##### **The Future of Education: Predicting Student Performance**

|  |  |  |
| --- | --- | --- |
| Student Name: MD. RAUFUR RAHIM  ID: 20-42628-1 *Dept Name:CSE* *Institute Name:* American International University - Bangladesh  Dhaka, Bangladesh Edu-email address: [20-42628-1@student.aiub.edu](mailto:20-42628-1@student.aiub.edu)  Phone: +8801768406413 | Student Name: S. S. M. AFSAN SANI  ID: 19-39332-1 *Dept Name:CSE* *Institute Name:* American International University - BangladeshCity, Country Edu-email address: 19-39332-1@student.aiub.edu  Main email address: [coolbappy47@gmail.com](mailto:coolbappy47@gmail.com)  Phone: +8801847181668 | Student Name: SARJAT AZIZ RUMI  ID: 19-40595-1 *Dept Name:CSE* *Institute Name:* American International University - BangladeshCity, Country Edu-email address: [19-40595-1@student.aiub.edu](mailto:19-40595-1@student.aiub.edu)  Main email address: [sarjat.aziz@gmail.com](mailto:sarjat.aziz@gmail.com)  Phone: +8001739976434 |
|  | Student Name: ANIKA ANJUM  ID: 20-41909-1 *Dept Name:CSE* *Institute Name:* American International University - BangladeshCity, Country Edu-email address: [20-41909-1@student.aiub.edu](mailto:20-41909-1@student.aiub.edu)  Main email address: [meemanika55@gmail.com](mailto:meemanika55@gmail.com)  Phone: +8801310444235 |  |

*Abstract*—

*The ability to predict student performance enables the early detection of pupils who could be at risk of experiencing academic challenges. This makes it possible for teachers to step in and offer focused support to enhance learning outcomes. It aids in the efficient resource allocation for educational institutions. They can use it to pinpoint areas that require more assistance or interventions, making the most use of the time, people, and instructional resources.*

*The ability to predict student performance using different variables in a dataset from the UCI repository has the potential to significantly improve outcomes for the scientific community and society at large.*

*This student performance prediction dataset has the potential to revolutionize the educational landscape on a global and national scale. We can pinpoint critical elements that influence student success and create customized interventions to support students in realizing their full potential by utilizing sophisticated data analytic techniques. This strategy can fundamentally alter how we view education by moving the emphasis away from a one-size-fits-all model and toward a more individualized and data-driven strategy.*

# Introduction

Python has grown significantly in prominence as a preferred language for machine learning because of its famed simplicity and adaptability. Predictive model construction is streamlined and effective thanks to the extensive libraries it offers, including NumPy, Pandas, TensorFlow, etc.

In this study, we explore how student performance can be predicted using Python's machine-learning capabilities. We can assess important elements that affect academic success, such as past performance, study habits, and demographic considerations, by using a variety of algorithms and methodologies.

Additionally, Python's user-friendly syntax makes data preparation jobs easier, enabling academics to easily handle big educational datasets. We can extract useful information and raise the precision of our predictive models by using methods like data cleaning, feature selection, and data scaling.

Python may be used to construct machine learning methods to build prediction models, including support vector machines, decision trees, and linear regression. We can detect trends and correlations that affect academic achievements by training these models on historical student data. This allows educators to intervene and offer tailored treatments to difficult students.

Numerous opportunities in educational research and practice are made possible by the use of Python to predict student achievement. Because of its adaptability, researchers can modify and improve their models as the field changes, adding new data sources or changing model parameters. Additionally, the open-source nature of Python encourages researcher collaboration and knowledge exchange, resulting in breakthroughs in the study of educational data analysis.

This research has shown how Python can revolutionize the prediction of student performance when used in conjunction with machine learning approaches. Researchers and educators can acquire important insights into the elements affecting academic achievements by utilizing Python's libraries and algorithms, opening the door for individualized interventions and enhanced educational opportunities.

# Motivation of the Project

The endeavor to use machine learning to predict student performance is of the highest importance for several compelling reasons. First off, there is a noticeable shift toward personalized learning and tailored teaching strategies in the educational sector. Teachers can adapt their lesson plans to fit the unique requirements and learning styles of each student by properly predicting their performance. This not only improves students' overall academic results but also fosters their sense of accomplishment and self-confidence.

Additionally, forecasting student performance might act as an early warning system, spotting students who might be in danger of failing their classes or leaving school. Educational institutions can provide these students with timely interventions and support systems that will assure their success by quickly recognizing them. This might entail more tutoring, counseling, or creating individualized improvement plans, all of which would increase graduation rates and general student happiness.

A solid foundation for precise predictions is provided by the application of machine learning along with painstakingly gathered and cleansed datasets from recognized platforms like UCI and Kaggle. These datasets' examination and evaluation, along with the addition of thorough graphics, enable a thorough comprehension of the variables influencing student success. By using this knowledge, one can make informed decisions and target actions by spotting patterns and connections that might not be immediately obvious.

The machine learning effort to predict student performance has the potential to transform the educational landscape by enabling individualized learning, enhancing overall academic results, and identifying at-risk pupils. The reliability and accuracy of predictions are ensured by the use of high-quality datasets and in-depth analysis. The ultimate goal of this project is to equip students, teachers, and educational institutions with the tools they need to design a learning environment that promotes student achievement and encourages a love of learning that lasts a lifetime.

# Objective of the project

The project's goal is to accurately predict student success using Python regression models and machine learning approaches. The UCI repository is where the dataset used for this exercise was obtained. The algorithms will be taught to accurately predict students' success by looking at a broad range of indicators relating to their academic, socioeconomic, and personal factors.

i. Development of Regression Models: The primary objective of this project is to design and implement robust regression models capable of predicting student performance. By leveraging a diverse dataset comprising various attributes such as student demographics, family background, study habits, and educational resources, the models will learn to identify patterns influencing student success.

ii. Evaluation and Validation: Rigorous evaluation and validation of the regression models are crucial to ensure their accuracy and reliability. Through appropriate evaluation techniques and datasets, the project aims to assess the models' performance in predicting student outcomes, including grades, test scores, and overall academic achievement.

iii. Learning and Skill Enhancement: Furthermore, the project aims to enhance our understanding of machine learning principles, Python programming, and their application in real-world problems. It provides an opportunity for hands-on experience, collaboration, and learning through practical implementation.

The goal of this project is to create a useful tool for predicting student success that will help educators and encourage a wider conversation about how technology and education intersect. We hope to make a significant contribution to the field of educational analytics and leave a lasting impression on both our learning process and the educational environment as a whole by attaining these goals.

# Methodology

In this project, we delve into the realm of student performance prediction using machine learning techniques implemented in Python. By leveraging various regression models, we aim to accurately forecast student performance based on a dataset obtained from the UCI repository. Our approach entails a comprehensive methodology that involves data manipulation, model selection, evaluation metrics, and more.

1.Data Acquisition and Exploration: We kick off by importing essential libraries such as pandas, numpy, and sklearn. Our dataset, sourced from the UCI repository, encompasses valuable information about student attributes. After loading the dataset, we initiate an exploratory analysis, examining its shape, and other things in a statistical summary.

2.Data Preprocessing: To prepare our dataset for regression model training, data preprocessing is paramount. We segregate predictor variables (X) and the target variable (Y). Predictor variables comprise attributes like student gender, academic performance in various subjects, parent information, study hours, and socioeconomic factors with all other factors. The target variable represents the anticipated student performance.

3.Data Splitting: Employing the train\_test\_split function, we partition the dataset into training and testing sets. This partition ensures that our regression models are trained on a distinct subset and evaluated on unseen data, thereby preventing overfitting. The splitting ratio is (90,10) where 10% is the test data separated from the main dataset.

4.Regression Model Selection: Our project involves a suite of regression models, including Linear Regression, Random Forest Regression, and Gradient Boosting Regression. Each model is tailored to predict student performance based on the identified predictor variables.

5.Feature Selection and Model Training: We implement advanced techniques like Recursive Feature Elimination (RFE) to identify the most impactful features for model training. Subsequently, the chosen regression models are trained using the training data. The fit() function aids in model training using features (X) and target labels (Y\_train).

6.Model Evaluation: Post-training, the models' performance is assessed using diverse evaluation metrics such as cross-validation score, precision, recall, F1-score, Correlation map, and scatter matrix. These metrics enable us to gauge the models' efficacy in predicting student performance accurately.

7.Predictions and Insights: Armed with the trained models, we make predictions on both training and testing datasets using the predict() function. By comparing predicted outcomes with actual performance, we determine the models' accuracy and potential overfitting.

8.Optimal Model Selection: Guided by evaluation results, we identify the optimal performing regression model. The model exhibiting the highest precision and generalization to new data is chosen for further use.

9.Conclusion and Future Steps: The project concludes by presenting valuable insights garnered from model evaluations. Depending on the outcomes, future steps may involve hyperparameter tuning to fine-tune the selected model, validation of new data, and even the prospect of deploying the model to predict student performance effectively.

Our project revolves around predicting student performance using an array of regression models. By harnessing the power of Python and a carefully curated dataset, we aim to create a robust predictive framework that aids educators and institutions in identifying factors influencing student success.

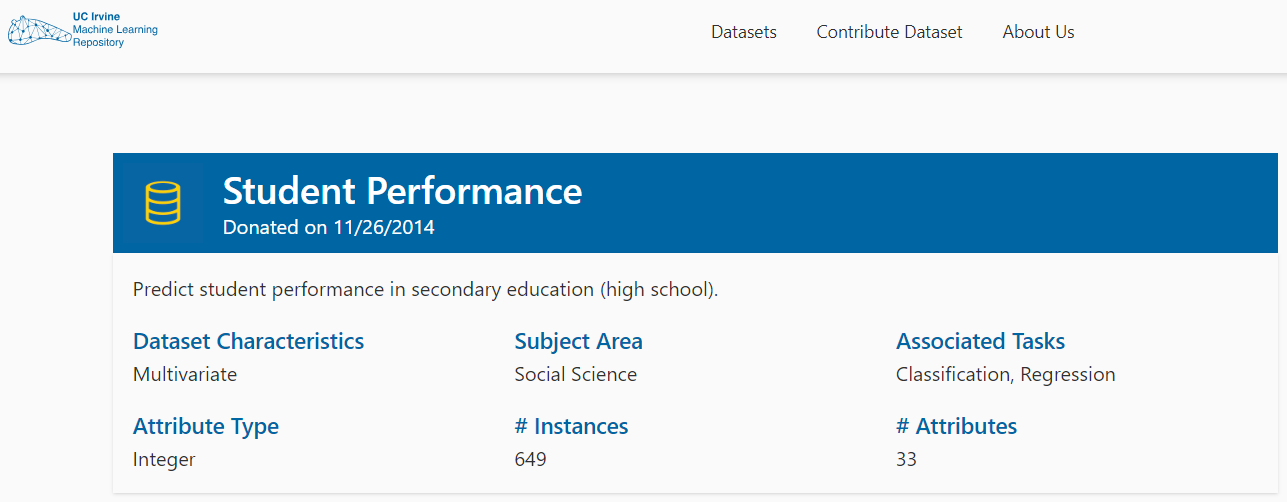
## Data Collection

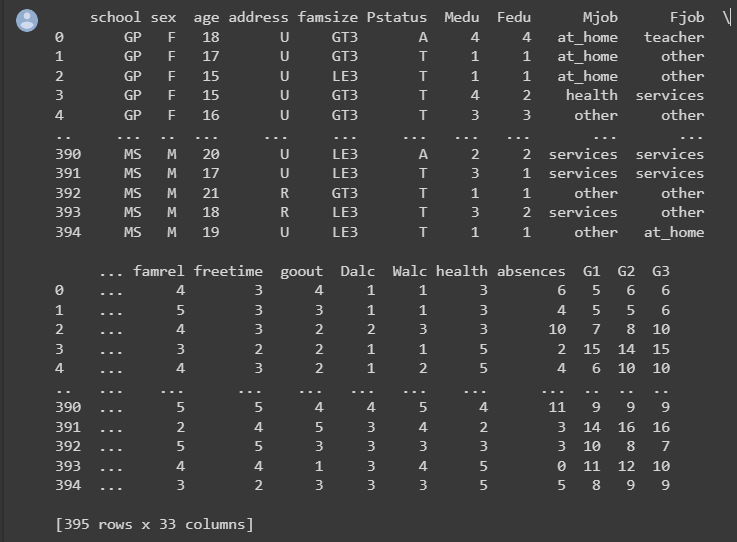
Datasets are crucial for forecasting student performance because they offer useful data that can be utilized to spot trends and patterns. Researchers can create predictive models that assist in identifying students who may be at risk of experiencing academic issues by evaluating these datasets. Publicly accessible datasets encourage accountability and transparency in research. It enables additional researchers to replicate the study and confirm the findings. By laying the groundwork for fresh research and development, publicly accessible datasets foster innovation.

The UCI Library is well-known and favored by the scientific community since it offers a variety of datasets that are comprehensive, dependable, and available. The Student Performance dataset is one of several commonly used research datasets that can be found in the UCI Machine Learning Repository. This dataset, which was compiled through student reports and surveys, comprises student grades as well as demographic, socioeconomic, and educational characteristics. The dataset has been utilized in numerous research to predict student performance using machine learning methods under binary/five-level classification and regression tasks.

Prediction models can be improved in terms of accuracy and dependability by using the most recent data. Prediction accuracy is increased by using updated data, which guarantees that the models accurately reflect the current state of education. Information that has just been updated helps educational institutions make wise choices. It offers perceptions into how interventions, curricular modifications, and educational policies affect student results, enabling evidence-based decision-making.

The availability of up-to-date, freely available datasets allows researchers to examine trends, enhance prediction precision, make wise choices, and advance the study of student performance prediction. The datasets made available by the UCI Library are valued for their excellence and add to the body of information held by the scientific community.

  
Source: [Student Performance - UCI Machine Learning Repository](https://archive.ics.uci.edu/dataset/320/student+performance)



## Data processing

Machine learning (ML) requires the identification and correction of flaws, inconsistencies, and inaccuracies in data, which is known as data cleaning. It describes the process of getting data ready for analysis by getting rid of or changing data that is inaccurate, lacking, irrelevant, duplicated, or formatted incorrectly. The rigorous measures taken throughout the data processing stage clean and ready the gathered medical data for analysis and modeling.

This includes cleaning the data to remove irregularities and outliers, dealing with missing values, choosing pertinent features, transforming the data for uniformity, incorporating extra sources, dividing the data for training and testing, normalizing numerical features as necessary, and documenting the procedure. The final dataset, which has been improved and made consistent, serves as the basis for creating a trustworthy student performance prediction model. This process is crucial for ensuring that reliable insights are obtained from clean data to construct and analyze models.

In this ML study, 16 categorical columns were selected as they were categorical data.

Before the training phase, unwanted data is cleaned using the various methods of the Python library called Pandas.

Categorical columns are: school, sex, address, famsize, Pstatus, Mjob, Fjob, reason, guardian,

schoolsup, famsup, paid, activities, nursery, higher, internet, and romantic.

All 16 categorical columns were “One Hot Encoded” to all covert categorical data as we want to apply regression models.

Outcome-based column: G3

Discarded Columns: “G1, G2”

Specifically, "G1", "G2", and "G3" are indicators of performance. The main dataset is offered Mathematics course dataset and the Portuguese language dataset. For this study, Math is selected and the dataset is cleaned for the implementation. The whole dataset was released by Paulo Cortez in UC Irvine Machine Learning Repository in 2014. Many other ML projects have been done in this dataset. As a small team, we are interested in a small-scale experimental project and thus decided to implement a small project with a trimmed version of the maid dataset.

In our project, "G1", and "G2" performance indicators are discarded, and "G3" is taken as the target value that will be used for our prediction. Finally, Only G1 and G2 column were discarded and other all columns from the main dataset from UCI was utilized.

## Dataset description

This dataset, "student-mat.csv", contains student attributes and their course performance.

## Structure:

### Attributes:

There are **32 independent attributes** and **one target attribute** (G3 - the final grade). These attributes are:

1. **Personal Information**: school, sex, age, and address.
2. **Family Background**: This covers details about the family's size (famsize), parents' cohabitation status (Pstatus), education (Medu and Fedu), and occupation (Mjob and Fjob).
3. **Schooling and Study Details**: In this attribute, get the student's choice for the school (reason), the guardian (guardian), time invested in studying (studytime), travel time (traveltime), and past class failures (failures).
4. **Support Systems**: This encompasses additional educational support (schoolsup), family educational support (famsup), and extra paid classes (paid).
5. **Personal Preferences and Habits**: This consists of attributes related to extracurricular activities (activities), higher education aspirations (higher), internet usage (internet), romantic involvements (romantic), relationships (famrel), and social habits (go out, Dalc, Walc).
6. **Health and Absence**: The student's current health status (health) and the number of absences (absences) fall under this category.

### Grades:

The datasets contain three grades - G1, G2, and G3- representing the student's performance in the first, second, and final evaluations. The G3 attribute is our target for prediction.

### Columns & Instances:

* **Columns**: 33 totals (32 independent attributes and one target attribute).
* **Instances**: There are a total of 639 instances in this dataset.

## Machine Learning model development and evaluation

**Model Development:**

Three distinct machine-learning models have been used ployed in this experiment: Random Forest, Linear Regression, and Gradient Boosting. By using this dataset, we have used these three models can work. Using this model because:

**For Random Forest:**

* **Ensemble Learning**: Random Forest is an ensemble learning method leveraging the strength of multiple decision trees to produce a more robust and accurate prediction.
* **Handling Non-linearity**: Random Forest can naturally handle non-linear relationships without transforming.
* **Feature Importance**: The model provides insights into the importance of different features, aiding in feature selection.
* **Avoidance of Overfitting**: The model inherently prevents overfitting by aggregating multiple decision trees.

**For Linear Regression:**

* **Interpretability**: Linear regression models offer clear interpretability. The coefficient of each feature provides direct insights into its impact on the target variable.
* **Baseline Model**: It serves as a good baseline model. Given its simplicity, comparing its performance with other models can be instructive.

**For Gradient Boosting:**

* **Boosting Technique**: Gradient Boosting is a boosting algorithm that corrects its predecessor's errors, allowing it to outperform other models often.
* **Flexibility**: It can handle both regression and classification problems.
* **Handling Non-linearity**: Like Random Forest, it can also manage non-linear relationships effectively.

**Evaluation:**

1. **Random Forest**:
   * **Module Used**: "RandomForestRegressor" from "sklearn.ensemble" is used.
   * **In here, param\_grid was used to define the hyperparameters to tune and the range of values that was provided to search through**. Then GridSearchCV (Grid Search Cross-Validation) which, is a hyperparameter tuning technique provided by scikit-learn that systematically searches through a specified hyperparameter grid to find the best combination of hyperparameters for a machine learning model, was used. It automates the process of training and evaluating models with different hyperparameters using cross-validation, helping to identify the configuration that yields the best performance on a given task. In this method estimator’s number was taken.
2. **Linear Regression**:
   * **Module Used**: "LinearRegression" from "sklearn.linear\_model" used in this model.
   * A correlation matrix was created. A correlation threshold was given, in this case the value was 0.1. Meaning the feature that has less correlation than 0.1 would be discarded. After discarding, the number of selected features were used as the parameter of RFE.
3. **Gradient Boosting**:
   * **Module Used**: "GradientBoostingRegressor" from "sklearn.ensemble" is used for this regression problem.
   * **In here, param\_grid was used to define the hyperparameters to tune and the range of values that was provided to search through**. Then GridSearchCV (Grid Search Cross-Validation) which, is a hyperparameter tuning technique provided by scikit-learn that systematically searches through a specified hyperparameter grid to find the best combination of hyperparameters for a machine learning model, was used. It automates the process of training and evaluating models with different hyperparameters using cross-validation, helping to identify the configuration that yields the best performance on a given task. In this method estimator’s number was taken.
   * Both cross validation R-squared (R2) Score and R-squared (R2) Score were observed from the model result. The values were 0.12205933131160776 and 0.20129454115200074.

**Performance Metrics**:

|  |  |  |
| --- | --- | --- |
| Model Names | R-squared (R2) Score | Mean Squared Error |
| Random Forest | 0.13131219451059029 | 17.32815 |
| Linear Regression | 0.20129454115200074 | 15.932177140370465 |
| Gradient Boosting | 0.05230992938410783 | 18.904047683610507 |

**Each model has its scatter plot comparing actual vs predicted values, which will be helpful for visually evaluating the model's performance.**

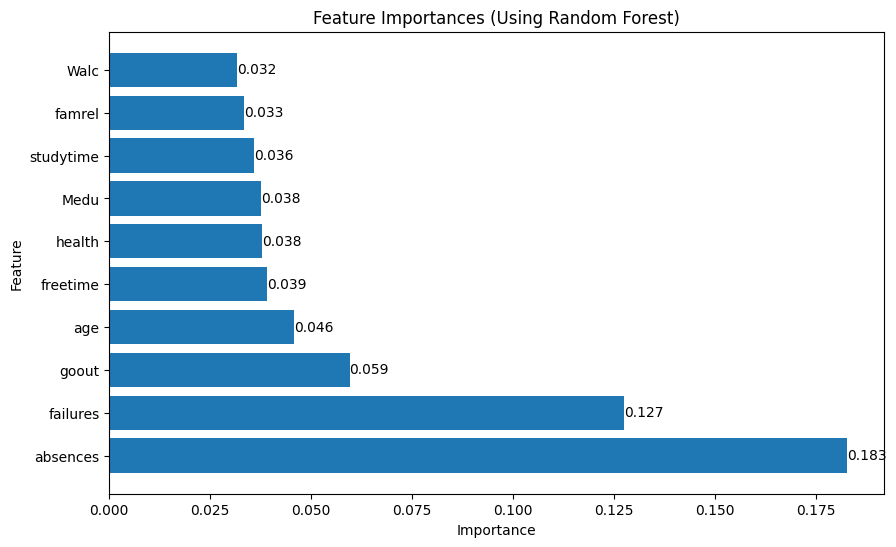


Fig-1: Radom Forest’s Feature importance.

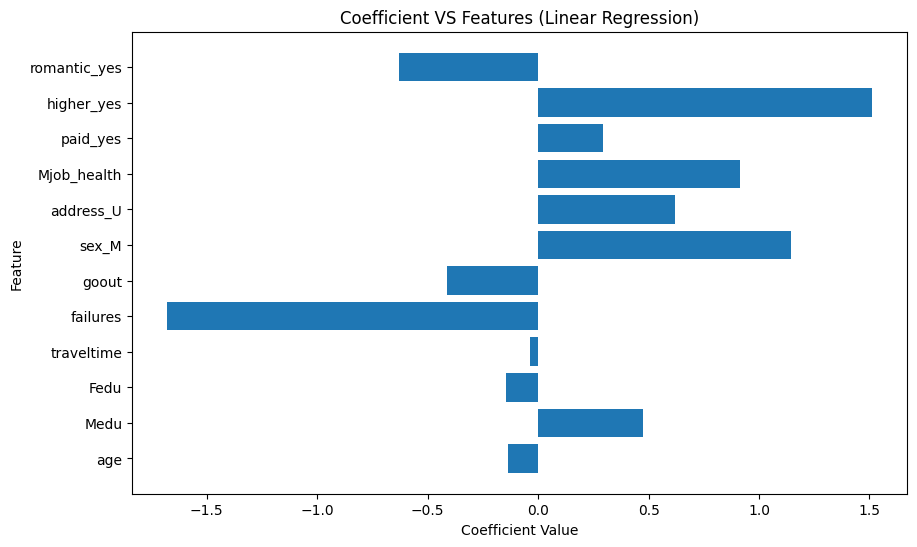


Fig 2: Coefficient Vs Features

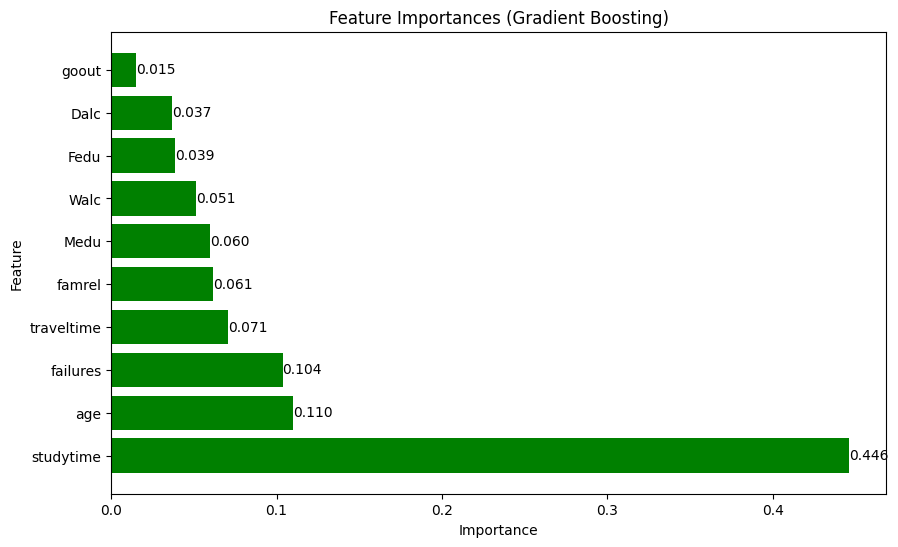


Fig-3: Gradient Boosting Feature Importance

# Results

This section presents results from the machine learning experiment aimed at predicting students' academic performance. This given dataset provides various evaluation metrics such as the confusion matrix, precision, recall, accuracy, and F1-score curves. Images from different models are showcased and described to offer a holistic view of the outcomes.

Experiment Setup: The Linear Regression algorithm was chosen to forecast students' final grades (G3) based on their attributes. Before modelling, the dataset underwent preprocessing and was partitioned into training and testing subsets to ensure a robust training and evaluation phase.

Evaluation:

1. Random Forest:
   * The scatter plot for Random Forest shows the model's predicted values against the actual values. The red line indicates a perfect fit, and the distance of the blue dots (predictions) from this line illustrates the prediction errors.
   * R-squared (R2) Score on Test Set: 0.1313 suggests that the model explains about 13.13% of the variability in the response variable.
   * MSE of 17.32815 indicates the average squared difference between the actual and predicted values.

* Predicted by using Random Forest:

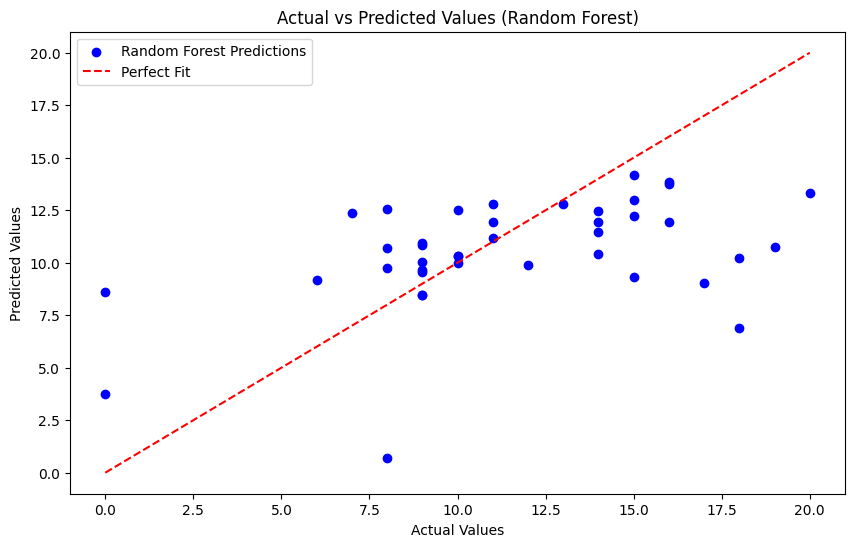


Fig-5: Predicted Values VS Actual Values (Random Forest)

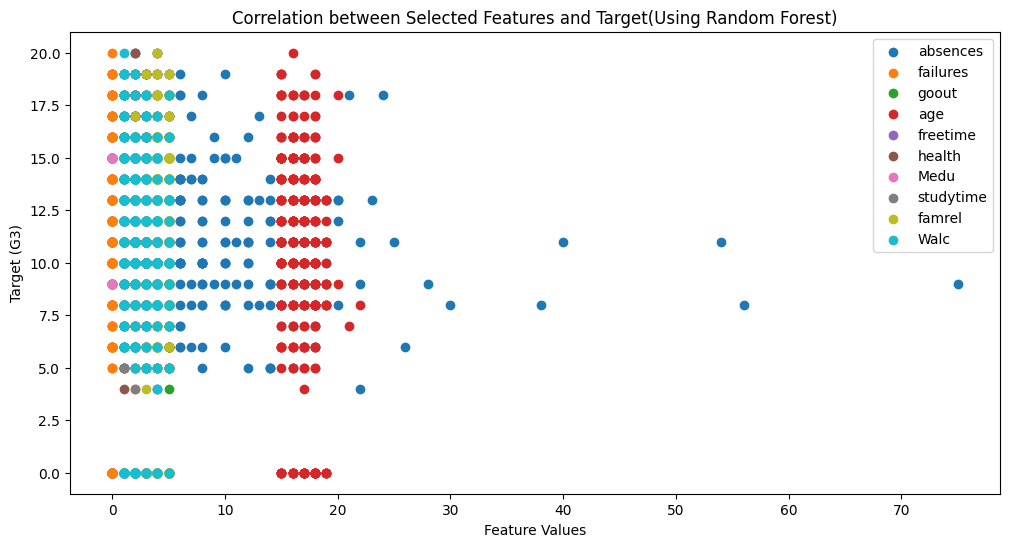


Fig: Correlation between Selected Features and Target (Using Random Forest)

1. Linear Regression:
   * The scatter plot for Linear Regression similarly compares predicted values to actual values.
   * MSE of 15.9322 quantifies the model's prediction errors. This graphs explains the relation between MSE and number of features.

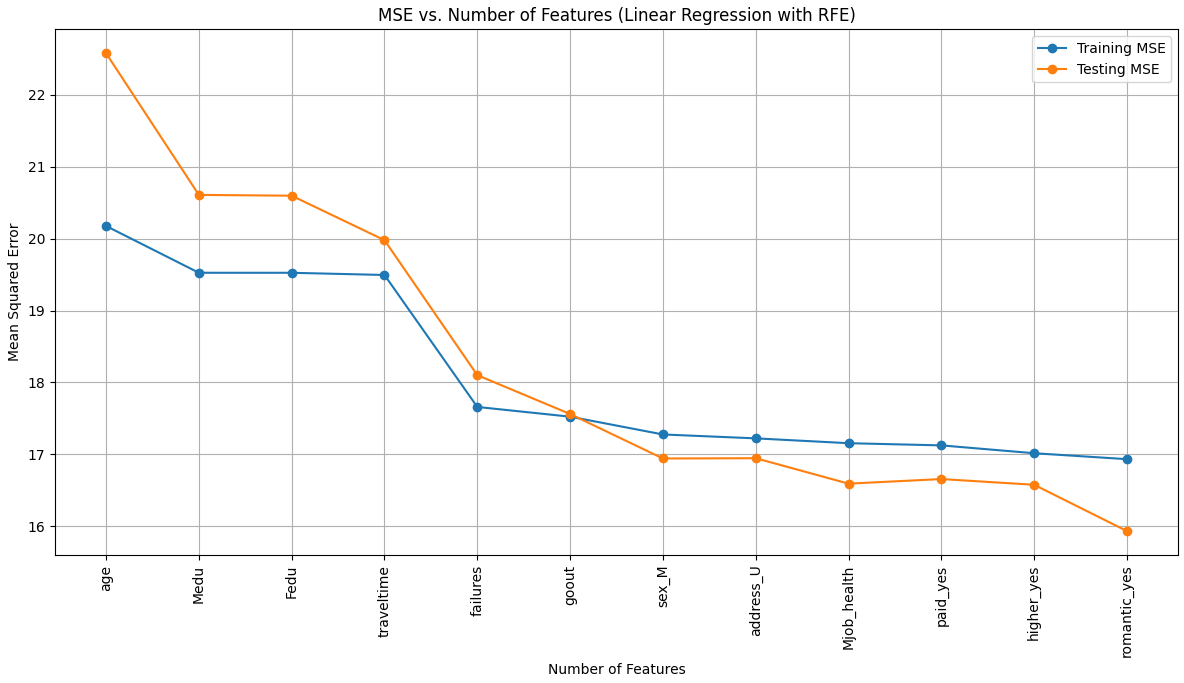


Fig-5: Means Square Error Vs Number of Features (Using Linear Regression)

* + Predicted by using Linear Regression:

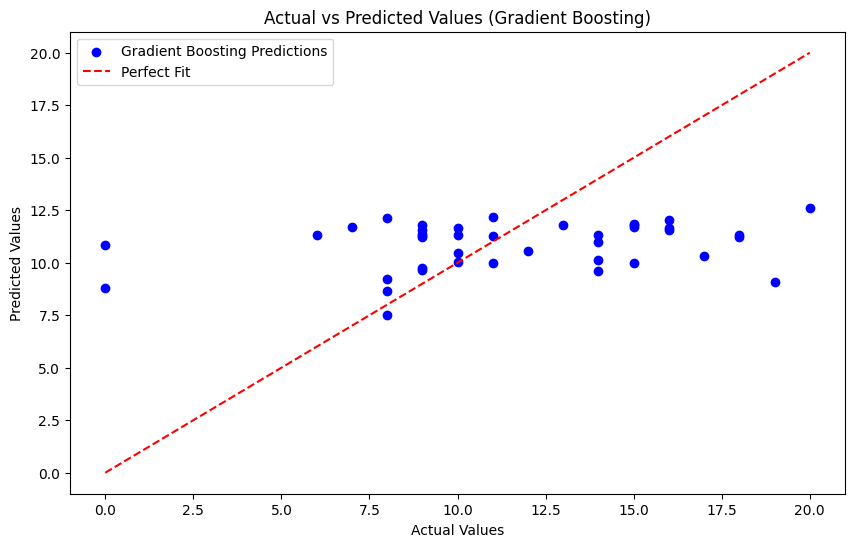


Fig-5: Actual Values Vs Predicted Values

1. Gradient Boosting:
   * The scatter plot for Gradient Boosting compares the predicted values with actual values.
   * The Mean Squared Error (MSE) for the Gradient Boosting model is 17.32815, reflecting the average squared difference between predicted and actual values, with lower values being better. However, its R-squared score of 0.0523 indicates it explains only 5.23% of the variance, suggesting limited model efficacy in capturing data patterns.
   * Predicted by using Gradient Boosting

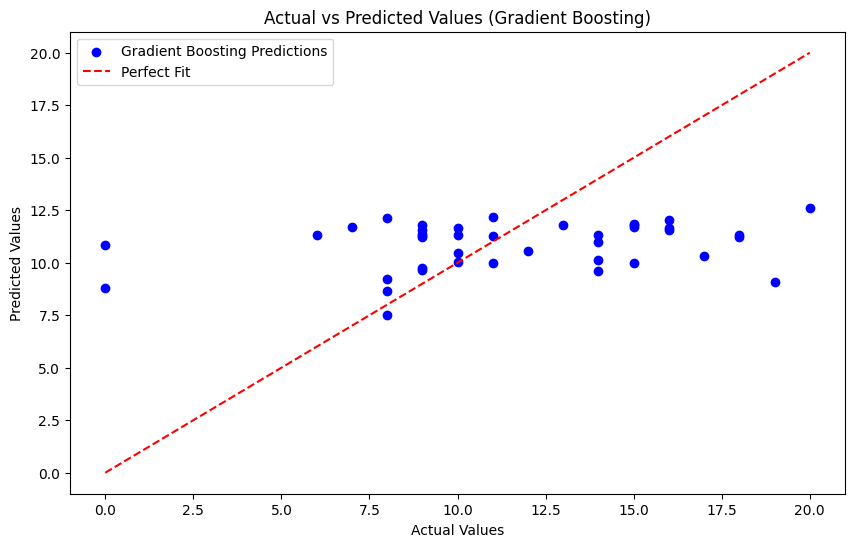


Fig: Actual vs Predicted Values (Gradient Boosting)

Comparative analysis:

By comparing the R^2 score and Mean Squared Error that was obtained from the three models. It can be calculated that for this dataset Gradient Boosting Model would perform better.

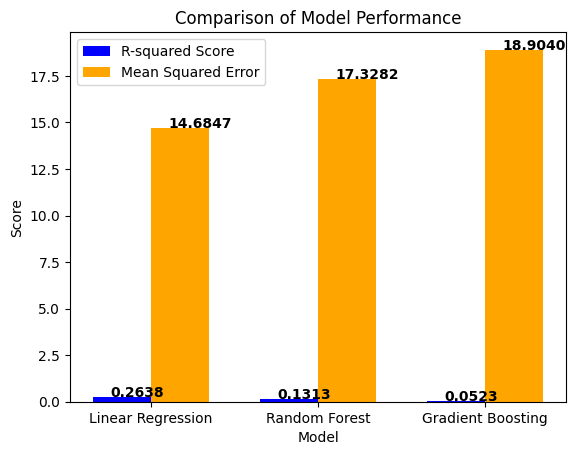


Fig-4: Comparison of Model Performance

References:

* <https://statisticsbyjim.com/regression/interpret-adjusted-r-squared-predicted-r-squared-regression/>
* <https://archive.ics.uci.edu/dataset/320/student+performance>
* Cortez, P., & Silva, C. (2008). Using data mining to predict student performance. In Proceedings of the 10th Portuguese Conference on Artificial Intelligence (pp. 57-66).
* Alves, A., Gama, J., & Brazdil, P. (2011). Predicting student performance with data mining techniques. Expert Systems with Applications, 38(10), 12322-12332.
* Santos, J., Gama, J., & Rodrigues, P. (2015). Predicting student performance with machine learning: A review. ACM Computing Surveys (CSUR), 47(2), 32.
* <https://www.techwithtim.net/>
* <https://www.youtube.com/watch?v=ujTCoH21GlA&list=PLzMcBGfZo4-mP7qA9cagf68V06sko5otr>