Using Moodle Test Scores to Predict Success in an Online Course

Dorotea Bertović Marina Mravak Kristina Nikolov Nikolina Vidović

Data Science and Engineering Data Science and Engineering Data Science and Engineering Data Science and Engineering
Faculty of Science *Faculty of Science* *Faculty of Science* *Faculty of Science*
Split, Croatia Split, Croatia Split, Croatia Split, Croatia
dbertovic@pmfst.hr mmravak@pmfst.hr knikolov1@pmfst.hr nvidovic@pmfst.hr

***Abstract*—Many higher education institutions use a free, open-source learning management system (LMS) called Moodle that provides all the necessary tools for teachers to create virtual classrooms using the Internet. In this environment, teachers request reports detailing which course resources and activities learners accessed, including access times. Therefore, teachers can check individual student performance, whether students viewed a particular resource or participated in some activities in a certain period, based on reports known as log files. This paper deals with the analysis of data based on the scores of student online tests obtained using Moodle log files. Based of the collected data, which contain student data of the first year of undergraduate study Introduction to Programming at the University of Split, Faculty of Science and Mathematics in Croatia, pre-processing and export analysis of the data is carried out. Furthermore, prediction methods that include machine learning algorithms are used to predict student's final grade. It was shown that the highest accuracy of 82.35%, AUC and Cohen kappa with values of 0.766 and 0.706 were achieved with the Linear SVC algorithm for predicting student's performance. However, the challenge we faced is the lack of data, where three ways to achieve the most accurate performance model are presented. Also, we examined importance of considering significant attributes that influence student performance prediction results.**

***Index Terms*—e-learning systems, prediction, student performance, data mining**

I. Introduction

With the development of technology, e-learning systems have also evolved. E-learning systems are implemented in universities to provide blended or completely online learning. In the beginning, these systems were called learning management systems (LMS) and were extremely simple, or precisely, just websites with some basic features. Years later, there has been a significant development in the learning system. With the COVID pandemic outbreak, the use of online learning systems has grown significantly. The global pandemic has deeply shaken education systems to the core, with the impact having directly affected 1.6 billion students in more than 200 countries, that is more than 94% of the world's student population [1].

Educational organizations are struggling to define the impact that new technologies are having on learning and stu- dent success, and are striving to increase graduation rates. Therefore, it is necessary to find approaches that improve assessment, offer adequate feedback and guaranteed student success. Researchers introduce several approaches and tools with different LMSs in the form of accessible online dash- boards which allow teachers to monitor the progress of their learners and at the same time allow learners to visualize their learning process. Those tools can help to improve student's self-regulation and academic success [2], [3]. Furthermore, their scope is to develop intelligent systems for early warning capable to provide suitable feedback for maximizing student achievement [4]. Equally important is to consider students' preferences in the learning process by understanding each student's learning style [5].

Students can access the LMS using their accounts. There is a database that stores all the system's information: personal information about the users (profile), academic results and user's interaction data. Accordingly, LMS contains log files which represent a vast quantity of data such as number of logins, number of accesses to elements of an online course, number of assignments completed, number of days in the online course, activities grades, term grade, course grade and so on, which allows a descriptive overview of students' behavior that could influence their academic performance. There are research papers that serve a collected data from multiple learning environments such as student information system (SIS) and LMS, including mobile applications that record video interactions of students which can be considered to provide better insights into video learning analytics and student performance [6].

Systems like Moodle (www.moodle.org) contain large amounts of data on student behaviour and engagement. Due to this data, two areas of research emerged: Educational Data Mining (EDM) and Learning Analytics (LA) [7]. EDM develops some new tools using education data, while LA is measures, collects and analyzes the data.

The main area of this study is the prediction of student success, which helps for early warning, knowing whether the student's actions and already earlier learning results lead to dropping out, failing or succeeding in the course. In other words, it is useful to know which variables are related to student performance, and then instructors could trigger actions to improve the learning process of students. Therefore, one of the biggest challenges is to improve the quality of the edu- cational process, which would also improve the performance of students. Each instructor would thus receive an assignment related to teaching methodology where they would apply more up-to-date methods and concepts to meet the requirements and provide assistance to unsuccessful students. By doing so, students have insight into their knowledge and early results that they should understand in order to take further steps to improve at an earlier stage of learning.

There are several classification techniques used to predict student performance based on whether a student passed or failed an online course or what grade the student received. For this purpose, well-known classification algorithms such as decision

trees, random forests, k-nearest neighbors (kNN) and others are used to investigate which techniques are better for predicting student performance. There were experiments with 24 university courses which show that it is only feasible to directly transfer predictive models or apply them to different courses with an acceptable accuracy and without losing porta- bility under some circumstances [8].

It is important to note that features or variables that are potential predictors play an important role, and in this study some of them are scores of colloquia and tests that students take during online course. If a reasonable prediction can be achieved only with test scores, a simpler implementation of the systems for predicting student performance is possible.

The goal of the study presented in this paper is to inves- tigate the data obtained using the Moodle system of students who attended the undergraduate Introduction to programming course based on classification methods in machine learning.

To understand the existing study of predicting students performance, it is important to define right research questions. This paper focuses on the use of different algorithms for predicting student grade based on particular features to answer two research questions:

1) Which important features significantly influenced student grade prediction performance?
2) What is the effect of different prediction methods on measures of success in predicting students' grades?

A comparison of individual approaches was discussed and conclusions based on research questions were drawn. During the research, we encountered deficiencies in some of the data, which of course affected the results of the model for predicting student performance.

In the next section, the related work is provided. After that, focus is on the methodology, including the research questions and the description of used methods. Afterward, the results and the discussion are presented. Lastly, the overall conclusion is outlined at the end of the paper.

II. Related work

Due to a large amount of data, student performance prediction becomes increasingly challenging. Researches conducted various analyses and used different data mining techniques to improve student achievement and predict student performance. This section describes the related works found on the Web of Science platform (https://www.webofknowledge.com/), which provides access to an unrivaled breadth of world-class research literature linked to a rigorously selected core of journals. Several filtering steps were performed considering primary studies published in English in journals and conferences using prediction of student performance based on Moodle log files, including the period from 2018 to 2022.

In [9], researches investigate procrastination in predicting achievement of students in online courses. They predict the course achievement in Moodle through their procrastination behaviour using student's homework submission data by considering their active, inactive, and spare time for homework, along with homework grades. The classification methods used include linear and radial basis function kernel support vector machine (L-SVM and R-SVM), Gaussian processes (GP), Decision Tree (DT), random forest (RF), neural network (NN), AdaBoost (ADB), and Naive Bayes (NB). Their results show precision and accuracy of L-SVM algorithm of 87% and 84%. Considering their results it is recommended to use a less complex approach that is easy to implement, interpret, and use by practitioners to predict students' course achievement with a high accuracy and possibly take remedial actions early in the semester.

The aim of [10] is to explore the utilization and impact of digital technologies on an e-learning platform. Student success with the e-learning system was evaluated using a mixed-method: Social Network Analysis, k-Means Clustering, and Multiple Linear Regression. There is quite high precision of 0.73, high accuracy of 0.68 and a very good F1-score of 0.7. The data set of a postgraduate course of five modules in the Hellenic Open University is exploited and divided into six different periods during the academic year, as described in [11]. Using data mining methodologies, machine learning techniques and statistical tests, eight different target categories were predicted, such as the grades of the six different written assignments, the average grade and the final examination marks for the 23 different sections of the course's modules. The polarity and emotions results for each student and tutor as well as their interactions in the module were the inputs for the prediction models. The models' comparison ranked first the Additive Regression (AR) model for the objectives of written assignments' and final examinations grades as well as the SMOreg (SVR) for the average grade. In addition, there was an obvious tendency to improve model predictions, with the presence of sentiments' variables.

A collection of models (exploratory factor analysis, multiple linear regressions, cluster analysis, and correlation) to early predict the academic performance of students was provided in [12]. Results indicated that the major contribution to the prediction of the academic student performance is made by four factors: access, questionnaire, task and age. Also, it is remarkable that Age was identified as a negative predictor of the performance of students.

Employing machine learning techniques on the Moodle logs, authors in [13] analyzed a prediction model for students at risk of dropping out. 28% of the students who started well in the course failed to complete it. Students who completed the course with poorer grades did not access the teaching materials. Only 2% of the students who started with difficulties in the assessments managed to complete the course. The model predicted individual student performance and helped identify the risk of failure and provide timely assistance.

III. Methodology

In this section, the e-learning data mining process is described. As shown in Figure 1, the methods used to address research questions are described by the framework of the e-learning data mining process proposed by Wang [14] which consists of five research tasks, including data collection, data pre-processing, model construction, model definition and model application.

A. Data collection

We collected data during Introduction to programming course held in the fall of 2021 at the University of Split, Faculty Of Science, Croatia involving a total n = 165 students. The course teaches the students the basics of programming in Python, and it was organized as a blended learning course. The

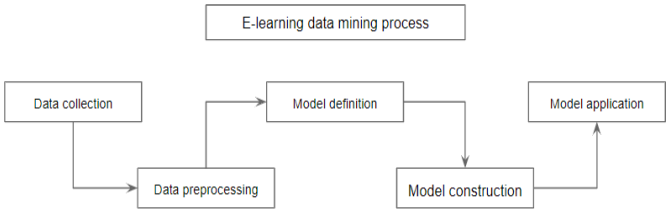

Fig. 1. Framework for e-learning data mining

online course material consists of 36 lessons accompanied by online exercises, quizzes and another set of online tests (related to course modules about variables, loops, functions, arrays, and data files), which the students took approximately every three weeks.

The course had 2 hours of lectures and 2 hours of exercises per week, and the students were free to complete their weekly assignments, online quizzes and online tests. The grading was based on two in-class exams (colloquia) (50% each of the overall score) or a final practical exam where the student had to reach 50% of the overall score and, finally, an oral exam to pass the course.

For the study, the source of collected data is the Moodle database used to extract the log files, which contain textual data that is saved in a .csv format file. Due to the needs of the study, the collected data contain the percent scores of online tests and colloquiums, so there was no need for additional processing with Python libraries, but several data processing actions in Excel were used. The collected data was prepared and accordingly anonymized by the teacher of the course doc.dr.sc. Ani Grubisˇic´. A total of 165 students are registered in the Moodle system for the Introduction to programming course. Accordingly, an Excel file containing a total of 165 records was composed.

As shown in Table 1, the attributes of the collected data are described. The data consists of student id, categorical group of studies, first-time enrollment represented by values 0 or 1, practical tests scores, practical/oral pass (values 0 or 1) and final grade scores represented by values 0, 2, 3, 4 and 5 where 0 represent failure of the student, while 5 is the highest score.

TABLE 1
VARIABLES OF DATA COLLECTION

Name	Description
#No	Number of the student [#No]
StudentID	ID number of the student
Group	Study of the student [CS, CST, MAT, MCS]
Re-enrolled student	Second enrollment in the course [Yes=1, No=0]
Initial test	Percent score of initial test [%]
Variables test	Percent score of test for lecture Variables [%]
Loops test	Percent score of test for lecture Loops [%]
Functions test	Percent score of test for lecture Functions [%]
Colloquium1	Percent score of colloquium1 [%]
Arrays test	Percent score of test for lecture Arrays
Data files test	Percent score of test for lecture Data files [%]
Colloquium2	Percent score of colloquium2 [%]
Colloquia total	Sum of colloquia scores [%]
Practical exam pass	Is practical exam pass [Yes=1, No=0]
Oral exam pass	Is oral exam pass [Yes=1, No=0]
Final grade	Category of final grade [0, 2, 3, 4, 5]

The following is a more detailed description of some of the variables listed in Table 1:

- **#No**: a numerical variable representing the student's serial number. It is a number from 1 to 165.
- **StudentID**: a textual variable representing the student's unique identification number.
- **Group**: a categorical variable that represents the group of studies that students attend. The data set contains four categories of study groups attending the course: CS (Computer Science), CST (Computer Science and Technology), MAT (Mathematics), MCS (Mathematics and Computer science). Student can belong to only one of the mentioned groups.
- **Re-enrolled student**: a binary variable that represents whether the student has enrolled in the course more than once or is attending the course for the first time. A value of 0 represents a student enrolling in the course for the first time, and a value of 1 corresponds to students who attend the course more than once.
- **Initial test**: a numerical variable that represents the percentage of the initial test, with which students demonstrated their prior knowledge of the basics of the course.
- **Variables test, Loops test, Functions test, Arrays test, Data files test**: numerical variables that represent the percent score of online tests that correspond to specific modules of the course, respectively variables, loops, functions, arrays and data files. Students take the above-mentioned tests online after a specific course module has been covered, each is held on average every third week.
- **Colloquium1, Colloquium2**: numeric variables that represent the percent score of colloquiums. Each colloquium was held after eight weeks of online lectures. A student who achieves 50% in each colloquium and successfully passes the oral exam also passes the course. Based on the points of the colloquium and the passed conditions of the oral exam, the final grade is defined.
- **Colloquia total**: numeric variable that represents the total percent score of first colloquium and second colloquium.
- **Practical exam pass, Oral exam pass**: a binary variables representing whether the student successfully or unsuccessfully passed the practical/oral exam. The value 0 represents a student who did not pass the practical/oral exam, and the value 1 corresponds to a student who successfully passed the practical/oral exam.
- **Final grade**: a categorical variable that represents the group of final grades that a student can achieve. The data set contains five categories of final grade: 0 (the student failed the course), 2, 3, 4 and 5 (the highest grade that student can achieve). A student can achieve only one of the listed grades.

B. Data exploration and pre-processing

We pre-processed the collected data in order to carry out further analysis. In any data analysis, it is important to check the relevance of variables to remove redundant or noisy features. For this reason, the variables number of students, student id, study group and total scores of colloquia were removed. The second step was filling out undefined or missing data as students' non-participation in assigned course tasks. It is important to understand that handling missing values is one of

the key aspects of data pre-processing when training prediction models. Missing values are handled using different imputation techniques which estimate the missing values from the training data. Thus, the key point is to decide which im- putation technique use to obtain the most efficient measure of central tendency of the data to replace missing values. We used three imputation techniques: zero imputation, mean imputation and median imputation of each variable. Accordingly, three differently filled data sets were analyzed for comparison when creating the prediction models.

Since data has varying scales and some algorithms do not make assumptions about data distribution, we normalized data before predictions. From *sklearn.preprocessing*[1] package, *MinMaxScaler* estimator was used as a scaling technique method which transforms features by scaling it as default between 0 and 1.

Furthermore, a statistical indicator called the correlation matrix is used, a table showing Pearson's coefficients, measures of the strength of the linear relationship between variables. In the experiment, the correlation matrix was used for the statistical description of the data, accordingly also for the selection of features, the process of selecting the input variables that contribute the most to the prediction models.

C. Model definition and construction

For final grade score prediction, we used almost all variables excluding student number, student ID, study program and data previously filled by null, mean and median values of variables. For the predictive data analysis, we used the simple and efficient library sklearn[2] for machine learning and statistical modelling, including classification. Variables like colloquia usually represent the final grade in education. So, we built model for final grade prediction based on only these two variables. Finally, considering the online tests for particular course module, another model for practical exam pass/fail score prediction was created.

For the classification, the following algorithms were used: C-Support Vector Classifier (SVC), Linear SVC, MLP Classifier, Random Forest Classifier, Logistic Regression, KNeighbors Classifier, Gaussian NB and Decision Tree Classifier from the *sklearn* library. For each model, the training set contained 80% of collected data and 20% for testing.

Lastly, the predicted final grades generated by the models and actual grades were compared.

D. Model application

We typically evaluate the quality of the classification model using classification evaluation metrics and the most frequently used one is accuracy. Cohen's kappa score and accuracy were used to assess the overall performance of the classifiers, and area under the ROC curve (AUC) was used to assess the ability of the classifier to correctly identify the specific grade. After the models are defined and evaluated on the testing data, accuracy, AUC and Cohen's kappa were calculated based on predicted and actual data, then applied and compared for all classification algorithms in analysis.

Fig. 2. Confusion matrix

Additionally, we presented the results of confusion matrices which provide the comparison of values like True Positives, False Positives, True Negatives and False Negatives, as shown in Figure 2.

IV. Results and Discussion

To make a prediction based on the collected data set, we used Python programming language, but also libraries such as *pandas*[3], *numpy*[4], *seaborn*, *matplotlib*[5] and *scikit-learn* in the *Google Colaboratory*[6], the free environment that runs Jupyter notebook in the cloud and stores its notebooks on Google Drive.

A. Data Analysis of Student success

A summary statistic of our dataset is shown in Table 2 and contains the mean value, standard deviation, minimum value, median value and maximum value.

Here we explored data in details by visualizing plots using the *seaborn*[7] package to get the number of students per study program, the number of students who enrolled in the course for the first time and how many students with failed initial tests passed the course.

Figure 3 shows the number of students per study program who enrolled in the course. Most of them were students in Computer Science (CS), Computer Science and Technology (CST) then students in Mathematics (MAT) and few students in Mathematics and Computer Science (MCS).

TABLE 2
Summary statistics

	Mean	Std	Min	Median	Max
Re-enrolled student	0.30	0.46	0.00	0.00	1.00
Initial test	44.03	17.66	10.00	40.00	90.00
Variables test	72.82	13.52	21.05	73.68	100.00
Loops test	54.39	21.34	20.00	50.00	100.00
Functions test	58.80	20.28	8.33	58.33	100.00
Colloquium1	30.44	26.25	0.00	24.50	100.00
Arrays test	55.58	25.16	0.00	63.54	95.83
Data files test	53.51	23.03	0.00	59.09	95.45
Colloquium2	49.93	36.24	0.00	52.50	100.00
Colloquia total	50.68	56.25	0.00	21.00	200.00
Practical exam pass	0.54	0.50	0.00	1.00	1.00
Oral exam pass	0.45	0.50	0.00	0.00	1.00
Final grade	1.43	1.74	0.00	0.00	5.00

[1] https://scikit-learn.org/stable/modules/preprocessing.html
[2] https://scikit-earn.org/stable/
[3] https://pandas.pydata.org/
[4] https://numpy.org/
[5] https://matplotlib.org/
[6] https://colab.research.google.com/
[7] https://seaborn.pydata.org/

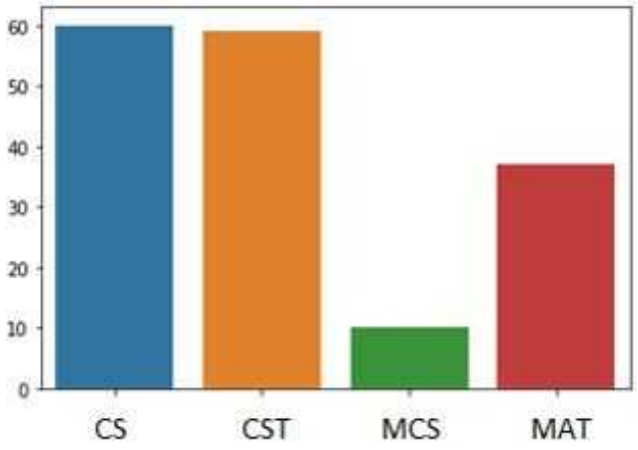

Fig. 3. Number of students per study program

Figure 4 shows the total number of students who enrolled in the course for the first (marked in blue) and second time (marked in orange). Most students (n = 116) enrolled in the course for the first and a large number of them (n = 50) enrolled for the second time. Students were grouped according to the study program and divided into those who enrolled in the course for the first time and those who re-enrolled. The graph shows that most students (n = 80) who enrolled for the first time were students in Computer Science and Computer Science and Technology. 39 students re-enrolled in the course. Out of 47 students in Mathematics and Mathematics and Computer Science, 11 enrolled in the course.

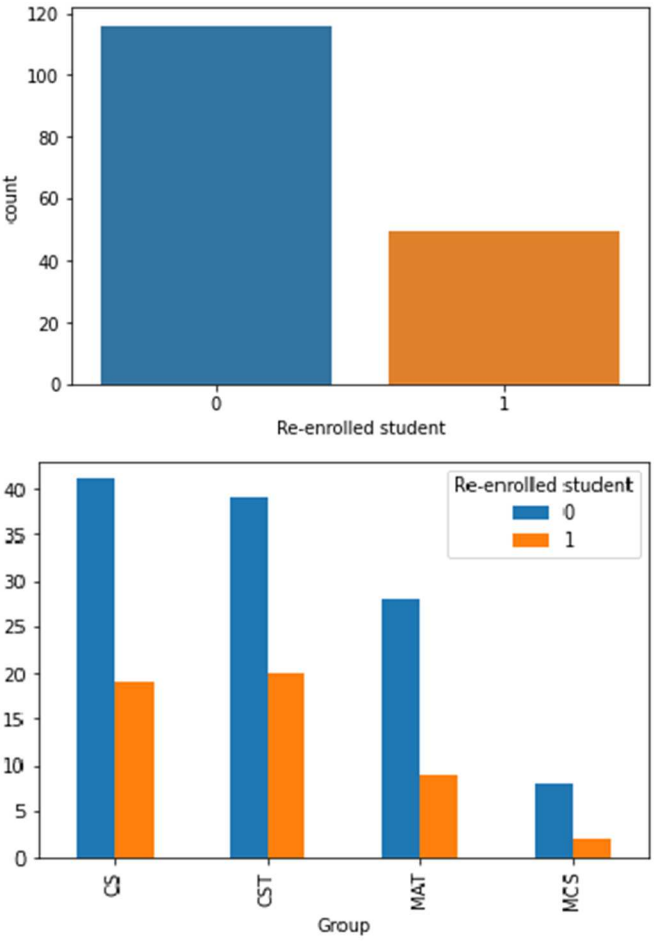

Fig. 4. Number of re-enrolled students per study program

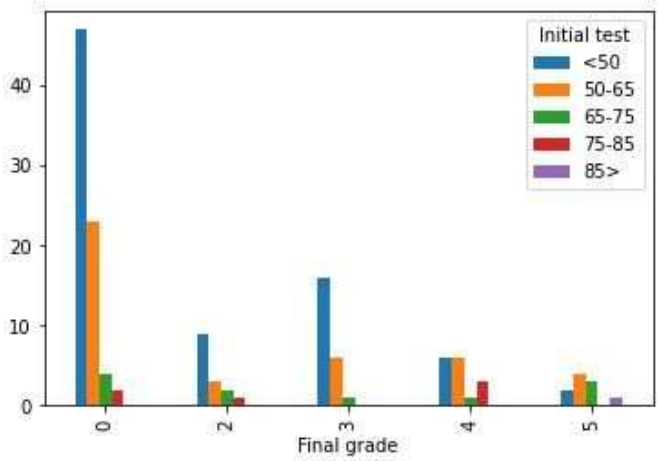

Fig. 5. Initial test scores and actual final grades per student group

Figure 5 shows the initial test scores and actual final grades per student group. Student grades correspond to the number of points achieved on the initial test. Less than 50 points equal to grade 1, 50 to 65 points (2), 65 to 75 points (3), 75 to 85 (4), and above 85 points (5).

We see that the highest percentage of students (n = 47) who scored less than 50% in the initial test did not get a positive final grade. It is also the case with 23 students who scored more than 50% and less than 65%. A small number of students were successful in the initial test. One student who scored over 85% on the initial test and got a final grade of 5.

Among students who had a final grade of 5, there were only 2 re-enrolled students out of 10, while among students who did not have a positive final grade, there were 30 re- enrolled students out of 92. For students with high success, the lowest value of the Colloquia total was 20. The highest score was 200, which is the maximum value of achieved score only achieved by one student. For students who did not achieve positive success, the lowest score in Colloquia total was 0. There were 28 students, of whom 7 were re-enrolled students. The highest score for these students was 100, which is also the passing of the practical exam. There were two such students, but they did not pass the oral exam. Among 50 re-enrolled students, 20 received a positive final grade, and 54 of the 116 students enrolling in the course for the first time passed. It can be concluded that the pass percentage is slightly higher among students enrolling in the course for the first time, as opposed to re-enrolled students.

Further, using the data visualization plot, we represented correlation coefficients which calculate the correlation between all variables in the data. As a result, we obtained which variables have statistical relationships, whether causal or not (Figure 6). We can observe that the Final grades do not correlate with 're-enrolled in the course' attribute. We find correlation between the Initial test and Variables 0.31, with Loops test 0.15 correlation, with Functions test -0.0096 correlation and everything after the Colloquium1 achieves much higher correlations with it.

With the Arrays test, the Colloquium1 has a correlation 0.48, while with the Data files test it has a correlation 0.58. The Colloquium1 correlates by 0.65 with the Final grade, and Colloquium2 achieves a much higher correlation of 0.84. The Practical exam pass correlates by 0.76, and the highest correlation is achieved by the Oral exam passed with 0.92.

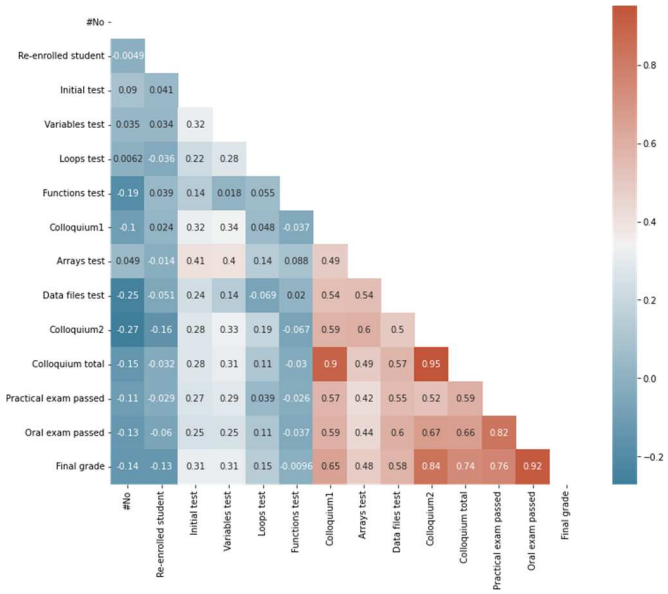

Fig. 6. Correlation analysis of data

The Practical exam pass has slightly less correlation with online tests. The Variables test it has a correlation 0.29, while with the Loops test it has a very small correlation of only 0.039, and with the Functions test again reaches a negative correlation of -0.026. The Practical exam pass and the Arrays test correlate 0.42, and the Data files test 0.55.

Online tests Arrays and Data files after the Colloquium1 have been shown in both cases to correlate more with both Initial test and Practical exam pass than online tests Variables, Loops and Functions before Colloquium1.

E. Prediction

As we found positive correlations between the Final grade and the Practical exam pass, except with the online tests Variables, Loops and Functions, we can use this data for modeling and prediction.

The first step is to replace the NaN values with zero, median and mean since we had much data with NaN values. In Colloquium2 the number of NaN values that appear was 96. After the replacement, we divided the data into variables that needed prediction and what to predict.

Afterwards, we split the dataset into training and testing data and then trained the various algorithms to find the best prediction model.

In Table 3, we can conclude that models filled with the median values obtained the best final grade prediction results. Also, models with zeros and mean values achieved very good results. The model with mean and median values in which we used Linear SVC was the best model for final grade prediction with an accuracy of 82.35%.

TABLE 3
PREDICTION OF FINAL GRADE

	Accuracy (zero data)	Accuracy (mean data)	Accuracy (median data)
SVC	0.794	0.794	0.794
Linear SVC	0.765	0.824	0.824
Random Forest	0.706	0.677	0.706
Logistic Regression	0.765	0.794	0.794
K Nearest Neighbors	0.735	0.735	0.765
Gaussian Naive Bayes	0.765	0.677	0.677
Decision Tree	0.765	0.735	0.706
MPL	0.765	0.794	0.794

TABLE 4
PREDICTION OF PRACTICAL PASSED

	Accuracy (zero data)	Accuracy (mean data)	Accuracy (median data)
SVC	0.5882	0.5882	0.5588
Linear SVC	0.5588	0.6764	0.6176
Random Forest	0.5294	0.5294	0.5588
Logistic Regression	0.5294	0.6176	0.6471
K Nearest Neighbours	0.6176	0.5588	0.5294
Gaussian Naive Bayes	0.6176	0.5588	0.5882
Decision Tree	0.5588	0.3824	0.4412
MPL	0.5294	0.6176	0.6471

The models with the lowest accuracy of 67.65% were those with the mean values using the Random Forest and Gaussian Naive Bayes algorithms, where Gaussian Naive Bayes also showed the same accuracy as the median values.

Table 4 shows models' accuracy in predicting Practical exam pass/fail scores. The model filled with the mean values achieved the best accuracy of 67.65% using the linear SVC algorithm. The model with the mean values data had the lowest accuracy of only 38.24% using the Decision Tree algorithm. Finally, Figure 7 shows the confusion matrix for the mean model using the Linear SVC algorithm that achieved the best Practical exam pass/fail score prediction. We can observe that out of 34 predicted values, 11 were incorrectly predicted. Of these 11 incorrectly predicted values, 5 had predicted value is 0 while the actual value was 1, and 6 had predicted value of 1 whereas the actual value was 0.

Models with mean and median values using the Linear SVC algorithm had the highest accuracy in the final grade prediction and the highest AUC score of 0.766. For the same models using the same algorithms, we obtained the highest Cohen's kappa score of 0.706.

Likewise, the lowest AUC score was 0.651 for also, models with mean and median values that used the Gaussian Naive Bayes algorithm achieved the lowest accuracy of all models the final grade prediction. We also got the lowest Cohen's kappa score of 0.47 for these models.

When predicting the practical exam pass/fail scores, the highest AUC score was 0.675 for the mean values data where we used the Linear SVC algorithm and had the highest accuracy of all models. The mean values data which used the MLP Classifier algorithm had slightly lower accuracy. For the same models we got the highest Cohen's kappa score of 0.348.

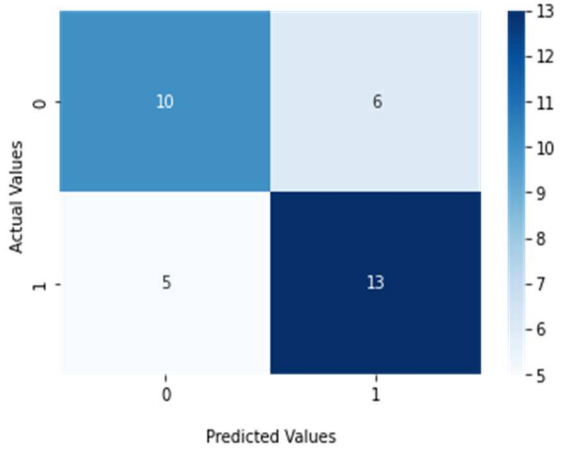

Fig. 7. Confusion matrix for data using Linear SVC algorithm

The model with mean values using the Decision Tree Classifier algorithm had the lowest AUC score of 0.437 for practical exam pass/fail prediction. We also got the lowest Cohen's kappa score of -0.176 for that model. We can notice that the model for practical exam pass/fail prediction had lower AUC score and Cohen's kappa scores than in the model for predicting the final grade prediction. Likewise, the models with the highest accuracy had the highest AUC and Cohen's kappa scores and vice versa.

The best results were given by models with mean values in both predictions, but also the worst ones only using different algorithms. When predicting the final grade, the model with median values was just as good as the model with mean values, but also worse. Besides, one of the challenges we encountered was the lack of data, where we presented the solution in three different ways to achieve the most efficient and accurate model performance. Second challenge was the fact that students accessed tests and exams online without teacher supervision. Consequently, there is always the possibility that one of the students cheated, which of course cannot be really detected.

V. Conclusion

This paper describes several steps of pre-processing the collected data to predict new data using different classification algorithms. We collected data from the first year undergraduate course Introduction to programming at the University of Split, Faculty of Science, Croatia. The results showed that the online tests Variables, Loops and Functions did not correlate with the final grade scores and the practical exam pass/fail score, unlike the other online tests Arrays and Data files. We did not get very accurate results in predicting the practical exam pass/fail score, although the results were promising. One of the limitations of this research was the lack of some data, which we had to replace it with zeros, median and mean values and examine which model would achieve the best accuracy prediction.

In this research, the main goal was to predict students' final grades and practical exam pass/fail scores. Using a data that was filled with median and mean values and using the Linear SVC algorithm, we achieved the best accuracy, AUC and Cohen kappa scores for predicting the final grade. Besides, predicting the practical exam pass, we got the best model results with mean values using the same algorithm. Therefore, there are important attributes that significantly influenced the success of predicting student final grades, which are Colloquium1 and Colloquium2, where Colloquium2 correlated slightly more with the Final Grade than Colloquium1. As we mentioned, the best algorithm for predicting the final grade was the Linear SVC algorithm, while the other algorithms also had very good results. Compared to the prediction of the practical exam pass, we had a very wide range of results of models we used. As a result, we did not have such a good accuracy, AUC and Cohen kappa scores, as well as in predicting the final grade because the variables used for prediction did not high correlate with the Practical exam pass variable. However, we wanted to test how good a model we could actually get with those variables.

Moodle data that predicts success would be very useful for teachers as well as students. Teachers could consider the attributes that most affect the final success of the students and monitor the progress of the students. Therefore, they can conclude which course modules are more difficult for the students in order to focus on better processing those modules so that students could achieve the best possible results. Students could also track their progress and based on their current data, consider the final grade the model predicts for them and accordingly study harder if the grade is not satisfying. For these reasons, this research contributes to teachers and students all over the world in order to enable the best possible advancement in education.

However, there are important factors such as teaching experience and technological equipment that negatively influence teachers' attitude to engage in digital learning objects (DLOs) and digital simulation tools (DSTs) [15]. For this reason, it is necessary to enable teachers to take courses that enable tutorials for tools and LMS used in certain educational institutions, and also to finance technological equipment in order to improve the e-learning system. Furthermore, many students use smartphones to access the LMS, but as the research [16] points out, most often for the purpose of data storage. Thus, educational institutions must emphasize that Moodle is one of the effective learning tools, and focus on the increasing emphasis on "student collaboration".

A limitation that appeared during this research is checking the student's knowledge online, for which it is not certain whether the student may have cheated on the test. Also, missing data affected the model itself, giving worse results. In the future, professors should exercise a little more supervision in such examinations so that they do not receive "false" data. In the future, we will work on a model that predicts success on the oral exam based on the practical exam. We used 9 data attributes to predict the final grade and 5 attributes to predict practical exam pass/fail scores, which may include even more
data attributes in the future.

We plan to introduce additional activities such as group assignments and some competitions in lectures and exercises to predict the grades and better determine individual student progress and success.

Acknowledgment

This paper was written as a research project for the graduate course "Learning Analytics" at the Faculty of Science, University of Split, Croatia.

References

[1] Dilkhaz, Yaseen & Mohammed, Dilkhaz. (2022). The web-based behavior of online learning: An evaluation of different countries during the COVID-19 pandemic. 263-267. 10.25082/AMLER.2022.01.010.

[2] Safsouf, Yassine & Mansouri, Khalifa & Poirier, Franck. (2021). TaBAT: Design and Experimentation of a Learning Analysis Dashboard for Teachers and Learners. Journal of Information Technology Education:Research. 20. 331-350. 10.28945/4820.

[3] Šarić-Grgić, Ines & Grubišić, Ani & Stankov, Slavomir & Robinson, Timothy. (2018). Concept-Based Learning in Blended Environments Using Intelligent Tutoring Systems.

[4] Anagnostopoulos, Theodoros & Kytagias, Christos & Xanthopoulos, Theodoros & Georgakopoulos, Ioannis & Salmon, I. & Psaromiligkos, Yannis. (2020). Intelligent Predictive Analytics for Identifying Students at Risk of Failure in Moodle Courses. 10.1007/978-3-030-49663-0 19.

[5] Ikawati, Yunia & Rasyid, M. & Winarno, Idris. (2021). Student Behavior Analysis to Predict Learning Styles Based Felder Silverman Model Using Ensemble Tree Method. EMITTER International Journal of Engineering Technology. 9. 92-106. 10.24003/emitter.v9i1.590.

[6] Hasan, Raza & Palaniappan, Sellappan & Mahmood, Salman & Abass, Ali & Sarker, Kamal. (2021). Dataset of Students' Performance Using Student Information System, Moodle and the Mobile Application "eD-ify". Data. 6. 1-10. 10.3390/data6110110.

[7] Lemay, David & Baek, Clare & Doleck, Tenzin. (2021). Comparison of

Learning Analytics and Educational Data Mining: A Topic Modeling Approach. Computers and Education: Artificial Intelligence. 2. 100016. 10.1016/j.caeai.2021.100016.

[8] Lopez-Zambrano, Javier & Lara, Juan & Romero, Cristobal. (2020). Towards Portability of Models for Predicting Students' Final Performance in University Courses Starting from Moodle Logs. Applied Sciences. 10. 354. 10.3390/app10010354.

[9] Hooshyar, Danial & Pedaste, Margus & Wang, Minhong & Huang, Yueh-Min & Lim, Heuiseok. (2020). Predicting course achievement of university students based on their procrastination behaviour on Moodle. Soft Computing. 24. 10.1007/s00500-020-05110-4.

[10] Rakic´, Slavko & Tasic, Nemanja & Marjanovic, Ugljesa & Softic, Selver & Luftenegger, Egon & Turcin, Ioan. (2020). Student Performance on an E-Learning Platform: Mixed Method Approach. International Journal of Emerging Technologies in Learning (iJET). 15. 187-203. 10.3991/ijet.v15i02.11646.

[11] Gkontzis, Andreas & Kotsiantis, Sotiris & Kalles, Dimitris & Panagiotakopoulos, Christos & Verykios, Vassilios. (2020). Polarity, emotions and online activity of students and tutors as features in predicting grades. Intelligent Decision Technologies. 14. 409-436. 10.3233/IDT-190137.

[12] Mart´ınez, Sonia & Agapito, Javier & Pamplona, Sonia. (2020). Early prediction of undergraduate Student's academic performance in completely online learning: A five-year study. Computers in Human Behavior. 115. 10.1016/j.chb.2020.106595.

[13] Tamada, Mariela & Giusti, Rafael & Netto, Jose´. (2022). Predicting Students at Risk of Dropout in Technical Course Using LMS Logs. Electronics. 11. 468. 10.3390/electronics11030468.

[14] Sunil & Doja, M.. (2017). Data Mining Techniques to Discover Students Visiting Patterns in E-learning Resources. International Journal of Computer Science and Mobile Computing (IJCSMC). 6. 363-368.

[15] Poultsakis, Stefanos & Papadakis, Stamatios & Kalogiannakis, Michail & Psycharis, Sarantos. (2021). The management of Digital Learning Objects of Natural Sciences and Digital Experiment Simulation Tools by teachers. 1. 58-71. 10.25082/AMLER.2021.02.002.

[16] Papadakis, Stamatios & Kalogiannakis, Michail & Sifaki, Eirini & Vidakis, Nikolas. (2018). Access Moodle Using Smart Mobile Phones. A Case Study in a Greek University. 10.1007/978-3-319-76908-0 36.