Early and Dynamic Student Achievement Prediction in E-Learning Courses Using Neural Networks

Ioanna Lykourentzou, Ioannis Giannoukos, George Mpardis, Vassilis Nikolopoulos and Vassili Loumos

Multimedia Technology Laboratory, School of Electrical and Computer Engineering, National Technical University of Athens, Zographou Campus, 15773 Athens, Greece,

Email:{ioanna, igiann, gmpardis, vnikolop}@medialab.ntua.gr; loumos@cs.ntua.gr

Abstract

The increasing popularity of e-learning has created a need for accurate student achievement prediction mechanisms, allowing instructors to improve the efficiency of their courses by addressing specific needs of their students at an early stage. In this paper, a student achievement prediction method applied to a ten-week introductory level e-learning course is presented. The proposed method uses multiple feed-forward neural networks to dynamically predict students' final achievement and to cluster them in two virtual groups, according to their performance. Multiple choice test grades were used as the input data set of the networks. This form of tests was preferred for its objectivity. Results showed that accurate prediction is possible at an early stage, more specifically at the third week of the ten-week course. In addition, when students were clustered, low misplacement rates demonstrated the adequacy of the approach. The results of the proposed method were compared against those of linear regression and the neural network approach was found to be more effective in

all prediction stages. The proposed methodology is expected to support instructors in providing better educational services as well as customized assistance according to students' predicted all prediction stages. The proposed methodology is expected to support instructors in providing better educational services as well as customized assistance according to students' predicted level of performance.

Key words: e-learning, neural networks, student achievement prediction, student clustering, multiple choice tests

Introduction

The emergence of e-learning introduced significant improvements in the way courses are taught and delivered, making this new form of education widely acceptable. Contrary to the classical approach, e-learning does not require the physical presence of students in a classroom; it is more flexible, less expensive and it facilitates managing large numbers of students, as established by Karlovcec et al. (2005).

The widespread adoption of e-learning introduced new possibilities as well as new challenges. Current research focuses on improving the quality of this type of education by introducing innovative tools and methods. Such innovations may include mechanisms to adapt the difficulty level of exercises, to arrange students in homogeneous groups and to motivate them according to their predicted level of performance.

These mechanisms require a method capable of accurately predicting student achievement as early as possible. This paper describes the implementation of such a method, which uses neural networks and multiple choice test data to early estimate the final grades of e-learning students. Neural networks were chosen for their efficiency in function approximation, in our case the function that relates final student grades to their test grades during the course, and for their generalization ability. Multiple choice tests are considered to be an objective method for student grading and, as such, they formed the data set of this study. The results of the method are then used to dynamically cluster e-learning students into two groups based on their predicted level of performance. This clustering mechanism can be used by the instructor to select educational material more suitable to the abilities of each group of students. Student achievement prediction can also help identify weak trainees at early stages and properly assist them to better cope with course requirements.

This paper is structured as follows: First, we describe the results and limitations of relevant research literature. Next, the proposed method is presented, including a description of the course, upon which the method was applied, and an introduction to the theoretical basis of feed-forward neural networks. The experimental results obtained by this new approach are then presented, compared to the linear regression method and discussed. The paper concludes with the main findings of this study, potential applications and future work.

Relevant Literature

Relative studies based on neural networks have been carried out on data from school, college and distance education courses aiming to predict student achievement. Junemann et al. (2007) used neural networks to predict future student schooling performance based on students' family, social and wealth characteristics. The aforementioned work focused on predicting the achievement of 15 year old students on Reading, Math and Science courses. Wand et al. (2002) used neural networks to estimate the number of errors that a trainee will make based on problem-specific attributes and the trainees' current level. This prediction however, was applied on a single examination to optimize the selection of the problems the student was asked to solve in latter steps of the same examination. The prediction of academic performance, using college student data was studied by Cripps (1996). In this work, various demographic features (age, gender and race) as well as college entrance examination results were used to train a neural network in order to estimate student program completion and final grade. Nonetheless, no new data that occurred throughout student progress were used to dynamically upgrade this estimation. Sheel et al. (2001) compared neural networks and statistical modeling to cluster students into two groups using a single mathematical placement test. Student data derived from distance education were used by Kalles et al. (2006) and Kotsiantis et al. (2004) to predict success or failure in final exams through multiple techniques including neural networks. The data covered demographic characteristics, homework assignment grades and plenary class meeting attendance levels.

The aforementioned studies referred to different types of education, such as classical and distance education. They did not however focus on the distinctive attributes of e-learning and more specifically its fully computerized features as well as the extended interactivity among instructors and trainees which these courses present. These studies also relied on demographic characteristics and homework assignments often corrected by different instructors introducing, in consequence, a certain amount of subjectivity.

On the other hand, the proposed prediction method can easily be integrated into an e-learning management system and, through the technique of dynamic student clustering, further promote interactivity. Additionally, this study relies on data from multiple choice tests which are considered to be objective in grading.

Methods

The proposed method is applied on the results of multiple choice tests from an e-learning course. Contrary to written assignments, the correction of multiple choice tests is unambiguous and objective, as established by Haladyna (2004), thus minimizing extraneous factors which might influence the data.

This study is based on an introductory level e-learning course on Computer Networks & Communications. The course is provided by the e-learning team of the Multimedia Technology Laboratory of the National Technical University of Athens, Greece (Medialab–NTUA, 2007). The course is delivered through the Moodle platform (Moodle, 2007), an open source Learning Management System.

The data originate from a 10-week course offered twice: in Spring 2006 and Spring 2007. The total number of students enrolled in the Spring 2006 course was 37, out of which, 32 successfully completed the course. The number of students enrolled in the Spring 2007 course was 28, out of which, 25 successfully completed it. The course content is predetermined and fixed, so no significant changes were made from one semester to the other. The fact that all course attributes remain unchanged, allows the proposed method to produce valid results when applied in later semesters.

As far as the testing material is concerned, students completed four twenty-question multiple choice tests each semester. These tests were conducted in the first, third, fourth and sixth week, after completing each of the four main chapters the course was divided into. At the end of each ten-week course, a final multiple choice test of forty questions was conducted, as an overall assessment of students' comprehension on the entire course content.

Finally, students had the choice of dropping the course and re-attend it on a forthcoming semester without additional cost.

Feed-Forward Neural Networks

Neural networks have been successfully applied to various research and industry fields to perform tasks including forecasting, data classification and regression analysis.

A typical feed-forward neural network (FFNN), as described by Haykin (1999), consists of one or more hidden layers of neurons. In this type of network, neuron connections, called synapses, do not form a directed cycle. The information moves only forward, from the input to the output nodes. During its learning phase, the network is presented with a set of examples which form the network training set. Each example consists of an input vector and the corresponding output vector. The goal of the FFNN training is to minimize a cost function typically defined as the mean square error between its actual and target outputs, by adjusting the network synaptic weights and neuron biases. More specifically, these network parameters are adjusted based on the back-propagation (Rumelhart et al., 1986a, 1986b) algorithm. According to this algorithm, information is passed forwardly from the input nodes, through the hidden layers, to the output nodes and the error between the desired and the actual response of the network is calculated. Then, this error signal is propagated backwards to the input neurons adjusting the network weights and biases. This process is repeated for each example in the training set. As soon as the entire training set has been presented to the network, an epoch has elapsed. The training phase may consist of several epochs. A popular approach to optimize the performance of back-propagation is the

Levenberg-Marquardt algorithm (Hagan and Menhaj, 1994), which has been found to increase the speed convergence and effectiveness of the network training.

During its training, a FFNN may end up memorizing the training data, and thus lose its ability to generalize from the training samples to an unseen population. This phenomenon is called overfitting and can be avoided by using a separate data set called the validation set. The FFNN parameters are estimated based only on the training set, and the performance of the network is evaluated by computing the mean square error on the validation set. When the network performance is found to deteriorate, meaning that overfitting occurred, training stops and the weights and biases of the best previously trained network are stored. The training phase can be terminated by reaching a minimum in the cost function, meeting the performance goal or by detecting that the validation set produced increasing mean square error.

Finally, after the training is finished, the network test phase takes place. During this phase, unseen data are presented to the trained network to evaluate its performance. These data compose the test set which is disjoint to both the training and the validation data sets.

Strengths and limitations

Neural networks present various strengths which make them suitable for regression analysis and prediction tasks. One of their main advantages is that they are universal function approximators. They can approximate arbitrary continuous functions to any degree of accuracy (Hornik et al., 1989; Cybenko, 1989; Funahashi, 1989; Hornik, 1991, 1993). As a result, neural networks have the ability to efficiently map nonlinear relationships between their input and output.

Additionally, based on the function they have approximated, they can generalize. A neural network can learn from examples and correctly predict the output of unseen data, even if its training set contains noisy information. The robustness of neural networks, in the presence of noise in the input data, is one of their most significant advantages (Thrun, 1994).

Another strength that neural networks present is that they are data-driven instead of model-driven. This means that they do not a priori assume an explicit relationship model among the data, as model-based linear or nonlinear methods do. Instead, they make their predictions based on the actual model that exists among the data of the problem.

Real-world problems are often nonlinear and the relationship among their data is difficult to describe analytically. Usually in such problems, the only available information is past data and the prediction of future performance can either be made by assuming the data model or by approaching it using a machine learning technique, like neural networks. The unique characteristics that the latter present – arbitrary function approximation, nonlinearity, generalization capability - often make them appropriate for real-world applications.

Finally, trained neural networks can quickly make predictions on unseen data. This characteristic, along with their high degree of accuracy, makes them suitable for applications where training needs to be made sporadically but predictions should be made in real-time.

Nevertheless, besides their strengths neural networks also present certain limitations. Firstly, in problems where the relationship among the data is linear with little noise, they may not perform better than linear statistical methods (Zhang et al., 1998).

Secondly, neural networks usually require more time for training than linear methods due to the number of iterations needed to achieve their optimal prediction. More specifically, while minimizing the cost function during training, they may be trapped in local minima, not achieving the optimal solution. To overcome this, multiple training iterations usually take place and the most efficiently trained network is selected (Iyer and Rhinehart, 1999).

Another limitation that neural networks present is their dependency on the size and quality of the data used for their training (Haykin, 1999). The more indicative the examples of the problem they are presented with, the more accurate the predictions they are expected to make. In addition, although they can infer a correct solution based on noisy data, they have difficulty in making correct predictions on data which are contradictory to the ones used for their training.

Finally, neural networks are black-box methods. As such, they cannot be analyzed in great detail like linear models and the data relationship that they approach cannot be easily described (Andrews et al. 1995).

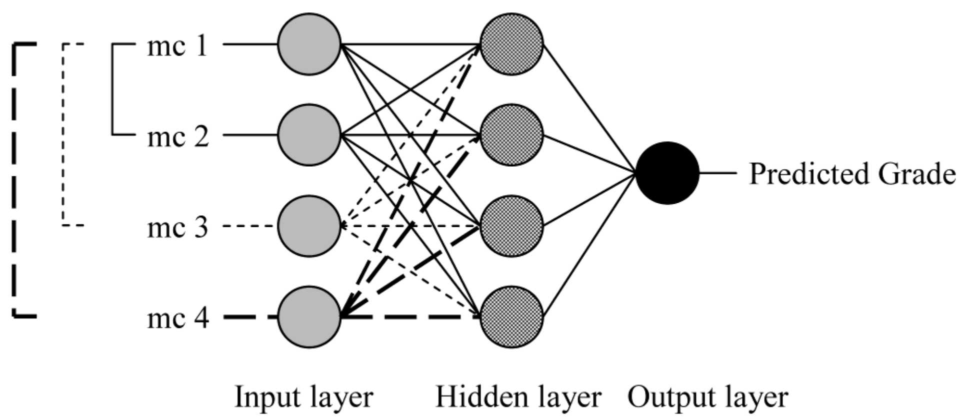
Experimental Results and their Interpretation

Neural Networks for Student Achievement Prediction

In this paper, three feed-forward neural networks were implemented to predict student achievement, by approximating the function which maps students' early test scores to their final test grade. To implement the networks, the MATLAB 2007b (MATLAB, 2007) platform was used.

The procedure included the following steps: First, the data set, to be fed into the neural networks, was extracted from the Learning Management System Database. Since the output vector of the neural networks consists of the results of the final multiple choice test completed by the students at the end of the course, only data from the 57 students that successfully completed the Spring 2006 and Spring 2007 courses were used. The above data set was then divided into three separate sets, namely the training, validation and test set. The network training set consisted of 85% of the Spring 2006 student grade data, which corresponds to 27 randomly selected students. The remaining 15%, that is, data from 5 students were used as the validation set. The network test set consisted of data from the 25 students that successfully completed the Spring 2007 course. The use of one semester data (Spring 2006) for the training phase and the other semester data (Spring 2007) for the test phase was chosen to examine whether neural networks are capable of predicting unseen student data from a different semester.

Next, the training phase took place. During this phase three X-4-1 feed-forward neural networks were constructed, as shown in figure 1. These networks received as input (X) the grades of the first two (mc1 and mc2), three (mc1, mc2 and mc3) and four (mc1, mc2, mc3 and mc4) multiple choice tests respectively. A separate neural network using as input only the first multiple choice test grades (mc1) was not constructed because it would be too early to extract safe results based on this test alone. Training was conducted using the Levenberg-Marquardt algorithm implemented in the MATLAB Neural Network Toolbox. During the experiments, for each one of the three networks, 100 training iterations took place and the best trained network was kept. Computational time for the training phase of each network, on a PC with a 3GHz Intel processor, did not exceed 1 minute.



1st neural network (NN1)
2nd neural network (NN2)
3rd neural network (NN3)
mc multiple choice test grades

FIG. 1. The three neural networks used.

The networks, as shown in figure 2(i-iii), were efficient in minimizing the performance criterion (mean square error) between their response and the examinees' final test grade. In these figures, the vertical axis corresponds to the mean square error and the horizontal to the epochs needed for the training to finish.

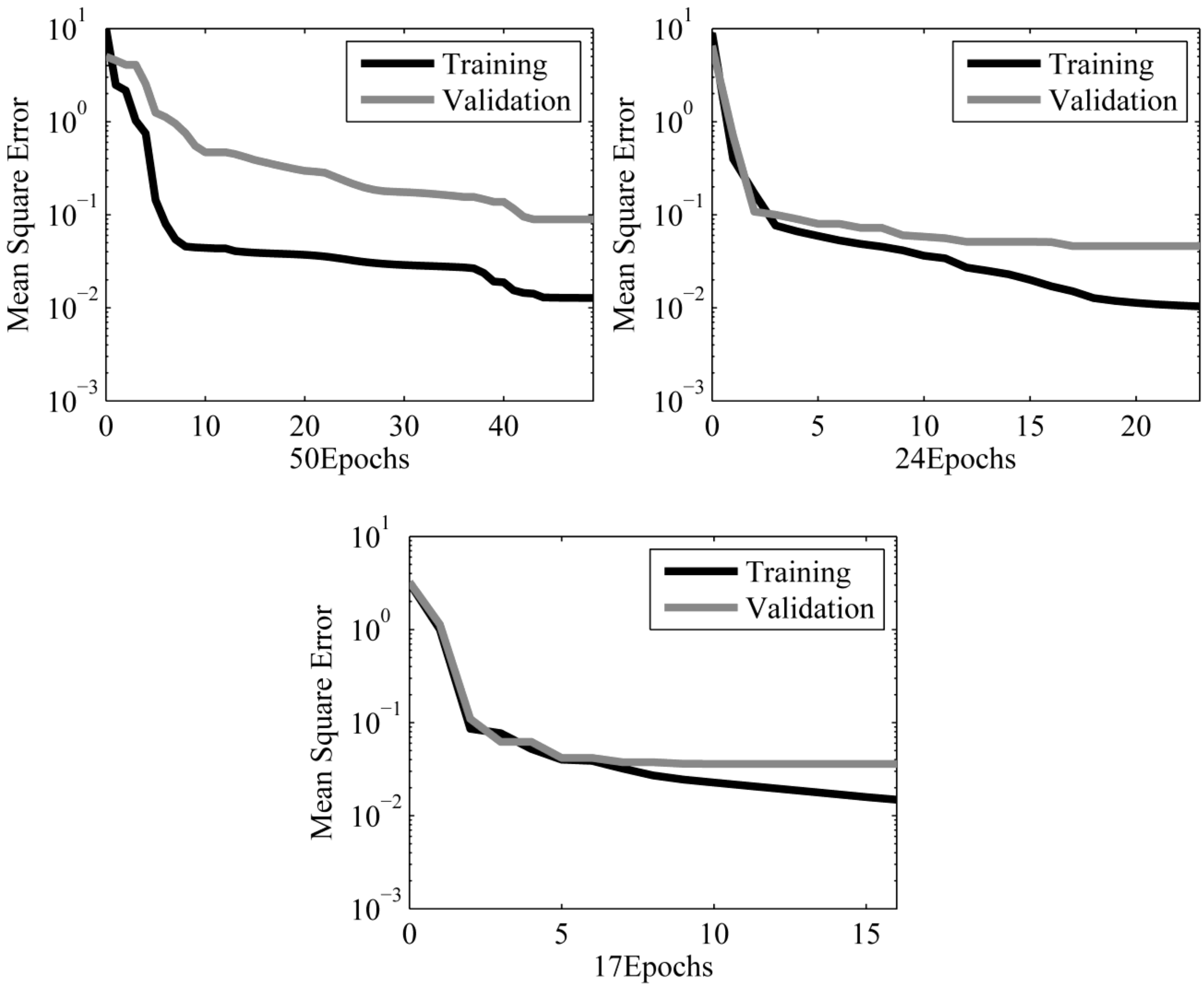


FIG. 2. (i)-(iii) Neural Network training

Finally, the trained neural networks were presented with the test data set to examine their training quality. Computational time during the test phase for each network was less than 1 second. The performance of the networks during the test phase is shown in figure 3(i-iii). These figures depict the relation between the response of the neural network (vertical axis) and the desired response (horizontal axis). Results showed that student grade prediction is possible in an early stage, that is the third week of the course, achieving a correlation coefficient R value equal to 0.9154. The third multiple choice test was found to be the most representative of the trainees' final achievement grade, given that adding the results of this test to the network input increased its performance to R=0.9453. The sixth-week test fine-tuned the output of the last neural network to achieve a prediction of R=0.9521.

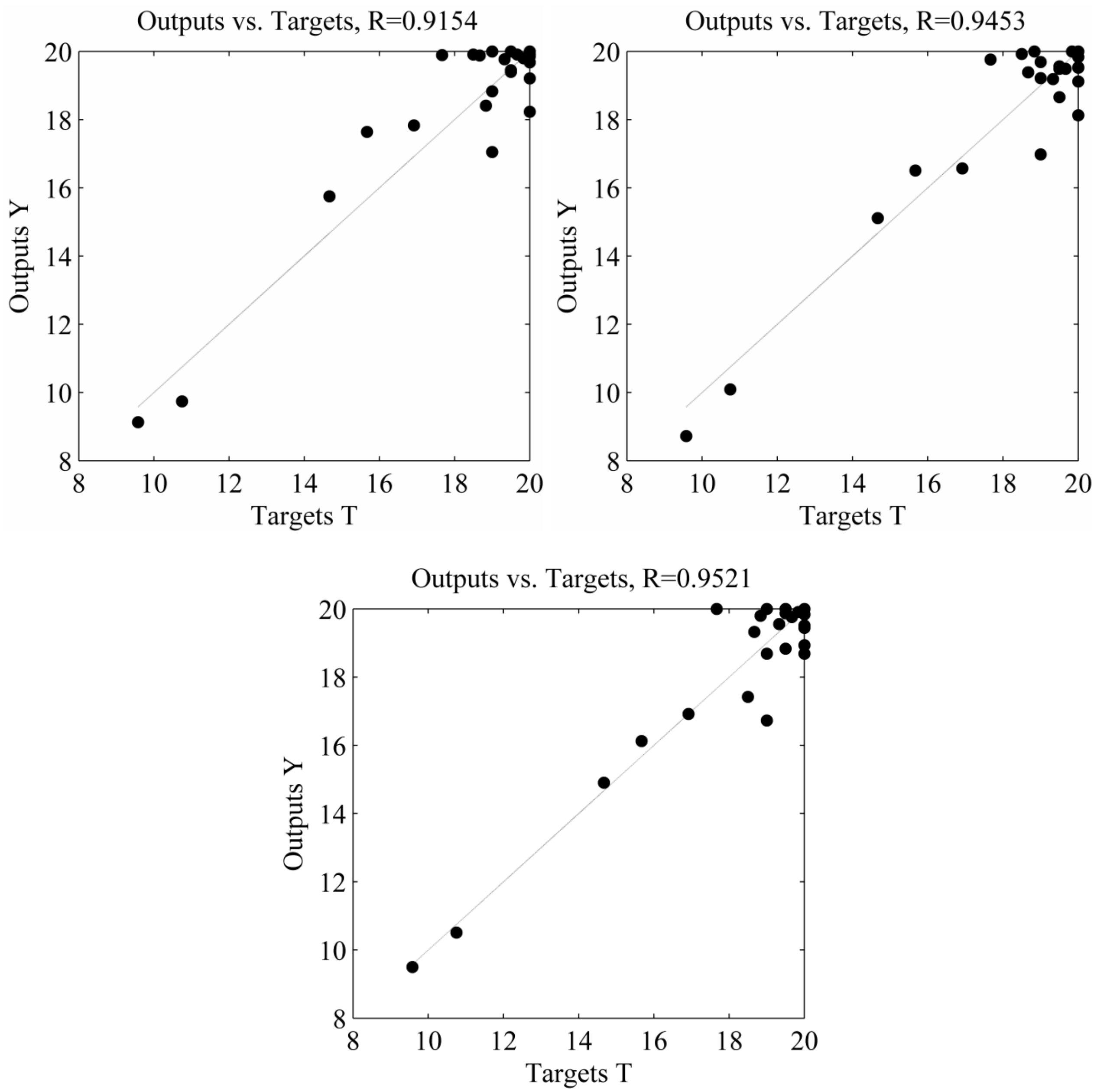


FIG. 3. (i)-(iii) Testing the neural networks

In the e-learning course under study, students with mediocre and low grades tend to drop out to re-attend the course in a following semester, while students who achieve high grades persist and successfully complete it. As a result, most of the available test data range between 16 and 20 in a scale of 20.

Next, six indicative network performance examples are presented in Table 1. Student 1 is an example of good student performance which was accurately predicted by the proposed method. This student scored 20, 19.67, 17 and 19 in the four multiple choice tests during the course and achieved a total score of 19.66 in the final test. All three neural network predictions of his performance were very close to the student's actual score and more specifically 19.91, 19.49 and 19.76. The second student, depicted in Table 1, performed moderately. This student scored 18, 17, 14.17 and 17.5 during the course and achieved a total of 14.67 in the final multiple choice test. In this case also, the proposed method correctly identified the student's performance estimating it to be 15.75 (NN1), 15.11 (NN2) and 14.9 (NN3). Student 3 is an example of weak learner performance. During the course, this student achieved scores of 16, 11.34, 13 and 12 and his score in the final examination was 9.58. In this case as well, neural networks were accurate in estimating the student's final grade to be 9.13 (NN1), 8.72 (NN2) and 9.5 (NN3). The proposed neural network method also managed to predict students with variations in their performance. For instance, student 4 scored well (20 and 18.67) in the first two multiple choice tests of the course. However, in the third multiple choice test the student's performance suddenly dropped to 12.5 and rose again to 16 in the fourth test. Despite the variations during the course, the student's final achievement was very good, 19.5. In this case, the first neural network (NN1) predicted the student's final achievement to be 19.4. Then, even if the student's performance dropped, the second neural network (NN2) prediction was also very accurate (18.66). The third network (NN3) predicted the student grade to be 19.87 which is also close to the real grade achieved. One may also observe that although the student's performance seems to deteriorate in the last two multiple choice tests, the predictions made by the neural networks were accurate throughout the course. Similarly, the neural network method estimated most student final grades accurately.

However, neural networks were not able to accurately predict specific irregular student performance examples. Table 1 depicts two such examples, namely students 5 and 6. Student 5 performed well, scoring 19, 20, 16.5 and 17 in the four multiple choice tests, but achieved 17.67 in the final test. Based on this good performance during the course, the student's mediocre final grade was difficult to predict. Subsequently, the final grade predicted by the three neural networks was 19.89 (NN1), 19.76 (NN2) and 20 (NN3). Student 6 demonstrates the opposite performance, achieving moderate test grades, namely 17, 16.67, 15.5 and 15, but received a high final grade 19. The three neural networks (NN1-3) in this case approximated the student's achievement to be 18.83, 16.98 and 16.73, receiving as input the first two, three and four multiple choice tests respectively. The student's sudden performance improvement in the final test was not consistent with his mediocre results of the first four tests and therefore could not be accurately estimated.

TABLE 1. Indicative student performance prediction examples of the Spring 2007 test set

	Student 1	Student 2	Student 3	Student 4	Student 5	Student 6
Mc1	20	18	16	20	19	17
Mc2	19,67	17	11,34	18.67	20	16,67
Mc3	17	14,17	13	12.5	16,5	15,5

Mc4	19	17,5	12	16	17	15
Final	19,66	14,67	9,58	19.5	17,67	19
NN1	19,91	15,75	9,13	19.4	19,89	18,83
NN2	19,49	15,11	8,72	18.66	19,76	16,98
NN3	19,76	14,9	9,5	19.87	20	16,73

As the results show, the proposed technique demonstrates a high degree of accuracy even in the early stages of the course. In e-learning courses, which lack face-to-face communication among the instructor and the students, the estimation of the students' performance level and consequently the learning actions suitable for each one of them is more difficult. The proposed method can help the instructor to efficiently estimate the performance level of each student, therefore reducing the instructor's workload and ensuring that the appropriate learning actions will be taken for the right student. These actions may include dynamically adapting the course material to the specific needs of each student as the course progresses, encouraging weak learners to increase their effort and motivating well-performing learners to further progress in their studies. Furthermore, the wide adoption of neural networks for both scientific and commercial purposes has led to a variety of tools available, facilitating instructors to easily implement and use the proposed FFNN method. These tools include NeuroDimensions NeuroSolutions, with add-ins for computational environments such as MS Excel or MATLAB (NeuroSolutions 2008), Stuttgart Neural Network Simulator (SNNS, 2008), Siemens ECANSE (ECANSE, 2008) and the Neural Network Toolbox of MATLAB (Matlab, 2007). Using the Graphical User Interface of one of the aforementioned tools, the instructor can train the networks easily and quickly (in the experiments mentioned in this study training time for each network did not exceed 1 minute). This training needs to be made only once, at the beginning of each semester, not increasing in consequence the instructor's workload significantly. Then, throughout the semester, he can use the trained networks to estimate the student performance of the course in progress both efficiently and quickly (experimental testing time did not exceed 1 second).

The method was also found to produce effective results without requiring a large data set. This was verified by the experiments conducted using Spring 2006 course data to train the neural networks and Spring 2007 data to test them. Thus, in case the course structure changes, predictions can be made again in a short period of time.

Although neural networks generally require some amount of time to complete their training, this is not considered to be a major drawback for the purposes of the specific application. More specifically, e-learning instructors need to train the networks only once, at the beginning of each semester, using previous course data. Then, to monitor student grades during the course, the trained networks can be used to quickly produce estimations.

However, during the experiments, the proposed technique failed to accurately predict the performance of certain students. This happens because the training set either lacks similar student performance examples or includes contradictory student examples. Nonetheless, as more data are collected and used to retrain the neural networks, at the beginning of each course semester, the method is expected to yield increasingly better results.

Comparison between Neural Networks and Multiple Linear Regression

In order to evaluate the network results, the proposed method was compared to the commonly used technique of multiple linear regression (Beck, 2000; Feng, 2005; Kotsiantis, 2004). Similarly to the neural network method, the first two, three and four Spring 2006 tests were used as independent variables to construct three multiple linear models (LR1, LR2 and LR3) to approximate the students' final achievement. Then, the calculated variable parameters were used to predict Spring 2007 final test grades. Linear regression estimations were conducted using the Statistics Toolbox of the Matlab 2007b platform. Both estimation and prediction time for this method did not exceed 1 second. Table 2, presents the multiple linear regression and the corresponding neural network results.

TABLE 2. Neural Network (NN) and Multiple Linear Regression (LR) results

	NN1	LR1	NN2	LR2	NN3	LR3
R	0,9154	0,8100	0,9453	0,7613	0,9521	0,7691
Mean absolute error	0,74	1,30	0,67	1,48	0,63	1,44

According to the results of the comparison between the NN and LR methods, neural networks performed better than multiple linear regression in each prediction stage in terms of both correlation coefficient (R) and mean absolute error. More specifically, compared to the multiple linear regression approach, neural networks were found to present higher correlation between their prediction results and the target final student grades. As more multiple choice tests are added to the methods, increasing the problem complexity, the correlation achieved by multiple linear regression tends to decrease (0.81, 0.7613, 0.7691), while the respective correlation achieved by the neural network solution increases (0.9154, 0.9453, 0.9521).

In addition, the mean absolute error of the neural network prediction method was approximately half the corresponding multiple linear regression error in all three stages of the process. This result indicates that the neural network technique was more efficient in mapping the nonlinearities that relate student performance during the course to their final achievement and thus provide a considerably more accurate estimation of final student grades.

As far as method performance time is concerned, the estimation of the linear regression parameters was faster (less than a second) than the neural network training phase (approximately 1 minute) per prediction stage. However, given that neural networks produce significantly better prediction results and that they need to be trained only once at the beginning of each course, their higher computational cost during training does not render them impractical. Finally, when making predictions based on new data during the course, both methods provided results at approximately the same speed.

Dynamic Student Clustering

After the completion of each multiple choice test, prediction results can be used to dynamically cluster students with similar abilities, in order to efficiently identify and meet the requirements of each group. As established by Feldhusen et al. (2004), clustering learners according to their ability is essential to help them achieve their optimum performance and further enhance their motivation to learn. On the contrary, non-homogeneous grouping may result in low student achievement and motivation. As soon as successful grouping takes place, a number of actions can be taken to further improve the learning process, as proposed by Oakley et al. (2004).

In the e-learning course under study, two virtual groups, A and B, are discerned based on the students' final achievement grades. The grade threshold T between the two groups was determined by the instructors to be 18.5 in a scale of 20, as the level of the course was introductory and it was thus considered to be relatively easy for the students. Therefore, a student is allocated to Group A if his final test grade exceeds T, otherwise this student is considered to belong to Group B. Student placement in each group was previously conducted during the course using the prediction results of linear regression. In this section, the proposed method is compared to linear regression in terms of student clustering efficiency and its results are discussed.

Table 3 depicts the correct and incorrect student placements made by both methods for groups A and B. Group A consists of 18 out of the total 25 students of the test set, while Group B consists of 7. Initially, NN1 achieved a relatively accurate student classification, as 15 out of the 18 Group A and 5 out of the 7 Group B students were correctly placed. LR1 on the other hand, had 1 more misplacement on Group B, but the number of the misplaced students in Group A was double. The performance of the neural network method further improved in the second prediction (NN2), where the number of the correctly placed students in Group A reached 16 while the respective LR2 estimation dropped to 10. Finally, NN3 fine-tuned the neural network method, correctly clustering 23 out of the total 25 participating students. LR3 efficiency in this case remained unchanged.

TABLE 3. Student clustering results of NN and LR methods on the Spring 2007 test set

Group	Total Students	Placement	NN1	LR1	NN2	LR2	NN3	LR3
A	18	Correct	15	12	16	10	17	10
		Incorrect	3	6	2	8	1	8
B	7	Correct	5	4	5	5	6	5

		Incorrect	2	3	2	2	1	2

The proposed method was more successful in correctly placing students to the group they belong than linear regression, even in an early stage. As the course progressed and students took more multiple choice tests, NN correct classifications gradually increased, while correct LR classifications decreased. Moreover, despite the fact that the two methods performed equally in determining group B, LR incorrectly added to that group many students from Group A, making it less homogeneous.

Non homogeneous grouping may result in providing less assistance to mediocre or weak students, obstructing the progress of students who perform well and increasing the instructor’s workload. On the other hand, accurate allocation of students into groups, based on their performance, can be beneficial to the learning process, since it leads to focused training to meet the needs of each group.

Figures 4 and 5 depict the students that the two methods placed in Groups A and B in every stage of their prediction. Figures 4 (i) and (ii) correspond to the NN method and figures 5 (i) and (ii) to the LR method. In these figures, four marks are shown for each student, one for each prediction and one depicting the student’s final test grade. A vertical line exists in every figure depicting the grade threshold T (18.5) between the two groups. This line also forms a barrier between the correct and incorrect predictions. More specifically, in figures 4 (i) and 5 (i), the marks placed on the right side of the line depict the correct placements in Group A, while the marks on the left side are incorrect predictions. The opposite happens in figures 4 (ii) and 5 (ii), where the correct and incorrect placements for each student in Group B are presented.

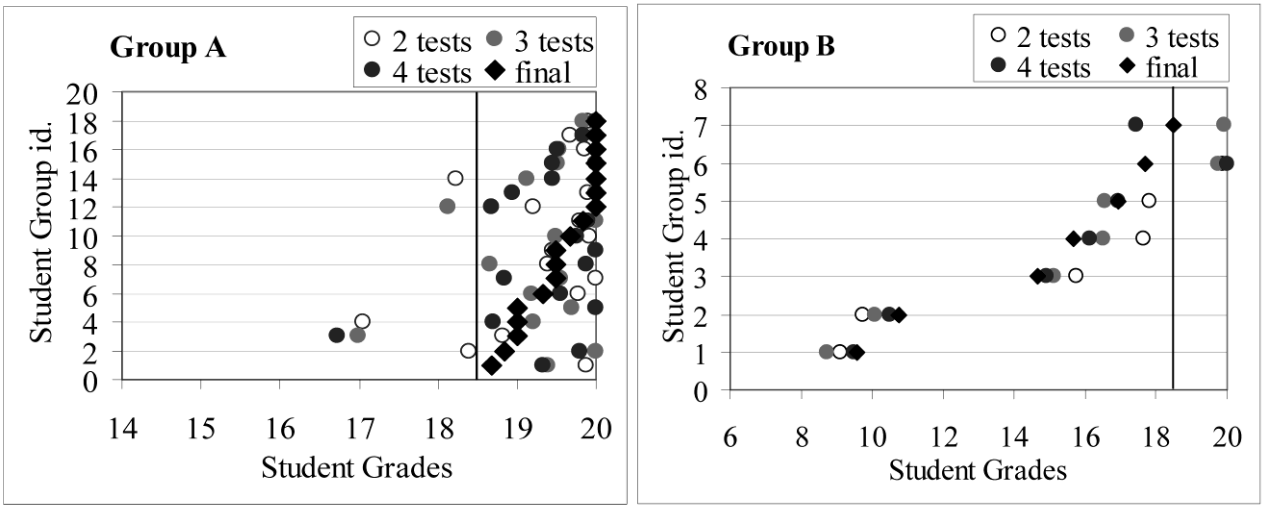


FIG. 4. (i)-(ii). Neural network placements in Groups A and B

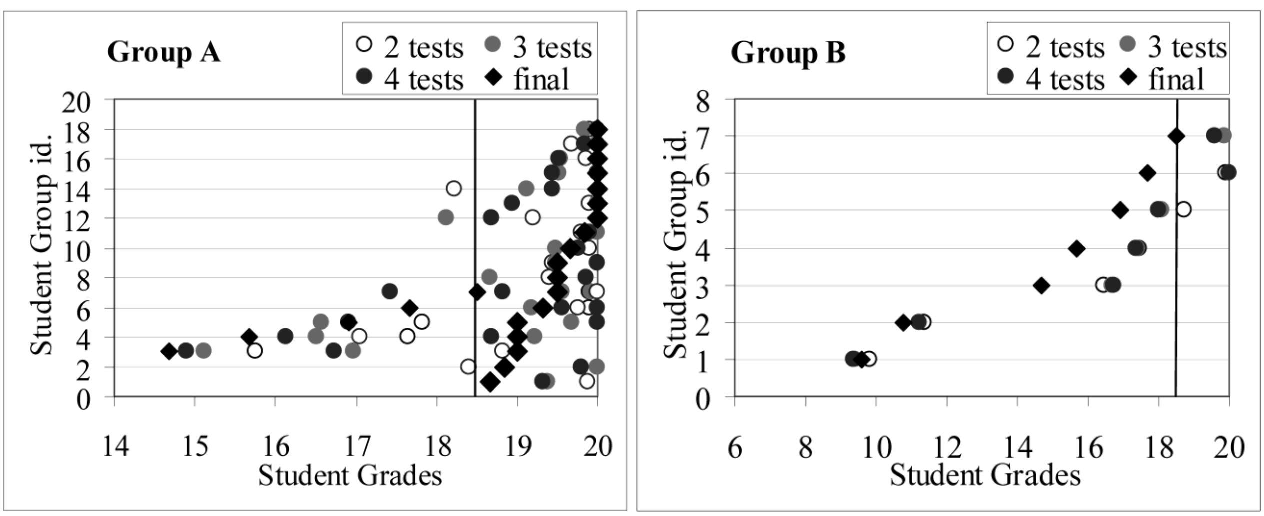


FIG. 5. (i)-(ii). Linear regression placements in Groups A and B

Neural networks correctly clustered the majority of the students for both groups, retaining low error values. On the other hand, LR had approximately the same classification results in Group B, but presenting higher grade estimation error. As far as Group A is concerned, LR had significantly more student misclassifications, in addition to a considerably higher prediction error.

Based on the student clustering analysis above, it can be concluded that according to the proposed method, accurate student clustering is possible in an early stage, more specifically, at the third week of the course. The results of this clustering technique can be used by instructors to determine the most appropriate learning actions for each group of students and provide them with further assistance tailored to their needs.

Conclusions and Future Work

In this study, three feed-forward neural networks are used to gradually predict the final grades of students, who took an introductory level e-learning course. The input data set of each network consisted of the results of the multiple choice tests completed during the course, while the network target output was the final multiple choice test completed at the end of the course. Multiple choice tests were used to ensure objectivity.

This method was then used to dynamically cluster e-learning students into two virtual groups, according to their predicted achievement, in order to help instructors address the specific needs of each group and adapt student training accordingly. Accurate student allocation into groups can lead to optimized student performance and enhanced motivation.

The results of the proposed method were compared against those of linear regression in terms of the correlation and mean absolute error achieved. Neural networks reached higher correlation at all prediction stages, while their error was approximately half the error of linear regression. Consequently, neural networks can make accurate grade predictions in an early stage, more specifically at the third week of a ten-week course. As far as dynamic student clustering is concerned, the proposed method was also found to be more efficient than linear regression. It presented lower failure rates, which decreased even further during the course, while the respective linear regression rates increased.

Concluding, the proposed method can be a useful tool for predicting students' final performance and dynamically grouping them, using multiple choice tests conducted during the course. It is therefore expected to provide an effective solution especially for e-learning courses, where instructors have fewer means of estimating future student performance due to the distance that this educational process entails. Future work will focus on testing the method on a greater number of students and incorporating it in the platform of a Learning Management System.

References

Andrews, R., Diederich, J., & Tickle, A. B. (1995). Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-Based Systems, 8*(6), 373-389.

Beck, J. E., & Woolf, B. P. (2000). High-Level Student Modeling with Machine Learning. In *Intelligent Tutoring Systems* (Vol. 1839/2000, pp. 584-593): Springer Berlin / Heidelberg.

Cripps, A. (1996, February 17-19). *Using Artificial Neural Nets to Predict Academic Performance.* Paper presented at the Proceedings of the 1996 ACM Symposium on Applied Computing, Philadelphia Pennsylvania, United States.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems (MCSS), 2*(4), 303-314.

Feldhusen, J., & Moon, S. (2004). Grouping gifted students: Issues and concerns. In *Grouping and Acceleration Practices in Gifted Education* (pp. 81-90): Corwin Press.

Feng, M., Heffernan, N., & Koedinger, K. (2005, July 9–13). *Looking for Sources of Error in Predicting Student's Knowledge.* Paper presented at the Proceedings of AAAI workshop on educational data mining, Pittsburgh, Pennsylvania.

Funahashi, K.-I. (1989). On the approximate realization of continuous mappings by neural networks. *Neural Networks, 2*(3), 183-192.

Hagan, M. T., & Menhaj, M. B. (1994). Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks, 5*, 989-993.

Haladyna, T. M. (2004). *Developing and Validating Multiple-Choice Test Items*: Routledge.

Haykin, S. (1999). *Neural Networks A Comprehensive Foundation* (2nd ed.): Prentice-Hall.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks (Vol. 4, pp. 251-257): Elsevier Science Ltd.

Hornik, K. (1993). Some new results on neural network approximation. *Neural Networks, 6*(9), 1069-1072.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks, 2*(5), 359-366.

Iyer, M. S., & Rhinehart, R. R. (1999). A method to determine the required number of neural-network training repetitions. *Neural Networks, IEEE Transactions on, 10*(2), 427-432.

Junemann, M. A. P., Lagos, P. A. S., & Arriagada, R. C. (2007). Neural Networks to Predict Schooling Failure/Success. In *Nature Inspired Problem-Solving Methods in Knowledge Engineering* (Vol. 4528/2007, pp. 571-579): Springer Berlin / Heidelberg.

Kalles, D., & Pierrakeas, C. (2006). Analyzing Student Performance in Distance Learning with Genetic Algorithms and Decision Trees. *Applied Artificial Intelligence, 20*(8), 655-674.

Kang, S. Y. (1992). *An investigation of the use of feedforward neural networks for forecasting.* Kent State University.

Karlovcec, N., Skala, T., & Saina, S. (2005, October 24-28). *Computer Science Education: Differences Between E-learning and Classical Approach.* Paper presented at the Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education, Vancouver, Canada.

Kotsiantis, S., Pierrakeas, C., & Pintelas, P. (2004). Predicting Students' Performance in Distance Learning Using Machine Learning Techniques. *Applied Artificial Intelligence, 18*(5), 411-426.

Matlab. (2007). *Matlab Environment*, from http://www.mathworks.com/products/matlab/

Medialab. (2007). *E-Learning Services, Multimedia Technology Laboratory, National Technological University of Athens*. Retrieved January 23, 2007, from http://elearn.medialab.ntua.gr/

Moodle. (2007). *Moodle LMS*, from http://moodle.org/

Oakley, B., Brent, R., Felder, R., & Elhajj, I. (2004). Turning Student Groups into Effective Teams. *Journal of Student Centered Learning, 2*(1), 9-34.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986a). Learning internal representations by error propagation. *Parallel distributed processing: Explorations in the micro-structure of cognition, 1:foundations*, 318–362.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986b). Learning representations by back-propagating errors. *Nature*, *323*(1), 533–536.

Sheel, S. J., Vrooman, D., Renner, R. S., & Dawsey, S. K. (2001). A Comparison of Neural Networks and Classical Discriminant Analysis in Predicting Students Mathematics Placement Examination Scores. In *Computational Science - ICCS 2001* (Vol. 2074/2001, pp. 952-957): Springer Berlin / Heidelberg.

Thrun, S. B. (1994). *Extracting provably correct rules from artificial neural networks* (No. IAI-TR-93-5): Institut for Informatik III Universitat Bonn, Germany.

Wang, T., & Mitrovic, A. (2002, December 3-6). *Using neural networks to predict student's performance.* Paper presented at the Proceedings of the International Conference on computers in Education, Auckland, New Zealand.

Zhang, G., Eddy Patuwo, B., & Y. Hu, M. (1998). Forecasting with artificial neural networks:: The state of the art. *International Journal of Forecasting*, *14*(1), 35-62.

Figure Legends

FIG. 1. The three neural networks used.

FIG. 2. (i)-(iii) Neural Network training

FIG. 3. (i)-(iii) Testing the neural networks

FIG. 4. (i)-(ii). Neural network placements in Groups A and B

FIG. 5. (i)-(ii). Linear regression placements in Groups A and B