Finding Association Rules in College Course Progression

Pedro Melgueira, Luís Rato

Universidade de Évora
pedromelgueira@gmail.com,lmr@di.uevora.pt

Abstract. Association Rules are a data mining technique that aims at finding patterns in data that explain how different elements of the data influence each other. In this project this technique is used to find associations that describe the trends of college course completions and results. This project used a dataset from the University of Évora for rule finding. This paper shows how the dataset had to be preprocessed first in order to be mined, and describes the techniques and algorithms used.

Keywords: Association Rules, Data Mining

1 Introduction

College courses in a given curriculum influence each other in terms of how they are completed, with what grades, and at the same time as what courses. These influences are often observed among students with course completions with similar grades, always failing some courses while completing others, and other patterns. The interest in finding these relations comes from the need to characterize a curriculum as a whole and try to identify what are the trends in course completion among students.

The data used in this project came from the Department of Informatics of the University of Évora. This data contains information on course completions and failures arranged by student and year. From this data it is possible to determine many patterns concerning the trends in completion associated with resulting grades. The dataset had some deficiencies that needed to be address, before any study could be ran on them. They are described in Section 2.

To find such patterns, a data mining technique called Association Rules is used. Association Rules are defined in [5] and [6] and have been used in other situations in [4, 2, 1]. Section 3 explains how these rules are used and how they were implemented in order to get results. Some experiments were done to the dataset to find Association Rules. The experiments, and their results are discussed in Section 4. Section 5 draws some conclusions.

2 Dataset and Preprocessing

The dataset used in this project originated from the records of the Department of Informatics from the University of Évora. The records contain a listing for every course completion or failure from the students of that department. Such records have been generated over the course of several years in the department. The older entries date from 1995. There are entries for the three cycles of study, which are the Licentiate, Masters, and PhD degrees.

The records are compiled into a dataset which associates information on a student, a course and the results of completing that course or not, so there is an entry for every time a student has completed a course along with the final grade it had, and for every time a student has failed a course. The total number of entries in the dataset is 52264. Not all data entries are useful. Counting the number of entries for each of the three study cycles, 49461 entries are found to be first cycle entries, while only 2438 and 366 are found to be second and third cycles entries, respectively. Because their number is so low, these entries are removed from the dataset, so any analysis will only be made to the courses of the first cycle.

The fields of the dataset are shown in table 1. Note that the names of these fields have been translated from Portuguese, the original language, to English. The fields containing information about the students must be made anonymous for privacy reasons. The field *Student Name* is simply removed and each student number gets replaced by a new integer number.

Table 1. The fields of the original dataset.

School Year	Degree	Department
Course Code	Course Name	Regime
Credits	School Course	Edition
Speciality	Semester	Season
Type	Student Number	Student Name
Student Type	Grade	Result
Final Result		

Some of the fields are considered useless. They are the *Course* field, which only has a course code, the *Department* field, and the *Degree* field, which contains the same information for every entry of the first cycle. The fields *Edition* and *Speciality* don't have any information in first cycle entries, so they are also useless. Because of these reasons, these fields are removed from the dataset.

Because the dataset was built over many years, the differences in how records are kept has changed from time to time. Noticeably, around the years of 2006-07, after the Bologna Process was signed, the changes to each field are substantial. As consequence of the changes, each field of the dataset doesn't have a specific domain. A simple Python script was implemented to determined the domain of each field in the dataset.

The first deficiency found dealt with student numbers. In early records the numbers for every cycle were an integer number, but in post Bologna records some numbers have a letter prepended to it, which relates to the cycle the student is in. Another deficiency dealt with the names and codes of courses which change over the years. Same courses exist in both early and late records in the sense that they share the same name, but they are considered to be different because in most cases the curriculum of those courses changed.

Some entries had courses that appeared in the dataset only once or twice. It was determined that these are entries from students who changed from other licentiate degrees to study Informatics. Because these entries are so rare and only added noise to the data, they are simply removed from the final dataset.

The last 3 fields, *Grade*, *Result*, and *Final Result*, seem redundant. First, a grade should always be an integer number ranging from 0 to 20. If the final grade of a course is greater or equal to 10, then the student is approved, otherwise the student fails. Whether the student gets approved or not should be written in the *Result* field. But this is not observed in the dataset, there are many entries that have the grade as a missing value and some even have negative values. In cases where a student doesn't get approved, the result field may have the reason for him not getting approved, be that because he skipped the evaluation, because he quited, etc.

To avoid confusion, and to have a straightforward version of the dataset, the preprocessing took every entry and defined that if that entry had a positive grade, then the entry gets rewritten with that grade and the value *Approved* in the *Result* field. If it has something else, then it is assumed that the student didn't get approved. The fields for grade and result will always have the values 0 and *Not Approved*. The *Final Result* field is redundant and simply removed from the dataset.

Preprocessing was made with a Python script. The preprocessed dataset contains the fields listed in table 2.

Table 2. Fields of the preprocessed dataset.

School Year	Course Code	Course Name
Regime	Credits	Semester
Season	Type	Student Number
Student Type	Grade	Result

3 Association Rules

3.1 Rules

Association rules are a data mining technique that aim at finding probabilistic associations between events. One of the first uses of such a technique [5] dealt with looking into data from supermarket sales. By taking into account several sales, it was possible to find hidden directional relations between sold items, for example, it was determined that people who bought beer also bought diapers, but not the other way around.

Finding this kind of rules may be useful for several reasons. In this project the objective is to find how the behavior of completing certain courses determines the behavior of other courses. The experiments done are details in section 4.

Each association rule has the form

$$A \to B,$$

where A and B are non-empty sets of items. The rule is read as, "if the items in set A are observed, then there is a good probability that the items in set B are observed". In the supermarket example stated, the rule would be read as, "if the items in A are sold, then the items in B are probably sold to".

To calculate association rules from a dataset, first the data must be organized into baskets. In the supermarket example, each basket would simply be the items in a sale. Generally, a baskets is simply a set of items that have some important relation. The way baskets are constructed may be different in each experiment made to the same dataset. Section 4 details how baskets are constructed for each experiment in this project.

3.2 Frequent Item Sets

Having all the baskets, the occurrence of each item in the baskets is counted. A item shouldn't appear more then once in a basket because they are sets, so the expected count should be any value from 0 to the total number of baskets. The support of an item is defined as the ratio between the count of a set containing only that item and the total number of baskets,

$$sup(\{i\}) = \frac{count(\{i\})}{N}, \tag{1}$$

where $sup(\{i\})$ is a function that represents the support of set $\{i\}$, $count(\{i\})$ is a function that represents the count of set $\{i\}$, and N is the total number of baskets. An item is said to be frequent if its support is above a certain threshold S_i.

Once the single items are counted and their support is calculated, the same process is done for pairs of items. The pairs in question are seen as all the combinations of items. To simplify, only pairs made out of frequent items are considered. This is done because a frequent item set can't be more frequent then the items that make it up. The support for an item set $\{I\}$ is calculated using equation 1.

Like before, an item set is frequent if its support is above some threshold S.

The same process could be repeated for sets with three items. From there item sets with more items could be found. In this project only sets with two items are searched because there isn't enough data to find higher order item sets with reasonable support levels.

3.3 Association Rules From Frequent Item Sets

Having a list of frequent item sets, the association rules are calculated. From an item set (a, b) the rules $a \to b$ and $b \to a$ may be constructed. Just because an item set is frequent, doesn't mean that the rules constructed from them are usable. So there are three measures that must be calculated for every rule. The first measure is the support of the rule, which is equivalent to what is done in equation 1.

$$sup(x \to y) = sup(\{x, y\}).$$

The second measure is called confidence and it is define as,

$$conf(x \to y) = \frac{sup(x \to y)}{sup(\{x\})}.$$

Confidence is an estimate of the probabilistic value $P(y \mid x)$. Rules with a confidence value close to 1 are considerate to be strong rules, for their antecedent strongly implies their consequent.

The last measure if lift, defined as,

$$lift(x \to y) = \frac{sup(x \to y)}{sup(\{x\})\ sup(\{y\})}.$$

As stated in [5], "The lift of the rule relates the frequency of co-occurrence of the antecedent and the consequent to the expected frequency of co-occurrence under the assumption of conditional independence.". A value for lift equal to 1 indicates the two items in the rule are independent, if the value is greater than 1 the rule indicates a positive co-occurrence. The higher the lift, the stronger the rule is.

Once these measures are calculated for each rule, the useful ones must be selected. Rules that have low confidence, or a value for lift close to 1 are not useful. A useful rule is a rule that has the values of support, confidence and lift above certain thresholds. However, there is no explicit general way of finding useful values for these three measures. In this project, because the dataset wasn't so big, the values for these parameters were adjusted by hand to limit the amount of rules yielded by the algorithm. In same cases, as it will be seem, the majority of rules yielded by the algorithm have values that are too low to be taken into account.

The method used to calculate the association rules follows the A-Priori algorithm. In order to have an efficient solution for this project, an implementation of this algorithm was made from scratch using the Python programming language. This implementation allowed different experiments to be done specifically for this dataset without having to change anything in the source or the dataset, therefore providing a flexible solution for the study at hand.

4 Tasks and Experiments

The experiments made are organized by tasks. Each task specifies a way the preprocessed dataset was turned into baskets and how the experiments were made with each particular basket. As mentioned before, the courses listed in the dataset change significantly after the Bologna process. Because of that the described experiments were first executed for the whole dataset, and then for a portion of the dataset that only contained entries listed after 2006.

The complete results for these experiments are hosted at [3]. The entries are kept in their original language, Portuguese. The following section presents some tasks and discusses their results.

4.1 Task 1

Something that should be expected from any given student is that if that student completes a course with certain grades, then he will very likely complete similar courses with similar grades. For example, if a student completed the course Programming I with grade 18, then probably he will complete Programming II with a grade similarly high. Task 1 intends on finding similar rules. A rule found in this task should be read as, "If a student finishes course x_1 with grade y_1, then he will probably finish course x_2 with grade y_2". Both the antecedent and consequent of those rules are a single item. Each item is a compound between a course and a grade. All these items belong to a single student, therefor, each baskets represents all the course completions for a single student. Given that only course completion matters in this task, there will only be items with grades greater or equal to 10.

The domain for the grade is an integer value from 10 to 20. It was considered that the domain was to big, so these values were replaced by grade classes, which are $[0, 10[$, $[10, 13[$, $[13, 16[$, $[16, 19[$, and $[19, 20]$.

After executing the first experiment, it is observed that the vast majority of rules found are all rules with items with grade class $[10, 13[$. A second experiment was done which only allowed items with grades greater or equal to 13, therefore excluding the first class. The parameters for both experiments are in table 3. The value for confidence is lower on the second experiment because of fewer data entries.

Table 3. Task 1 parameters

Exp.	1	2
(S_i) Single Support	0.1	0.1
(S) Support	0.1	0.1
(C) Confidence	0.4	0.1
(L) Lift	2	2

The results for the first experiment with all data entries showed that there is a high probability that the courses for Physics II and Mathematical Analysis I will be completed with a grade class of [10, 13[given that some other courses were also completed with that grade class. In fact this trend is observable until the 24th rule.

In the results for the dataset with entries after 2007 it is seen that there aren't as many rules with similar consequents. But the course Declarative Programming shows up with some frequency as a consequent in the rules with highest confidence.

Something true for both results is the fact that nearly all the rules found have a grade class of $[10, 13[$ in the antecedent and consequent. Because of this, the second experiment was made to find rules out of this grade class. The results of second experiment, however, show a majority of rules in which both the antecedent and consequent are courses finished with a grade class of $[13, 16[$. No real trend in course completion was noted in the second experiment.

4.2 Task 2

Task two tries to identify patterns in the approval results of two courses which were taken at the same time. The objective with this task is to find courses that are incompatible, meaning that if a student tries to make course A and B at the same time, he will probably fail one, or if a student completes course A, then he will probably complete course B in the same semester, or fail course C also in the same semester.

Each basket of this task contains items referring to a single student, in a single school year in one of the two semesters. Each item in the baskets is a compound of a course and a result, with the result being approved or not approved.

To approach these questions three experiments were made. The first experiment takes into account both approved and non approved courses. Because the dataset has a greater number of non approved entries then approved ones, a second experiment was made only with approved data entries. A third experiment was also made only with non approved data entries. Table 4 contains the parameters used in each experiment.

Table 4. Task 2 parameters

Exp.	1	2	3
(S_i) Single Support	0.05	0.01	0.05
(S) Support	0.05	0.01	0.05
(C) Confidence	0.4	0.4	0.4
(L) Lift	1	2	2

In both datasets, the calculated rules in the second experiment show that almost all the rules found relate a course which was not approved with another course also not approved. It is observable that almost every rule contains courses from the first year. This may be influenced by the fact that first year courses are usually attended by many students who never graduate and end up dropping out.

In the second experiment, containing only approved entries, in the dataset after 2007, it is observable that almost all the yielded rules contain courses from the third (and last) year as their consequent and antecedent. This shows that most students tend to finish those sets of courses together and there isn't much overlap between completing courses from the last year with courses from other years. No rules were found that contained courses of different years, something that was actually unexpected. A example that was to be expected in this regard were rules that associated harder courses of one year being completed along with easier courses of the following year, but no rule like this was found.

Lastly, the results of the third experiment were not much different from the rules of the first experiment.

5 Conclusion

This project had two important parts, one related to preprocessing and the other to Association Rule mining.

When working with datasets that were not originally built to be mined in specific ways, preprocessing is never a direct and trivial step. In this project it was showed how a dataset from the University of Évora was examined in order to be usable later. The resulting preprocessed dataset is anonymous, and has a consistent method for displaying student grades and approval, unlike the original one. The preprocessed dataset also does not have redundant fields, or fields which do not have relevant information. Lastly, the preprocessed dataset was free from entries with added noise to the data without being usable, for example, the entries with courses which only appeared once or twice.

This project presented a way to use Association Rules to find trends in how college courses are completed or failed. The rules found, displayed certain trends that were to be expected, such as the strong association found in the failure of first year courses and completion of third year courses. Also, some trends that were to be expected were also shown not happen. From the proposed tasks and experiments in this article, more tasks and more complex experiments can be made. For example, one could ask to find Association Rules with the antecedent being an approved course and the consequent being a failed course.

This project can be used as a basis for similar experiments with data from different curriculum and different universities. If there are similar datasets with more students and which span more years, then it might be possible to find more interesting trends in course completion.

References

1. Bernard Kamsu-Foguem, Fabien Rigal, and Félix Mauget. Mining association rules for the quality improvement of the production process. *Expert Systems with Applications*, 40(4):1034 – 1045, 2013.

2. Beatrice Lazzerini and Francesco Pistolesi. Profiling risk sensibility through association rules. *Expert Systems with Applications*, 40(5):1484 – 1490, 2013.
3. Pedro Melgueira. Results discussed in this paper, January 2015. https://gist.github.com/petermlm/9bfed104ca706ff4e8de.
4. Alfonso Montella. Identifying crash contributory factors at urban roundabouts and using association rules to explore their relationships to different crash types. *Accident Analysis & Prevention*, 43(4):1451 – 1463, 2011.
5. Anand Rajaraman and Jeffrey David Ullman. *Mining of Massive Datasets*. Cambridge University Press, New York, NY, USA, 2011.
6. Mohammed J. Zaki and Wagner Meira Jr. *Data Mining and Analysis: Fundamental Concepts and Algorithms*. Cambridge University Press, New York, NY, USA, 2014.