LAB University of Applied Sciences

# Thesis

## Using data analytics in Moodle to monitor and improve student performance

LAB University of Applied Sciences
Bachelor of Business Administration, Business Information Technology
2025
Mariia Sobinina

**Abstract**

| Author(s) | Publication type | Completion year |
|---|---|---|
| Mariia Sobinina | Thesis, UAS | 2025 |
| | Number of pages<br>36 | |

Title of the thesis

**Using data analytics in Moodle to monitor and improve student performance**

Degree, Field of Study

Bachelor of Business Administration, Business Information Technology

Name, title and organisation of the client

Juho Ratava, Associate professor, LUT

Abstract

The research explores the application of data analytics in Moodle learning management system to monitor and enhance student performance in higher education. The study analyses various performance metrics and what they can tell us about students' behaviour during the course. It investigates key factors affecting the student's success or failure. Predicting student success is important because it enhances learning outcomes and reduces drop-out events. Using a mixed-method approach, the research combines quantitative analysis of Moodle time logs and grading data with qualitative analysis of data gathered via questionnaire.

Machine learning model predicting project success was developed using MATLAB's Classification Learner app. Despite exploring multiple classification algorithms, the predictive model did not archive satisfactory accuracy, indicating limitations in the gathered data or problems in modelling approach. The study highlights correlations between learning behavior and academic outcomes.

Keywords

Learning Management System (LMS), Moodle, Learning analytics, Student success, Machine learning, MATLAB

Contents

Appendices

Appendix 1. Questionnaire form.

Appendix 2. MATLAB code (logs and grading events).

Appendix 3. MATLAB code (times).

Appendix 4. MATLAB code (merging tables and visualisation).

Appendix 5. MATLAB code (data normalisation and analysis).

# 1 Introduction

Face-to-face learning, online learning (e-learning) and blended learning are three common ways of learning nowadays, and each has its strengths and weaknesses. Face-to-face learning is a traditional model where students and teachers meet in person.

Online learning is defined as the opposite of the traditional method, its feature is that the physical classroom is replaced with out-of-class learning offered by web-based technologies (Nortvig et al. 2018). Online learning was discussed even in the late 1990s, still, only the COVID-19 pandemic accelerated the process of switching to online learning, as in many countries, it was the only possible way of learning during the lockdown (Gherheș et al. 2021, Cannaos 2024). During the pandemic, people noticed the main benefits of online learning: the students' ability to study at their own pace, the time economy and the reduction in travel expenses. (Cannaos 2024).

According to Graham (2013), blended learning can be defined as the integration of face-to-face and online instruction. Educational institutions widely adopt this method, as it is supposed to combine the advantages of traditional and online learning (Dziuban et al. 2018). Nevertheless, when the course is organised in a blended format, there is always a chance that some students will prefer an online format and skip offline classes. In this case, tracking progress, analysing performance and providing support can be challenging, as there is no contact interaction with a certain regular frequency.

## 1.1 Background and purpose of the study

Learning management systems (LMS) are widely used in modern teaching, especially for online (and blended) teaching and learning. Moodle is the most popular and preferred open-source LMS. (Gamage et al. 2022.)

This research aims to explore the usage of data analytics within Moodle to monitor student performance, focusing on online and blended learning. The study analyses various performance metrics (learning analytics) and what they can tell us about students' behaviour during the course. By examining grades and time logs, the research aims to provide insights into how Moodle data can identify struggling students, optimise instructional strategies, and provide better support to the students during the course. Additionally, the questionnaire answers providing information about background and studying habits are taken into account.

## 1.2 Research question

The main research question of this study is: How can data obtained from learning analytics be used to monitor and improve student performance? The study explores how completing course assignments during the semester relates to final assignment (project) success. It is checking if any consistent patterns can predict the final result of the course based on the progress during the semester and students' background and study habits. During the research process, the main trends are identified, and the possibility of building a model to identify students at high risk of failing the course is checked.

The primary data for the research includes data that is collected from Moodle, such as log data (timestamps and user actions), grading data (project and homework grades), and questionnaire answers (questions about educational background and study habits, together with consent to use their data). The data is preprocessed using Microsoft Office Excel and MATLAB.

The research focuses on identifying patterns related to students' academic performance. The success of the project is evaluated both on final and interim data. Descriptive statistics are used to show general patterns in student activity and grading. Data is anonymised and normalised. Correlation analysis is performed to assess the strength and direction of the relationship between the variables. The MATLAB Classification Learner is used to build and test models that classify students based on their activity data, performance, educational background and study habits using different machine learning algorithms. The models' performance is evaluated using different performance measures. The research is supported by visualisation, including scatter plots, histograms, and heatmaps, generated using Excel and MATLAB. The findings of the research are expected to contribute to a better understanding of how learning analytics can be used to monitor and enhance student performance.

The limitation of the study is that it focused on analysing grading and time log data retrieved from Moodle and students' data about their background and study habits gained from the survey. The data is gathered at the end of the fall semester, so all the participants are expected to complete all course activities. The research is missing the data about the students who lost motivation during the semester, and all failing results are the result of the low project grade.

The detailed description can be found in the separate chapter dedicated to the research process (Chapter 3).

# 2 Theoretical knowledge base

## 2.1 Learning management systems

A learning management system (LMS) is software that manages the teaching and learning process without the constraints of time and place (Sanchez et al. 2024). From the students' point of view, LMS is a modern educational tool that offers an easy and organised way to access educational materials, a personal calendar notifying about deadlines, and a platform for communication with peers and teachers (Gryshuk 2025). The main feature of LMS, which makes it suitable for any type of learning (including e-learning), is unlimited accessibility, as most LMSs can be accessed via an internet browser or a user application (Sanchez et al. 2024).

LMS is a convenient way for educators to create, organise and deliver educational courses (Gryshuk 2025). The Learning Tools Interoperability (LTI) allows LMS to integrate remote tools and content in a standardised way (1Edtech). In other words, it makes various educational tools and resources accessible directly from the LMS without needing to log in separately to these tools. Along with being a platform for delivering educational materials, LMS allows tracking and analysis of student performance data based on information on how students interact with learning materials (Qazdar et al. 2024).

There are two main types of LMS deployment: proprietary and open-source. An LMS is considered proprietary or commercial when its software is licensed under an exclusive legal right of the copyright holder (Pillai & Kevin 2013 according to Barreto et al, 9-17). Popular examples of proprietary LMSs are Blackboard, Desire2Learn, Litmos, Topyx, Saba (Barreto et al, 9-17). They usually come with vendor support and built-in integrations. They can cost a lot and are less flexible for customisation. In contrast, open-source LMS code is publicly accessible, it is available for extension and modification depending on the user's needs. (Cavus & Zabadi 2014). Examples of open-source LMS are Moodle, ATutor, Sakai, Forma LMS (Barreto et al, 9-17). Open-source systems offer greater autonomy and community-driven innovation.

## 2.2 Moodle

Martin Dougiamas, Moodle's Founder and CEO, designed one of the most popular LMS nowadays based on his personal distance learning experience via shortwave radio (the best available technology in the 1970s) and desire to enable high-quality education in all corners of the world. The system was called Moodle, which is the acronym for Martin's Object-Oriented Dynamic Learning Environment. He created the first course and made his first post on

Moodle.com website in November 2001. Moodle 1.0 was released in August 2002, and it continues to be actively developed. (Moodle c.) Nowadays, it is one of the best-known LMSs with more than 200 million users from 242 countries (Moodle d).

Moodle is an LMS used at LAB University of Applied Sciences and LUT University (eLAB, eLUT). Moodle is an open-source LMS. Moodle has a range of built-in analytic tools and plugins that help to collect, measure, analyse and report student data. Moodle provides LMS analytics through built-in reports, which are based on log data. (Moodle a.) Moodle has a Gradebook, and activities such as Assignments and Quizzes send grades to it. The course report section provides data about activity and participation, and general course logs. (Moodle b.) Moodle supports Learning Tools Interoperability (LTI) (Moodle e). Moodle can integrate with a wide range of external tools and applications, for example, MATLAB Grader. It is a browser-based tool developed by MathWorks which allows one to create, manage, deliver and automatically grade assessments from MATLAB. (MathWorks a).

## 2.3 Data analytics in education

According to Vanthienen & De Witte (2018, 2), data analytics is a set of techniques and applications that allow data exploration, analysis and visualisation. With the growing digitalisation of society, the volume of data produced has increased over the past decades. Every step a person takes in the online world is recorded. The usage of LMS generates a huge amount of data for every student and course. This data can be useful when it is analysed and converted into indicators reflecting the student's progress or course success. (Qazdar et al. 2024). Data analytics in education can provide plenty of opportunities to improve the learning and teaching experience (Vanthienen & De Witte 2018, 2).

From the student's perspective, data analytics helps by providing real-time feedback and enriching the learning experience. Data analytics is a useful tool for a teacher as it helps better trace and take targeted actions for improvement, creates quality indicators, and pays extra attention to possible fraudulent actions. As well, data analytics in education is used to measure the performance of teaching staff or educational institutions in general. (Vanthienen & De Witte 2018, 2-3.)

## 2.4 Learning analytics

There are different ways to define learning analytics (LA). The Society for Learning Analytics Research defines LA as "the measurement, collection, analysis, and reporting of data about learners and their contexts, for purposes of understanding and optimising learning and the environments in which it occurs", focusing mostly on the occurring processes (Nguyen et al.

2020). Nguyen et al. (2020) emphasise applying data analytics techniques and tools to enhance learning and teaching. The other approach is to define LA as software algorithms used for the prediction or detection of unknown aspects of the learning process based on historical data and current behaviour (Moodle f).

LA can be utilised in various ways depending on the aims of learners, teachers, and organisations. Key applications include understanding and predicting user behaviour, improving the courses, and facilitating early intervention and support for students at high risk of failing. Among the benefits of leveraging LA are making data-informed decisions, forecasting events and performance based on patterns and trends, and reducing costs. The common LA data sources are login frequency, resource usage, course progress, assessment results, completion rates, and feedback surveys. (Digital Learning Institute.) LA are interesting to researchers and teachers as LA can reveal hidden patterns in students' behaviour.

The Society for Learning Analytics Research mentions four common LA methodologies. These are descriptive, diagnostic, predictive and prescriptive analytics. Descriptive analytics describes what happened in the past; it is usually based on feedback and data analysis of all stages of the student lifecycle. Diagnostic analytics is applied to understand and find reasons why something happened. Predictive analytics is used to make forecasts and predictions based on historical data and identified patterns. Prescriptive analytics advises on possible outcomes based on data-driven recommendations. (Digital Learning Institute, SOLAR.)

## 2.5 Educational data mining

Data mining is a process of identifying patterns and extracting useful insights from large data sets using machine learning and statistical analysis (IBM 2024). Educational data mining is a process that converts raw educational data into possibly useful information for educational research and practices (Hooshyar et al. 2020). The data mining techniques that can be applied in education are classification, clustering, regression, social network analysis, and pattern mining (Vanthienen & De Witte 2018, 13).

Classification is a technique where data is grouped into predefined classes. Clustering is similar to classification, but it divides data points into groups according to similarities or differences, and these groups are not predefined. Regression analysis is the technique which finds relationships in data by predicting outcomes based on predetermined variables. (IBM 2024). Predicting whether a student passes or fails the course based on engagement data and academic performance during the semester is an example of classification. An example

of clustering can be grouping students based on similar backgrounds and learning behaviours. Regression can be used to predict student final grades based on homework submissions (Vanthienen & De Witte 2018, 13).

## 2.6 Machine learning

Machine learning is a branch of artificial intelligence that teaches computers to learn from data and make decisions and predictions without being directly programmed (IBM 2021).

Machine learning has two approaches: supervised learning (classification and regression and unsupervised learning (clustering). In supervised learning, the algorithm is trained on labelled data (input-output pairs) so it can learn to predict outcomes for new, unseen data. In unsupervised learning, the algorithm finds hidden patterns or intrinsic structures in input data. (IBM 2021, MathWorks c.)

A machine learning model is an algorithm that identifies patterns within the data to perform classification or regression tasks. Common classification models include linear regression, logistic regression (used for yes/no predictions), decision trees (which split data into branches like trees), support vector machines (which draw lines to separate groups), k-nearest neighbors (which look at the closest examples), and neural networks (IBM, IBM 2021, IBM 2023).

## 2.7 Related research

Many researchers have studied how data collected from LMS can be used to predict learning outcomes. Different research articles define various variables to be predicted. Many potential variables can be used as prediction outcomes, but the most common ones in the literature are learners' behaviour and academic performance, such as grades and the risk of dropping out. Predicting the risk of failing is usually based on forecasting learning outcomes (grades), which can be done by predicting success (pass/fail) or numerical or alphabetical grades (Pigeau et al. 2019, Li et al. 2017 according to Moreno-Marcos et al. 2020). Pérez-Lemonche et al. (2017) and Nachouki et al. (2023) predicted the final course grade using machine learning. Sweeney et al. (2015) tried to predict students' course grades for the next enrollment term.

Another important aspect is the predictors, or the selected variables used to build the predictive models. There is a great variety of choices for predictors. Just as example the following categories of predictors can be mentioned: content consumption activity (logs to the course page, total number of accesses to the contents), course communication engagement (number of messages, number of words in the messages, number of ratings emitted/received),

time-related (time between the assignment opened and deadline, time of completion the assignment), demographical (age, gender, race, language skills, location), work experience, educational background, enrolment-related (payment fee for the course, signup date), attendance, delivery mode (face-to-face, online or hybrid) and so on (Sulaiman& Mohezar 2006, Moreno-Marcos et al. 2020). Ruipérez-Valiente et al. (2017) found that the strongest predictor was the progress in problems (i.e. the grade in completed assignments), and Ren et al. (2016) mentioned engagement to be a strong predictor (Moreno-Marcos et al. 2020).

Namoun & Alshanqiti (2021) analysed 62 articles focused on predicting student performance using learning analytics and found that most studies used either statistical models (about 45%) or supervised learning models (about 40%) to predict academic performance. The authors mention that regression analysis was the most frequently used prediction technique (about 52%), artificial neural networks and tree-based models ranked second overall (each about 15%) (Namoun & Alshanqiti 2021).

## 2.8 Research methods

### 2.8.1 Data anonymisation.

Data anonymisation is used if the collected data contains personally identifiable information. Various data anonymisation techniques exist to protect individual privacy and reduce the risk of unauthorised disclosure. (Murthy et al. 2019.)

Data masking hides the original data with modified content. It has two branches of methods: pseudonymization and anonymisation. Pseudonymization is a process during which the data is transformed into another form, which keeps privacy. It has several techniques, such as masking, tokenisation, and data blurring. Data masking is a good way to get rid of students' names and IDs, as they can be masked with student numbers (for example, Student 001). Anonymisation is a process of encrypting or removing personal information from data, it can destroy a pattern of data. The anonymisation techniques are suppression and generalisation. (Tachepun & Thammaboosadee, 2020). Data suppression is removing an entire part of the data in a dataset or replacing it with a value that does not have meaning (for example, "*****"). Data suppression is used for columns with sensitive data not required for analysis (for example, IP address). (Murthy et al. 2019.)

### 2.8.2 Correlation analysis

In statistics, the correlation refers to the relationship between two variables. Correlation is a statistical measure that describes the strength and direction of a relationship between two

variables. Positive correlation occurs when two variables change similarly, increasing or decreasing together. A negative correlation is where one variable increases and the other decreases. The strength of correlation is measured by the correlation coefficient r. If |r| is close to 1, the variables are strongly correlated, |r| >0,7 – the variables are highly correlated, 0,5 < |r| < 0,7 – the variables are moderately correlated, and 0,3 < |r| < 0,5 - the variables have a low correlation. (Andrews.edu.) In MATLAB, the corrcoef function computes the Pearson correlation coefficient between variables (MathWorks e).

### 2.8.3 Classification Learner

MATLAB Classification Learner is an application which trains models to classify data using supervised machine learning. It allows performing automated training to find the best classification model type. (MathWorks b.)

MATLAB's Classification Learner app requires identifying the response and the predictors, the type of validation and whether part of the data is left as a testing set.

Model validation is used to estimate how well a trained model will perform on new data and to prevent overfitting. Common methods include cross-validation, holdout validation, and resubstitution. Cross-validation is recommended for small datasets as it divides the data into parts, trains the model multiple times, and averages the results for reliable accuracy. Holdout validation is faster and more suitable for large datasets, using a portion of the data for validation and the rest for training. Resubstitution uses the same data for both training and testing, often leading to overfitting and unrealistic performance estimates, and is generally not recommended (MathWorks d).

### 2.8.4 Model performance measures

In classification analysis, the confusion matrix is one of the main tools for evaluating a classifier's performance. In Table 1, the columns represent the predictions, and the rows are for the actual classes. TP (true positive) is the number of correctly classified positive cases, TN (true negative) is the number of correctly classified negative cases, FN (false negative) is the number of positive cases incorrectly classified as negatives, and FP (false positive) is the number of negative cases incorrectly classified as positive. (Akosa 2017).

Table 1. Confusion matrix for two-class classification

| | Classified positive | Classified negative |
|---|---|---|
| Actual positive | TP = True Positive | FN = False Negative |
| Actual negative | FP = False Positive | TN = True Negative |

Accuracy is the most commonly reported model evaluation metric. Accurary evaluates the overall efficiency of the algorithm. Accuracy is calculated with the formula (1), where TP, TN, FP and FN are confusion matrix items as presented in Table 1. Unfortunately, accuracy can be a misleading evaluation measure if the data is unbalanced. (Akosa 2017).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

Sensitivity measures the accuracy of positive cases. Sensitivity measures how well the classifier identifies the positive / minority class. Specificity measures the accuracy of negative cases. Specificity tells more about the classifier's efficiency on the negative / majority class. Sensitivity is calculated with the formula (2) and specificity is calculated with the formula (3), where TP, TN, FP and FN are confusion matrix items as presented in Table 1. (Akosa 2017, Chicco et al. 2021.)

$$Sensitivity = \frac{TP}{TP + FN} \quad (2)$$

$$Specificity = \frac{TN}{TN + FP} \quad (3)$$

There are different combined performance measures such as geometric mean, discriminant power, F-measure, balanced accuracy, Matthew's correlation coefficient, Cohen's Kappa, Youden's Index, and likelihoods (Akosa 2017). According to Akosa (2017), it is important to consider a combination of different measures when dealing with class-imbalanced data. Chicco et al. (2021) report that the Matthews correlation coefficient (MCC) is more reliable than balanced accuracy in two-class evaluation.

The balanced accuracy, which is calculated with the formula (4), is the average of sensitivity and specificity. It gives a fairer measure when the classes are not balanced. If a model only does well because it focuses on the bigger class, balanced accuracy will be lower than regular accuracy. (Chicco et al. 2021.)

$$Balanced\ accuracy = \frac{Sensitivity + Specificity}{2} \quad (4)$$

The Matthews correlation coefficient (MCC) works well with imbalanced data. MCC is calculated with the formula (5), where TP, TN, FP and FN are confusion matrix items as presented in Table 1. It is a correlation coefficient between the observed and predicted classifications. The value goes from -1 to +1: +1 means perfect predictions, 0 stands for random predictions, –1 means the predictions are completely wrong. (Chicco et al. 2021.)

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (5)$$

Kohen's Kappa (or Kappa) measures how much better a model's accuracy is compared to expected by random chance. Kappa is calculated with the formula (6). The range is the same as for MCC described earlier (Akosa 2017).

$$Kappa = \frac{Total\ accuracy - Random\ accuracy}{1 - Random\ accuracy} \quad (6)$$

where

$$Total\ accuracy\ (accuracy) = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

$$Random\ accuracy = \frac{(TN + FP)(TN + FN) + (FN + TP)(FP + TP)}{(TP + TN + FP + FN)^2} \quad (8)$$

where TP, TN, FP and FN are confusion matrix items as presented in Table 1.

# 3 Research

## 3.1 Research overview

The primary data for this research was collected from Moodle and consists of log data, grading data, and questionnaire responses from a single course over one semester. The data was preprocessed using Microsoft Office Excel and MATLAB. A correlation matrix was created and visualised as a heatmap in MATLAB to explore relationships between variables.

Classification models were developed using the MATLAB Classification Learner. Two types of analyses were conducted: one model based on data from the entire semester, and another model for early prediction, based on data collected at the beginning of the course. The models aimed to predict project completion outcomes and classify students as either passing or failing the project. The predictors used included homework grades, the median time to complete homework, the total number of logs and grading events, the homework submission day, and questionnaire results. For each model, the predictors were selected (predictors varied depending on the model), and the data were divided into a validation (80%) and a test set (20%).

Data were anonymised and normalised before analysis. Descriptive statistics were used to show general patterns in student activity and grading. Correlation analysis assessed the strength and direction of relationships between variables. Model performance was evaluated using a range of classification performance metrics to determine the predictive accuracy and reliability of the machine learning algorithms.

## 3.2 Data collection

### 3.2.1 Course overview

Data for the research was collected during one semester. Participants were Master's degree students enrolled in the course LES10A170 Applied Mathematics I at Lappeenranta–Lahti University of Technology LUT, during the Autumn 2024 semester and consented to use their data. The general number of students registered for the course was 267, and only 224 students participated (43 students suspended their participation in the course). 180 students completed the required assignments and received grades. A total of 109 students gave consent to use their data.

The course grade was calculated as an average of homework completion (50%) and project work results (50%). Homework consisted of 11 sets, and each set included tasks equal to 5

points (55 points total). Homework assignments were created with MATLAB Grader. Students had an unlimited number of attempts to pass the tests, but they had a time limit of one week to complete a set of exercises. The project work was a final task and was mandatory to pass the course. The maximum for project work was 20 points.

### 3.2.2 Moodle data

Moodle provides useful learning analytics, such as when the student last time accessed the course page. This data can be viewed in the Participant's section. Students' grades were collected from the Grades section on the Moodle course page.

Moodle allows checking the logs in the Report's section, the logs give extensive information about what is happening on the course page. The logging event has the following information:

- Time: The time column consists of the date and time
- User full name: The user performing the action
- Affected user: The recipient of the action (in our case, the user receiving the grade)

  Depending on the event, the data in one of these columns can be missing. For example, for viewing a course module, only the username is visible; viewing a course module does not affect other users. But for "User graded"-event only the affected user is presented (as grading is done by the system).
- Event context: The context provides information about the course module.
- Component: The module type (for example, File, System, External tool)
- Event name: The type of action performed on the module (for example, Course viewed, Course module viewed, User graded)
- Description: It is a description of the event, where the IDs of users and elements are mentioned.
- Origin: the selected sources
- IP address

The description of the fields is partially taken from this source (Rotelli & Monreale 2023).

### 3.2.3 Survey questionnaire.

A questionnaire was created to gather information about prior programming background and level of education, current working status, moving due to the studies and study habits. The questionnaire form is in Appendix 1.

The survey was live on the Moodle page of the course for one month starting 25 November 2024. A total of 124 students completed the survey, and 109 gave consent to use their data.

## 3.3 Data analysis

The logs report does not provide information about what grade was given to the student; it just gives information that the user was graded. Also, it does not track how long the user stayed on this page. To simplify the calculations, the assumption was made that the time required for solving the exercise is equal to the difference between the first access to the exercise page and the time of receiving the best grade for this exercise. The idea is to compare this data with similar data, so this assumption seems adequate.

The questionnaire results can provide some interesting course statistics and shed some light on outliers and explain them. Also, it could be useful for creating groups of students with similar backgrounds or studying habits and analysing them as a group.

### 3.3.1 Analysis of time-log data, homework and project grades

The raw data was collected from the Moodle gradebook. The initial analysis of homework and project grades was done in Excel: total homework and project points were converted into completion share based on the maximum score.

The files containing time logs and questionnaire information were downloaded from Moodle and loaded into MATLAB, as well as the file with grade data. From the questionnaire file, the students who gave consent to use their data were selected. All later data manipulations are done only with the data, which refers to students giving consent to use their data.

MATLAB is a great tool for processing large volumes of uniform data. MATLAB's statistical and visualisation tools were used for a descriptive analysis of the data.

The MATLAB code (Appendix 3) calculates the time required for each exercise completion for each student as the difference between the time when the student opened the exercise for the first time and the time when the student received the highest grade. Also, it checks on which day of the week the student submitted the best solution. The code generates summary statistics to describe the data (mean, median, minimum and maximum submission

time and the most frequent submission day) based on time logs data for all exercises for each student. The total number of students' logs to the course page and grading attempts was also recorded (Appendix 2). These data values are used in further analysis.

The grade, times and questionnaire tables were merged. Relationships between variables were visualised using a scatter plot (Appendix 4). Figures 1-6 include data about both passed and failed the course. Red points represent students who failed the course, while black points indicate those who passed. Different time statistics were analysed, but in further analysis, only the median time is used, as this value gives a more detailed idea of the student's behaviour.

According to Figure 1, most students with high homework scores also performed well in the project, suggesting a positive trend. However, a few students with high homework scores still failed, indicating that consistent homework performance does not always lead to project success.

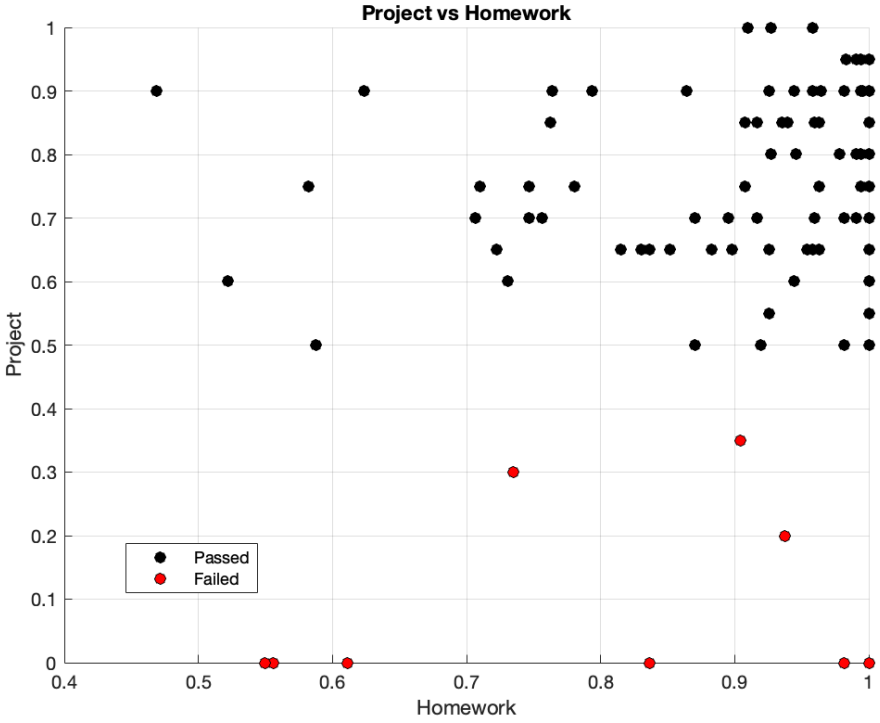


Figure 1. Scatter plot: the relationship between students' homework completion and project grades.

The relationships between log data and students' performance are presented in Figures 2 and 3. The relationship between the homework grade and the total number of grading attempts is presented in Figure 4. All figures show that students with higher project and homework scores tend to have more log entries, suggesting a possible correlation between engagement (as measured by logging activity or grading attempts) and performance. However, some students with frequent log entries still failed the course, and on the other hand, students with low numbers of logs performed well, same of grading attempts.

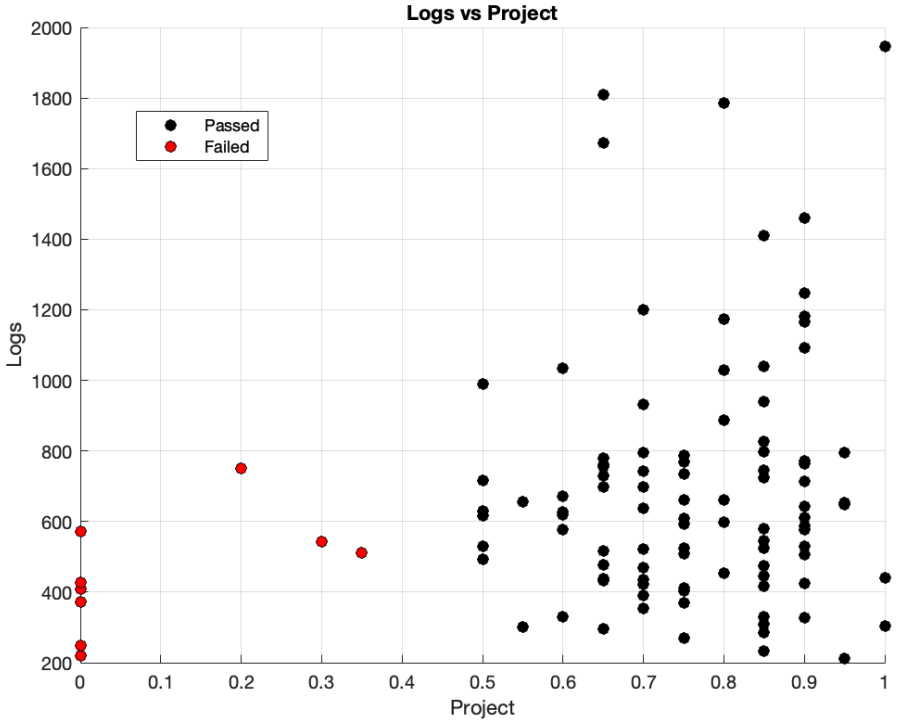


Figure 2. Scatter plot: the relationship between the number of log entries and students' performance in the project.

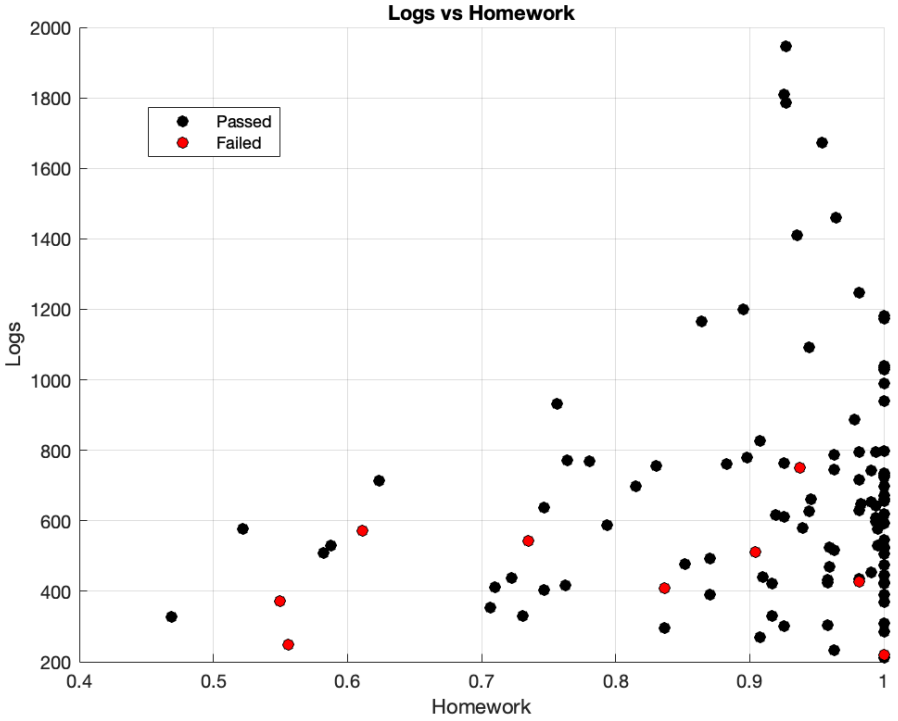


Figure 3. Scatter plot: the relationship between the number of log entries and students' performance on homework.

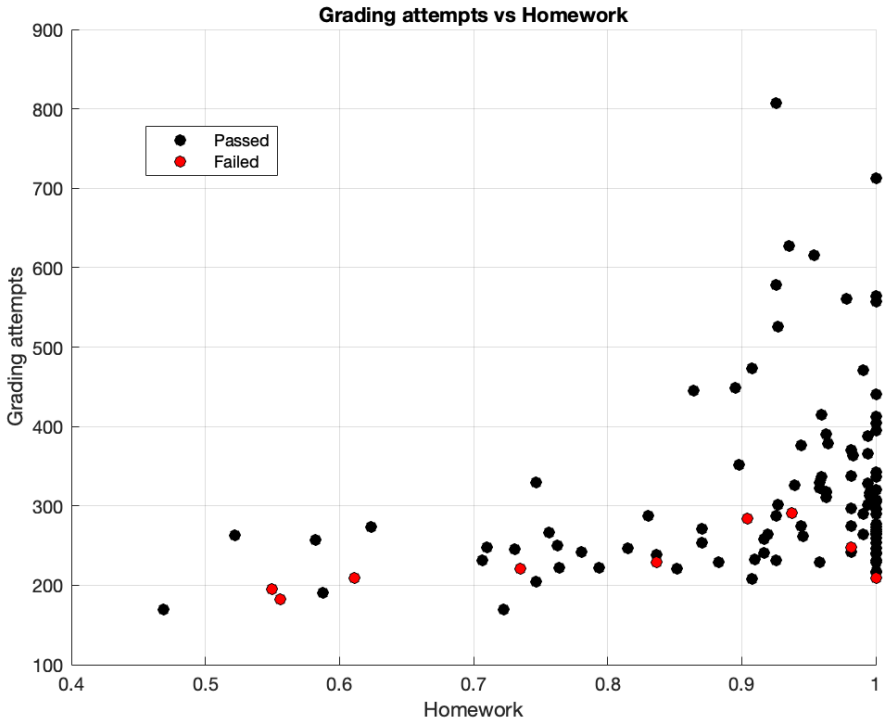


Figure 4. Scatter plot: the relationship between the homework grade and the total number of grading attempts.

Figure 5 presents the relationship between the median time (on a logarithmic scale) and the total number of log entries. While most students cluster around lower median times and moderate logging activity, there is significant variability across the range. Failed students are concentrated in the lower logging range and lower median time, suggesting that limited engagement may be associated with weaker outcomes. However, the majority of students with the same indicators passed the course.

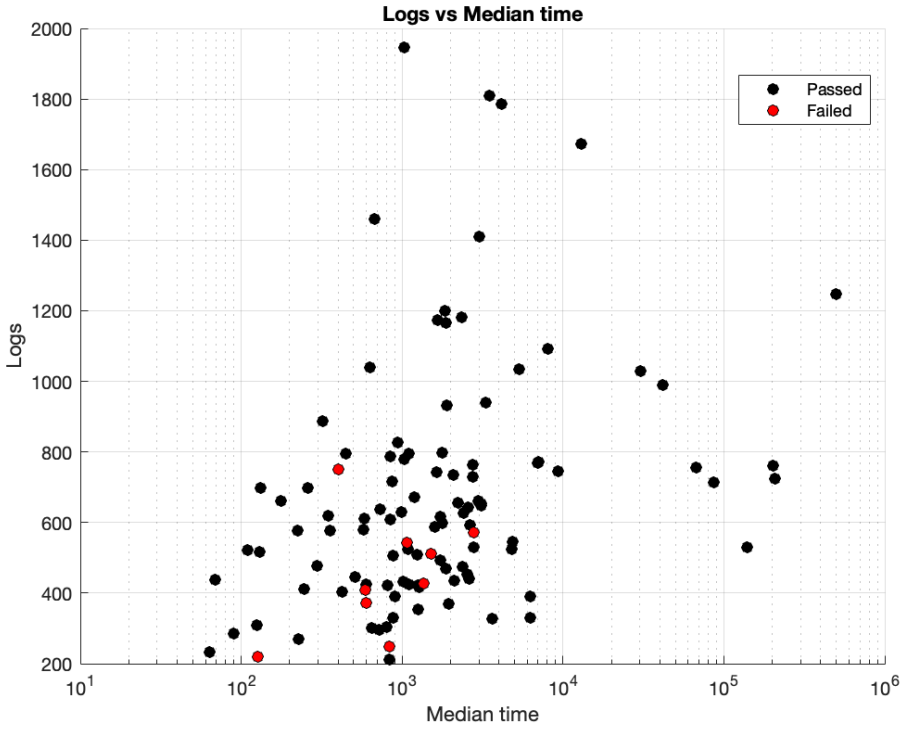


Figure 5. Scatter plot: the relationship between the median time (on a logarithmic scale) and the total number of log entries.

Figure 6 provides information about the relationship between the homework submission day and the total number of log entries. In most cases, the deadline for returning exercise sets was Monday at 23:59. Most submissions are made on Monday, indicating a tendency for last-minute returns. Failed students are concentrated in the same zone, so last-minute submissions may correlate with lower outcomes.

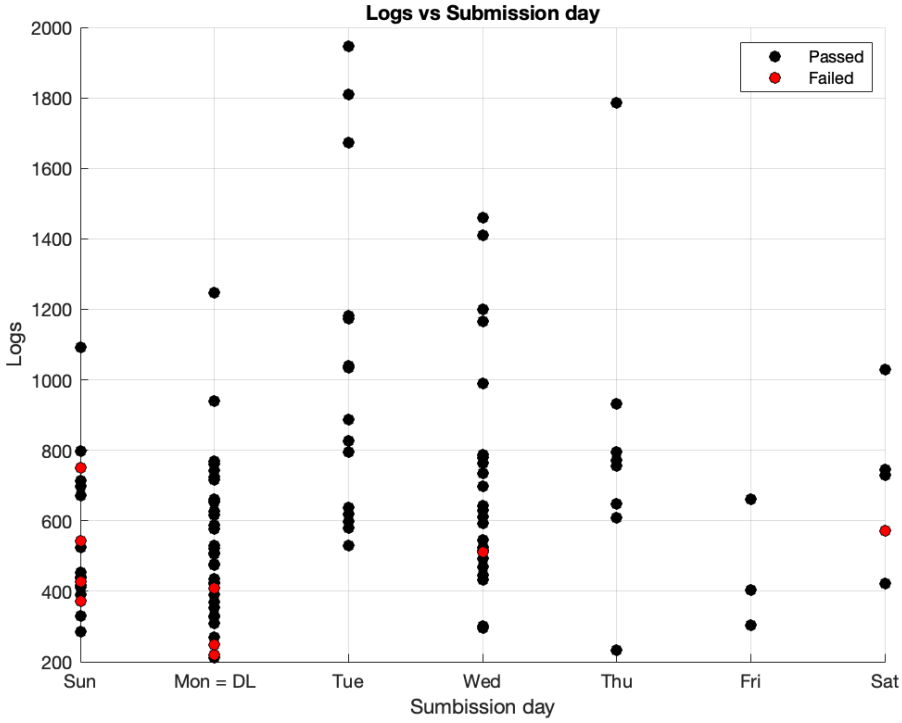


Figure 6. Scatter plot: the relationship between the submission day of the homework and the total number of log entries. DL stands for deadline.

Additionally, questionnaire results were visualised as histograms, that can be seen in Figure 7.

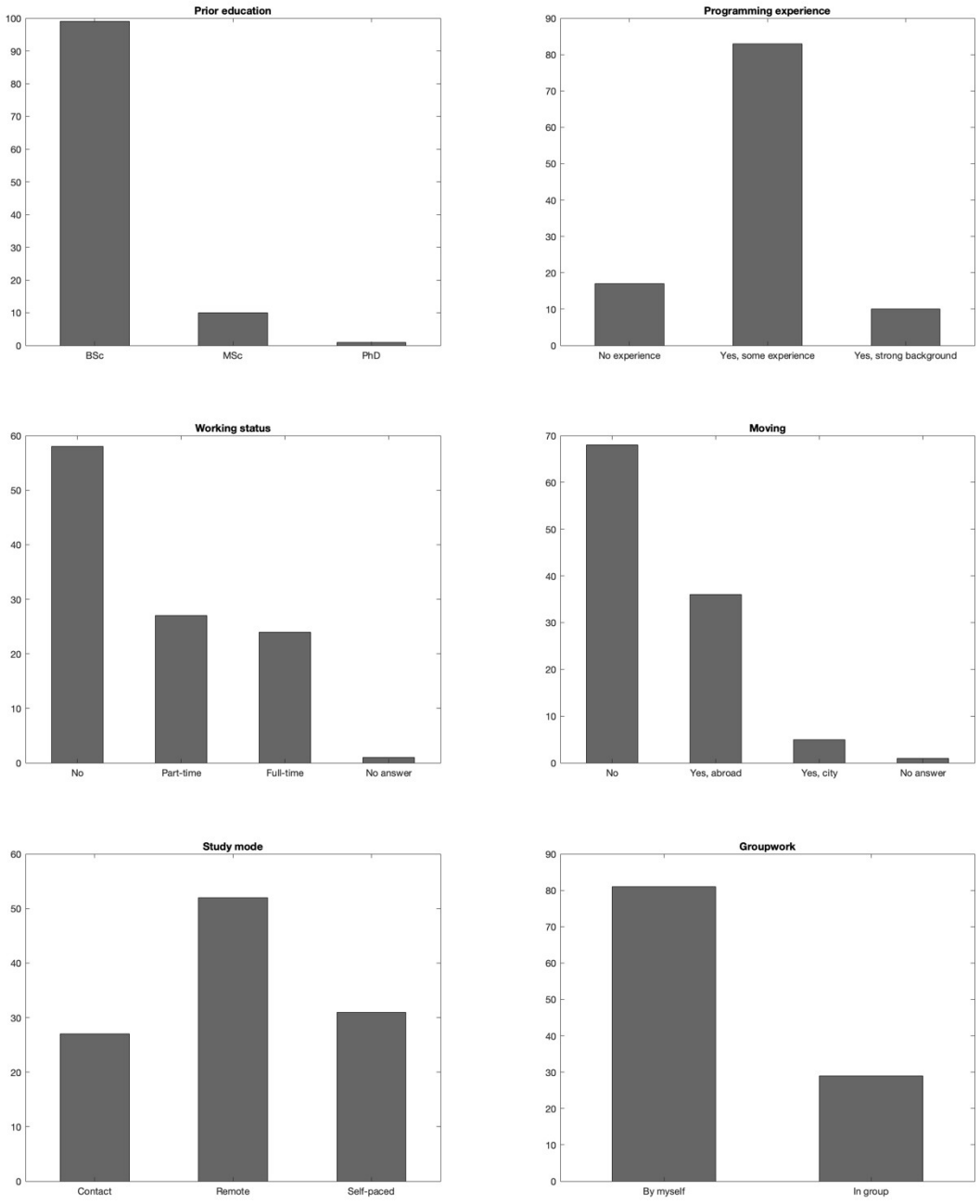


Figure 7. Histograms: the questionnaire results.

The histograms illustrate various background characteristics of the student population. The course is a part of a Master's degree, but few students already have a Master's or PhD degree. Most students reported having some programming experience, while only a small number had a strong background or no experience at all. About half of the students are not working, and the second half are employed either part-time or full-time. Regarding relocation, most students had not moved for their studies, but a significant number had moved from abroad, and a few had moved within Finland. Regarding study mode, remote learning

was the most common, followed by self-paced and contact (in-person) learning. Finally, the majority of students preferred to work by themselves rather than in groups.

### 3.3.2 Data normalisation and correlation analysis

The merged table was modified and normalised (Appendix 4). In other words, the values were modified in such a way that they fall between 0 and 1. The maximum values for the time-related data were found among all the students, and the data were normalised by dividing each value by this maximum value. For the responses to the questionnaire, the values from 0 to 1 corresponding to the answers were assigned; the step depended on the number of response options. Normalising the submission day was performed as follows: value 0 corresponds to the day when the exercise set was opened, and value 1 corresponds to the deadline. All other weekdays have values between 0 and 1 with a step of 0,167.

Based on the results, a correlation matrix and a heatmap were created (Figure 8).

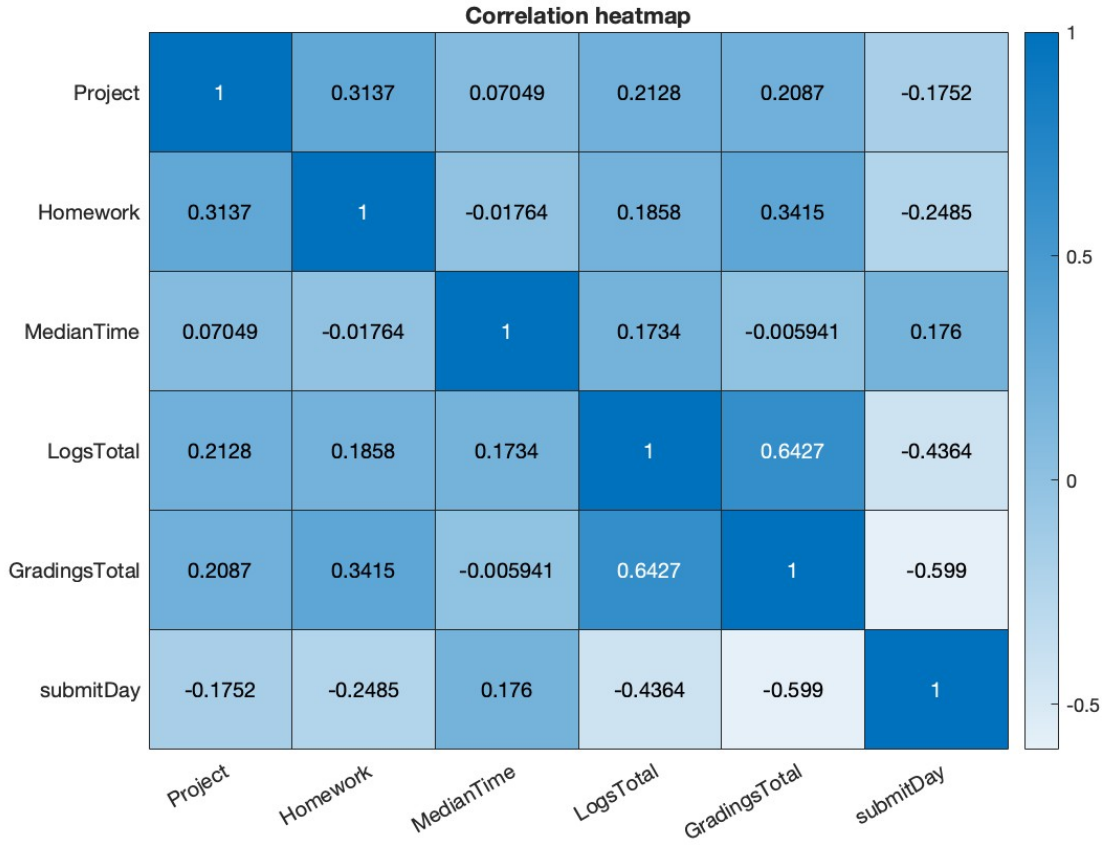


Figure 8. Correlation heatmap.

There is a high correlation between logs and grading attempts, which is easy to understand as increased student activity in the system naturally leads to more grading events. Unfortunately, no other strong correlation between the variables was observed. There is a low positive correlation between homework and project grades, and a low negative correlation between the total number of student logs and how close to the deadline the student submits homework.

The positive correlation between homework grades and project grades was expected, and the results obtained were consistent with the data presented in the articles. Fernandez et al. (2006) report little correlation between individual student performance on homework and other activities such as quizzes, tests and final examinations. The authors mention that high homework scores do not lead to high test scores (Fernandez et al. 2006).

The second observed relationship is between the variables indicating how close to the deadlines the student submitted homework assignments on average and the total number of the student's logs to the course page on Moodle. It means that students with more total logs returned their homework in advance. That can be explained by their interest in the course or high motivation.

### 3.3.3 Prediction of project pass based on the course results and questionnaire answers

MATLAB Classification Learner was used to classify the data and build the prediction models. Figure 9 shows the starting window of the MATLAB Classification Learner, where the main parts of classification are defined: response, predictors, validation scheme and setting aside data for test. The response was initially the project grade (later changed to project completion – passing or failing the project). The predictors were all discussed data above: homework grade, median time for homework completion, the total number of logs and grading events, the homework submission day and questionnaire answers (information about prior programming background and level of education, current working status, moving due to the studies and study habits). The default validation option (5-fold cross-validation) was used. 20% of the data was set aside or reserved for testing; it was used as test data to evaluate the model's performance after training. In other words, in total, the entire dataset consists of 109 records: 88 records are used for learning (validation set) and 21 records are used for testing (test set) (Table 2).

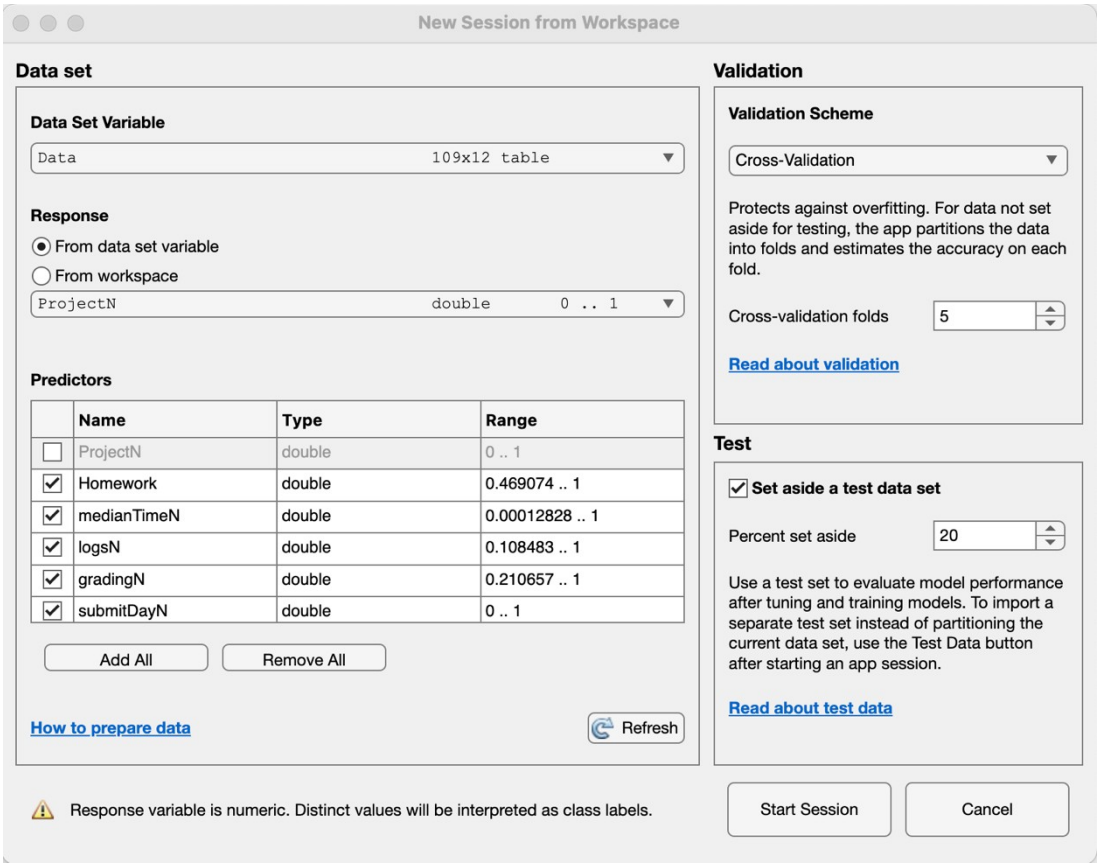


Figure 9. MATLAB Classification Learner. Starting window.

Table 2. Actual passes and fails.

| | Validation set | Test set |
|---|---|---|
| Actual “fail” | 8 | 1 |
| Actual “pass" | 80 | 20 |

After starting the session, all the possible models were trained and later tested.

In the first attempt, grades were input as continuous values between 0 and 1, which led to poor accuracy (the highest validation accuracy, about 19% for the Neural network model). The result of the training for different models is presented in Figure 10.

**Session:** ClassificationLearnerSessionProject
**Training Data:** Data Observations: 88 Predictors: 10 Response Name: Project Response Classes: 12
**Validation:** 5-fold cross-validation
**Test Data:** Data Observations: 21

| Favorite | Model Number | Model Type | Status | Accuracy (Validation) | Accuracy (Test) |
|---|---|---|---|---|---|
| ☐ | 2.28 | Neural Network | Trained | 19.32 % | - |
| ☐ | 2.30 | Neural Network | Trained | 19.32 % | - |
| ☐ | 2.16 | KNN | Trained | 17.05 % | - |
| ☐ | 2.21 | KNN | Trained | 15.91 % | - |
| ☐ | 2.24 | Ensemble | Trained | 15.91 % | - |
| ☐ | 2.7 | Efficient Linear SVM | Trained | 14.77 % | - |

Figure 10. Model 1: Grades were input as continuous values between 0 and 1.

In the second attempt, the problem was simplified by converting grades into binary values - pass (1) or fail (0). This significantly improved the model's performance, achieving around 90% accuracy for the validation set and 95% for the test set. The result of training and testing different models is presented in Figure 11.

**Session:** ClassificationLearnerSession-final
**Training Data:** Data Observations: 88 Predictors: 11 Response Name: ProjectN Response Classes: 2
**Validation:** 5-fold cross-validation
**Test Data:** Data Observations: 21

| Favorite | Model Number | Model Type | Status | Accuracy (Validation) | Accuracy (Test) |
|---|---|---|---|---|---|
| ☐ | 2.7 | Efficient Logistic Regression | Tested | 90.91 % | 95.24 % |
| ☐ | 2.8 | Efficient Linear SVM | Tested | 90.91 % | 95.24 % |
| ☐ | 2.11 | SVM | Tested | 90.91 % | 95.24 % |
| ☐ | 2.14 | SVM | Tested | 90.91 % | 95.24 % |
| ☐ | 2.15 | SVM | Tested | 90.91 % | 95.24 % |
| ☐ | 2.16 | SVM | Tested | 90.91 % | 95.24 % |
| ☐ | 2.18 | KNN | Tested | 90.91 % | 90.48 % |
| ☐ | 2.19 | KNN | Tested | 90.91 % | 95.24 % |
| ☐ | 2.20 | KNN | Tested | 90.91 % | 95.24 % |
| ☐ | 2.21 | KNN | Tested | 90.91 % | 95.24 % |
| ☐ | 2.23 | Ensemble | Tested | 90.91 % | 95.24 % |
| ☐ | 2.33 | Kernel | Tested | 90.91 % | 95.24 % |
| ☐ | 2.34 | Kernel | Tested | 90.91 % | 95.24 % |
| ☐ | 2.4 | Discriminant | Tested | 89.77 % | 90.48 % |

Figure 11. Model 2: Grades were binary: 0 = failed or 1 = passed.

The models give high percentage accuracy but are not effective in predicting failing grades (Figures 12 and 13). Figure 12 represents the original dataset. In Figure 13, the correct model predictions are marked as dots, and the incorrect ones with crosses. It means that the prediction is worse than random.

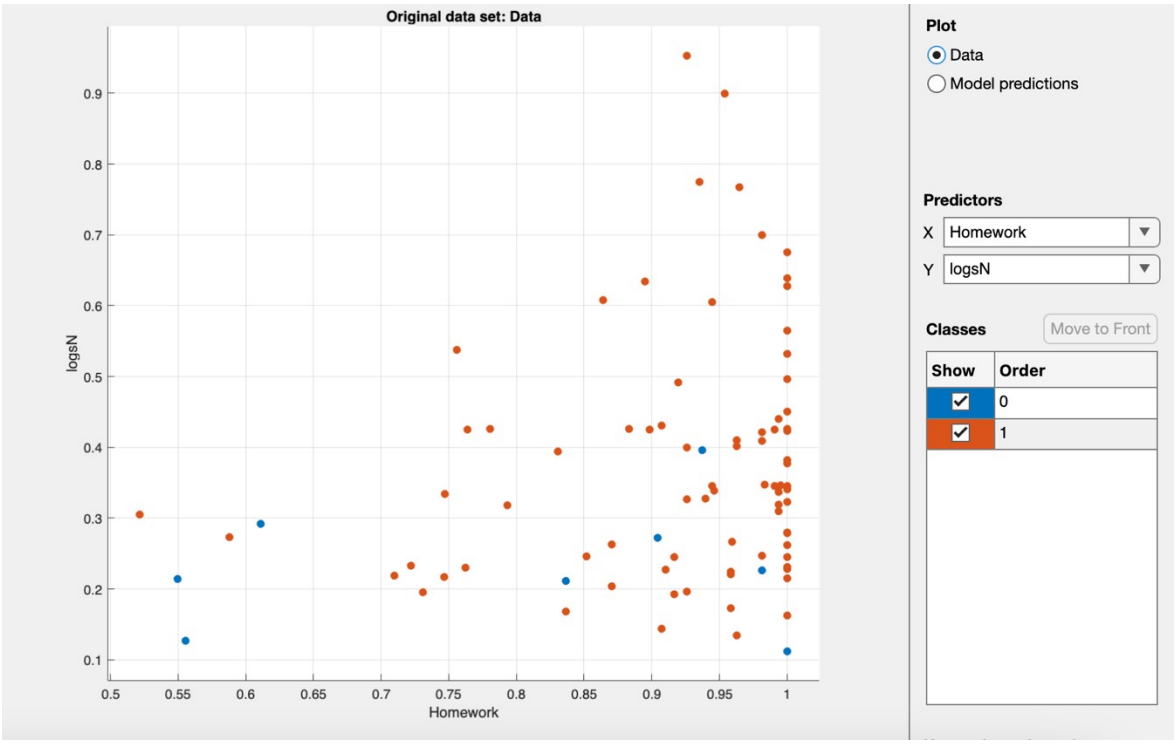


Figure 12. Model 2.7: Efficient Logistic Regression. Scatter plot: Logs vs Homework (training set). The original data set.

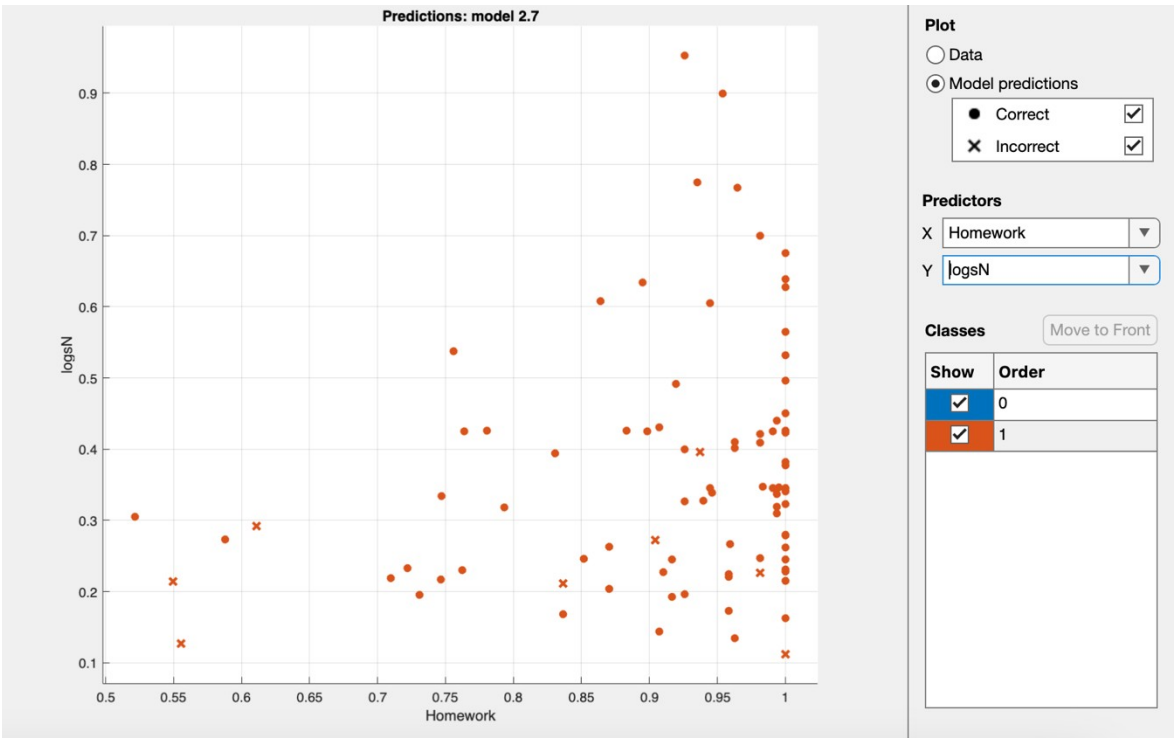


Figure 13. Model 2.7: Efficient Logistic Regression. Scatter plot: Logs vs Homework (training set). Model predictions.

For each model, the validation matrices are provided by MATLAB Classification Learner. As a demonstration example, the confusion matrices for 2.7 Efficient Logistic Regression and 2.4 Linear Discriminant models for both validation and test sets are presented in Tables 3-6. The Linear Discriminant model (compared to the Efficient Logistic Regression model) is better in predicting failure in the validation set, but fails in the test set too.

Table 3. Model 2.7: Efficient Logistic Regression. Confusion matrix: validation set.

| | Classified “fail” | Classified “pass” |
|---|---|---|
| Actual “fail” | 0 | 8 |
| Actual “pass" | 0 | 80 |

Table 4. Model 2.7: Efficient Logistic Regression. Confusion matrix: test set.

| | Classified “fail” | Classified “pass” |
|---|---|---|
| Actual “fail” | 0 | 1 |
| Actual “pass" | 0 | 20 |

Table 5. Model 2.4: Linear Discriminant. Confusion matrix: validation set.

| | Classified “fail” | Classified “pass” |
|---|---|---|
| Actual “fail” | 2 | 6 |
| Actual “pass" | 3 | 77 |

Table 6. Model 2.4: Linear Discriminant. Confusion matrix: test set.

| | Classified “fail” | Classified “pass” |
|---|---|---|
| Actual “fail” | 0 | 1 |
| Actual “pass" | 1 | 19 |

The balanced accuracy and Kappa are evaluated on par with the accuracy provided by MATLAB. Also, Matthew's correlation coefficient was planned to be evaluated, but this cannot be done as it involves division by zero in the majority of cases. Some performance metrics (accuracy, sensitivity, specificity and balanced accuracy) were evaluated based on confusion matrices. They are presented in Figure 14.

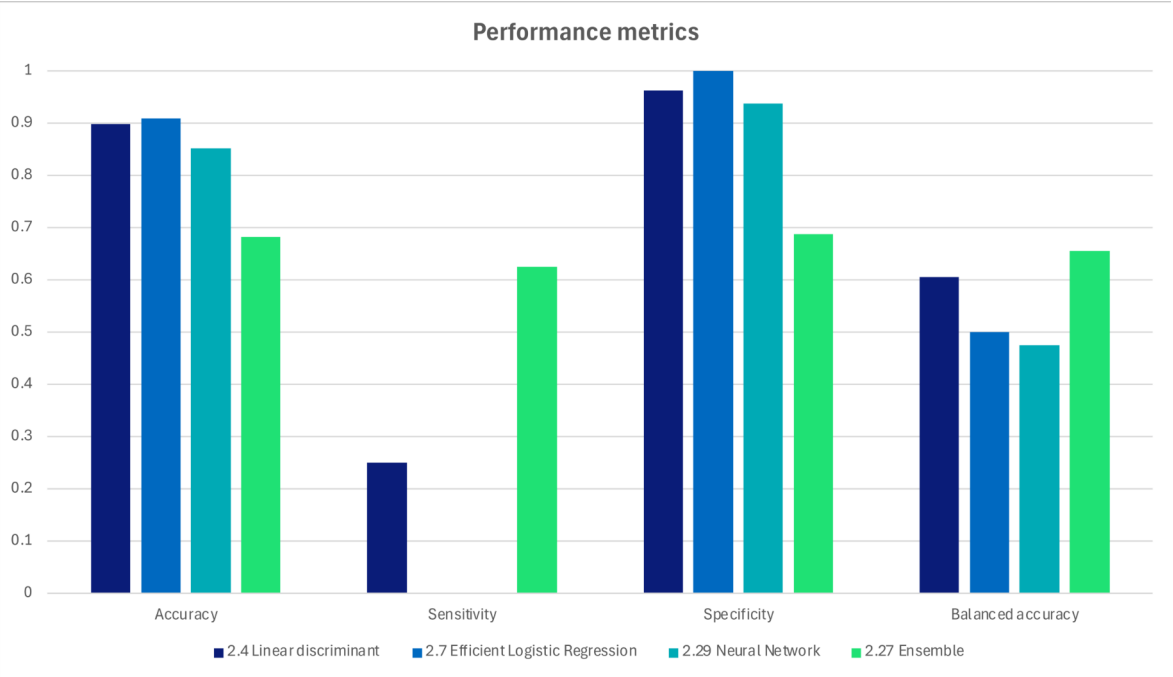


Figure 14. Performance metrics for different models.

The models with high accuracy show low balanced accuracy due to low sensitivity. Kappa is presented in Figure 15, both for validation and test data sets. For a couple of models, Kappa is positive, but even if the validation set gives a positive Kappa, the test set Kappa is negative.

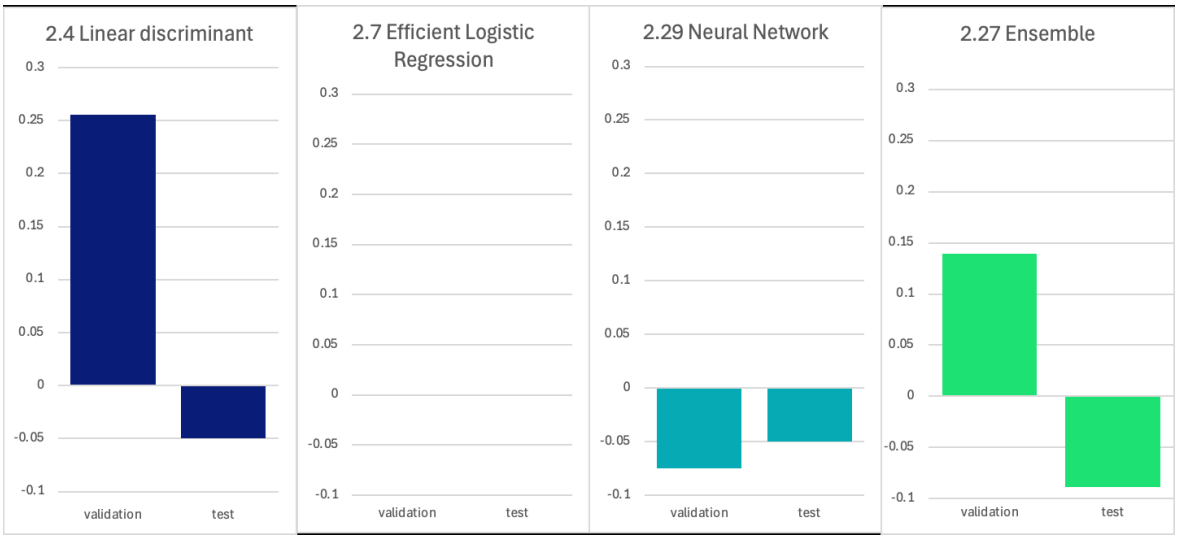


Figure 15. Kappa for validation and test sets for different models.

Unfortunately, all the models show low prediction performance; it is hard to detect the predictors of failure. Most likely, the models do not have enough data to be trained to detect failed project grades.

### 3.3.4 Prediction of project pass based on interim results

Another interesting question for discussion is whether it is possible to predict the passing of the project based on interim results. The idea was similar to the prediction based on final results. Several attempts were made: the first model was based only on homework performance, the second model considered questionnaire results, and the third included data from Moodle. For the first two models, the total scores for completing two, four and six homework assignments were used. For the third model, the total score for completing four exercise sets and the analysis of time-log data were used.

The initial analysis of homework and project grades was done in Excel: homework points for the required number of sets were calculated and converted into completion share based on the maximum score. Based on the data, a correlation matrix and a heatmap were created (Figure 16). The homework cumulative scores show a high correlation, which was expected. But they do not show a correlation with the project completion.

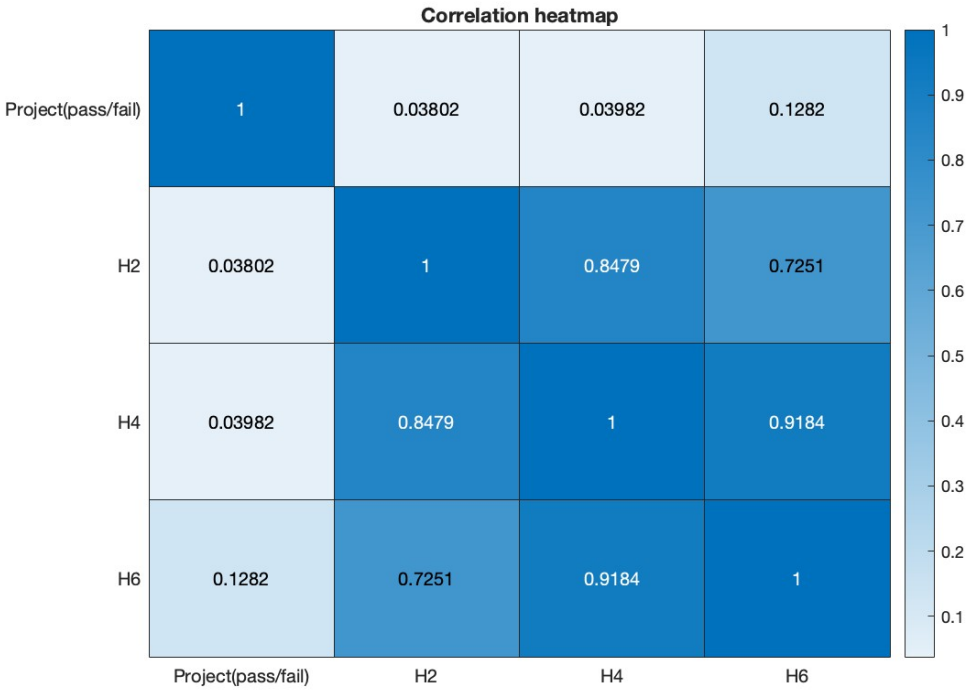


Figure 16. Correlation heatmap. H2 stands for the cumulative score for completing two homework assignments, H4 – four, and H6 – six.

The second correlation matrix includes the cumulative score for completing four homework assignments (H4) and median time, number of logs and number of grading for these four exercise sets (they were calculated as described earlier in 3.2.1). The correlation heatmap (Figure 17) shows the high correlation between logs and grading attempts, which is easy to interpret given that increased student activity in the system naturally leads to more grading events. Also, there is a moderate correlation between homework grade and logs and grading attempts. But no correlation with the project completion was observed.

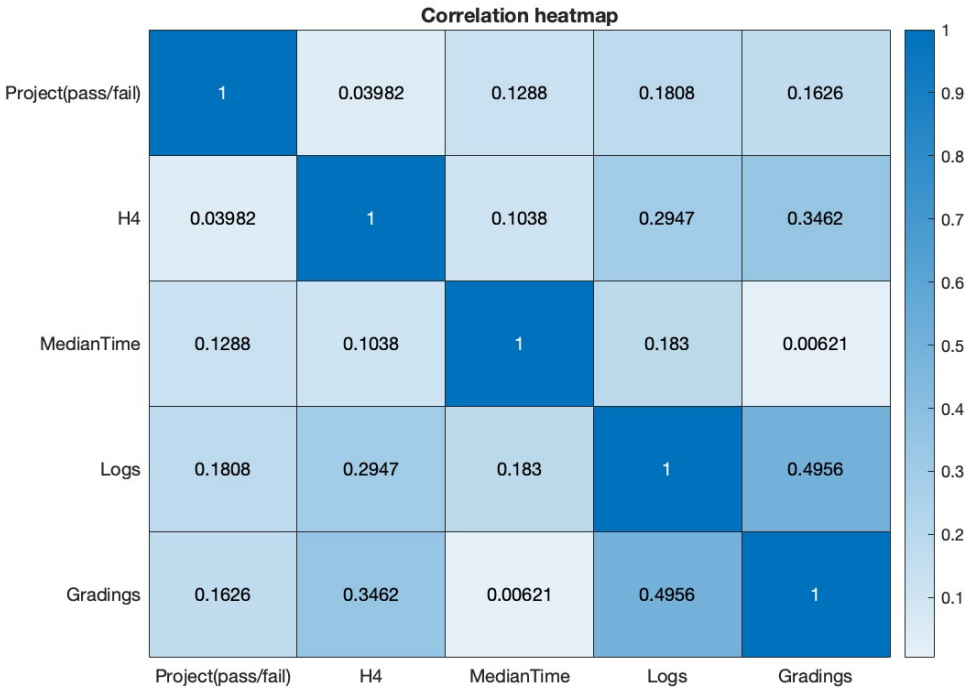


Figure 17. Correlation heatmap. H4 - the cumulative score for completing four homework assignments.

Despite the negative results obtained from the correlation matrices, an attempt was made to build the prediction models. The response for all the models was the project completion (pass/fail). The predictors used in the first model were the cumulative scores for completing the homework assignments (H2, H4, H6). The questionnaire results were added for the second model. For the third model, just one cumulative score for completing the homework assignment (H4), median time, logs and grading data together with questionnaire results were used. The accuracy of the models was analysed as described earlier (3.2.3), and the results are presented in Figure 18.

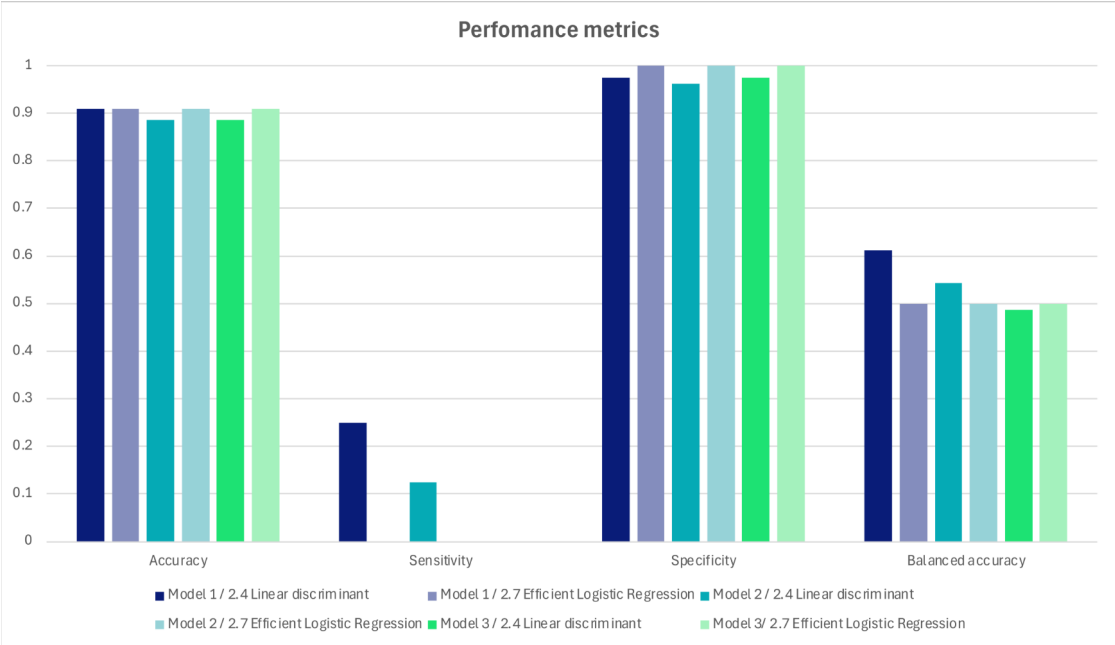


Figure 18. Performance metrics for different models.

The two models for each set of predictors were used: 2.4 Linear discriminant and 2.7 Efficient Logistic Regression. Both models show high accuracy, but due to the low sensitivity, the balanced accuracy is low. Kappa was calculated: for Efficient Logistic Regression models, it is 0 for all sets of predictors; the results for 2.4 Linear discriminant model are presented in Figure 19. Same as discussed earlier, the models are not accurate: even if the validation set gives a positive Kappa, the test set Kappa is negative or close to zero.

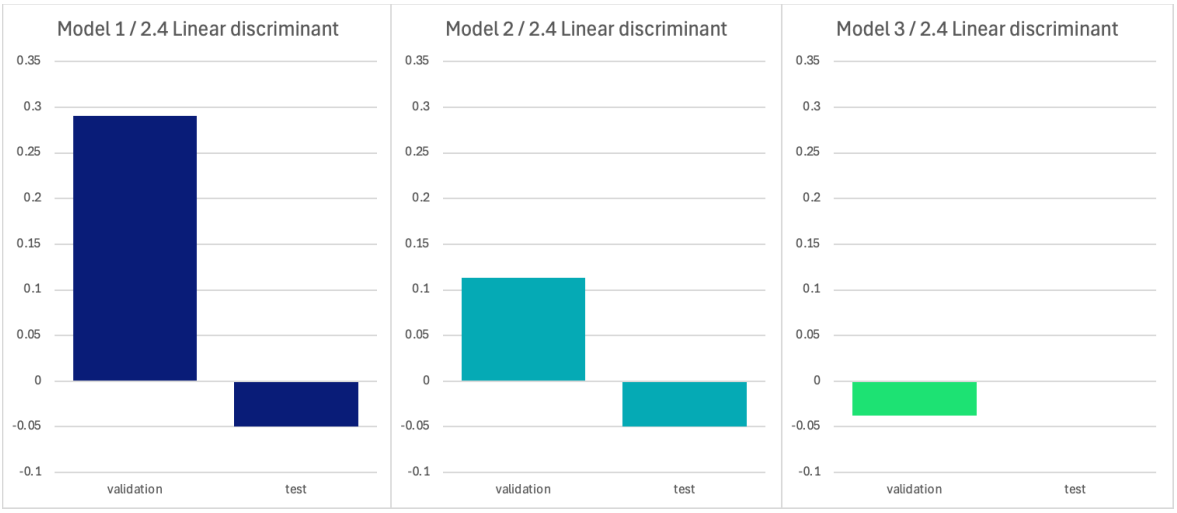


Figure 19. Kappa for validation and test sets for different models.

Similar to the previously discussed models based on final results, these models are unable to predict minor outcomes (failing the project).

## 3.4 Research summary

This study explored how learning analytics data can be utilised to monitor and improve student performance, focusing on how completing homework during the semester relates to success in the final project. Also, questionnaire results provided extra data characterising prior experience and study habits.

The results showed that students with higher homework scores and who were active (with more logins and grading attempts) usually performed better in the final project. However, there were exceptions, and no strong pattern was found that could clearly predict who would succeed or fail.

Different machine learning models were built to predict student success in the project based on various data, but none of them worked well. This shows that student performance is affected by many factors, and it is difficult to predict outcomes based only on the collected data.

# 4 Discussion

The goal of this study was to find out if learning analytics, along with information about student background and study habits, could be used to track and improve their performance. The main question was whether student activity during the course (doing homework, logging into Moodle and submitting homework assignments) could help predict how they would perform in the final project.

The results showed some clear patterns. The trends predicting success in the final project were completing the homework with high scores, accessing the course page and performing various actions on it a high number of times (more than 800) and returning homework in advance. Even though some trends were clear, it was not enough to build a reliable model to predict who would fail. There are a few possible reasons explaining this. First, the study only used data from one semester and one course, so the number of students was small. Second, all students finished the course: there were no dropouts, so the data did not give information on how performance changes when students lose motivation or stop participating. Third, the homework was checked automatically with MATLAB Grader, the tool provided instant evaluation and allowed the assignment submission multiple times (most of the students tried until the maximum points were received). For the project assignment, there was only one attempt. Fourth, most of the students who failed did so because of problems with the final project, such as plagiarism or not following the project instructions. And the most important reason is that people's behaviour is hard to predict.

To improve future research, it would be helpful to include more students from several semesters and to collect data at the beginning of the course instead of just at the end. It might also help to combine Moodle data with other information, such as participation in lectures and exercise sessions and student feedback on the course, to find the reason for failings.

This study was motivated by a desire to improve student learning outcomes and reduce course failures. Unfortunately, the study did not lead to a working prediction model, but it shows that learning analytics can provide interesting insights into students' engagement during the course and success at the final.

## References

1Edtech. LTI. Retrieved on 22 January 2025. Available at https://www.1edtech.org/standards/lti

Akosa, J.S. 2017. Predictive Accuracy: A Misleading Performance Measure for Highly Imbalanced Data. SAS. Available at https://support.sas.com/resources/papers/proceedings17/0942-2017.pdf

Andrews.edu. Correlation Coefficients. Retrieved on 23 April 2025. Available at https://www.andrews.edu/~calkins/math/edrm611/edrm05.htm

Barreto D., Rottmann A., Rabidoux S. Learning Management Systems. Choosing the Right Path For Your Organization. Retrieved on 22 January 2025. Available at https://edtechbooks.s3.us-west-2.amazonaws.com/pdfs/147/_147.pdf

Cannaos, C. & Onni, G. & Casu, A. 2024. Blended Learning: What Changes? Sustainability 16(20), 8988.

Cavus N. & Zabadi T. 2014. A Comparison Of Open Source Learning Management Systems. Procedia - Social and Behavioral Sciences 143 (2014) 521-526

Chicco, D., Tötsch, N. & Jurman, G. 2021. The Matthews correlation coefficient (MCC) is more reliable than balanced accuracy, bookmaker informedness, and markedness in two-class confusion matrix evaluation. BioData Mining 14, 13

Digital Learning Institute. Learning Analytics: The Ultimate Guide. Retrieved on 4 February 2025. Available at https://www.digitallearninginstitute.com/blog/learning-analytics-the-ultimate-guide

Dziuban, C., Graham, C.R., Moskal, P.D. et al. 2018. Blended learning: the new normal and emerging technologies. Int J Educ Technol High Educ 15, 3

eLAB. Moodle. Retrieved on 22 January 2025. Available at https://elab.lab.fi/en/it-instructions-and-study-tools/study-systems/moodle

eLUT. Moodle. Retrieved on 22 January 2025. Available at https://elut.lut.fi/en/it-instructions-and-study-tools/study-systems/moodle

Fernandez, A., Saviz, C., & Burmeister, J. 2006. Homework As An Outcome Assessment: Relationships Between Homework And Test Performance. Proceedings of the ASEE Annual Conference & Exposition, 18–21 June 2006, Chicago, Illinois.

Gamage, S.H.P.W., Ayres, J.R. & Behrend, M.B. 2022. A systematic review on trends in using Moodle for teaching and learning. IJ STEM Ed 9, 9

Gherheș, V., Stoian, C.E., Fărcașiu, M.A., Stanici, M. 2021. E-Learning vs. Face-To-Face Learning: Analyzing Students' Preferences and Behaviors. Sustainability 2021, 13, 4381

Graham, C. R. 2013. Emerging practice and research in blended learning. In M. G. Moore (Ed.), Handbook of distance education. 3rd ed. New York: Routledge, 333–350

Gryshuk, R. 2025. EducateMe. What is an LMS? Brief Overview. Retrieved on 29 March 2025. Available at https://www.educate-me.co/blog/what-is-lms

Hooshyar, D., Pedaste, M., & Yang, Y. 2020. Mining Educational Data to Predict Students' Performance through Procrastination Behavior. Entropy, 22(1), 12

IBM 2021. What is machine learning? Retrieved on 10 April 2025. Available at https://www.ibm.com/think/topics/machine-learning

IBM 2023. What are support vector machines (SVMs)? Retrieved on 10 April 2025. Available at https://www.ibm.com/think/topics/support-vector-machine

IBM 2024. What is data mining? Retrieved on 4 February 2025. Available at https://www.ibm.com/think/topics/data-mining

IBM What is the k-nearest neighbors (KNN) algorithm? Retrieved on 10 April 2025. Available at https://www.ibm.com/think/topics/knn

LAB a. Thesis Bachelor's Degree. Retrieved on 30 November 2024. Available at https://elab.lab.fi/en/completing-studies/theses/thesis-bachelors-degree

Li, X., Wang, T. and Wang, H. 2017. Exploring N-gram features in clickstream data for MOOC learning achievement prediction. Proc. DASFAA, Suzhou, China, pp. 328–339

MathWorks a. MATLAB Grader. Retrieved on 21 November 2024. Available at https://se.mathworks.com/products/matlab-grader.html

MathWorks b. Classification Learner. Retrieved on 10 April 2024. Available at https://se.mathworks.com/help/stats/classificationlearner-app.html

Mathworks c. Machine Learning in MATLAB. Retrieved on 10 April 2024. Available at https://se.mathworks.com/help/stats/machine-learning-in-matlab.html

MathWorks d. Select Data for Classification or Open Saved App Session. Retrieved on 10 April 2024. Available at https://se.mathworks.com/help/stats/select-data-and-validation-for-classification-problem.html

MathWorks e. corrcoef. Retrieved on 10 April 2024. Available at https://se.mathworks.com/help/matlab/ref/corrcoef.html

Moodle a. Learning Analytics For Moodle. Retrieved on 6 November 2024. Available at https://moodle.com/functionality-with-moodle/learning-analytics-for-moodle/

Moodle b. Tracking process. Retrieved on 5 December 2024. Available at https://docs.moodle.org/405/en/Tracking_progress

Moodle c. The Moodle Story. Retrieved on 22 January 2025. Available at https://moodle.com/about/the-moodle-story/

Moodle d. News: Thanks to educators around the world, we've reached 200 million education resources on Moodle sites! Retrieved on 22 January 2025. Available at https://moodle.com/news/200-million-education-resources-on-moodle-sites/

Moodle e. News: What is LTI and how it can improve your learning ecosystem. Retrieved on 22 January 2025. Available at https://moodle.com/news/what-is-lti-and-how-it-can-improve-your-learning-ecosystem/

Moodle f. Analytics. Retrieved on 31 January 2025. Available at https://docs.moodle.org/405/en/Analytics

Moreno-Marcos, P. M., Pong, T.-C., Muñoz-Merino, P. J. and Delgado Kloos, C. 2020. Analysis of the Factors Influencing Learners' Performance Prediction With Learning Analytics. IEEE Access, vol. 8, pp. 5264-5282

Murthy, S., Abu Bakar, A., Abdul Rahim, F., Ramli, R. 2019. A Comparative Study of Data Anonymization Techniques. In: Proceedings of the IEEE 5th International Conference on Big Data Security on Cloud (BigDataSecurity), High Performance and Smart Computing (HPSC), and Intelligent Data and Security (IDS), 27–29 May 2019, Washington, DC, USA. IEEE, 306–309

Nachouki, M., Mohamed. E.A., Mehdi, R., Naaj, M.A. 2023. Student course grade prediction using the random forest algorithm: Analysis of predictors' importance. Trends in Neuroscience and Education, Volume 33

Namoun, A., & Alshanqiti, A. 2021. Predicting Student Performance Using Data Mining and Learning Analytics Techniques: A Systematic Literature Review. Applied Sciences, 11(1), 237

Nguyen, A., Gardner, L., & Sheridan, D. 2020. Data Analytics in Higher Education: An Integrated View. Journal of Information Systems Education, 31(1), 61-71

Nortvig, A. M., Petersen, A. K., & Balle, S. H., 2018. A Literature Review of the Factors Influencing E- Learning and Blended Learning in Relation to Learning Outcome, Student Satisfaction and Engagement. The Electronic Journal of e-Learning, 16(1), pp. 46-55.

Pérez-Lemonche, Á., Martínez-Muñoz, G. & Pulido-Cañabate, E. 2017. Analysing event transitions to discover student roles and predict grades in MOOCs Proc. ICANN, pp. 224–232.

Pigeau, A., Aubert, O. & Prié, Y. 2019. Success prediction in MOOCs: a case study. Proc. EDM, Montreal, QC, Canada, pp. 390-395

Qazdar A., Qassimi S., Hassidi O., Hafidi M., Abdelwahed E.H., Melk Y. 2024 Learning Analytics for Tracking Student Progress in LMS. Research Square Preprint. Available at https://doi.org/10.21203/rs.3.rs-1505417/v1

Rotelli, D. & Monreale, A. 2023. Processing and Understanding Moodle Log Data and Their Temporal Dimension. Journal of Learning Analytics, 10(2), 126-141. https://doi.org/10.18608/jla.2023.7867

Sanchez, L., Penarreta, J. & Soria Poma, X. 2024 Learning management systems for higher education: a brief comparison. Discov Educ 3, 58 (2024). https://doi.org/10.1007/s44217-024-00143-5

SOLAR. What is Learning analytics? Retrieved on 4 February 2025. Available at https://www.solaresearch.org/about/what-is-learning-analytics/

Sulaiman, A. & Mohezar, S. 2006 Student Success Factors: Identifying Key Predictors. Journal of Education for Business, vol. 81, no. 6, pp. 328-333

Sweeney, M., Lester, J. & Rangwala, H. 2015. Next-term student grade prediction IEEE International Conference on Big Data (Big Data), Santa Clara, CA, USA, pp. 970-975

Tachepun, C. & Thammaboosadee, S. 2020. A Data Masking Guideline for Optimizing Insights and Privacy Under GDPR Compliance. Proceedings of the 11th International Conference on Advances in Information Technology, 1–9. https://doi.org/10.1145/3406601.3406627

Vanthienen, J. & De Witte, K. 2018. Data Analytics Applications in Education. Abingdon: Taylor & Francis Group.

## Appendices

Appendix 1. Questionnaire form.

We are asking for consent to collect information for a better understanding of the learning experience during this course. The data collected will be used for course development purposes and research purposes, potentially including publication in thesis and scientific papers.

The gathered data includes grades and time logs, as well as asking a few questions about your background and study habits. All the data presented will be anonymized to ensure your privacy.

Participation is entirely voluntary, and you can request to withdraw your data at any time without any consequences. All information will be handled confidentially and used solely for the purposes mentioned above.

Please take a moment to answer the following questions. Thank you in advance for your valuable input!

1. Do you allow us to use your data?
    - Yes
    - No
2. Do you have any prior programming experience?
    - No experience
    - Yes, I have some experience
    - Yes, I have a strong background in programming
3. What is your highest level of education completed?
    - Bachelor’s degree
    - Master’s degree
    - PhD
4. What is your current working status?
    - Not working

- Working part-time
- Working full-time
- Prefer not to say

5. Did you move at the beginning of the semester?
    - No
    - Yes, I moved from abroad
    - Yes, I moved from another city
    - Prefer not to say
6. What is your preferred study mode in this course?
    - Contact learning (in-person classes)
    - Remote learning (online classes)
    - Self-paced learning (no teaching contact)
7. How do you usually complete your assignments?
    - By myself
    - With other students

Appendix 2. MATLAB code (logs and grading events).

```
A = readtable ("allLogs.xlsx");
%total logs
table1 = groupcounts(A, 'UserFullName');
table1.Properties.VariableNames{'GroupCount'}='TotalLogs';
%total gradings
table2 = groupcounts(A, 'AffectedUser');
table2.Properties.VariableNames{'AffectedUser'}='UserFullName';
table2.Properties.VariableNames{'GroupCount'}='TotalGrades';

table = innerjoin(table1,table2, 'Keys', "UserFullName");
table= table(:,["UserFullName", "TotalLogs", "TotalGrades"]);

A.Time = datetime(A.Time, 'InputFormat', 'dd/MM/yy, HH:mm:ss');
cutoffDate =datetime(2024, 10, 7);
A2 =  A(A.Time < cutoffDate, :);
% logs H4
table3 = groupcounts(A2, 'UserFullName');
table3.Properties.VariableNames{'GroupCount'}='TotalLogsH4';
% gradings H4
table4 = groupcounts(A2, 'AffectedUser');
table4.Properties.VariableNames{'AffectedUser'}='UserFullName';
table4.Properties.VariableNames{'GroupCount'}='TotalGradesH4';

tableH4 = innerjoin(table3,table4, 'Keys', "UserFullName");
tableH4= tableH4(:,["UserFullName", "TotalLogsH4", "TotalGradesH4"]);

%save
table= innerjoin(table,tableH4, 'Keys', "UserFullName");
save("allLogsCount.mat", 'table');
```

Appendix 3. MATLAB code (times).

```
% getIDs
A = readtable ("logs"+number+".xlsx");
% leave only active actions
A.Component = string(A.Component);
AOpen = A(A.Component == "External tool",:);
% take learner ID
AOpen.LearnerID = extractBetween(AOpen.Description, 'The user with id
&#039;', '&#039; viewed the &#039;lti&#039; activity with course module
');

data = AOpen(:, ["UserFullName", "LearnerID"]);
table = unique(data, 'rows');
save("ids.mat", "table");

% times
number="";
for i=1:11
    if any(i==[7,10,11])
        j_max=4;
    else j_max=5;
    end
    for j=1:j_max
        number=char(i+"_"+j);
        number

A = readtable ("logs"+number+".xlsx");
%leave only active actions
A.Component = string(A.Component);
AOpen = A(A.Component == "External tool",:);
%take learner ID
AOpen.LearnerID = extractBetween(AOpen.Description, 'The user with id
&#039;', '&#039; viewed the &#039;lti&#039; activity with course module
');

AOpen = AOpen(:, ["Time", "UserFullName", "LearnerID", "Description"]);
%date and time to time format
AOpen.Time = string(AOpen.Time);
AOpen.Time = datetime(AOpen.Time, 'InputFormat', 'dd/MM/yy, HH:mm:ss',
'Timezone', 'UTC');

%earliest date = the date when student opened the assignment
T_min = groupsummary(AOpen, "LearnerID", "min", "Time");

AGrade = A(A.Component == "System",:);
valueToRemove="Juho Ratava";
AGrade(AGrade.UserFullName == valueToRemove,:)=[];
AGrade.LearnerID = extractBetween(AGrade.Description, 'for the user with
id &#039;', '&#039;.');
AGrade = AGrade(:, ["Time", "UserFullName", "LearnerID"]);
AGrade.Time = datetime(AGrade.Time, 'InputFormat', 'dd/MM/yy, HH:mm:ss',
'Timezone', 'UTC');

%when sumbit the best result
[AGrade.DayNumber,AGrade.DayName] = weekday(AGrade.Time);
Weekday = AGrade(:,["LearnerID", "DayNumber"]);
d=char("day"+number);
Weekday.Properties.VariableNames{'DayNumber'} = d;
```

```
WeekdayName = AGrade(:,["LearnerID", "DayName"]);
dd=char("dayName"+number);
WeekdayName.Properties.VariableNames{'DayName'} = dd;

T_max = groupsummary(AGrade, "LearnerID", "max", "Time");

C = innerjoin(T_max , T_min,'Keys', 'LearnerID');
C.delta = C.max_Time-C.min_Time;
C= C(:, ["LearnerID", "delta"]);
C2=C;
t=char("time"+number);
C.Properties.VariableNames{'delta'} = t;

if number=="1_1"
    T= load('ids.mat');
else T =load('timesv2.mat');
end
T_existing = T.table;
%T_existing.LearnerID = string(T_existing.LearnerID);
%timetable.LearnerID = string(timetable.LearnerID);
T_updated = outerjoin(T_existing,C, "Keys","LearnerID", 'MergeKeys',
true);
table= T_updated;
save("timesv2.mat", 'table');

if number=="1_1"
    D= load('ids.mat');
else D =load('daysv2.mat');
end
D_existing = D.table;
%T_existing.LearnerID = string(T_existing.LearnerID);
%timetable.LearnerID = string(timetable.LearnerID);
D_updated = outerjoin(D_existing,Weekday, "Keys","LearnerID",
'MergeKeys', true);
table= D_updated;
save("daysv2.mat", 'table');

if number=="1_1"
    DD= load('ids.mat');
else DD =load('daysnamesv2.mat');
end
DD_existing = DD.table;
%T_existing.LearnerID = string(T_existing.LearnerID);
%timetable.LearnerID = string(timetable.LearnerID);
DD_updated = outerjoin(DD_existing,WeekdayName, "Keys","LearnerID",
'MergeKeys', true);
table= DD_updated;
save("daysnamesv2.mat", 'table');

    end
end

% mode, avg, median, min, max

DD_updated;
DD_updated.MostPopularWeekday = mode(DD_updated{:, 3:end}, 2);

D_updated.meanDay = mean(D_updated{:, 3:end},2,'omitnan');
D_updated.medianDay = median(D_updated{:, 3:(end-1)},2,'omitnan');
```

```
D_updated.MostPopularWeekday = mode(D_updated{:, 3:(end-2)}, 2);
Dtransfer = D_updated(:, ["LearnerID", "meanDay","medianDay", "MostPopu-
larWeekday"]);

T_updated.avgTime = mean(T_updated{:, 3:end}, 2, 'omitnan');
T_updated.minTime = min(T_updated{:, 3:(end-1)}, [], 2, 'omitnan');
T_updated.maxTime = max(T_updated{:, 3:(end-2)}, [], 2, 'omitnan');
T_updated.medianTime = median(T_updated{:, 3:(end-3)},  2, 'omitnan');

T_updated = outerjoin(T_updated,Dtransfer, "Keys","LearnerID",
'MergeKeys', true);
T =load('timesv2.mat');
save("timesv2.mat", 'T_updated');


% same code was used for intermediate times; the values were saved in
timesv2H4.mat
```

Appendix 4. MATLAB code (merging tables and visualisation).

```
%too many tables, but I follow the rule “it works, do not touch it”
%reading tables from Moodle
Q = readtable ('Questionnaire.xlsx');
G = readtable('FinalGrades.xlsx');
% removing dublicated answers for Qyes-table
Q1=sortrows(Q, "Response", "descend");
[~, idx] = unique(Q1.ID, "stable");
Q1 = Q1(idx, :);
% name and surmane are in different columns, full name column created
G.FullName = G.Name + " " +G.Surname;
Qyes= Q1(Q1.Q00_contest == "1 : Yes", :); % only those who allow to use
their data
Qyes(:,["Response","SubmittedOn_","Username", "Institution", "Department",
"Course", "Group"] )=[]; % extra data removed
Qyes.Properties.VariableNames{'ID'} = 'LearnerID';
Qyes(Qyes.FullName=="Mariia Sobinina",:)=[];
A= load('timesv2.mat');
A=A.T_updated;
DataTimes = innerjoin(Qyes, A,"Keys","LearnerID");
DataTimes(:, "UserFullName")=[];
DataTimesShort = DataTimes(:, ["LearnerID", "FullName", "avgTime",
"minTime", "maxTime", "medianTime", "medianDay", "meanDay", "MostPopular-
Weekday"]);
%same for 4 sets
A4= load('timesv2H4.mat');
A4=A4.T_updated;
DataTimesH4 = innerjoin(Qyes, A4,"Keys","LearnerID");
DataTimesH4(:, "UserFullName")=[];
DataTimesShortH4 = DataTimesH4(:, ["LearnerID", "medianTime"]);
DataTimesShortH4.Properties.VariableNames{'medianTime'} = 'medianTimeH4';
DataTimesShort = innerjoin(DataTimesShort, Data-
TimesShortH4,"Keys","LearnerID");
B=load('allLogsCount.mat');
B=B.table;
B.Properties.VariableNames{'UserFullName'}='FullName';
LogsSummary = innerjoin(Qyes, B,"Keys","FullName");
LogsvsGrades = innerjoin( LogsSummary, G,"Keys","FullName");
Data = innerjoin(G,Qyes, 'Keys', 'FullName'); % data joined. table includes
survey answers and grades
Data = innerjoin(Data, DataTimesShort, 'Keys', 'FullName');
LogsvsGradesvsTimes = innerjoin(Data,LogsvsGrades, 'Keys', "FullName");
LogsvsGradesvsTimes.Properties.VariableNames {'H2_LogsvsGrades'} = 'H2';
LogsvsGradesvsTimes.Properties.VariableNames {'H4_LogsvsGrades'} = 'H4';
LogsvsGradesvsTimes.Properties.VariableNames {'H6_LogsvsGrades'} = 'H6';
LvsGvsT = LogsvsGradesvsTimes(:, ["FullName", "LearnerID", "Homework_Data",
"Project_Data","FinalGrade_Data", "avgTime","minTime","maxTime", "medi-
anTime","meanDay","medianDay","MostPopularWeekday","medianTimeH4","Total-
Logs","TotalGrades","TotalLogsH4","TotalGradesH4", "H2","H4","H6"]);
LvsGvsT.medianTime =seconds(LvsGvsT.medianTime);
% scatter plots
a = LvsGvsT.MostPopularWeekday;
b= LvsGvsT.Homework_Data;
scatter(a,b, 'black', 'filled', 'MarkerEdgeColor','black');
hold on
idx = (LvsGvsT.FinalGrade_Data ==0);
scatter(a(idx), b(idx),'red', 'filled', 'MarkerEdgeColor','black')
title('Logs vs Submission day ');
legend('Passed', 'Failed');
```

```
xlabel('Submission day');
ylabel('Logs');
% weekdays for submission days
%xticks([1 2 3 4 5 6 7]); % Set the tick positions
%xticklabels({ 'Sun','Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat'});
%log scale for median time
%set(gca, 'XScale', 'log');
grid on
hold off
HvsPplusQ= Data(:,["FullName", "LearnerID", "Homework", "Project", "avg-
Time", "minTime", "maxTime", "medianTime","MostPopularWeekday", "Q00_educa-
tion", "Q00_programmingExperience", "Q00_workingStatus", "Q01_moving",
"Q02_studyMode", "Q03_groupwork"]);
%removed definitions of the answers
HvsPplusQ.Q00_education=
str2double(extractBefore(HvsPplusQ.Q00_education,2));
HvsPplusQ.Q00_programmingExperience= str2double(extractBefore(HvsP-
plusQ.Q00_programmingExperience,2));
HvsPplusQ.Q00_workingStatus= str2double(extractBefore(HvsPplusQ.Q00_work-
ingStatus,2));
HvsPplusQ.Q01_moving= str2double(extractBefore(HvsPplusQ.Q01_moving,2));
HvsPplusQ.Q02_studyMode=
str2double(extractBefore(HvsPplusQ.Q02_studyMode,2));
HvsPplusQ.Q03_groupwork=
str2double(extractBefore(HvsPplusQ.Q03_groupwork,2));
% histograms
C = categorical(HvsPplusQ.Q00_education,[1 2 3],{'BSc','MSc','PhD'});
h = histogram(C,'BarWidth',0.5, 'FaceColor','black');
title('Prior education');
C = categorical(HvsPplusQ.Q00_programmingExperience,[1 2 3],{'No experi-
ence','Yes, some experience','Yes, strong background'});
h = histogram(C,'BarWidth',0.5, 'FaceColor','black');
title('Programming experience');
C = categorical(HvsPplusQ.Q00_workingStatus,[1 2 3 4],{'No','Part-
time','Full-time', 'No answer'});
h = histogram(C,'BarWidth',0.5, 'FaceColor','black');
title('Working status');
C = categorical(HvsPplusQ.Q01_moving,[1 2 3 4],{'No','Yes, abroad','Yes,
city', 'No answer'});
h = histogram(C,'BarWidth',0.5, 'FaceColor','black');
title('Moving');
C = categorical(HvsPplusQ.Q02_studyMode,[1 2 3],{'Contact','Remote','Self-
paced'});
h = histogram(C,'BarWidth',0.5, 'FaceColor','black');
title('Study mode');
C = categorical(HvsPplusQ.Q03_groupwork,[1 2],{'By myself','In group'});
h = histogram(C,'BarWidth',0.5, 'FaceColor','black');
title('Groupwork');
% saving data
HvsPplusQ= Data(:,["FullName", "LearnerID", "H2","H4","H6","Homework",
"Project", "avgTime","medianTime", "minTime", "maxTime", "MostPopularWeek-
day", "medianTimeH4","Q00_education", "Q00_programmingExperience", "Q00_-
workingStatus", "Q01_moving", "Q02_studyMode", "Q03_groupwork"]);
LogsvsGrades = LogsvsGrades(:,["FullName", "TotalLogs", "TotalGrades","To-
talLogsH4","TotalGradesH4"]);
HvsPplusQ1 = innerjoin(HvsPplusQ, LogsvsGrades, 'Keys', "FullName");
save ('dataForAnalysys.mat', 'HvsPplusQ1');
```

Appendix 5. MATLAB code (data normalisation and analysis).

```
T = load ('dataForAnalysys.mat');
T=T.HvsPplusQ1;
%T.deltaHomework = T.HomeworkMiddle -T.Homework;
T.Q00_education= str2double(extractBefore(T.Q00_education,2));
T.Q00_programmingExperience= str2double(extractBefore(T.Q00_programmingExpe-
rience,2));
T.Q00_workingStatus= str2double(extractBefore(T.Q00_workingStatus,2));
T.Q01_moving= str2double(extractBefore(T.Q01_moving,2));
T.Q02_studyMode= str2double(extractBefore(T.Q02_studyMode,2));
T.Q03_groupwork= str2double(extractBefore(T.Q03_groupwork,2));

T.avgTime =seconds(T.avgTime);
T.medianTime = seconds(T.medianTime);
T.minTime =seconds(T.minTime);
T.maxTime =seconds(T.maxTime);
T.medianTimeH4 = seconds(T.medianTimeH4);

tAvgMax= max(T.avgTime);
tMedianMax= max(T.medianTime);
tMinMax = max(T.minTime);
tMaxMax = max(T.maxTime);
tMedianH4Max = max(T.medianTimeH4);
nLogsMax = max(T.TotalLogs);
nLogsH4Max = max(T.TotalLogsH4);
nTotalGradesMax = max(T.TotalGrades);
nTotalGradesH4Max = max(T.TotalGradesH4);

T.avgTimeN = T.avgTime /tAvgMax;
T.medianTimeN = T.medianTime /tMedianMax;
T.minTimeN = T.minTime /tMinMax;
T.maxTimeN = T.maxTime /tMaxMax;
T.medianTimeH4N = T.medianTimeH4 /tMedianH4Max;
T.logsN = T.TotalLogs / nLogsMax;
T.logsH4N = T.TotalLogsH4 / nLogsH4Max;
T.gradingN = T.TotalGrades / nTotalGradesMax;
T.gradingH4N = T.TotalGradesH4/nTotalGradesH4Max;

% normalizing weekdays
% Tue (3) = 0, ... Mon(2) = 1 as deadline; step 0.167
%M = containers.Map({'1','2','3','4','5', '6','7'},[0.833 1 0 0.167 0.333
0.5 0.666]);
M = containers.Map([1 2 3 4 5 6 7],[0.833 1 0 0.167 0.333 0.5 0.666]);
T.submitDayN = arrayfun(@(x) M(x), T.MostPopularWeekday);
%normalizing questionnaire
%Q1 - education
% #1 (BSc) = 0, #2 (MSc) = 0.5, #3 (PhD) = 1
M = containers.Map([1 2 3],[0 0.5 1]);
T.Q1_Education = arrayfun(@(x) M(x), T.Q00_education);
%Q2 - programming experience
% #1 (no exp) = 0, #2 = 0.5, #3 = 1
M = containers.Map([1 2 3],[0 0.5 1]);
T.Q2_Exp = arrayfun(@(x) M(x), T.Q00_programmingExperience);
%Q3 - working status
% #1 (no) = 0, #2 (part-time)= 0.33, #3 (full-time) = 0.66, #4 (no answer) =
1
M = containers.Map([1 2 3 4],[0 0.33 0.66 1]);
T.Q3_Work = arrayfun(@(x) M(x), T.Q00_workingStatus);
%Q4 - moving
% #1 (no) = 0, #2 (abroad)= 0.66, #3 (city) = 0.33, #4 (no answer) = 1
```

```
M = containers.Map([1 2 3 4],[0 0.66 0.33 1]);
T.Q4_Moving = arrayfun(@(x) M(x), T.Q01_moving);
%Q5 - study mode
% #1 (contact) = 1, #2(online)= 0.5, #3(self-paced)= 0
M = containers.Map([1 2 3],[1 0.5 0]);
T.Q5_StudyMode = arrayfun(@(x) M(x), T.Q02_studyMode);
%Q6 - groupwork
% #1 (myself) = 0, #2(group)= 1
M = containers.Map([1 2],[0 1]);
T.Q6_Groupwork = arrayfun(@(x) M(x), T.Q03_groupwork);
T.ProjectN = double (T.Project>0.49);

% correlation matrix final
Data =T(:, ["Project","Homework","medianTimeN", "logsN", "gradingN","submit-
DayN"]);
Grade = T(:, "Project");

dataMatrix = table2array(Data);
corrMatrix = corr(dataMatrix);
varNames = {"Project","Homework","MedianTime", "LogsTotal","GradingsTo-
tal","submitDay"};
heatmap(corrMatrix, "Title", "Correlation matrix", "YDisplayLabels", var-
Names, "XDisplayLabels",varNames);

%correlation matrix intermediate v1

Data =T(:, ["ProjectN","H2", "H4","H6",]);
Grade = T(:, "Project");

dataMatrix = table2array(Data);
corrMatrix = corr(dataMatrix);
varNames = {"Project(pass/fail)","H2", "H4","H6" };
heatmap(corrMatrix, "Title", "Correlation heatmap", "YDisplayLabels",
varNames, "XDisplayLabels",varNames);

%correlation matrix intermediate v2

Data =T(:, ["ProjectN", "H4","medianTimeH4N", "logsH4N", "gradingH4N"]);
Grade = T(:, "Project");

dataMatrix = table2array(Data);
corrMatrix = corr(dataMatrix);
varNames = {"Project(pass/fail)", "H4","MedianTime", "Logs","Gradings" };
heatmap(corrMatrix, "Title", "Correlation heatmap", "YDisplayLabels",
varNames, "XDisplayLabels",varNames);


%data for analysis int
Data =T(:, ["ProjectN","H2", "H4","H6","Homework","medianTimeH4N",
"logsH4N","gradingH4N", "Q1_Education", "Q2_Exp","Q3_Work", "Q4_Moving",
"Q5_StudyMode", "Q6_Groupwork"]);

%data for analysis final
Data =T(:, ["ProjectN","Homework","medianTimeN", "logsN","gradingN","submit-
DayN", "Q1_Education", "Q2_Exp","Q3_Work", "Q4_Moving", "Q5_StudyMode",
"Q6_Groupwork"]);
```