Mining Moodle Logs for Grade Prediction: A methodology walk-through

Álvaro Figueira
CRACS / INESCTEC & University of Porto
Rua do Campo Alegre, 1021/1055
4169-007 Porto, Portugal
arf@dcc.fc.up.pt

ABSTRACT
Research concerning mining data from learning management systems have been consistently been appearing in the literature. However, in many situations there is not a clear path on the data mining procedures that lead to solid conclusions. Therefore, many studies result in ad-hoc conclusions with insufficient generalization capabilities. In this article, we describe a methodology and report our findings in an experiment which one online course which involved more than 150 students. We used the Moodle LMS during the period of one academic semester, collecting all the interactions between the students and the system. These data scales up to more than 33K records of interactions where we applied data mining tools following the procedure for data extraction, cleaning, feature identification and preparation. We then proceeded to the creation of automatic learning models based on decision trees, we assessed the models and validate the results by assessing the accuracy of the predictions using traditional metrics and draw our conclusions on the validity of the process and possible alternatives.

CCS Concepts
Applied computing → E-learning • Computing methodologies → Classification and regression trees.

Keywords
Data mining; Moodle; Log data; Grade prediction; Resource usage; Data mining process.

1. INTRODUCTION
Learning analytics have been gaining relevance as the amount and access to educational data have also been exponentially growing, and made available to educators and data scientists. Thought the topic is not recent is has been providing educators, academics, students, organizations and other interest parties with unprecedented deeper insights on technology mediated education, as we can see in [1], for example.

Currently, the data sources for conducting educational data mining vary from collecting logs from Learning Management Systems (LMS), such as Moodle, but also from other environments that have been used for educational purposes [2], such as social networks, instant messaging systems, etc. In fact, mining Moodle data is not a new research topic, thus several attempts have been made using sets of diversified data and text mining techniques to elaborate on methodologies and to construct knowledge [3] and [4].

Some studies have focused on collecting user interactions on Moodle forums to conduct social network analysis [5] and social graphs [6]. Further studies have also included the interaction of users with Moodle's activities and resources to generate node-edge graphs to provide SNA based interactive visualizations of such interactions [7].

Regarding the diversification of datasets used to extend the application learning analytics, other studies have integrated Moodle logs with Facebook groups' logs [8], mining both systems simultaneously and generating leaning analytics' visualization panels, integrated into Moodle. Also, considering instant messaging systems, other studies have also performed integrated mining using techniques such as SNA, sentiment analysis, etc., and have provided visual learning analytics [9], both expressing relations among users [10] or networked content [11].

Along with the above mentioned exploratory studies, other research focused on pattern detection and grade prediction have also been gaining relevance. This includes, for instance, mining Moodle logs to detect patterns among resource usage in order to profile students and behaviors [12], as well as creating synthetic variables that describe student behavior in resource usage in order to predict passing or failing [13].

Effectively, the complexity of the mining techniques applied to the optimization of learning analytics has been increasing. This is noticeable in recent studies that make use of special types of decision trees to predict students' final grades [14], thus there is a particular stream of research directing data mining not only to learning analytics but to predictive analytics, using forecast techniques.

However, despite the large amount of contributions that have been emerging in the field, the distinct data mining techniques that have been joined for research purposes have been introduced and depicted as a set of means to an end, according the specific desired analytics' outcomes of each context. In fact, we have not yet encountered a fully comprehensive data mining walk-through and prediction process aimed at illustrating general principles and how they are applicable to a dataset. Thus, our research focuses on outlining and illustrating the comprehensiveness of such mining process principles and its application to the current structure of the dataset Moodle logs provide.

The educational context chosen to accomplish this research purpose consists of a first-year bachelors' degree course, "Technical Communication" (DPI1001), which main objectives are to provide students with the development the skills needed to communicate technical subjects to different audiences using written, oral and multimedia presentations. The adopted teaching methodology included lectures about the writing technical reports, thesis or scientific articles.

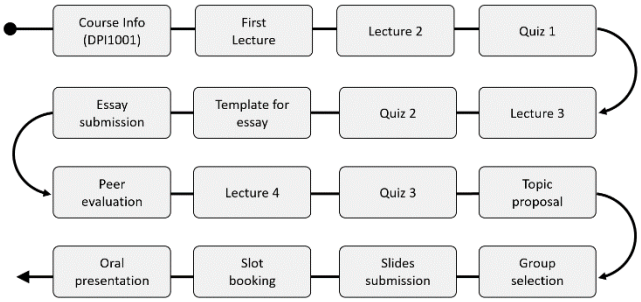

Figure 1. DPI main activities.

Therefore, considering our main research purpose and contribution is to describe the steps of this data mining process of the Moodle logs using a case study to support the general concepts.

The paper is structured as follows: we begin by describing how to obtain the Moodle logs, and hoe to prepare them for the preliminary exploratory analysis, which is described in section 3. In section 4, we describe how to identify features that can be used by a learning model and how to implement and assess it. Finally, in section 5 we synthesize our findings in the case study and draw our conclusions.

2. THE MOODLE LOG DATA

Moodle stores a basic reduced set of interactions between the user and the system. Usually these interactions correspond to a new screen display or a different central panel content. The main point here is that Moodle does not log every user interaction, nor the time that the user is using some resource. Instead, what is logged is the access to some resource or new visualization of information.

However, even with these basic set of logged events and actions it is possible to extract important information and knowledge as we show in the following sections.

2.1 Data Retrieval from Moodle

The first step to obtain the logged data from Moodle is to access the course administration and from there the option 'Reports', which has the sub-option 'Logs' as depicted in Figure 2.

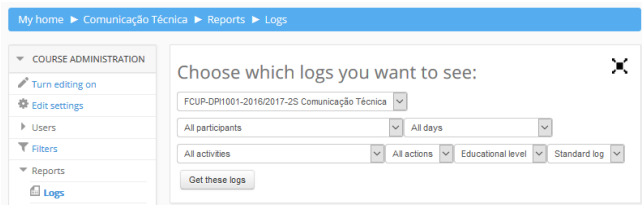

Figure 2. Accessing the reporting tool.

The drop-down boxes correspond to: 1) the course to get the logs from; 2) the participant, or all participants (or even a group of participants); 3) the day or period of days for the log report; 4) the activity to report from or all the activities; 5) the type of action performed by the course participants; 6) the access level of the participants and, 6) the log report format.

Conference'10, Month 1–2, 2010, City, State, Country.

DOI: http://dx.doi.org/10.1145/12345.67890

Actions in Moodle logs are divided into four categories: create, view, update and delete actions which are performed in activities made available either by default or by the teacher. The access level is mainly useful for situations in which we need to separate teacher actions from student actions.

The returning log data is then compiled into a csv file format or other well-known format.

The standard retrieved fields are:

- Time
- User full name
- Affected user
- Event context
- Component
- Event name
- Description
- Origin
- IP address

The retrieved data was composed of 30228 rows, each with a filled value in every of the above-mentioned column fields. These data correspond to a one semester course period for a first-year bachelor's degree.

2.2 Data Preparation

Any data mining process needs to delve in a process of data cleaning and transformation of the retrieved raw data into some format that can be used for analysis.

The very first operation was no semi-anonymize the data. We remember that in each row we had the 'User full name". We wanted to remove that field and substitute it with a unique individual reference (and Id). We used the 'description' filed to build such Id. To better understand the data transformation, we present two examples of values in field 'description':

- `The user with id '1032' viewed the log report for the course with id '419'.`
- `The user with id '3489' viewed the 'resource' activity with the course module id '62986'.`

Therefore, these values include references to user Ids and resources' Ids. Therefore, we applied the following data transformations as expressed in Listing 1:

Listing 1. Replacing full-names by Ids

1. Create new column Id
2. Fill-in the Id value with the Id mined from the description field
3. Create table with two columns (full-name and Id) and 30228 rows.
4. Fill in the table with the values taken from the original table
5. Remove duplicate cells

As a result, we built a dictionary-based table which allowed us to delete the column 'User full name'. We proceeded similarly to obtain the values for columns 'Affected user' and a new column named 'resourced Id', which refers to the resource Id that was used.

By default, Moodle logs also retrieve a 'Time' field which is comprised of the data and time for which the resource/action began to be used/taken. For a matter of data wrangling we separate this filed into two: one column for the data and another column for the hour (ie, in the format hh:mm).

Field 'event context' was also parsed to extract the type of event in which the activity was incurring and the respective 'component' field. Again, we used data transformations based on the creation of dictionaries, which in turn had their terms identified by tokenization of the string values. Examples of such values are:

- `{Referendum: Group enrollment for oral presentation} and {Course module updated}`
- `{Page: Grades in the tests} and {Course module visualized}`
- `{Assignment: Slides submission} and {The submission's state was visualized}`

In the above examples, the transformations undertaken allowed us to identify all the events and to associate them with classes of actions taken during these events.

2.3 Data Statistics

Our data preparation phase led us characterize the data. We have counted the number of distinct students accesses, of different resources being used; of events logged by the system. Then, we were able to differentiate between 4 types of actions performed by students: creations, visualizations, updates and deletions. We must stress that these numbers are not always equal to what would be expected. For example, if there are *n* students enrolled in the course one would expect to have *n* different student Ids. However, what happens is that the number is *m* (< *n*), which can be explained by the fact that some students dropped out even before the beginning of the course. The same type of situation applies for the events and resources which may not be used or experienced by all the students. The other way around is possible: there are students that do use one resource more times than those it was supposed to. For example, in one activity students must pick up a slot for an oral presentation and some of them changed it two, three or more times. Other examples are students that accessed learning material for a particular lecture several times, while others did not access them at least once.

As a synthesis, the global numbers of our collected data led us to identify as unique:

- Students: 161
- Resources: 416
- Events: 64

However, the number of these resources/events usage is not well balanced. That is, there is a small number that are much used and a lot of them that are used just once. As is clear from Figure 3 the number of different resources and of different events drops very rapidly forming the standard 'long tail' pattern.

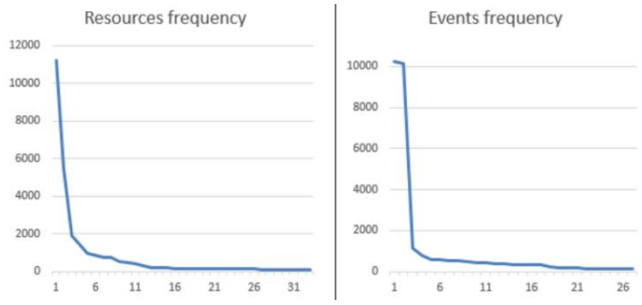

Figure 3. Different resources and events frequency

As a consequence, we decided to consider only resources and events that were actually used/experienced at least 100 times. This assumption led us to the numbers:

- Different resources to consider: 33
- Different events to consider: 27

We now analyze the types of actions performed by all the course participants.

Table 1. Number of types of actions by role.

Actions	AllParticipants	Teacher	Students
Create	2886	388	2498
View	25117	1621	23496
Update	2084	954	1130
Delete	147	87	60
Total	*30234*	*3050*	*27184*

The number of visualizations is one order of magnitude above creations and updates, which in turn are on order above deletions. The proportions are similar either for the teacher and for the students. This situation clearly derives from two facts: the first is that for accessing every creation, update or delete page, we need first to visualize it; the second, relates to the natural curiosity of people, associated with the fear to "act" in spite of just "observe".

3. EXPLORATORY ANALYSIS

Having the stabilized set of data, we performed an exploratory analysis which comprehended verifying the distribution of values in each field, the detection of possible outliers, the detection of NA (not available) values. This analysis is also important to 'understand' the data and have a first feeling on how the values may be related and distributed. Our standard analysis performed on all features is to identify the minimum, the maximum, the average, the median and the number of NA values. We complement the analysis with a graphical representation of the distribution of values using a column chart for the frequency of the values, which are represented in figures 3 to 8.

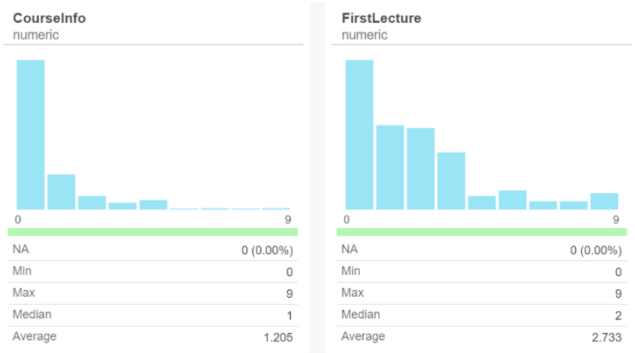

Figure 4. Course info and 1st lecture access.

As we can see most students had a least one access to the course info material and to the first lecture. It is important to explain that the handouts from the first lecture include important information about the course structure, the evaluation methodology and the assessment. Therefore, it is reasonable to expect this material to have slightly more accesses than the other one. Curiously, there is at least one student for of these handouts that has accessed it 9 times. Nevertheless, we see that the distribution of accesses is much smoother for FirstLecture, leading to a more diversity of behavior.

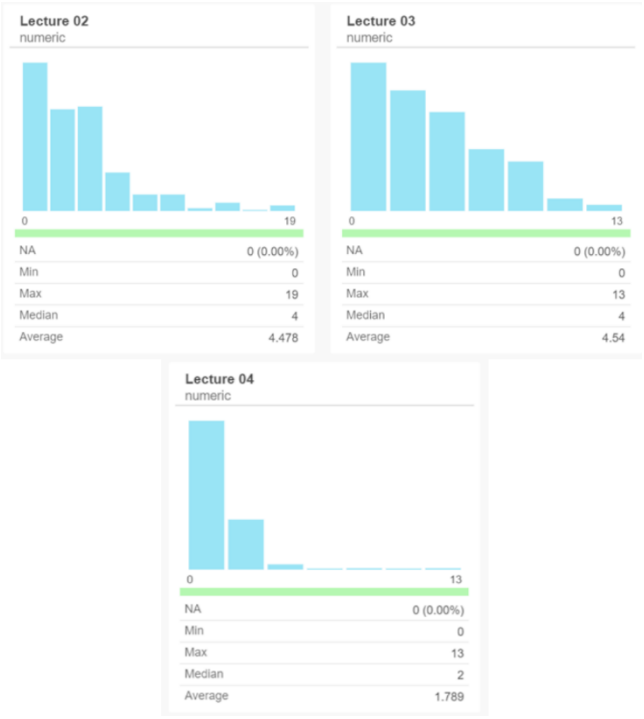

Figure 5. Three main lectures handouts access.

In Figure 5 we can compare the accesses to material from the lectures that preceded the three quizzes. It is clear that lectures 2 and 3 with more than 4 accesses on average were perceived as more important to students. This characteristic extends also to the distribution of values which is much more condensed in accesses to lecture 4 (mainly around 2 accesses) than in the other two lectures, with a small better distribution in lecture 3. This situation will be reflected when we expose the correlations between features and grades but also with the "Gini factor", when assessing the learning model.

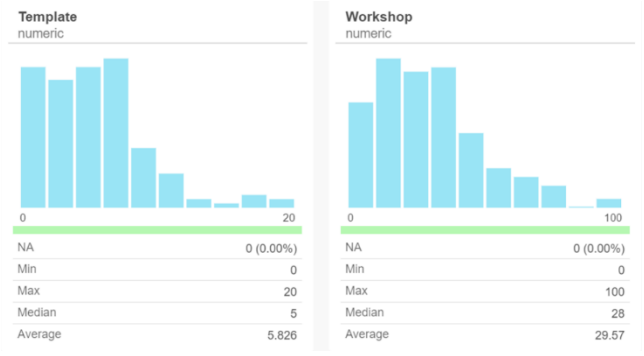

Figure 6. Article template and workshop access.

In Figure 6 we can compare the access to the provided template for the students to write an essay and the workshop module from where they had to submit it and evaluate the work of their peers. Although it is reasonable to admit that a student may access the workshop module on average about 30 times (considering the possible several own submissions and the grading of their peers). Note that each student that had made a submission of an essay had to assess the essays of four other colleagues. Therefore, it is fair to expect that this process didn't complete in the same "run" and several accesses to the module were needed. On the other hand, it is much more difficult to understand why students had to access a single template more than 5 times on average.

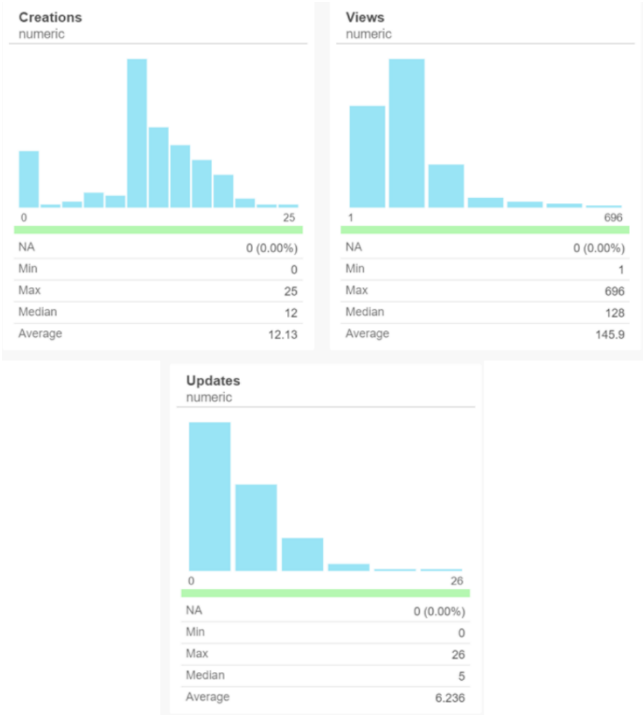

Figure 7. Frequency of different actions.

As for the 'actions', we decided to comment only the three with most expressive in numbers: the creations, views and updates (cf. Figure 7). It is interesting to notice that in 'the creations' there is a near-to-similar distribution to "normal", and, in fact, the average value stands in the middle of the range. Views have largely more accesses, but the distribution is so right skewed that there must be many students with only one or few visualizations. The updates reflect the traditional distribution of this class: very right-skewed with many students having 0 updates. This means that these students didn't do any "improvement" of their work, either in the form of a re-submission of their essay or of the slides, or even when assessing their peers. It either is trait of very zealous and devoted group of students or it is a trait of quite the opposite: a group of students that really don't mind if they find a better solution or if they submit just in time that there is no margin for improvements.

The remaining features have a very similar distribution of accesses for CourseInfo, i.e., it is intensively right-skewed with zeros at the left. We replace the representation of their values in a chart by presenting in Table 2. The remaining variables.

Table 2. The remaining variables.

Activity	Max	Median	Average
GroupSelection	90	6	8.453
ProposeTopic	98	4	9.894
SlidesSubmission	38	1	3.217
SlotBooking	30	1	2.553
GradesInComponents	10	0	0.8944

GroupSelection is an activity in which students select colleagues for team work in such a way that a previous agreement between group participants is needed before making the online selection.

Therefore, the bigger the number of accesses the less agreement was established offline. Accesses in the order of tens are clearly an indicator of indecision and bad planning for group creation.

On the other hand, if we consider activities that reveal a proactive action from students, the average of accesses is dramatically reduced. For example, there was almost no resubmission of slides (students did not want to improve their work), the most common number of accesses to the topic proposal forum was 4, which a very small number if we consider that every student should propose a topic, a number close to 160 would be much more reasonable.

Grades in components is just a page explaining the grade obtained by each student in each assessment component of the course. Therefore, its access by students would be motivated by curiosity or an intention to understand where did the perform worse in order to improve. Being one way or another, the average number of accesses to this page was less than once per student.

4. MODEL CREATION

4.1 Feature identification and creation

In the previous section we identified the resources and events that were mostly accessed and experienced by the students. To this set we added a set of features composed of types of actions performed by students (creations, visualizations, updates and deletions). This all set is therefore comprised of 19 features which we list below:

- DPI1001-2016/2017-2S
- Forum: Forum News
- UC info
- File: DPI info
- File: Lecture02
- File: Lecture03
- File: Lecture04
- Page: Model / Template for the essay
- Workshop
- Page: Grades in the tests
- Group Selection
- Forum: Propose topic for presentation
- Assigment: Slide submission
- Slot booking for oral presentation
- Page: grades in components
- Creations
- Views
- Updates
- Deletions

As our goal is to have a system capable of predicting grades based on the use of the activities, as logged by Moodle, we want to verify the correlation between these features with our dependent variable – which was the actual "Grade" that the student obtained.
We used the Pearson correlation to perform all pairs of correlations between all the features and variable Grade. The results are listed in Table 3:
According to Pearson's model, number of "Creations" and "Workshop" accesses have a strong correlation with the obtained grade. As expected, a "creation" activity involves not only a commitment by the student but also a responsibility in face of their peers. It is fair to assume that the best students are more prone to creations than lower-grade students.
Without entering into too much detail it is interesting to note that access to Lecture 3 was a more important feature to distinguish students' grades than Lecture 4 and 2.

Table 3. Pearson's correlations.

Correlations with "Grade"		
Creations	0.758	Strong
Workshop	0.629	
Updates	0.580	Moderate
Template	0.575	
Views	0.567	
Lecture 03	0.476	
DPI1001 info	0.471	
Lecture 04	0.374	Weak
ProposeTopic	0.360	
Lecture 02	0.359	
GroupSelection	0.344	
GradesInTests	0.313	
SlidesSubmission	0.312	
SlotBooking	0.308	
CourseInfo	0.296	
GradesInComponents	0.247	
FirstLecture	0.235	

This analysis led us to create a model featuring the 17 variables listed in the Table 3, plus one column for the actually assigned grade to the student.
We then proceeded to create our base table to train a model for prediction. This table should be comprised of all the valid available observations for all the columns gathered as important for the analysis. Therefore, our base table should be created with one row per student; as we have 161 different students, the table is of size 161 × 18 cells.

4.2 Training the model

We used random forests as a learning algorithm because we wanted later on be able to understand some of the decisions taken by the model. Use used a split of 0.25 between training and testing phases which left us with 120 rows for training and 41 rows for testing.
The split was made picking rows randomly between the first set.
It must be noted that the grade to predict is understood as a label and therefore, a prediction of 15 to a real value of 16 is not better than a prediction of 12. For this type of learning models what matters is the real precision and accuracy of the system.

4.2.1 Dealing with classes with empty samples

However, we had another important problem which is the lack of classes to predict. In Figure 8 we show the histogram of the given grades in DPI1001 for 2017. A value of zero does not necessarily means that the student got a null for his/her work. It mainly does mean that there is some missing component of the evaluation procedure that is missing and, therefore, he/she cannot complete the course until this/these component(s) have been completed.

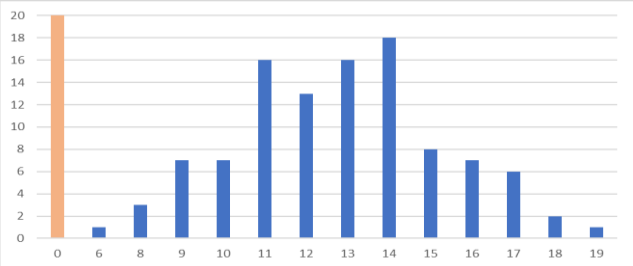

Figure 8. Histogram of grades.

It is important to note that not every point in the normal scale from 0 to 20 if filled. That is, for example there hasn't be given a max grade, and grades between 0 and 5, and 7 are also missing (the whole scale was not used while grading students).

Therefore, the learning algorithm cannot 'learn' to predict labels it has not been presented before (it has not knowledge about). In Table 4 we present the results of the evaluation of this preliminary model. Note that 'Acc' stands for accuracy, 'MR' for miss rate, and 'size' for the sample data size for the given class.

Table 4. Evaluation of Training Data.

Class	F score	Acc	MR	Prec	Recall	Size
0	**0.683**	0.896	0.104	0.667	0.700	20
6		**0.992**	0.008		0.000	1
8		0.976	0.024		0.000	3
9		0.928	0.072	0.000	0.000	7
10		0.928	0.072	0.000	0.000	7
11	0.410	0.816	0.184	0.348	0.500	16
12	0.133	0.792	0.208	0.118	0.154	13
13	0.194	0.800	0.200	0.200	0.188	16
14	0.245	0.704	**0.296**	0.194	0.333	18
15	0.125	0.888	0.112	0.125	0.125	8
16	0.200	0.936	0.064	0.333	0.143	7
17		0.936	0.064	0.000	0.000	6
18		0.976	0.024	0.000	0.000	2
19		**0.992**	0.008		0.000	1

As we can see there are cells which are empty because their values could not be computed. This situation is due to the non-balanced number of observations between classes. It is no surprise that class '0' has the best f-score and precision because it also has the highest number of samples. On the other hand, class '6' has an impressive accuracy of 0.992 because there is a single sample, therefore the model overfitted its performance to include this case.

In order to obtain a balanced number of observations amongst the classes we needed to create new, artificial, classes such that the variance on the number of observations belonging to each class would roughly be the same.

Table 5. Evaluation of training data (balanced classes).

Class	F Score	Acc	MR	Prec	Recall
2	0.087	0.826	0.174	0.100	0.077
3	0.100	0.851	0.149	0.111	0.091
4	0.071	0.785	0.215	0.063	0.083
5		**0.893**	0.107	0.000	0.000
6	0.357	0.851	0.149	0.333	0.385
7	0.054	0.711	**0.289**	0.045	0.067
8		0.843	0.157	0.000	0.000
9	0.273	0.868	0.132	0.250	0.300
10	0.160	0.826	0.174	0.154	0.167

We used an automatic categorization method, to create 10 classes based on equal frequency (quantile) of observations. With the new classes run again the random forest learning model, with a split between train and test of 25%, and we allowed final node sizes of 1 single value.

The results are expressed in Table 5. We removed the first class which was concerned with the zeros. This kind of situations to not present challenges for prediction because they deal with the completion or not completion of mandatory evaluation components.

The new table does not have the data size column precisely because all the categories have the same number of observations. The down side is that the category-labels do not reflect the actual grade of the students. On the other hand, we now see many more categories with values computed. Using the "Mean decrease Gini" index, represented in Figure 9, we see that the Views were an important feature to create splits, followed by the Workshop, curiously by the accesses to course info, then by the Creations, and the updates complete the top5.

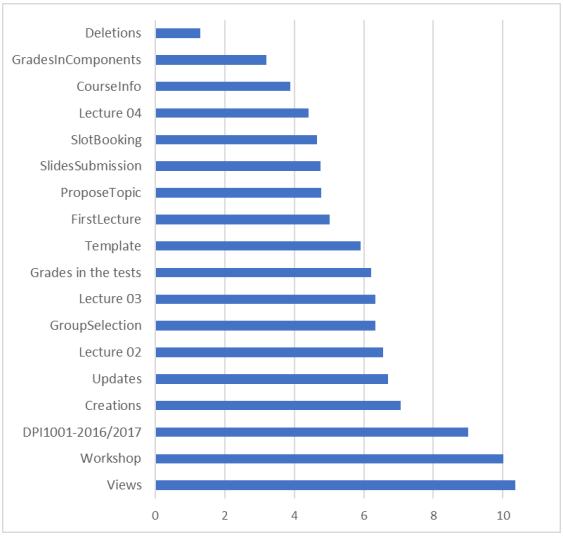

Figure 9. Mean decrease Gini index.

4.2.2 The Decision Tree

When using decision trees split decisions take into consideration the variable that best separate groups and the information gain from that separation. Without a global panorama of all the information splits can lead to misleading trees (actually, this is why random forests are best suited for these cases). Our algorithm created a tree that can be described by the following output:

Listing 2. R description of a classification tree for the train set

```
node), split, n, deviance, yval
          * denotes terminal node

1) root 125 3484.80000 10.240000
     2) Deletions< 1.5 111 3209.02700  9.810811
       4) Lecture04>=1.5 55 1560.43600  8.654545
         8) Lecture02< 5.5 29  922.96550  7.034483
          16) Workshop>=35.5 11  312.00000  4.000000 *
          17) Workshop< 35.5 18  447.77780  8.888889 *
         9) Lecture02>=5.5 26  476.46150 10.461540
          18) GroupSelection< 8.5 13  330.76920  8.692308 *
          19) GroupSelection>=8.5 13   64.30769 12.230770 *
       5) Lecture04< 1.5 56 1502.83900 10.946430
        10) ProposeTopic>=10 9  420.22220  7.555556 *
        11) ProposeTopic< 10 47  959.31910 11.595740
          22) ProposeTopic< 0.5 25  733.04000 10.280000 *
          23) ProporTopic>=0.5 22  133.81820 13.090910
            46) Template>=4.5 11   56.54545 11.636360 *
            47) Template< 4.5 11   30.72727 14.545450 *
3) Deletions>=1.5 14   93.21429 13.642860 *
```

Listing 2 represents a description of a tree as produced by the R programming language (in the case using the rpart library). The tree should be read from top to bottom, the beginning of each line represents the label of a node and indentation is used to denote child nodes. After the node label comes the condition, the number of observations at the node, the loss or error at the node (not normalized) and the predicted class.

This particular tree used only 7 features (Lecture 2, Lecture 4 Deletions, GroupSelection, ProposeTopic, Template and Workshop). However, with different hyperparameters it would be possible to obtain a different tree configuration.

Now this model should be tested against the test-sample to verify the accuracy and precision of the results of the model. More than certainly we don't get results of 100% in both parameters nor even in a single one. However, by 'tuning' up the system we may increase the accuracy and precision of the model up until a threshold which may be acceptable for as a good predictive model.

The evaluation of the model is made according to the number of true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). Precision is defined as TP / (TP + FP) and accuracy is defined as (TP + TN) / (all cases).

4.2.3 Using other learning algorithms

In the previous section we described a decision tree trained with a sample of 125 observations (leaving the rest for the testing). This could be done with different percentages, for example 80%-20% or 70%-30% are also common splits. Moreover, we tuned the testing to accept certain information gain situations. These are all hyperparameters that may be tuned in order obtain a model that has better results either on training but also on the testing phase.

In the case of this paper that tuning is out of scope because it loses generality on the data mining process, in particular during the machine learning fine tuning. We know for own experience that there isn't a model that "fits all", and Kaggle competitions are the best example of that.

5. CONCLUSIONS

We have described a process for the prediction of grades based on a data mining methodology which includes a machine learning step, considering its training, testing and fitting phases. Our approach was to use a case study of Moodle logs taken from a first-year higher education course – "Technical Communication" – for which about 33k records were available, created from a total of 161 different students. This data set allowed us to undertake the process of data extraction and exploratory data analysis. We then explained how to create a basic table composed of features for model creation which comprehends a training and a testing phase, and finally we pointed out some possibilities to tune the model in order to better predict both the values in the training, but particularly in the testing phase.

5.1 Conclusions from the case study

In this subsection, we present a synthesis of the conclusion we can draw from our research on the particular case study that was presented:

- It was expected that student acceded once to the course info and to the first lecture, where most of the course description, assessment methodology and syllabus was presented. However, some students didn't access these materials while other acceded them 9 times.
- The accesses to handouts from principles on writing technical documents (lecture 2) and handouts on evaluation of these documents (lecture 3) have similar average number of accesses, while handouts about electronic presentations have a much-reduced number than the first two. We believe students think the part of the course was easier for them.
- Students had to many accesses to the template, where once could be enough. This is almost incomprehensible unless they do not keep an own storage are of files for the course.
- Number of accesses to the workshop and creations is fair, however, when comparing with the amount of visualizations we understand that there is some fear to commit mistakes and a poor proactivity.
- The reduced number of updates indicates that there is not a generalized will to improve their work or that the submissions are made just by the deadline.
- The group selection took too long to be made which is also an indicator of poor offline planning and agreement between members.
- The propose topic as an incredible low number of accesses (about 10 on average), while it would be expected to be more than 160. This means that the activity totally failed in its purpose to captivate students' attention to their peers' proposals and teachers' comments.

5.2 Contributions

As contribution from this work, we believe that using our case study we could explore a series of important data mining concepts and procedures, while in a process leading to a prediction activity. We addressed the topic of anonymizing the data, and of creating context between data present in different fields by data transformation mechanisms. We discussed an important part of data cleaning when determining which data to exclude from our dataset using the 'decreasing graphs'. We used the grasped dimension reduction by using correlations and re-categorization procedure to approach the important aspect of unbalanced classes in machine learning. We then applied a decision tree learning algorithm capable of explaining its created model (as opposed to 'blackboxes' as in neural networks) and touched the methodology to derive the prediction capability of the model and how to improve it.

6. ACKNOWLEDGMENTS

This work is supported by the ERDF – European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT (Portuguese Foundation for Science and Technology) within project «Reminds/ UTAP-ICDT/EEI-CTP/0022/2014»

7. REFERENCES

[1] Cela, K. L., Sicilia, M. Á., & Sánchez, S. (2015). Social network analysis in e-learning environments: A Preliminary systematic review. Educational Psychology Review, 27(1), 219-246.

[2] Cavus, N. and Zabadi, T., 2014. A comparison of open source learning management systems. Procedia-Social and Behavioral Sciences, 143, pp.521-526.

[3] Dawson, S., 2010. 'Seeing'the learning community: An exploration of the development of a resource for monitoring online student networking. British Journal of Educational Technology, 41(5), pp.736-752.

[4] Jamali, M. and Abolhassani, H., 2006, December. Different aspects of social network analysis. In Web Intelligence, 2006. WI 2006. IEEE/WIC/ACM International Conference on (pp. 66-72). IEEE.

[5] Scott, J., 2017. *Social network analysis*. Sage.

[6] Figueira, Á. and Laranjeiro, J., 2008, October. Work in progress—iGraphs for characterization of online communities.z In Frontiers in Education Conference, 2008. FIE 2008. 38th Annual (pp. 10-13). IEEE.

[7] Silva, A. and Figueira, Á., 2012, April. Depicting online interactions in learning communities. In Global Engineering Education Conf. (EDUCON), 2012 (pp. 1-8). IEEE

[8] Oliveira, L. and Figueira, Á., 2016. EduBridge Social-Bridging Social Networks and Learning Management Systems. In CSEDU (1) (pp. 162-171).

[9] Poon, L.K., Kong, S.C., Yau, T.S., Wong, M. and Ling, M.H., 2017, June. Learning Analytics for Monitoring Students Participation Online: Visualizing Navigational Patterns on Learning Management System. In *International Conference on Blended Learning* (pp. 166-176). Springer, Cham.

[10] Oliveira, L. and Figueira, A., 2017, April. Visualization of sentiment spread on social networked content: Learning analytics for integrated learning environments. In *Global Engineering Education Conference (EDUCON), 2017 IEEE* (pp. 1290-1298). IEEE.

[11] Silva, A. and Figueira, Á., 2012, July. Visual analysis of online interactions through social network patterns. In Advanced Learning Technologies (ICALT), 2012 IEEE 12th International Conference on (pp. 639-641). IEEE.

[12] Ziebarth, S., Chounta, I.A. and Hoppe, H.U., 2015. Resource access patterns in exam preparation activities. In Design for Teaching and Learning in a Networked World (pp. 497-502). Springer International Publishing.

[13] Figueira, A., 2015. Predicting results from interaction patterns during online group work. In Design for Teaching and Learning in a Networked World (pp. 414-419). Springer.

[14] Figueira, Á., 2017, April. Communication and resource usage analysis in online environments: An integrated social network analysis and data mining perspective. In Global Engineering Education Conference (EDUCON), 2017 IEEE (pp. 1027-1032). IEEE.