Project: **Students Performance Evaluation**

Author: ***Halidu Abdulai***

Introduction: In educational institutions, instructors face the crucial task of assessing and evaluating student performance to tailor course materials effectively and enhance their learning experience. Manual evaluation of students' performance can be time-consuming, taking days or even weeks, particularly in large classes. This is where automation becomes invaluable.

In this project, we harness the power of machine learning algorithms to automate the assessment and evaluation of student performance, offering a more efficient alternative to manual methods. We provide machine learning models with a range of features, including student performance in projects, assignments, and online class interactions. These models predict the likely grade for each student in the given course.

Our dataset is derived from a fully online nine-week machine learning course hosted on the Moodle online learning management system. Our project's goal is to implement supervised learning techniques to predict students' final grades in an online course, ultimately enhancing the educational experience for both students and instructors.

Data preprocessing: The first step towards the project was to perform data preprocessing. Data preprocessing stage is a critical foundation for any machine learning project, ensuring that the machine learning models are provided with clean, structured, and relevant data for training and evaluation. By meticulously preparing the data, we set the stage for accurate and reliable predictions.

Data in its raw form could contain errors, inconsistencies, or missing values. During the preprocessing stage, we identify and rectify these issues. Missing values may be imputed using appropriate techniques, and any outliers or anomalies are addressed to maintain data integrity. Fortunately for us, the data did not contain any missing or inconsistent values. No outliers were present either.

The most important phase in the data preprocessing stage was selecting which features to include in training the models. The dataset had 48 features and 107 samples. Due to the high dimensionality and few samples, it was of utmost importance to select only the features that contributed much to the models predictions. For this, we identified that features such as the ID of the student and Week1_stat1 were not going to contribute to the model’s prediction and for this, they were removed. Also, the Week8_Total feature was in a way describing the grades of the students. By looking at this feature alone, the model could predict the students’ final grades with high accuracy so we found out that including this feature will introduce bias into

the model. We subsequently removed this feature as well. Other features such as the students interactions on the online platform was found to not be well defined for the students final grade, hence add such features were going to introduce some sort of complexities which would lead the developed model to learn unnecessarily complex patterns. For this reason, we removed those features as well.

Data analysis: After the preprocessing phase, the data was explored for insights. The feature that was of utmost importance to us to visualize was the target variable, the students' final grade. We wanted to gain an understanding on how the final students' final grade was distributed. For this, we made two different plots for the students' final grade distribution as shown below.

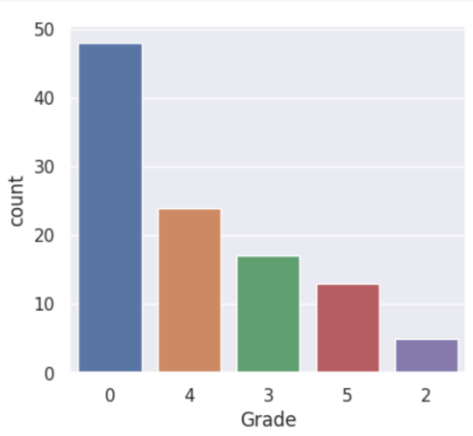

Figure 1: Bar chart of students' final grade distribution

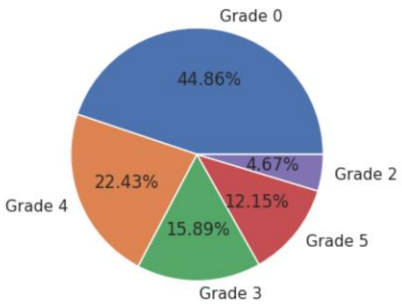

Figure 2: Pie chart of students' final grade distribution

From the charts above, the first question we asked ourselves was that how come so many students failed the course (got grade 0)? Was it that the course was very difficult to earn a passing grade or the students just didn't put in much effort to earn a passing mark? We also observed that even though almost 50% of the class failed the course, around 23% of the class got a grade 4. About 13 students also got a grade 5, which shows they were able to put in tremendous effort for such an achievement. We found out that a good number of the students also got average grades (grade 3). We found out also that very few of the students found themselves on the edge of the passing mark (grade 2).

To back up our claim that the Week8_Total feature could possibly introduce bias (cheating) in the model's predictions, we visualized its distribution and found out that it indeed would introduce bias in the model's prediction. We even concluded that there wouldn't be any need for a machine learning model to perform predictions on the students' final grade since we humans could just easily take a look at this feature and tell which grade a student will get.

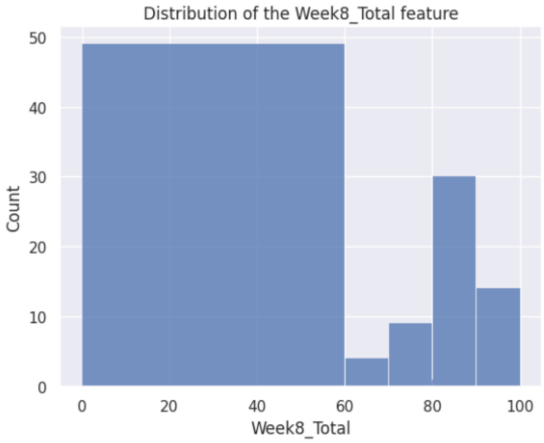

Figure 3: Histogram of Week_8_Total

The distributions of the selected features were also visualized. This visualization helped us to understand better how much marks each feature carried and its impact on the students' final grades.

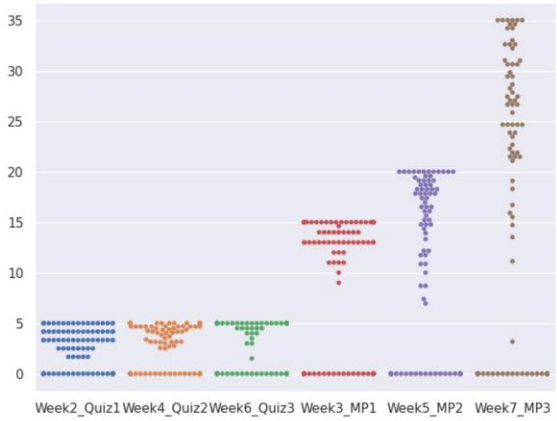

Figure 4: Swarm plot of the selected features.

Model performance and evaluation: We fed the preprocessed data to two different machine learning models and a summary of the models performance is presented next.

CLASSIFICATION SUMMARY OF THE SGD CLASSIFIER

	precision	recall	f1-score	support
0	1.00	0.90	0.95	10
2	1.00	0.00	0.00	1
3	0.33	1.00	0.50	3
4	1.00	0.60	0.75	5
5	1.00	0.33	0.50	3
accuracy			0.73	22
macro avg	0.87	0.57	0.54	22
weighted avg	0.91	0.73	0.74	22

CLASSIFICATION SUMMARY OF THE RANDOM FOREST CLASSIFIER

	precision	recall	f1-score	support
0	1.00	1.00	1.00	10
2	1.00	1.00	1.00	1
3	1.00	1.00	1.00	3
4	0.83	1.00	0.91	5
5	1.00	0.67	0.80	3
accuracy			0.95	22
macro avg	0.97	0.93	0.94	22
weighted avg	0.96	0.95	0.95	22

Figure 4: A summary of the models' performance

We found out that the model that performed best on the classification task was Random Forest classifier. It achieved an overall accuracy of 95%. Given the small size of the dataset, we found this performance to be descent. A noticeable thing worth mentioning is that all the two models achieved an incredible performance on instances belonging to grade 0. We found out that the reason was simple - there were about 45% of instances belonging to this class, and as such, the model was able to perform well on this class. The three most important features for the best model are: Week5_MP2, Week7_MP3, and Week6_Quiz3.

It turns out that mini project 3 and mini project 2 were very important in determining the students' final grade. A confusion matrix was made to understand the classes on which the models were performing well at, and the ones they had tough time in correctly predicting, and the results is as shown in the figures below.

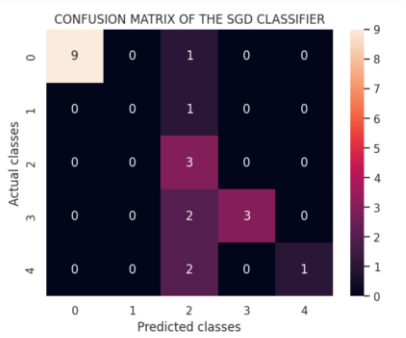

Figure 5: Confusion matrix of the SGD Classifier

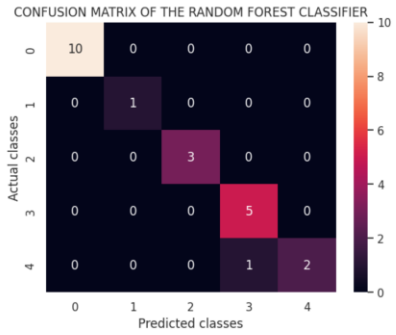

Figure 6: Confusion matrix of the Random Forest Classifier

From the confusion matrix above, we can observe the SGD model didn't perform that well on instances belonging to class 3 and class 4. However, it did well on predicting instances that belong to class 2(grade 3) and class 0 (grade 0). The random forest correctly identified all instances belonging to class 0 (grade 0) as well as correctly predicting the instances that belong to class 1 (grade 2), class 2 (Grade 3) and class 3 (Grade 4). It however made some mistakes in identifying the class 4 instances.

Each feature's importance and how much they contributed to the random forest classifier's prediction is given below.

{'Week5_MP2': 0.3502166844101747,

'Week7_MP3': 0.2996041190625922,

'Week6_Quiz3': 0.12241106240923523,

'Week4_Quiz2': 0.11636543609986086,

'Week2_Quiz1': 0.07134586123696002,

'Week3_MP1': 0.040056836781177024}

Conclusion: In this project, we explored the possibility of evaluating students' performance using supervised learning. We employed various classification algorithms using the Sklearn library.

We faced the difficulty of determining which features to include in training the model. After a thorough research and visualization, we were able to determine the most prevalent features for the classification task. Also, we had a tough time in determining the best hyper-parameters for the random forest classifier. We succeeded by visualizing the error produced by the model and using search grid, we were able to determine which hyper-parameters were best for the classifier.

Another challenge that we faced was that there very few samples (107 data samples) to be used for both training and testing. Due to the small dataset size, the performance of the models was not very high as the best model was able to achieve a classification accuracy of 95%. With more data samples, we could easily reach around 98% - 99% accuracy.

In future work, we aim to have more data samples at our disposure so that we can train models with very high prediction accuracies.