Profiling Users in Gamification

Sofia Mendes Gonçalves
Instituto Superior Técnico
Lisbon, Portugal
sofia.m.goncalves@tecnico.ulisboa.pt

Cláudia Antunes
Instituto Superior Técnico
Lisbon, Portugal
claudia.antunes@tecnico.ulisboa.pt

***Abstract*—The implementation of gamification in an educational context had a significant impact not only on the motivation but also on the learning performance of students. Gamified environments revealed different behaviors from different students in the same conditions, making it interesting to define student profiles.**

Educational Data Mining concerns the development of methods for exploring data that come from educational environments. It aims to better understand the student's learning process and to identify their learning settings to improve educational outcomes.

This thesis proposes the exploration of the knowledge discovery process to build a predictive model to early detect students' learning profiles on data collected from a gamified course. Given the sparsity of the data, a prime consolidation in a Data Warehouse was performed to ease the data mining process. In the profiling phase, we compared the performance of four learning algorithms on two differently labeled datasets, over several phases throughout the semester.

The results showed that the characteristics of the datasets and the selected hyperparameters consisted of essential factors to varying the performance of the classifiers.

Our approach ensures the possibility of predicting the students' learning profiles by five weeks into the semester.

***Index Terms*—gamification, gamified course, profiling, student modeling**

I. Introduction

Over the years, the integration of game elements in contexts other than games has shown promising results in terms of improvement of people's engagement in a specific activity [9] [24]. This use of game design elements in non-game contexts is called *gamification* [8].

Gamification emerged and started to be applied in several areas, such as marketing, productivity, health, finance, education, news, media, among others. In education, it plays an important part in increasing student engagement and improving the learning experience, since today's students have grown in more interactive environments and consider traditional learning demotivating [22]. However, different students have different learning styles, and it is important to explore the pedagogical effects of gamification in terms of impact in each student [2].

Educational Data Mining (EDM) is a discipline concerned with the development of methods to explore data that comes from educational settings in order to understand the students better. Therefore, through the application of data mining techniques, it is possible to group students according to their characteristics, to predict student's performance in terms of performance, score, and mark, and to create profiles by developing cognitive models of students considering its motivation, learning styles and learning behaviors [19].

The objective of this work is to explore the knowledge discovery process to do an early detection of student's learning profiles, based on their interaction with a gamified course, and further study how to adapt the learning experience to its individual needs to improve the learning experience. This will assist educators on identifying with which type of students they are dealing with in order to take adequate measures to help them obtain better results.

In the last decade, several studies tried to explain how the behavior of the student affects its performance, and to predict that performance through simple formulas [17] or tabular tools [5], disregarding the temporal nature of the data. Others explored the sequential nature of data as an important factor in student profiling [21] [23], the anticipation of results based on the exploration of temporal precedences [18], and few attempted to use sequence classifiers [16] and clusters [13] with promising results.

My proposal is to create a predictive model of student behavior by building classifiers that take into account the temporality and sequentiality of the data for anticipating and improving student profiles.

II. Gamification

The term *gamification* originated online back in the year 2008 but it was only in the second half of 2010 that it saw widespread adoption. Although the term is relatively new, the idea of gamification is not. Games and simulations have been used by the military for hundreds of years which made them pioneers in the use of these tools to explore its applications [8]. Also, there are several books dated from the 1960's that explore how games affect life and psychology and even movies from the 1980's that approach this theme [24]. A *game* is an interactive activity characterized by a rule-based system which continually provides challenges to one or several players. They are portrayed by a fictional context in the form of a story, graphics, and music [11] and encourage players to compete towards goals and involve them in an active learning process in order to fulfill those goals and master the game mechanics [11] [7].

As defined by Deterding et al. (2011), *gamification* consists of the use of game design elements in non-game contexts [8].

A. Gamification in an Educatonal Context

Along the years, the number of papers published on gamification in an educational context has been growing suggesting that the theme is becoming a popular subject for academic inquiry. In 2015, a search made by Dicheva et al. (2015) returned around 5.000 results for papers that discuss explicitly gamification in educational contexts [10]. There are several studies that unveiled many advantages of implementing game elements in education, as students are able to get immediate feedback, have access to information on demand, develop an important role in a community, are able to take control of and evaluate their own learning behavior and even improve their capacity of working as a team [11]. In online learning, gamification solves the problem of the lack of motivation on the part of the student, related with the limited capacity of interacting with the teacher or with classmates.

Hamari et al. (2016) studied the engagement, flow, and immersion in game-based learning to understand if challenging games effectively help students learn [14]. Although games are designed for entertainment and leisure, serious games and game-based learning are designed for training and educating. Although some students consider challenges unpleasant or arduous, the majority prefers to face a challenging task and value themselves more when achieving their goal. The perception of their competence leads to a feeling of accomplishment and raises motivation [15]. Also, the challenge is responsible for increasing learning since the student applies a wide range of strategies in order to succeed.

So, game-based approaches combined with motivation techniques are a promising replacement for overly theoretical classes with possible unfavorable schedules [22]. Cheong et al. (2013) conducted a study where he used a gamified quiz to evaluate IT undergraduate students in which they reported that the quiz improved their grades, learning effectiveness and also their enjoyment and engagement [6]. Domínguez et al. (2013) proposed a gamified approach to an e-learning course where students had the alternative to take exercises either by the conventional education system, in which they had to read a PDF file, or via a gamified system. Results showed that students opting by the gamified approach obtained better exam grades and reported higher engagement [11]. Glover et al. (2016), in a similar study, chose to add achievement badges to an online learning environment to support and encourage participation in learning activities and also obtained positive results [12].

All previously mentioned studies used a "one-size-fits-all" approach and did not explore the pedagogical effects of gamification in terms of its impact on different students in different ways. To try to bridge that gap, Barata et al. (2013) gamified an MSc course by adding game elements such as experience points and levels, badges and a leaderboard [2]. Also, the course included a set of challenges and quests which granted the students experience points when completed.

III. Educational Data Mining

Studies based on student's interactions with instrumental educational software and online learning as the ones previously mentioned generate a considerable amount of data, which can be explored and exploited in order to understand how students learn [19]. The exponential growth of educational data and its use to investigate scientific questions within educational research consists of one of the biggest challenges faced by educational institutions [19].

EDM is defined by the International Educational Data Mining Society (IEDMS) as "an emerging discipline, concerned with developing methods for exploring the unique types of data that come from educational settings and using those methods to understand the students better, and the settings which they learn in." The IEDMS is responsible for promoting scientific research and development in the field of educational data mining and for supporting collaboration between the members of the community through the organization of the EDM conference, which has occurred every year since it was first held in 2008, in Montreal, Quebec, and through the assembling of several journal articles submitted by the community in order to create the Journal of Educational Data Mining. This society supports the sharing of data techniques and brings together a community of computer scientists, learning scientists and researchers [20].

A. Goals of EDM

Data gathered from schools, colleges, universities, and other learning institutions that work modern forms and methods of teaching is not restricted to the interactions between the student and the system, such as the results of online quizzes or its behaviour while navigating on the platform. There are many ways of obtaining relevant information, and that may include data from collaborative students, extracted from text forums where they communicate and clarify doubts, administrative data, related to the school or the teacher, demographic data, such as gender, age, and student grades, and data from student affectivity, which can be measured by its motivation, for example. So, there are many different types of data available for mining that although being specific to the educational area, have intrinsic semantic information and multiple levels of hierarchy, which means that can be related to the student, to the assignments, or to the questions [19] [20].

The existence of data from thousands of institutions and students with similar learning experiences but in different contexts gives leverage for studying contextual factors on learning and learners [1]. EDM has the potential to analyze important questions in individual differences [1] and enables data-driven decision-making to improve the current education methods and learning materials [20]. However, depending on the viewpoint of the final user and on the problem that needs solving, it is possible to consider many specific objectives in EDM such as understanding how to structure or restructure classes, organize the evaluation methods and distribute the materials based on the usage of the platform and on the performance data; identifying students that need more feedback, study advice or

other type of help; deciding which kind of feedback, advice or help would be more effective; and even understanding how to help learners in finding useful material, either individually or in collaboration with colleagues [20].

B. Student Profiling: Related Work

In the last decade, there have been several efforts to explain how the behavior of the student affects its performance and to predict that performance, so that the educator can know as soon as possible with which type of students he is dealing with, to identify if the student needs guidance or only to provide feedback. In these studies, data was mainly explored through simple formulas [17] or tabular tools [5], despising intrinsic temporal data.

Liang et al. (2017) analyzed the behavior characteristics of online learners in MOOCs and the factors affecting the student profile [17]. The authors used the Jaccard coefficient for labeling the students as different types, depending on the behavior and the duration of the performance. In order to predict the students' success, the authors propose the use of classification algorithms. Concerning learning attitude, this study points out that e-learning courses require specific objectives, inner motive, synchronous feedback and shall allow the learners to be independent. Regarding achievement prediction, the study shows that students that present a specific behavior during the first half of the semester are more likely to maintain it until the end.

Bydzovská et al. (2017) address student performance prediction by searching for patterns using classification and regression algorithms and by analyzing students' social behavior data [5]. The two main tasks are predicting students' success or failure and predicting final grades.

Other studies, such as the ones performed by Silva et al. (2014) and Vale et al. (2014), explored the sequential nature of data to help profiling students [21] [23]. Silva et al. (2014) explored a multi-dimensional algorithm to find multi-dimensional patterns in educational environments and model student behaviors and conclude that there is an improvement on students' results prediction when comparing with the same classifiers trained without multi-dimensional patterns [21].

Vale et al. (2014) show how to apply biclustering on educational data and how to use its results as features to predict student's performance [23].

In other studies, McBroom et al. (2017) approached this subject by exploring temporal precedences among data [18]. The authors investigated techniques used to identify and follow the development of student behavior over the semester and focused specifically on the application of those techniques to a junior computer science course. They gathered the most common behaviors of students, followed how those behaviors changed over time, and studied the relationship between the behaviors and the final exam outcomes.

There were only a few attempts to use sequence classifiers [16] and clusters [13] to study profiling with promising results that should be thoroughly explored in the gamifying context. Lee et al. (2014) propose a data-driven approach capable of learning a player's movements in a sequential decision-making process by using supervised method to predict the next movement of the player based on past gameplay data [16].

Finally, Klinger et al. (2016) propose an evolutionary pipeline which can be applied to learning data that aims to improve cluster stability over multiple training sessions [13].

IV. The Course

Multimedia Content Production (MCP) is a gamified Master of Science course, taught at Instituto Superior Técnico, in Lisbon, yearly, during a whole semester. The course blends theoretical and lab lectures with a gamified Moodle platform and instead of the typical grading system, students are awarded Experience Points (XP) which vary from 0 XP to the top grade, 20.000 XP. For every 1.000 XP, the student increases a level, going form level 0 to level 20, which can be equated to the university's traditional grading system. In order to be approved in the course he or she has to reach level 10, that is 10.000 XP.

Students are awarded XP whenever they complete an evaluation item from a course activity. The current evaluation methodology includes lab assignemnts (15% of the final grade), a Multimedia Presentation (15% of the grade), Quizzes (30% of the final grade), submissions for the Skill Tree (25% of the grade) and a set of collectible badges which are granted by completing achievements (15% of the final grade and students may work for an extra of 5%).

The Moodle platform is the main environment where the interaction with the several activities of the course occurs. It is there where students go to access the course materials, take the weekly quizzes, submit the lab assignments, and take questionnaires regarding their player style and their opinion towards the gamified experience. The platform also includes a discussion forum where they can interact with colleagues and professors, access the latest announcements of the course and cooperate by answering each other's questions and discussing the course topics. Teachers may grade relevant posts from 0 to 5.

One of the main elements of the gamified experience is the *leaderboard*. It is a webpage that is accessible from the Moodle platform and it displays the list of enrolled students in descending order, according to their accumulated XP, showing the top scores in the first positions of the list. Besides this, each student has a *personal profile* where he or she can track the amount of XP collected since the beginning of the semester, the *collected badges* as well as the *completed achievements*, and access personal statistics about their progress in the course activities.

As mentioned, at some time during the semester, teachers provide *questionnaires* to understand the student's relationship with the course. In the first weeks, a questionnaire to predict a learning style is available and each question covers a dimension according to the possible learning style. Along the semester more questionnaires are made available to understand the level of engagement with the course, access some changes

that may have happened since the beginning of the course and improve the course in future editions.

Although the course follows the structure mentioned for the last few years, there are several changes that occurred along the years. The course has been adapted to the needs of the gamified experience and according to the student's feedback by the end of the semester. Some of those changes include the replacement of a final exam by regular quizzes along the weeks of the semester, addition and replacement of achievements and corresponding badges, as well as skills from the Skill Tree.

A. The Data

The data provided was extracted from the Moodle platform and consisted of three main folders: one containing all the posts from the discussion forums from the academic year of 2010/2011 until 2018/2019, another one with the students' answers to questionnaires regarding their learning style, from the academic year of 2018/2019, and the last one containing the logs regarding every interaction with the platform along with some metadata from the last nine years. The data was transformed and loaded into a Data Warehouse in order to consolidate it in a suitable format for further student profile prediction

V. Student Profiling

Barata et al (2014) present an experiment based on student's performance on the MCP course where they try to understand which data better characterizes a student type [4]. Using cluster analysis, four main student types were identified: Achievers, Regular, Halfhearted, and Underachievers. They also identified that besides the different levels of participation and performance, the accumulated XP over time was also an indicator of each type of student [3].

To perform the profiling of the students, I first started by labeling the dataset which contained all the students as records and the correspondent performance by the end of the semester as attributes, based on the studies previously referred to. Known the correspondent student type, I used the same labels in seven other datasets extracted from the data warehouse just as before for the final performance, but now for the performance in different weeks through the semester. In total, the number of datasets that will be used for classification is eight, and consist of the students' performance after three, five, seven, nine, eleven, thirteen and fifteen weeks, and also by the end of the semester.

The next step was to preprocess the data to further feed the machine learning algorithms. The chosen classifiers were Naïve Bayes, Decision Trees, Random Forests and Gradient Boosting which had to be built and tuned to achieve the best results for each moment in the semester.

Finally, with the built classifiers, it was possible to perform profile prediction in different phases of the semester to understand how soon a student type can be predicted with high accuracy and high sensitivity for each of the classes.

A. Labeling Criteria

Since the provided data does not include an assignment of a profile to each student, I started by labeling the dataset which contained the calculated measures for each student based on the XP accumulation curves presented by Barata et al (2016).

Given the rough similarity in the evolution of the curves and the fact that the points of each curve representing the final XP are close to equidistant from each other, I opted by taking into account only the final XP and label the students using two methods: quartiles and percentiles. The first consist of a separation of data into four bins where the first contains the lowest 25% of numbers, the second between 25,1% and 50%, the third between 51% and 75%, and the fourth the highest 25% of numbers. The second separates data according to percentiles and in this case four bins where considered: the first contains all the students with final XP below 63%, the second between 63,1% and 75%, the third between 75,1% and 85%, and the fourth with XP above 85%. In both cases, the students placed in the first bin are labeled as Underachievers, in the second bin as Halfhearted, in the third as Regular, and in the fourth as Achievers.

By the end of this phase, two datasets were created, in order to perform two different experiments: one regarding labeling according to quartiles, and other according to percentiles.

B. Evaluation Criteria

For cases where the target variable classes of a dataset are a majority of one class, that is, when the dataset is imbalanced, and also when dealing with a multi-class problem, using accuracy alone to evaluate the performance of a classifier can be misleading. Figure 2 represents the clear imbalanced distribution of the classes for the *percentiles* datasets.

The most commonly used performance measure is *accuracy* which measures the fraction of predictions that the model guessed as correct. However, given the imbalanced nature of the datasets, in this work, we will be taking into account the *recall (sensitivity)* of the classifiers for each class, as the goal is to classify a student as being of a specific type correctly.

C. Preprocessing

Both the *quartiles* and *percentiles* datasets have 40 attributes and one target variable, which is one of the four student types. All attributes are numerical and continuous, except the "Pass/Fail" which is categorical. The datasets contain a total of 580 records each. All the actions performed in the preprocessing phase were applied to both datasets except when stated otherwise.

1) Data Cleaning: Since the attributes of the dataset correspond to the measures extracted from the aggregated fact tables of the data warehouse, we expect that the missing values are nonexistent, as most of the measures consist of sums and counts. However, when box plotting the distribution of the final XP per year, it is possible to detect some outliers. An outlier, in this case, is a student whose final XP does not fit the distribution for that year. The number of outliers consisted of only 3% of the data, and the impact on the Naïve Bayes

classifier performance was minimal, so the decision made was to remove them. The fact that all the outliers appear below the ”minimum” of each boxplot also supported this decision. This puts as a possibility that the outliers consist of dropout students, which do not belong in the considered student-types.

These same outliers were also removed from the datasets regarding the remaining weeks.

Finally, I have removed the *StudentID* and *Year* attributes as they are considered irrelevant to the classification process.

2) Data Reduction: Since irrelevant data may contribute to a decrease in the accuracy of some models, I performed feature selection on both datasets. The optimal number of features was selected according to how the students performed until then.

Feature selection was only performed when feeding data to the Naive Bayes classifier, as the remaining tree-based algorithms have their built-in feature selection methods and we are dealing with very few attributes.

3) Data Transformation: In this phase, the categorical attribute was binarized, as its values could only be “Pass” or “Fail”.

The remaining attributes had to be normalized, since the chosen algorithm for the Naïve Bayes classifier was GaussianNB and it has to follow a Gaussian distribution. However, for the remaining tree-based algorithms, scaling is not necessary because these classifiers are not affected by different scales, since node partitions happen by comparing a feature to a threshold value.

4) Resampling: By analysing the class distribution for the *quartiles* datasets, it is clear that the number of students labeled as some profile is very close to the number of students labeled as the others. Although the number of *Achievers* consists of a third of the number of *Underachievers*, it is possible to assume that the datasets are relatively balanced and that there is no need for resampling in this case. However, for the *percentiles*, by taking a look at Figure 2, the datasets are highly imbalanced, being the number of *Underachievers* close to 350 and the remaining student types between 60 and 80.

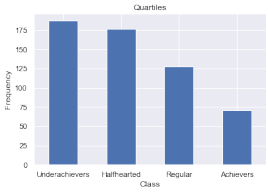

Fig. 1. Class distribution for the *quartiles* dataset

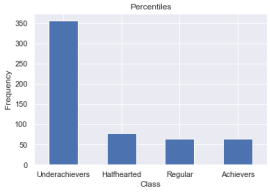

Fig. 2. Class distribution for the *percentiles* dataset

In order to avoid the possibility of the classifiers being biased for the majority class, I used the Synthetic Minority Oversampling Technique (SMOTE) algorithm to generate synthetic data and randomly create a sample of attributes regarding all classes but the majority class. The performances of the classifiers were tested with and without this balancing technique to understand its impact on the performance of the classifier.

D. Classification

I followed the premise of dividing the data in a training set, to optimize the model's parameter values and build up the model, and a test set, to evaluate the optimized model. Since the datasets are medium-sized, with a total of 580 samples, I applied Cross-Validation with 10 folds.

The graphic representations of the performance of each classifier in each experiment contain not only the mean accuracy for the Cross-Validation process but also the 95% confidence intervals for which the performance may deviate from the actual value.

1) Baseline: The performance of the Naïve Bayes algorithm for the *quartiles* and the *percentiles* datasets is represented in Figure 3 and Figure 4, respectively. On the one hand, for the *quartiles* datasets, Figure 3 shows an increase of 45% of the accuracy, from 3 weeks until the end of the semester. Before the end of the semester, the highest values for accuracy occur by 7 weeks, with 37%. On the other hand, Figure 4 indicates a less steep slope from 3 weeks until the end of the semester. This slope indicates an increase of around 15% for the classifier without resampling. In spite of this small increase, the Naïve Bayes classifier seems to present much better prediction accuracy for the earlier weeks, and a value of accuracy very close to the end of the semester already after 7 weeks. By this time, it would be possible to predict the type of student very closely to the type predicted by the end of the semester. However, accuracy is only a good measure when the target variable classes in the data are nearly balanced, which does not happen in this case.

Fig. 3. Performance of the Naïve Bayes classifiers for the *quartiles* datasets

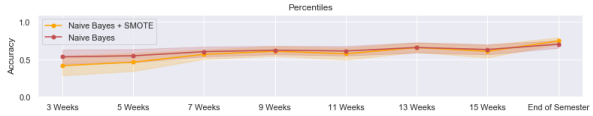

Fig. 4. Performance of the Naïve Bayes classifiers for the *percentiles* datasets

In order to avoid a misleading analysis of the graphic representations, the performance for the Naïve Bayes algorithm with resampling is also represented. These values are very close to the ones for the classifier with no resampling, but Figure 5 shows that with SMOTE, on average, the sensitivity for the majority class decreases, and increases for the minority classes. Taking a look at the plot for 3 weeks, although the sensitivity for the majority class decreases, the percentage of the students who are being correctly classified as *Regular* increases to 47%, when without SMOTE it was 30%. By the end of the semester, SMOTE contributes to an increase of around 14% of the sensitivity of the *Regular* and *Achiever* classes.

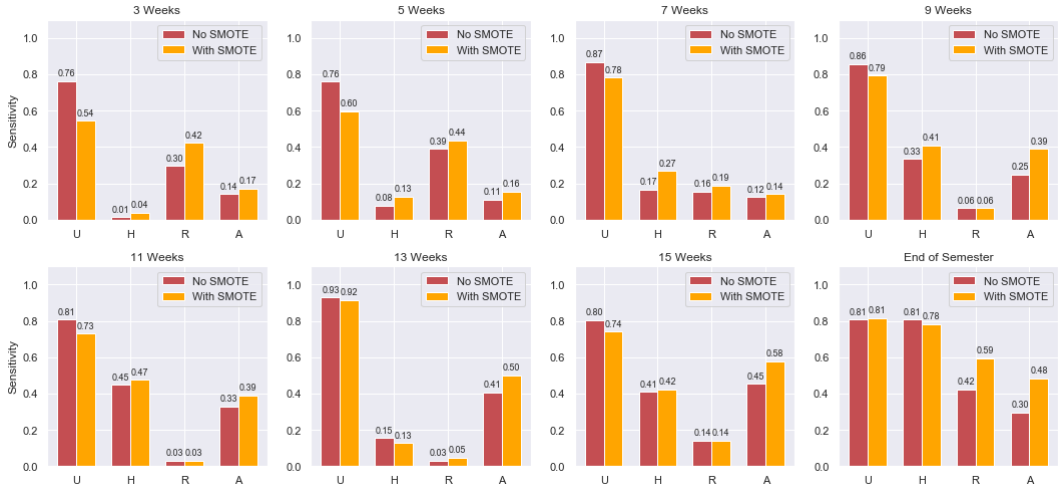

Fig. 5. Naïve Bayes classifiers' sensitivity for each class along the weeks

2) Decision Trees: Without tuning any of the available parameters of the Decision Tree classifier, it may result in a tree with an unnecessary number of nodes, which means a highly complex algorithm that may result in low performance for unseen data and in an overfit model. In order to solve this issue, I observed the classifier behavior when changing some specific parameters and its impact on the predictive power of the model.

Decision Trees have a high number of hyperparameters which require fine-tuning in order to obtain the best model which reduces the generalization error as much as possible. In this case, I opted by focusing on four different hyperparameters: max_depth, min_samples_split, min_samples_leaf, and max_features.

Figure 6 and Figure 7 represent the accuracy for each of the classifiers with 10-Fold Cross-Validation for the *quartiles* and *percentiles* datasets, respectively. As expected, tuning the hyperparameters contributed to an increase in the accuracy of the classifiers in an overall vision. With the tuned parameters, for the *quartiles* datasets, the classifier has an accuracy of around 40% by 3 weeks into the semester. The highest values for accuracy before the end of the semester occur at 7 weeks reaching around 55% and stand relatively stable until 15 weeks.

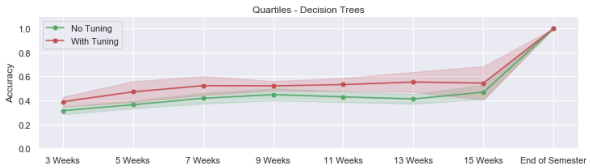

Fig. 6. Performance of the Decision Tree classifiers for the *quartiles* datasets

As for the *percentiles* datasets, the improvement of the classifier with hyperparameter tuning is also clear. Figure 7 represents in detail the obtained results for the datasets. Without tuning, the earliest phase when accuracy hits the highest values occurs by 7 weeks, which means that by this time, it is possible to predict the student profile with almost 56% accuracy. With no tuning and resampling, by 7 weeks the values for accuracy stabilize until 15 weeks with values around 53%. The classifiers present the best results with tuned hyperparameters. From 3 until 15 weeks, there is a difference of only 8% of accuracy, starting at 64%. The line reaches its peak at 9 weeks with an accuracy of around 69%, however the difference between 7 and 9 weeks is about 1%, and between 7 and 15 weeks the values for accuracy remain stable around 68%, meaning that by 7 weeks, it is possible to predict the student profiles with 68% of accuracy. Applying SMOTE to the tuned hyperparameters results in a decrease of 30% for the 3 weeks dataset when comparing to the tuned classifiers, however, by 5 weeks the classifier reaches 50% of accuracy and the peak is hit also at 7 weeks just as before.

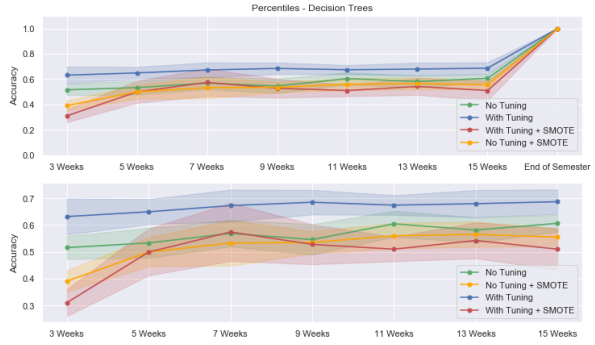

Fig. 7. Performance of the Decision Tree classifiers for the *percentiles* datasets

Once again, the fact that SMOTE affects the performance of the classifiers with hyperparameter tuning, decreasing accuracy is deceiving. Figure 8 shows resampling the data leads to an increase of sensitivity for the minority classes, and it is visible that the percentage of correctly classified students is relatively balanced for all the classes.

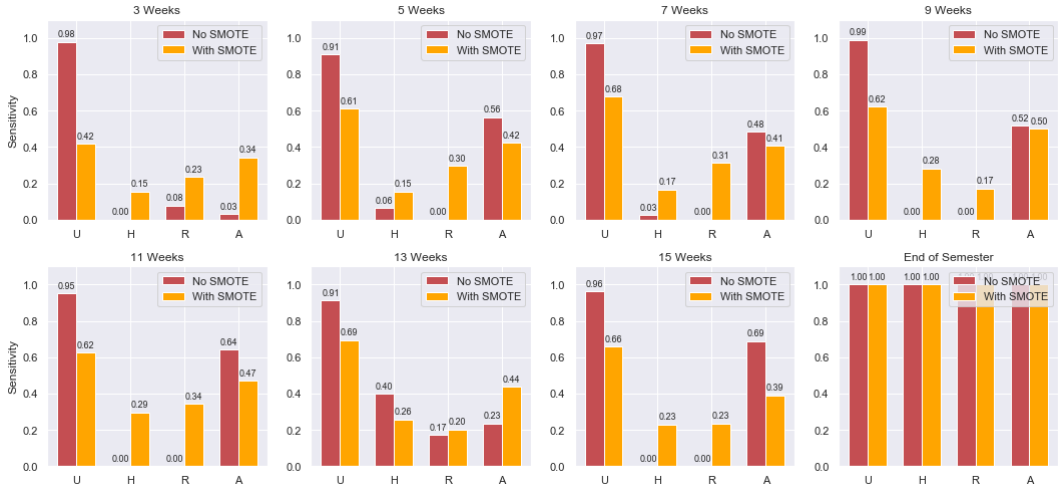

Fig. 8. Tuned Decision Tree classifiers' sensitivity for each class along the weeks

3) Random Forests: As a Random Forest is an ensemble algorithm that takes the average of many Decision Trees to arrive at a final prediction, adding to the hyperparameters chosen for the Decision Trees, another main parameter to consider is the number of trees in the forest (n_estimators). In this case, the higher the number of trees, the better, however, adding a lot of trees can slow down the training process.

For the *quartiles* datasets, without or with hyperparameter tuning, the performance of the classifiers is very similar, being the values for the classifiers with tuned parameters more accurate in around 10% at each of the considered weeks. For these datasets, before the end of the semester, the peak is hit at 15 weeks which is very late in the semester and very close to the end. The evolution of the performance of the classifiers can be followed in Figure 9.

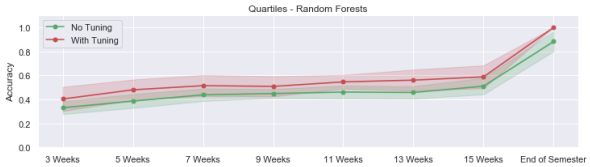

Fig. 9. Performance of the Random Forest classifiers for the *quartiles* datasets

For the *percentiles* datasets, tuning the parameters without resampling the data shows very promising results as confirmed by Figure 10. By 3 weeks into the semester the classifier predicts the student types with 65% of accuracy slowly increasing and reaching a stable value of 69% by 7 weeks. In general, with or without hyperparameter tuning, the Random Forest classification algorithm performs very well, showing slightly worse accuracy (minus 10%) by 3 weeks and by the end of the semester. As for the remaining tests, although in general the results for accuracy seem relatively worse when resampling, Figure 11 proves that the classifiers with tuning and SMOTE are not biased towards the majority class as the sensitivity for the remaining classes increases. Both of the tests performed with SMOTE present higher values for accuracy by the end of the semester, and for the classifiers with tuning and SMOTE, these results are even better as the sensitivity for each class equals 1, meaning that by this time, the classifier has an accuracy of 100%.

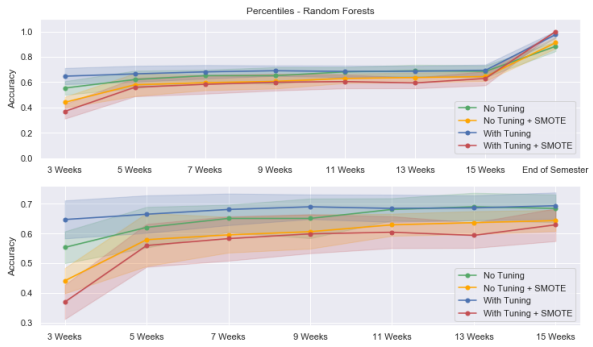

Fig. 10. Performance of the Random Forest classifiers for the *percentiles* datasets

4) Gradient Boosting - XGBoost: Similarly to the Decision Tree and Random Forest algorithms, in order to improve the model, hyperparameter tuning is required. There is a broad number of hyperparameters which require tuning in order to optimize the algorithm. In this work, I opted by choosing the following booster parameters: max_depth, learning_rate, n_estimators, min_child_weight, subsample, and colsample_bytree.

Figure 12 shows the performance of the XGBoost classifiers for the *quartiles* datasets. Overall, the difference of the accuracy for each of the weeks is minimal, being the most evident the difference of close to 10% by 7 weeks. Before the end of the semester, the best results for accuracy occur by 11 weeks for both tuned and non-tuned classifiers.

As for the *percentiles* datasets, the classifiers also performed very similarly, being the best results from the tuned model with no SMOTE remaining relatively constant along the weeks but

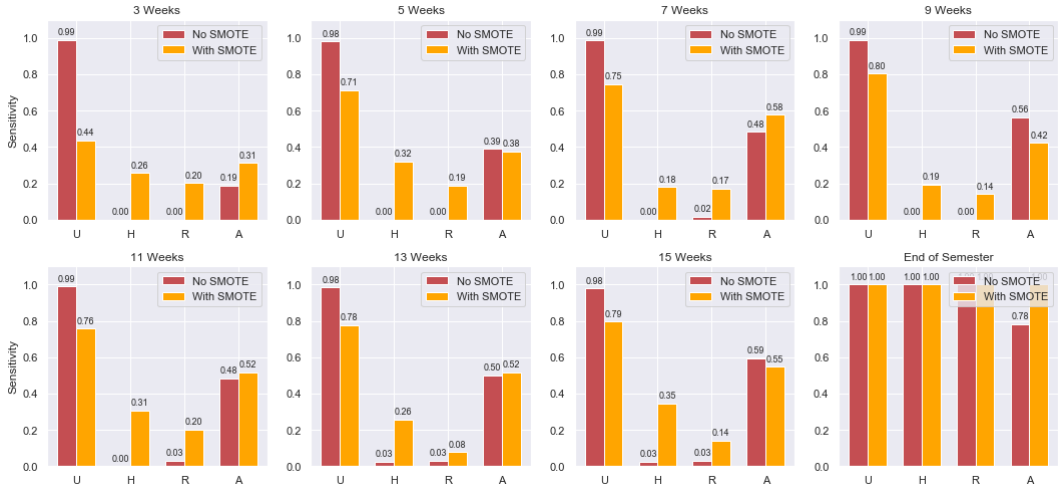

Fig. 11. Tuned Random Forest classifiers' sensitivity for each class along the weeks

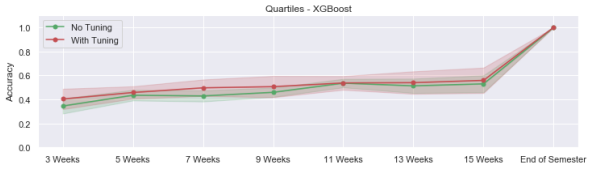

Fig. 12. Performance of the XGBoost classifiers for the *quartiles* datasets

reaching a peak of around 70% of accuracy by 9 weeks. In order to clarify the differences between the classifiers, Figure 13 represents the performance along the first 15 weeks in detail. With no tuning, the peak is hit by 7 weeks and accuracy stabilizes on values around 67%. With SMOTE, both the plots represent a decrease of accuracy in general but by 5 weeks, both the classifiers' accuracy remains stable apart from a small decrease at 9 weeks for the classifier with no tuning. The increase of the sensitivity for each of the classes with SMOTE is represented in Figure 14.

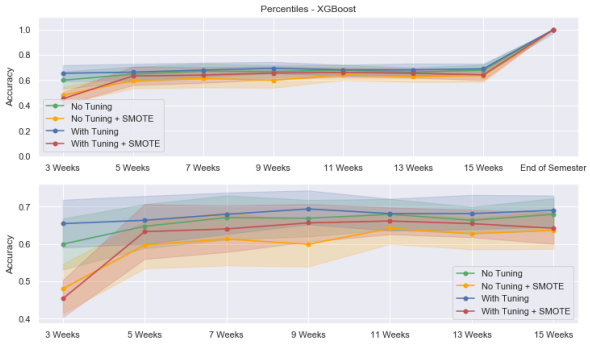

Fig. 13. Performance of the XGBoost classifiers for the *percentiles* datasets

5) Overall Performance: From the analysis of the graphs of each of the algorithms, for both the *quartiles* and *percentiles* datasets, the highest values for accuracy belong to the classification algorithms with tuned hyperparameters and without SMOTE. However, as previously stated, despite the overall decrease of accuracy, the implementation of SMOTE contributed to an increase of the classifiers' sensitivity for the minority classes. As the *percentiles* datasets are imbalanced and contain more than two classes, in this overall analysis I will consider the performance of the classifiers with hyperparameter tuning and SMOTE as the best performance for the *percentiles* datasets.

Figure 15 and Figure 16 gather the performance of each of the best performing classifiers on the *quartiles* and *percentiles* datasets respectively. The performance of the Naïve Bayes algorithm is also represented as base line learning algorithm.

Observing the performance of the algorithms in detail for the first 15 weeks represented in Figure 15 it is possible to admit for the *quartiles* datasets that although the values for accuracy are not very high to take assured conclusions about the student type prediction, there is a noticeable growth of the performance of the classifiers, as it increased close to 20% for each of the algorithms between 3 and 15 weeks.

By 7 weeks, XGBoost, Decision Trees and Random Forests predict the student types with close to 50% of accuracy, with large confidence intervals. However, by 9 weeks, the Random Forest classifiers present a smaller confidence interval and a higher value of accuracy when comparing to the other tree-based classifiers.

So, for these datasets the earlier phase where profile prediction can be done is by 9 weeks, with 52% accuracy for the Random Forest classifier. However, a fraction of right predictions of 50% by the middle of the semester is the same as a random choice of class by the classifiers and there is a risk of existing no connection between the features and the class.

For the *percentiles* datasets, the detailed performance of the best classifiers of each algorithm by the first 15 weeks is represented in Figure 16.

In this case, all the tree-based classifiers surpass 50% of accuracy after 5 weeks which is earlier than after 7 weeks

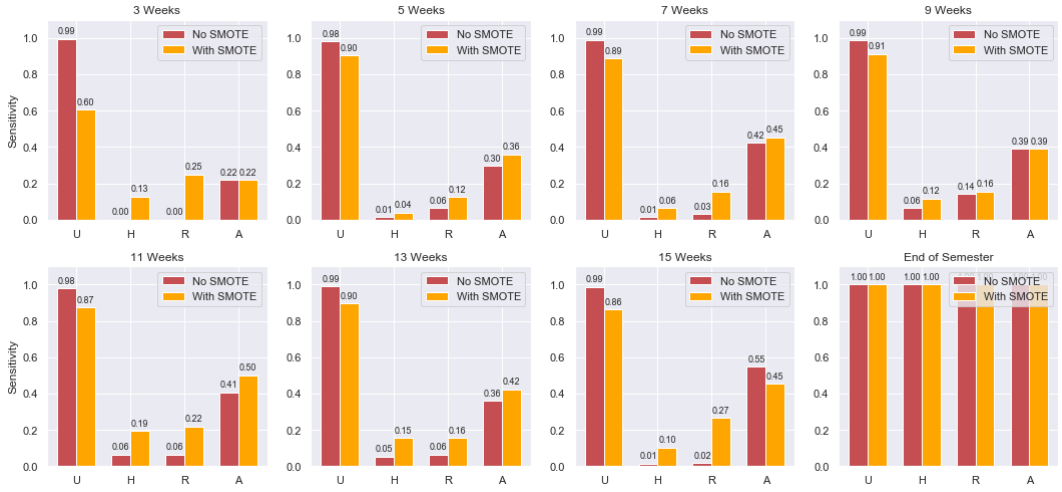

Fig. 14. Tuned XGBoost classifiers' sensitivity for each class along the weeks

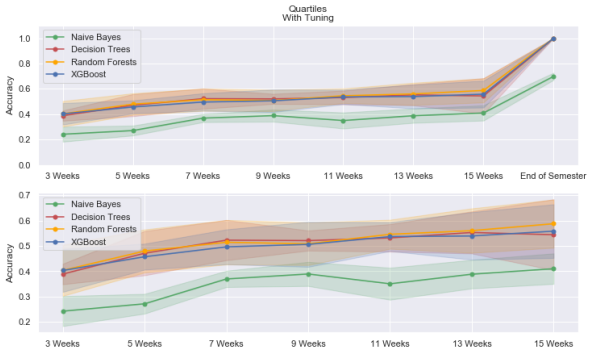

Fig. 15. Classifiers' performance for the *quartiles* datasets with tuned hyperparameters

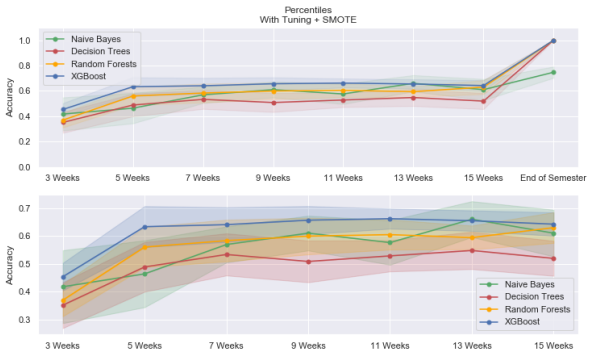

Fig. 16. Classifiers' performance for the *percentiles* datasets with tuned hyperparameters and SMOTE

for the *quartiles* datasets. The Decision Tree classifiers have a performance very close to the one for the same classifiers on the *quartiles* datasets. The peak is hit after 7 weeks with 53% accuracy. As for the Random Forest classifiers accuracy is maintained over 50% beyond 5 weeks. Beyond 7 weeks, the accuracy stabilizes in a range of 57 to 62%. Finally, for the XGBoost classifiers, after 5 weeks the values of accuracy are relatively stable in a range of 63 to 66%.

Given the fact that the fraction of right predictions for these datasets is slightly better than for the *quartiles* datasets, and that the XGBoost classifier presents a better overall performance, it is possible to assume that the earlier phase where it is possible to predict a student type occurs by 5 weeks. By then, 63% of the students are correctly classified, and since we are evaluating the classifiers taking into account the sensitivity for each of the classes, we're before more reliable results.

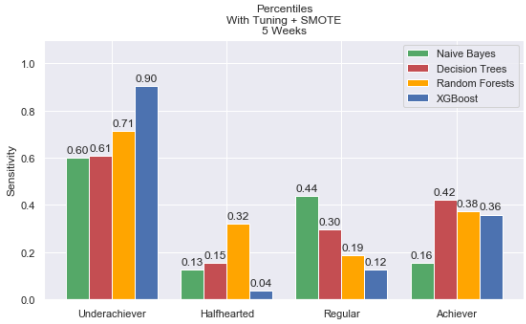

Fig. 17. Class sensitivity for each of the classifiers for the *percentiles* datasets by 5 weeks

Figure 17 represents class sensitivity for each of the studied classifiers by 5 weeks into the semester. By this time, XGBoost has higher sensitivity for the Achievers and Underachievers, whereas Random Forests have more balanced sensitivity, with higher values for Underachievers, Halfhearted and Achievers. Decision Trees admit that by this time, the main percentage of students belongs to the Underachievers, Regular and Achievers. The Naïve approach is more sensitive to Underachievers and Regular students.

After 5 weeks, the values of sensitivity for the Underachiever and the Achievers remains relatively stable for each of the classifiers until the end of the semester. However, for the

Halfhearted and Regular classes, the sensitivity for one may be higher than for the other in one week, and two weeks later may be lower again, meaning that until the end of the semester, students are mainly fighting belonging to one of these two classes.

The possibility of predicting student profiles by 5 weeks into the semester is supported by the fact that access to activities, such as the Skill Tree and the Achievements, is granted since the beginning of the semester. These activities consist of a total of 40% of the final grade. Since most of the regular courses happen to hand project assignments by midterm, the opportunity for students to excel on this gamified course occurs before that time.

VI. Conclusions

This thesis aimed to build a predictive model to early detect students' learning profiles by using machine learning techniques on data collected from a gamified course.

From the four evaluated models, results showed that the overall best performing model was XGBoost, which outperformed the other models, Naïve Bayes, Decision Trees, and Random Forests in both of the experiments, being the most effective method for performing student profiling.

The performance of each model on the student profiling task depends on several factors, such as the choices made on the preprocessing phase, the size of the dataset, how balanced the dataset is, and on the hyperparameter chosen for tuning. However, the experiments have shown that even when dealing with balanced datasets, the classification task is still hard to perform with satisfying results.

For the experiment regarding the imbalanced dataset, the best results for accuracy were obtained for tuned classifiers with no resampling. However, these classifiers presented low sensitivity for the minority classes, so by taking both accuracy and sensitivity into account, the most reliable results consisted of the ones obtained for the tuned classifiers with SMOTE.

The primary concerns of this work derive from the fact that this consists of a hard problem, where we are acknowledging four highly imbalanced classes and where the classifiers return low values of average accuracy and individual class sensitivity for each of the considered weeks, except for the end of the semester. However, an analysis of the evolution of the accuracy and class sensitivity along the weeks shows that the most significant difference between both measures occurs between 3 and 5 weeks. Beyond this phase, the evaluation measures tend to stabilize. Therefore, the earlier stage where it is possible to predict the students' learning profiles is after 5 weeks, with an accuracy of around 63%.

References

[1] RSJD Baker et al. Data mining for education. *International encyclopedia of education*, 7(3):112–118, 2010.

[2] Gabriel Barata, Sandra Gama, Joaquim Jorge, and Daniel Gonçalves. Engaging Engineering Students with Gamification: An empirical study. *Vs-Games 2013*, (January):1–8, 2013.

[3] Gabriel Barata, Sandra Gama, Joaquim Jorge, and Daniel Gonçalves. Early prediction of student profiles based on performance and gaming preferences. *IEEE Transactions on Learning Technologies*, 9(3):272–284, 2016.

[4] Gabriel Barata, Sandra Gama, Joaquim AP Jorge, and Daniel JV Gonçalves. Relating gaming habits with student performance in a gamified learning experience. In *Proceedings of the first ACM SIGCHI annual symposium on Computer-human interaction in play*, pages 17–25. ACM, 2014.

[5] Hana Bydžovská. A comparative analysis of techniques for predicting student performance. In *Proceedings of the 9th International Conference on Educational Data Mining 2016*, 2016.

[6] Christopher Cheong, France Cheong, and Justin Filippou. Quick quiz: A gamified approach for enhancing learning. In *PACIS*, page 206, 2013.

[7] Sebastian Deterding, Staffan Bjork, Lennart E. Nacke, Dan Dixon, and Elizabeth Lawley. Designing Gamification: Creating Gameful and Playful Experiences. *CHI 2013: Changing Perspectives, Paris, France*, pages 3263–3266, 2013.

[8] Sebastian Deterding, Dan Dixon, Rilla Khaled, and Lennart Nacke. From game design elements to gamefulness: defining gamification. In *Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments*, pages 9–15. ACM, 2011.

[9] Sebastian Deterding, Rilla Khaled, Lennart E Nacke, and Dan Dixon. Gamification: Toward a definition. In *CHI 2011 gamification workshop proceedings*, volume 12. Vancouver BC, Canada, 2011.

[10] Darina Dicheva and Christo Dichev. Gamification in education: Where are we in 2015? In *E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*, pages 1445–1454. Association for the Advancement of Computing in Education (AACE), 2015.

[11] Adrián Domínguez, Joseba Saenz-De-Navarrete, Luis De-Marcos, Luis Fernández-Sanz, Carmen Pagés, and José Javier Martínez-Herráiz. Gamifying learning experiences: Practical implications and outcomes. *Computers and Education*, 63:380–392, 2013.

[12] Ian Glover. Student perceptions of digital badges as recognition of achievement and engagement in co-curricular activities. In *Foundation of digital badges and micro-credentials*, pages 443–455. Springer, 2016.

[13] Markus Gross, Barbara Solenthaler, Severin Klingler, and Tanja Käser. Temporally Coherent Clustering of Student Data. *Proceedings of the 9th International Conference on Educational Data Mining*, pages 102–109, 2016.

[14] Juho Hamari, David J Shernoff, Elizabeth Rowe, Brianno Coller, Jodi Asbell-Clarke, and Teon Edwards. Challenging games help students learn: An empirical study on engagement, flow and immersion in game-based learning. *Computers in Human Behavior*, 54:170–179, 2016.

[15] S Lamborn, F Newmann, and G Wehlage. The significance and sources of student engagement. *Student engagement and achievement in American secondary schools*, pages 11–39, 1992.

[16] Seong Jae Lee, Yun-en Liu, Zoran Popović, and Zoran Popovi. Learning Individual Behavior in an Educational Game : A Data-Driven Approach. *Proceedings of the 7th International Conference on Educational Data Mining (EDM)*, (Edm):114–121, 2014.

[17] Kun Liang, Yiying Zhang, Yeshen He, Yilin Zhou, Wei Tan, and Xiaoxia Li. Online Behavior Analysis-Based Student Profile for Intelligent E-Learning. *Journal of Electrical and Computer Engineering*, 2017, 2017.

[18] Jessica McBroom, Bryn Jeffries, Irena Koprinska, and Kalina Yacef. Mining Behaviours of Students in Autograding Submission System Logs. *Edm '16*, pages 159–166, 2016.

[19] Cristóbal Romero and Sebastián Ventura. Educational data mining: a review of the state of the art. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 40(6):601–618, 2010.

[20] Cristobal Romero and Sebastian Ventura. Data mining in education. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 3(1):12–27, 2013.

[21] A Silva and C Antunes. Mining Multi-dimensional Patterns for Student Modelling. *Educational Data Mining 2014*, pages 393–394, 2014.

[22] Bahar Taspinar, Werner Schmidt, and Heidi Schuhbauer. Gamification in education: A board game approach to knowledge acquisition. *Procedia Computer Science*, 99(October):101–116, 2016.

[23] André Vale and Sara C Madeira. Mining Coherent Evolution Patterns in Education through Biclustering. *Proceedings of the 7th International Conference on Educational Data Mining (EDM)*, pages 391–392, 2014.

[24] Gabe Zichermann and Christopher Cunningham. *Gamification by design: Implementing game mechanics in web and mobile apps*. " O'Reilly Media, Inc.", 2011.