Predicting student's outcome in Virtual Learning Environment

Supervisor:

Dr. Hamid Karimi

By:

Zahed Golabi

December 2023

Table of Contents

1.Introduction 1
2. Pre-processing 1
2.1 Filling 1
2.2 Replacing 1
2.3 Feature Extraction 2
2.4 Dropping 2
2.5 Encoding 2
3. Exploratory Data Analysis (EDA) 2
3.1 Data statistics 2
3.2 Features' distribution 2
3.3 Features' correlation with label 5
3.4 Features' scatter with label 6
3.5 Categorical features with total_activities 8
3.6 Outlier detection 9
3.7 Feature importance 10
4. Model Selection 13
4.1 Binary classification 14
4.2 Multi-class classification 16
4.2.1 Class is balanced 17
4.2.2 Class is imbalanced 18
5. Conclusion 19

1.Introduction

This document aims to provide a comprehensive solution for a data science task. The task requires us to predict the final result of students in a virtual learning environment with which students interact. The provided dataset consists of 29227 records and 810 columns. Each record reveals a student's demographic data, interaction data quantified as 'clicks' related to a specific activity, and the final course result.

Below is a brief description of each column:
code_module: The unique id for each course [AAA, BBB, CCC, DDD, EEE, FFF, GGG]
code_presentation: The semester when the course was presented [2013B, 2013J, 2014B, 2014J]
id_student: The student's unique id
gender: The student's sex
region: The student's region
highest_education: the student's literacy
imd_band: Index of Multiple Deprivation
age_band: The age range of a student
disability: whether a student is disabled
Columns 10–809: Click data pertaining to 20 types of activities for 40 weeks. For example, 'dataplus_1' records the number of clicks on dataplus in the VLE for week 1.
Column 810: Final course result: 'Distinction,' 'Pass,' 'Fail,' or 'Withdrawn' A withdrawn status signifies that the student dropped out of the course.

2. Pre-processing

The first step to mining information out of data is to know our domain and make sure data is clean! We have to gain a comprehensive understanding of the attributes we are working with and make sure the data feeding to our model is edible! There are different ways to conduct this, and it's called pre-processing.

Pre-processing converts the raw data to something that is usable and efficient for our prediction model. Dealing with null or duplicated values, replacing or removing abnormal values, feature extraction, feature selection, and other techniques are some preprocessing methods.

2.1 Filling

Some columns have missing values that need to be filled. In our dataset, column 'imd_band' has 1054 empty rows. To fill them, we compare the 'region' of empty rows with the 'region' of the whole dataset and fill empty rows with the mode 'imd_band' of rows that have a similar 'region'.

2.2 Replacing

To enhance clarity, 'code_presentation' values were replaced by more clear ones; therefore, ['2013B', '2013J', '2014B', '2014J'] was substituted for ['2013-Feb', '2013-Oct', '2014-Feb', '2014-Oct'] respectively. Also, entries for 'imd_band' with '20-Oct' were replaced by '10-20%'.

2.3 Feature Extraction

Some new features have been created so as to build a robust and accurate model and to gain better performance for the model. To do so, 22 new features were added to the dataset. For each activity, the sum of clicks during 40 weeks was calculated, which resulted in 20 new features. In addition, the total activities in which each student participated and the total clicks for the whole 40 weeks were computed.

2.4 Dropping

After the feature extraction phase, 32 features that have precious information have remained.

2.5 Encoding

There are some categorical features that have to be encoded to feed the model. For example, the 'gender' column with values 'F' and 'M' was converted to 0 and 1.

3. Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a process of analyzing and summarizing data sets to gain insights and identify patterns in the data. EDA is used to understand the data, detect anomalies, and identify relationships between variables. It is an important step in the data analysis process, as it helps identify potential problems with the data and formulate hypotheses for further analysis.

3.1 Data statistics

Below are some general and statistical information about our dataset:

Data Shape: (29227, 32), 29227 rows, 32 columns

Data statistics of some features:

	total_subpage_clicks	total_url_clicks	total_activities	total_clicks
count	29227	29227	29227	29227
mean	1.340507	29.442673	78.890273	1164.660075
std	0.575191	66.815555	55.299983	1543.697493
min	0	0	7	7
25%	1	5	30	185
50%	1	17	71	605
75%	2	37	118	1526
max	5	5106	281	22000

3.2 Features' distribution

Below are some visualizations of the distribution of features.

Figure 1. Label distribution

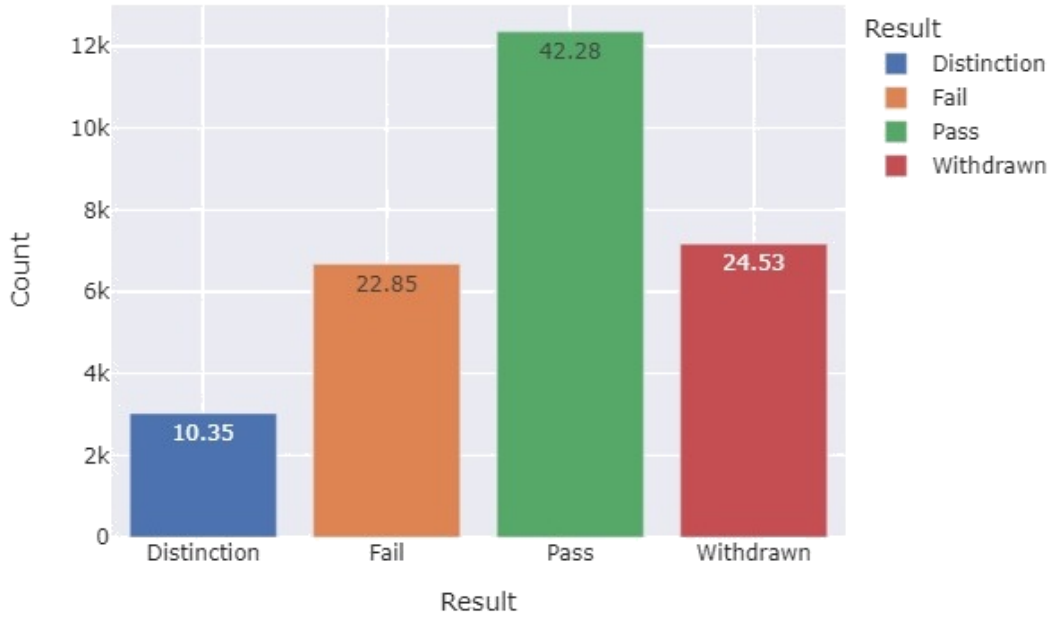

Fig. 1. Distribution of our target feature (**Label**)

As we can see in Figure 1, 42.28 percent of students passed the courses, while 22.85 percent failed them.

Figure 2. Region distribution

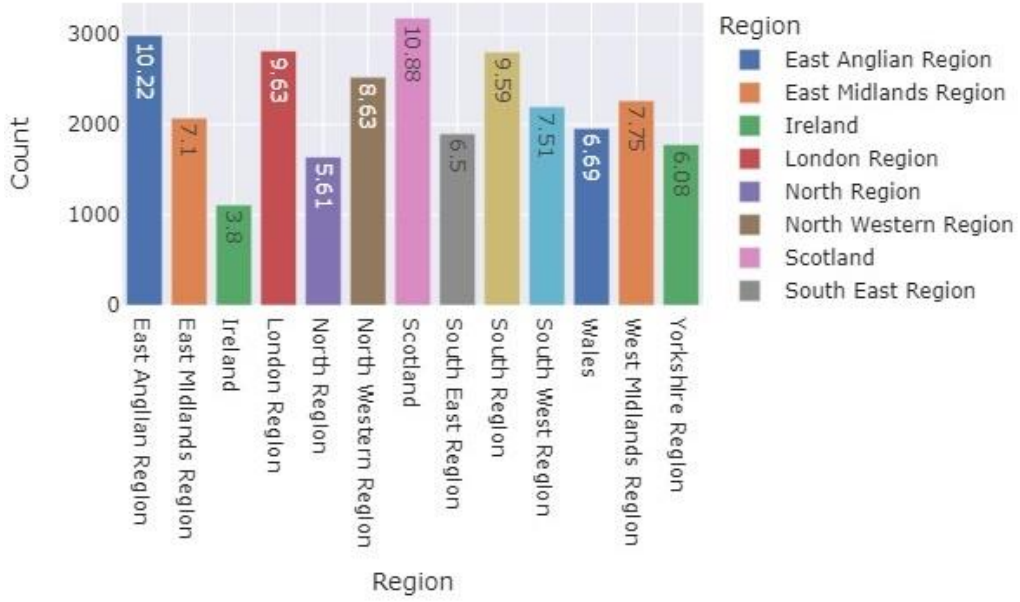

Fig. 2. Distribution of region

As shown in Figure 2, most students who took the courses are from Scotland, and the least are from Ireland.

Figure 3. imd_band distribution

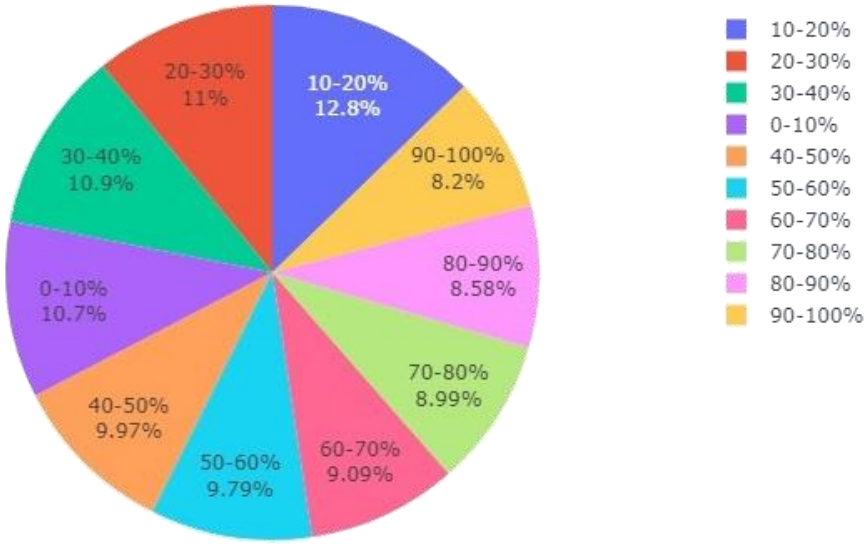

Fig. 3. Distribution of imd_band

This is clearly indicated in Figure 3, where various indexes of deprivation are almost equally distributed among the whole dataset.

Figure 4. Degree distribution

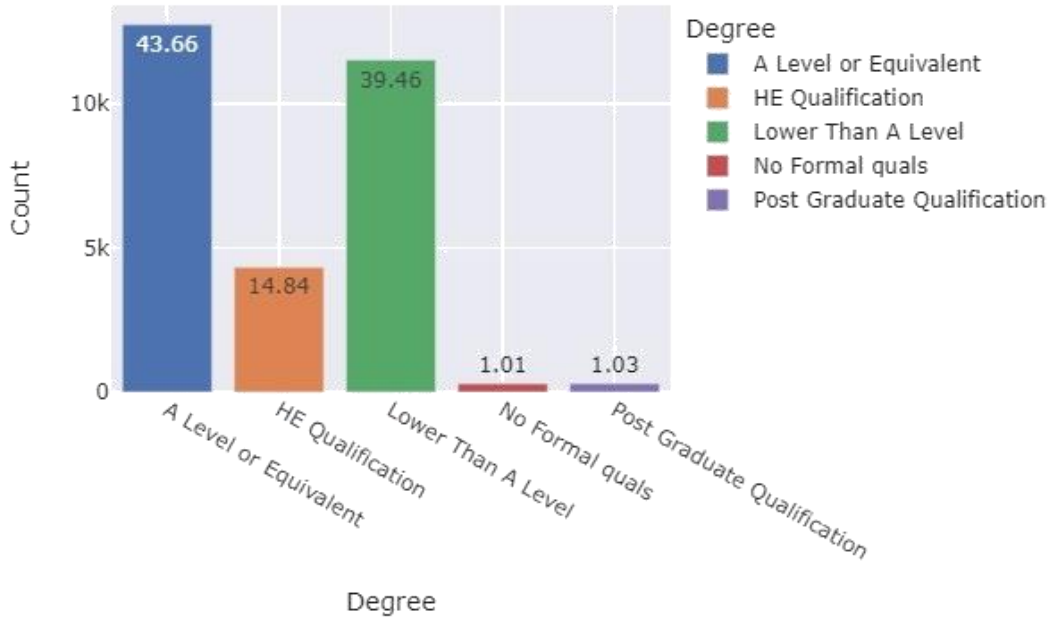

Fig. 4. Distribution of Degree

Figure 4 reveals that most of the students belong to the A level, and only 1.03 percent of them have Post Graduate Qualification.

Figure 5. Gender distribution

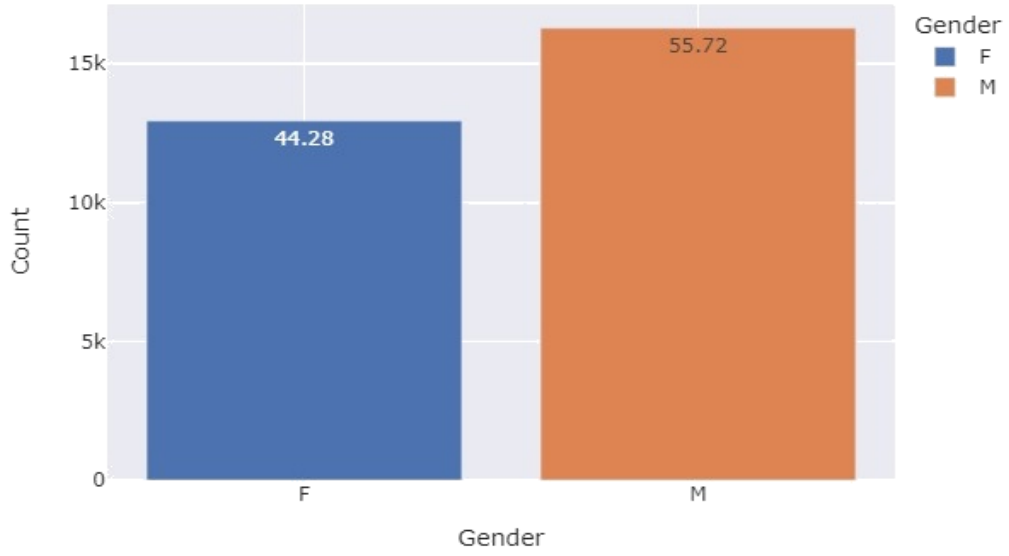

Fig. 5. Distribution of Gender

According to Figure 5, 55.72 percent of the students are male, and the rest are female

3.3 Features' correlation with label

Below are some distributions of features with label.

Figure 6. code_module

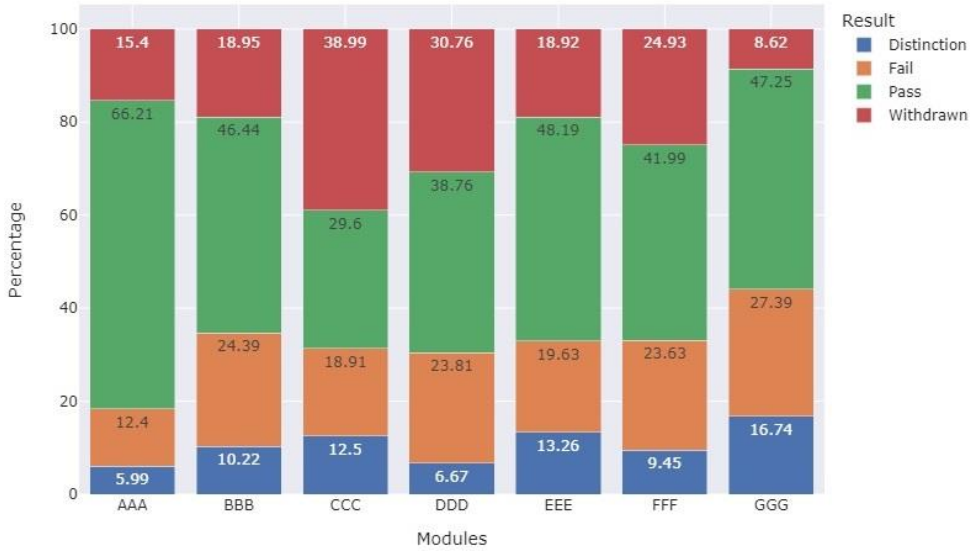

Fig. 6. Correlation code_module with label

As Figure 6 indicates, course AAA has the highest pass rate, while CCC has the lowest.

Figure 7. age_band

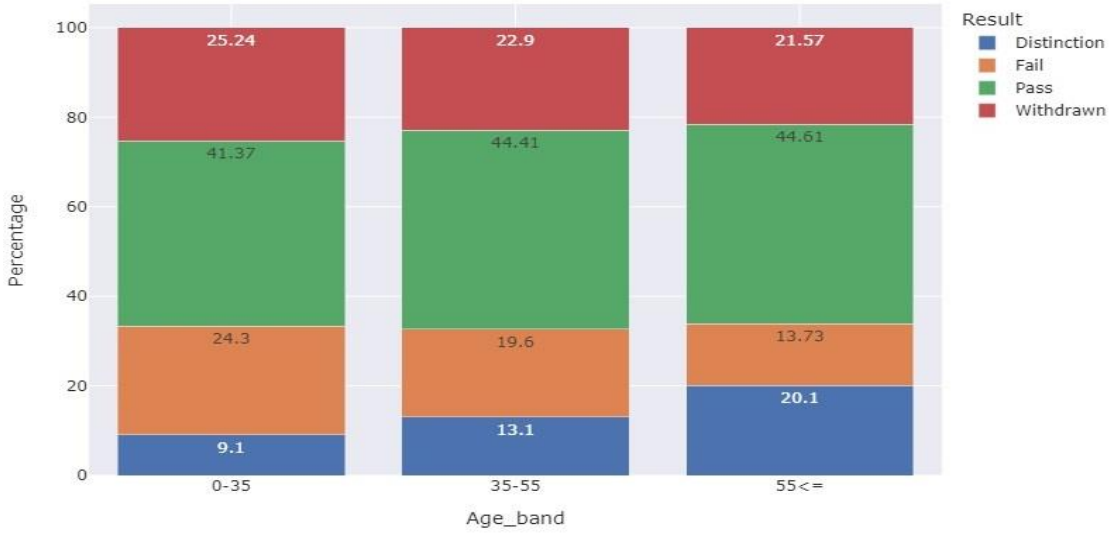

Fig. 7. Correlation age_band with label

Figure 7 illustrates that among age bands, the elderly have the highest distinction percent, while the young have the lowest. In addition, the percentage of young people who failed is almost twice that of the elderly.

3.4 Features' scatter with label

The next step is to check the scatter of features with each other and also monitor those features with label.

Fig. 8. Scatter total_activities with total_clicks

As it could be seen in Figure 8, the more activities a student does, the more likely they are to pass or be distinct in their courses.

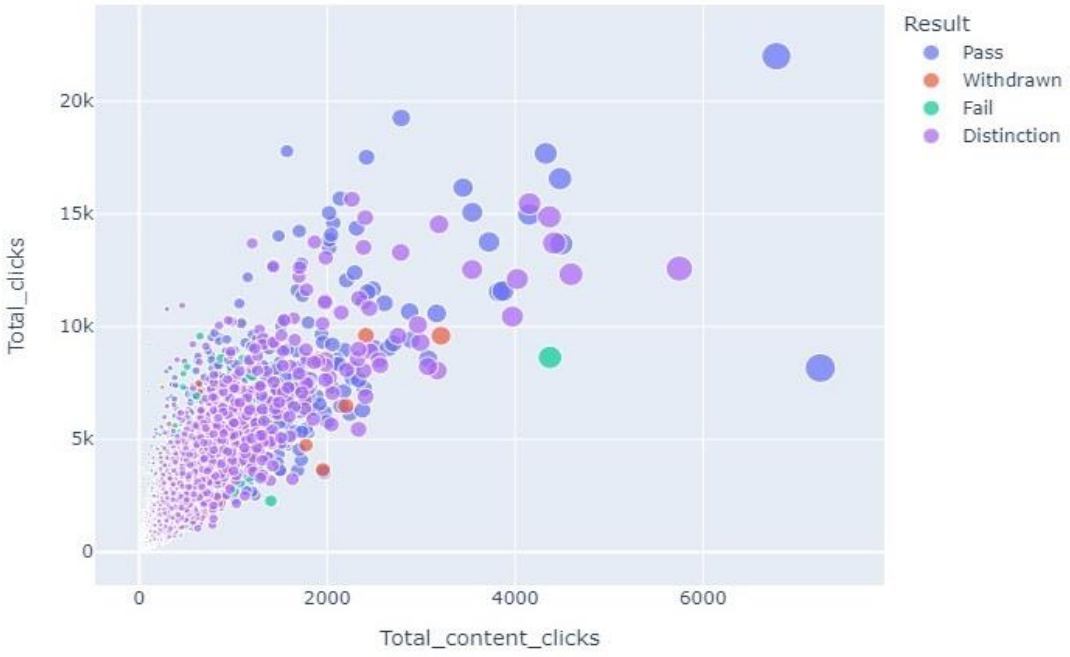

Fig. 9. Scatter total_content_clicks with total_clicks

Figure 9 demonstrates that total_content_clicks and total_cliks have a positive correlation and also contribute to distinguishing between different labels.

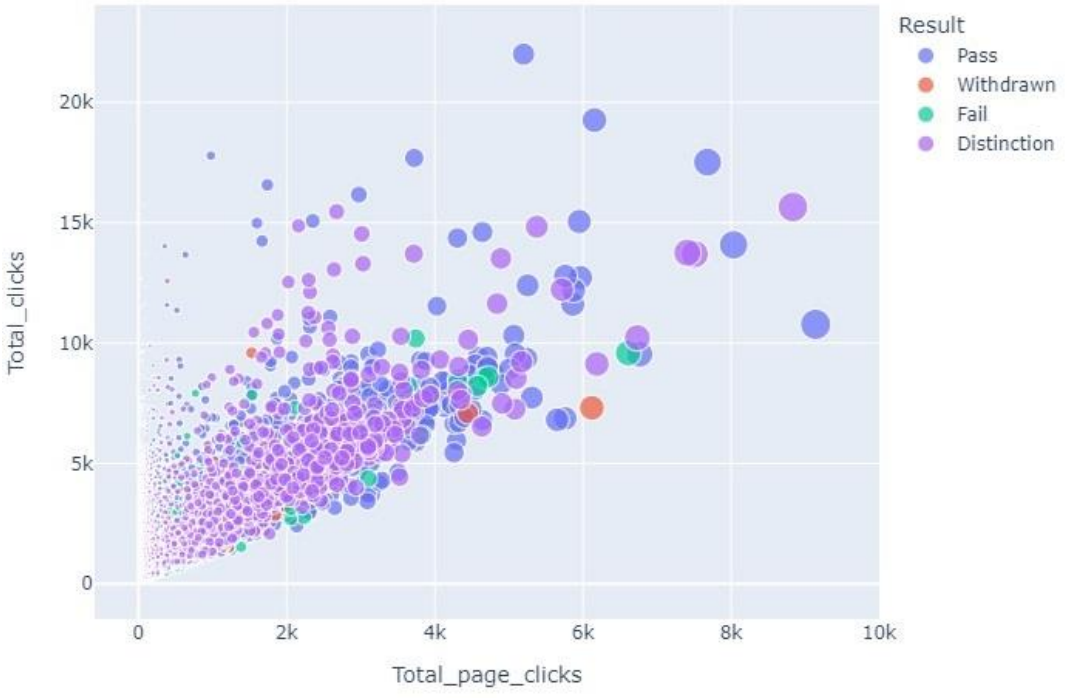

Fig. 10. Scatter total_page_clicks with total_clicks

Figure 10 shows that the feature total_page_clicks, like toal_activities, has an explicit relationship with label.

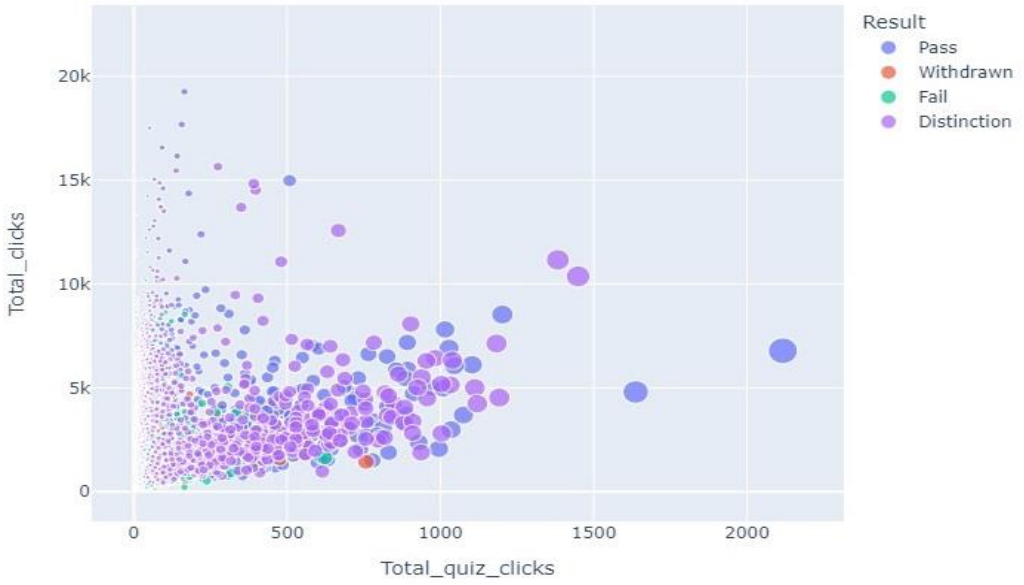

Fig. 11. Scatter total_quiz_clicks with total_clicks

According to Figure 11, for each student whose total_quiz_clicks is above 630, they have completed the courses successfully.

3.5 Categorical features with total_activities

A categorical or nominal variable is one that has two or more categories, but there is no intrinsic ordering to the categories. As we can see in our dataset code_module will be considered a categorical variable.

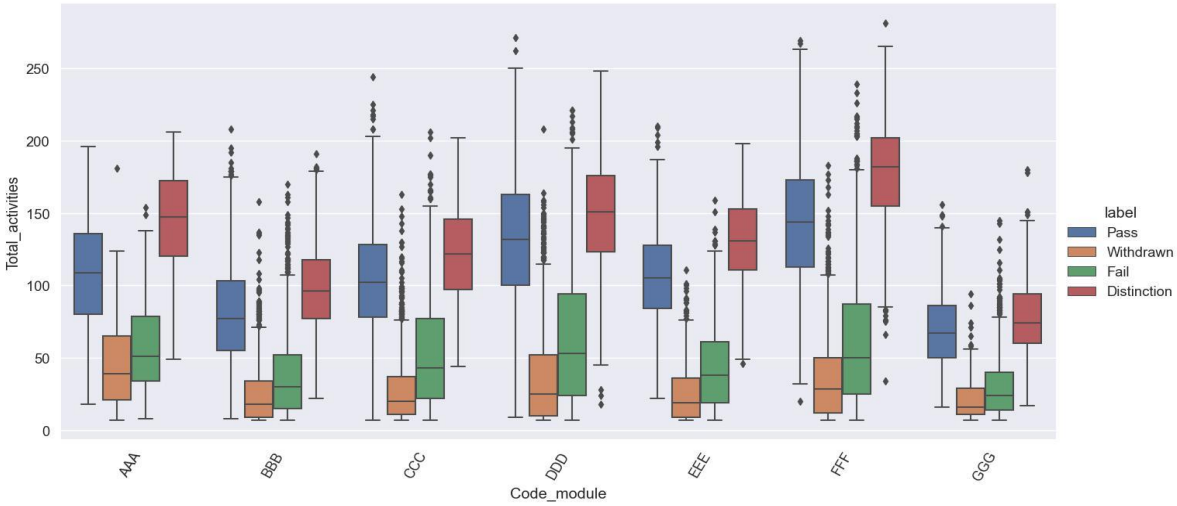

Fig. 12. Code_module with total_activities

It is obvious in Figure 12 that to fulfill a course, the students had to take part in many activities

in the virtual learning environment.

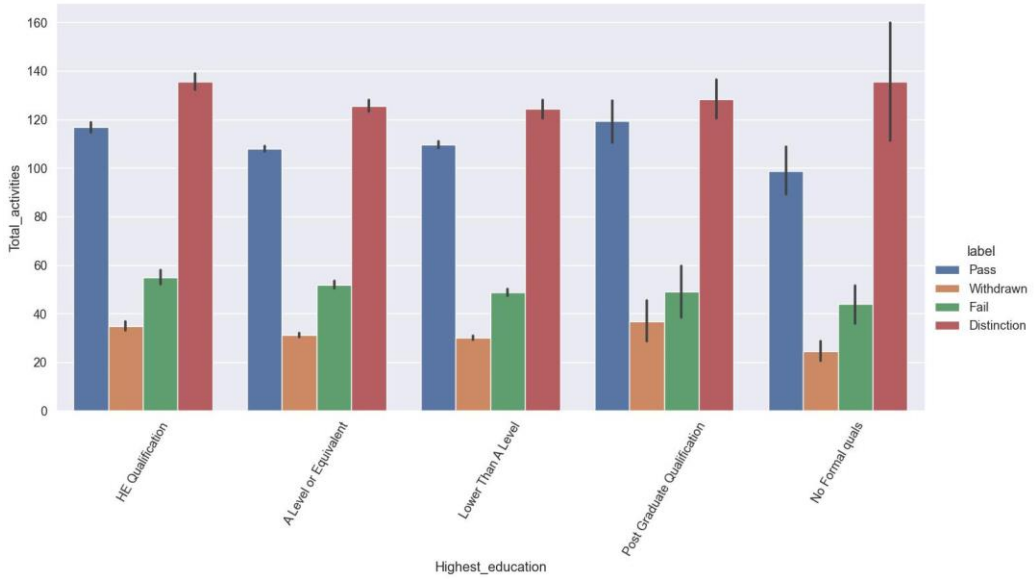

Fig. 13. Highest_education with total_activities

In Figure 13, bars related to pass and distinction are the highest ones, which show total_activities has a significant correlation with the label.

3.6 Outlier detection

Outlier detection is a crucial step in data analysis that involves identifying data points that deviate significantly from the rest of the dataset. These outliers can have a significant impact on the accuracy of classification and clustering models, making it essential to detect and handle them during data pre-processing. By removing or handling outliers, we can ensure that our models are trained on reliable and accurate data, leading to better performance and results.

As we have multiple features, we can use multivariate outlier detection methods such as Mahalanobis Distance or LOF (Local Outlier Factor). These methods take into account the correlation between the features and can detect outliers that are not detected by univariate methods.

For outlier detection in our case, we first employed the **LOF** algorithm. Any data records that contain outlier objects were removed from the dataset. However, this method failed to lead to better performance, so in turn, we just removed some data points that EDA suggests to be outliers, for instance, the green data point in Figure 14.

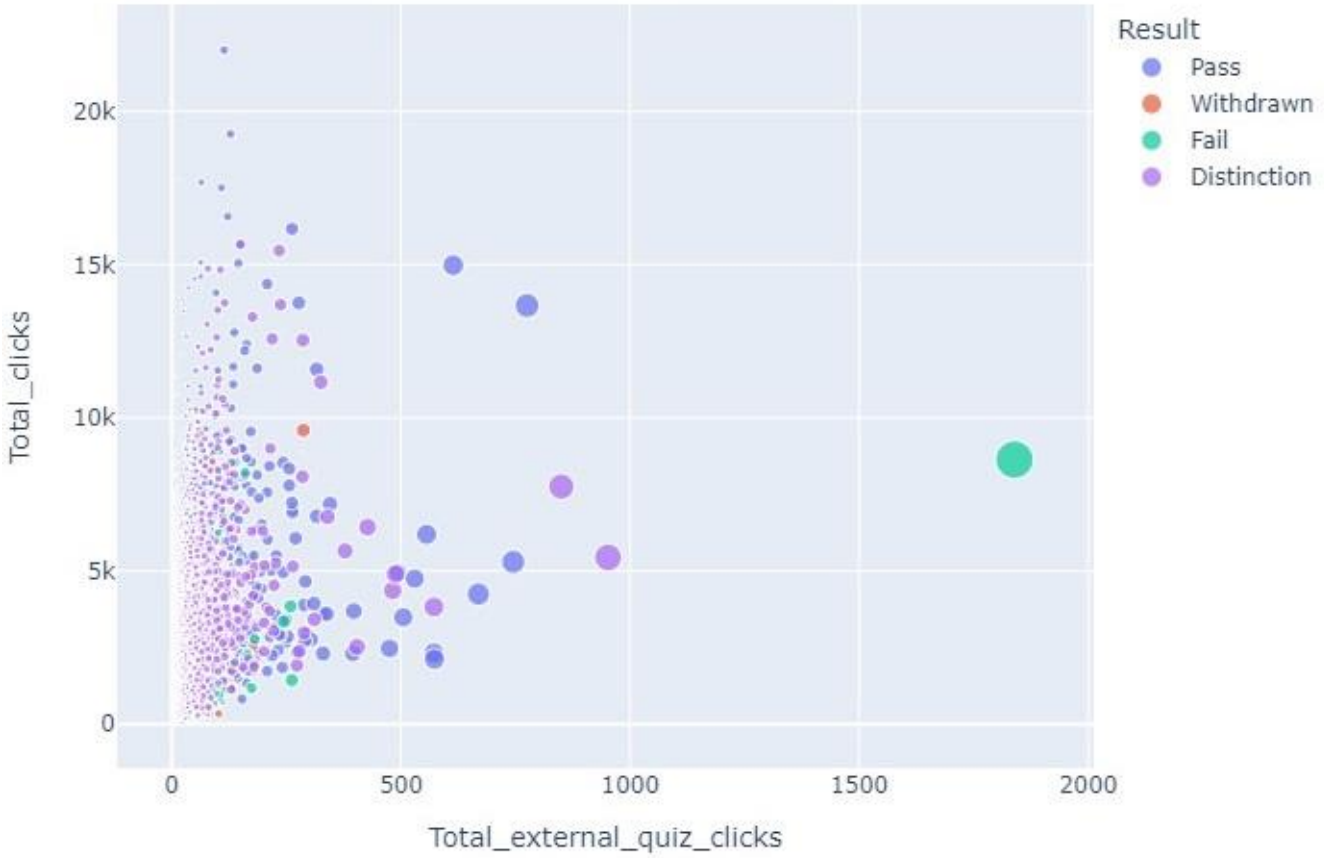

Fig. 14. Outlier detection using EDA

3.7 Feature importance

Feature importance involves calculating a score for all input features in a machine learning model to determine which ones are the most important. In our case, we have chosen three different methods to calculate and extract the best features in order to build a robust model.

Figures 15, 16, and 17 reveal three practical methods to determine the importance of each feature in predicting the output.

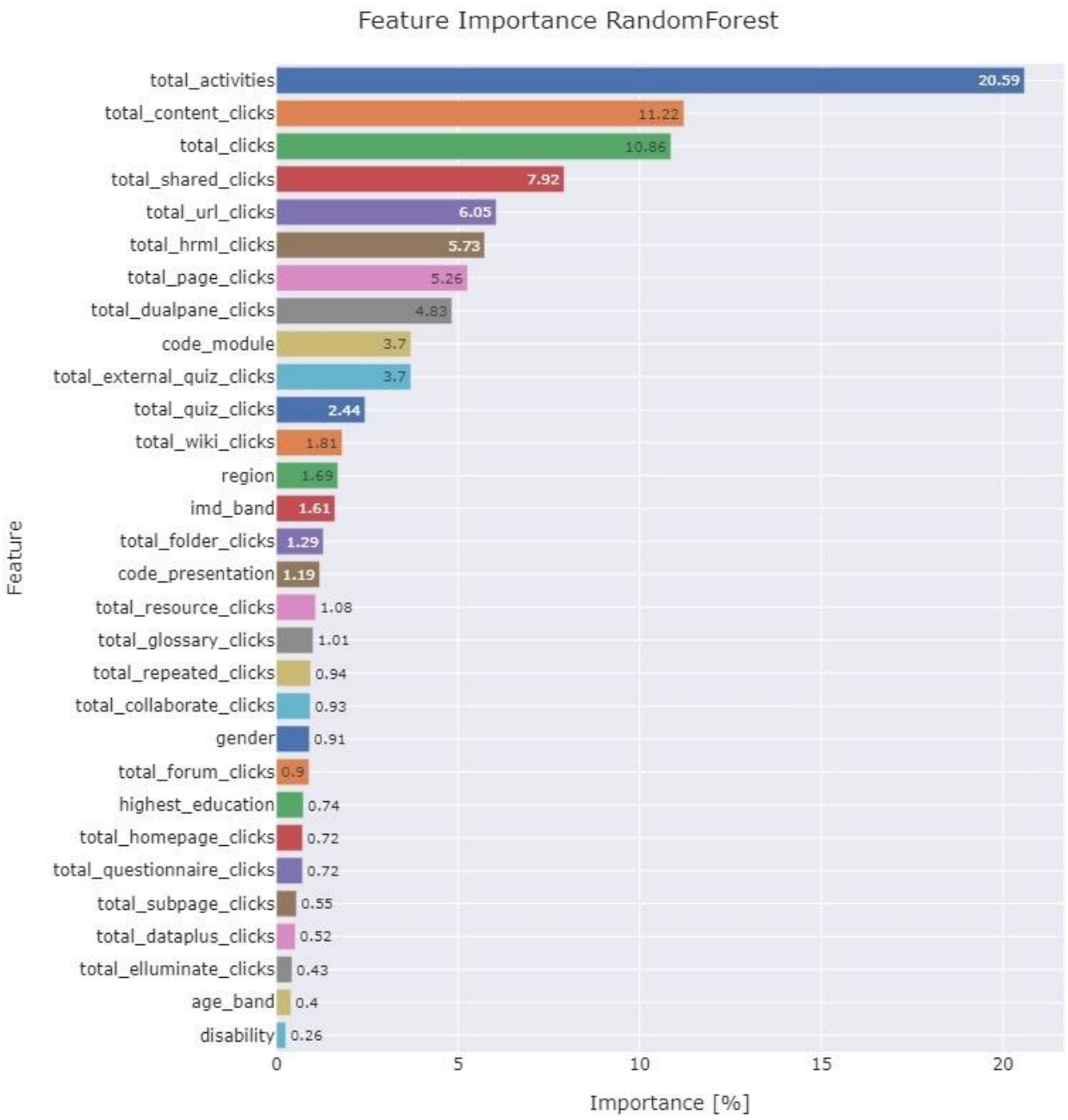

Fig. 15. Feature importance – Random Forest

This is indicated in Figure 15 for the three features total_activities, total_content_clicks, and total_clicks, Random Forest ranks them at the highest correlation to the label, so in our model we need to focus on them in order to build a high-performance model.

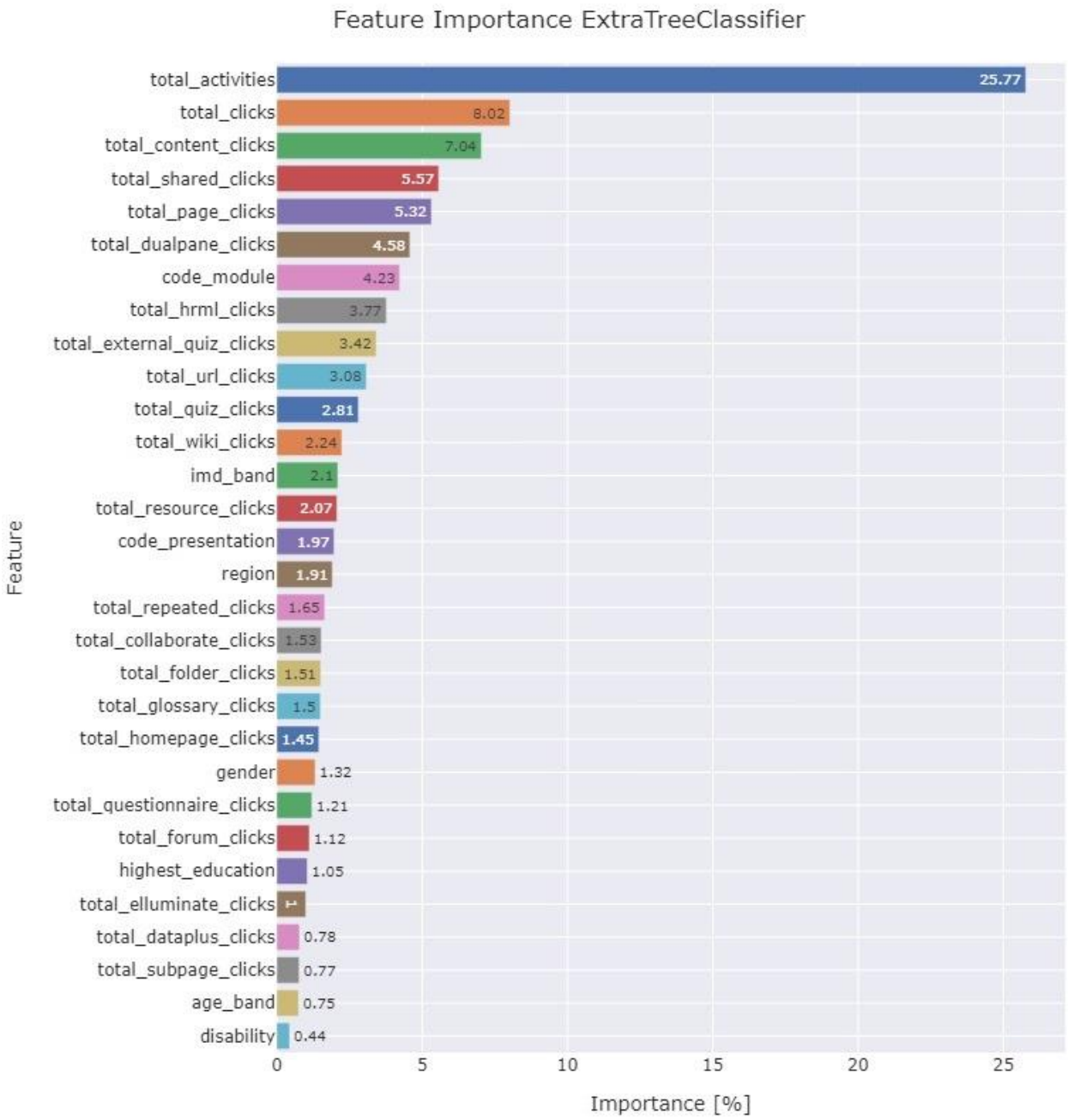

Fig. 16. Feature importance – Extra Tree Classifier

It is clear in Figure 16 that similar to Random Forest, Extra Tree Classifier selected total_clicks, total_activities, and total_content_clicks as the most important ones.

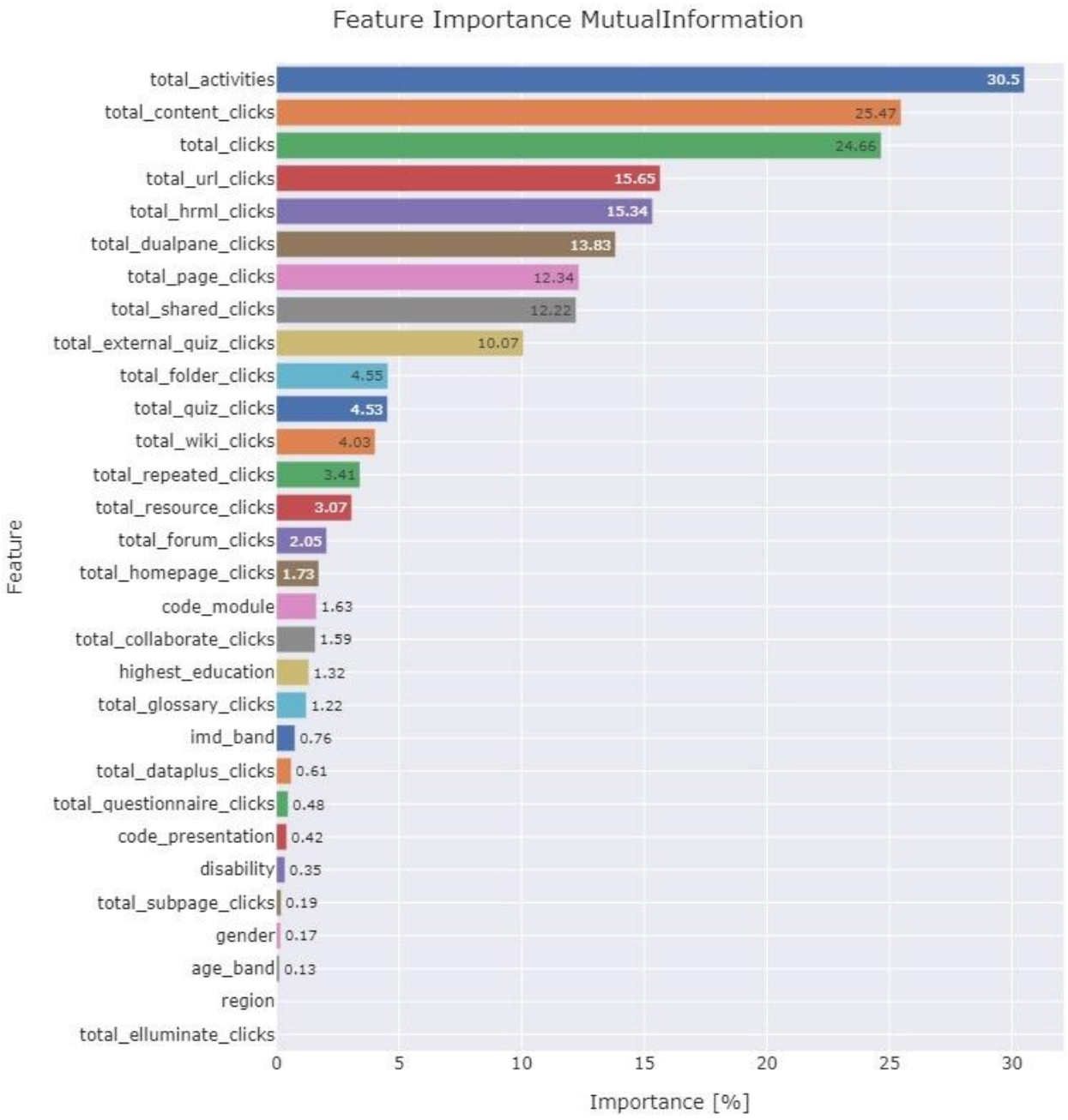

Fig. 17. Feature importance – Mutual Information

Figure 17 supports our previous hypotheses that the three features total_activities, total_clicks, and total_content_clicks will play a pivotal role in predicting the output.

4. Model Selection

After completing the pre-processing and EDA steps, the next step is to select an appropriate model

to fit the data. Since the target variable is categorical, classification models would be a good choice. There are several classification models to choose from, including logistic regression, support vector machine, decision trees, gradient boosting, and random forest.
Our task is to predict the final result for each student.

4.1 Binary classification

The final result has four different labels: Distinction, Pass, Fail, and Withdrawn, so to convert it to a binary classification, we need to do the following:
Labels 'Pass' and 'Distinction' are considered 'Success'.
Labels 'Fail' and 'Withdrawn' are considered 'Failure'.

After modifying the label, the distribution for 'Success' and 'Failure' is as follows:

class	**count**
Success	15381
Failure	13837

Eight models have been trained in order to predict the label. For each model, 30 percent of the data was used as test data to evaluate the performance of the model.

Below are the metrics for the models:

Model	Accuracy	Precision	Recall
Random Forest	0.8819	0.9328	0.8570
Gradient Boosting	0.8811	0.9337	0.8552
K-Nearest Neighbor	0.8391	0.8999	0.8156
Decision Tree	0.8245	0.8269	0.8395
Logistic Regression	0.8026	0.8626	0.7858
Neural Network	0.7508	0.6650	0.8310
Support Vector Machine	0.7461	6142	0.8682
Naive Bayes	0.7044	0.5126	0.8792

Figure 18 demonstrates the accuracy of each model:

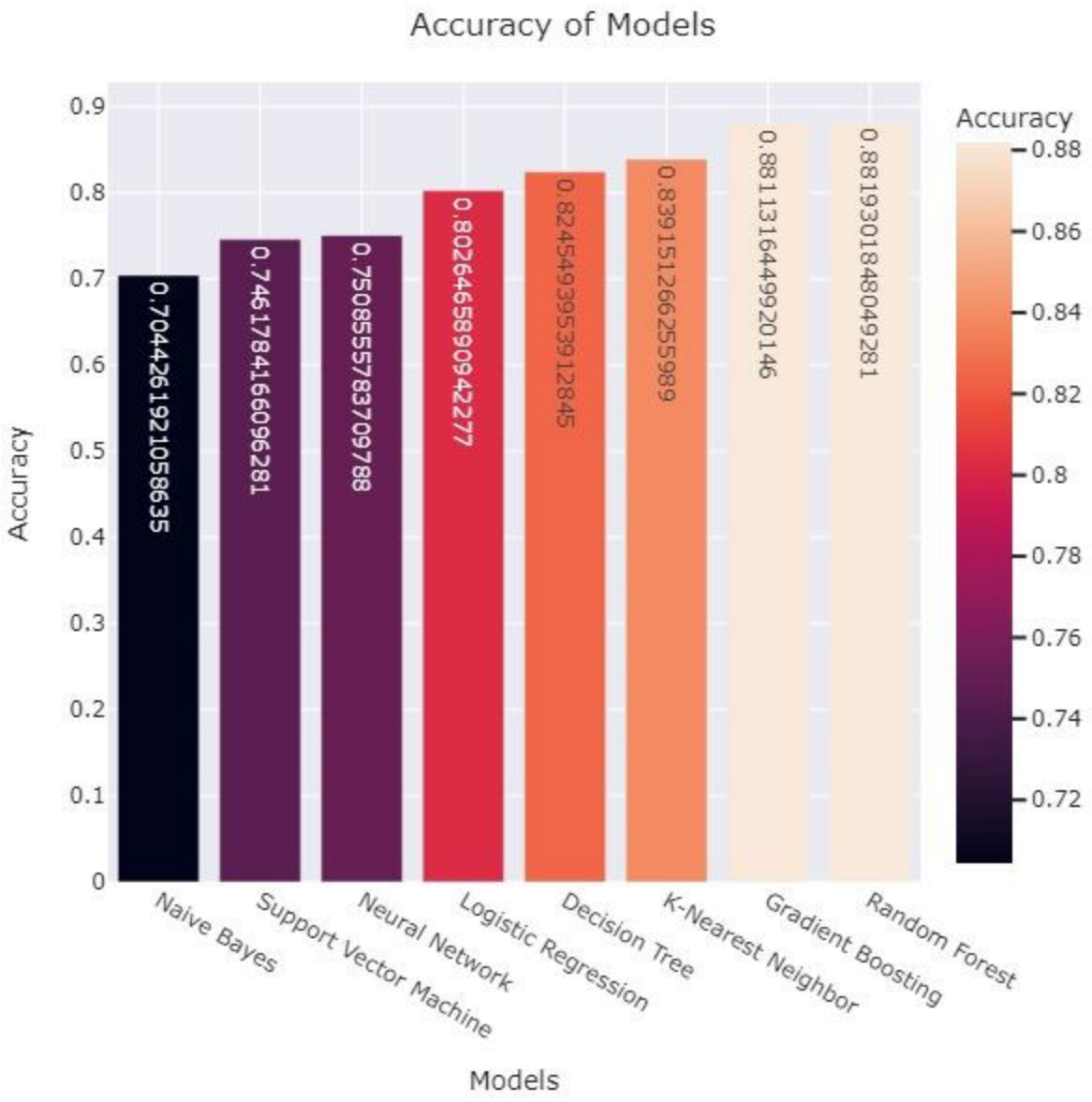

Fig. 18. Performance of models

According to Figure 18, Random Forest has the best accuracy for the data, which is expected since those algorithms utilize partitioning at different levels in order to make a prediction.

Figure 19 depicts the confusion matrix for Random Forest:

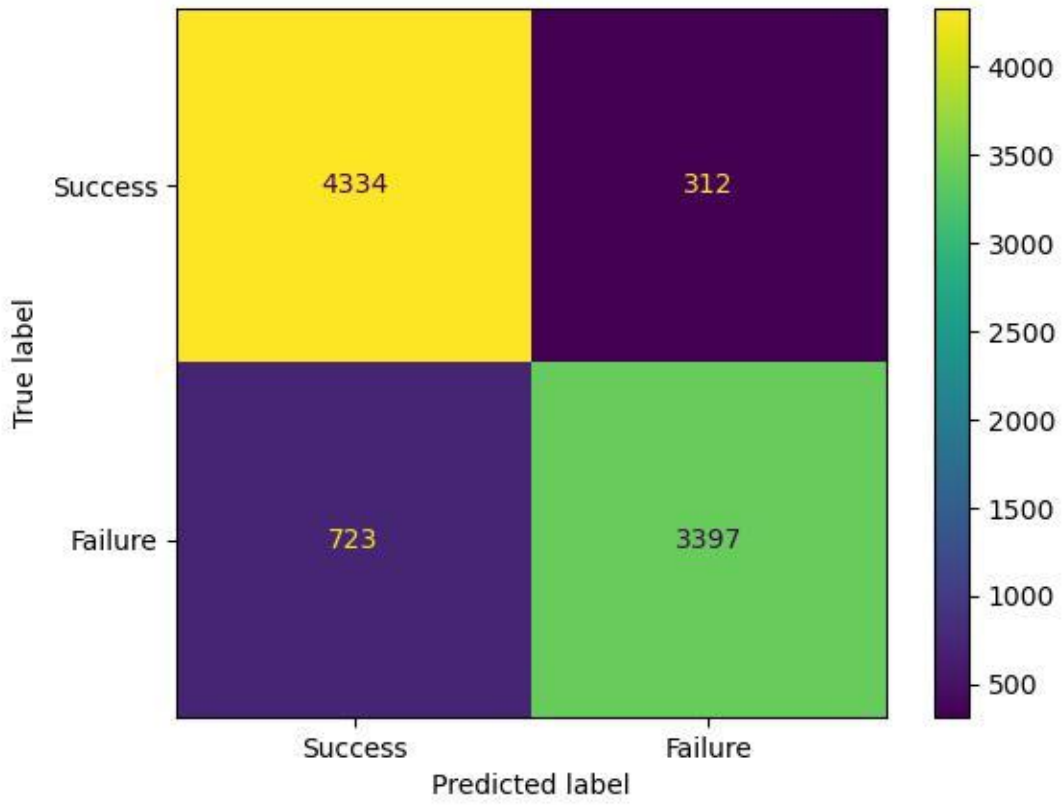

Fig. 19. Random Forest Confusion Matrix

4.2 Multi-class classification

The final result is inherently multiclass, with four different classes: 'Fail', 'Withdrawn', 'Pass', and 'Distinction'. Below is the distribution for each class:

class	count
Fail	6673
Withdrawn	7164
Pass	12357
Distinction	3024

For this classification task, we have implemented two scenarios:

1- **Class is balanced**
2- **Class is imbalanced**

4.2.1 Class is balanced

In this scenario, we suppose that data is equally distributed among classes and fit data to Random Forest with no modification.
Here are the results:

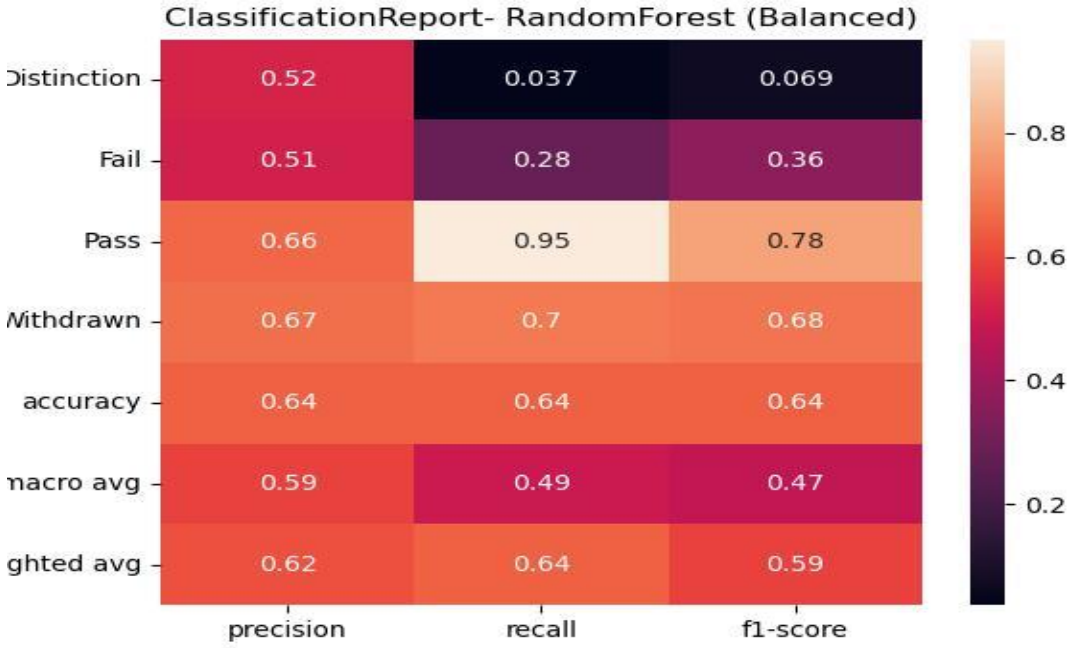

Fig. 20. Classification Report- Random Forest (Balanced)

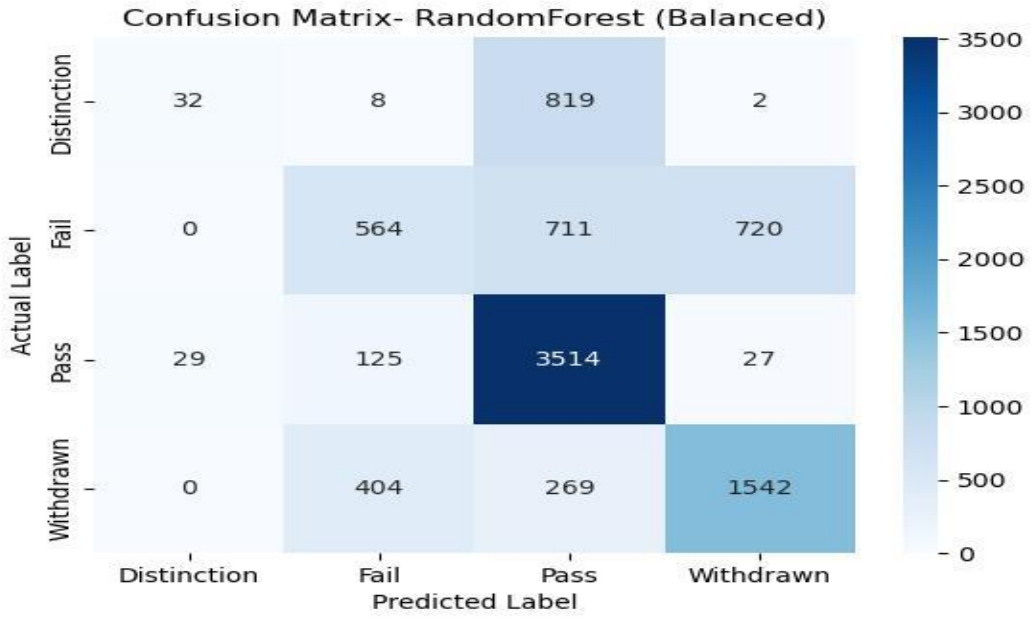

Fig. 21. Confusion Matrix- Random Forest (Balanced)

Provided the class is balanced, we observe in the confusion matrix and classification report that Random Forest has a better performance for predicting 'Pass' and 'Withdrawn' compared to 'Fail' and 'Distinction'. This is probably because the data records for those classes (Pass and Withdrawn) are larger than the two others. In addition, although Random Forest has reasonable accuracy, it fails to compute an acceptable f1-score and recall value for Pass and Withdrawn. With those limitations, we come up with another hypothesis that the class is imbalanced.

4.2.2 Class is imbalanced

For this task, we evaluate some sampling techniques to generate data points to solve the issue of imbalance. There are some algorithms, such as SMOTE, NEAR_MISS, RANDOM_OVERSAMPLING, and RANDOM_UNDERSAMPLING.

SMOTE, which stands for synthetic minority over-sampling technique, takes the minority class and generates samples based on distances among data points related to minority class, in our data 'Distinction'.

For our case, we utilized the four above algorithms and found out that Smote has the best performance among others.

Below are the results:

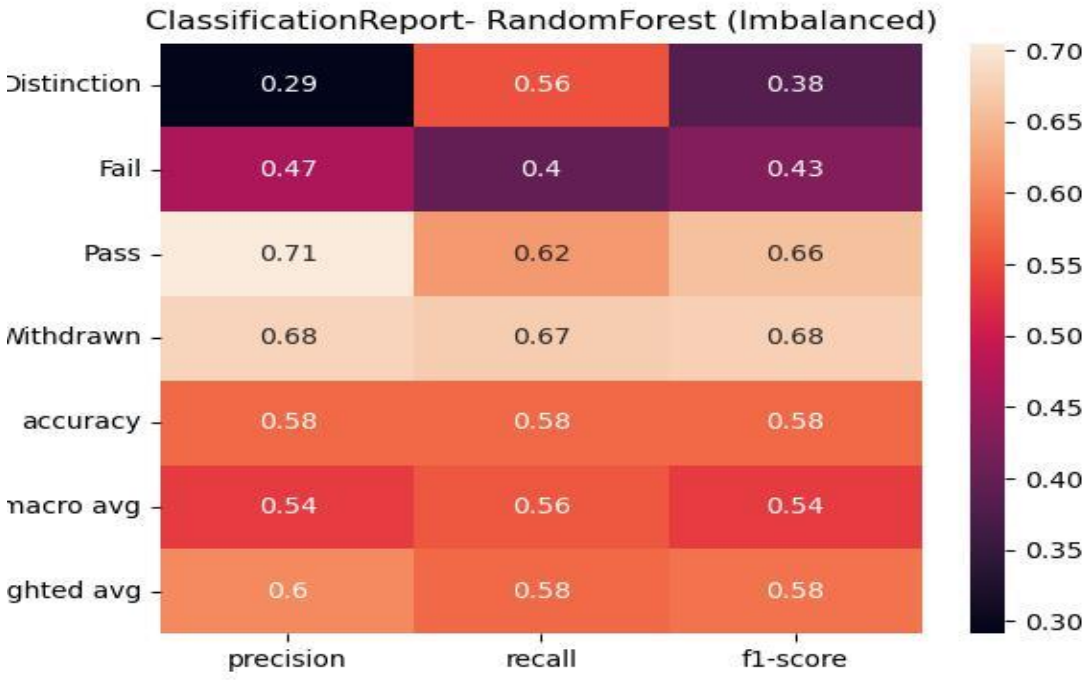

Fig. 22. Classification Report- Random Forest (Imbalanced)

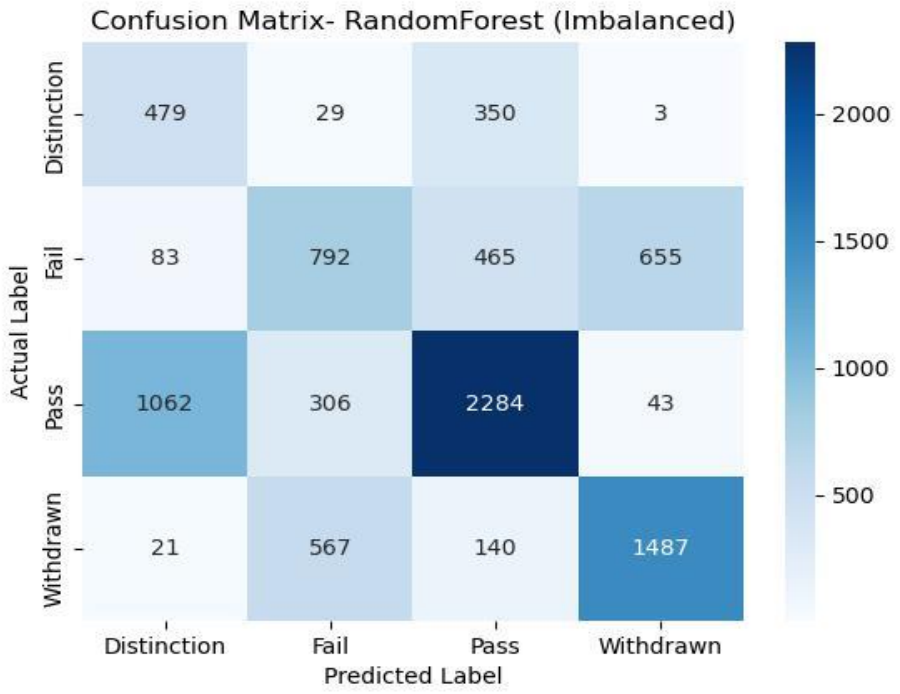

Fig. 23. Confusion Matrix- Random Forest (Imbalanced)

Figures 22 and 23 depict that supposing an imbalance of classes and using the smote algorithm remarkably enhance recall and f1-score for minority classes, Distinction and Fail.

5. Conclusion

So, in conclusion, it seems the **Random Forest** model is a better predictor for our case, but we should take into account that changing and tuning the hyperparameters of the models can somehow change the result.