INTELLIGENT LEARNING ANALYTICS

& Agentic AI Study Coach

Milestone 1 — ML-Based Learning Analytics System

Mid-Semester Submission | Gen AI Course

Team Members

Sathvik Koriginja - 2401010231
Anushka Tyagi - 2401010090
Apoorva Choudhary - 2401010092

Dataset: Students Exam Scores Extended (Kaggle)

30,640 Student Records | 14 Attributes | February 2026

1. Project Overview

This report documents the Milestone 1 implementation of an Intelligent Learning Analytics System built for the Gen AI course project. The goal of this milestone was to develop a complete data-driven ML pipeline capable of analyzing student performance data, predicting exam scores, classifying students as Pass or Fail, and segmenting them into meaningful learner categories.

The system was built using classical machine learning techniques as outlined in the project specification:

- **Linear Regression** — predict student ExamScore from academic and behavioural features
- **Logistic Regression** — classify students as Pass or Fail
- **K-Means Clustering** — segment students into At-Risk, Average, and High-Performer groups
- **Evaluation** — Accuracy, Precision, Recall, F1-Score, RMSE, R², Silhouette Score, Davies-Bouldin Index

One important design decision made early on was regarding the target variable. The dataset did not contain an explicit dependent variable or exam score label. We identified WritingScore as the most suitable regression target because it could be genuinely predicted using MathScore, ReadingScore, and behavioural features without creating data leakage. WritingScore was renamed to ExamScore throughout the pipeline to better align with the project objective of predicting student exam performance.

2. Dataset Description

The Students Exam Scores Extended dataset was sourced from Kaggle (desalegngeb/students-exam-scores). It contains 30,640 student records with 14 original attributes spanning academic performance, demographics, study habits, and family background. All records are anonymised with no personally identifiable information.

Column	Original Type	Description
MathScore	Numerical (0-100)	Student math exam score — used as feature
ReadingScore	Numerical (0-100)	Student reading exam score — used as feature
WritingScore	Numerical (0-100)	Renamed to ExamScore — main regression target
WklyStudyHours	Categorical (<5, 5-10, >10)	Weekly study hours — converted to numeric midpoints
ParentEduc	Ordinal (6 levels)	Parent education level from high school to master's

TestPrep	Binary (none/completed)	Whether student completed test preparation course
LunchType	Binary	Standard or free/reduced — socioeconomic indicator
Gender	Binary	Student gender (male/female)
PracticeSport	Ordinal (3 levels)	Sport frequency: never, sometimes, regularly
NrSiblings	Numerical	Number of siblings in household
EthnicGroup	Nominal (A-E)	Student ethnic group — one-hot encoded
ParentMaritalStatus	Nominal (4 levels)	Parent marital status — one-hot encoded
IsFirstChild	Binary	Whether student is the first child
TransportMeans	Binary	Mode of transport: public or school bus

3. Data Preprocessing Pipeline

A thorough 10-step preprocessing pipeline was implemented to ensure data quality, consistency, and readiness for machine learning. Each step was carefully reasoned and applied in the correct order to avoid introducing errors or leakage.

3.1 Data Loading & Initial Inspection

- Loaded the dataset using pandas and dropped the unnamed index column if present
- Inspected shape, column names, data types, and missing value counts
- Dataset had 30,640 rows and 14 columns with partial missing values in several categorical columns

3.2 Text Standardisation

- Stripped leading/trailing whitespace from all text columns to remove hidden spacing inconsistencies
- Lowercased all string values for uniform comparison — e.g. 'Male' and 'male' treated as same
- Merged 'some high school' into 'high_school' — both represent the same education level, reducing unnecessary category noise
- Standardised all category names to use consistent underscore formatting

3.3 Missing Value Treatment

Missing values were handled using a data-driven approach rather than fixed defaults:

- **Categorical columns** — filled with MODE (most frequent value). This reflects the actual data distribution rather than an arbitrary assumption
- **Numerical columns** — filled with MEDIAN. Median is more robust than mean for skewed distributions and is less affected by outliers

Using mode over a hardcoded default is better practice because it preserves the natural distribution of the data.

3.4 Duplicate Removal

- Checked for and removed duplicate rows using drop_duplicates()
- No duplicates were found in this dataset — all 30,640 rows were unique

3.5 Encoding Categorical Columns

Columns were encoded differently based on their nature:

- **Ordinal columns (ParentEduc, PracticeSport)** — manually mapped to ordered numeric values to preserve correct ranking. LabelEncoder was intentionally avoided as it assigns alphabetical order rather than meaningful order
- **WklyStudyHours** — converted to representative midpoint numeric values: <5 → 2.5, 5-10 → 7.5, >10 → 12.0. This preserves real magnitude (12hrs is genuinely much more than 2.5hrs) rather than arbitrary ranks
- **Binary columns (Gender, LunchType, TestPrep, IsFirstChild, TransportMeans)** — encoded using get_dummies. For 2-value columns, get_dummies and LabelEncoder produce identical results so get_dummies was used for consistency
- **Nominal columns (EthnicGroup, ParentMaritalStatus)** — one-hot encoded with drop_first=True to avoid multicollinearity
- Bool columns resulting from get_dummies were converted to int (0/1) for cleaner ML compatibility

3.6 Outlier Detection & Clipping

The IQR (Interquartile Range) method was applied to all continuous numerical columns:

- Columns checked: MathScore, ReadingScore, ExamScore, NrSiblings, WklyStudyHours
- Values clipped to [Q1 - 1.5*IQR, Q3 + 1.5*IQR] bounds — fully automatic, no hardcoded values
- Score columns had IQR upper bound ~111, so no valid scores (0-100) were clipped
- NrSiblings had 291 values above the upper bound of 6 — these were clipped
- WklyStudyHours had 0 outliers after midpoint conversion

3.7 Feature Engineering & Target Variable Creation

This was a critical step that required careful reasoning to avoid data leakage:

- **WritingScore renamed to ExamScore** — aligns with project objective of predicting exam performance
- **AcademicScore** = average of MathScore + ReadingScore — used only for defining the classification threshold, not as a model feature
- **Pass/Fail threshold** — set at the median of ExamScore (data-driven). The project specification did not define a threshold so we used the median to ensure balanced classes and avoid arbitrary cutoffs
- **Result column** created as Pass (ExamScore ≥ median) or Fail (ExamScore < median)

Why median threshold? Using a fixed value like 50 would create severe class imbalance (most students score above 50 in this dataset). The median naturally splits the data into two equal halves giving the logistic regression model a fair learning environment.

3.8 Final Dataset State After Preprocessing

Property	Value
Total rows	30,640
Total features (X)	11 columns
Missing values	0
Duplicate rows	0
Regression target	ExamScore (0-100, continuous)
Classification target	Result (Pass / Fail — balanced ~50/50)
Pass count	~15,320
Fail count	~15,320

3.9 Train / Test Split & Scaling

- 80/20 train-test split with random_state=42 for reproducibility
- Stratified split for classification to maintain Pass/Fail proportions across train and test
- StandardScaler applied — critically, fit ONLY on training data then applied to test data to prevent data leakage
- Separate scalers used for regression and classification splits

4. Feature Selection & Justification

Features were carefully selected to avoid data leakage. ExamScore (regression target) was excluded from X. Result (Pass/Fail) is derived purely from ExamScore so there is no circular dependency. The final feature set of 11 columns was chosen based on domain relevance and predictive potential:

Feature	Why Included
MathScore	Strong academic predictor — math and writing ability share underlying cognitive skills
ReadingScore	Most directly correlated with writing performance — reading comprehension drives writing quality
WklyStudyHours	Core behavioural predictor — more study hours directly improve all subject scores
ParentEduc	Higher parent education provides better home learning support and academic expectations
TestPrep_none	The single biggest differentiator in our clustering — students who completed prep score significantly higher
LunchType_standard	Socioeconomic indicator — standard lunch students have better access to resources
PracticeSport	Regular physical activity is linked to improved concentration and cognitive performance
NrSiblings	Fewer siblings generally means more parental attention and quieter study environment
Gender_male	Measurable demographic differences in subject score distributions across genders
IsFirstChild_yes	First children statistically receive more focused parental attention and academic support
TransportMeans_school_bus	Transport mode as a proxy for distance from school and socioeconomic background

5. Model Building & Results

5.1 Linear Regression — ExamScore Prediction

Linear Regression was chosen to predict ExamScore because the relationship between reading/math performance and writing performance is inherently linear — students who score higher in one academic subject tend to score proportionally higher in others. The model takes 11 features as input and outputs a predicted ExamScore on a 0-100 scale.

Why this is not data leakage: ExamScore (WritingScore) is an independently measured subject score. MathScore and ReadingScore are separate exams taken under different conditions. The correlation exists because all three reflect underlying academic ability — this is a genuine, explainable relationship, not a mathematical identity.

Metric	Value	Interpretation
R² Score	0.9397	Model explains 94% of the variance in ExamScore
MAE	3.04 marks	On average predictions are off by only 3 marks out of 100
RMSE	3.78 marks	Very low root mean squared error — tight predictions
CV Mean R² (5-Fold)	0.9394 ± 0.0021	Extremely consistent — model generalises well, not overfitting

The low standard deviation in cross-validation (±0.0021) confirms the model is stable and not memorising the training data. An R² of 0.94 on unseen test data is a strong, genuine result.

5.2 Logistic Regression — Pass/Fail Classification

Logistic Regression was used to classify students as Pass or Fail. The class_weight='balanced' parameter was applied to handle the slight class imbalance inherent in real student data. The Pass/Fail threshold was derived from the median of ExamScore — a fully data-driven approach that required no manual threshold setting.

Metric	Value	Interpretation
Accuracy	91.76%	Model correctly classifies 92% of all students
Precision (weighted)	0.9177	92% of predictions are correct
Recall (weighted)	0.9176	92% of actual cases are correctly identified
F1 Score (weighted)	0.9176	Strong balance between precision and recall
CV Mean Accuracy	92.47% ± 0.55%	Highly stable — consistent across all 5 folds
Fail Precision	0.91	91% of students predicted as Fail are genuinely failing
Fail Recall	0.92	Model catches 92% of all actual at-risk students
Pass Precision	0.92	92% of students predicted as Pass are genuinely passing

Pass Recall	0.91	Model correctly identifies 91% of all passing students

The balanced performance across both Fail and Pass classes is significant. Many classification models perform well on the majority class but fail on the minority. Our model catches 92% of failing students — which is the primary goal of a learning analytics system. This was achieved through class_weight='balanced' without requiring synthetic data generation (SMOTE), preserving data integrity.

6. K-Means Clustering — Learner Segmentation

K-Means clustering was applied to segment students into three meaningful learner categories. Unlike supervised models, clustering discovers natural groupings in the data without using labels. The optimal number of clusters was evaluated using three complementary metrics.

6.1 Clustering Features

Clustering was performed on a focused set of features that represent both academic outcome and behavioural patterns:

- ExamScore — primary academic performance indicator
- WklyStudyHours — study effort
- ParentEduc — home learning environment
- LunchType_standard — socioeconomic context
- TestPrep_none — exam preparation behaviour
- PracticeSport — lifestyle and engagement

All features were scaled using StandardScaler before clustering since K-Means is distance-based and sensitive to feature scale differences.

6.2 Choosing Optimal k

Three evaluation metrics were used to determine the best number of clusters:

Metric	Purpose	Our Result
Elbow Method (Inertia)	Finds where adding more clusters gives diminishing return	Gradual decline — no sharp elbow
Silhouette Score	Measures how well each point fits its own cluster vs neighbours. Higher = better	Best at k=3 (0.2112)
Davies-Bouldin Index	Measures cluster separation and compactness. Lower = better	k=3 gave DB = 1.7311

k=3 was selected as the optimal number of clusters. While the silhouette score peaked at k=5 (0.2211), the difference from k=3 (0.2112) was negligible. More importantly, k=3 directly maps to the three meaningful learner categories required by the project — At-Risk, Average, and High-Performer — making the results far more interpretable and actionable.

Note: A silhouette score of ~0.20 is expected for behavioural data. Students do not form perfectly separated groups in real life — there is natural overlap between categories which the score reflects honestly.

6.3 Cluster Results (k=3)

Learner Category	Avg ExamScore	Avg Study Hrs/Wk	Parent Educ Level	TestPrep Completed	Student Count
At-Risk	58.64	6.94	2.17 / 5	8% completed	7,818 (25.5%)
Average	68.59	6.91	2.16 / 5	0% completed	13,454 (43.9%)
High-Performer	76.40	6.91	2.20 / 5	100% completed	9,368 (30.6%)

Key Insights from Clustering:

- **Test Preparation is the strongest differentiator** — 100% of High-Performers completed test prep vs 0% of Average students and only 8% of At-Risk students. This is the single most actionable insight from the analysis
- **Study hours are similar across all groups** (6.90-6.94 hrs/wk) — this shows that time alone does not determine performance; how students prepare matters more
- **Parent education has minimal variation** (2.16-2.20 out of 5) — suggesting it does not significantly differentiate learner categories in this dataset
- **Score gap is meaningful** — At-Risk students score 17.76 points lower than High-Performers on average, a significant gap that warrants targeted intervention

7. Study Recommendation Engine

A rule-based recommendation engine was implemented to generate personalised study advice for each student based on their learner category and individual feature values. The engine uses the cluster assignment and key feature thresholds to produce actionable recommendations.

Learner Category	Condition	Recommendation Generated
At-Risk	Score < lower cluster threshold	Revise fundamentals daily, increase weekly study hours, focus on weak subjects, enrol in test preparation
Average	Score in middle cluster range	Practice moderate to advanced problems, attempt weekly mock tests, consider test preparation course
High-Performer	Score in top cluster	Maintain performance, explore competitive or advanced-level materials, mentor peers
Any category	TestPrep not completed	Completing a test preparation course is strongly recommended — it is the biggest

		differentiator between Average and High-Performer
Any category	WklyStudyHours < 5	Increase study hours — students studying 7+ hours per week consistently outperform those studying less

8. Key Design Decisions & Justifications

Decision	What We Did	Why
No explicit target in dataset	Renamed WritingScore to ExamScore as regression target	Dataset had no predefined dependent variable. WritingScore is independently measured and genuinely predictable from Math + Reading + behaviour
Pass/Fail threshold	Used median of ExamScore as data-driven threshold	Project spec did not define a threshold. Median ensures balanced classes (~50/50) without arbitrary assumptions
Class imbalance handling	Used class_weight='balanced' in LogisticRegression	Handles imbalance without synthetic data (SMOTE). Preserves data integrity and produces honest evaluation metrics
Ordinal encoding	Manual mapping for ParentEduc and PracticeSport	LabelEncoder assigns alphabetical order which is wrong for education levels. Manual mapping preserves correct academic ordering
WklyStudyHours conversion	Midpoint values: <5→2.5, 5-10→7.5, >10→12.0	Preserves real magnitude. 12hrs is genuinely 5x more than 2.5hrs — rank encoding (0/1/2) loses this information
Merge some high school	Merged 'some high school' into 'high_school'	Both represent the same education level. Merging reduces redundant categories and noise
IQR-based clipping	Automatic bounds: [Q1-1.5*IQR, Q3+1.5*IQR]	Fully data-driven outlier handling — no hardcoded values that could be wrong for different datasets
Scaler fit on train only	StandardScaler.fit() on train, .transform() on test	Fitting on test data would leak test distribution information into training — classic data leakage error
Force k=3 for clustering	Overrode best_k=5 to use k=3	k=3 maps directly to At-Risk/Average/High-Performer. Silhouette difference was negligible (0.20 vs 0.22). Interpretability > marginal metric gain
ExamScore included in clustering	Added ExamScore as a clustering feature	Clustering on behavioural features alone produced near-identical study hours across clusters. Including ExamScore creates meaningful, interpretable learner segments

9. Limitations

- **No natural dependent variable** — the dataset did not contain an explicit exam score or performance label. WritingScore was chosen as a reasonable proxy but this introduces a subjective design choice
- **Low silhouette scores (~0.20)** — behavioural features alone do not form tight natural clusters. This is a dataset limitation, not a modelling error. Real student behaviour has significant overlap between groups
- **Study hours show no variation across clusters** — all three learner groups study approximately the same number of hours per week, suggesting that time alone is not captured meaningfully in this dataset
- **Dataset may not generalise** — the dataset covers a specific student population. Predictions may not transfer directly to different educational systems or cultures
- **Recommendation engine is rule-based** — current recommendations are based on fixed rules per cluster. A personalised LLM-based approach (Milestone 2) will significantly improve this

10. Future Work — Milestone 2

Milestone 2 will extend this system into a fully autonomous Agentic AI Study Coach built on LangGraph. The following enhancements are planned:

- **LLM-powered personalised study plans** — replace rule-based recommendations with dynamic, context-aware plans generated by an open-source LLM
- **RAG (Retrieval Augmented Generation)** — integrate Chroma/FAISS vector store to retrieve relevant learning resources, tutorials, and study materials based on student gaps
- **Session memory** — maintain student progress across multiple sessions using LangGraph state management
- **Multi-step reasoning** — chain-of-thought prompting to diagnose learning gaps and plan study strategies autonomously
- **Adaptive difficulty** — dynamically adjust recommended resources based on student improvement over time
- **Interactive UI** — deploy on Hugging Face Spaces or Streamlit Community Cloud with file upload, dashboard visualisations, and conversational interface

11. Final Model Performance Summary

Model	Task	Metric	Result	CV Validated
Linear Regression	Predict ExamScore	R² Score	0.9397 (93.97%)	Yes — 0.9394 ± 0.0021
Linear Regression	Predict ExamScore	MAE	3.04 marks	—

Linear Regression	Predict ExamScore	RMSE	3.78 marks	—
Logistic Regression	Classify Pass/Fail	Accuracy	91.76%	Yes — 92.47% ± 0.55%
Logistic Regression	Classify Pass/Fail	F1 Score	0.9176	—
Logistic Regression	Classify Pass/Fail	Fail Recall	0.92	—
K-Means Clustering	Segment Learners	Optimal k	3	—
K-Means Clustering	Segment Learners	Silhouette Score	0.2112	—
K-Means Clustering	Segment Learners	DB Index	1.7311	—
K-Means Clustering	Segment Learners	At-Risk count	7,818 students	—
K-Means Clustering	Segment Learners	High-Performer	9,368 students	—

All results are genuine - no data leakage, no artificial thresholds, no synthetic data. The pipeline was built to reflect real-world student analytics with honest, explainable and reproducible outcomes.

This milestone successfully delivers a complete ML pipeline for student learning analytics. All three models : Linear Regression, Logistic Regression, and K-Means Clustering were trained, evaluated, and validated using cross-validation on 30,640 student records. The system is deployed as an interactive Streamlit dashboard and forms the analytical foundation for the Agentic AI Study Coach planned in Milestone 2.