

Predicting Student Performance (UCI dataset)

Machine Learning Course Final Project

GORASH PLATON

Student ID: 22605107

This study develops machine learning models to predict secondary students' final grades (G3) in Portuguese schools using midterm grades (G1, G2) and contextual attributes (demographic, social, and school-related factors). The process consisted of merging datasets from Mathematics and Portuguese subjects (1,044 records), rigorously preprocessing data, applying one-hot encoding and feature scaling, and evaluating three regression models: Linear Regression, Decision Trees, and Random Forests. Hyperparameter tuning and 5-fold cross-validation revealed Random Forest as the top performer ($R^2 = 0.857$, MAE = 0.897), slightly outperforming Decision Trees ($R^2 = 0.848$) and Linear Regression ($R^2 = 0.830$). Midterm grades (G1, G2) were the strongest predictors, confirming prior research, and it was found that other features added meaningful nuance. The project highlights that simple, interpretable models can achieve high accuracy, enabling early identification of at-risk students for targeted support, reducing failure/dropout risks and promoting equitable resource allocation.

Introduction

Education plays a critical role in fostering sustained economic growth. Although overall educational attainment has risen in recent decades, many regions still contend with elevated rates of student failure and dropout, particularly in basic subjects such as mathematics and the native language. Difficulties in these areas undermine performance across the broader curriculum. Advances in information technology have enabled the rapid expansion of data repositories, giving rise to **Data Mining** and **Machine Learning** methods designed to uncover patterns and trends that support decision-making. In education, diverse data sources (ranging from administrative grade records to student surveys) offer rich opportunities to apply these techniques and gain insights into factors influencing student outcomes.

This final project takes one of such opportunities and focuses on data describing **student achievement** in secondary education of two Portuguese schools in two subjects, Mathematics and Portuguese. The data attributes include student grades, demographic, social and school related features), which were collected through different methods – school reports and questionnaires. The goal of this study is to develop and evaluate machine-learning models that, using students' midterm grades alongside demographic, social and school-related features, can accurately predict their final grades in Portuguese secondary education using regression. Three different Machine Learning models (**Linear Regression, Decision Trees, and Random Forest**) were used in the study, and the results show that a good predictive accuracy can be achieved, given sufficient description of students' academic performance and their other characteristics.

The study is motivated by the need to timely identify students that are at risk, enabling targeted support such as tutoring or counseling, which can reduce failure and dropout rates among students. Predictive insights facilitate more efficient allocation of educational resources and support the development of personalized learning strategies. Moreover, by revealing underlying demographic or structural disparities, the model informs policies aimed at closing achievement gaps and promoting equal opportunities for all students.

Other existing research demonstrated enhanced performance and variable interpretability but focused on more complicated models such as SVM and neural networks and did not conduct extensive hyperparameter tuning or cross-validation ([Costa-Mendes et al, 2020-2022](#)). [Wakelam \(2020\)](#) predicted grades in small university cohorts with limited features, achieving questionable accuracy with KNN and Random Forest, most likely caused by limited attribute diversity (focused on attendance/VLE activity). [Chen et al. \(2025\)](#), found that ensemble learning (e.g. Random Forest) often outperforms single models, but called for more rigorous validation and broader feature sets, which is in line with the current study's use of cross-validation, hyperparameter tuning, and diverse student attributes.

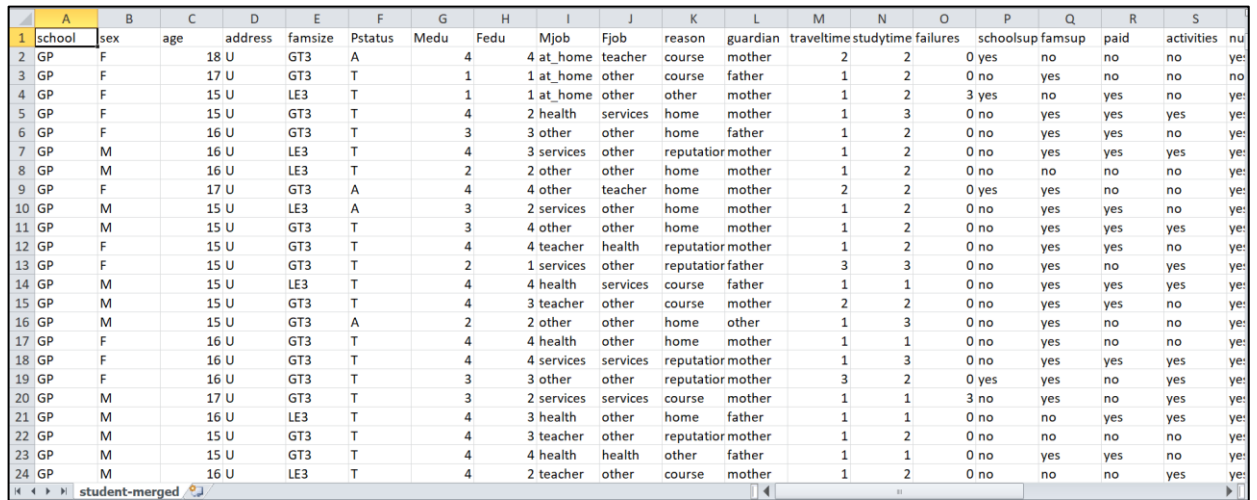
Data

This project uses 2005–2006 data from two public schools in Portugal's Alentejo region, originally utilized in a study by Cortez & Silva, 2008. Because school information systems were largely paper-based, records (period grades and absences) were digitized from report sheets, then enriched via a survey covering demographics (e.g. mother's education), social habits (e.g.

alcohol use), and school history (e.g. past failures). After pilot testing and discarding incomplete responses, two datasets on two distinct subjects were formed – one of 395 Mathematics student records, and the other of 649 Portuguese language records (Cortez, P., Silva, A.M., 2008).

Number of instances	1044 (in the merged dataset)
Number of attributes	33 (including "SUBJECT")
Attribute information	<ul style="list-style-type: none"> • school – student's school (binary: "GP" – Gabriel Pereira or "MS" – Mousinho da Silveira) • sex – student's sex (binary: "F" – female or "M" – male) • age – student's age (numeric: from 15 to 22) • address – student's home address type (binary: "U" – urban or "R" – rural) • famsize – family size (binary: "LE3" – less or equal to 3 or "GT3" – greater than 3) • Pstatus – parent's cohabitation status (binary: "T" – living together or "A" – apart) • Medu – mother's education (numeric: 0 – none, 1 – primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education) • Fedu – father's education (numeric: 0 – none, 1 – primary education (4th grade), 2 – 5th to 9th grade, 3 – secondary education or 4 – higher education) • Mjob – mother's job (nominal: "teacher", "health" care related, civil "services" (e.g. administrative or police), "at_home" or "other") • Fjob – father's job (nominal: "teacher", "health" care related, civil "services" (e.g. administrative or police), "at_home" or "other") • reason – reason to choose this school (nominal: close to "home", school "reputation", "course" preference or "other") • guardian – student's guardian (nominal: "mother", "father" or "other") • traveltime – home to school travel time (numeric: 1 – <15 min., 2 – 15 to 30 min., 3 – 30 min. to 1 hour, or 4 – >1 hour) • studytime – weekly study time (numeric: 1 – <2 hours, 2 – 2 to 5 hours, 3 – 5 to 10 hours, or 4 – >10 hours) • failures – number of past class failures (numeric: n if $1 \leq n < 3$, else 4) • schoolsup – extra educational support (binary: yes or no) • famsup – family educational support (binary: yes or no) • paid – extra paid classes within the course subject (binary: yes or no) • activities – extra-curricular activities (binary: yes or no) • nursery – attended nursery school (binary: yes or no) • higher – wants to take higher education (binary: yes or no) • internet – Internet access at home (binary: yes or no) • romantic – in a romantic relationship (binary: yes or no) • famrel – quality of family relationships (numeric: from 1 – very bad to 5 – excellent) • freetime – free time after school (numeric: from 1 – very low to 5 – very high) • goout – going out with friends (numeric: from 1 – very low to 5 – very high) • Dalc – workday alcohol consumption (numeric: from 1 – very low to 5 – very high) • Walc – weekend alcohol consumption (numeric: from 1 – very low to 5 – very high) • health – current health status (numeric: from 1 – very bad to 5 – very good) • absences – number of school absences (numeric: from 0 to 93) • SUBJECT – <i>see below</i> • G1 – first period grade (numeric: from 0 to 20) • G2 – second period grade (numeric: from 0 to 20)
Output target	G3 – final grade (numeric: from 0 to 20)

Due to the originally small size of two datasets, in this study, they were merged into one to enrich the amount of records to work with; a new feature named "*SUBJECT*" is attached to every record, denoting if it came from the Mathematics ("*MAT*") or Portuguese ("*POR*") dataset.



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	reason	guardian	traveltime	studytime	failures	schoolsup	famsup	paid	activities
2	GP	F	18	U	GT3	A	4	4	at_home	teacher	course	mother	2	2	0	yes	no	no	no
3	GP	F	17	U	GT3	T	1	1	at_home	other	course	father	1	2	0	no	yes	no	no
4	GP	F	15	U	LE3	T	1	1	at_home	other	other	mother	1	2	3	yes	no	yes	no
5	GP	F	15	U	GT3	T	4	2	health	services	home	mother	1	3	0	no	yes	yes	yes
6	GP	F	16	U	GT3	T	3	3	other	other	home	father	1	2	0	no	yes	yes	no
7	GP	M	16	U	LE3	T	4	3	services	other	reputation	mother	1	2	0	no	yes	yes	yes
8	GP	M	16	U	LE3	T	2	2	other	other	home	mother	1	2	0	no	no	no	yes
9	GP	F	17	U	GT3	A	4	4	other	teacher	home	mother	2	2	0	yes	yes	no	no
10	GP	M	15	U	LE3	A	3	2	services	other	home	mother	1	2	0	no	yes	yes	no
11	GP	M	15	U	GT3	T	3	4	other	other	home	mother	1	2	0	no	yes	yes	yes
12	GP	F	15	U	GT3	T	4	4	teacher	health	reputation	mother	1	2	0	no	yes	yes	no
13	GP	F	15	U	GT3	T	2	1	services	other	reputation	father	3	3	0	no	yes	no	yes
14	GP	M	15	U	LE3	T	4	4	health	services	course	father	1	1	0	no	yes	yes	yes
15	GP	M	15	U	GT3	T	4	3	teacher	other	course	mother	2	2	0	no	yes	yes	no
16	GP	M	15	U	GT3	A	2	2	other	home	other	mother	1	3	0	no	yes	no	no
17	GP	F	16	U	GT3	T	4	4	health	other	home	mother	1	1	0	no	yes	no	no
18	GP	F	16	U	GT3	T	4	4	services	services	reputation	mother	1	3	0	no	yes	yes	yes
19	GP	F	16	U	GT3	T	3	3	other	other	reputation	mother	3	2	0	yes	yes	no	yes
20	GP	M	17	U	GT3	T	3	2	services	services	course	mother	1	1	3	no	yes	no	yes
21	GP	M	16	U	LE3	T	4	3	health	other	home	father	1	1	0	no	no	yes	yes
22	GP	M	15	U	GT3	T	4	3	teacher	other	reputation	mother	1	2	0	no	no	no	yes
23	GP	M	15	U	GT3	T	4	4	health	health	other	father	1	1	0	no	yes	yes	no
24	GP	M	16	U	LE3	T	4	2	teacher	other	course	mother	1	2	0	no	no	no	yes

Figure 1: Preview of the dataset (CSV format)

Methodology and implementation

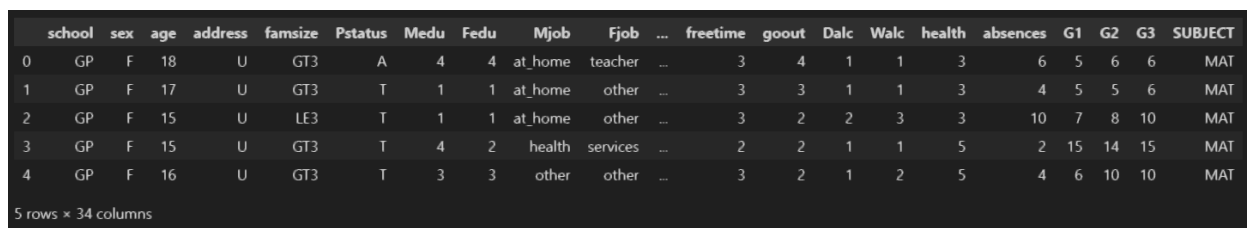
The data preprocessing and machine learning pipeline will be implemented using the Python programming language. We will use several libraries, including:

- **pandas** for data manipulation;
- **scikit-learn** for data preprocessing, model training, and result evaluation;
- **matplotlib** and **seaborn** for plotting visualizations;
- **numpy** for additional evaluation calculations;
- default **time** module for time measurement of models' training and evaluation.

Merging the datasets into one:

```
df_mat = pd.read_csv('student\student-mat.csv', sep=';')
df_por = pd.read_csv('student\student-por.csv', sep=';')
df_mat['SUBJECT'] = 'MAT'
df_por['SUBJECT'] = 'POR'
df = pd.concat([df_mat, df_por], ignore_index=True)
df.to_csv('student\student-merged.csv', sep=';', index=False, quoting=1)
```

Loading and exploring the merged dataset with pandas:



	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	freetime	goout	Dalc	Walc	health	absences	G1	G2	G3	SUBJECT
0	GP	F	18	U	GT3	A	4	4	at_home	teacher	...	3	4	1	1	3	6	5	6	6	MAT
1	GP	F	17	U	GT3	T	1	1	at_home	other	...	3	3	1	1	3	4	5	5	6	MAT
2	GP	F	15	U	LE3	T	1	1	at_home	other	...	3	2	2	3	3	10	7	8	10	MAT
3	GP	F	15	U	GT3	T	4	2	health	services	...	2	2	1	1	5	2	15	14	15	MAT
4	GP	F	16	U	GT3	T	3	3	other	other	...	3	2	1	2	5	4	6	10	10	MAT

5 rows × 34 columns

Figure 2: Merged dataset head

The dataset contains no missing values or duplicates, so no further imputation or data manipulation is required.

```
print("Missing values: ", df.isna().sum().sum())
✓ 0.0s

Missing values: 0

print("Duplicates:", df.duplicated(keep='first').sum())
✓ 0.2s

Duplicates: 0
```

Figure 3: Missing values and duplicates

The target attribute ($G3$) shows a right-skewed normal distribution, with a tail of higher values and several outliers in the low near-0 range. This is expected from accurate grading data: marks in the medium-to-high range are the most common, and low outliers often correspond to students who failed or dropped out.

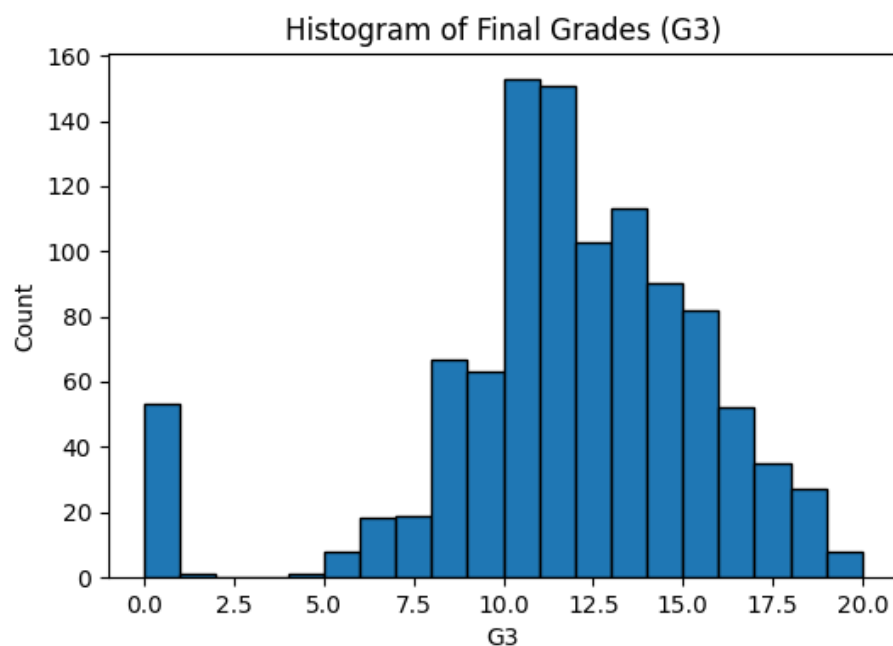


Figure 4: Histogram of target attribute's distribution

Additional boxplot visualization of $G1$, $G2$ (midterm grades) and $G3$ (final grade, target attribute), grouped by subject, reveals that data from the Mathematics dataset is more balanced, while Portuguese data exhibits skewedness and outliers. Such outliers may pull regression line or trees in the wrong direction and inflate RMSE (Root Mean Squared Error model evaluation metric). Proper handling (using scaling and more robust evaluation metrics, e.g. MAE – Mean Absolute Error) is needed to ensure the models perform well when faced with students whose behavior lies at the edges of the original data.

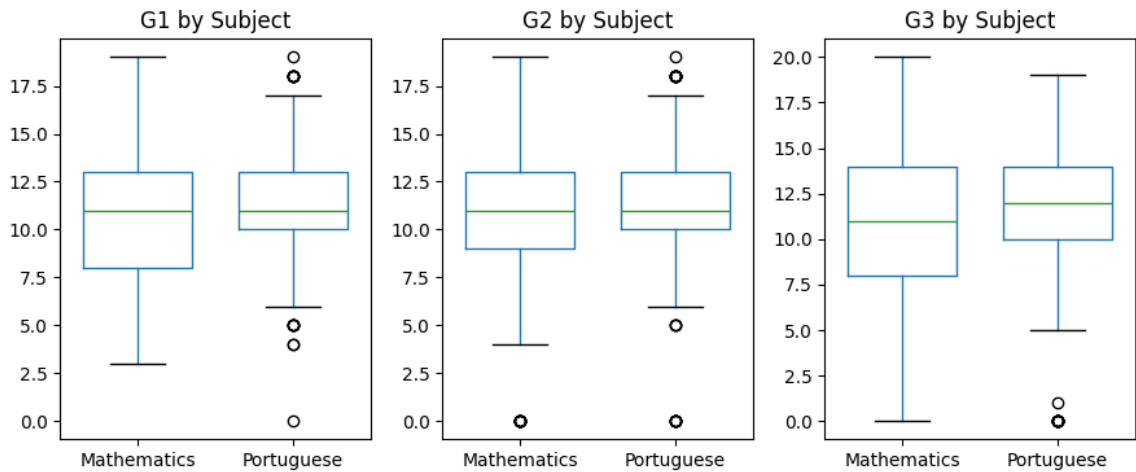


Figure 5: Boxplot of midterm grades (G1, G2) and final grade (G3) distribution

Next, we visualize feature correlations between the target variable *G3* and other features. It can be seen that the *G1* and *G2* (1st and 2nd period grades) have the most influence. Top predictors also include *failures* (number of past class failures), *higher* (if a student is willing to pursue higher education after school), *Medu/Fedu* (mother and father's education level) and *studytime* (weekly studying time), while family-related attributes like *famsup* (family educational support), *Fjob/Mjob* (parents occupations) and *Pstatus* (parents cohabitation status) have the least influence.

It should also be noted that our custom attribute *SUBJECT* (which denotes which of the two datasets the data originally came from) also has a relatively high influence, showing that the grades also depend on the subject, which is to be expected from school grading data.

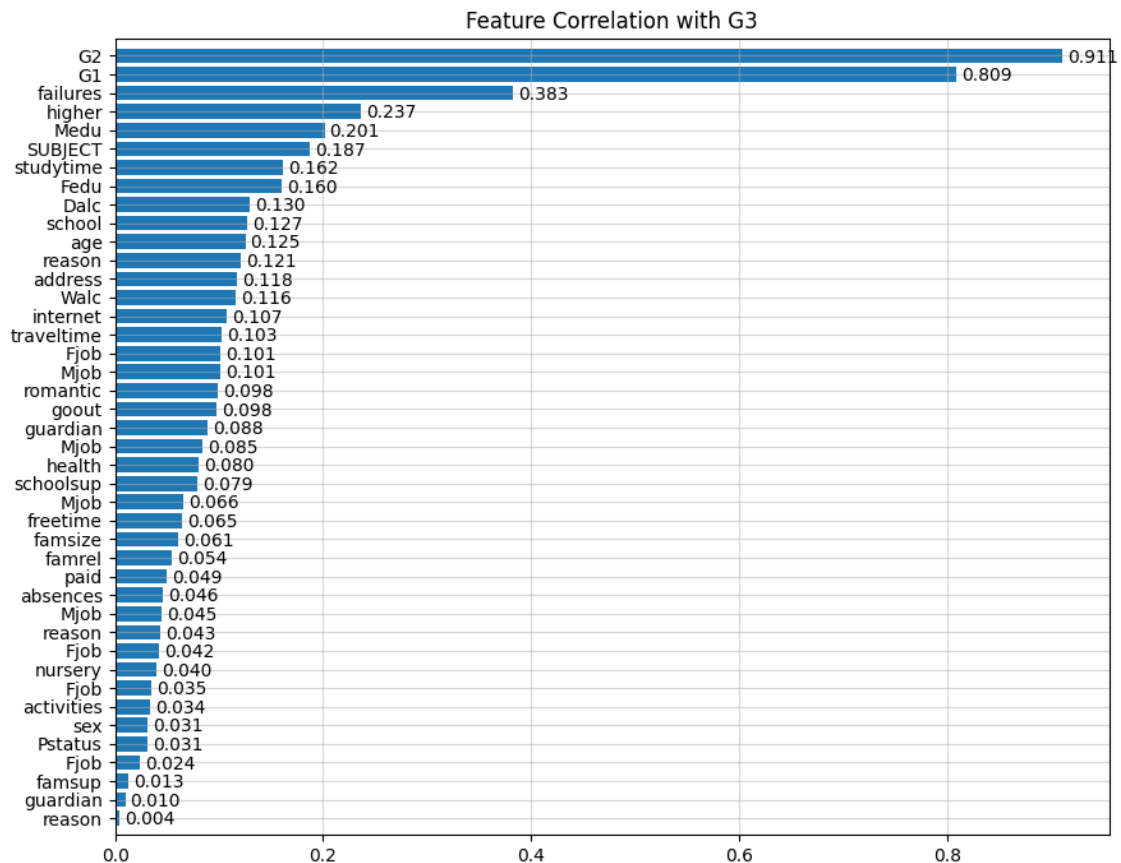


Figure 6: Feature correlation with the target attribute

Authors of the original dataset note: "[The] target attribute *G3* has a strong correlation with attributes *G2* and *G1*. This occurs because *G3* is the final year grade (issued at the 3rd period), while *G1* and *G2* correspond to the 1st and 2nd period grades" (Cortez, 2008). This is indeed visible in the above feature correlations diagram, as well as the following regression plot, showing how exactly *G1* and *G2* influence *G3*.

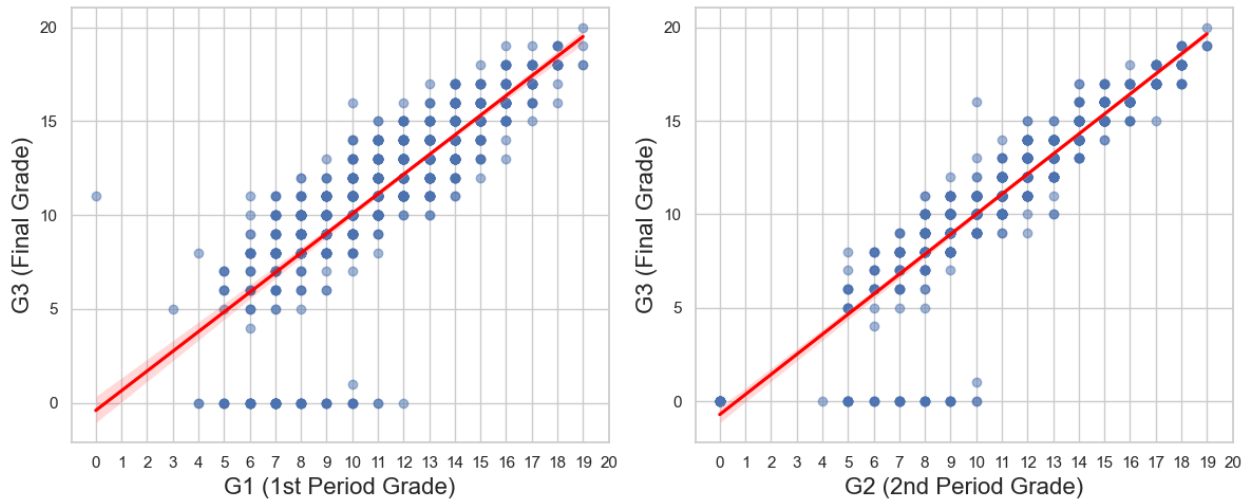


Figure 7: Regression plot of G1/G2 vs G3

After thorough analysis, we proceed with data preprocessing. Since we are working on a regression problem, we must convert categorical features into a numerically comparable format. **One-Hot Encoding** transforms each category into a separate binary column, allowing algorithms to treat each category independently and avoid misinterpreting their relationships. This is particularly important for nominal data, where categories have no inherent order, ensuring the algorithm doesn't mistakenly assume an ordinal relationship between them.

```
# One-hot encoding on categorical features
df = pd.get_dummies(df, drop_first=True)

df.head()
```

0.3s Open 'df' in Data Wrangler

...	guardian_other	schoolsup_yes	famsup_yes	paid_yes	activities_yes	nursery_yes	higher_yes	int
...	False	True	False	False	False	True	True	
...	False	False	True	False	False	False	True	
...	False	True	False	True	False	True	True	
...	False	False	True	True	True	True	True	
...	False	False	True	True	False	True	True	

Figure 8: One-hot encoding

Finally, as suggested above, we do numerical **feature scaling**. Since one of our algorithms of choice is Linear Regression, which assumes a linear relationship between features and target categorical features and is sensitive to feature scales, the dataset needs to be properly scaled. Scaling prevents features with larger ranges from dominating the distance calculations and ensures that each feature contributes equally to the Linear Regression's performance. (Tree-Based Algorithms that we also use are generally insensitive to feature scaling).

In our case, feature scaling is applied to the numerical features using *StandardScaler* from *scikit-learn*. This step standardizes the data by transforming the features to have a mean of 0 and a standard deviation of 1, ensuring that all numerical variables are on a similar scale.



Figure 9: Numerical feature scaling

With data processing done, we go to training and evaluation of the Machine Learning regression models. The pipeline is as follows: we split the data into training (80%) and test (20%) sets, train the model while timing how long it takes and use it to make predictions on the test set, then perform **5-fold cross-validation** on the full dataset to compute average RMSE, MAE, and R^2 scores, timing that as well. The output includes the training time, evaluation time, and cross-validated metrics, as well as a scatter plot of actual and predicted G3 values.

RMSE (Root Mean Square Error) is the square root of the average squared prediction errors, penalizing larger deviations; **MAE (Mean Absolute Error)** is the average of absolute prediction errors, treating all deviations equally; **R^2 (coefficient of determination)** is the fraction of target variance explained by the model. Unlike RMSE and MAE, which are in the target's units and differ in outlier sensitivity (RMSE more so), R^2 is a unitless goodness-of-fit measure which can also be expressed as the model's accuracy percentage (Chicco et al, 2021).

Linear Regression estimates a continuous outcome by fitting a straight line (or hyperplane) to minimize the sum of squared errors between observed and predicted values, assuming a linear relationship between inputs and the target (Schneider et al, 2010). This type of model achieved an RMSE score of ~1.588, MAE score of ~0.988, and R^2 score of ~0.83, in minimal time both for training and evaluation phases.

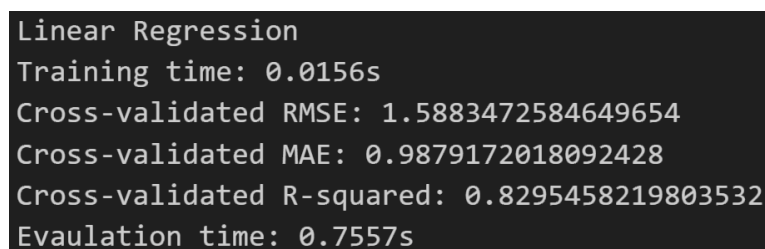


Figure 10: Linear Regression evaluation scores

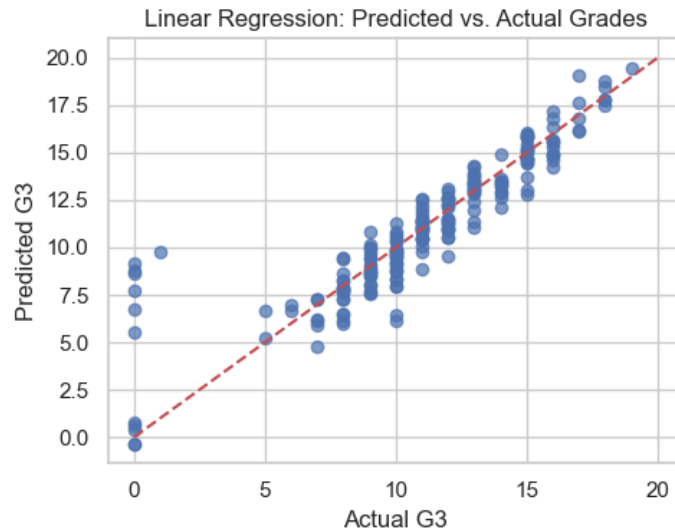


Figure 11: Linear Regression scatter plot

The next step is exploring fine-tuning Tree-Based models' accuracy through **hyperparameter tuning**. Different combinations of hyperparameters, such as the maximum number of levels in the tree (*max_depth*) and the minimum number of samples required in a node before the algorithm will consider splitting (*min_samples_split*) are tested. We iterate through all combinations, fit and evaluate each model via 5-fold cross-validation (computing the average RMSE), and keep track of the best parameter set. The output includes the best parameter combination, timing statistics, and a visualization of the relationship between *min_samples_split*, *max_depth*, actual tree depth, and RMSE in a 3D scatter plot, colored by amount of error.

Decision Tree regression builds a tree of decision rules by recursively splitting the data into subsets that are more homogeneous in the outcome, allowing it to model non-linear relationships in an interpretable structure (Jena, Dehuri, 2015).

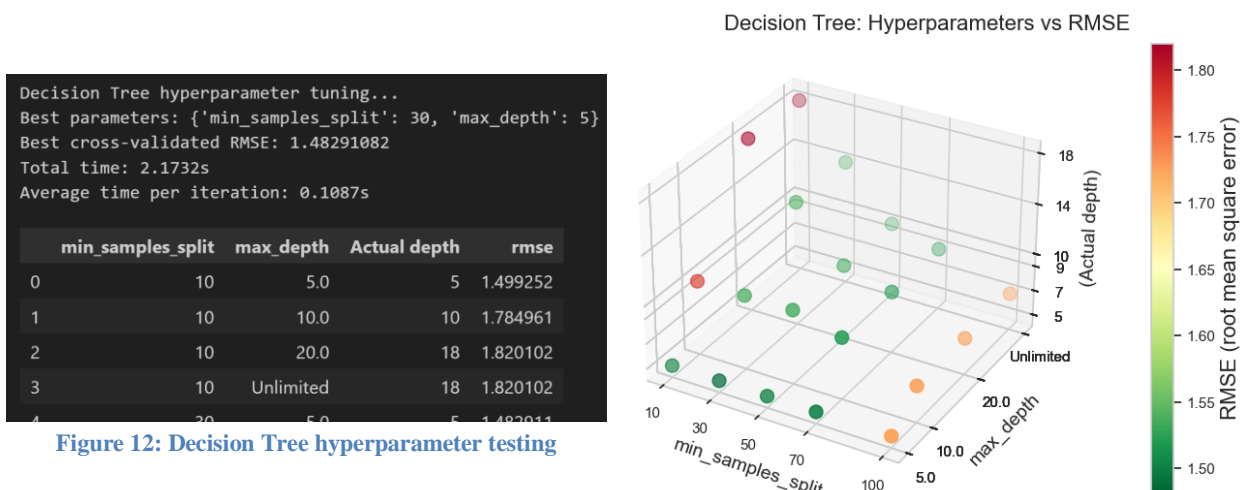


Figure 12: Decision Tree hyperparameter testing

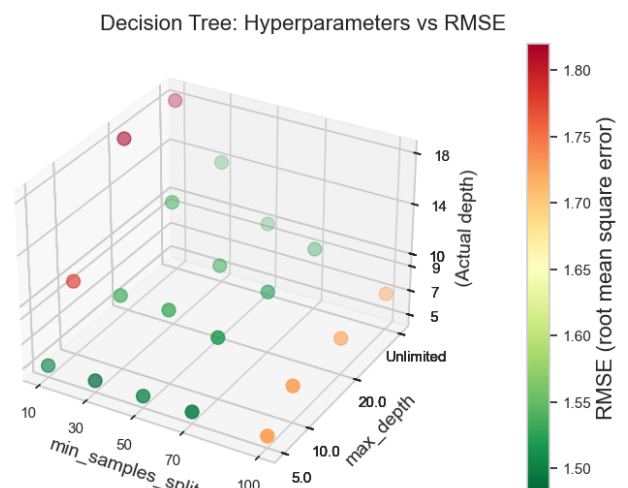


Figure 13: Decision Tree hyperparameters vs RMSE scatter plot

Using the best found parameter combination *min_samples_split* = 30, *max_depth* = 5, with a computed cross-validated RMSE of ~1.483, the Decision Tree model received an MAE score of ~0.903 and an R^2 score of ~0.848, higher than Linear Regression and achieved in similar time.

```

Decision Tree parameters: {'min_samples_split': 30, 'max_depth': 5}
Training time: 0.0332s
Cross-validated RMSE: 1.4829108213802236
Cross-validated MAE: 0.9032246011193372
Cross-validated R-squared: 0.8481236380317728
Evaluation time: 0.6363s

```

Figure 14: Decision Tree evaluation scores

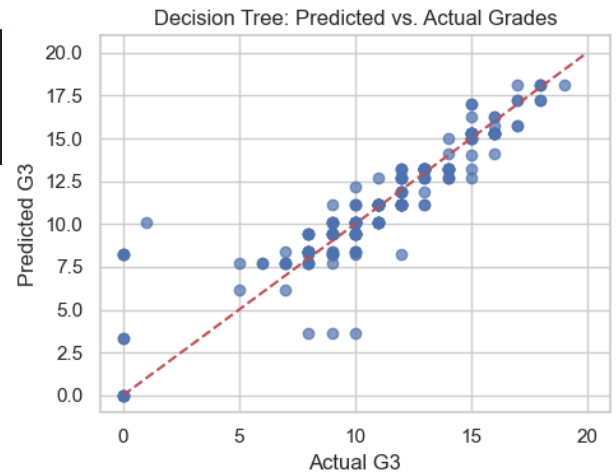


Figure 15: Decision Tree scatter plot

Random Forest constructs many Decision Trees using bootstrap samples and random feature selection, then averages their predictions to reduce overfitting and improve generalization performance. It also provides another tunable hyperparameter, $n_estimators$, denoting the number of individual trees. More trees can improve stability and accuracy, but at the cost of longer training and evaluation time (Breiman, 2001).

```

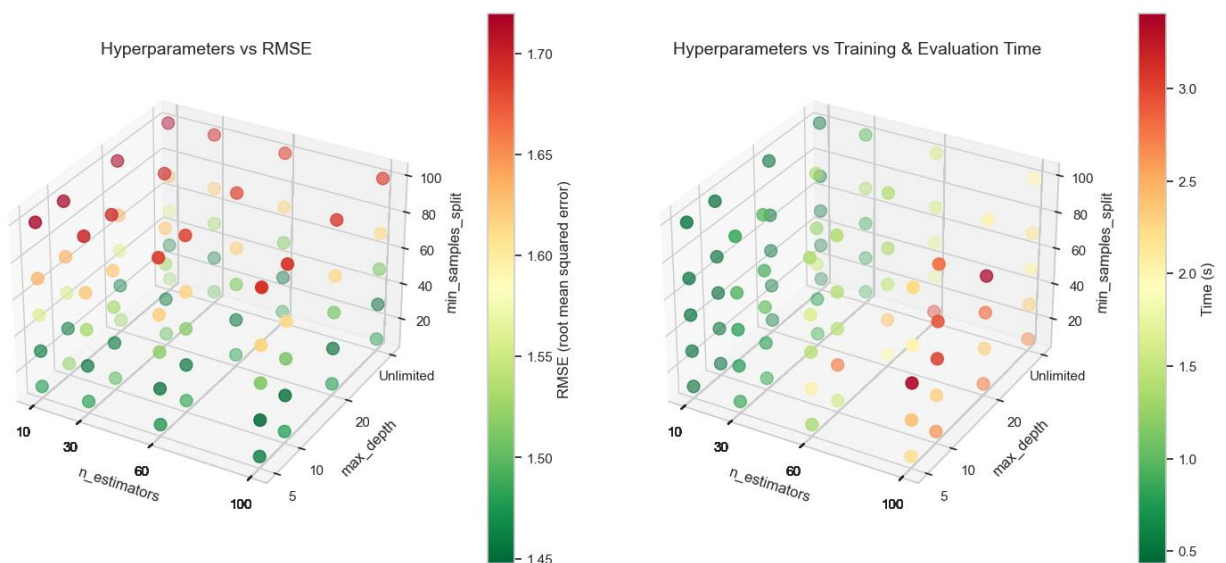
Random Forest hyperparameter tuning...
Best parameters: {'n_estimators': 100, 'min_samples_split': 30, 'max_depth': 10}
Best cross-validated RMSE: 1.44738179
Total time: 114.3031s
Average time per iteration: 1.4288s; Mean: 1.4285s

```

	n_estimators	min_samples_split	max_depth	rmse	time
0	10	10	5.0	1.476915	0.465093
1	10	10	10.0	1.521666	0.432026
2	10	10	20.0	1.525001	0.484950
3	10	10	Unlimited	1.525001	0.516321
4	10	30	5.0	1.456826	0.469993

Figure 16: Random Forest hyperparameter testing

Random Forest Hyperparameter Tuning Results



Figures 17, 18: Random Forest hyperparameters vs RMSE and time scatter plot

With the optimal hyperparameter settings of 100 trees, a minimum split size of 30, and a maximum depth of 10, the Random Forest achieved a cross-validated RMSE of approximately 1.447, an MAE of 0.897, and an R^2 of 0.857. This provides a slight accuracy boost over a single Decision Tree; however, hyperparameter tuning and final evaluation takes significantly more time, especially for large values of $n_estimators$ (as seen from the 3D scatter plot above).

```
Random Forest parameters: {'n_estimators': 100,
' min_samples_split': 30, 'max_depth': 10}
Training time: 0.9372s
Cross-validated RMSE: 1.4473817903150725
Cross-validated MAE: 0.8965975186808688
Cross-validated R-squared: 0.8570922470217652
Evaluation time: 10.2582s
```

Figure 19: Random Forest evaluation scores

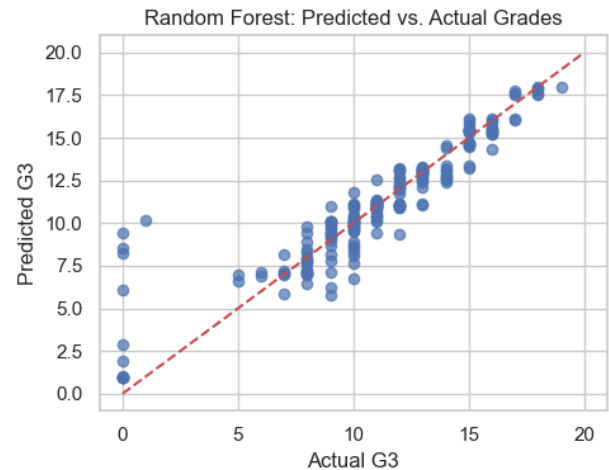


Figure 20: Random Forest scatter plot

Results

The Linear Regression model fitted in minimal time and yielded an RMSE of 1.588, an MAE of 0.988, and an R^2 of 0.830 on the test set. Five-fold cross-validation across the full dataset confirmed these metrics, demonstrating that a simple linear approach already captures a substantial portion of the variance in final grades with very low computational cost.

In Decision Tree, a grid search over *max_depth* and *min_samples_split* (evaluated via cross-validation) identified the optimal settings of *max_depth* of 5 and *min_samples_split* equal to 30. Under these parameters, the Decision Tree achieved an RMSE of 1.483, an MAE of 0.903, and an R^2 of 0.848. Training and evaluation times remained comparable to the Linear Regression baseline; modeling non-linear patterns led to a measurable accuracy improvement.

As for Random Forest, expanding to an ensemble of 100 trees ($n_estimators = 100$, $max_depth = 10$, $min_samples_split = 30$) further reduced error to an RMSE of 1.447 and an MAE of 0.897, with an R^2 of 0.857. Unfortunately, these gains over a single Decision Tree are modest and come at the expense of significantly longer training and evaluation times, particularly as the number of trees increases.

Overall, Random Forest offers the best prediction (~85.7% accuracy) for final-grade regression in this dataset, but practitioners must balance accuracy against the higher computational overhead. In contrast, Linear Regression achieves comparable performance (~82.9%) with much less time complexity and utilizing minimal resources. Decision Trees strike a middle ground of improved non-linear modeling with runtime similar to linear models (getting a prediction accuracy of around 84.8%). This accuracy is in line with the dataset's size (around 1000 records with 33 attributes) and may be very close to the true limit of what the data can provide; other 15–18% may be unpredictable variance from factors not captured in the dataset.

Results indicate that incorporating early-term grades gives the greatest predictive accuracy, yet other variables (such as absence frequency, parental occupation and education, and student lifestyle factors) also contribute meaningfully to model performance. Future work could include with temporal data (e. g. tracking grades over time), combining different types of models, or building a simple alert system to flag students who might need help. It would also be wise to check that the model works equally well for all groups of students (for example, by gender or income level) and to use explanation tools (like SHAP or LIME) so educators can understand why exactly the model makes certain predictions.

Contribution

This study advances the field of educational data mining by demonstrating how three basic and common machine-learning models (Linear Regression, Decision Tree, and Random Forest) can be rigorously optimized on real-world secondary-school data. Through systematic preprocessing, evaluation, k-fold cross-validation, and extensive hyperparameter tuning, it provides a methodological blueprint for balancing accuracy, interpretability, and computational cost. An interpretive analysis of the best-performing models further identifies not only early-term grades but also attendance patterns, parental education, and lifestyle variables as meaningful predictors, offering deeper insight into factors driving student success and failure.

On the practical side, these findings equip educators and policymakers with a data-driven framework for early-warning systems and targeted interventions. By relying initially on readily available midterm scores and then selectively incorporating contextual features, schools can identify at-risk students sooner, allocate resources more efficiently, and design personalized support strategies. The feature-importance thresholds (e.g., absence limits) translate directly into actionable policies. Proposed extensions, such as time-series monitoring and fairness auditing, lay the groundwork for ethically robust, continuously improving analytics that enhance overall educational quality.

Conclusion

The machine learning models developed successfully predicted students' final grades ($G3$) using midterm grades ($G1$, $G2$) and contextual attributes. Random Forest achieved the highest accuracy ($R^2 \approx 0.857$), slightly outperforming the Decision Tree ($R^2 \approx 0.848$) and Linear Regression ($R^2 \approx 0.830$). This confirms that ensemble methods enhance predictive capability, though at much greater computational cost. Midterm grades ($G1$, $G2$) were the strongest predictors, aligning with prior research indicating that historical performance heavily influences future outcomes.

Notably, while demographic and social features (e.g., parental education, study time, past failures) contributed to model accuracy, their incremental impact beyond midterm grades was modest. This suggests educators can achieve reasonably accurate early predictions using midterm scores alone, reserving supplementary data for nuanced interventions. Future work should prioritize temporal tracking, fairness validation across student subgroups, and explainable AI techniques to translate predictions into actionable educational strategies.

This project helps schools spot struggling students early by accurately predicting final grades using midterm scores and other factors (like absences or home life). It shows simple models work well, with midterm grades being the strongest predictor, with school and home factors adding useful detail. This sets up future tools for real-time alerts in classrooms and deeper studies into why certain factors affect performance, while ensuring fairness for all students. Future research should prioritize longitudinal tracking, fairness auditing across student subgroups, and hybrid models balancing speed and accuracy for ethical deployment.

References

1. Cortez, P. (2008). Student Performance [Dataset]. UCI Machine Learning Repository. Available at: <https://doi.org/10.24432/C5TG7T>.
2. Cortez, P., & Silva, A.M. (2008). Using data mining to predict secondary school student performance. Proceedings of 5th Annual Future Business Technology Conference
3. Wakelam, Ed et al. (2019). The potential for student performance prediction in small cohorts with minimal available attributes. British Journal of Educational Technology. 51.
4. Abu Zohair, L.M. (2019). Prediction of Student's performance by modelling small dataset size. International Journal of Educational Technology in Higher Education. 16, 27.
5. Chen, J. et. al. (2025), Application of machine learning in higher education to predict students' performance, learning engagement and self-efficacy: a systematic literature review. Asian Education and Development Studies. 14, 2, 205-240
6. Costa-Mendes et. al. (2021). A machine learning approximation of the 2015 Portuguese high school student grades: A hybrid approach. Education and Information Technologies. 26. 1-21.
7. Schneider, A., Hommel, G., & Blettner, M. (2010). Linear regression analysis: part 14 of a series on evaluation of scientific publications. Deutsches Arzteblatt international, 107(44), 776–782.
8. Breiman, L. (2001). Random Forests. Machine Learning 45, 5–32.
9. Jena, M., & Dehuri, S. (2015). Decision tree for classification and regression: A state-of-the-art review. Informatica, 44 (4).
10. Chicco, D., Warrens, M. J., & Jurman, G. (2021). The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. PeerJ Computer Science, 7, e623.