**Capstone Project: Predictive Modelling of Exam Scores Based on Student Lifestyle Behavior**

Isaac Onyach

25th April 2025

Contents

[Predictive Modelling of Exam Scores Based on Student Lifestyle Behavior 3](#_Toc196504640)

[Introduction 3](#_Toc196504641)

[Dataset Description 3](#_Toc196504642)

[Initial dataset exploration 4](#_Toc196504643)

[Data visualization 5](#_Toc196504644)

[Correlation analysis 7](#_Toc196504645)

[Preprocessing the data 8](#_Toc196504646)

[Step 2. Training a Linear Regressor Model 10](#_Toc196504647)

[Step 3. Training a Random Forest Model for Comparison 14](#_Toc196504648)

[Explanation of Key results 15](#_Toc196504649)

[Conclusion 16](#_Toc196504650)

# Predictive Modelling of Exam Scores Based on Student Lifestyle Behavior

## Introduction

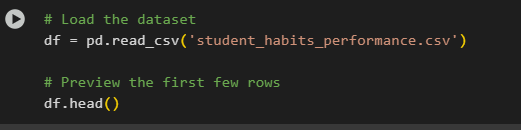
Beyond classroom instructions, a student’s academic performance is influenced by a lot of many other factors. Nevertheless, with the growing accessibility to data, educational researchers and academic institutions now have the capacity to explore how students’ daily lifestyle choices affect their performance. Moreover, understanding these relationships can offer meaningful insights for educators, students, and policy-makers aiming to improve learning outcomes.

In this project, we explore the relationship between a student’s daily habits and academic performance. The modern student juggles study time with screen time, diet, sleep, and social life, all of which have an impact on how they perform in school. The goal of the project is to analyze and predict exam scores based on lifestyle factors such as study hours, social media usage, sleep duration, mental health, and more. By the end of the project, we should answer the question, ‘What habits most strongly influence a student academic performance? And can we use these to make reliable predictions.

**Problem statement:** “How do students’ daily habits and lifestyle choices impact their academic performance, and can these behaviors be used to accurately predict exam scores?”

## Dataset Description

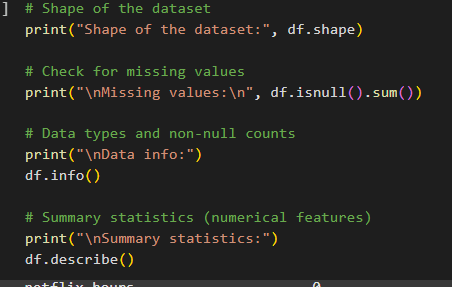
The first step in the project involved importing the dataset titled *student\_habits\_performance.csv.* This dataset contained a record of 1000 students. The information in the dataset included the students’ daily habits, lifestyle choices, and exam scores. The dataset was loaded into a Pandas dataset for processing and analysis.



The head() and tail() function helped in understanding the nature and structure of the dataset, which was important in giving a glimpse into the variables, their formats, and potential inconsistencies of missing data. The data set contained both numerical (study hours, sleep hours, e.t.c) and categorical features (gender, diet quality, etc.). This step helped us familiarize with the data, which was essential for preprocessing tasks such as handling missing values, converting data types, and encoding categorical variables.

### Initial dataset exploration

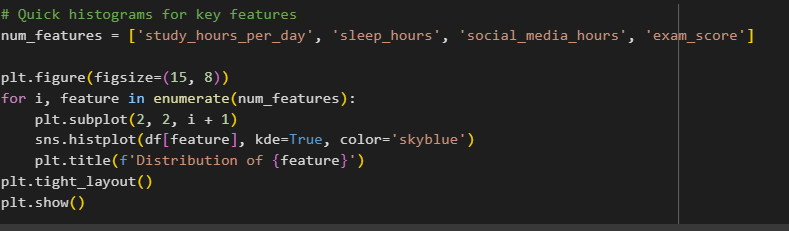
Several key exploration commands were executed to assess the structure and scope of the data set. The dataset had 1000 rows and 16 columns, each of which represented a student and their associated behavioral and academic attribute.



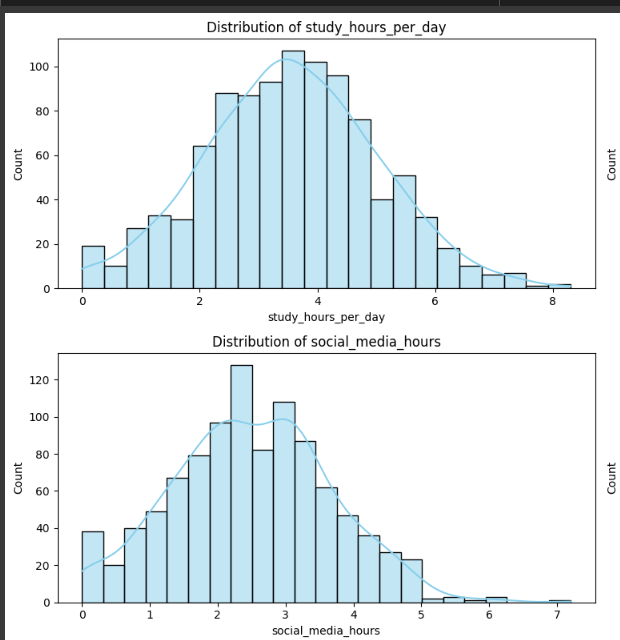
From a missing value analysis, only one column, parental\_education\_level contained missing entries (91 out of 1000 entries), all the other columns were complete. The dataset included a combination int64, float64, and object data types. The object data-type columns represented categorical variables while the int and float represented numerical data types. The student column was a unique identifier which served no purpose in the project and had to be dropped from the analysis. The df.describe() generated a summary statistic for all numerical columns which provided insights that laid the groundwork for data cleaning, analysis, and modelling.

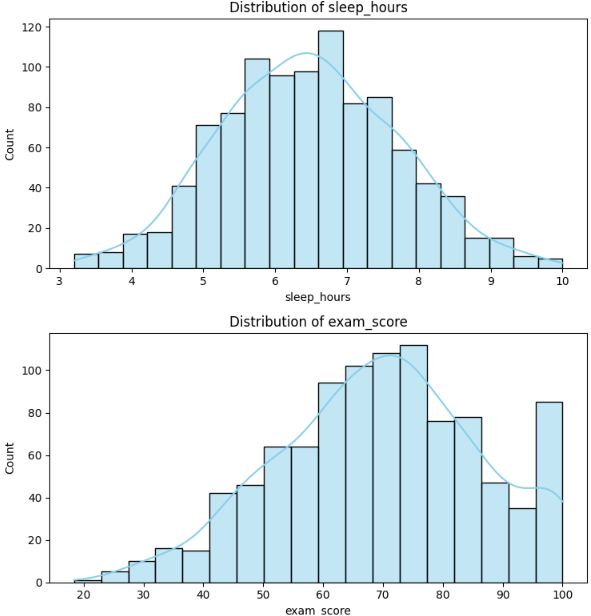
### Data visualization

We then generated histograms to gain deeper insight into the distribution of variability of important numerical variables. The selected histograms were “study\_hours\_per\_day, sleep\_hours, social\_media\_hours, and exam\_score.” The visualization, generated by the code snippet below, was to help observe outliers, skewness, spread clustering, and behavioral trends.



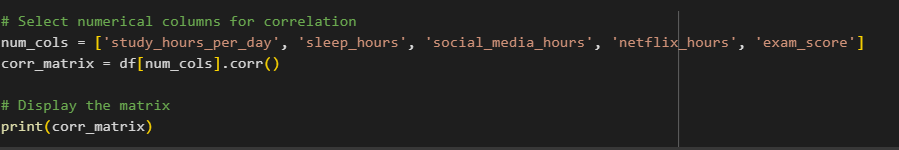
One of the outliers was in a student with 100 exam score but few hours of study. Most of the features were normally distributed. Majority of the students were also studying between two to five hours, and exam scores ranged between 50 and 90.





## Correlation analysis

A correlation analysis evaluates the relationship between key numerical variables and student performance. The generated correlation matrix helped quantify how strongly and in what direction two variables are related.



Key findings from the correlation matrix were as follows:

|  |  |  |
| --- | --- | --- |
| Variable | Correlation with exam\_score | Interpretation |
| study\_hours\_per\_day | 0.83 | Strong positive correlation — students who study more tend to score significantly higher. |
| sleep\_hours | 0.12 | Weak positive correlation — suggests some benefit from more sleep, but not a strong driver. |
| social\_media\_hours | –0.17 | Moderate negative correlation — heavier social media usage tends to be linked with lower exam scores. |
| netflix\_hours | –0.17 | Moderate negative correlation — more time spent watching Netflix may impact performance negatively. |

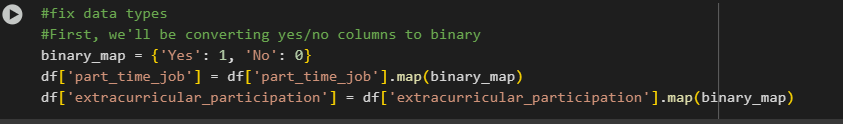
Nevertheless, the results from the correlation matrix aligned with expectations. For instance, study hours pr day was by far the most influential variable among all the variables analyzed.

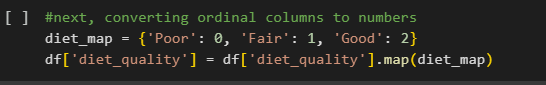
### Preprocessing the data

Having gained a clear understanding of the dataset structure, the next step involved cleaning and preparing the data for modelling. An earlier analysis had shown that only the *parental\_education\_level* had missing values. Moreover, since high school was the most frequent value in that particular column, missing values were imputed using ‘mode.’

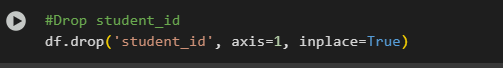


Next step involved handling categorical variables. Two columns (part\_time\_job and extracurricular\_participation) used Yes/No responses. These were converted to binary format (1/0) for model compatibility.

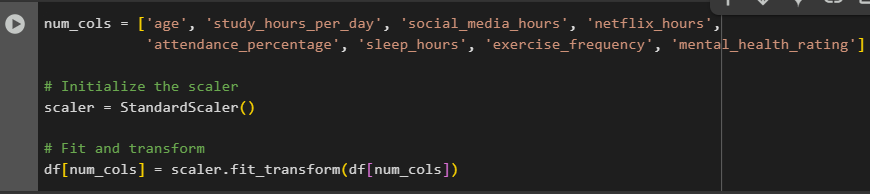






Irrelevant features were dropped in this stage since it served no predictive values.  


Moreover, since gender is a nominal categorical variable with more than two categories (female, male, other), it was one-hot encoded to avoid implying any ordinal relationship. The dummy column gender\_female, was dropped to prevent multicollinearity. Final data check was done using the df.info() function and it showed that all object types had been handled. Hence, the dataset was ready for modelling.

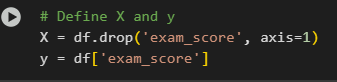
Lastly, since we were trying to predict an exam\_score, which is a continuous numerical value (0-100), we were to use linear regression. Thus, we had to scale our numerical features. The code for performing the feature scaling on the numerical columns of the dataset is shown below.  


The columns with numerical data listed in the snippet above are standardized to have a mean of 0 and a standard deviation of 1. Next, the standardization transforms the data such that mean = 0, and standard deviation = 1. The fit\_transform() function computes the mean and standard deviation for each column and applies the scaling using the computed and standard deviation.

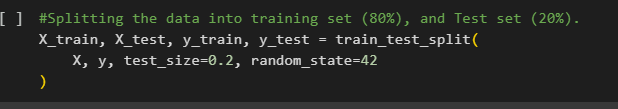
Scaling is important in linear regression because the model performs better when features are on a similar scale, especially those that use gradient descent optimization. Features with larger scaling usually dominate the model’s learning process, leading to biased results. Standardization ensures that each feature contributes equally to the model.

## Step 2. Training a Linear Regressor Model

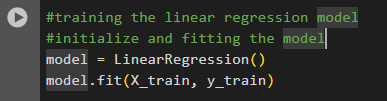
Step 2 involved training a linear regressor model to predict exam scores based on the given features, evaluate its performance, and output key metrics. The first step involed defining (x) and (y). x contained all features (independent variables) excluding exam\_score of course, while (y) contained only the target variable, which is the exam\_score.

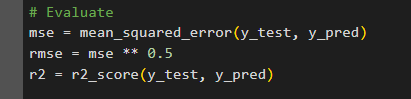


The data was then split into train/test sets. 20% of the data was reserved for testing. Random\_state=42 ensured reproducibility (same split every time). This was necessary for evaluating the model for unseen data (avoid overfitting).



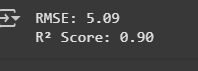
The model uses linear regression, meaning that it uses ordinary least squares (OLS) to try and fit a linear relationship between x and y. Moreover, the fit() function learns the coefficient for each feature in x-train, and use these to predict the y\_train.

  
 Lastly, it was time to make predictions and evaluate performance. The snippet predicts exam scores for the test set. Moreover, the snippet,

 can be tabulated as below

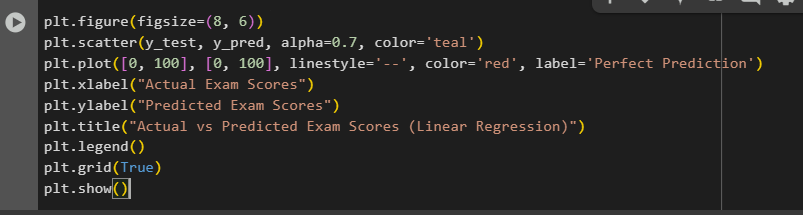
|  |  |  |
| --- | --- | --- |
| **Metric** | **Formula** | **Interpretation** |
| MSE |  | Average squared error (lower = better). |
| RMSE |  | Error in original units (e.g., "5.09 points"). |
| R² Score |  | % of variance explained (0.90 = 90%). |

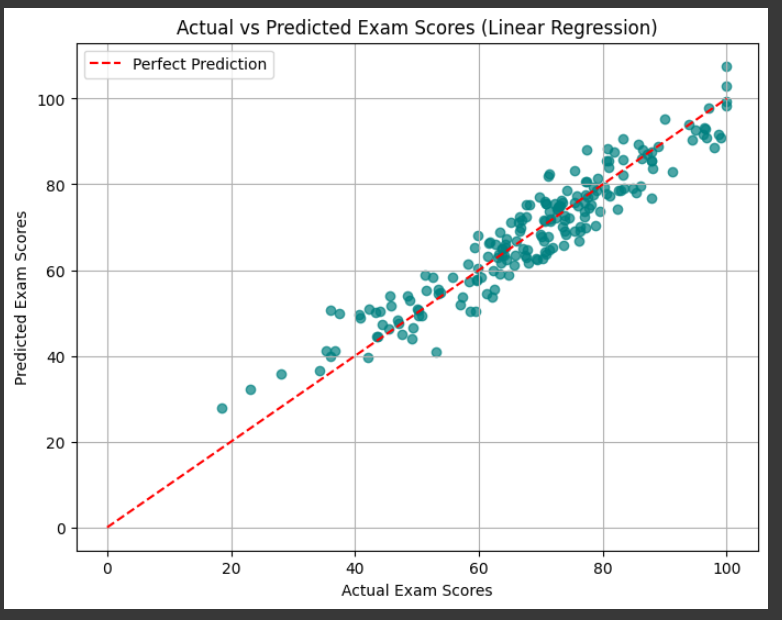
The results,



An RMSE = 5.09 means that on average, the model's predictions are off by ~5.09 points (in exam\_score units). On the other hand, R² = 0.90 explains 90% of the variance in exam scores which is a fairly excellent performance.

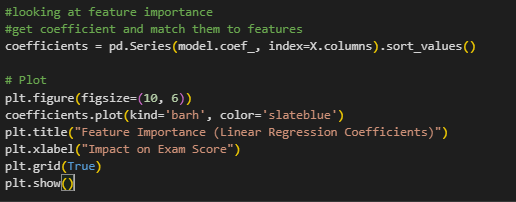
Next, trying to visualize the actual and predicted exam scores to see how close my model predictions were in reality, I developed a code that generates a scatter plot, comparing the actual exam score (y\_test) with the predicted exam score (y\_pred). Here is the code below.

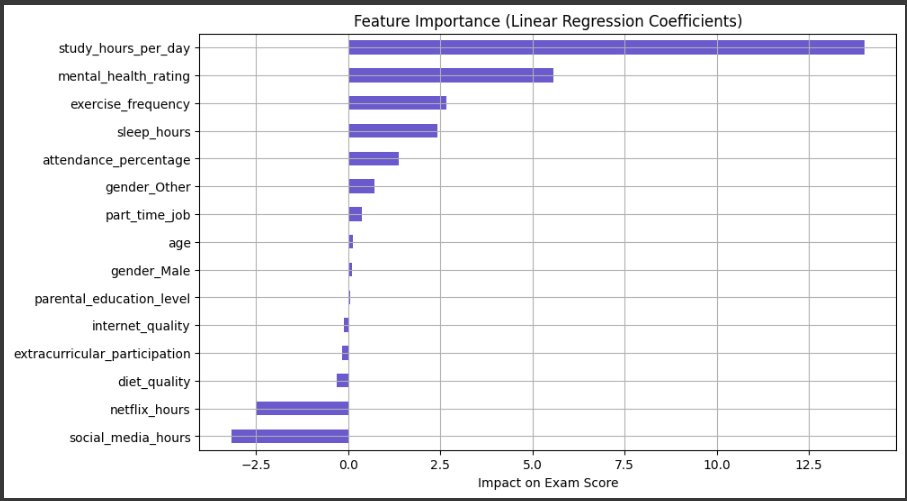


The resultant visualization   


In the image above, the dots represent the actual exam scores. the dots above the perfect prediction line are overpredicted while those below it are underpredicted. However, since the model's spread shows an RMSE of 5 points, the scatter is expected. Nevertheless, the Actual vs Predicted plot for our Linear Regression model shows a strong alignment with the ideal prediction line. Most predictions fall close to the 45° reference, suggesting the model is accurately estimating exam scores.

The next step was looking at feature importance. Hence we wrote a code that generated a horizontal bar plot to visualize the feature importance based on the coefficients learned by the linear regression model.

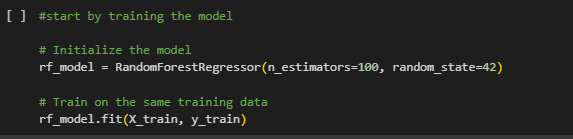




In the code and image above, each coefficient represents the change in the predicted exam score for a 1-unit increase in the feature. Given our scaled features pitting all inputs into the same scale, meaning that the size of each coefficient directly shows how important the variable is to the prediction, study hours per day is by far the most positively influential habit. In its interpretation, for each standard deviation increase in study time, exam scores increased by almost 14 points. clearly supporting the common idea that more studying leads to better academic performance - common sense now backed by data. Also, social media usage has the strongest negative impact on exams scores, suggesting that heavy social media use is distracting and cuts into study time and rest.

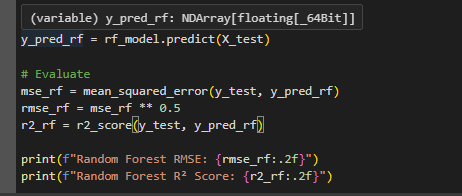
## Step 3. Training a Random Forest Model for Comparison

**`** training a random Forest (RF) model alongside the linear regression model serves several purposes. For starters, it helps handle non-linear relationships. While linear regression assumes a straight-line relationship between features and the target, Random Forest can capture complex, non-linear patterns that linear models might miss. Moreover, while linear regression is sensitive to outliers, RF predicts across many trees, reducing the impact of outliers.



By setting random\_state=42, the model ensures reproducibility. Training is performed on the same dataset (X\_train, y\_train) used for the linear regression, allowing for a fair comparison.

The code below performs predictions and evaluations for the trained Random Forest Model.



The results are as follows   


### Explanation of Key results

|  |  |  |
| --- | --- | --- |
| **Metric** | **Linear Regression** | **Random Forest** |
| RMSE | 5.09 | 6.25 |
| R² Score | 0.9 | 0.85 |

A Linear Regression model and Random Forest Regressor received equivalent data for training and testing purposes to determine model effectiveness. The Linear Regression model demonstrated superior performance than the Random Forest because it achieved better accuracy results and provided greater explanatory power.

The Linear Regression model achieved a lower RMSE of 5.09, compared to 6.25 for the Random Forest. This means that, on average, the Linear Regression model's predictions were approximately 1.16 points closer to the actual exam scores. Additionally, Linear Regression posted a higher R² score of 0.90, compared to 0.85 for the Random Forest, indicating that it explained 5% more variance in the students' academic performance.

All three input features together with exam\_score demonstrate a strong linear correlation within the provided dataset. The strong results from linear regression models suggest that the studied dataset lacks significant complex feature interactions which typically lead to Random Forest outperformance. The Random Forest model might have encountered opportunities for slight overfitting but such cases become unlikely without adjusting custom hyperparameters.

Random Forests provide useful results in data sets which display nonlinearity and threshold effects. A tree-based model will work efficiently when the data shows that scores suffer with less than two daily study hours yet additional benefits end after five hours of study time. The effective detection of nonlinear patterns can be achieved by adjusting the max\_depth or min\_samples\_leaf parameters or performing partial dependence analysis.

## Conclusion

The research initiative studied how life habits among university students influence academic outcomes by investigating their final exam scores. The structured data science workflow incorporated data loading followed by cleaning and exploration then proceeded to feature engineering and model development.

Analysis showed a strong positive link between study duration and exam results together with moderate negative associations between screen use for social media and Netflix. A substantial amount of the dataset remained clean after the processing phase since only the "parental\_education\_level" field contained empty values that were filled by mode imputation.

Modeling began with encoding all categorical variables so they could function properly while numerical features received the necessary scaling procedure. The analysis relied on the implementation of two regression models for evaluation purposes.

***Linear Regression***

RMSE: 5.09

R² Score: 0.90

***Random Forest Regressor***

RMSE: 6.25

R² Score: 0.85

Linear Regression produced superior results than Random Forest since it seems that academic performance relates to student habits in a primarily linear fashion. The linear regression model both delivered easy-to-understand coefficients and demonstrated that excessive study duration generates the highest positive effects but social media use creates negative outcomes.

The project proved the effectiveness of data science modeling to identify practical insights about academic results. Study time duration coupled with decreased social media usage has proven to deliver substantive performance improvements according to the obtained data.