Assignment Cover Sheet

BSC Creative Computing

To be completed **electronically** by the student and submitted with each piece of work. Please upload this completed cover sheet via Turnitin

Assignment Title: Machine Learning (Model Development- 60%)

Tutor: Usman Ahmad

Student Name: Dhovie Costimiano

Student Number: 638148

Introduction

I developed a student performance prediction model to assist me to know the influence of various academic factors on student success. I needed an easy method to determine probable pass, probable fail, and probable at risk of a student. I have used an actual dataset, which contained attendance, assessment marks, not submitted on time, and participation rates. This was aimed at creating a model that would be that fast at categorizing the performance of a student based on these aspects. The key finding was that the model was capable of making good predictions of performance. I also made a Gradio app user-friendly so that people could easily test the model.

Problem Statement and Objectives.

The primary issue that I attempted to overcome was to find the students who could fail or be in danger until the release of final grades. Being a working student, I understand that it is not always possible to find time and remain focused, and most students are concerned with the same problem. This was aimed at creating a model that would predict academic performance through actual data. I also wanted the model to be easy and simple to use and hence it can assist students and teachers make better decisions at an early time. The goals were to identify useful features, learn an effective model and design an interface that is clean to make predictions. The model had to be useful and precise enough, but not so complex that individuals cannot comprehend the functionality of it. I also needed the project to be a manageable one as I could only work on it at night and during the weekends.

Data and Features

The data was based on an actual student performance CSV file. It included the records of various students (Not real students. Made or synthetic data)with such information such as the attendance percentage, the assessment results, the late submissions, and the level of participation. I made sure the data did not have missing values before training the model and removed any error. I coded the participation feature with label encoding because it was discrete. I then used the StandardScaler to scale the numeric features so that they would be on the same scale. I also divided the data into training and testing sets to estimate the model correctly. The last feature group was attendance, assessment score, late submissions and participation.

Model Development

Another model that I employed in the creation of a prediction model is a Random Forest Classifier. Random Forest is a powerful classification algorithm since it is a combination of numerous decision trees. It is also suitable in mixed data types and this fits this data. I began by dividing the data into training and testing data sets in a 80/20 proportion. This assisted me in testing the model without confusion of training and test data. To maintain consistency of the results I made use of a random state.

I then used StandardScaler on the numerical features. Scaling proved to be significant as the scores of attendance and assessment fell in different scales. Scaling ensured that the model was fair to every feature. LabelEncoder was also used to encode the participation feature. This transformed the values (Low, Medium, High) of the text into the numeric values.

In the model, I have considered the following hyperparameters: number of trees = 200, max depth = 10 and random state = 42. These environments have been selected so that both performance and overfitting can be balanced. I used the training data to train the model and used the test set to test it. The accuracy and classification report was checked after the training. The report reflected the ability of the model to forecast every class (Pass, At Risk, Fail). I also considered feature importance in order to check what factors had the greatest impact on the prediction. The most valuable features were attendance and assessment score. I did not make the model very complex due to the limited amount of time and a stable outcome.

Lastly, I created a Gradio interface to enable the easy use of the model. The interface allows one to type the student data and get a prediction immediately. This contributed to the project being more complete and practical. The predicted status and confidence score is also displayed in the app.

Results and Evaluation

The model scored highly in the test set in terms of accuracy. It was revealed that the model was good at predicting the Pass and the At Risk classes as indicated in the classification report. The Failed class also contained fewer samples hence the results were not as stable as in the Failed class. In general, this model was effective in the identification of students who could be in need of assistance. According to the results of the feature importance, attendance and assessment scores were the most significant factors. This was comparable with what we would see in actual life: students who are highly attending and assessed with good scores would pass. The Gradio app allowed the testing of various inputs and the model in terms of its responses. The findings were also reliable, and this contributed to confidence in the model.

Conclusion and Future Work

The project demonstrated that a basic model could be used to forecast the student performance by using actual data. Random Forest model was effective and provided helpful forecasts. The model was also easy to use through the Gradio app. In the next work, I would include more features such as study time, number of classes, or past grades. I would also gather additional information to enhance the effectiveness of the model in the classification of the "Fail" category. I may experiment with alternative algorithms such as the XGBoost or Logistic Regression. Finally, feature selection and hyperparameter tuning are another thing I would attempt to make the model more dependable.

References

1. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). *Scikit-learn: Machine Learning in Python*. Journal of Machine Learning Research, 12, 2825-2830.
2. Brownlee, J. (2020). *A Gentle Introduction to Random Forests in Machine Learning*. Machine Learning Mastery.
3. Gradio Documentation. (2024). *Gradio: Build and share ML apps*. Retrieved from https://gradio.app/
4. StandardScaler Documentation. (2024). *scikit-learn preprocessing*. Retrieved from https://scikit-learn.org/