# Mini Project 2 : Students Performance

Harry Setiawan Hamjaya

*Abstract*— **This project entails the analysis and prediction of students' final grades in an online nine-week machine learning course using supervised learning techniques. The primary objectives of the project are split in three main folds. The first process is preparing the dataset for modeling from Data Visualization to Data Preprocessing. The second process employ two distinct classification model, in this project Logistic Regression or Random Forest for predicting students' final grades. The last process is evaluating the models with accuracy and F1 metrics.**

**Keywords: Data Visualization, Data Preprocessing, Logistic Regression, Random Forest, Accuracy Metric, F1 Metric**

## I. Introduction

This paper focuses on a unique project centered around online education. Specifically, data was collected from a nine-week machine learning course hosted on Moodle. The primary goal of this endeavor is to employ supervised learning techniques to predict the final grades of students in the digital classroom. Notably, the project allows the selection of two different supervised learning methods, offering diverse analytical perspectives.

The dataset includes information from 107 enrolled students, each represented anonymously. This data encompasses various aspects of their academic journey, including grades in mini projects, quizzes, peer reviews, and the overall grade. It also incorporates data on their interactions with the course. Mini project deadlines are set in weeks 3, 5, and 7, while quiz deadlines occur in weeks 2, 4, and 6.

This research explores the intersection of data science and education, aiming to understand the factors influencing student outcomes in the digital learning environment. Through a systematic process involving data preparation consist of data analysis and visualization, classification model training, performance assessment, and feature analysis, this study seeks to enhance online learning experiences.

## II. Data processing

### A. Dataset

The information in the dataset covers how students did in their online course. It includes their scores in 3 mini projects, 3 quizzes, 3 peer reviews, and their final grade. It also looks at what students did in the course, like when they viewed modules, submitted assignments, or participated in discussions. The mini projects were due in weeks 3, 5, and 8, while quizzes were due in weeks 2, 4, and 8.

There are different categories of actions, such as those related to the course content, assignments, grades, and forums. The dataset also contains 9 grades, such as Week2_Quiz1, Week3_MP1, and so on, and 36 logs like Week1_Stat0, Week1_Stat1, and so forth.

### B. Feature Data

There are several lists of features that must be created by our self, especially since we can't use all the given feature data that was given. Two features that must be dropped:

- Week1_Stat1: Because of all the values are equal.
- ID: Because of the values are string.
- Week8_Total: Because this is the benchmark of the best parameter model.

On the other hand, there are several features that needs additional efforts as they aren't provided in the csv, such as:

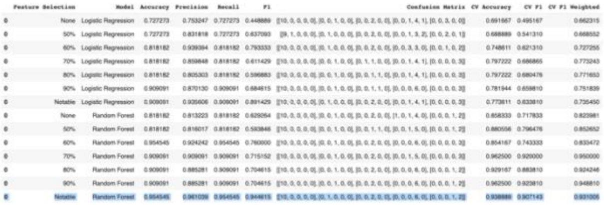
- All_MP: Sum of all Mini Project scores.
- All_PR: Sum of all Peer Review scores.
- All_Quiz: Sum of all Quiz scores.
- All_Score: Sum of all scores.

For the feature selection, we could consider several approaches such as:

- We chose the features that had more correlation to improve the model prediction performance. The floating number indicates the minimum correlation between features and label. It started with the feature minimum correlation of 0.5, 0.6, 0.7, 0.8, and lastly 0.9.
- The "Notable" features indicate the most important feature based on the definitions, in this case, we could assume grades are the most important features. Since it doesn't matter when someone open the lecture but didn't achieve the grade because late submission or probably didn't intend to do it compared to other who complete all the task while didn't click on any lecture.
- Finally, we still must compare with using all features, to determine whether the feature selection is effective or not.

At the first Iteration, we could try to benchmark the best model by applying the Feature Selection and Machine Learning towards the dataset with "Week8_Totaal". The challenge arises from the removal of Week8_Total which can only be used as benchmarking. According to the benchmark evaluation metrics Table 1, Random Forest with Notable features have the highest Accuracy, Precision, Recall and F1-Score which is around 95% (show with the highlight blue color). On the other hand, Logistic Regression best model is trained with Notable features with approximately 90% Accuracy, Precision, Recall and F1-Score Metrics.

Table 1: Benchmark Evaluation Metrics



| | Feature Selection | Model | Accuracy | Precision | Recall | F1 | Confusion Matrix | CV Accuracy | CV F1 | CV F1 Weighted |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | None | Logistic Regression | 0.727273 | 0.753247 | 0.727273 | 0.448889 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 4, 1], [0, 0, 3, 0, 0]] | 0.691667 | 0.495167 | 0.662315 |
| 0 | 50% | Logistic Regression | 0.727273 | 0.831818 | 0.727273 | 0.637093 | [[9, 1, 0, 0, 0], [0, 1, 0, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 3, 2], [0, 0, 2, 0, 1]] | 0.688889 | 0.541310 | 0.668552 |
| 0 | 60% | Logistic Regression | 0.818182 | 0.939394 | 0.818182 | 0.793333 | [[10, 0, 0, 0, 0], [0, 1, 0, 0, 0], [0, 0, 2, 0, 0], [0, 0, 3, 3, 0], [0, 0, 1, 0, 2]] | 0.748611 | 0.621310 | 0.727255 |
| 0 | 70% | Logistic Regression | 0.818182 | 0.859848 | 0.818182 | 0.611429 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 1, 1, 0, 0], [0, 0, 1, 4, 1], [0, 0, 0, 0, 3]] | 0.797222 | 0.686865 | 0.773243 |
| 0 | 80% | Logistic Regression | 0.818182 | 0.805303 | 0.818182 | 0.596883 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 1, 1, 0], [0, 0, 1, 4, 1], [0, 0, 0, 0, 3]] | 0.797222 | 0.680476 | 0.771653 |
| 0 | 90% | Logistic Regression | 0.909091 | 0.870130 | 0.909091 | 0.684615 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 1, 1, 0], [0, 0, 0, 6, 0], [0, 0, 0, 0, 3]] | 0.781944 | 0.659810 | 0.751839 |
| 0 | Notable | Logistic Regression | 0.909091 | 0.935606 | 0.909091 | 0.891429 | [[10, 0, 0, 0, 0], [0, 1, 0, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 4, 1], [0, 0, 0, 0, 3]] | 0.773611 | 0.633810 | 0.735450 |
| 0 | None | Random Forest | 0.818182 | 0.813223 | 0.818182 | 0.629264 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [1, 0, 1, 4, 0], [0, 0, 0, 1, 2]] | 0.858333 | 0.717833 | 0.823981 |
| 0 | 50% | Random Forest | 0.818182 | 0.816017 | 0.818182 | 0.593846 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 1, 1, 0], [0, 0, 1, 5, 0], [0, 0, 0, 1, 2]] | 0.880556 | 0.796476 | 0.852652 |
| 0 | 60% | Random Forest | 0.954545 | 0.924242 | 0.954545 | 0.760000 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 0, 6, 0], [0, 0, 0, 0, 3]] | 0.854167 | 0.743333 | 0.833472 |
| 0 | 70% | Random Forest | 0.909091 | 0.909091 | 0.909091 | 0.715152 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 5, 0], [0, 0, 0, 0, 3]] | 0.962500 | 0.920000 | 0.950000 |
| 0 | 80% | Random Forest | 0.909091 | 0.885281 | 0.909091 | 0.704615 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 0, 6, 0], [0, 0, 0, 1, 2]] | 0.929167 | 0.883810 | 0.924246 |
| 0 | 90% | Random Forest | 0.909091 | 0.885281 | 0.909091 | 0.704615 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 0, 6, 0], [0, 0, 0, 1, 2]] | 0.962500 | 0.923810 | 0.948810 |
| 0 | Notable | Random Forest | 0.954545 | 0.961039 | 0.954545 | 0.944615 | [[10, 0, 0, 0, 0], [0, 1, 0, 0, 0], [0, 0, 2, 0, 0], [0, 0, 0, 6, 0], [0, 0, 0, 1, 2]] | 0.938889 | 0.907143 | 0.931006 |

## III. DATA ANALYSIS

For the visualization analysis for this analysis:

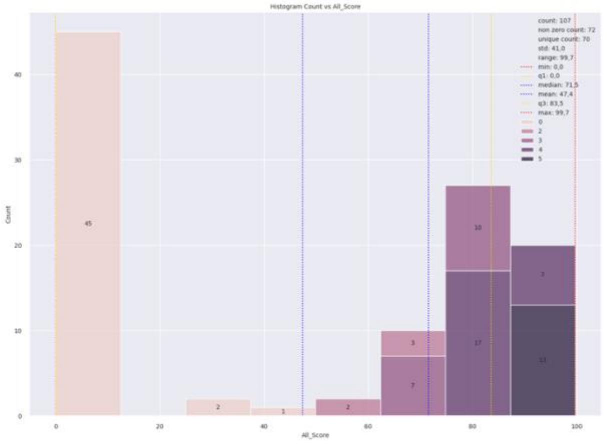


Figure 1: The distribution of grades toward All Scores

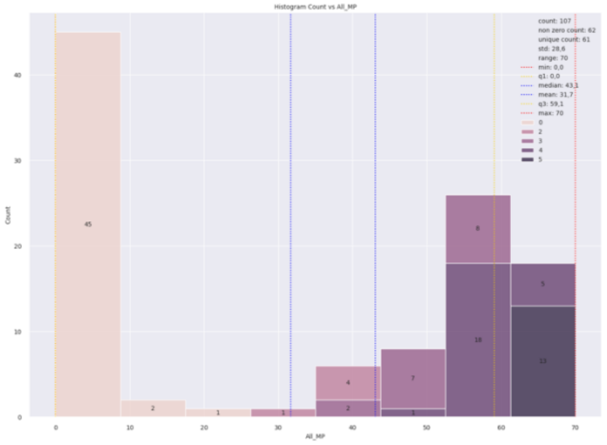


Figure 2: The distribution of grades toward All MP

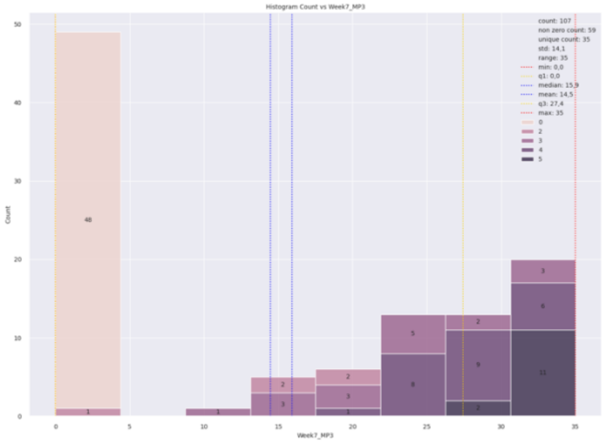


Figure 3: The distribution of grades toward Week7_MP3

From figure 1 all score distribution, figure 2 all Mini Project Score distribution, and figure 3 the Mini Project 3 score distribution (the single largest score that could be achieved in this class), we can easily note that there is a significant difference for the people who get grade either 0 or 5 because of they are in the other side of each distribution. Compared to such the status 0, 1, 2, 3 features respective to figure 4, 5, 6, 7.

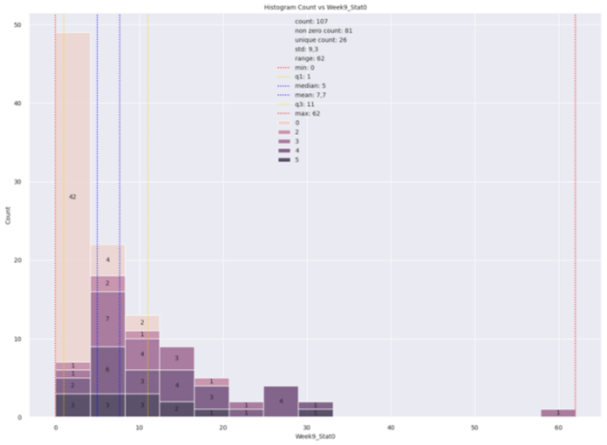


Figure 4: The distribution of grades toward Week9_Stat0

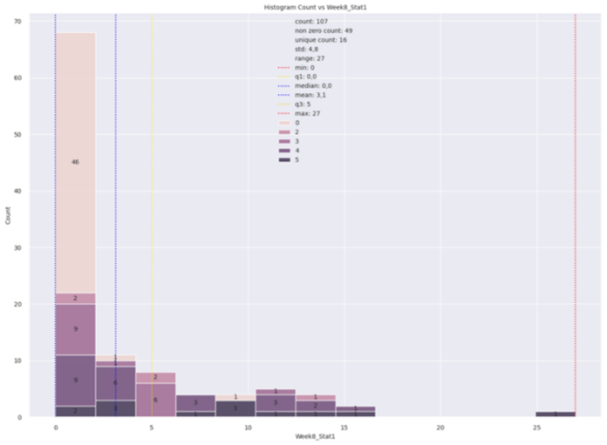


Figure 5: The distribution of grades toward Week8_Stat1

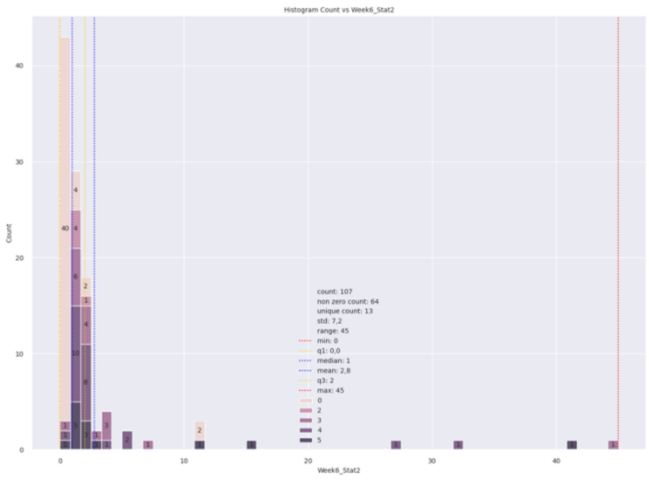


Figure 6: The distribution of grades toward Week6_Stat2

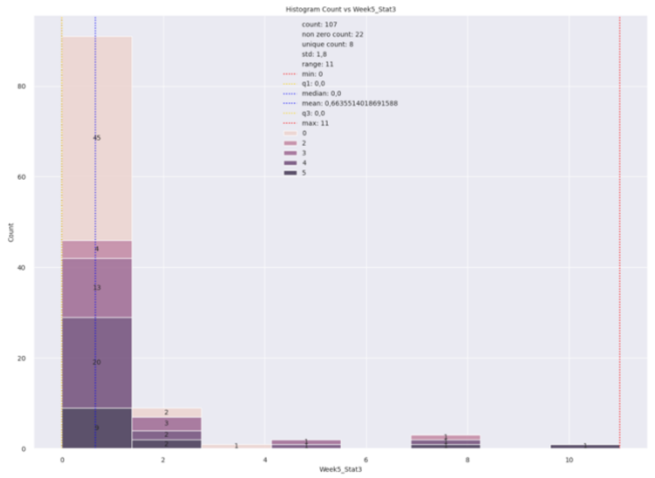


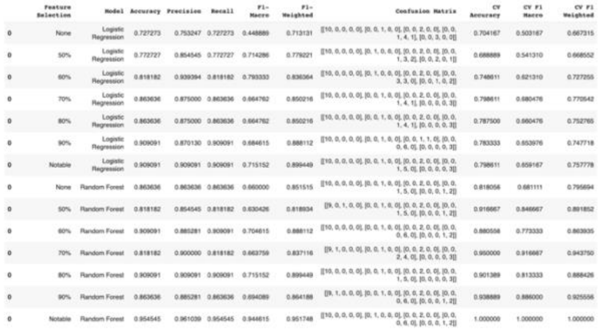
Figure 7: The distribution of grades toward Week5_Stat3

Based on the visualization, it is evident that the most important feature would be around "Grades" which consist of Total Scores, Total Mini Project, and Largest Mini Project Scores. Then we can proceed with the training and evaluation.

## IV. Training and Evaluation

Every batch of training would produce the evaluation table such as the Table 2 as the Best Model Evaluation Metrics with around 95% Accuracy, Precision, Recall and F1-Score on test dataset and 100% for the Train Dataset with Cross Validation Method. The idea of each batch training was reducing the number of "Notable" features such that we could achieve at least the same result as the benchmark result at table 1.

Table 2: Best Model Evaluation Metrics



|  | Feature Selection | Model | Accuracy | Precision | Recall | F1-Macro | F1-Weighted | Confusion Matrix | CV Accuracy | CV F1 Macro | CV F1 Weighted |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | None | Logistic Regression | 0.727273 | 0.753247 | 0.727273 | 0.448889 | 0.713131 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 4, 1], [0, 0, 3, 0, 0]] | 0.704167 | 0.503167 | 0.667315 |
| 0 | 50% | Logistic Regression | 0.772727 | 0.854545 | 0.772727 | 0.714286 | 0.779221 | [[10, 0, 0, 0, 0], [0, 1, 0, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 3, 2], [0, 0, 2, 0, 1]] | 0.688889 | 0.541310 | 0.668552 |
| 0 | 60% | Logistic Regression | 0.818182 | 0.939394 | 0.818182 | 0.793333 | 0.836364 | [[10, 0, 0, 0, 0], [0, 1, 0, 0, 0], [0, 0, 2, 0, 0], [0, 0, 3, 3, 0], [0, 0, 1, 0, 2]] | 0.748611 | 0.621310 | 0.727255 |
| 0 | 70% | Logistic Regression | 0.863636 | 0.875000 | 0.863636 | 0.664762 | 0.850216 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 4, 1], [0, 0, 0, 0, 3]] | 0.798611 | 0.680476 | 0.770542 |
| 0 | 80% | Logistic Regression | 0.863636 | 0.875000 | 0.863636 | 0.664762 | 0.850216 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 4, 1], [0, 0, 0, 0, 3]] | 0.787500 | 0.660476 | 0.752765 |
| 0 | 90% | Logistic Regression | 0.909091 | 0.870130 | 0.909091 | 0.684615 | 0.888112 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 1, 1, 0], [0, 0, 0, 6, 0], [0, 0, 0, 0, 3]] | 0.783333 | 0.653976 | 0.747718 |
| 0 | Notable | Logistic Regression | 0.909091 | 0.909091 | 0.909091 | 0.715152 | 0.899449 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 5, 0], [0, 0, 0, 0, 3]] | 0.798611 | 0.659167 | 0.757778 |
| 0 | None | Random Forest | 0.863636 | 0.863636 | 0.863636 | 0.660000 | 0.851515 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 5, 0], [0, 0, 0, 1, 2]] | 0.818056 | 0.681111 | 0.795694 |
| 0 | 50% | Random Forest | 0.818182 | 0.854545 | 0.818182 | 0.630426 | 0.818934 | [[9, 0, 1, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 5, 0], [0, 0, 0, 1, 2]] | 0.916667 | 0.846667 | 0.891852 |
| 0 | 60% | Random Forest | 0.909091 | 0.885281 | 0.909091 | 0.704615 | 0.888112 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 0, 6, 0], [0, 0, 0, 1, 2]] | 0.880556 | 0.773333 | 0.863935 |
| 0 | 70% | Random Forest | 0.818182 | 0.900000 | 0.818182 | 0.663759 | 0.837116 | [[9, 1, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 2, 4, 0], [0, 0, 0, 0, 3]] | 0.950000 | 0.916667 | 0.943750 |
| 0 | 80% | Random Forest | 0.909091 | 0.909091 | 0.909091 | 0.715152 | 0.899449 | [[10, 0, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 1, 5, 0], [0, 0, 0, 0, 3]] | 0.901389 | 0.813333 | 0.888426 |
| 0 | 90% | Random Forest | 0.863636 | 0.885281 | 0.863636 | 0.694089 | 0.864188 | [[9, 1, 0, 0, 0], [0, 0, 1, 0, 0], [0, 0, 2, 0, 0], [0, 0, 0, 6, 0], [0, 0, 0, 1, 2]] | 0.938889 | 0.886000 | 0.925556 |
| 0 | Notable | Random Forest | 0.954545 | 0.961039 | 0.954545 | 0.944615 | 0.951748 | [[10, 0, 0, 0, 0], [0, 1, 0, 0, 0], [0, 0, 2, 0, 0], [0, 0, 0, 6, 0], [0, 0, 0, 1, 2]] | 1.000000 | 1.000000 | 1.000000 |

Starting from 13 features from the Notable Feature List: 'All_Score', 'All_MP', 'Week7_MP3', 'All_PR', 'All_Quiz', 'Week5_MP2', 'Week3_MP1', 'Week3_PR1', 'Week5_PR2', 'Week7_PR3', 'Week2_Quiz1', 'Week4_Quiz2', 'Week6_Quiz3'. We can try reducing the number of the Notable Feature, 1 by 1 until we achieve the best Evaluation Metrics.

In this case, using only 5 features is the best for Random Seed = 42 with Split 80-20. Standardization is basically an option for model such as Random Forest and Decision Tree, but because the other model is Logistic Regression, the standardization became a mandatory step. By stating up the random state of each model as 0, we would like the machine learning model could be easily replicated for the other purpose.

The visualization for evaluation metric would be start with the ROC Curves, we can easily notice the only problem for Logistic Regression is with the class 2, around 5% from being perfect with the AUC Scores, On the other hand, Random Forest performs better by having the perfect AUC Scores.

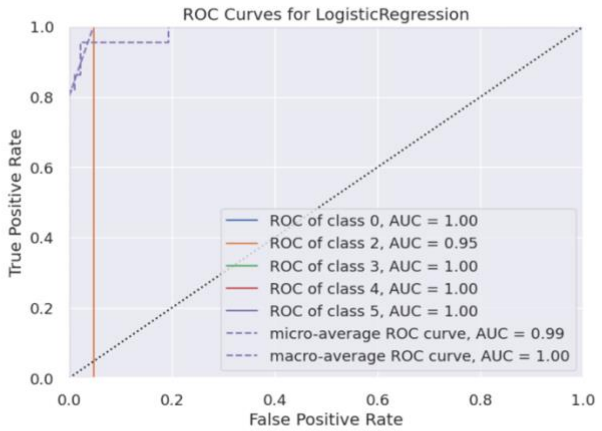


Figure 8: The ROC curves for Logistic Regression

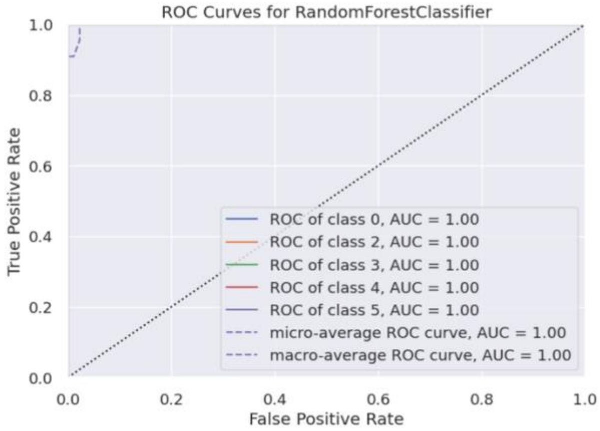


Figure 9: The ROC curves for Random Forest

The other visualization for evaluation metric would be Precision-Recall Trade off, Logistic Regression is having 91% average precision scores from figure 10. while Random Forest is having 99% average precision scores from figure 11.

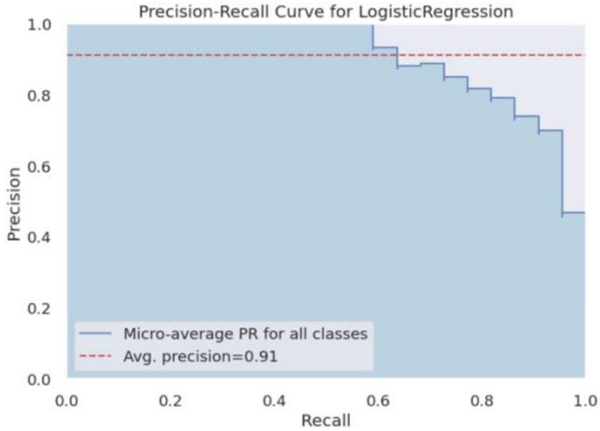


Figure 10: The Precision-Recall Tradeoff for Logistic Regression

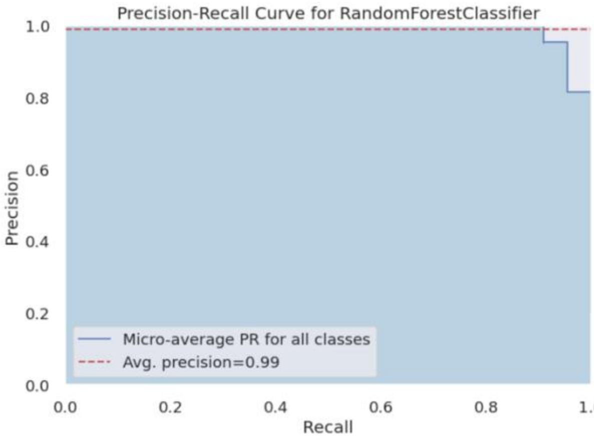


Figure 11: The Precision-Recall Tradeoff for Random Forest

The last visualization for evaluation metric would be Cross Validation Distribution since we are considering two important metrics which are accuracy and f1, we would have figures 12 and figures 13 respectively for both Accuracy and F1.

From Cross Validation (Training Point of View):

- The feature selection performs well for Logistic Regression, because the range of the interquartile of the model with feature selection (Light Color) especially for 70%, 80% and "Notable" improve based on the boxplot graph compared to the model with all features (Bright Color).
- Furthermore, the feature selection performs good for Random Forest, because the range of the interquartile of the model with feature selection (Light Color) especially for 70% and the notable is more compressed based on the boxplot graph compared to the model with all feature (Bright Color).

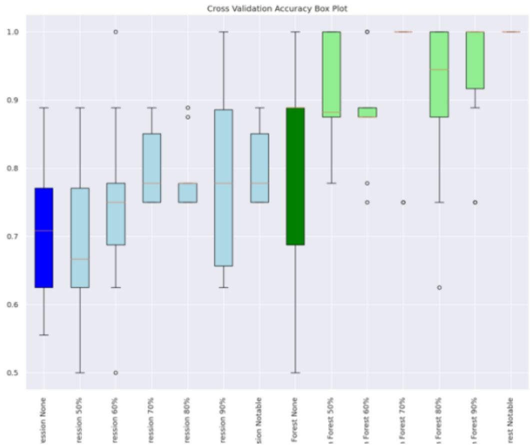


Figure 12: The Average Accuracy Cross Validation Distribution

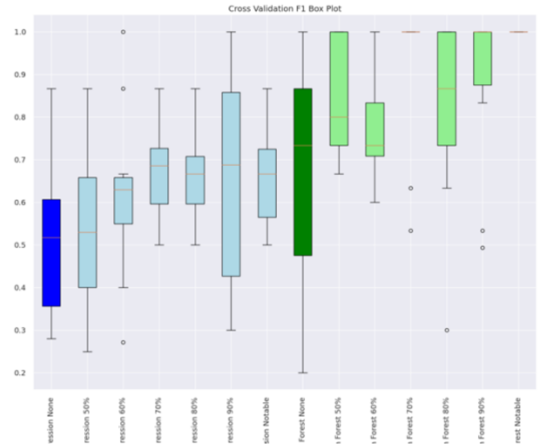


Figure 13: The Average F1-Score Cross Validation Distribution

From Test Data

- According to the evaluation metrics chart, Random Forest with Notable features have the highest Accuracy, Precision, Recall and F1-Score which is around 95%.
- On the other hand, Logistic Regression best model is trained with Notable features with approximately 90% Accuracy, Precision, Recall and F1-Score Metrics.
- Moreover, from the Cross Validation Average and Boxplot Comparison, It is evident that Random Forest with Notable Features has a higher score at the Train and Validation Dataset as well as the Test Dataset.

The final analysis will be the feature usage, with figure 14 and 15 respectively to each Logistic Regression and Random Forest feature importance.

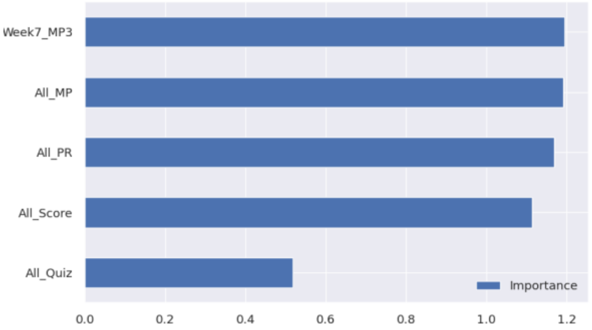


Figure 14: Logistic Regression Feature Importance

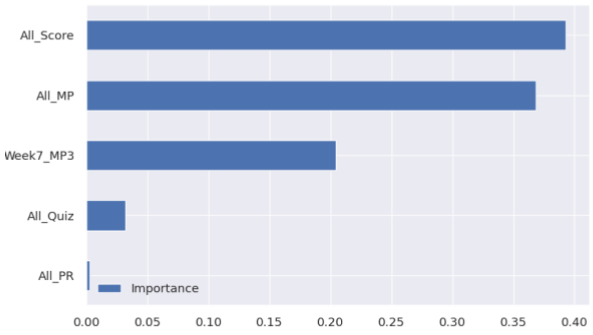


Figure 15: Random Forest Feature Importance

There is a difference feature usage between these models. As the best models Random Forest is prioritizing:

1. All_Score
2. All_MP
3. Week7_MP3

On the other hand, Logistic Regression is prioritizing:

1. Week7_MP3
2. All_MP
3. All_PR
4. All_Score

By utilizing more features doesn't mean Logistic Regression is a better model because of the Average score of Random Forest is still beating it.

## V. Conclusion

In conclusion, Student grade performance prediction is a multiclass classification problem. Starting from analysing the feature data, such as removing the non-unique features, remove duplicates and Null value is the first important step that should be taken by the Data Scientists. The second step is eliminating features that aren't important for the modelling. In this second step, there are several ways such as considering the correlation between features and labels, using the wrapper model such as random forest or decision tree to extract the feature importance or by the definition of the feature if we as the data scientist understand the domain to eliminate certain feature. Easily demonstrated if we can proof either by visualizing the dataset distribution or testing the dataset with statistical test.

One bottleneck that could be faced by every people who try to achieve the 90% Accuracy was the lack of important feature. As we know the definition of each single feature in the dataset is a single score, then we have to sum some of them to achieve the total. Since I believed that there isn't any machine algorithm that could sum the features, that's why besides Feature Selection, Feature Engineering is also playing an important roles. If we grasp the sense of the features, engineering the features wouldn't be that hard.

For the training aspect, I split the data to 80-20 for the Train and Test split. It is also good to notice that the train should be split as Train and Validation using the K-Fold Cross Validation method in order to ensure the stability of the trained models. In the evaluation criteria, I try to use the Accuracy and F1 as final evaluation. Accuracy is important but the real deal for an Imbalanced dataset is F1 score. It was because the spread of the student grade is highly imbalanced especially for Grade 0.

The best model is the Random Forest Model with average 100% Accuracy and F1 score toward Train-Validation Dataset and around 95% Accuracy and F1 toward Test Dataset. The top three feature importance are All_Score which is the summation of all Mini Projects, Peer Reviews, and Quizzes, All_MP is the summation score of all Mini Project from week 3, 5, and 7, and Week7_MP3 itself because it has the largest point/score contribution compared to all other grades indicator. Although Logistic Regression share the same three important features, but because of the usage of All_PR, make a slight difference between the result and affecting the model performance evaluations.

Although there might be a lot of room for improvement, especially in areas such as Grade8_Total shouldn't be in the specification, because it was so important for the Data Scientist to recognize by himself. I believe the overall project has provided invaluable experience as a data scientist. This experience includes analysing the raw data, pre-processing, visualizing and engineering the data, training and evaluating the models, and documenting the key points that have been learned during the second mini-project.