively on the training set. This process uncovers insights about the engineered features, examines value distributions, and evaluates feature relevance. Relationships between features are explored, outliers identified, and correlation matrices generated to assess associations among features and with the target variable. The findings guide decisions on scaling continuous variables, encoding categorical features, and removing irrelevant or redundant attributes such as `id_student` that do not contribute to predictive modelling.

Table 6
Feature preprocessing methods applied before model training

Feature	Scaling/Encoding Method
code_module	One-Hot Encoding
code_presentation	One-Hot Encoding
date_registration	Standard Scaler
module_presentation_length	Standard Scaler
gender	One-Hot Encoding
region	One-Hot Encoding
highest_education	Standard Scaler
age_band	Standard Scaler
num_of_prev_attempts	Standard Scaler
studied_credits	Standard Scaler
days_active_norm	Standard Scaler
disability	One-Hot Encoding
total_clicks	Standard Scaler
weighted_score	Standard Scaler
banked_rate	Standard Scaler
late_rate	Standard Scaler
fail_rate	Standard Scaler
final_result (target)	Distinction/Pass/Fail=1; Withdrawn=0

3.4.5 Scaling and Encoding the Dataset As detailed in Table 6, One-Hot Encoding is applied to categorical variables such as `gender` and `disability`, transforming each category into its own binary feature. This step is essential since models like logistic regression and SVM require numerical inputs and cannot handle raw categorical strings.

Continuous numerical features are normalised using Standard Scaling, which adjusts values to have a mean of 0 and a standard deviation of 1 [CITE]. This is particularly important for models sensitive to feature magnitudes, including MLP, LR, and SVM, which rely on gradient-based optimisation. For instance, `studied_credits` may vary between 30 and 600, whereas `num_of_prev_attempts` ranges from 0 to 5. Without scaling, features with larger ranges could disproportionately influence the model [CITE].

Ordinal categorical variables such as `age_band`, which have a meaningful order but are not inherently numeric, are first label-encoded (e.g., "0–35" $\rightarrow$ 0, "35–55" $\rightarrow$ 1, "55<=" $\rightarrow$ 2) and subsequently scaled using StandardScaler to maintain consistency with other numerical features.

The target variable `final_result` is converted into a binary outcome: students who passed, failed, or obtained distinction are labelled as 1 (indicating module completion), while those who withdrew are labelled as 0, aligning with the objective of predicting dropout versus continuation.

—-

3.5 Machine Learning Models

As this is a classification task, we employ Logistic Regression (LR), Support Vector Machine (SVM), Multilayer Perceptron Classifier (MLP), and Random Forest (RF) models. These models are trained on the training dataset, then evaluated on the test set using two key metrics: accuracy and macro-averaged F1 score, which balances precision and recall across all classes.

3.5.1 Model Selection Justification The models selected for this task: Logistic Regression (LR), Support Vector Machines (SVM), Multi-Layer Perceptron (MLP), and Random Forest (RF), represent a diverse set of classification strategies. Each model brings distinct advantages suitable for the challenges of student dropout prediction:

Logistic Regression (LR): LR serves as a strong baseline due to its simplicity, interpretability, and computational efficiency [CITE]. It establishes a benchmark for evaluating more complex models and provides insight into the linear separability of the data. Moreover, its probabilistic outputs are useful for setting risk thresholds when identifying students at risk of dropping out [CITE].

Support Vector Machines (SVM): SVMs are effective for high-dimensional data and can perform well even in the presence of class imbalance, particularly when using techniques such as class weighting [CITE]. By employing kernel functions like RBF or polynomial kernels, SVMs can model non-linear decision boundaries, making them capable of capturing more intricate patterns in student behaviour [CITE].

Multi-Layer Perceptron (MLP): The MLP, a feedforward neural network, is chosen for its ability to learn non-linear relationships and complex feature interactions that linear models might overlook [CITE]. It is especially valuable when underlying patterns in the data are not easily separable or explicitly encoded. Although MLPs require careful hyperparameter tuning and are more computationally intensive, they can uncover hidden structures in student engagement or performance indicators [CITE].

Random Forest (RF): RF is a robust ensemble learning method that combines multiple decision trees to improve generalisation

and reduce overfitting [CITE]. It is particularly effective at handling non-linear relationships, missing values, and mixed data types (numerical and categorical) [CITE]. Random Forest also provides feature importance metrics, offering interpretability and insight into which student characteristics most strongly influence dropout risk. Its inherent bagging mechanism makes it less sensitive to noise and outliers, enhancing predictive stability across diverse student profiles.

3.5.2 Rationale for Using a Diverse Set of Models We selected models from distinct algorithmic categories, linear (LR), kernel-based (SVM), neural (MLP), and ensemble tree-based (RF), to capture a wide spectrum of learning behaviours and model capacities. This diversity allows us to:

- Compare the effectiveness of linear versus non-linear models in predicting student dropout.
- Evaluate how model performance varies under different data conditions (e.g., early vs full datasets).
- Examine trade-offs between predictive performance and model interpretability.
- Assess each model's ability to generalise to unseen data.

The primary objective is to identify students who are likely to withdraw accurately. Accordingly, greater emphasis is placed on recall (and precision) for the dropout class, as correctly flagging at-risk students is essential for enabling timely interventions and academic support.

All models are tuned using GridSearchCV for hyperparameter optimisation, paired with 5-fold cross-validation to ensure stable and generalisable performance estimates. Each model is then evaluated across four temporal subsets of the dataset: early, midpoint, late, and full, to examine how predictive performance changes throughout a module. The early and midpoint datasets are of particular interest, as they represent periods where intervention is most effective. In contrast, the late and full datasets serve as benchmarks for assessing the maximum achievable predictive performance.

References

[1] M. R. Marcolino, T. R. Porto, T. T. Primo, *et al.*, "Student dropout prediction through machine learning optimization: Insights from moodle log data," *Scientific Reports*, vol. 15, p. 9840, 2025. DOI: 10.1038/s41598-025-93918-1. [Online]. Available: https://doi.org/10.1038/s41598-025-93918-1.

[2] Á. Kocsis and G. Molnár, "Factors influencing academic performance and dropout rates in higher education," *Oxford Review of Education*, vol. 51, no. 3, pp. 414–432, 2024. DOI: 10.1080/03054985.2024.2316616. [Online]. Available: https://doi.org/10.1080/03054985.2024.2316616.

[3] OECD, *Education at a Glance 2019: OECD Indicators*. Paris: OECD Publishing, 2019. DOI: 10.1787/f8d7880d-en. [Online]. Available: https://doi.org/10.1787/f8d7880d-en.

[4] J. Bryson. "University dropout rates reach new high, figures suggest." BBC News. (Sep. 28, 2023), [Online]. Available: https://www.bbc.co.uk/news/education-66940041.

[5] C. Foster and P. Francis, "A systematic review on the deployment and effectiveness of data analytics in higher education to improve student outcomes," *Assessment & Evaluation in Higher Education*, vol. 45, no. 6, pp. 822–841, 2019. DOI: 10.1080/02602938.2019.1696945. [Online]. Available: https://doi.org/10.1080/02602938.2019.1696945.

[6] J. Kuzilek, M. Hlosta, and Z. Zdrahal, "Open university learning analytics dataset," *Scientific Data*, vol. 4, p. 170 171, 2017. DOI: 10.1038/sdata.2017.171. [Online]. Available: https://doi.org/10.1038/sdata.2017.171.