Predicting student performance with Neural Networks

Leon Gerritsen
ANR: 637922
SNR: 2005340

Thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Data Science: Business and Governance,
at the School of Humanities
of Tilburg University

Thesis committee:

Rianne Conijn
Menno van Zaanen

Tilburg University
School of Humanities
Tilburg, The Netherlands
May 2017

Abstract

In recent years, Neural Networks have seen widespread and succesful implementations in a wide range of data mining applications, often surpassing other classifiers. This study aims to investigate if Neural Networks are a fitting classifier to predict student performance from Learning Management System data in the context of Educational Data Mining. The dataset used for this study is a Moodle log file containing log information about 4601 students over 17 undergraduate courses. To assess the applicability of Neural Networks, we compare their predictive performance against six other classifiers on this dataset. These classifiers are Naive Bayes, k-Nearest Neighbors, Decision Tree, Random Forest, Support Vector Machine and Logistic Regression and will be trained on data obtained during each course. The features used for training originate from LMS data obtained during the length of each course, and range from usage data like time spent on each course page, to grades obtained for course assignments and quizes. After training, the Neural Network outperforms all six classifiers in terms of accuracy and is on par with the best classifiers in terms of recall. We also assessed the effect course predictors have on predictive performance by leaving out the course identifiers in the data. This does not affect predictive performance of the classifiers. Furthermore, the Neural Network is trained on individual course data to assess difference in classification performance between courses. Results show that half of these course classifiers outperform generally trained classifiers. The importance of individual predictors used for classification was also investigated, with previously obtained grades contributing most to succesful predictions. We can conclude that Neural Networks outperform the six other algorithms tested on this dataset and could be succesfully used to predict student performance.

Contents

1 Introduction

This section serves to introduce the background and problem statement of this study as well as the formulated research questions.

1.1 Background

In recent years, the use of internet-based educational tools has grown rapidly (Jordan, 2014) as well as the research surrounding them (see Figure 1). These tools provide a clear advantage for students and teachers alike, with the ability to access and share course data from anywhere in the world, track student progress and provide rich educational content. These tools generate vast amounts of data obtained in a non-obtrusive manner that can give a better look into the way students learn and interact with course materials. The challenge is to put these data to good use to improve on the educational process.

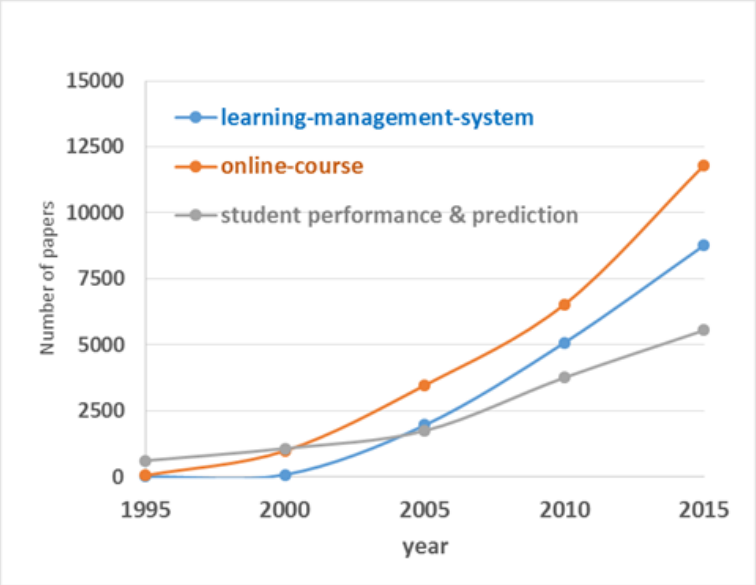

Figure 1: Number of papers in Educational Data Mining related fields. source: Google Scholar.

One of the purposes these data can be used for is the prediction of whether a student is going to pass or fail a course. Being able to predict student performance enables a teacher or educational institution to provide appropriate assistance to students that are at risk to miss the mark. Assisting them in a timely manner will reduce the number of students failing a course and may indirectly reduce the amount of students dropping out of their educational program. This is a societal interest that can have a positive impact on students, parents, teachers and educational institutions alike.

The analysis and extraction of information and patterns from vast amount of data is called Data Mining. When the data comes from an educational setting we are dealing with a subdomain of data mining called Educational Data Mining, or EDM. This is a field of research that applies data mining, statistics and machine learning to data derived from educational environments. It seeks to extract meaningful information from vast amounts of raw data that can be used to improve and understand learning processes (Scheuer & McLaren, 2012). In order to extract interesting information, like predicting if a student requires academic assistance, we can

make use of machine learning algorithms that can automatically predict this outcome based on the data. In the field of EDM, a wide set of machine learning algorithms have already been used to various degrees of success like Naive Bayes Classifiers, k-Nearest Neighbors, Random Forests, Decision Tree Classifiers, Support Vector Machine algorithms and Neural Networks (Romero & Ventura, 2010).

The family of classifiers this study focuses on, Neural networks, have shown promising results in domains like speech recognition (Graves & Jaitly, 2014), computer vision (Venugopalan et al., 2014), recognizing music (Costa, Oliveira, & Silla, 2017), playing complex games like GO (Wang et al., 2016) and economic forecasting (Nametala, Pimenta, Pereira, & Carrano, 2016), but their use in EDM has thus far been limited compared to other classification algorithms (Baker & Inventado, 2014). This can be partly explained by their difficulty to set up, the lack of convenient all in one packages that are easy to use and the often long training times (Gaur, 2012). But they do offer clear benefits over other machine learning algorithms. They are able to classify instances in domains that are not linearly-separable and can handle noisy and complex data (Schmidhuber, 2015). These properties make them especially suited for a domain like EDM where the data, given the fact that it is based on human behvaior, can be complex, might contain irrelevant entries as well as non linear relations. Assessing their applicability for the EDM domain by comparing their performance to that of other classifiers could therefore result in new insights.

Most EDM studies investigating student performance prediciton have used small samples with little diversity in the courses they analysed (Gašević, Dawson, Rogers, & Gasevic, 2016). This results in potentially low portability of the results due to the small sample size and differences that might exist between courses; a liberal arts course requires a different approach from a technical course. Furthermore, most studies use a wide variety of grades obtained in previous courses or previous academic curricula (Kovacic, 2012; Moucary, Khair, & Zakhem, 2011),where these previous grades have been shown to be strong predictors of future academic success (Richardson, Abraham, & Bond, 2012; Dollinger, Matyja, & Huber, 2008). But these grades might not always be available to use as predictors, thus limiting the predictive capacity of the algorithms devised in these studies.

The goal of the classifiers will be to predict if a student will require academic assistance, because he is at risk to fail the course, or does not require any assistance. As such, it can be cast as a binary classification problem, where the two prediction labels are "requires assistance" for students that are at risk to fail the course, and "does not require assistance" for students that are not at risk. The data used to perform this classification is extracted from the logfile of a Learning Management System (LMS) containing information about 4601 students over 17 courses. This allows us to compare the predictive performance between courses and assess if predictors identifying individual courses have an effect on performance. Additionaly, the effects of sample size and the importance of individual predictors will be investigated.

1.2 Research Questions

The aim of this study is to assess to what extent Neural Networks are suitable for student performance prediction from LMS data. With the data used for prediction consisting of 4601 students and 17 courses, which can be considered a large sample size. This performance asessment will be performed using two metrics: accuracy and recall. Accuracy is the percentage of correctly classified instances (does or does not require assistance), while recall gives and indication if the classifier let students requiring academic assistance slip through (False Negatives). In order to

determine their applicability, three research questions have been formulated.

The first research question aims to assess the accuracy and recall performance of Neural Networks, while using all available predictors, compared to that of a majority baseline and 6 other classification algorithms:

- Naive Bayes
- k-Nearest Neighbors
- Random Forest
- Decision Tree
- Logistic Regression
- Support Vector Machine

1: Is there a difference in accuracy and recall between Neural Networks and six other classifiers for student performance prediction?

Two different courses might require a very different studying approach, resulting in a very different LMS usage and thus a difference in LMS predictor data. The second research question will therefore seek to assess if differences between courses affect predictive performance by leaving out a predictor (Course ID) that indicates to what course an instance belongs to. The classifiers will be trained on all available course data but excluding the Course ID predictor. The accuracy and recall for each of the seven classifiers will then be compared among each other and to the results of the previous research question.

2: Does exclusion of course predictors influence classification accuracy and recall performance?

The third research question aims to assess the effect each course has on predictive performance by training the Neural Network on data from each individual course. The accuracy and recall for each course can then be compared to assess the difference between courses. And, because each course has different amounts of participants, varying from 59 to 1092, the effect of sample size can also be examined by comparing classification performance.

3: Is there a difference in accuracy and recall between training the Neural Network on all the course data at once versus training it on individual course data?

1.3 Outline

The thesis will have the following structure: Section 2 contains a literature study on related work. In section 3 the experimental setup is described. Section 4 presents the results and section 5 features the discussion of experimental results , recommendations for future research and the conclusions.

2 Background

The following section wil start with an overview of the current state of EDM. This will be followed by an analysis of Neural Networks, while the state of Neural Networks in student performance prediction is explained in subsection 2.3. Finally, subsection 2.4 explores the use of other classifiers in student performance prediction.

2.1 Educational Data Mining

Research performed in this study can be classified under EDM. EDM is a sub-group of data mining that focusses on researching, developing and applying various automated methods to explore large-scale data coming from educational settings. This is done to increase the understanding of the way students learn, study educational questions and improve the effectiveness of teaching and learning activities (Papamitsiou & Economides, 2014). This goal is achieved by transforming the raw data into information that can have a direct impact on educational practice and research (Romero & Ventura, 2010).

EDM is becoming increasingly widespread nowadays. The past couple of years has seen a rapid rise of the number of research papers dedicated to EDM in its various forms (Peña-Ayala, 2014). This has been linked to the increase of available educational data and the widespread availability of cheap computing power and accessible digital tools (Johnson & Samora, 2016). With such a wide availability of high-quality data and the potential to derive valuable educational insights, educational institutions, governments and researcher are increasingly looking for ways to put these techniques to good use.

The data analyzed in EDM come from various sources like Learning Management Systems, administrative data from universities and schools and other structured or unstructured databases pertaining to education. Due to the habitually large size of these databases, they require a computerized approach to discern the patterns and relationships they contain (Witten, Frank, Hall, & Pal, 2016). In this study, the data contains interaction records for 4601 students, making it too voluminous to derive useful insights by hand or through non-automated means. In this case, an EDM approach is recommendable to extract the information it contains.

In this study, the insights that are extracted from the data concern the prediction of student performance, which is a subdomain of EDM. This can be used to prevent students from failing courses by intervening in their educational process, predict a students potential to plan an optimal curriculum, give students insight in their learning process or develop more effective instruction techniques (Dietz-Uhler & Hurn, 2013). The focus of this study is the applicability of specific machine learning techniques in the prediction of whether a student does or does not require academic assistance for a certain course. Such knowledge can help to prevent students from dropping out of their courses or educational program (Campbell, Oblinger, et al., 2007).

2.2 Neural Networks

The classifier this study focuses on to predict student performance belongs to the family of Neural Networks. Neural Networks are algorithms that mimic the way our brain works. They consist of an array of interconnected nodes that exchange information among each other (see Figure 2), comparable to the way our neurons, connected by dendrites and axons, exchange information. They learn iteratively over time by observing different examples, similarly to how children can learn skills from their parents by observation. However, unlike children that

can learn recognize and object after only observing it once, Neural Networks often require a greater set of observations to attain sufficient predictive capacity, as they are notoriously data hungry (Cerny & Proximity, 2001; Karpathy et al., 2014). Neural Networks differ from other classification algorithms in the fact that internally, information is processed in a parallel way, comparable to how our brain functions. This differs from the serial processing that many other algorithms like Decision Tree classifiers use (Maren, Harston, & Pap, 2014).

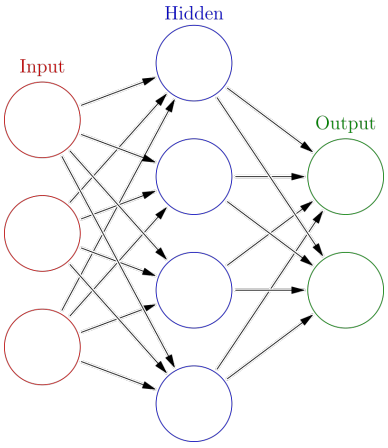

Figure 2: The structure of a Neural Network (Glosser.ca, 2013).

The property that makes Neural Networks interesting for complex domains like computer vision, playing complex games like GO or understanding human speech, is their ability to derive answers from complex and imprecise data (Srivastava, Hinton, Krizhevsky, Sutskever, & Salakhutdinov, 2014). Neural Networks are able to detect patterns that are too complicated for humans or other machine learning algorithms to pick up. And have been used successfully in numerous business applications for pattern recognition, prediction and classification (Gaur, 2012). These properties make them particularly suited for a domain like EDM where large quantities of data are available and where the data is often noisy and not linearly separable owing to the fact that it is based on human behavior.

Neural Networks also have certain disadvantages. It is at present impossible to derive how a network came to a certain output. Compared to a Decision Tree classifier in which one can follow a set of steps to arrive at a classification, a Neural Network doesnt store any explicit representation of the way it achieved its result. As such, a Neural Network is often referred to as a black box where information goes in, and a result comes out (Srivastava et al., 2014). They might therefore not be suited in some cases where the decision process needs to be explicit. This would be the case with expert system giving medical advice for which the decision process needs to be checkable. In our case, this black box nature is not a dealbreaker, as the consequence of mislabeling someone is not critical and will in most cases not require explicit explanation (Parasuraman, Sheridan, & Wickens, 2000). Another drawback, is that Neural Networks require vast amounts of data to achieve satisfactory performance and are therefore not suitable for every dataset (Cerny & Proximity, 2001; Karpathy et al., 2014). However, given the size of the dataset used in this study, with records of 4601 students, this will most likely not be an obstacle.

2.3 Neural Networks in Student Performance Prediction

Despite certain downsides, Neural Networks show promising results in EDM. They have been applied to student performance prediction, with various studies plotting them against other

machine learning algorithms. However, differences are observed between studies in terms of performance, used predictors, sample size, data transformation, number of distinct courses and number of labels that need to be predicted.

Sample size, by which we mean the number of students in a dataset, can influence the generalizability and portability of a model (Cerny & Proximity, 2001; Halevy, Norvig, & Pereira, 2009). Many EDM studies exist that make use of large samplesize to predict student performance, which use algorithms like Decision Trees, Bayesian Classifiers or Support Vector Machines. Good examples of which are a study by Jayaprakash et al. (2014) using a sample of 15150 students and another study with data from 10330 students (Kabakchieva, 2013). But studies involving Neural Networks have, as far as we have found, only used smaller sample sizes, with the maximum encountered being 649 students, in a study by Cortez et al (2008). Other studies we encountered relied on even smaller sample sizes for their predictions. Agrawal and Pandya(2015) used 100 students and Moucary et al.(2011) 73 students. Although the performance reported in these Neural Network studies are high, the portability of their findings is hard to determine. The network trained on 73 students might achieve good results for those students, but it might fare significantly worse if presented with new students that might come from another university or from another academic year. More students in a dataset should result in a more diverse sample and could therefore improve generalizability of the findings (Payne & Williams, 2005).

The second differentiating factor between studies is the amount of courses contained in their datasets. Courses can vary widely in length, difficulty and content. An art history course is vastly different from a course about linear algebra, and require very different study approaches (Ramsden & Entwistle, 1981). The history course might require more memorization, which could translate in more time spent reading slides in an LMS, while a Linear Algebra course might require more practical application, which could be done by performing more quizes and assignments. If we would use a model trained on the art history course data to predict performance for the linear algebra course students, the results might be dissapointing due to their differences. If one wants to use a trained model to predict performance on different courses, it would be wise to have trained that model on as many courses as possible to account for possible differences between theses courses. In the studies involving student performance prediction with Neural Networks we encountered, a majority used data coming from one course (Moucary et al., 2011; Agrawal & Pandya, 2015; Calvo-Flores, Galindo, Jiménez, & Piñeiro, 2006; Oladokun, Adebanjo, & Charles-Owaba, 2008) or two courses (Cortez & Silva, 2008).

Another factor affecting performance is the difference in predictors used for classification. Predictors being used in student performance prediction vary widely across studies, often encountered predictors are age (Oladokun et al., 2008; Kabakchieva, 2013) or assignment grades (Moucary et al., 2011; Romero, Ventura, & García, 2008). But also more uncommon predictors have been used. For example, whether a student is in a romantic relationship (Cortez & Silva, 2008) or the strictness of their parents (Agrawal & Pandya, 2015). Various predictors have different predictive qualities. Some correlate strongly to the student performance while others have only limited influence. Research has shown that the best predictor for student success is previously obtained grades, which can come from previous courses, assignments made during a course or from entry-tests (Richardson et al., 2012; Dollinger et al., 2008). If a student scored well on a test for a specific course, he will likely score well on future tests for that course. This predictor is used in all of the student performance prediction studies. However, previous grades might not always be available for various reasons. As an example we can take Massive Open Online Courses (MOOCs) where everyone can subscribe without having to enter previous academic achievements. Or previous grades are not available because it's the first course a stu-

dent is taking at a certain educational institution. Being able to perform accurate prediction, without knowing previous grades enables a more flexible usage of the predictive model.

2.4 Classifiers

In order to have a reference to compare the performance of the Neural Network against, we used six different classifiers to perform predictions on the data. These are k-Nearest Neighbors, Naive Bayes, Support Vector Machine, Logistic Regression, Decision Tree and Random Forests. These algorithms have all seen global use in EDM over the past years (Romero & Ventura, 2010).

The k-Nearest Neighbors (k-NN) algorithm looks at what known instances are close to the instance we want to predict to perform classification. To perform optimally, the k parameter (number of neighbors) needs to be tuned. The larger the value of k gets, the less influence noise will have on the classification, but the decision boundaries between different classes will become less separable (Everitt, Landau, Leese, & Stahl, 2011). In the context of student performance prediction, the algorithm obtained an accuracy of 57% on a dataset of 10330 students (Kabakchieva, 2013) predicting five different performance labels (bad, average, good, very good and excellent) and an accuracy of 62.9% on a dataset of 566 students (Stapel, Zheng, & Pinkwart, 2016) where it predicted a pass or fail .

The Naive Bayes classifier is a simple probabilistic classification algorithm which main advantages are its speed, simplicity and versatility (Kabakchieva, 2013). It is popular as a baseline algorithm (Gupte, Joshi, Gadgul, Kadam, & Gupte, 2014). One of the caveats of this algorithm is that it assumes that all predictor variables are independent, which might not always be the case.

Support Vector Machines classify data by constructing a hyperplane in high-dimensional space that separates the classes (Meyer, 2015). Using a special technique SVMs can also achieve non-linear classification (Murty & Raghava, 2016). They have shown robust practical performance in a broad variety of domains like image recognition, text mining and bioinformatics (Saunders, Stitson, Weston, Bottou, & Smola, 1998). They have obtained accuracies of 86.3% on a sample of 395 students (Cortez & Silva, 2008) where five labels had to be predicted, and 86.26% on a sample of 15150 students (Jayaprakash et al., 2014) where a pass or fail was predicted.

Logistic regression is a regression model that has a categorical variable as output. Although not frequently mentioned in the EDM studies we encounterd, with one occurence in Stapel et al (2016) where it obtained an accuracy of 68.2%, it is widely used in other data mining domains (Witten et al., 2016) and is therefore included in this study.

Decision Tree classifiers are a predictive modelling approach that uses a tree like structure for its representation. Decision Trees are often used to identify the most optimal strategies to reach a certain target in real-world settings because their output is easily transformable in step by step directions (Baker & Inventado, 2014). This property, combined with their fast training time (Rokach & Maimon, 2014) explains their widespread use in EDM. Jayaprakash et al. (2014) obtained a prediction accuracy of 85.92% with the Decision Tree, on a dataset of 15150 students by predicting a pass or fail. Kabackhieva (2013) only managed an accuracy of 63.1% on a dataset of 10330 students predicting five levels of student performance. The difference in performance can be explained by the difference in setup and purpose of the two studies. The study by Kabakhieva (2013) seeks to devise a model that predicts academic performance, which serves as a tool to determine if someone should be admitted to a specific academic program. As

such, it only uses historic data like grades and statistics obtained during the student's previous educational program, and more general statistics like age and gender. Additionally, it seeks to predict five different labels, ranging from bad to excellent performance, which can have an impact on the predictive accuracy. The study by Jayaprakash et al. (2014) seeks to devise a model that predicts whether or not a student will pass a course, and serves as an early alert system. They also use predictors from previous educational programs, and combine them with predictors gathered from the LMS course data. A variation on the Decision Tree classifier is the Random Forest which is an ensemble technique that uses a collection of Decision Trees to obtain a prediction.

The accuracy and number of classes that need to be predicted vary widely between studies. A recapitulation of results from studies involving binary classification, which have to predict two labels (for example: pass or fail, or requires assistance, or doesn't require assistance) can be found in Table 1. The variability in accuracies among studies could be attributed to the diversity and size of the data and the quality and type of the predictors that were used. Some studies use predictors that were gathered only during previous educational programs, like previously obtained grades (with different variations) or if a student has repeated a class. Other studies use these predictors suplemented with data gathered during a course, like time spent online or number of documents opened.

Table 1: Classification accuracies from studies predicting 2 labels (pass or fail)

	Sample	NN	SVM	DT	RF	NB	k-NN	LR
Jayaprakash, et al. (2014)	15150	-	86.3%	85.9%	-	84.1%	-	-
Stapel, et al. (2016)	566	-	-	71.5%	67.9%	65.4%	62.9%	68.2%
Agrawal, et al. (2015)	100	97.0%	-	91.0%	96.0%	80.0%	-	-
Calvo, et al. (2006)	240	80.2%	-	-	-	-	-	-

NN: Neural Network, SVM: Support Vector Machine, DT: Decision Tree, RF: Random Forest, NB: Naive Bayes, k-NN: k Nearest Neighbors, LR: Logistic Regression.

3 Method

In this section the steps that were performed to extract the predictors are explained and the training process and parameter tuning for the classifiers are clarified.

3.1 Dataset

The data used in this study was provided by the Eindhoven University of Technology. It is a log file of their Learning Management System Moodle containing every single user action logged by the system. It spans the academic year 2014-2015 and contains log information about 17 courses with a total of 4601 students. The large samplesize heightens the probability of the data being diverse and representative of a wide variety of students (Halevy et al., 2009). We will use this large dataset to assess the importance of samplesize on the classification performance of the Neural Network. Furthermore, the availability of the 17 courses allow us to compare the effect courses have on classification performance which can give an insight if the model could be applicable to other unseen courses.

In the LMS, students have access to the Moodle homepage from which they can access various course pages belonging to their curriculum. The course pages contain information about the course, announcements from the teachers, pages where the teachers can post course documents like presentation slides or PDFs and a forum where students and teachers can post their questions and answers. Messages can also be sent from one user to another from within the Moodle interface, but these are not specifically tied to a certain course. All the interactions that happen on the course pages, like opening a document, posting on the forum or accessing a certain page, are logged in various tables of the system log.

The Moodle log file is provided in the form of a SQL database containing 359 tables, each table representing a log of a specific Moodle component. For example, the logs pertaining to the accessing of documents and pages are stored in the "mdl27_logstore_standard_log" table, which is also the largest table in the dataset, containing 6.8 million rows and 21 columns. Every aspect that is logged internally by Moodle can be found in a table of this database.

3.2 Experiments

In order to answer the three research questions, three experiments are set up.

Experiment 1 To answer the research question: Is there a difference in accuracy and recall between Neural Networks and six other classifiers for student performance prediction?

In this experiment all predictors (see Table 2) are used as input for the classifiers. The classification accuracy and recall will be compared between classifiers (see Table 3) to determine relative performance.

Experiment 2 To answer the research question: Does exclusion of course predictors influence classification accuracy and recall performance?

In this experiment all predictors except CourseID are used as input for the classifiers. The classification accuracy and recall are compared between classifiers as well as to the results from experiment 1 to determine the influence of course predictors. In order to get an idea of the influence of certain predictors, the Random Forest classifier will be used to generate feature importance statistics.

Experiment 3 To answer the research question: Is there a difference in accuracy and recall between training the Neural Network on all the course data at once versus training it on individual course data?

In this experiment all predictors are used and the data is partitioned by course. A Neural Network is trained on the data of each course and classification performance will be measured for each individual course. These results will be compared to that of the Neural Network that was trained in experiment 1. Also, the effect of sample size will be evaluated by comparing accuracies and sample sizes between classifiers. Furthermore, the feature importances of the best performing classifier will be extracted to assess which features contribute most to good classification performance.

3.3 Predictor Calculation

The data in the database cannot be used as is. Predictors that encode certain properties of the data need to be extracted first. The following predictors were extracted from the data:

- Course ID

- Number of sessions
- Total time online
- Average length of login session
- Total number of clicks
- Average number of clicks per session
- Regularity of logins (frequency)
- Longest time without activity
- Number of forum posts
- Number of messages sent
- Number of quizzes participated
- Average quiz grades
- Number of assignments participated
- Assignment grades

Each predictor (except for the number of messages) is calculated for each student/course combination, for courses taken during the 2014-2015 academic period. The data was extracted and cleaned using the R statistical programming language in RStudio. The data manipulation library DPLYR (Wickham & Francois, 2016) was used to help in this task. The data itself is located in a MySQL database, which was accessed from R with the RMySQL (Ooms, James, DebRoy, Wickham, & Horner, 2016) library. It allows for SQL queries to be executed within R and the results of these queries can then be stored in an R Dataframe. After calculating and cleaning the predictors, they were stored in a Dataframe and saved as a comma-separated values (CSV) file format which can be imported to a Python environment from which the classifiers are trained and executed.

In order to prevent outliers from impacting the data and predictions in unwanted ways, the data of each course is limited to the 10 weeks the course is active. This prevents student actions, like accessing the course page later in the academic year to check assesment grades, to impact predictors like study regularity. The first step was to delete any data that did not belong to the 2014-2015 academic year which lasted from 2014-9-1 to 2015-7-4. Because no information was available as to when each course took place, we had to determine it statistically. The academic year at the TU Eindhoven consists of four quartiles, with each course belonging to exactly one quartile. During the 2014-2015 year, the quartiles were as follows: quartile 1: 2014-9-1 to 2014-11-8, quartile 2: 2014-11-10 to 2015-1-31, quartile 3: 2015-2-2 to 2015-4-18 and quartile 4: 2015-4-20 to 2015-7-4. For each course, the median of all the session dates was calculated. The course was then assigned to the quartile in which the median for that course occured. All course data occuring outside of the quartile in which a course took place was discarded.

Each predictor was calculated as follows:

Course ID: The course ID indicates what course an action belongs to. Every action logged in tables of the database has a corresponding course ID that can directly be extracted from the courseid column or can be retrieved from a related table. For example, the table

mdl27_quiz_grades does not contain the course to which a quiz grade belongs to, but contains a unique identifier for each quiz that can be used to lookup the corresponding course in the mdl27_quiz table.

Number of Sessions, Total Time Online, Average Session Length: Most students do not log out of their Moodle sessions, they just close their browser which does not leave a trace in the logfile. This makes it harder to determine the exact end time of a session. In order to get a general idea of the length the rule that 20 minutes of inactivity (no new actions) would mean the end of a session was used. The session information was extracted from the mdl27_logstore_standard_log table, containing every action performed by a user. Each row in that table corresponds to an action and contains the course that action belongs to, the type of action (logging in, viewing a page, etc.), and the time at which that action occurred. The end time of a session was set as the time of the last action before the inactivity timer elapsed. Also switching from one course to another leads to the end of that session and the start of a new session for another course. The time of first activity in a certain course is used as the start time. To get the session length the start time was subtracted from the end of session time. Sessions that lasted less than 30 seconds were excluded from the list, because they may not correspond to study activities but rather to checking updates on the course page. The total time online by a student for one course is calculated by adding all the session lengths together. The average time per session was calculated by dividing the total time by the number of sessions. Some students did not use the LMS, this means that they have a total time online of 0. Rows with 0 time online were removed as they do not reflect the usage of Moodle. This resulted in the removal of 107 student rows from the data.

Total Number of Clicks, Average Clicks per Session: The click information was extracted from the mdl27_logstore_standard_log table by counting the number of actions of a certain user during one course. Each action, like opening the course page, viewing a document or posting a forum message were counted as a click. The average number of clicks per session was obtained by dividing the total number of clicks by the number of login sessions.

Number of Forum Posts: The mdl27_forum_posts table contains each forum post made by a certain user, but does not contain information about which course the forum post belonged to. In order to determine what forum belonged to what course, the mdl27_forum_discussions table was used. It was then possible to count the number of posts per course and user.

Number of Messages Sent: The amount of messages sent by a certain user could be extracted from the mdl27_message table. This predictor is not tied to a specific course as messages are sent from the general Moodle interface from one user to another.

Regularity of logins: This predictor was calculated using the sessions list that was calculated for the session statistics above. A list was created with the times between sessions, the standard deviation of that list was calculated. A lower standard deviation means a higher regularity.

Longest time without activity: It was determined by looking for the longest time between two sessions during an academic unit of 10 weeks (the time during which the course was active). Holiday were accounted for in the inactivity time as they might skew the inactivity time towards holiday periods. It was done by subtracting the total time of the holiday in seconds from the inactivity time when that period of inactivity occurred during a holiday.

Average Quiz Grades, Number of Quizzes Participated: The grades for quizzes were extracted from the mdl27_quiz_grades table. The grades are mostly on a scale from 0 to 10, but some have more exotic scales going up to 31. The information about what scale was used

and which course a quiz belonged to was extracted from the mdl27_quiz table. The number of quizzes varies from one course to another. It is therefore necessary to take the average of a students quiz grades over one course. The number of quiz grades for one course is used to determine the amount of quizzes a student has participated in, with the reasoning that if a student has a grade for a quiz, he or she has participated in said quiz. Missing quiz grades were replaced by the mean of the available quiz grades, while the missing values for number of quizes participated were replaced by 0. This resulted in the replacement of 2250 missing values for both the quiz grades and number of quizes participated.

Assignment Grades, Number of Assignments Made: The same process was applied for the extraction of the assignment grades. These are stored in the mdl27_assign_grades table, the corresponding course and scales can be found in the mdl27_assign table. For two courses, all assignment grades were set to 0, which can most likely be attributed to an error. As such, the grades for these two courses, corresponding to 18 grades, were deleted. Additionally, not all courses made use of assignments. Missing assignment predictors were replaced by the mean of the available assignment grades, and the missing values for number of assignments made were replaced by 0. This resulted in the replacement of 4418 missing values for both grades and number of assignments made.

Exam Grade: In order to have labels that will be used for predicting, the student grades need to be extracted for each course. These can be found in the totalgrades table. This table contains the students exam grade and the final grade for the course. The final grade is weighted average of the exam grade and assignment and quiz grades. This means that it is directly dependent on the quiz and assignment grades which are used as predictors. This direct dependence is something we want to avoid. The exam grade was therefore used as the label to be predicted as it is not directly dependent on any of the other features. If a student participates in the exam, even if he or she hands in an empty answer sheet, they still receive a grade of 1.0. If the grade is equal to 0 it means that the student did not participate in the exam. All rows with exam grades that were equal to 0 or with missing grades were removed from the data, which resulted in the removal of 240 student rows. Because binary classification is performed (requires assistance or does not require assistance), the grades needed to be transformed from a numerical value (7.1/10 for example) to a categorical one that matches the intent of this study, which is to create a system to detect students that are at risk to fail a course. This is achieved by assigning a 1 (requires assistance) to all grades below 5.5 and a 0 (does not require assistance) to all grades between 5.5 and 10.

Descriptive statistics for the predictors can be found in Table 2.The data contains information for 4601 students, but certain predictors are not available for all students. Some courses did not make us of quizes or assignments for example, as such, only a limited amount of quiz and assignment predictors are available, 2351 and 183 respectively. Missing values have been replaced by either 0 or the mean of that feature column depending on the case.

3.4 Data Preparation

Once the predictors were extracted, the data needed to be normalized. Normalization is necessary for some machine learning algorithms to work properly, like the k-Nearest Neighbor that relies on distance metrics for its objective function. If a feature has a range of values (variance) that surpasses that of other features, it might dominate the objective function of the classifier and make it difficult for other features with smaller variance to influence the learning process. The normalization was performed using Scikit-Learns Standard_Scaler function. This scaler

Table 2: Descriptive statistics of used predictors

Predictor	Occurence in Data	Mean	SD
Number of sessions	4601	33.4	22.4
Total session length (min)	4601	715.1	519.3
Average session length (min)	4601	20.9	9.1
Number of actions	4601	660.9	663.4
Average actions per session	4601	18.4	12.1
Mean assignment grade	183	3.7	4.2
Number of assignments made	183	2.0	1.8
Mean quiz grade	2351	6.4	1.8
Number of quizes made	2351	6.9	2.6
Messagessent	56	1.8	5.1
Nr. of forum posts	69	1.8	1.6
Regularity (hours)	4601	94.9	72.1
Maximum inactivity time (hours)	4601	275.8	149.8
Grade	4601	5.8	2.0
CourseID	4601	-	-

works by subtracting the column mean and dividing by the column standard deviation for each column. This results in a mean of 0 and variance of 1 for each feature.

Three sub datasets were created for each experiment. For experiment 1, all predictors were used. For experiment 2 the CourseID column was removed from the predictor list. For experiment 3, the data was partitioned in 17 subsets, one for each course.

3.5 Classifiers

In order to predict student performance, various classifiers were used. All the algorithms were implemented in the Python programming language. The Neural Network was created using Keras which is a high level Neural Network API that runs on top of TensorFlow. It is aimed at enabling fast prototyping and testing of Neural Networks. The underlying Neural Network API TensorFlow is a machine learning platform developed by Google. In TensorFlow, data is represented in the form of tensors, which are multidimensional arrays that hold the prediction data. Different operations can be performed on these tensors (like normalization, multiplication or classification) as they flow between these operation-nodes, which is where the name TensorFlow originates from.

For all three experiments, various combinations of hyper-parameters were tested for the Neural Network. But for all three, the same set of parameters provided the best results: a Neural Network model consisting of 3 hidden layers with 16 hidden units each. For each layer, the rectified linear unit (ReLU) activation function was used. To set an upper bound for the maximum norm of the weights for each node, we set maxnorm to 5. Weights were updated after every 10 iterations, by setting the batch size to 10. Adam, an optimization function that calculates adaptive learning rates for each parameter, was used with a learning rate of 0.05. Each classifier was trained over 100 iterations. The network was optimized using binary cross-entropy as the loss function.

For the k-Nearest Neighbors, Naive Bayes, Support Vector Machine, Logistic Regression,

Decision Tree and Random Forest classifiers, their implementations in Scikit-Learn were used. This is a machine learning library for Python that contains a wide variety of machine learning algorithms as well as tools to simplify the training, testing and prediction process. Their hyper-parameters were tuned by hand. For the k-Nearest Neighbors algorithm, k=60 was optimal for experiment 1, and k=50 for experiment 2. Logistic Regressions performed best in experiment 1 with a tolerance of 0.2 and experiment 2 with a tolerance of 0.3, both with L1 as the penalization norm. For the Random Forest, the best performance was achieved with 200 trees for both experiments. For the Support Vector Machine, an RBF kernel was selected to maximize performance. Finally, the Naive Bayes classifier did not require any hyper-parameter tuning.

3.6 Result Evaluation

In order to evaluate the performance and thus applicability of the various classification algorithms, the accuracy and recall metrics are used. Accuracy is the percentage of correctly classified instances among the total number of classified instances. This is a widely used evaluation metric in machine learning, which makes it a good metric to compare performance between studies. A higher accuracy means a more accurate and thus higher performing algorithm. When using the algorithm to predict if a person will require academic assistance or not, we want to minimize the number of students that are labeled as not requiring assistance, that are in fact at risk of failing the course. We do not want students that require help slipping through the system unnoticed. We can measure the propensity of the algorithm for these false negatives by measuring the recall. Recall measures how many relevant instances are successfully selected and is calculated by dividing the number of true positives by the number of true positives plus false positives. Maximizing recall will ensure that as few help requiring students go through the system unnoticed. Ideally both accuracy and recall are maximized to obtain the most suitable classification algorithm with the best parameters.

To have a reference to compare the classifiers against, a majority baseline was defined. It looks at the repartition of the labels in the data and always predicts the most frequently occuring label.This allows us to get a better insight in the repartition of the labels within the data. For experiments 1 and 2, with the data for all 17 courses, the majority class is the "does not require assistance" label; out of the 4601 students, 58.3% passed their course and thus do not require any academic assistance. This results in an accuracy of 58.3% but a recall of 0 for this majority baseline. The recall is 0 because it does not predict any "requires assistance" labels. For experiment 3 this majority baseline is calculated for each course individually.

In order to evaluate the importance of certain predictors, feature importance statistics can be generated that give an indication as to which predictor relatively contribute most to a correct prediction. These statistics are extracted from the Random Forest classifier.

One of the risks during the training and parameter tuning phase is over-fitting the classifiers (Srivastava et al., 2014). This means that the parameters of the algorithm are tailored to achieve maximum classification performance on the examples in the training data, but the model does not perform well on other never seen before instances. In order to minimize the risk of over-fitting, 10 fold Cross validation was used for the training and validation phase. This technique splits the training data in 10 folds, at each iteration, nine folds are used to train the algorithm, while the left over fold is used to measure the accuracy of the classifier after training to validate their performance. This process is then repeated for the remaining nine folds, the mean accuracy for all these folds is then calculated and gives an appropriate representation of

the performance of the classifier on new data. Once the classifiers have been trained and the best parameters have been chosen, we record the accuracy and recall.

4 Results

In this section, the results from the three experiments are presented. The first subsection presents the classification results of seven classifiers using all available predictors for all courses. The second subsection contains the classification results of these classifiers using all the predictors excluding CourseID. And finally, the last subsection presents the classification results of the Neural Network for each individual course.

4.1 Experiment 1: classification using all predictors

In this experiment, all predictors (see Table 2) are used to forecast whether a student will pass or fail the course. The classifiers considered for this experiment are k-Nearest Neighbors, Naive Bayes, Support Vector Machine, Logistic Regression, Decision Tree, Random Forest and Neural Network as well as the majority baseline for reference. The results for this experiment can be found in the left part of Table 3.

When looking at the results, the accuracy varies widely between classifiers, ranging from 52.8% to 66.1%. The majority baseline, which predicts a negative outcome (student does not require assitance) for every instance, obtained an accuracy of 58.3% while the Naive Bayes classifier, performed lowest with an accuracy of 52.8%. The Neural Network outperforms all other classifiers when it comes to accuracy with a score of 66.1% followed by Logistic Regression which has an accuracy score that is 3.5% lower than the Neural Network, at 62.4%. Analysis of variance was used to determine if the difference in means between all the classifiers in this experiment, including the majority baseline, is significant. The analysis was performed using the output of the 10-fold cross validation, which provides 10 accuracies for each classifier, one for each fold of the cross validation. The results can be found in Appendix A7. The analysis resulted in a p-value below 0.05, confirming that the difference in results between these classifiers is significant. We then continued by testing the significance of the difference between the Neural Network and the next two best performing classifiers. The difference between the Neural Network and Logistic Regression as well as between the Neural Network and the k-Nearest Neighbors were both significant with $p<0.05$.

Recall results also show a wide range of scores, ranging from 57.1% to 86.7%. The Neural Network obtained a recall of 84.9%, which put it in the third place behind Logistic Regression (85.2%) and k-Nearest Neighbors (86.7%). ANOVA showed with high certainty ($p<0.05$) that the differences between all the classifiers in terms of recall is significant. The differences between the Neural Network and the two best performing classifiers, Logistic Regression and k-Nearest Neighbors was statistically not significant with a p-value largely exceeding 0.05. Results of the significance tests for recall between the various classifiers can be found in Appendix A8.

This experiment showed that the Neural Network surpasses all other considered classifiers in terms of accuracy, while the recall is on par with the remaining high performing classifiers. This makes the Neural Network an excellent candidate for student performance prediction on this dataset using all predictors.

Table 3: Classifier Performance with and without CourseID predictor

	All Predictors		Predictors excl. CourseID	
Classifier	Accuracy	Recall	Accuracy	Recall
Baseline	0.583	0.000	0.583	0.000
kNN	0.607	**0.867**	0.599	0.868
Naive Bayes	0.571	0.822	0.570	0.820
SVM	0.597	0.825	0.618	**0.887**
Logistic Regression	0.624	0.852	0.618	0.870
Decision Tree	0.528	0.571	0.532	0.550
Random Forest	0.568	0.627	0.601	0.721
Neural Network	**0.661**	0.849	**0.652**	0.821

4.2 Experiment 2: classification using all predictors except CourseID

In this experiment we want to measure the effect of knowing what course an instance belongs to has on the classification performance. To do this we left out the predictor that indicates the course: CourseID. The results of this experiment can be found on the right side of Table 3.

In this experiment, the overall accuracies range from 53.2% to 65.2%. The majority baseline, obtained an accuracy of 58.3%. The Neural Network surpasses all the other classifiers when accuracy is considered with a score of 65.2%. It performs 3.4% higher than the second best classifiers, the Support Vector Machine and Logistic Regression that both obtained an accuracy of 61.8%. Using ANOVA the difference between all classifiers in terms of accuracy was significant with $p<0.05$ (see Appendix A7). The differences in accuracy between the Neural Network and Logistic Regression as well as between the Neural Network and the Support Vector machine were also both significant with $p<0.05$.

When recall is considered, the Neural Network ranked fourth with 82.1% while the Support Vector Machine obtained the highest recall at 88.7%. Analysis of variance showed a significant difference between all the considered classifier(see Appendix A8). When looking at individual classifiers, the difference between the Neural Network and k-Nearest Neighbors is significant while the differences between the Neural Network and the Support Vector Machine and Logistic Regression are not.

When comparing these results to those obtained in experiment 1, no clear trend can be observed. Some classifiers saw a slight dip in accuracy compared to the previous experiment, the k-Nearest Neighbors went from 60.7% tot 59.9% for example. While others rose slightly: the Support Vector Machine went from an accuracy of 59.7% to 61.8%. The average accuracy for all the classifiers went from 59.4% in experiment 1 to 59.9% in experiment 2. To determine if this average increase is significant, we used analysis of variance to test the hypothesis that both groups have the same mean. We obtained a p-value far superior to 0.05, which allows us to say with high confidence that the means for both experiments are not significantly different, and thus, results from one experiment are not superior to the other. The recall scores did not contain any generally observable upward or downward trend either. Slight decreases and increases between the experiments can be observed. The average recall went from 77.3% in experiment 1 to 79.1% in experiment 2. Once again analysis of variance was used to test the hypothesis that both groups have the same mean. A p-value superior to 0.05 was obtained, allowing us to conclude that for the recall, the means for both experiments are not significantly different.

In order to get a better insight in the importance of each predictor, feature importance statistics were extracted from the data including CourseID using the Random Forest classifier. The results can be found in Figure 3. These measures can give an insight in how informative certain features are for classification. The feature with the highest importance is regularity, a measure of how regularly a student accessed the course page, with 12.9%. CourseID obtained a lower importance score at 5.3%, ranking ninth out of fourteen features while Mean Quiz Grade was ranked eight with 8.7%. Predictors like assignment grade, number of assignments made, number of messages sent and number of forum posts all had feature importance below 1.0% and thus have low predictive value. The low importance of assignment grade, while still being a previously obtained grade, could be attributed to the fact that it is only available as a predictor for 2 out of the 17 courses.

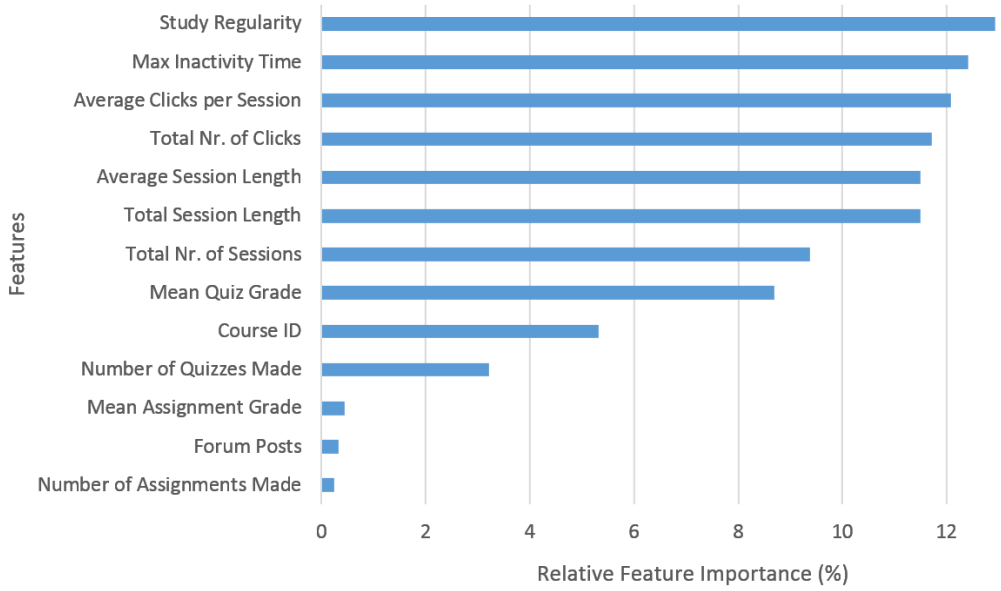

Figure 3: Random Forest relative feature importance.

We can observe that the removal of Course ID has a relatively low impact on prediction accuracy. Additionally, the difference between experiment 1 and 2 are not statistically significant and Course ID has low feature importance. This means that the knowledge of what instance belongs to what course has little impact on the classification outcome for this data.

4.3 Experiment 3: predicting performance per course with the Neural Network

In this experiment, the Neural Network is trained on the data of each course individually and its classification accuracy and recall are measured for each course. This in order to determine if there is an advantage to training the network on all the data or on data for each individual course as well as examining the differences in predictive performance between courses. The results of this experiment can be found in Table 4.

One result that stands out are those for Experimental Physics 2. They are particularly high compared to the performance of the other courses, with both an accuracy and recall of 98.8%. This can most likely be attributed to the fact that in the data, all the 151 students except for 2, passed the course, which comes down to a 98.8% succes rate. This is also directly reflected in the

Table 4: Performance of Neural Network per course

Course Nr.	Course	Accuracy	Baseline Accuracy	Recall	Students	Assignment Grades	Quiz Grades
1	Calculus A	0.581	0.519	0.576	432	-	-
2	Calculus B	0.720	0.519	0.636	1092	-	-
3	Calculus C	0.784	0.730	0.670	222	-	-
4	Calculus for Engineering	0.375	0.712	0.649	114	-	-
5	Set Theory and Algebra	0.735	0.779	0.602	62	-	-
6	Lineaire Algebra and Vector Calculus	0.738	0.684	0.629	107	-	-
7	Linear Algebra	0.775	0.602	0.633	73	-	-
8	Experimental Physics 1	0.814	0.685	0.702	165	-	Yes
9	Experimental Physics 2	0.988	0.988	0.988	151	-	Yes
10	Behavioral Research Methods 2	0.650	0.580	0.737	126	-	Yes
11	Applied Natural Sciences A	0.742	0.685	0.614	710	-	Yes
12	Applied Natural Sciences B	0.488	0.521	0.539	748	-	Yes
13	Condensed Matter	0.610	0.672	0.460	64	-	Yes
14	Intro to Psychology	**0.893**	0.735	**0.761**	143	Yes	Yes
15	Linear Algebra 1	0.567	0.628	0.486	59	-	-
16	Statistics	0.608	0.506	0.609	273	-	Yes
17	The Effectiveness of Mathematics	0.825	0.667	0.733	60	Yes	Yes
	Average	0.699		0.648			

majority baseline classifer that attained an accuracy of 98.8%. The Neural Network probably taught itself to predict every future instance as a "does not require assistance" without looking at any predictors. This would not help in identifying students requiring assistance as the classifier would probably set every instance to pass. Therefore, to avoid this result from skewing the data, it was left out in the calculation of the average accuracy and recall, and from comparisons with other course classifiers.

When looking at the remaining results, prediction accuracies range from 37.5% for the Calculus for Engineering course to 89.3% for Intro to Psychology. When all these scores are averaged, we obtain an accuracy of 69.9%, higher than the general Neural Network accuracy in experiment 1 where 66.1% accuracy was obtained. The recall scores also have high variability ranging from 46.0% for Condensed Matter to 76.1% for Intro to Psychology. When averaged over all courses, the recall is 64.8%, which is lower than the recall for the Neural Network in experiment 1, where it was 84.9%.

We continue by investigating the effect of previous grades (quiz and assignment mean grades) on classification performance. The highest performing classifier, for Intro to Psychology (accuracy 89.3% and recall 76.1%), has quiz and assignment grades as predictors. To get an indication of the importance of this predictor, the Random forest classifier was run on the Intro to Psychology data, and feature importances were extracted, see Appendix A5. This put quiz grade as the most important feature with 16.1%, with the next best two predictors being study regularity at 11.4% and average number of clicks per sessions with 10.7%. This confirms the importance of Quiz Grades and thus previously obtained grades. However, the assignment grades only obtained a low feature importance score of 2.0%. This low importance for the assignment grade could be attributed to the nature of the grades. Some grades might only be a record of someone handing in a certain task and do not reflect how well that task was performed, while other scores are an indication of how well someone performed on an assignment or quiz, for example, getting 74% of the answers right. The latter grade gives more information about how well the student understands the subject matter and could thus be more informative in a classification

scenario.We also extracted feature importances for The Effectiveness of Mathematics course, which also has both quiz and assignment grades as predictors. These feature importances can be found in Appendix A6. In this case, the assignment grades were the most important feature with 22.4% followed by average number of clicks per session with 13.5%. Mean Quiz grade finished last with 0% importance, which could once again be attributed to the nature of the grade. For this course, the assignment grade is the most important feature while quiz grades finished last. One thing that does stand out for both courses is the fact that a previously obtained grade is the most important predictor. This is in line with studies by Richardson et al. (2012) and Dollinger et al. (2008).

Another relation that can be examined using the data from this experiment is the association between sample size and classification performance. One relatively high scoring course, Linear Algebra, has 73 participants and obtained an accuracy of 77.5% with a recall of 63.3%, which is on par with results from the general classifier from experiment 1 in terms of performance. While classifiers trained on larger sample sizes, like Calculus A (432 participants) or Applied Natural Sciences B (748 participants) obtained lower accuracies of 58.1% and 48.8%, and lower recalls of 57.6% and 53.9% respectively. In order to get a clearer picture of the the relation between accuracy and sample size, it has been graphed in Figure 4. No clear relation between sample size and accuracy can be derived from this graph. This would indicate that for this study sample size does not influence predictive performance.

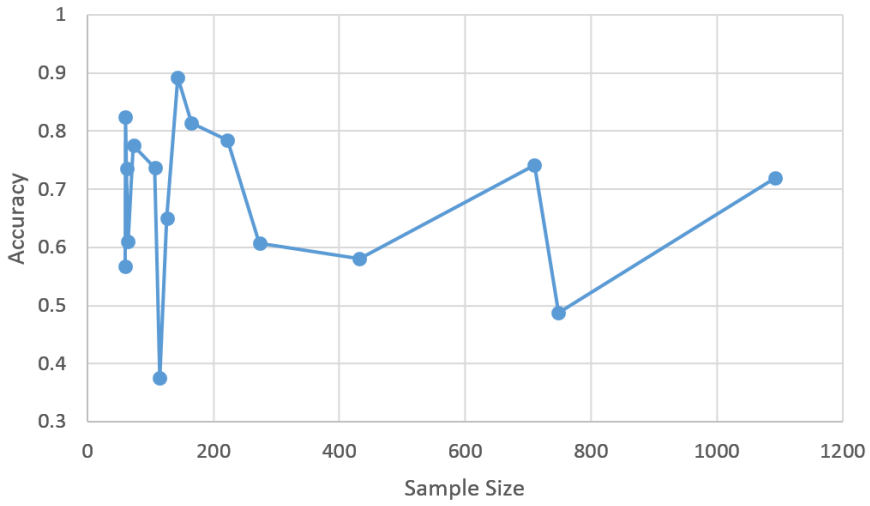

Figure 4: Relation between classification accuracies per course and the samplesize

When trained on each individual course, the Neural Network obtains a lower average accuracy but a higher average recall. Furthermore, results between individual course classifiers vary widely, some falling well below performance seen in experiment 1 (Calculus for Engineering), while others surpass it by a large margin (Intro to Psychology), making them excellent predictors of student performance. This individual training approach should therefore be used on a case to case basis where for each course, the algorithm is trained and assessed individually.

5 Discussion

In this section a general discussion of our research will be provided as well as directions for future research, followed by our conclusions.

5.1 Experiments and Results

The goal of this study was to assess to what extent Neural Networks can be used to predict student performance: assessing whether they did or did not need academic assistance, based on LMS data. In order to answer this problem statement, the following research questions were formulated:

RQ1: Is there a difference in accuracy and recall between Neural Networks and six other classifiers for student performance prediction?

RQ2: Does exclusion of course predictors influence classification accuracy and recall performance?

RQ3: Is there a difference in accuracy and recall between training the Neural Network on all the course data at once versus training it on individual course data?

RQ1: Is there a difference in accuracy and recall between Neural Networks and six other classifiers for student performance prediction?

To answer this question, we compare the performance of the Neural Network to that of six other classification algorithms: k-Nearest Neighbors, Nave Bayes, Support Vector Machine, Logistic Regression, Decision Tree and Random Forest as well as to a majority baseline classifier.

In the first experiment, where all available predictors were used, the Neural Network outperformed every other classifier when looking at accuracy. It managed to predict 66.1% of the student performance outcomes correctly. The second best algorithm, Logistic Regression achieved a 62.4% accuracy and the k-Nearest Neighbor achieved an accuracy of 60.7%. The differences between the Neural Network and the other two best classifiers is statistically significant. When looking at recall, the Neural Network achieved a third place with 84.9%, whereas the highest measured recall was 86.7% for the k-Nearest Neighbors and second best 85.2% for Logistic Regression. However these differences turned out to be statistically not significant with a p-value exceeding 0.05. This means that the Neural Network is on par with the best classifiers in terms of recall and outperforms them in terms of accuracy.

Most other studies do not give recall measures and only focus on accuracy (Jayaprakash et al., 2014; Kabakchieva, 2013; Cortez & Silva, 2008). This means that we can compare accuracy scores between different studies but can not do so for recall. Kabackhieva (2013) obtains similar accuracy scores to those obtained here using Decision Trees (63.1% accuracy) and k-Nearest Neighbours (57% accuracy). Other studies like Cortez et al (2008) and Moucary et al (2011) obtained far higher accuracy scores with Neural Networks, with 88.3% and 91.78% respectively. The difference in accuracy can probably best be explained by the predictors used and the number of classes being predicted in those studies. These studies had used predictors like SAT scores, previous GPAs and admission test grades that contributed to the final grade. These data have been shown to be the most reliable predictors of student success (Richardson et al., 2012; Dollinger et al., 2008). Also, some of these studies did not perform binary classification, involving two labels, but predicted four or even five different performance labels. For example Kabackhieva (2013) used bad, average, good, very good and excellent as labels, which can have a direct impact on the classification performance. For our study we performed binary classification

(does or does not require assistance) and had access to quiz grades and assignment grades. But these predictors were not present for all the courses: we had access to quiz grades for 9 out of 17 courses, and assignment grades for 2 out of 17 courses. Additionaly we did not have acces to any grades that were obtained before the course started, which can have high predictive value (Bai, Chi, & Qian, 2014). If those would be added, predictive performance would most likely increase.

Everything considered, when looking at accuracy, the Neural Network outperforms all other classifiers in this study, which agrees with results by Agrawal et al. (2015).In terms of recall, it is on par with the best performing classifiers tested here. Considering these two performance indicators we can say that the Neural Network is an excellent classifier to predict student performance, and can thus be used to predict whether a student requires academic help. Additionally, due to the limited previous grades predictors available in this dataset, the Neural Network and the other classifiers likely do not attain high accuracy and recall scores compared to other comparabel studies that do contain these predictors. Should these become available, predictive performance would most likely improve.

RQ2: Does exclusion of course predictors influence classification accuracy and recall performance?

In the previous experiment, the Course ID, which indicates what course a certain instance belongs to, was included as a predictor. The reasoning behind it being that it could enable the classifiers to better take the specifics of a certain course into account: different courses might require different study approaches and could vary in LMS usage and thus in LMS predictor data. Leaving this predictor out means no explicit distinction is made in the data as to what course an instance belongs to. This was done to assess if knowledge of the corresponding course influences the performance of the classifiers.

The result of this experiment was a slight gain in accuracy for all the classifiers, with an average increase of 0.5%. However when analysed, this difference was found to be statistically not significant with a p-value exceeding 0.05. When looking specifically at the performance of the Neural Network, its accuracy decreased by 0.9% and recall by 2.8%. For the remaining classifiers, the differences in recall between experiment 1 and 2 varied slighlty, but turned out to be statistically insignificant. So leaving the CourseID out does not greatly affect the performance of the classifier. This could mean that differences between courses do not greatly affect predictive performance on this data when the classifier is trained over a multitude of courses. A similar result was found by Conijn et al. (2017) where it was found that 8% of the variance resided at the course level, and 48% at the student level.

The feature importance statistics from Figure 3 extracted from the Random Forest Classifier give a better insight in how certain features influence the ability of the classifier to predict the correct labels. One feature that stood out in these statistics was the regularity measure, which is a representation of how regularly a student accessed the course page and its content over the duration of the course. It obtained the highest ranking with 12.9%, meaning that knowledge of the regularity of page access is the relative best predictor for student performance when the classifier is trained on a wide variety of courses. Quiz grades placed eighth, with 8.7% which is relatively high, given the fact that these grades are only available for approximately half the students, thus indicating that they are potentially important predictors for student performance. The other previous grade predictor, assignment grades, only obtained an importance score of 0.4%. This low score could largely be attributed to the fact that this predictor is only available for 2 out of the 17 courses. The Couse ID predictor was ranked ninth in terms of feature importance with 5.5%, this makes it of relatively low importance compared to the eight other

predictors ranked above it.

The removal of Course ID had no significant impact on classification performance compared to experiment 1. It was also found that Course ID has low feature importance. We can therefore conclude that for this case, where the classifier is trained on data from all the courses, leaving out the course identifier does not affect predictive performance and thus the course has low influence on generally trained classifiers.

RQ3: Is there a difference in accuracy and recall between training the Neural Network on all the course data at once versus training it on individual course data?

We wanted to assess the effect on classification performance when the Neural Network is trained on each course individually. The data was split up by course and the Neural Network was trained and evaluated on each individual course. Compared to the performance of the Neural Network in experiment 1, which obtained an accuracy of 66.1% and recall of 84.9%, some course classifiers obtained a clear performance increase. The classifier for Intro to Psychology obtained an accuracy of 89.3% with a recall of 76.1%. This performance surpasses results found by Oladokun et al. (2008) which obtained an accuracy of 74.0% with a Neural Network on one course and used admission scores and previous course scores as predictors to predict three performance labels. It is also on par with performance obtained by Calvo et al. (2006) who had an accuracy of 80.2%. For other course classifiers, performance decreased compared to that of experiment 1. Calculus for Engineering achieved an accuracy of 37.5% with 64.9% recall, lower than the majority baseline accuracy for that course of 71.2%.

Performance increased for 10 out of 17 courses compared to the performance from experiment 1. But they still did did not match findings from other studies that used Neural Networks, where accuracies of 90.7% (Cortez & Silva, 2008) or 97.0%(Agrawal & Pandya, 2015) were reported. These differences can most likely be attributed to the predictors used and the different number of classes being predicted in these studies. For example, Cortez et al. (2008) used five different labels for student performance which can influence the predictive behaviour. Additionaly, the predictors being used could play a big role in the performance disparity. These studies had access to a wider range of previously obtained grades: grades from previous courses, high-school grades and college entrance examination scores, all of them important predictors for future academic performance (Bai et al., 2014). While we only had access to a limited set of quiz and assignment grades that were gathered during the ten weeks each course lasted, which could potentially explain the discrepancy in accuracies. Although predictive performance is lower than that found in other studies, a classifier achieving an accuracy of 89.3% and recall of 76.1% can still be succesfully used in an academic system aimed at providing help to students that are at risk to fail a course.

When it comes to sample size, the variations between studies, ranging from 73 (Moucary et al., 2011) to 649 students (Cortez & Silva, 2008) and sample size variations within the courses of this study, did not directly correspond to differences in performance. The accuracies obtained on individual courses with the Neural Network and their corresponding samplesizes were plotted in Figure 4, showing no clear relation between the two variables. This could mean that the sample size has no effect on classification performance, or that the minimum sample size threshold was already surpassed: a sample size of sixty students might already be large enough for this kind of classification task.

Results from the third experiment showed that some course classifiers obtained higher performance than obtained in experiment 1, whilst others scored below it. The use of classifiers trained on individual course data should therefore be assessed on a case to case basis.

5.2 Future Research

By pointing out limitations of this study we can define some suggestions for future research.

- The data used for prediction only represents information about the student right before the exam. Should the classifier really be used as a preventive instrument to decrease academic dropout, the data should be sampled at different intervals during the course. Intervening just before the exam might not be useful anymore: a timely approach is better suited (Campbell et al., 2007).
- One limitation in this study was the lack of consistent previous grades, one of the best predictors for future academic performance. In our case, we had access to a few (assignment grades for 2 out of 17 courses, quiz grades for 9 of them) but these did not contribute much to the general predictive performance due to their rarity. For future research, obtaining these grades could be a way to increase predictive performance.
- Another direction for future research could focus on the minimum sample size needed in student performance prediction to attain good prediction performance. In this study, there was no apparent link between sample size and predictive performance. This could be due to the fact that we already reached a certain minimum threshold for sample size. The knowledge of minimum sample size would allow one to determine if the automated warning approach is suitable for small classrooms.
- The last direction for future research is the social acceptability of decisions made by a computer when it comes to these preventive measures. Especially given the black box nature of Neural Networks: will students accept the fact that they are being contacted by an academic advisor based on a decision made by a Neural Network and data that was gathered from their online activities. Research should be performed among students regarding the acceptability of such measures.

5.3 Conclusion

The goal of this thesis was to assess to what extent Neural Networks can be used to predict student performance based on LMS data.

We showed that predictive performance of the Neural Network on all courses at once exceeded that of other classifiers in terms of accuracy, while being on par with other classifiers in terms of recall. Leaving out the course predictor did not have a major impact on this performance. However,the Neural network and the six other classifiers did not outperform findings from other studies, most likely attributable to the difference in used predictors and in study setup. We then trained the Neural Network on each course individually, which resulted in an increase in performance for some course classifiers and a decrease for others compared to the performance of the Neural Network in the first experiment. The effect of samplesize was investigated, but no relation between samplesize and accuracy was found. Additionaly, the feature importance analysis showed that previously obtained grades were the most valuable predictors for individually trained classifiers.

We can conclude that Neural Networks are applicable to student performance prediction and outperform classifiers like k-Nearest Neighbors, Naive Bayes, Support Vector Machine, Logistic Regression, Decision Tree and Random Forests when general training is used. Additionally,

the use of Neural Networks trained on individual course data resulted in good predictive performance for the majority of the courses, but should be assessed on a course to course basis. Furthermore, in order to attain the highest possible predictive performance, previous grades should be included in the data.

References

Agrawal, R., & Pandya, M. (2015). Data mining with neural networks to predict students academic achievements. *ICJST*, *7*(2). Retrieved from http://www.ijcst.com/vol72/1/19-richa-shambhulal-agrawal.pdf

Bai, C.-e., Chi, W., & Qian, X. (2014). Do college entrance examination scores predict undergraduate gpas? a tale of two universities. *China Economic Review*, *30*, 632–647.

Baker, R. S., & Inventado, P. S. (2014). Educational data mining and learning analytics. In *Learning analytics* (pp. 61–75). Springer. Retrieved from https://www.researchgate.net/profile/Paul_Inventado/publication/278660799

Calvo-Flores, M. D., Galindo, E. G., Jiménez, M. P., & Piñeiro, O. P. (2006). Predicting students marks from moodle logs using neural network models. *Current Developments in Technology-Assisted Education*, *1*, 586–590.

Campbell, J. P., Oblinger, D. G., et al. (2007). Academic analytics. *EDUCAUSE review*, *42*(4), 40–57.

Cerny, P. A., & Proximity, M. A. (2001). Data mining and neural networks from a commercial perspective. In *Orsnz conference twenty naught one* (pp. 1–10). Retrieved from https://pdfs.semanticscholar.org/0d07/80d71b5c8b1c907c19ce73fb2cc4929e9976.pdf

Conijn, R., Snijders, C., Kleingeld, A., & Matzat, U. (2017). Predicting student performance from lms data: A comparison of 17 blended courses using moodle lms. *IEEE Transactions on Learning Technologies*, *10*(1), 17–29.

Cortez, P., & Silva, A. M. G. (2008). Using data mining to predict secondary school student performance. .. Retrieved from https://repositorium.sdum.uminho.pt/bitstream/1822/8024/1/student.pdf

Costa, Y. M., Oliveira, L. S., & Silla, C. N. (2017). An evaluation of convolutional neural networks for music classification using spectrograms. *Applied Soft Computing*, *52*, 28–38. Retrieved from http://www.inf.ufpr.br/lesoliveira/download/ASOC2017.pdf

Dietz-Uhler, B., & Hurn, J. E. (2013). Using learning analytics to predict (and improve) student success: A faculty perspective. *Journal of Interactive Online Learning*, *12*(1), 17–26. Retrieved from http://s3.amazonaws.com/academia.edu.documents/33450256/12.1.2.pdf

Dollinger, S. J., Matyja, A. M., & Huber, J. L. (2008). Which factors best account for academic success: Those which college students can control or those they cannot? *Journal of research in Personality*, *42*(4), 872–885. Retrieved from http://s3.amazonaws.com/academia.edu.documents/32955335/Dollinger

Everitt, B. S., Landau, S., Leese, M., & Stahl, D. (2011). An introduction to classification and clustering. *Cluster Analysis, 5th Edition*, 1–13.

Gašević, D., Dawson, S., Rogers, T., & Gasevic, D. (2016). Learning analytics should not promote one size fits all: The effects of instructional conditions in predicting academic success. *The Internet and Higher Education*, *28*, 68–84. Retrieved from https://www.researchgate.net/profile/Dragan_Gasevic/publication/283730873_Learning_analytics_should_not_promote_one_size_fits_all_The_effects_of_instructional_conditions_in_predicting_academic_success/links/5645fd5c08ae9f9c13e72c60.pdf

Gaur, P. (2012). Neural networks in data mining. *International Journal of Electronics and Computer Science Engineering (IJECSE, ISSN: 2277-1956)*, *1*(03), 1449–1453. Retrieved from https://pdfs.semanticscholar.org/9bc2

Glosser.ca. (2013). *Licensed under creative commons attribution-share alike 3.0 unported.* Retrieved from https://commons.wikimedia.org/wiki/File:Colored_neural

_network.svg

Graves, A., & Jaitly, N. (2014). Towards end-to-end speech recognition with recurrent neural networks. In *Icml* (Vol. 14, pp. 1764–1772). Retrieved from http://www.csri.utoronto.ca/graves

Gupte, A., Joshi, S., Gadgul, P., Kadam, A., & Gupte, A. (2014). Comparative study of classification algorithms used in sentiment analysis. *International Journal of Computer Science and Information Technologies*, *5*(5), 6261–6264. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.660.5055&rep=rep1&type=pdf

Halevy, A., Norvig, P., & Pereira, F. (2009). The unreasonable effectiveness of data. *IEEE Intelligent Systems*, *24*(2), 8–12. Retrieved from https://research.google.com/pubs/archive/35179.pdf

Jayaprakash, S. M., Moody, E. W., Lauría, E. J., Regan, J. R., & Baron, J. D. (2014). Early alert of academically at-risk students: An open source analytics initiative. *Journal of Learning Analytics*, *1*(1), 6–47. Retrieved from http://www.learning-analytics.info/journals/index.php/JLA/article/viewFile/3249/4011

Johnson, D., & Samora, D. (2016). The potential transformation of higher education through computer-based adaptive learning systems. *Global Education Journal*, *2016*(1). Retrieved from http://web.a.ebscohost.com/abstract112407351

Jordan, K. (2014). Initial trends in enrolment and completion of massive open online courses. *The International Review of Research in Open and Distributed Learning*, *15*(1). Retrieved from https://www.researchgate.net/profile/Katy_Jordan/publication/260316869

Kabakchieva, D. (2013). Predicting student performance by using data mining methods for classification. *Cybernetics and Information Technologies*, *13*(1), 61–72. Retrieved from https://www.degruyter.com/downloadpdf/j/cait.2013.13.issue-1/cait-2013-0006/cait-2013-0006.pdf

Karpathy, A., Toderici, G., Shetty, S., Leung, T., Sukthankar, R., & Fei-Fei, L. (2014). Large-scale video classification with convolutional neural networks. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 1725–1732).

Kovacic, Z. (2012). Predicting student success by mining enrolment data. *Research in Higher Education Journal*, *15*, 120. Retrieved from https://repository.openpolytechnic.ac.nz/bitstream/handle/11072/1486/Kovacic%20~%20Predicting%20student%20success%20by%20mining%20enrolment%20data.pdf?sequence=1&isAllowed=y

Maren, A. J., Harston, C. T., & Pap, R. M. (2014). *Handbook of neural computing applications*. Academic Press.

Meyer, D. (2015). *Support vector machinesthe interface to libsvm in package e1071.(2014)*. Retrieved from https://cran.r-project.org/web/packages/e1071/vignettes/svmdoc.pdf

Moucary, C., Khair, M., & Zakhem, W. (2011). Improving students performance using data clustering and neural networks in foreign-language based higher education. *The Research Bulletin of Jordan ACM*, *2*(3), 27–34. Retrieved from http://ijj.acm.org/volumes/volume2/no3/ijjvol2no3p1.pdf

Murty, M., & Raghava, R. (2016). Kernel-based svm. In *Support vector machines and perceptrons* (pp. 57–67). Springer. Retrieved from https://link.springer.com/chapter/10.1007/978-3-319-41063-0_5

Nametala, C. A., Pimenta, A., Pereira, A., & Carrano, E. G. (2016). An automated investment strategy using artificial neural networks and econometric predictors. In *Proceedings of the xii brazilian symposium on information systems on brazilian symposium on information systems: Information systems in the cloud computing era-volume 1* (p. 21). Retrieved

from http://dl.acm.org/citation.cfm?id=3021982

Oladokun, V., Adebanjo, A., & Charles-Owaba, O. (2008). Predicting students academic performance using artificial neural network: A case study of an engineering course. *The Pacific Journal of Science and Technology*, *9*(1), 72–79. Retrieved from http://www.akamaiuniversity.us/PJST9_1_72.pdf

Ooms, J., James, D., DebRoy, S., Wickham, H., & Horner, J. (2016). Rmysql: Database interface and 'mysql' driver for r [Computer software manual]. Retrieved from https://CRAN.R-project.org/package=RMySQL

Papamitsiou, Z. K., & Economides, A. A. (2014). Learning analytics and educational data mining in practice: A systematic literature review of empirical evidence. *Educational Technology & Society*, *17*(4), 49–64. Retrieved from http://ai2-s2-pdfs.s3.amazonaws.com/d817

Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2000). A model for types and levels of human interaction with automation. *IEEE Transactions on systems, man, and cybernetics-Part A: Systems and Humans*, *30*(3), 286–297. Retrieved from https://pdfs.semanticscholar.org/4209/578fea72d5c334be8945f5b74641feb0908e.pdf

Payne, G., & Williams, M. (2005). Generalization in qualitative research. *Sociology*, *39*(2), 295–314.

Peña-Ayala, A. (2014). Educational data mining: A survey and a data mining-based analysis of recent works. *Expert systems with applications*, *41*(4), 1432–1462. Retrieved from http://s3.amazonaws.com/academia.edu.documents/41380156

Ramsden, P., & Entwistle, N. J. (1981). Effects of academic departments on students'approaches to studying. *British Journal of Educational Psychology*, *51*(3), 368–383. Retrieved from http://onlinelibrary.wiley.com/doi/10.1111/j.2044-8279.1981.tb02493.x/full

Richardson, M., Abraham, C., & Bond, R. (2012). Psychological correlates of university students' academic performance: a systematic review and meta-analysis. *Psychological bulletin*, *138*(2), 353. Retrieved from http://emilkirkegaard.dk/en/wp-content/Psychological-correlates

Rokach, L., & Maimon, O. (2014). *Data mining with decision trees: theory and applications*. World scientific. Retrieved from https://eric.univ-lyon2.fr/~ricco/tanagra/fichiers/fr_Tanagra_DM_with_Decision_Trees.pdf

Romero, C., & Ventura, S. (2010). Educational data mining: a review of the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, *40*(6), 601–618. Retrieved from https://www.researchgate.net/profile/Paulo_Cortez3/publication/228780408

Romero, C., Ventura, S., & García, E. (2008). Data mining in course management systems: Moodle case study and tutorial. *Computers & Education*, *51*(1), 368–384. Retrieved from http://www.sciencedirect.com/science/article/pii/S0360131507000590

Saunders, C., Stitson, M. O., Weston, J., Bottou, L., & Smola, A. (1998). *Support vector machine-reference manual.* Royal Holloway, University of London. Retrieved from https://eprints.soton.ac.uk/258959/1/SVM_Reference.pdf

Scheuer, O., & McLaren, B. M. (2012). Educational data mining. In *Encyclopedia of the sciences of learning* (pp. 1075–1079). Springer. Retrieved from http://link.springer.com/referenceworkentry/10.1007%2F978-1-4419-1428-6_618

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, *61*, 85–117. Retrieved from https://pdfs.semanticscholar.org/463c

Srivastava, N., Hinton, G. E., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine*

Learning Research, *15*(1), 1929–1958. Retrieved from https://pdfs.semanticscholar.org/6c8b

Stapel, M., Zheng, Z., & Pinkwart, N. (2016). An ensemble method to predict student performance in an online math learning environment. In *Proceedings of the 9th international conference on educational data mining, international educational data mining society* (pp. 231–238).

Venugopalan, S., Xu, H., Donahue, J., Rohrbach, M., Mooney, R., & Saenko, K. (2014). Translating videos to natural language using deep recurrent neural networks. *arXiv preprint arXiv:1412.4729*. Retrieved from https://arxiv.org/pdf/1412.4729.pdf

Wang, F.-Y., Zhang, J. J., Zheng, X., Wang, X., Yuan, Y., Dai, X., ... Yang, L. (2016). Where does alphago go: from church-turing thesis to alphago thesis and beyond. *IEEE/CAA Journal of Automatica Sinica*, *3*(2), 113–120. Retrieved from https://www.researchgate.net/profile/Yong_Yuan14/publication/309428013

Wickham, H., & Francois, R. (2016). dplyr: A grammar of data manipulation [Computer software manual]. Retrieved from https://CRAN.R-project.org/package=dplyr

Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). *Data mining: Practical machine learning tools and techniques*. Morgan Kaufmann. Retrieved from https://www.elsevier.com/books/data-mining/witten/978-0-12-804291-5

A Appendix

Table 5: Feature Importances, Course: Intro to Psychology

Feature	Relative Importance (in %)
Mean Quiz Grade	16.1
Study Regularity	11.4
Average Clicks per Session	10.7
Number of Quizzes Made	10.4
Total Session Length	10.1
Max Inactivity Time	9.7
Total Nr. of Sessions	9.6
Total Nr. of Clicks	9.4
Average Session Length	9.3
Mean Assignment Grade	2.0
Number of Assignments Made	1.2
Messages Sent	0.0
Forum Posts	0.0

Table 6: Feature Importances, Course: The Effectiveness of Mathematics

Feature	Relative Importance (in %)
Mean Assignment Grade	22.4
Average Clicks per Session	13.5
Total Nr. of Clicks	11.9
Study Regularity	11.6
Total Session Length	10.3
Max Inactivity Time	10.0
Total Nr. of Sessions	9.6
Average Session Length	8.9
Number of Assignments Made	1.9
Number of Quizzes Made	0.0
Mean Quiz Grade	0.0
Forum Posts	0.0
Messages Sent	0.0

Table 7: Results of ANOVA on accuracy scores between classifiers

Experiment 1				
Group 1	Group 2	F-score	p-value	Significant (95%)
	All classifiers*	6.04	0.000018	Yes
Neural Network	Logistic Regression	13.69	0.001639	Yes
	k-Nearest Neighbors	26.12	0.000073	Yes
Experiment 2				
Group 1	Group 2	F-score	p-value	Significant (95%)
	All classifiers*	12.54	0.000000	Yes
Neural Network	Logistic Regression	5.26	0.034134	Yes
	Support Vector Machine	10.40	0.004692	Yes

** These classifiers are: the majority baseline, Naive Bayes, k-Nearest Neighbors, Support Vector Machine, Decision Tree, Random Forest and Logistic Regression.*

Table 8: Results of ANOVA on recall scores between classifiers

Experiment 1				
Group 1	Group 2	F-score	p-value	Significant (95%)
	All Classifiers*	8.43	0.000001	Yes
Neural Network	Logistic Regression	0.01	0.944420	No
	k-Nearest Neighbors	0.45	0.509635	No
Experiment 2				
Group 1	Group 2	F-score	p-value	Significant (95%)
	All classifiers*	14.16	0.000000	Yes
Neural Network	Logistic Regression	1.06	0.316988	No
	Support Vector Machine	8.69	0.008612	Yes
	k-Nearest Neighbors	2.92	0.104488	No

** These classifiers are: the majority baseline, Naive Bayes, k-Nearest Neighbors, Support Vector Machine, Decision Tree, Random Forest and Logistic Regression.*