UNSW
SYDNEY

Student Performance Modelling with Neural Networks

Rayhan Rahman

1. Problem Specification

This project aims to use neural networks to develop a supervised learning model to predict if a student will be an academically high performed based on socio-demographic, academic, and lifestyle features. A student will be considered a high performer if their final grade (G3) is 15 or higher out of 20. This defines this problem as a **binary classification** problem, where the response variable "G3_binary" takes the value 1 for high performers and 0 for non-high performers.

The input dataset consists of 395 observations from Portuguese secondary school students, each with 30 features. These features include parental education levels, access to educational support, alcohol consumption, time spent studying, number of past failures, and absences etc. I've included both categorical features (e.g. guardian, parent job, access to internet) as well as numeric features (e.g. age, travel time, study time). More information on these characteristics is covered in the **Dataset Collection** section.

Our main goal is to identify the key predictors of strong academic achievement and train the classification model to generalise this into unseen data. We could also potentially use these insights to inform schools of at-risk students, looking at predicted low-performers instead.

To evaluate the performance of this model, the fulling classification metrics will be used:

- Accuracy: Overall correctness of predictions
- Precision: Proportion of true positive predictions out of all positive predictions
- Recall (Sensitivity): Proportion of actual high performers that are correctly identified (true positives)
- F1-Score: Harmonic mean of precision and recall to balance false positives and false negatives
- ROC-AUC: Evaluates the model's discriminatory ability across different thresholds

2. Data Collection

The dataset being used for the project is the "Student Performance" database, extracted from the UC Irvine Machine Learning Repository. The data dictionary can be found in the appendix, **2.1**.

3. Data Cleaning and Preprocessing

Before the Exploratory Data Analysis was carried out, the data was cleaned, with an initial inspection confirming there were no missing values in the dataset. As "G1" and "G2" are highly correlated with G3 (G3 is the mean of the two), these rows were dropped to ensure the model does not rely on these proxy values for prediction.

All numerical variables were standardized using "StandardScaler" ensuring they have zero mean and unit variance, preventing features with larger numeric scales such as "absences" from disproportionately influencing the model.

Categorical variables on the other hand needed to be transformed using appropriate encoding techniques. Binary categorical features such as "sex" and "address" were converted using label encoding, while nominal variables were transformed using one-hot encoding to avoid imposing any ordinal relationships.

The dataset was then randomly shuffled to ensure that the initial order of the samples does not affect the model training, reducing any bias in the distribution of data. The shuffled data was then split into training (70%), validation (15%) and test (15%) sets, saving these pre-processed datasets as CSV files in a "processed data" folder to ensure consistency across al future analysis / modelling.

4. Exploratory Data Analysis

Exploratory data analysis was conducted on the pre-processed training dataset in order to reveal relationships between variables that we could use for future modelling. As shown in **Figure 4.1**, the dataset is quite imbalanced, with approximately 80% of the sample of students not achieving high performance.

To identify the key predictors in this dataset, we used a visualisation of the top 15 features holding the highest correlation with the target variable (**Figure 4.2**). Variables such as "Medu", "studytime", and "internet_yes" showed mild positive correlation with high academic performance, while features such as "failures", "Walc" and "goout" were negatively associated with good performance. Throughout all the features, the correlation coefficients remained moderate, indicating that there was no single dominant predictor, but rather a combination of contributing factors to high performance.

I also used box plots to illustrate the distributions of select features in order to observe their relationship with "G3_binary". From **Figures 4.3 – 4.6**, we can see that students with higher "studytime" and fewer "failures" are more likely to be high performers. Similarly, students who were frequently absent or had higher social activity would often perform worse academically. On the other hand, a student's health status appeared to have small predictive value as an indicator for academic performance, as the distributions were mostly similar between the two groups.

The pair plot (**Figure 4.7**) of "studytime", "failures", and "absences" shows a clear separation I the feature space between the low and high performers. However, there is minimal feature collinearity in the plots, showing that most of the features contribute independently from one another to the performance of students.

Overall, the findings from the exploratory data analysis show that high academic performance is very multifactorial, meaning it is influenced by a large combination of study behaviour and lifestyle choices. The EDA also helps to validate the pre-processed dataset, as we can observe how features were scaled, encoded, and it shows enough variation for effective model training.

5. Baseline Model: Logistic Regression

For this project's benchmark model, I chose to fit a logistic regression model to the pre-processed dataset. Logistic regression was chosen due to its simplicity and interpretability for binary classification tasks, such as predicting whether a student is a high performer or not.

The model was trained on the cleaned dataset using "train.csv", and then "val.csv" for validation. The standard logistic regression from "sklearn.linear_model" was implemented with the default parameters, as the purpose was to establish a simple and interpretable benchmark.

The fitted model achieved the following results on the validation dataset:

- Accuracy: 0.81
- ROC AUC: 0.74
- Precision (Class 1): 0.50
- Recall (Class 1): 0.18
- F1-Score (Class 1): 0.27

Looking at these results, we can see that the model did fairly well in identifying NON-high performers (class 0), with a precision of 0.84 and a recall of 0.96. However, it struggles to correctly identify high performers (class 1), with only 2 of the 11 correctly identified from the dataset. This imbalance is shown in the confusion matrix **Figure 5.1**, where false negatives far outweigh true positives.

The results from the logistic regression are indicative of a common challenge in imbalanced classification tasks. Despite a strong overall accuracy, the model is not as effective for identifying the minority class, which in this case is the "High Academic Performance" students. The deep learning model for this project will aim to improve the recall for high performers, allowing us to predict at a much greater accuracy.

6. Deep Learning Model: Dense Sequential Model

Model Outline

The first deep learning model implemented was a Dense Sequential Model. This architecture consists of a linear stack of layers, where the output of one layer is passed directly as the input to the next. Each neuron in each layer connects to all neurons in the subsequent layer, allowing the network to capture complex non-linear interactions. The model's simplicity makes it a strong starting point for binary classification tasks, as it is straightforward to construct and interpret, while being flexible enough to learn from both categorical and numerical features. As described in the data preprocessing stage, all categorical variables were one-hot encoded, and numeric features were standardised to ensure equal weight across dimensions. This architecture was chosen as a baseline deep learning model to compare against the logistic regression benchmark, particularly given the class imbalance challenge as Sequential Models can leverage class weighting to improve recall.

Hyperparameter Tuning

To enhance performance, systematic hyperparameter tuning was conducted across 16 candidate models, varying the number of hidden layers, neurons per layer, learning rate, and dropout rate. All models used ReLU activation in hidden layers and a sigmoid output layer for binary classification. The number of epochs was fixed at 50 with class weighting applied to address the imbalance in high and low performers.

From the results table, accuracy across models ranged from 60% to 78%, with recall varying between 0.18 and 0.73. Given the project's focus on identifying high-performing students, recall improvements were prioritised. Notably, Model 5 ([16, 8] layers, learning rate 0.001, no dropout) delivered the best balance with an F1-score of 0.50, recall of 0.73, and a ROC-AUC of 0.78. Other candidates such as Model 2 achieved slightly lower accuracy but competitive recall (0.64), illustrating the trade-off between precision and recall.

Overall, moderate architectures like [16, 8] outperformed deeper networks such as [64, 32], which struggled with overfitting. These findings underscore that careful tuning of architecture depth and learning rate is crucial for maximising recall while maintaining acceptable precision.

Model No.	Layers	Learning Rate	Dropout	Accuracy	Precision	Recall	F1-Score	ROC-AUC
1	[10]	0.001	0.0	0.600000	0.275862	0.727273	0.400000	0.771800
2	[10]	0.001	0.3	0.716667	0.350000	0.636364	0.451613	0.771800
3	[10]	0.010	0.0	0.700000	0.230769	0.272727	0.250000	0.628942
4	[10]	0.010	0.3	0.716667	0.285714	0.363636	0.320000	0.638219
5	**[16, 8]**	**0.001**	**0.0**	**0.733333**	**0.380952**	**0.727273**	**0.500000**	**0.784787**
6	[16, 8]	0.001	0.3	0.650000	0.291667	0.636364	0.400000	0.693878
7	[16, 8]	0.010	0.0	0.783333	0.400000	0.363636	0.380952	0.627087
8	[16, 8]	0.010	0.3	0.750000	0.250000	0.181818	0.210526	0.599258
9	[32, 16]	0.001	0.0	0.650000	0.250000	0.454545	0.322581	0.653061
10	[32, 16]	0.001	0.3	0.683333	0.318182	0.636364	0.424242	0.736549
11	[32, 16]	0.010	0.0	0.700000	0.181818	0.181818	0.181818	0.614100
12	[32, 16]	0.010	0.3	0.766667	0.363636	0.363636	0.363636	0.697588

7. Deep Learning Model: Wide and Deep Model

Model Outline

The second deep learning architecture applied was a **Wide & Deep Model**, implemented using the Keras Functional API. This hybrid design allows the model to combine the strengths of two learning pathways: a *wide branch* that memorises linear patterns and direct feature associations, and a *deep branch* that captures complex, non-linear interactions.

In this case, the wide branch consisted of one-hot encoded categorical features such as *school, internet access,* and *parental occupation*, which were fed directly into the output layer to preserve simple memorised rules. In parallel, the deep branch processed standardized numerical inputs such as *studytime* and *absences* through two hidden layers of 10–16 neurons, activated with ReLU functions and regularised using dropout. The outputs of both branches were concatenated into a shared representation, which was then passed through a final sigmoid-activated neuron to produce a binary prediction of whether a student was a high performer.

This architecture was chosen due to its theoretical suitability for problems with mixed categorical and numerical data, as it balances memorisation with generalisation.

Hyperparameter Tuning

To optimise model performance, eight Wide & Deep configurations were trained, varying the number of hidden layers (1 or 2), neurons per layer (10 vs. 16/8), learning rate (0.001 vs. 0.01), and **dropout rate** (0 or 0.3). Class weighting was applied to mitigate the dataset's imbalance, and early stopping was used to prevent overfitting.

The results showed ROC-AUC values up to 0.75 but generally weaker F1-scores compared to the sequential model. Model 2 ([10] neurons, learning rate 0.001, dropout 0.3) offered the strongest trade-off, with recall of 0.73, F1-score of 0.41, and ROC-AUC of 0.78. Meanwhile,

Model 7 ([16, 8] with learning rate 0.01, no dropout) achieved the highest accuracy (83%) but at the cost of lower recall (0.27).

These outcomes suggest that while the Wide & Deep design helped capture categorical interactions, the categorical features in this dataset provided limited incremental predictive value. As a result, the deep branch primarily drove performance, with the wide branch offering little additional benefit.

Model No.	Layers	Learning Rate	Dropout	Accuracy	Precision	Recall	F1-Score	ROC-AUC
1	[10]	0.001	0.0	0.583333	0.230769	0.545455	0.324324	0.617811
2	**[10]**	**0.001**	**0.3**	**0.616667**	**0.285714**	**0.727273**	**0.410256**	**0.784787**
3	[10]	0.010	0.0	0.766667	0.384615	0.454545	0.416667	0.710575
4	[10]	0.010	0.3	0.616667	0.269231	0.636364	0.378378	0.749536
5	[16, 8]	0.001	0.0	0.616667	0.269231	0.636364	0.378378	0.641929
6	[16, 8]	0.001	0.3	0.550000	0.233333	0.636364	0.341463	0.608534
7	[16, 8]	0.010	0.0	0.833333	0.600000	0.272727	0.375000	0.662338
8	[16, 8]	0.010	0.3	0.500000	0.193548	0.545455	0.285714	0.628942

8. Discussion

When comparing the two deep learning architectures, the Sequential Model clearly outperformed the Wide & Deep Model. The sequential approach achieved a higher F1-score (0.50 vs. 0.41) and ROC-AUC (0.78 vs. 0.75), demonstrating stronger discriminatory power and better balance between precision and recall. The Wide & Deep model, while theoretically advantageous for handling mixed feature types, struggled to deliver meaningful improvements due to the limited predictive contribution of categorical variables in this dataset. Both deep learning models, however, showed clear advantages over the logistic regression benchmark, which had a strong overall accuracy (81%) but a notably poor recall (0.27) for high performers. The sequential and Wide & Deep models significantly improved recall (0.73 and 0.73, respectively), ensuring more high-performing students were correctly identified, even if this came at the cost of slight reductions in precision. This highlights the effectiveness of deep learning in addressing class imbalance and capturing more complex patterns with the hyperparameter tunings than the baseline.

9. Ethical Concerns

Using deep learning models to predict student performance raises concerns around fairness, privacy, and interpretability. Socio-demographic features, such as parental education or access to resources, may inadvertently reinforce systemic inequalities, while the “black-box” nature of neural networks limits transparency and trust. Misclassifications risk stigmatizing students or denying them needed support, while overestimation could leave at-risk students without help. To mitigate these risks, such models should serve only as supportive tools, with safeguards like anonymization, bias audits, and human oversight to ensure responsible and equitable use.

Appendix

Data Dictionary

2.1 Data Dictionary

Variable	Description	Type	Values / Range
school	Student's school	Categorical	GP = Gabriel Pereira, MS = Mousinho da Silveira
sex	Student's sex	Binary	F = Female, M = Male
age	Student's age	Numeric	15–22
address	Home address type	Binary	U = Urban, R = Rural
famsize	Family size	Binary	LE3 = ≤3 members, GT3 = >3 members
Pstatus	Parent's cohabitation status	Binary	T = Together, A = Apart
Medu	Mother's education	Ordinal	0 = None, 1 = Primary, 2 = 5th–9th grade, 3 = Secondary, 4 = Higher education
Fedu	Father's education	Ordinal	0 = None, 1 = Primary, 2 = 5th–9th grade, 3 = Secondary, 4 = Higher education
Mjob	Mother's job	Categorical	teacher, health, services, at_home, other
Fjob	Father's job	Categorical	teacher, health, services, at_home, other
Reason	Reason for choosing this school	Categorical	home, reputation, course, other
guardian	Student's guardian	Categorical	mother, father, other
traveltime	Home to school travel time	Ordinal	1 = <15 min, 2 = 15–30 min, 3 = 30 min–1 hr, 4 = >1 hr
studytime	Weekly study time	Ordinal	1 = <2 hours, 2 = 2–5 hours, 3 = 5–10 hours, 4 = >10 hours
failures	Number of past class failures	Numeric	0–4
schoolsup	Extra educational support	Binary	yes / no
Famsup	Family educational support	Binary	yes / no
Paid	Extra paid classes within the course subject	Binary	yes / no
activities	Extra-curricular activities	Binary	yes / no
nursery	Attended nursery school	Binary	yes / no
Higher	Intends to pursue higher education	Binary	yes / no
internet	Internet access at home	Binary	yes / no
romantic	Currently in a romantic relationship	Binary	yes / no
Famrel	Quality of family relationships	Ordinal	1 (Very bad) – 5 (Excellent)
freetime	Free time after school	Ordinal	1 (Very low) – 5 (Very high)

Gout	Frequency of going out with friends	Ordinal	1 (Very low) – 5 (Very high)
Dalc	Weekday alcohol consumption	Ordinal	1 (Very low) – 5 (Very high)
Walc	Weekend alcohol consumption	Ordinal	1 (Very low) – 5 (Very high)
Health	Current health status	Ordinal	1 (Very bad) – 5 (Very good)
absences	Number of school absences	Numeric	0–93
G1	First period grade	Numeric	0–20
G2	Second period grade	Numeric	0–20
G3	Final grade	Numeric	0–20 (used to create binary target variable)
G3_binary	Binary label for high performers (G3 ≥ 15)	Binary	1 = High performer, 0 = Otherwise

In this model, the "G3_Binary" is the target variable, being a binary indicator for whether a student is a high performed or not. This shows this is a binary classification problem, with '1' representing a high performer and '0' representing non-high performers.

Exploratory Data Analysis Plots

Figure 4.1:

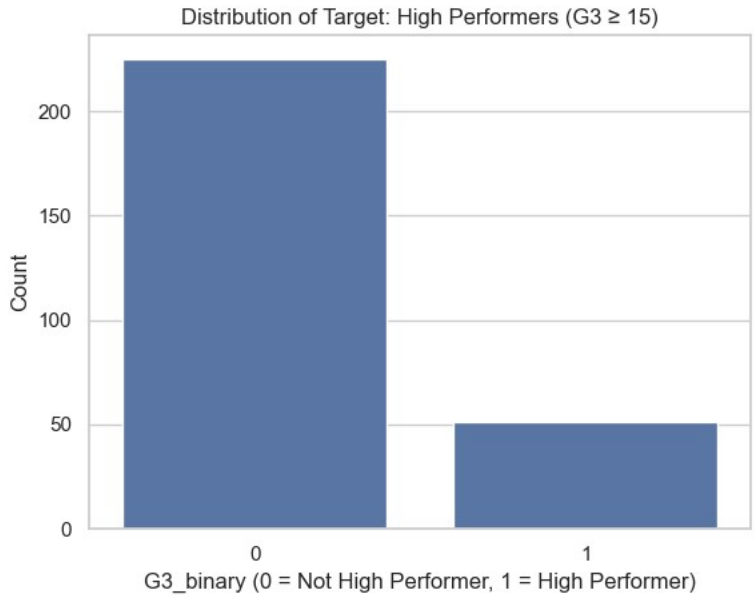


Figure 4.2:

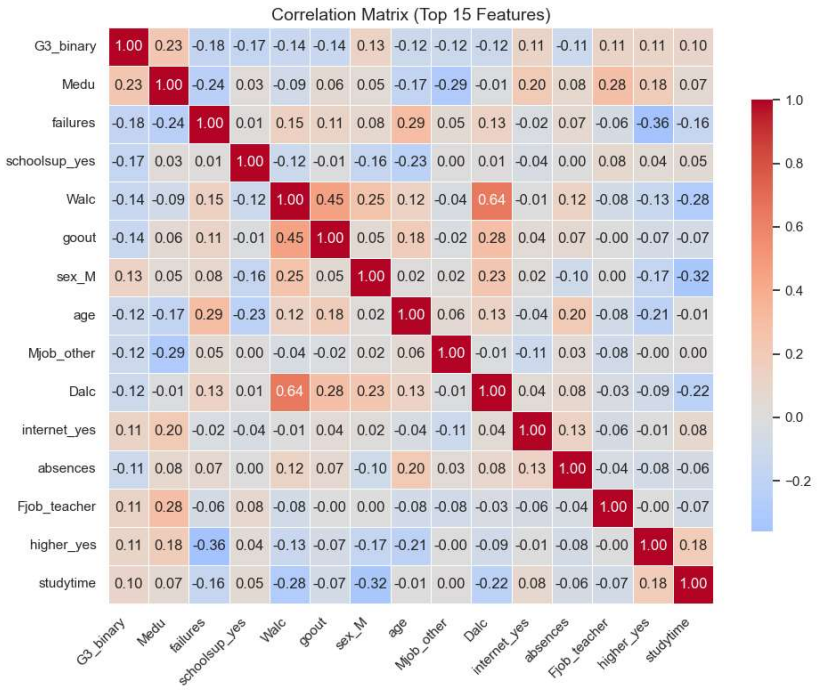


Figure 4.3:

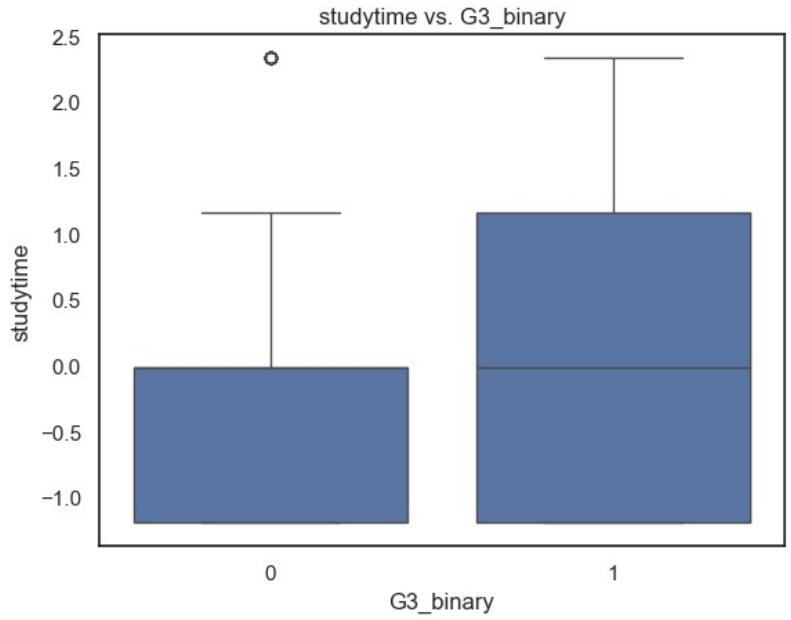


Figure 4.4:

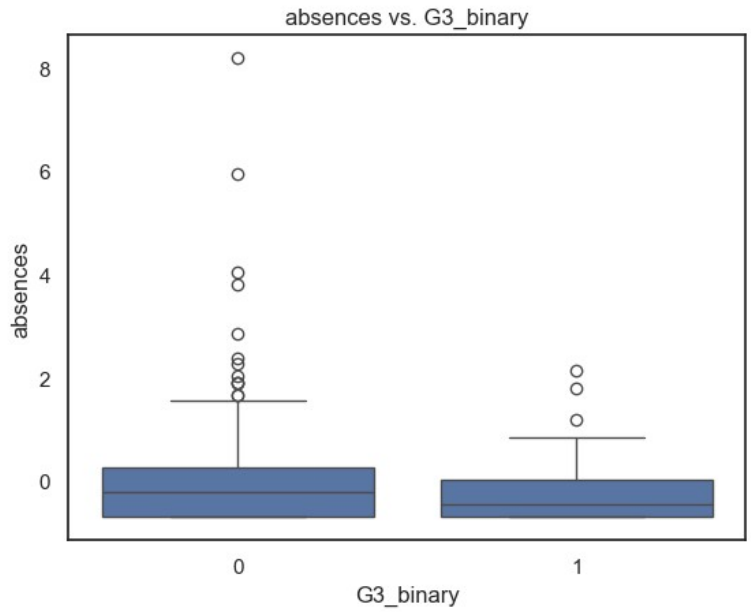


Figure 4.5:

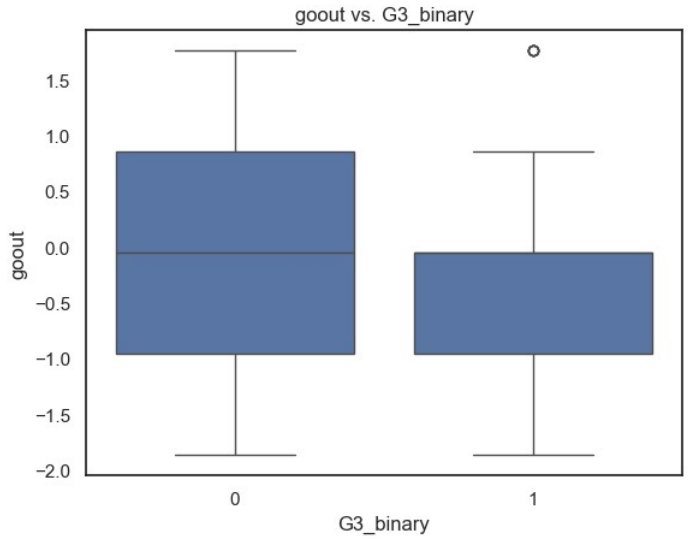


Figure 4.6:

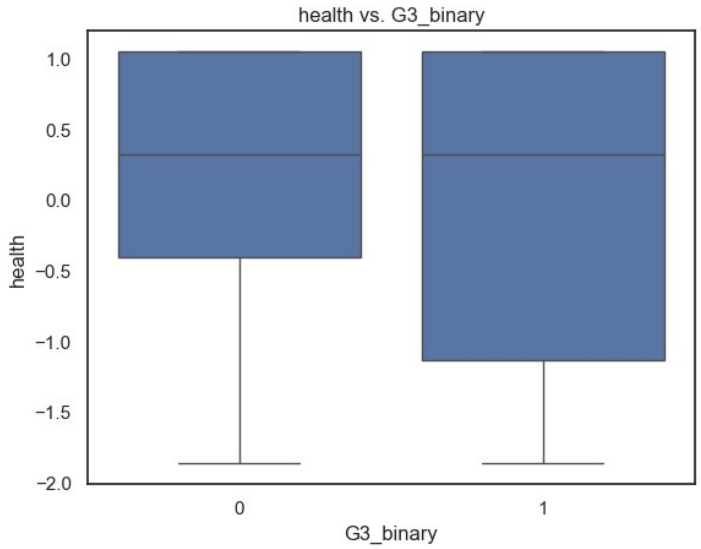
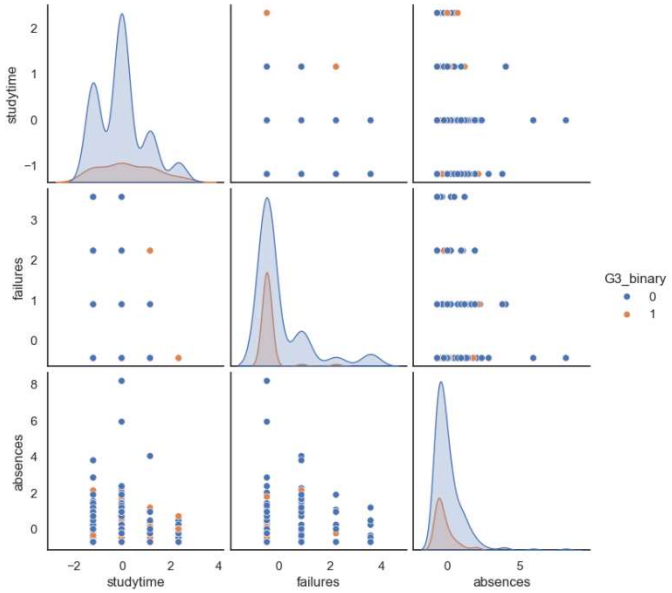
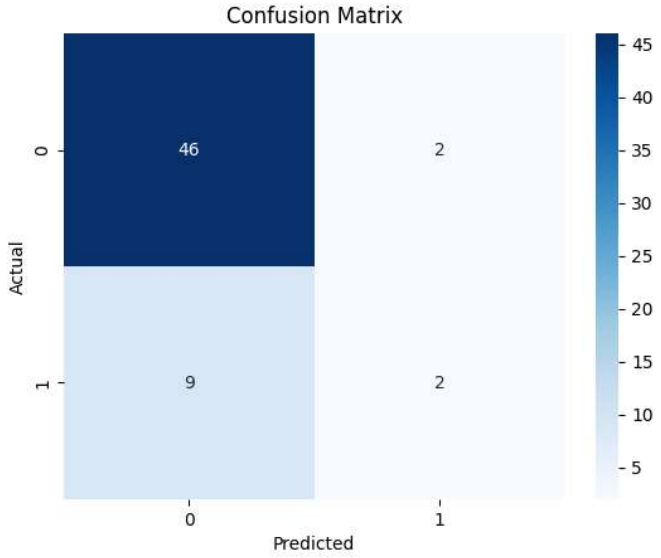


Figure 4.7:



Benchmark Model Results

Figure 5.1:


Confusion Matrix
46
2
9
2
0
1
Actual
Predicted
45
40
35
30
25
20
15
10
5