Improving the expressiveness of black-box models for predicting student performance

Carlos J. Villagr-Arnedo, Francisco J. Gallego-Durn, Faran Llorens-Largo,
Patricia Compan-Rosique, Rosana Satorre-Cuerda, Rafael Molina-Carmona ✉
{villagra, fgalleo, faraon, patricia, rosana, rmolina}@dccia.ua.es

Department of Computer Science and Artificial Intelligence
University of Alicante
Carretera San Vicente del Raspeig s/n
03690 - Alicante
Spain

Abstract

Early prediction systems of student performance can be very useful to guide student learning. For a prediction model to be really useful as an effective aid for learning, it must provide tools to adequately interpret progress, to detect trends and behaviour patterns and to identify the causes of learning problems. White-box and black-box techniques have been described in literature to implement prediction models. White-box techniques require a priori models to explore, which make them easy to interpret but difficult to be generalized and unable to detect unexpected relationships between data. Black-box techniques are easier to generalize and suitable to discover unsuspected relationships but they are cryptic and difficult to be interpreted for most teachers. In this paper a black-box technique is proposed to take advantage of the power and versatility of these methods, while making some decisions about the input data and design of the classifier that provide a rich output data set. A set of graphical tools is also proposed to exploit the output information and provide a meaningful guide to teachers and students. From our experience, a set of tips about how to design a prediction system and the representation of the output information is also provided.

Keywords: black-box models, prediction, student performance, graphical representation.

1 Introduction

Improving student performance, knowing their actual progress and trying to predict their results at the earliest stages of the learning process can be extremely important to act early and cut off the problems at the root. These needs have triggered several research lines in different fields aimed to find ways to predict the learning outcomes.

An interesting research line is modelling automatic prediction tools based on statistical analysis or machine learning techniques. An automatic prediction system is a technological tool that, from a number of variables that refer to behaviour or activity of students in a learning system, obtains in advance an estimation of the final student performance. This is a very general definition, so that each specific prediction system may differ in certain aspects: the techniques used to build the system, the input variables to measure the behaviour or activity of students and the output variables that offer an estimation of students' performance. The quality of the prediction system can also be

measured in different ways: by focusing on purely metrical aspects (for example, the accuracy we get with the proposed method), or in terms of the expressiveness of the output to provide real help to the student.

Beyond the technical aspects of the model, the focus in this paper is on the expressiveness of the system. Modern educational theories advocate a student-centred training, with a truly formative assessment, and not just a classification. Therefore, for a prediction model to be really useful it should offer something more than a mere classification of students at the end of the learning process. It must also be able to provide tools to adequately interpret progress, to detect trends and behaviour patterns and to identify the causes of learning problems. This way, the prediction model will truly guide the students and detect the actual problems of the learning process.

There is a large amount of techniques to implement prediction models. In Learning Analytics, there are some techniques that require a priori models to explore (white-box techniques), and other more general techniques not requiring a priori models (black-box techniques). White-box techniques are easy to interpret since the a priori model provides a straightforward explanation of the relationships between data. However this prior knowledge integrated in the model reduces the search space and so it may introduce an important bias. Moreover, they are difficult to generalize, since they follow an ad-hoc design. On the contrary, black-box techniques are easier to generalize, are suitable to discover unsuspected relations and oriented to automatic unsupervised computation. Unfortunately, the technical aspects are quite cryptic for most teachers and they are very difficult to interpret.

In this paper the use of black-box techniques is proposed to take advantage of the power and versatility of these methods, while making some decisions about the design of the system that allows a richer interpretation. Specifically, the design and construction of a prediction model based on a Support Vector Machine (SVM) is used (Villagra-Arnedo et al., 2015). The decisions about the input data set design, the number and distribution of the classes, the use of probabilistic classification and the exploitation of time factor, provide a richer output. Moreover, a set of graphical tools has been proposed to exploit the expressiveness of the output information and provide a meaningful guide to teachers and students.

In section 2, a background of the research is presented, including previous related works, a discussion about the current research extent and what problems remain unsolved, and the research questions that guide the further decisions to design the proposed system. Section 3 is devoted to present the proposed prediction model, the design aspects, the results of the prediction and the representation of the output information. In section 4, some tips about how to generalize the results of the experience are described. Finally, some conclusions and further research are presented in section 5.

2 Background

Prediction can be defined as the inference of some information (the predicted or dependent variables) from a combination of other data (the predictor or independent variables). A prediction model is an analysis tool that obtains the predicted variables from a small sample of data, considering the statistical validity of the model so that it can be applied to the whole population (Berland et al., 2014).

There is a large amount of techniques to implement prediction models. Kotsiantis (2012) makes an interesting review of the use of these techniques for educational

purposes (classification and regression algorithms, association rules, sequential patterns analysis, clustering and web mining). This study identifies Machine Learning techniques as an emerging field that aims to develop methods of exploration of educational data and to find meaningful patterns. It specifically states that data collected from Learning Management Systems (LMS) or Intelligent Tutoring Systems (ITS) can be useful in developing prediction algorithms based on Machine Learning. It also points out that most research is about building ad-hoc models.

There are some techniques that require a priori models to explore (they are easier to interpret but designed ad-hoc), and other more general techniques non-requiring a priori models (suitable to discover unsuspected relations and oriented to automatic unsupervised computation but not so easy to understand). The former are known as white-box techniques and the latter as black-box techniques. The following paragraphs provide a brief review of some works about prediction based on white-box and black-box techniques in the field of education.

Macfadyen and Dawson (2010) develop a white-box system to predict students' performance using frequency of use and time spent on learning activities in a LMS. They perform a correlation analysis of several variables and the final grade, obtaining thirteen significant correlations. They admit that the predictive power of simple correlations used as a priori model is very limited, since the results show the obvious: most active students in the LMS end up being the most successful. Therefore, additional analyses are performed using multiple and logistic regression to obtain an accuracy of 70% in predicting students at risk of failure. Anyway, the need of an a priori model is an important limitation for the analysis.

Another example is the work of Pedro et al. (2013). They analyse and predict enrolment in a course using data from an ITS. Again an a priori model is used, in this case logistic regression, to predict whether a student will enrol in a course using a combination of representative characteristics about emotional state, motivation, knowledge and other usage data. After conditioning these variables by statistical analysis, they find a combination thereof to detect students to be enrolled with an accuracy of nearly 70%. The solution is designed completely ad-hoc for the problem.

Decision trees technique is a habitual white-box method used in learning prediction. For instance, Hu et al. (2014) develop a system for early prediction of final student performance. Learning portfolios are used as input to the algorithm. The authors determine the time-dependent variables of the learning activities of students in a LMS and collect activity data for three different moments of the course (weeks 4, 8 and 13). Fourteen features are used, grouped into four types: access behaviour, use of online course materials, task status and participation in a discussion forum. They use three classification techniques based on decision trees, obtaining an overall accuracy of 95% in week 4. Combining these techniques with Adaboost (Freund et al., 1996) the accuracy is increased to almost 98%. The prediction system prototype is made up of a set of decision rules that automatically trigger alerts based on the values of the most significant dependent time-based variables. Despite the high accurate results, the proposal is highly dependent on the specific case, so it is difficult to be extrapolated to other experiences. Nevertheless, the fact of performing a prediction in several moments opens a new exploring path for building progressive prediction systems that can be used to detect trends.

Another example of the use of a priori models is the one proposed by Ley and Kump (2013) to collect data on the interaction of six people in a period of two months. Its aim is to find out if interactions with a professional learning system integrated into a workplace can be used to predict three levels of expertise (beginner, advanced or

expert). They compare a task-based approach and a knowledge event-based approach. They use linear and logistic regression and conclude that combining both types of data as input improves the prediction significantly. The use of three classes instead of just a binary classification provides a more expressive model that can be considered in other research experiences.

Xing et al (2015) explore the development of more usable prediction models and representations. They use data from a collaborative geometry problem-solving environment to construct the prediction model. First, they link online learning with online participation, and operationalize activity theory to holistically quantify students' participation in the computer-supported course. As a result, 6 variables, Subject, Rules, Tools, Division of Labour, Community, and Object, are constructed. They make an interesting analysis of variables prior to the application of a model and, as a result, the data dimensionality is diminished and the data are systematically contextualized in a semantic background. They apply Genetic Programming as prediction model. As a result, the system has a high prediction rate and an interesting interpretability. Unfortunately the generalization of the prior model is quite complex, requiring a deep study of the problem to establish the implied variables. Moreover, the proposal is naturally biased by the selection of the variables, so that other important conclusions can be unnoticed.

Focusing on black-box systems, Hämmäläinen and Vinni (2006) make a complete review of Machine Learning methods for prediction. They conclude that Support Vector Machines (SVM) are particularly good for classifying educational numerical data (they achieve up to 80% accuracy), although they highlight the difficulty of selecting the specific parameters. Beyond the technical difficulties, they identify the lack of data as the main problem of educational applications and warn about other problems related to the difficult interpretation of the results: some systems are limited to the classification of the student's profile in only two classes, while many attributes are not susceptible to binary classification. A possible solution is considering additional classes or estimate probabilities.

Lykourantzou et al. (2009) propose a prediction method using feed-forward neural networks to predict the final grade of students in an e-Learning course and classify them into two classes based on their performance. They use the answers to multiple-choice questions as input to the algorithm. Their results show that an accurate prediction is possible at an early stage (in the third week). However, the proposed method fails in predicting students' performance in specific cases and demonstrates the difficulty of interpreting the results.

Random Forest (RF) (Breiman, 2001) is a black-box technique that is quite widespread among research on prediction. For example, Petkovic et al. (2012) study the lack of objective assessment methods in teamwork. They apply an approach based on RF to train a prediction model for group activity, classifying factors and rules for assessing the effectiveness of teamwork. Schalk et al. (2011) construct a predictive system to identify students who are at risk of failing in introductory Mathematics and Physics courses. They also choose RF to model a prediction system that uses the data from previous Standard Administration Tests (SAT). However, it is very complex to explain the relationship between input and output.

Another interesting example is that of Huang and Fang (2013), where the authors compare four models of Machine Learning (belonging to both white-box and black-box categories), to predict student achievement, using some data such as a measure of the ability to solve problems and the learning outcomes in the form of grades. They conclude that to predict the average academic performance of a group, the simplest

linear regression model is the best choice, but to predict the performance of an individual, the best method is Support Vector Machines (SVM). As a conclusion of this study, a priori models (such as linear regression) should be avoided when complex relationships are present.

The explanation of black-box models has also sparked the interest of researchers in several fields. For instance, Palczewska et al. (2013) present an approach for computing feature contributions for random forest classification models, so that they determine the influence of each variable on the model. This influence is a source of explanations for the model. Although their results are very interesting, their contribution is only applicable to random forest techniques.

Another interesting work is the one of Rosenbaum et al. (2011). They use linear support vector machines to guide the optimization of a compound in some drug discovery stages. They consider that SVM have a convincing performance on large-scale data sets for this purpose, but they found them difficult to interpret. The goal of this study is to present a heat map molecule colouring technique to interpret linear SVM models. Based on the weights of a linear model, the visualization approach colours each atom and bond of a compound according to its importance for activity. The heat map colouring assists to determine the correct ligand orientation and enables the identification of substructures.

Baehrens et al. (2010) also try to shed light on the black boxes of nonlinear classifiers. They introduce a method that can explain the local decisions taken by arbitrary nonlinear classification algorithms. They use local gradients that characterize how a data point has to be moved to change its predicted label. For models where such gradient information cannot be calculated explicitly, they employ a probabilistic approximate mimic of the learning machine to be explained.

From this literature review several conclusions can be obtained. Kotsiantis (2012) points out that both white-box and black-box methods have been used to predict student performance, although most research is about ad-hoc models. In general, white-box techniques are designed ad-hoc for each case and they are difficult to be reused or adapted to new scenarios. Macfadyen and Dawson (2010) and Pedro et al. (2013) consider that the use of a priori models is an important limitation. White box techniques are very suitable for simple cases, particularly when there are linear relationships between variables, but they are difficult to be extrapolated to other experiences, as pointed out by Hu et al. (2014) and Xing et al. (2015). However, interpretation of white-box methods is usually quite straightforward, since the underlying models are known.

Black-box techniques are usually more accurate (Hämäläinen and Vinni, 2006; Lykourantzou et al., 2009), particularly when there are complex relationships (Huang and Fang, 2013), and easier to apply, since they are more general. Interpretation of black-box methods, however, is much more complex (Breiman, 2001; Lykourantzou et al., 2009; Schalk et al., 2011), since the underlying models are too complex to be easily explained.

Nevertheless, there are some examples of interpretation of the results of black-box predictors, mainly based on the study of the influence of each variable on the model (Palczewska et al., 2013) and the use of graphical interpretations such as heat maps (Rosenbaum et al., 2011) or local gradients (Baehrens et al., 2010). These works open up new possibilities of interpretation that are explored in this paper.

2.1 Discussion and research questions

Accuracy is just a measure of the capability of a given system to correctly classify an individual in the category it belongs to. In other words, it explains how good is the adaptation of the model to the given data and the specific problem. However, accuracy is neither a measure of the capability of the system to explain the relations between variables and to interpret the results nor even a measure of the capability of the model to generalize the results. Although important from a technical point of view, we are interested in providing an expressive prediction system to help people to face human problems; particularly, to guide students in their learning process.

Beyond the discussion about accuracy, traditional methods of learning analytics, based on statistical systems, need predefined models that explain the relationships between variables. Although knowing exactly the relationship between input and output variables can help develop strategies for learning aid, these methods have major drawbacks. On one hand, they are ad-hoc models that require a very thorough study of the particular learning process, so they are not easily generalizable, requiring rethinking each case and reformulating the prediction model. On the other hand, they are a priori models in which the explanatory model is set in advance (linear and quadratic models are very typical in education research), so they do not to explore other potentially useful models that could establish other relationships between data.

Black-box models, however, are much more generalizable and thus easier to adapt to each particular case. In addition, since they do not require the determination of a priori hypotheses, they are useful for exploring all types of relationships in an unlimited search field where unsuspected connections may appear. These models, however, are not easy to interpret: it is possible to identify what outputs are obtained for each input, but not why the connection is produced. The difficult interpretation of these models has led some researchers to discard them as tools to assist teachers and students. However, from our point of view, the problem of difficult interpretation can be solved by a number of strategies help understand the learning process without the need to interpret the underlying mathematical model. In fact, other authors have already identified the model interpretation as one of the key aspects for model evaluation. There have been some attempts to explain black-box models but most of them are not general for any model.

In this paper we propose the following research questions: Is it possible to propose an expressive interpretation of black-box systems for predicting student performance? How can the system be designed for providing this richer interpretation of the predictions?

3 Proposed model

Since the proposed system is set to be an aid for students and teachers, its results must be prepared for human interpretation. The aim of this work is, therefore, designing a powerful prediction model and leveraging the rich catalogue of data visualization methods to display the model output and support students and teachers in their decision-making.

3.1 Context

This experience is developed in the context of a subject about Computational Logic in Computer Engineering Degree. All the learning process is performed using a web site

that allows the interaction of students and teachers. Other than regular user management, the web provides several options for the students (download exercise statement, upload exercise solution, obtain exercise score, access their learning progress...) and for the teachers (upload new exercise statements, monitor the progress of their students...).

The core elements in the learning system are the exercises. Specifically, the system in its current version deals with logics problems, consisting on the solution of mazes of a game called PLMan (Villagr-Arnedo et al., 2009). In this game, students program the artificial intelligence of a Pac-Man like character, using a Prolog knowledge base. For each maze, the students have to program a set of rules so that the character can get all the dots, dodging the perils.

Mazes in PLMan are organized into 6 main stages (stages from 0 to 4 and a final checkpoint). The stages are sorted by increasing difficulty, so that as one moves from one stage to the next one, new knowledge about the programming language is required to pass the level. To progress from one stage to another, a minimum score is needed. Students have to beat the 6 stages to get the maximum grade. At each stage, students have to solve 1 to 5 different mazes (5 mazes at stage 0; 3 mazes at stages 1 and 2; 2 mazes at stage 3; 1 maze at stages 4 and checkpoint). Although the required knowledge and skills to solve every maze in a stage are the same, there are different levels of difficulty for the different possible mazes. To get each new maze for solving, students start by picking up their desired difficulty level among valid levels for the stage in which they are. Then, the system presents them with a random maze from their selected level of difficulty. Mazes can only be assigned once, so the same maze cannot be assigned to two different students.

Every possible solution can be delivered to the web system. The automated system runs the solution and evaluates its score and grade, mainly based on the total proportion of dots their PLMan character gets. The students get immediate feedback of their results and can redo the exercises to get a better grade. During this process, a large amount of data is collected both from the student interaction (usage or behavioural data) and from the student progression in the exercise resolution (learning data). All these data are stored in a database to be used for further analysis.

3.2 Data collection

During user interaction with the system a large amount of specific events are occurring. To take advantage of the flow of data that goes in and out the system, it is crucial to transform them into valuable information that can give advice for teachers about progress of the students. Building some informative characteristics well correlated with student performance and evolution is crucial to get good prediction results, so the first step is designing which information would be valuable to log (Villagra-Arnedo et al., 2016). Although the meaningful data depend on the specific problem, we consider that this feature set must consist of both behavioural data (data related to system usage) and learning outcomes (data related to exercise results).

In the particular case presented in this paper, a list of specific events that occur during the interaction between students and the system was considered (Villagra-Arnedo et al., 2016). All these events (see table 1) are logged in an event database, with their appropriate timestamp and related information. They made up the available behavioural data. Moreover, to make up the set of learning data, all learning results from these events were also considered: submission results, submission attempts at each maze, mazes solved (obtained percentage), time to solve each maze, etc. These learning

outcomes are qualified as potential predictors because of their intuitively logical relationship with the final performance.

Table 1. Logged events

Event	The student...
show_frontpage	enters in the system and sees the main status page
show_results	sees his or her results on a specific maze
select_difficulty	selects the difficulty he or she wants for a new maze
maze_download	downloads a maze
solution_submission_ok	submits a solution which compiles and executes ok
download_logs	downloads execution logs (used to replicate AI bugs)
solution_submission_error	submits a solution that does not work properly (with compilation errors)

All this information about the student activity is the primary matter to construct the set of features that make up the input of the predictive system, based on a Machine Learning algorithm.

3.3 Prediction model

The function of the predictive model is to make predictions of the students' grades at the end of the term, based on the data collected from the interactive web. These data, selected and organized into features, are normalized and given to a machine learning algorithm as input.

The predictive system classifies the student expected performance (measured as a grade in percentage) in three possible classes:

- HP: high performance (expected grade $\geq 80.5\%$).
- MP: medium performance ($57.5\% \leq \text{expected grade} < 80.5\%$).
- LP: low performance (expected grade $< 57.5\%$).

The reason to split the output in three classes is to get an adequate performance out of the classification algorithm, adjusted to the size of the sample (the sample has only 336 students, so three equilibrated classes of 112 individuals are proposed). The intervals have been chosen a posteriori, to get a balanced distribution of the students in the grade range.

The prediction system is based on a standard C-parameterized margin, SVM classifier (C-SVC) (Cortes and Vapnik, 1995) with Radial Basis Function (RBF) kernel (Vapnik, 1998, 1995), adding probability estimates using Pairwise Coupling (Wu et al., 2004). This is a very effective and efficient Machine Learning algorithm that works very well for general datasets like the one under analysis, as Huang and Fang (2013) and Hämäläinen and Vinni (2006) pointed out.

The proposed system has been implemented in the past term for a first year university course. There were around 400 students enrolled in the course, 336 of which finally participated in the experiments. Although the complete term consists of 15 weeks, only 11 weekly lessons are considered since the first 4 lessons are introductory. Consequently, 10 weeks of predictions are given, and then the system closes giving final grades to students in week 11.

For each week, all the information in the event database described in table 1 is processed to get up-to-date features describing students. A specific set of features is selected for the experiment, summing up 61 features (see table 2) (Villagra-Arnedo et al., 2016). The features are scaled to [0,1] range before passing them as a vector to the SVM, preserving the many values that are already 0. So, the SVM gets this feature vector as input, and a vector of 3 probabilities as output, which effectively gives the probability of a student being classified in each one of the 3 classes (HP, MP or LP).

Table 2. Input features

Features	Values
Number of frontpage visits	1 value
Number of maze downloads	1 value
Number of submissions per stage	6 value (1 for each stage)
Accumulated score	1 value
Average scores (percentage) per stage	6 values (1 for each stage)
Score (percentage) per maze and stage	15 values (1 for each maze of each stage)
Time (seconds) to finish each stage	6 values (1 for each stage)
Time (seconds) to finish each maze and stage	15 values (1 for each maze of each stage)
Difficulty levels selected per maze and stage	10 values (1 for each maze of each stage but the first)
Total	61 values

With the scaled features, a model selection is conducted to boost the performance of the SVM by picking up the best set of parameters. Since two parameters have to be selected (C regularization parameter, and γ amplitude of RBF), a 2D grid-search is performed, with parameter values varying exponentially in powers of 2 (Lin et al., 2003):

$$C \in \{2^i : i \in [-15, 15] \cap \mathbb{Z}\}, \gamma \in \{2^j : j \in [-15, 5] \cap \mathbb{Z}\}$$

For each pair of parameters, a leave-one-out Cross-Validation training gives its accuracy estimation. Finally, the two parameters corresponding to the best accuracy estimation are selected (Villagra-Arnedo et al., 2016).

After selecting the best set of parameters, 10 SVMs are trained, one for each week being considered. This improves expected performance by having specialized SVMs for each week. It is important to clarify that data used to train SVMs has to be from past, closed terms, because knowing the actual final performance achieved by students is required to train SVMs. Once SVMs are trained, they can be used to do real-time predictions for an ongoing term. These predictions are added to the prediction stack, where all predictions for past weeks are stored. Finished this prediction step, graphs and information are elaborated and presented to teachers for student evaluation. The last value in the weekly prediction stack always correspond to the week in course, whose predictions are made using its corresponding SVM as if the week had passed.

3.4 Accuracy results

Two experiments have been performed. In the first experiment, all the 61 features are used for training the system and predicting the performance. In the second experiment, only some features have been used for training. An a priori model, lineal correlation between every feature and the final grade, has been introduced, so that the most correlated features have been selected, assuming that a higher correlation implies a higher significance of the feature. There are several coefficients to measure the correlation degree. In this case, the most suitable is that of Spearman (Szczepeńska, 2011), due to the fact that some variables are discrete and the continuous ones may not be normal. The Spearman coefficient is calculated every week between every feature of experiment 1 and the final grade of the students. From the study, it can be concluded that, in general, learning features have a higher correlation than behavioural features. In fact, there are 7 learning features and only 1 behavioural feature with a high correlation coefficient (Spearman coefficient > 0.7). For this experiment, every week we have selected the 23 features with the higher correlation out of the total of 61, which can be

different for every week. The number of features (23) has been chosen so that no feature with at least moderate correlation (Spearman coefficient >0.4) is out of the sample.

As a measure of performance for the classifier, a pure classification technique has been used. Every classifier (one for each week), for every test sample gives three probability values as a result, that is, it gives the probability of the student of belonging to one of the three possible classes (high, medium or low performance). To compare the results of the experiments, the average accuracy of the classifiers of every week is obtained as the proportion of well-classified samples, considering that the sample is finally classified in the class with a higher probability. The accuracy values for both experiments are shown in table 3 and figure 1.

Table 3. Accuracy results

Week	Accuracy experiment 1 (61 features)	Accuracy experiment 2 (23 features)
1	0,46429	0,44345
2	0,47619	0,44643
3	0,51786	0,54464
4	0,55357	0,54762
5	0,55655	0,56250
6	0,60417	0,54464
7	0,63691	0,59524
8	0,66071	0,62798
9	0,67560	0,66964
10	0,74107	0,72619

Figure 1. Accuracy comparison for experiments 1 and 2.

From the accuracy results, it can be deduced that the best results are obtained for experiment 1, when all the features are used to predict. Although both experiments have very similar results, experiment 1 is the one showing a lower variance (the curve is smoother, so the classifier is more robust). This is an evidence that the use of a priori models may introduce a bias. Reducing the number of features used by the SVM prediction algorithm does not improve the results of prediction, even in the case of eliminating the features that are supposed to have no significant correlation with the

final grade. This result is due probably to the fact that the SVM algorithm is detecting non-linear relationships, which are lost by reducing the search space.

The results of accuracy corroborate what intuition tells: the more data, the better prediction. The results are modest (around 70% the last weeks). This is probably due to the class division: classes are discrete, whereas the grade is defined in a continuous space $[0,10]$. Thus, the consistency of the classification for samples that are on the border of two classes is much lower. This idea is reinforced if the class with the second higher probability is considered: in the vast majority of cases where winner class is wrong, the second option is the right with a very narrow error margin (typically <0.05). A detailed analysis of these results is beyond the scope of this paper. Moreover, these results of accuracy do not affect the main objective of the research.

3.5 Tools for result interpretation

For the teachers to provide meaningful feedback to students, the results of the prediction model must be easily interpretable. Nevertheless, simple tables of probabilities and values of accuracy are not expressive enough to provide a real help. In the following paragraphs, an explanation of the strategies used to represent and interpret the results is provided.

When designing the prediction model, the further interpretation of the results was taken into account, so that the best expressivity could be achieved. Three features have been considered in the design of the classifier: multiclass, probabilistic and progressive.

The first decision that helps to provide more expressivity is the use of a multiclass classifier instead of a dichotomic one. A multiclass classifier is usually a better choice when modelling a continuous space, since the more classes, the better discretization is obtained. In this experiment, the size of the learning set limits the number of classes, but three classes is expressive enough for this problem.

The design of the model integrates a second decision that increments the interpretability of the results: the use of probabilistic classification. True/false classification does not provide any degree of certainty about the classification. The additional information given by the value of probability provides a measure about the strength of the classification, and can be used to construct graphical tools that help in the interpretation of the results.

The third feature that is crucial for the interpretability is progressiveness. The system is design to obtain a prediction every week, so that trends and accumulated factors can be used to monitor the progression in the learning process and help the students when the risk of failure is detected.

In summary, the fact of using a progressive multiclass probabilistic prediction model provides much richer information that can be used for guiding the learning process. The following step is designing a set of representation tools that enhance the rich information with an expressive graphical output.

A first approach to the expressive representation of the results is the progression chart. A progression chart is the representation of the weekly prediction, showing both the probabilities and the final classification. A progression chart is obtained every week, adding the results of the current week to the prediction of the previous ones. As weeks pass, the progression chart is more and more meaningful, allowing the detection of trends. In figures 2 and 3, two progression charts belonging to the same student in weeks 4 and 10 are presented. Each area represents the probability of being classified as a HP (green), MP (gray) or LP (red) student. The circle indicates the final classification (given by the position in the graph and the colour) and the confidence of the

classification (given by the circle area, representing the difference between the probabilities of the class with the highest probability and the second one).

Figure 2. Progression chart for an individual student at week 4.

Figure 3. Progression chart for an individual student at week 10.

Progression charts can represent individual or group results. In the case of individual charts, it allows the detection of risk of failure of a student, and the effect of guiding. For instance, in figure 3, the student is classified as low or high performance in the first five weeks, with a low or medium level of confidence. However, the prediction of belonging to high performance class is dramatically reinforced from week 6, when the student reacts, so the performance results become much better in the last weeks.

Another interesting use of progression charts is for studying group results (figure 4). In this case, the chart represents the average probability for every class and every week, considering all the students in a given group. The interpretation is similar to that of the individual charts, but in this case trends about the whole group can be detected. Besides the comparison between groups, it is very interesting comparing the progression of the whole group in accordance to the learning plan. The chart in figure 4 is completed with a timeline representing the milestones in the learning plan, that is, deadlines and other events. It can be clearly detected that deadlines produce an increment in the work of

students, so that an improvement of the expected performance is produced just after the deadline.

Figure 4. Progression chart for a group at week 10, including the timeline with stages and milestones (MS).

Another possible tool is the use of heat maps. Heat maps are graphical representations of data in a matrix of rectangular tiles, using a colour scale to represent the value of the corresponding element of the data matrix (Wilkinson and Friendly, 2009). An ordering relation must be defined for rows and columns so that similar individuals are near each other. It has proven to be a useful tool to represent large sets of data that are difficult to interpret as numerical values, particularly in studying the performance of students' groups (Bowers, 2010).

When constructing heat maps, there are three main aspects to determine: the data to represent, the colour scale and the rows and columns order. In our case, the data to be shown are the predictions for all the students in a group during the term, so the columns are representing the current week and the row the predictions about a given student. The colour scale has been chosen to be consistent with the previous graphs: green for HP class, gray for MP class and red for LP class. The column order is quite obvious: an increasing chronological order, from week 1 to week 10. In the case of rows, the ordering is not so simple, so an order relation must be defined. A possibility is using some kind of clustering algorithm (Bowers, 2010) but in our case a simpler algorithm was enough to properly order the individuals. Specifically, considering the 10 weekly classifications for each student, two possible ordering criteria are considered (see figure 5): in the first one, "number of classifications criterion", students with a higher number of classifications as HP are first and, in case of equality, the number of classifications as MP is considered; in the second one, "last week criterion", the classification in week 10 is considered first, with students classified as HP on top and, in case of equality, previous weeks are considered. Results are very similar, and both orderings are suitable for further analysis.

A heat map is a compact and integral representation that can be useful for large groups since trends and patterns are easier to detect, and allows the comparison of different groups. The use of central and dispersion measures (such as mean and standard

deviation or median and interquartile range), give only an overview of the central tendency of a sample, so that the comparisons are not so meaningful.

Figure 5. Heat maps for a group of 50 students, using two different orderings: number of classifications criterion (left) and last week criterion (right).

In heat maps with three classes, like these, three bands can be identified, corresponding with the distribution of the classes (figure 6). Two parameters can be considered to study the general distribution of the classes:

- Band heights: They are determined by the number of individuals classified in every class. The best case occurs when the band width for HP class is as wide as possible, meaning that most students are classified as HP.

Figure 6. Bands and parameters of a heat map, corresponding to HP class (green), MP class (grey) and LP class (red).

- **Band slope:** It explains how much the classification results change from one week to the next one. In the extreme, vertical bands mean that there is a week when the classification changes dramatically, while horizontal bands imply a very robust classification from the beginning. If the system is correctly guiding the student learning process, bands should be as vertical as possible, meaning that the guiding tool is good at helping students to go from one performance class to a superior one.

Some other information can also be added to heat maps so that they can be even more expressive. The “confidence heat maps” include the confidence of the classification, that is, the difference between the probabilities of the class with the highest probability and the second one, represented as the circle area in the progression charts. To do so, confidence in the classification is divided into three different levels (just dividing in three intervals of the same size) assigning a more saturated colour for a higher level of confidence, but maintaining the colour hue. In figure 7, the confidence heat map using last week criterion is presented.

Figure 7. Confidence heat map for a group of 50 students, using the last week criterion.
Colour hue identifies the class and colour saturation the confidence level

The confidence heat maps adds the saturation as a measure of the confidence of the classifications. In most cases the accuracy of the prediction system increase in the last weeks, so the colour of the confidence heat maps reflects this increment by becoming more saturated as time passes by.

4 Some tips for designing meaningful prediction systems

Black-box systems are, by definition, difficult to interpret. The underlying mathematical models are complex and this is their main interest, since they can exploit complex relations between data. It may not be worthwhile trying to explain the model. Instead, a richer design of the system output can provide a lot of interesting information. From the study of the previous research and our experience, we dare to provide some tips about the prediction system design and the representation of the information.

Avoid cooking the data. Black-box models have the main advantage of not limiting the search space, so that no a priori models are needed, and no prior knowledge is added to the solution. It is convenient to exploit this feature and avoid limiting the search by selecting the data for the sake of reducing the data set size or eliminating noise: the result is that unexpected relationships may be gone with the noise and a bias is introduced. Moreover, if the sample is cooked to avoid noise, an artificial overfitting is introduced, with the consequent loss of generality. This fact can be observed in our case: experiment 2 eliminates data with little linear correlation at the expense of reducing accuracy because of other types of non-linear correlations.

Multiclass classifiers are preferable for continuous spaces. Unless the problem is naturally a dichotomic problem, avoid binary classifiers. A multiclass classifier will be more expressive than just using two possible classes, specially if discretizing a continuous space like that of numerical grades. Several authors have already detected the need to use several classes to be more expressive (Ley and Kump, 2013) or because of the nature of the attributes (Hämäläinen and Vinni, 2006).

Choose an adequate number of classes, depending on the sample size. The approximation to the problem in this paper is somehow funny but very common in learning contexts. The canonical problem is choosing an adequate sample size depending on the number of classes to be classified. This is the adequate approach when classes are well defined and there are no limitations to obtain as many samples as needed. But in other contexts, like ours, classes are not defined (the possible range of results is not naturally divided into intervals, for instance, student performance in terms of final grade, in a continuous space) and the amount of samples is limited. In this case, the problem is how to separate the complete space into intervals so that a good classifier is obtained. When the results are defined in a continuous space, it can artificially be divided into discrete classes but such conversion causes a loss of information. Since an infinite number of classes would ideally avoid this problem, our approach is dividing the space into as many classes as possible, limited by the size of the available sample. It is very difficult to give a general rule on the appropriate sample size since it depends on the signal-to-noise ratio and the complexity of the model, but several studies about this question can be found in the literature (Hastie et al., 2009).

Choose appropriate limits for the classes. Depending on the problem, it may be advisable to discretize the continuous space into intervals (each interval corresponding to a class) of the same size or not. If the interval size is not necessary to be equal, a possible solution is dividing the solution space in classes so that a similar amount of samples belongs to each class.

Probabilistic classification is more expressive than pure classification. Classifiers that are able to predict a probability distribution over the set of classes are much more meaningful than classifiers outputting the most likely class, since the first ones provide a degree of certainty along with the classification. This fact has already been pointed out by other authors such as Hämäläinen and Vinni (2006). The probabilistic predictions are

important for decision making because of the much richer knowledge about the problem provided by the additional information of classification “strength”.

Time is a factor to exploit. The possibility of using time as a feature to include in the prediction model must be considered. The use of time can help the detection of trends, the influence of accumulated factors and an actual study of progression, as indicated by Hu et al. (2014). If predictions can be made in different temporary moments the comparison with previous predictions will help to detect trends that cannot be detected with a single prediction in a given moment.

Use graphical tools to present the data. Graphical tools exploit the expressiveness of data. The use of charts of several types may help to provide a deeper and wider representation of the problem situation. Time and progression are very important in prediction systems so graphical tools should take advantage of it. An example is progression charts, which allow the detection of trends and inflection points. Moreover, they can be used for individual predictions (to detect students at risk of failure) or to deal with whole groups, representing the average behaviour of the group (to study more general trends). An interesting use of group progression charts is comparing the behaviour of the students in the group with the course planning, identifying the effects of the deadlines and the planned milestones in this behaviour. Since central and dispersion measures imply a loss of information about the distribution of the data in the group, it is interesting to provide other types of graphical representations that present all the data in an interpretable manner. For example, heat maps, already used by other authors such as Rosenbaum et al. (2011), provide much richer information to compare group behaviour and detect patterns and distribution of the individuals in the whole group. Finally, the prediction confidence can be used to develop other more complete graphs, such as the confidence heat maps.

These tips are not claiming to provide an exhaustive or mandatory list of desirable features. They are just a summary of the lessons that we have learned from our experience.

5 Conclusions and future work

We proposed two research questions to be answered in this paper. The first one was: Is it possible to propose a more expressive interpretation of black-box systems for predicting student performance? Black-box models usually make use of cryptic algorithms whose interpretations are out of the understanding of most teachers, but they can still be very interesting to explain phenomena in the field of education. From our experience, the way to explore is to design an adequate output for these systems and to exploit the graphical tools to add expressiveness.

The second research question was about how black-box systems can be designed for providing the desirable richer interpretation of the predictions. To give an answer to this question we have designed a prediction model for students' performance, proposing a set of features that make the system actually expressive and that could be useful for guiding the learning process. Special care has been taken when designing the experiments, including aspects such as the definition of the data to be collected, the algorithm to be used for the classification, the number and range of the classes, the output to be obtained, and so on. Besides, a set of graphical tools has been proposed to provide rich information of several types, which allow the detection of trends and behavioural patterns.

Finally, for this experience to be useful for further developments, a set of tips is proposed. This is an open set made of the main lessons learned while designing prediction systems that is subject to discussion and completion in the future. Precisely, this is one of the ways to follow in further research: improving the system, identifying new features that help to obtain more meaningful models and designing new graphical tools to exploit the richness of data.

Another future step in this research is analysing how to use this information to intervene in the learning process of the students. Knowing their actual progress, trying to predict their results in the earliest stages of the process, and correctly interpreting the causes of the possible failures, can be extremely important to early intervene and tackle the problems at the root.

The extension of the interpretation tips and tools to other context than education is another path to explore. Prediction systems can be used in any field to assist humans in decision-making. A reasoned and informed decision-making is based on the ability of the person or organization to anticipate events, collect meaningful data, analyse and transform them into complete and useful information. A major step forward will be extending this contribution to other disciplines to improve their decision processes.

6 References

Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., Müller, K.-R., 2010. How to Explain Individual Classification Decisions. *J. Mach. Learn. Res.* 11, 1803–1831.

Berland, M., Baker, R.S., Blikstein, P., 2014. Educational Data Mining and Learning Analytics: Applications to Constructionist Research. *Technology, Knowledge and Learning* 19, 205–220.

Bowers, A.J., 2010. Analyzing the Longitudinal K-12 Grading Histories of Entire Cohorts of Students: Grades, Data Driven Decision Making, Dropping Out and Hierarchical Cluster Analysis. *Practical Assessment, Research & Evaluation (PARE)* 15, 1–18.

Breiman, L., 2001. Random forests. *Machine learning* 45, 5–32.

Cortes, C., Vapnik, V., 1995. Support-Vector Networks. *Machine Learning* 20, 273–297. doi:10.1023/A:1022627411411

Freund, Y., Schapire, R.E., others, 1996. Experiments with a new boosting algorithm, in: *ICML*. pp. 148–156.

Hämäläinen, W., Vinni, M., 2006. Comparison of machine learning methods for intelligent tutoring systems, in: *In: Proc. of Int. Conf. in Intelligent Tutoring Systems*. pp. 525–534.

Hastie, T., Tibshirani, R., Friedman, J.H., 2009. *The elements of statistical learning: data mining, inference, and prediction*, 2nd ed. ed, Springer series in statistics. Springer, New York, NY.

Huang, S., Fang, N., 2013. Predicting student academic performance in an engineering dynamics course: A comparison of four types of predictive mathematical models. *Computers & Education* 61, 133 – 145. doi:http://dx.doi.org/10.1016/j.compedu.2012.08.015

Hu, Y.-H., Lo, C.-L., Shih, S.-P., 2014. Developing early warning systems to predict students' online learning performance. *Computers in Human Behavior* 36, 469 – 478. doi:http://dx.doi.org/10.1016/j.chb.2014.04.002

Kotsiantis, S.B., 2012. Use of Machine Learning Techniques for Educational

Proposes: A Decision Support System for Forecasting Students' Grades. *Artificial Intelligence Review* 37, 331–344. doi:10.1007/s10462-011-9234-x

Ley, T., Kump, B., 2013. Which User Interactions Predict Levels of Expertise in Work-Integrated Learning?, in: Leo, D.H., Ley, T., Klamma, R., Harrer, A. (Eds.), *EC-TEL, Lecture Notes in Computer Science*. Springer, pp. 178–190.

Lin, C.J., Hsu, C.-W., Chang, C.-C., 2003. A practical guide to support vector classification. National Taiwan U., www.csie.ntu.edu.tw/~cjlin/papers/guide/guide.pdf.

Lykourantzou, I., Giannoukos, I., Mpardis, G., Nikolopoulos, V., Loumos, V., 2009. Early and Dynamic Student Achievement Prediction in e-Learning Courses Using Neural Networks. *J. Am. Soc. Inf. Sci. Technol.* 60, 372–380. doi:10.1002/asi.v60:2

Macfadyen, L.P., Dawson, S., 2010. Mining LMS Data to Develop an “Early Warning System” for Educators: A Proof of Concept. *Computers & Education* 54, 588–599. doi:10.1016/j.compedu.2009.09.008

Palczewska, A., Palczewski, J., Robinson, R.M., Neagu, D., 2013. Interpreting random forest models using a feature contribution method. *IEEE*, pp. 112–119. doi:10.1109/IRI.2013.6642461

Pedro, M.O., Baker, R., Bowers, A., Heffernan, N., 2013. Predicting college enrollment from student interaction with an intelligent tutoring system in middle school, in: *Educational Data Mining 2013*.

Petkovic, D., Okada, K., Sosnick, M., Iyer, A., Zhu, S., Todtenhoefer, R., Huang, S., 2012. Work in progress: A machine learning approach for assessment and prediction of teamwork effectiveness in software engineering education, in: *Frontiers in Education Conference (FIE)*, 2012. pp. 1–3. doi:10.1109/FIE.2012.6462205

Rosenbaum, L., Hinselmann, G., Jahn, A., Zell, A., 2011. Interpreting linear support vector machine models with heat map molecule coloring. *Journal of Cheminformatics* 3, 11. doi:10.1186/1758-2946-3-11

Schalk, P.D., Wick, D.P., Turner, P.R., Ramsdell, M.W., 2011. Predictive assessment of student performance for early strategic guidance, in: *Frontiers in Education Conference (FIE)*, 2011. pp. S2H-1–S2H-5. doi:10.1109/FIE.2011.6143086

Szczepańska, A., 2011. Research Design and Statistical Analysis, Third Edition by Jerome L. Myers, Arnold D. Well, Robert F. Lorch, Jr. *International Statistical Review* 79, 491–492. doi:10.1111/j.1751-5823.2011.00159_12.x

Vapnik, V.N., 1998. *Statistical learning theory*, 1st ed. Wiley.

Vapnik, V.N., 1995. *The Nature of Statistical Learning Theory*. Springer-Verlag New York, Inc., New York, NY, USA.

Villagrà-Arnedo, C., Castel De Haro, M.J., Gallego-Durán, F.J., Pomares Puig, C., Suau Pérez, P., Cortés Vaíllo, S., 2009. Real-time Evaluation, in: *EDULEARN09 Proceedings, 1st International Conference on Education and New Learning Technologies*. IATED, Barcelona, Spain, pp. 3361–3369.

Villagra-Arnedo, C., Gallego-Durán, F.J., Compañ-Rosique, P., Llorens-Largo, F., Molina-Carmona, R., 2016. Predicting academic performance from behavioural and learning data. *International Journal of Design & Nature and Ecodynamics* 11.

Villagra-Arnedo, C., Gallego-Durán, F.J., Molina-Carmona, R., Llorens-Largo, F., 2015. Boosting the Learning Process with Progressive Performance Prediction, in: Conole, G., Klobučar, T., Rensing, C., Konert, J., Lavoué, É. (Eds.), *Design for Teaching and Learning in a Networked World*. Springer International Publishing, Cham, pp. 638–641.

Wilkinson, L., Friendly, M., 2009. The History of the Cluster Heat Map. *The American Statistician* 63, 179–184. doi:10.1198/tas.2009.0033

Wu, T.-F., Lin, C.-J., Weng, R.C., 2004. Probability Estimates for Multi-class Classification by Pairwise Coupling. *J. Mach. Learn. Res.* 5, 975–1005.

Xing, W., Guo, R., Petakovic, E., Goggins, S., 2015. Participation-based student final performance prediction model through interpretable Genetic Programming: Integrating learning analytics, educational data mining and theory. *Computers in Human Behavior* 47, 168–181. doi:10.1016/j.chb.2014.09.034

Figure captions

Figure 1. Accuracy comparison for experiments 1 and 2.

Figure 2. Progression chart for an individual student at week 4.

Figure 3. Progression chart for an individual student at week 10.

Figure 4. Progression chart for a group at week 10, including the timeline with stages and milestones (MS).

Figure 5. Heat maps for a group of 50 students, using two different orderings: number of classifications criterion (left) and last week criterion (right).

Figure 6. Bands and parameters of a heat map, corresponding to HP class (green), MP class (grey) and LP class (red).

Figure 7. Confidence heat map for a group of 50 students, using the last week criterion. Colour hue identifies the class and colour saturation the confidence level