Finamore et al. Journal of the Brazilian Computer Society (2020) 26:3
https://doi.org/10.1186/s13173-020-00097-0

Journal of the
Brazilian Computer Society

RESEARCH Open Access

A comparative analysis of two computer science degree offerings

Anna Carolina Finamore[1], Haydée G. Jiménez[2], Marco A. Casanova[2*], Bernardo P. Nunes[2,3], Ana Moura Santos[4] and António Pacheco Pires[5]

* Correspondence: casanova@inf.puc-rio.br
[2]Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ, Brasil
Full list of author information is available at the end of the article

Abstract

This article presents an in-depth analysis and comparison of two computer science degree offerings, viz., the Bologna BSc in Information Systems and Computer Engineering, offered by the Instituto Superior Técnico of the University of Lisbon, Portugal, and the BSc in Computer Science offered by the Pontifical Catholic University of Rio de Janeiro, Brazil. The analysis is based on the student transcripts collected from the academic systems of both institutions over circa one decade. The article starts with a description of the degrees and global statistics of the student population considered. Then, it presents a comparative analysis of the curricula, which focuses on how close students follow the recommended curricula, based on data visualization techniques and academic performance indexes. The indexes indicated a mismatch between the semesters that the curricula recommend for the courses and the semesters that students enroll in those courses. Furthermore, a visualization of course advances and delays indicated that a significant fraction of the students failed in the semester that the curricula recommend for the courses. The article moves on to present a comparative analysis of student performance in individual courses, and then applies a technique borrowed from Market Basket Analysis to investigate student performance in multiple courses that are taken in the same semester. The analysis pointed out sets of courses, at both degrees, that students are struggling with, when they take the courses in the same semester. Finally, the article summarizes the lessons learned, which invite academic administrators to reflect on the weaknesses and strengths of each degree analyzed. Specifically, the analysis suggests that the curricula should be reorganized to avoid that students take certain courses together, not because of conceptual reasons, but because students frequently fail if they do so. Some of these patterns are common to both degrees.

Keywords: Frequent itemset mining, Statistics, Data visualization, Educational Data Mining, Computer science degree

Introduction

Motivated by the pursuit of excellence, higher education institutions are using their students' data to achieve a competitive advantage. The excellence can be translated, for instance, in the international university rankings, which serve as a showcase to project

the reputation of the institution, consequently helping attract and retain good students, raise research funds, improve internal processes, and contribute to the society with professionals with a solid education. All these issues converge to one of the major challenges that higher education institutions must face: continuously improve the degrees offered to ensure that a high percentage of the students indeed graduate.

Two main research areas deal with educational data, namely, Educational Data Mining and Learning Analytics, each one having different origins according to their research communities. Both areas are recent and share the goals of improving and supporting the education at large, as well as research and practice in education [1, 2].

This article adopts visualization and data mining techniques to present an in-depth analysis and comparison of two computer science degree offerings, viz., the Bologna BSc in Information Systems and Computer Engineering (LEIC-A), offered at the Alameda campus of the *Instituto Superior Técnico* (IST) of the *University of Lisbon*, Portugal, and the BSc in Computer Science (BCC), offered by the *Pontifical Catholic University of Rio de Janeiro*, Brazil. The analysis is based on student transcripts collected from the academic systems of both institutions over circa 10 years.

The article starts with a description of the degrees and global statistics of the student population under consideration. Then, it presents a comparative analysis of the curricula, using data visualization techniques and academic performance indexes, which focus on how close students follow the recommended curricula. The academic performance indexes indicated a mismatch between the semesters that the curricula recommend for the courses and the semesters that students enroll in those courses. Furthermore, a visualization of course advances and delays indicated that a significant fraction of the students are not being approved in the semester that the curricula recommend for the courses.

The article moves on to present a comparative analysis of student performance in individual courses, and then applies a technique borrowed from Market Basket Analysis to investigate student performance in multiple courses, taken in the same semester. The analysis pointed out sets of courses, at both degrees, that students are struggling with, when they take the courses in the same semester.

Finally, the article summarizes the lessons learned and invites academic administrators to reflect on the weaknesses and strengths of each degree analyzed. Specifically, the analysis suggests that the curricula should be reorganized to avoid that students take certain courses together, not because of conceptual reasons, but because students frequently fail if they do so. Some of these patterns are common to both degrees.

A comparative analysis of student performance from two different institutions, located in distinct countries, at the granularity reported in this article, is not a simple task. It is challenging both to overcome the problem of obtaining the necessary data, as well as the problem of acquiring the background knowledge required to understand the data. However, the effort is well-justified since the results reported in this article indicated common problems that the students of both degrees struggle with, which are independent of the cultural and organizational differences between the academic institutions, degrees, and students' backgrounds. The findings suggest that the problems are intrinsic to the computer science curricula, as exemplified by the two degrees selected for analysis.

The main contributions of the article can be summarized as follows. It propose principles to visualize educational data, academic performance indexes that simplify the educational data analysis and comparison, and adequate mappings for effective application of Market Basket Analysis methods (pattern mining) on curriculum data. The findings suggest possible reorganizations of the curricula and, again, aim at uncovering patterns that are common to both degrees analyzed. Additionally, better personalized planning can be offered to the students before their enrollment in the next semester.

The remainder of this article is organized as follows. The "Related work" section summarizes related work. The "A first comparison of the degrees" section introduces the case study and presents a comparative analysis of the student population. The "A comparative analysis of student adherence to the curricula" section contains a comparative analysis of the curricula. The "A comparative analysis of student performance" section describes a comparative analysis of student performance. The "Conclusions and future work" section presents conclusions and future work.

Related work

Educational Data Mining (EDM) [3] is an interdisciplinary area that applies data mining techniques to educational data to address important educational questions [1, 4–6]. EDM is a recent area—with the first annual international conference held in 2008, followed by the Journal of Educational Data Mining, and by the first Handbook of Educational Data Mining, both in 2009—but the interest in this field is not recent [7–12]. The interest began in traditional education, and then the studies were intensified with the advent of distance education systems. In the early days, educational content was presented as static Web pages, and only statistics about the students' clickstreams and the Web site efficiency were investigated. Today, the statistics are fine-grained, carrying information about session duration, read material, completed quizzes, student achievements, etc. All of this information provides a mapping of the whole process of teaching and learning at different levels, according to the stakeholders' interest (students, teachers, degree coordinator, academic coordinator, etc.), leading the field to a higher level of freedom to investigate several areas of knowledge.

Pechenizkiy et al. [13] developed a curriculum mining software—based on process mining [14], data mining, and visualization techniques—to identify the recommended curriculum, the typical students' behaviors, the constraints, and the dropout patterns. Wang and Zaiane [15] also used process mining to analyze curriculum data, aiming at discovering sequences of courses taken by students. They found that, by analyzing different students' cohorts, one can uncover different needs and subsequently act on them, recommending specific course sequences to each student and giving new insights to administrators.

Campagni et al. [16] presented a data mining methodology to analyze the students' careers, using clustering and sequential patterns techniques. They introduced the concept of ideal career (without delay) to compare the students' behavior with the ideal career, confirming that good performance (graduation time and final grades) is attained whenever students follow the order of the ideal career. They also found frequent sequential patterns to classify students (good/not so good) according to the final grade and the length of studies, concluding that good students take most exams according to the curriculum recommended order. Asif et al. [17] followed a similar approach by

analyzing the students' progression performance during the degree, using a tuple to compare performances with respect to their first year and measure if the student's results increase, decrease, or stay the same.

Ochoa [5] proposed a list of metrics to be applied to academic data to measure the students' interactions with the recommended curriculum. Kumar and Chandra [18] applied association rules to graduation and post-graduation students' marks to check computer science students' performance in both degrees. Barbosa et al. [17] analyzed a curriculum structure of the computer science undergraduate students from 2005 to 2016 through a data mining technique, based on the synthetic control method. They compared the results with a linear regression model and proposed a visualization tool that depicts the comparison between the recommended curriculum and the structure found in data.

Buldu and Üçgün [19] applied the Apriori algorithm to the students of a Vocational Commerce High School, finding rules associated with the students' failed courses to apply strategies to overcome this situation. Chandra and Nandhini [20] applied association rules to the computer science undergraduate students of Nigeria to uncover hidden patterns in students' failed courses, which can be used to improve the recommended curriculum and the students' performance. Olaniyi et al. [21] analyzed the student failure pattern by applying the Apriori algorithm to North Central Nigeria, aiming at providing recommendations about a curriculum redesign.

Similarly to the approach taken in this paper, the studies in [19–21] applied association rules to the students' failed courses, in order to extract patterns that can be used as recommendation to students and to the department coordinators to avoid taking some courses together or the other way around to encourage some other courses to be taken together. This article analyzes and compares two degrees from different universities in different countries, namely, the IST/ LEIC-A and the PUC-Rio/BCC degrees, chosen as a case study. The goal is to uncover courses and course combinations that are problematic in both degrees and to analyze the suitability of the recommended curricula, independently of the cultural differences. The article advances our previous investigation on a single degree analysis [22].

As mentioned in the introduction, a comparison of student performance from two different institutions, at the granularity reported in this article, is not common in the related work reported in this section mostly due to the difficulty of obtaining the required data and the knowledge necessary to understand the data. The techniques adopted in the "A comparative analysis of student adherence to the curricula" section permit identifying courses that students experience difficulties and check if these experienced difficulties are common to both degrees. This last point differentiates this article from related work—that addresses student performance—since it depends on a detailed analysis of the course syllabus from both degrees to create a mapping between the curricula. The "A comparative analysis of student performance" section applies a technique borrowed from Market Basket Analysis to investigate student performance in multiple courses, taken in the same semester. The findings suggest possible reorganizations of the curriculum and, again, aim at uncovering patterns that are common to both degrees. The patterns should have an academic explanation and should not depend on the cultural differences between the students' backgrounds. Again, this analysis depends on a thorough understanding of the academic institutions, degrees, and

students' background being compared. For these reasons, this type of analysis is not commonly reported for degrees offered in different countries.

A first comparison of the degrees

This section first summarizes the characterization of the degrees undergoing analysis, which we recall are the bachelor degree in Information Systems and Computer Engineering (LEIC-A), offered at the Alameda campus of the Instituto Superior Técnico (IST), University of Lisbon, Portugal, and the bachelor degree in Computer Science (BCC), offered by the Pontifical Catholic University of Rio de Janeiro (PUC-Rio), Brazil. For simplicity, we refer to these degrees as IST/LEIC-A and PUC-Rio/BCC. Then, the section presents global statistics for both degrees.

Founded in 1911, the Instituto Superior Técnico (IST) is a public school of engineering, located in three campi, Alameda, TagusPark, and Loures. In 2014, IST had approximately 11,500 students, distributed among 19 undergraduate degrees, 31 master programs, and 33 Ph.D. programs. PUC-Rio is a private, non-profit university, founded in 1941, with a single campus. In the second semester of 2017, PUC-Rio had approximately 11,500 undergraduate students, distributed among 48 undergraduate degrees, and 2500 graduate students, distributed among 31 master programs and 25 Ph.D. programs. In general, at PUC-Rio, the recommended curricula are defined as a guide to the students, in the sense that students are free to choose the courses they want to take each semester, having only to respect the prerequisites; by contrast, at IST, each degree follows a strict sequence of courses.

Created over 25 years ago and restructured to meet the Bologna Process in 2006, IST/LEIC-A is designed for 3 years and is simultaneously offered at the Alameda and TagusPark campi. PUC-Rio/BCC, which was created in 2009, is designed for 4 years. Although IST does not impose a time limit to the studies duration, PUC-Rio has decreed a maximum duration of 8 years of studies. On average, during the period considered (see Table 1), IST/LEIC-A admitted 215 students per year, while this number was 25 for PUC-Rio/BCC.

The annual fee at IST is about 1100 € (in 2017); the admission process is based on the (Portuguese) National Exam ranking, and the student socio-economic profile is very heterogeneous. By contrast, the annual fee at PUC-Rio is approximately 13,000 € (in 2017). However, nearly 30% of the students of PUC-Rio/BCC have a full scholarship, that is, they do not pay tuition. The socio-economic profile of the student body is relatively heterogeneous. The admission process for PUC-Rio/BCC is quite similar to the

Table 1 Summary, by student status, of IST LEIC-A (3 years) from 2006 until 2016 and PUC-Rio/BCC (4 years) from 2009 until 2017

Status	Number of students	
	IST/LEIC-A	PUC-Rio/BCC
Graduated	908	15
Non-graduated	583	160
Enrolled	876	129
Total	2367	304

Engineering degrees, which means that students must have a reasonable proficiency in mathematics and the exact sciences.

The analysis in this and the next sections was based on student transcripts collected from the academic systems of both institutions over circa one decade. For IST/LEIC-A, the data collected cover from 2006 until 2016 and encompass 2367 students, from the Alameda campus. After the cleaning and transformation processes, the final dataset had 65,048 rows, which translate student-semester course information. For PUC-Rio/BCC, the data collected cover from 2009 until 2017 (inclusive) and encompass 304 students; the final dataset had 5150 rows. We call these sets of students the *student populations* and the data collected, the *student datasets*.

Each student in the population considered may have one of the following degree statuses:

- *enrolled*, when the student is still enrolled for the degree
- *graduated*, when the student successfully finished the degree
- *non-graduated*, when the student is neither enrolled nor has graduated, in which case, the status of the student may be as follows:
  - *canceled*, when the student formally canceled his enrollment for the degree
  - *expelled*, when the student had his enrollment for the degree canceled because he exceeded the maximum duration allowed for the degree, or for some other reason
  - *dropout*, when the student quitted pursuing the degree and neither formally canceled his enrollment nor was expelled

Table 1 shows the student population by status. Note that 38% of students of IST/LEIC-A graduated, while being only 5% for PUC-Rio/BCC. The low percentage for PUC-Rio/BCC is misleading, though, since very few students were admitted when the degree began to be offered, but this number increased significantly over the recent years. This means that Table 1 is comparing a small number of students that were admitted several years ago (and are now graduating) with a total population that increased significantly in recent years.

Figure 1 shows the percentage of graduations by semesters (not years), i.e., the duration in semesters that a student spends to obtain his/her graduation. At IST, students more frequently graduate in 6 (32%), 8 (17%), or 10 (12%) semesters, respectively. Graduation in one semester, approximately 15%, is due to students who were transferred from other institutions or other IST degrees or even due to students returning to IST from a previous computer science curriculum, which need only one semester to finish the degree. The comparison with PUC-Rio is poor since there are very few students who graduated in the period considered, as already explained; indeed, the (few) students that graduated spent between 9 and 13 semesters to conclude the degree.

Figure 2 shows the percentage of non-graduated students in semesters, i.e., the semester that the student was enrolled in when he/she changed the status to non-graduation (recall that the maximum time for PUC-Rio/BCC is 8 years). Observe that students frequently quit IST/LEIC-A at the 2nd, 4th, and 6th semesters, and not in the first semester of each academic year—this is probably related to the fee, which is an annual fee. Students frequently quit PUC-Rio/BCC in the first three semesters of the

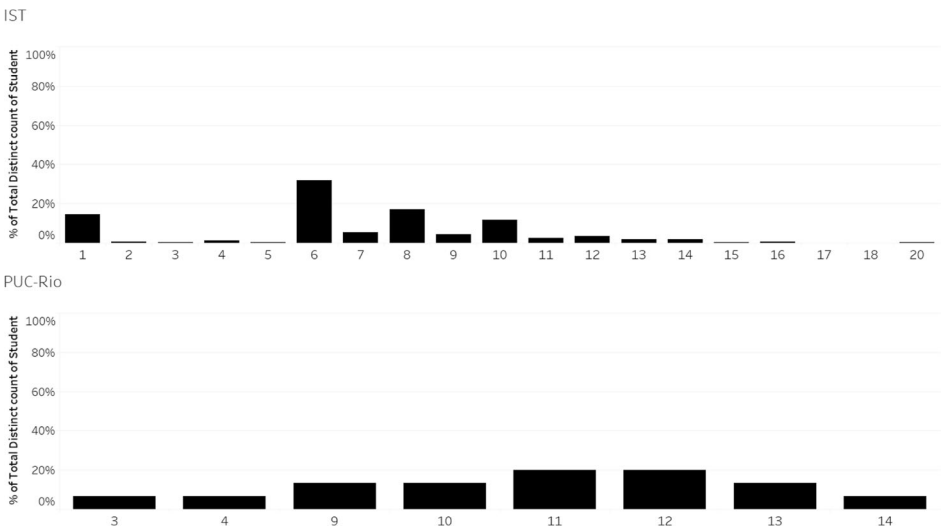

Fig. 1 The percentage of graduations by semesters, i.e., the duration in semesters that a student spends to obtain his/her graduation, for IST/LEIC-A (3 years) from 2006 until 2016 and PUC-Rio/BCC (4 years) from 2009 until 2017

degree; an interview with the degree coordinator revealed that students frequently quit PUC-Rio/BCC because they had a different perspective of the computer science degree—they often believe that computer science involves no mathematics. In such cases, the student ought to be redirected to the PUC-Rio Industrial Design degree, for example, which has an emphasis on Digital Media (and no mathematics). Although one may suspect that this is a phenomenon common to most computer science degrees, to the best of our knowledge, there is no comprehensive survey to support this statement.

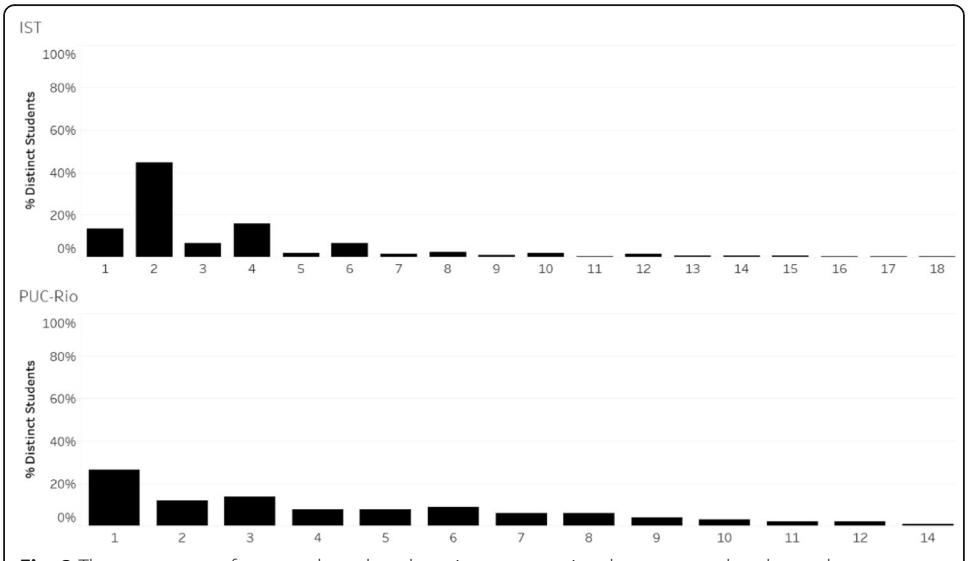

Fig. 2 The percentage of non-graduated students in semesters, i.e., the semester that the student was enrolled in when he/she changed the status to non-graduation, for IST/LEIC-A (3 years) from 2006 until 2016 and PUC-Rio/BCC (4 years) from 2009 until 2017

A comparative analysis of student adherence to the curricula

In this section, we investigate how close students follow the recommended curriculum, that is, if they take the courses in the recommended semester. The analysis is quantitative, based only on the student transcripts, as summarized in the "A first comparison of the degrees" section, and introduces indicators that the degree coordinator can use to assess student progress, much beyond computing the mere average grades in each course. Aspects related to the adequacy of the course syllabus vis-à-vis the degree objectives or the performance of the professors influence the results of the analysis but are not captured by transcript data. Course surveys, for example, would evaluate such aspects and would, therefore, complement the analysis of this section.

To compare the degrees, we restricted the analysis to those courses offered by IST/LEIC-A that have an equivalent at PUC-Rio/BCC—the equivalence was defined by teachers from both institutions. Table 2 lists the IST/LEIC-A courses, the equivalent courses at PUC-Rio/BCC, and an English translation of their names. In fact, about 76% of the IST/LEIC-A courses had an equivalent course in PUC-Rio/BCC, where this percentage is defined as follows:

$$CO = \frac{\text{number of courses of IST/LEIC–A equivalent to a course of PUC–Rio/BCC}}{\text{number of courses of IST/LEIC–A}}$$

Therefore, the degrees selected for analysis are similar with respect to their syllabi. The differences lie in their duration, enrollment policy (credit versus sequential), size of the student body, and maturity of the degrees, as explained in the "A first comparison of the degrees" section.

We also restricted the population to those students who took such courses. Furthermore, in the case of PUC-Rio/BCC, we selected students that followed one of the four different curricula available for the period considered (2009–2017), chosen as that with the largest number of students. For this reason, the total number of distinct students in each semester is lower than that considered in the "A first comparison of the degrees" section. In the case of this restricted student population, Table 3 shows the number of students by the total time they were enrolled in the degree, in semesters. Observe that the total number of students decreases with the number of semesters since students graduate or quit as they progress in the degree.

To answer the question about how close students follow the recommended curriculum, we first introduce a global degree index. Let S be a given set of students enrolled in a degree D over a period of time T measured in semesters. The *degree-semester adherence index*, denoted by $A_{D,t}$, measures how close students in S followed the recommended set of courses C_t for degree D at a given semester t in T, and it is defined as follows:

$$A_{D,t} = \frac{1}{n}\sum_{i=i}^{n}\frac{|E_{i,t} \cap C_t|}{|E_{i,t} \cup C_t|}$$

where n is the total number of students in S enrolled in D in semester t; $E_{i,t}$ is the set of courses student i in S enrolled in semester t; C_t is the set of courses recommended for semester t of D.

Table 2 A mapping between the courses offered by IST/LEIC-A and PUC-Rio/BCC

Courses—IST/LEIC-A (in Portuguese)	Equivalent courses—PUC-Rio/BCC (in Portuguese)	(English translation)
Álgebra Linear	Álgebra Linear I	Linear Algebra I
Cálculo Diferencial e Integral I	Cálculo de Uma Váriável	Differential and Integral Calculus I
Fundamentos de Programação	Programação para Informática	Foundations of Programming
Introdução à Arquitetura de Computadores	Introdução à Arquitetura de Computadores	Introduction to Computer Architecture
Cálculo Diferencial e Integral II	Cálculo a Várias Variáveis	Differential and Integral Calculus II
Lógica para Programação	Lógica para Computação	Logic for Programming
Matemática Discreta	Estruturas Discreta	Discrete Mathematics
Intr. Algoritmos e Estruturas de Dados	Estruturas de Dados Avançadas	Intr. to Algorithms and Data Structures
Análise Complexa e Equações Diferenciais	Equações Diferenciais e de Diferenças Finitas	Complex Analysis and Differential Equations
Sistemas Operativos	Software Básico	Operating Systems
Programação com Objetos	Programação Orientada a Objetos	Object-Oriented Programming
Mecânica e Ondas	Mecânica Newtoniana	Mechanics and Waves
Análise e Síntese de Algoritmos	Análise de Algoritmos	Analysis and Synthesis of Algorithms
Probabilidade e Estatística	Probabilidade e Estatística	Probability and Statistics
Interface Pessoa Máquina	Intr. à Interação Humano-Computador	Human-Computer Interaction
Teoria da Computação	Computabilidade	Theory of Computation
Redes de Computadores	Redes de Comunicação de Dados	Computer Networks
Organização de Computadores	Sistemas de Computação	Computer Organization
Bases de Dados	Bancos de Dados	Databases
Inteligência Artificial	Inteligência Artificial	Artificial Intelligence
Engenharia de Software	Modelagem de Software	Software Engineering
Compiladores	Compiladores	Compilers
Aspectos Prof. e Sociais da Engenharia	Ética Profissional	Computing and Society

Table 3 Number of students by the total time they were enrolled in the degree, in semesters, for IST/LEIC-A (3 years) from 2006 until 2016 and PUC-Rio/BCC (4 years) from 2009 until 2017

Total time in sem.	Number of students		Total time in sem.	Number of students	
	IST/LEIC-A	PUC-Rio/BCC		IST/LEIC-A	PUC-Rio/BCC
1	2362	155	12	194	14
2	1943	133	13	156	7
3	1688	110	14	99	2
4	1476	89	15	78	1
5	1377	78	16	49	1
6	1185	67	17	44	-
7	792	49	18	26	-
8	642	44	19	19	-
9	456	33	20	11	-
10	381	33	21	6	-
11	251	20	22	-	-

Note that the fraction in the summation is the Jaccard similarity index between $E_{i,t}$ and C_t [23], a popular similarity measure between two entities, defined as the cardinality of the intersection of their sets of characteristics divided by the cardinality of their union. Also, note that $A_{D,t} \in [0, 1]$, where $A_{D,t} = 0$ iff there are no students enrolled in any of the recommended courses for semester t of D, and $A_{D,t} = 1$ iff all students enrolled in exactly the recommended courses.

The *overall degree-semester adherence index* of degree D over the period of time T is then defined as the average of the degree-semester adherence indexes for the semesters of D over the period of time T for the given set of students S.

Figure 3 shows the overall degree-semester adherence index for IST/LEIC-A and PUC-Rio/BCC. This figure indicates that students are, in general, not following the recommended course order indicated by the curriculum, since this index is low already in the first semester. In the case of IST/LEIC-A, for instance, the curriculum adherence index of 0.59 for the first semester happens due to a curriculum revision in the academic year of 2014/2015, which changed two courses. Otherwise, if we separately consider the old and new versions of the curricula of IST/LEIC-A, the resulting curriculum adherence index would be close to one as a result of a strict enrollment policy. In the case of PUC-Rio/BCC, the main reason for the curriculum adherence index of 0.67 for the first semester is due to a more flexible choice of courses, since the curriculum is just a recommendation for the students. In later semesters, one possible reason for a low adherence index is a high failure rate (failed or non-evaluated students) in some earlier semester courses, which impairs enrollment in courses at later semesters, that is, failure (to pass) courses is a cumulative phenomenon with respect to this index.

We stress that the degree-semester adherence index is indeed applicable for IST/LEIC-A, albeit this degree follows a strict sequence of courses. Otherwise, the index would be uniformly 1, which is not the case. Indeed, if a student s fails in a course c that the IST/LEIC-A curriculum defines for a semester t, student s must re-enroll in c in semester $t + 1$, and so on, until s/he passes c. Hence, the more students fail to pass

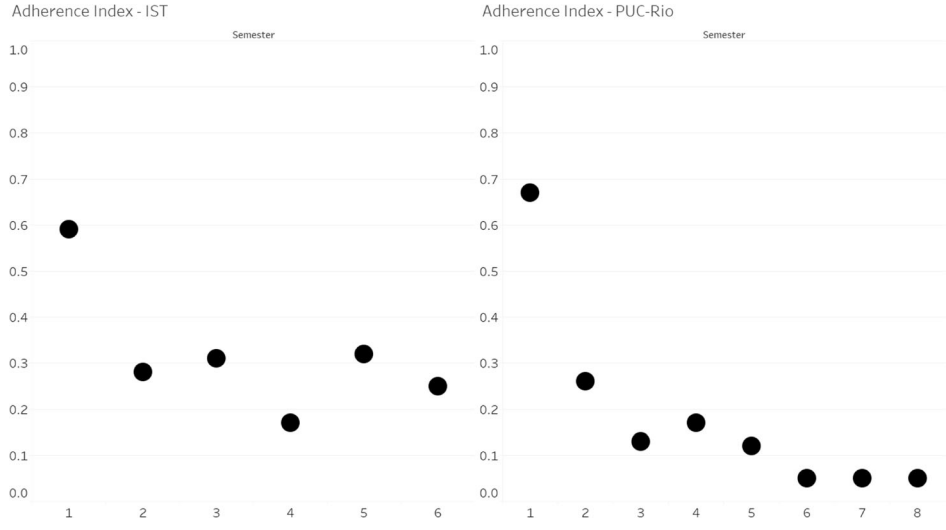

Fig. 3 Overall degree adherence index of IST/LEIC-A (3 years) from 2006 until 2016 and PUC-Rio/BCC (4 years) from 2009 until 2017

the courses defined for a given semester, the lower the index for IST/LEIC-A at that semester will be. By contrast, if s were a student of PUC-Rio/BCC, s/he might take c at a later semester, and not necessarily at $t + 1$, which forces s to enroll in courses that depend on c at even later semesters. Figure 3 reflects to some extent the effect of the enrollment policy followed by IST/LEIC-A, in so far as the degree adherence index of IST/LEIC-A (left-hand side) is greater than or equal to that of PUC-Rio/BCC (right-hand side) for all semesters but the first.

To be more specific about how close students follow the recommended curriculum, we resort to a visualization strategy that indicates how much students delay or advance courses, that is, in which semester they are successfully approved in a course, as compared with the semester the curriculum recommends for that course.

Figure 4 applies this strategy to IST/LEIC-A and PUC-Rio/BCC, with the courses ordered by the recommended semester in the IST/LEIC-A curriculum. The second column of the PUC-Rio/BCC part of the figure indicates the semester the PUC-Rio/BCC curriculum recommends for the course. The size of a box in each cell represents the proportion of the students approved in a given course at a given semester. The central column, labeled 0, corresponds to students approved in the semester recommended for the courses; columns labeled with a negative number, to the left, correspond to students approved in an earlier semester (– 1 means one semester earlier, etc.), and those labeled with a positive number, to the right, correspond to students approved in a later semester (+ 1 means one semester later, etc.). Observe that a common characteristic of both degrees is that students are usually approved in mathematics courses, such as "Cálculo Diferencial e Integral II," "Análise Complexa e Equações Diferenciais," and "Probabilidade e Estatística," in a semester which is later than the recommended semester for those courses. A possible reason could be that students are overloaded with CS course projects during the current semester, putting math courses apart and frequently failing in the final examinations of those courses. This was noticed at a given point by the degree coordinator, who now strictly overview and discuss with CS teachers the workload of the projects in advance. Another point to observe is that some students are transferred from other degrees. For instance, they started an Electrical Engineering degree and then applied for the CS degree, receiving equivalences in several courses

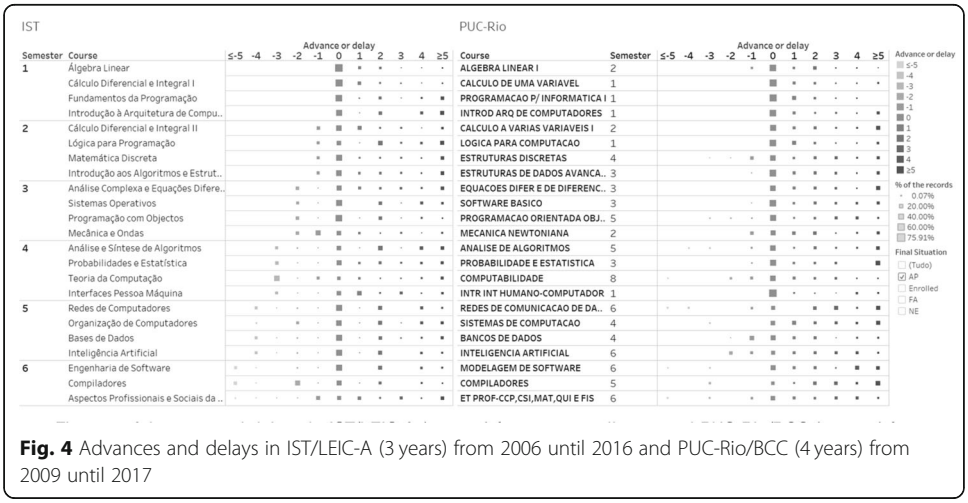

Fig. 4 Advances and delays in IST/LEIC-A (3 years) from 2006 until 2016 and PUC-Rio/BCC (4 years) from 2009 until 2017

but necessarily enrolling in others. An example is the "Compiladores" course, which is recommended in the 6th semester but which transferred students enroll in the first semester of their new degree.

In this analysis, we can identify the semesters in which students are not following the recommended curriculum and also the possible reasons for that, i.e., advances or delays in courses. However, it is not possible to reach any conclusion about the number of attempts a student makes to be approved.

A comparative analysis of student performance

This section first presents a comparative analysis of student performance in individual courses. Then, it applies a technique borrowed from Market Basket Analysis to investigate student performance in multiple courses taken in the same semester. The first part analyses courses independently from each other, whereas the second part considers possible course associations. The comparative analysis uses the same mapping between the courses and the same student population, as in the "A comparative analysis of student adherence to the curricula" section.

Let D be a degree, C be the set of courses of D, T be a period of time, understood here as a set of semesters, and S be a non-empty set of students taking degree D. We assume that T is equipped with a total order. With respect to a course c and a semester t, a student s may have one of the following final course statuses f:

- *approved* (AP), when student s successfully concluded course c in semester t
- *failed* (FA), when student s unsuccessfully concluded course c in semester t
- *non-evaluated* (NE), when student s took course c in semester t, without being formally evaluated

We use F to denote the set of all final course statuses.

A student record for D is simply a quadruple $(s, c, t, f) \in S \times C \times T \times F$ indicating that student s has status f for course c in semester t. A set $R \subseteq S \times C \times T \times F$ of student records is *consistent* iff:

- for any pair of records (s, c, t, f) and (s', c', t', f') in R, if $s = s'$, $c=c'$, and $t = t'$ then $f = f'$; intuitively, a student has a single status for a course in a given semester
- for any pair of records (s, c, t, f) and (s', c', t', f') in R, if $s = s'$, $c = c'$, and $f =$ *approval* then $t > t'$; intuitively, once approved, a student cannot be involved in the course (and hence cannot be approved twice in the same course, for example)

Figure 5 shows the status of the restricted student population for the set of courses considered in this section, where the dark gray section of a bar indicates the number of failed students, mid gray, non-evaluated, and light gray, approved. Note, for example, that "Cálculo Diferencial e Integral I" is a problematic course for both degrees, since it has a high failure rate.

"Human-Computer Interaction" ("Introdução à Interação Humano-Computador"—IHC) at PUC-Rio/BCC also calls attention since this course has a high failure rate, and yet it should be attractive to computer science students. An interview with the

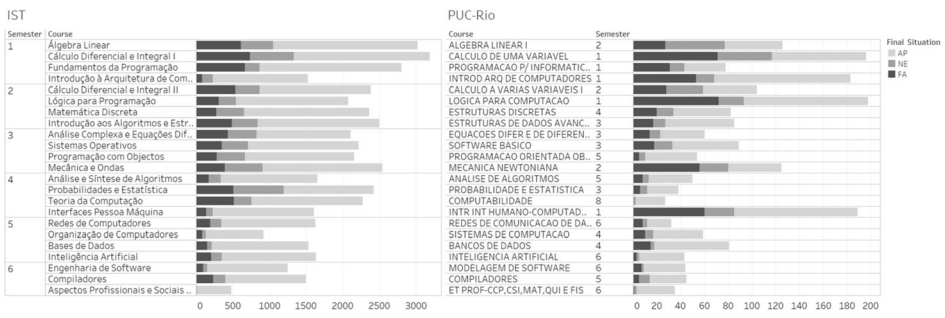

Fig. 5 Course statuses for IST/LEIC-A (3 years) from 2006 until 2016 and PUC-Rio/BCC (4 years) from 2009 until 2017

professor responsible for the course brought several facts that could explain the high failure rate: (1) IHC is a first semester discipline and is not a pre-requisite of any other course; (2) students often abandon the course and focus on "Differential and Integral Calculus I," which is a pre-requisite for other courses; and (3) students are freshman that often do not pay sufficient attention to cancel the course if they get a poor grade in the first test or fail to hand-in the often laborious assignments. Note that the first and, in part, the second points could be detected from the transcripts and the curriculum, but not the third point. Hence, the problem of IHC is an example of the limitations of our transcript- and curriculum-based analysis.

To further analyze student performance, we define the *difficulty index* as follows. Let $R \subseteq S \times C \times T \times F$ be a consistent set of student records, $SC \subseteq C$ be a set of courses, and $c \in C$ be a course. Define the sets as follows:

$appr[c] = \{s \in S / (\exists t \in T)((s, c, t, approved) \in R)\}$, the set of students that were approved in course c

$took[SC] = \{(s, c, t, f) \in R / c \in SC \wedge s \in appr[c]\}$, the set of records that refer to students approved in a course c in SC, whose cardinality is the number of times students took the courses until being finally approved

$succ[SC] = \{(s, c, t, f) \in R / c \in SC \wedge f = approved\}$, the set of approved records that refer to a course c in SC

The *difficulty index* for SC with respect to R, denoted by Δ_{SC}, is defined as follows:

$$\Delta_{SC} = \frac{|took[SC]|}{|succ[SC]|}$$

which is the average number of times students took some course in SC until approval. Note that $\Delta_{SC} \geq 1$ with $\Delta_{SC} = 1$ iff all students were approved the first time they enrolled in a course in SC (in the set of student records R); the higher Δ_{SC} is, the more difficult the set of courses SC is for the students. The difficulty index of a course c with respect to R is defined as $\Delta_{\{c\}}$ and is denoted simply as Δ_c. Finally, the difficulty index for a degree D with respect to R, denoted by Δ_D, is the difficulty index for the courses of D w.r.t. to R.

Figure 6 shows the difficulty index for the set of courses suggested for the ith semester, according to the curriculum for each of the degrees analyzed. Recall that IST/LEIC-A degree is planned for 3 years and that PUC-Rio/BCC is designed for 4 years. Figure 6 indicates that, for IST/LEIC-A, the courses recommended for the third

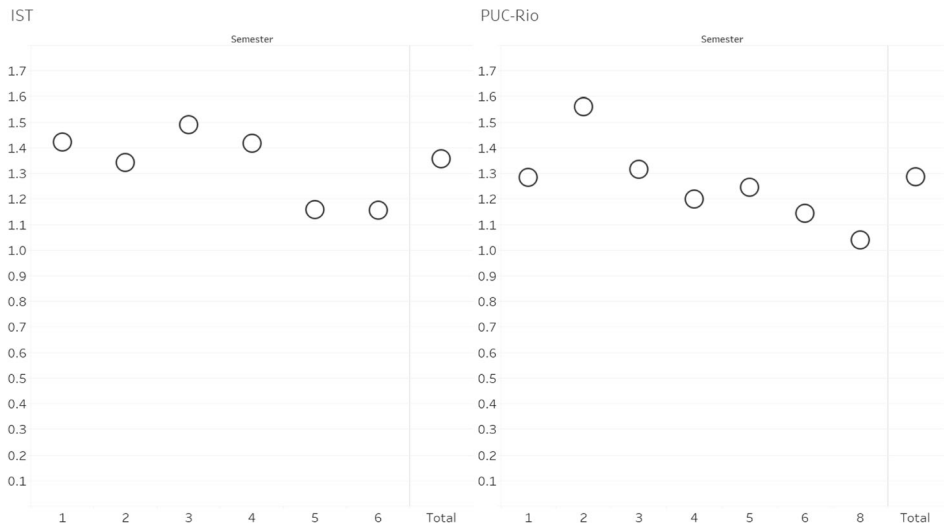

Fig. 6 Difficulty index for the set of courses suggested for the ith semester, for IST/LEIC-A (3 years) from 2006 until 2016 and PUC-Rio/BCC (4 years) from 2009 until 2017

semester need attention, since this set of courses has the highest difficulty index. With respect to PUC-Rio/BCC, this is true for the set of courses recommended for the second semester. Figure 6 also indicates that the difficulty index tends to decrease along the semesters for the set of courses planned for later semesters, in both degrees. There are two possible explanations: the more mature the student is, the better his/her performance; students that perform poorly tend to drop out earlier in the degree. A further analysis of the dropout rate per semester might shed some light on this issue. The average difficulty indexes for each degree are 1.35 for IST/LEIC-A and 1.28 for PUC-Rio/BCC.

Figure 7 shows the difficulty index for the courses considered in this section. It therefore conveys the same information as Fig. 5 but in a more concise way. Observe that, for IST/LEIC-A, the most problematic courses are "Cálculo Diferencial e Integral I," "Cálculo Diferencial e Integral II," "Análise Complexa e Equações Diferenciais,"

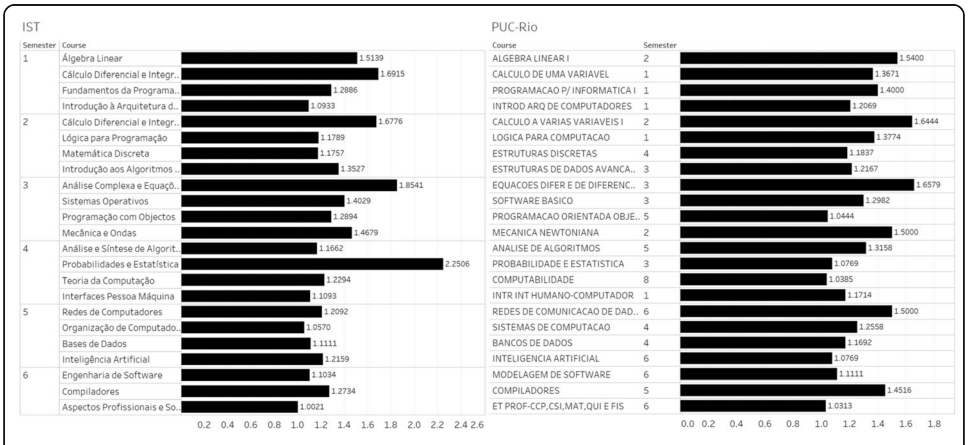

Fig. 7 Difficulty index for the courses of IST/LEIC-A (3 years) from 2006 until 2016 and PUC-Rio/BCC (4 years) from 2009 until 2017

"Probabilidade e Estatística" (with the highest difficulty index), and "Álgebra Linear." Using the title of the courses from IST/LEIC-A for PUC-Rio/BCC, we again have "Cálculo Diferencial e Integral II," "Álgebra Lineal," and "Análise Complexa e Equações Diferenciais" (with the highest difficulty index), but "Mecânica Newtoniana," "Redes de Comunicação de Dados," and "Compiladores" are also problematic. Figure 7 clearly indicates that, at both institutions, degree coordinators ought to investigate how integrated the math courses are with the rest of the curriculum, to mitigate the students' poor performance. "Redes de Comunicação de Dados" also requires a fair amount of math, but not "Compiladores," which deserves special attention at PUC-Rio/BCC.

We now turn to a comparative analysis of student performance for multiple courses, since it can provide more insights about the course distribution of the curriculum along the semesters. We consider a Market Basket Analysis technique, which interprets the degree as the store, the available courses as the available items, the set of courses that the student enrolled in (was approved, failed, or was not evaluated) as the basket, and the semester as the date. The analysis presented in this section focuses on student failure; the process of analyzing other student course statuses would be basically the same. The reader not familiar with Market Basket Analysis is referred to the appendix.

We again consider only those courses that are common to both degrees. We compute the sets of courses that students frequently fail, by semester, using the Apriori algorithm, with a support threshold of 5%.

Figure 8 shows the maximal frequent itemsets for the courses, which students failed, in their first semester, for each degree. Figure 9 depicts the same information but for the second semester. For example, observe from Fig. 8a (for IST/LEIC-A) that the first entry has three courses—{"Álgebra Linear," "Cálculo Diferencial e Integral I," "Fundamentos de Programação"}—which indicates that students frequently fail to pass simultaneously in all three courses, when they enroll in such courses in their first semester. This should be expected since students are probably overloaded with the math courses and also have to struggle with a third course that demands considerable work. By

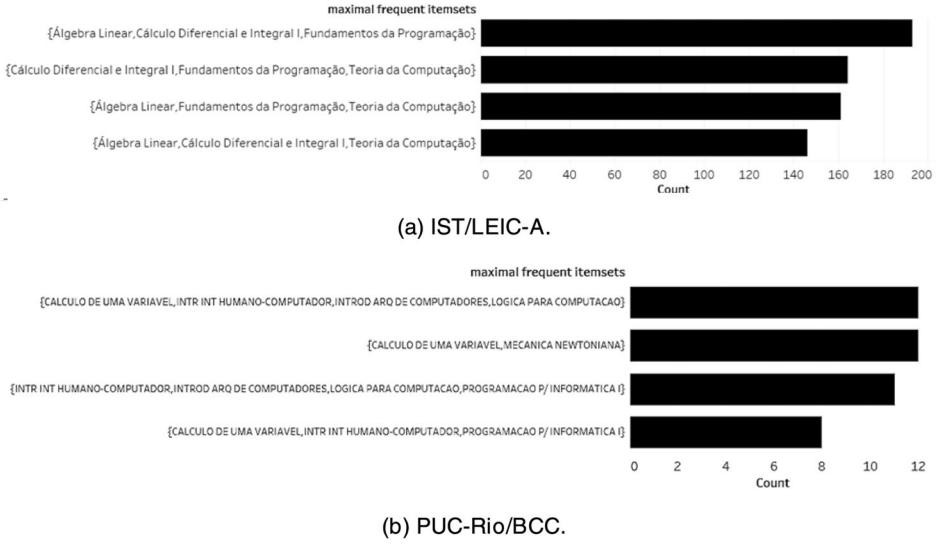

Fig. 8 Maximal frequent itemsets considering the failed attempts of students in the 1st semester with support of 5%. **a** IST/LEIC-A. **b** PUC-Rio/BCC

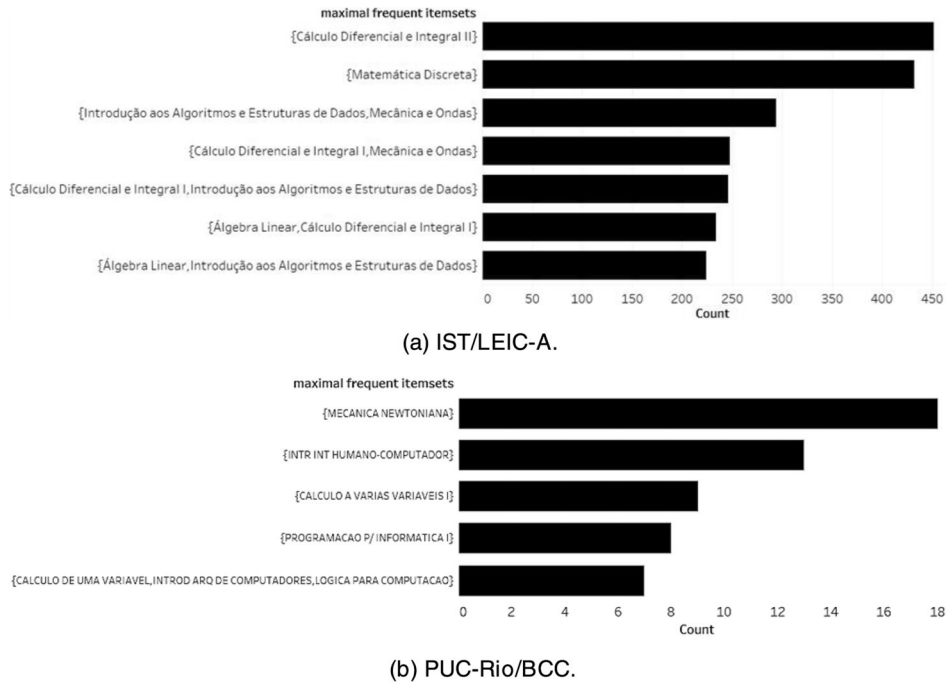

Fig. 9 Maximal frequent itemsets considering the failed attempts of students in the 2nd semester with support of 5%. **a** IST/LEIC-A. **b** PUC-Rio/BCC

definition, once a threshold is defined, if a set S with cardinality n is considered frequent, all subsets of S are also frequent. This means that the sets of cardinality 2 (e.g., {"Álgebra Linear," "Cálculo Diferencial e Integral I"}) and cardinality 1 (e.g., {"Álgebra Linear"}) are also frequent. This can be explained if one recalls that the enrollment process in IST is automatic, i.e., students must enroll again in all courses that they have failed before.

Also, observe from Fig. 8b (for PUC-Rio/BCC) that the first entry has four courses—{"Cálculo de Uma Variável," "Introdução à Interação Humano-Computador," "Introdução à Arquitetura de Computadores," "Lógica para Programação"}—which indicates that students frequently fail to pass in all four courses, when they enroll in such courses in their first semester at PUC-Rio/BCC. A possible explanation is along the lines of that raised earlier for "Human-Computer Interaction" and repeated here for clarity: (1) students often abandon "Human-Computer Interaction" and "Logic for Programming" and focus on "Differential and Integral Calculus I," which is a pre-requisite for other courses; (2) students are freshman that often do not pay sufficient attention to cancel a course, if they get a poor grade in the first test or fail to hand-in assignments.

Likewise, the third entry corresponds to a maximal frequent itemset with four courses also belonging to the PUC-Rio/BCC first recommended semester. This situation deserves special attention since the PUC-Rio/BCC curriculum recommends 5 courses for the first semester. This means a very heavy semester for the students.

In Fig. 9a, note that for the second semester of IST/LEIC-A, the pair {"Álgebra Linear," "Cálculo Diferencial e Integral I"} is a maximal frequent itemset (the sixth entry in the figure). However, this pair is a subset of two maximal frequent

itemsets of the first recommended semester, shown in Fig. 8a. This indicates that students frequently failed in these two courses in the first semester, re-enrolled in them in the second semester, and failed again.

Also note that the first two lines of Fig. 9a show singletons and likewise all lines, but the last, of Fig. 9b. These lines indicate single courses that the students frequently fail, whether or not they are taking other courses in the same semester.

Finally, we focus on maximal frequent 2-itemsets, that is, on pairs of courses which students frequently fail when they take both courses simultaneously in the same semester. This analysis is relevant since the standard curriculum should, based on the present analysis, recommend such courses for different semesters, and future students should avoid taking them in the same semester, if possible.

We computed the pairs of courses that students frequently fail, by semester, using the Apriori algorithm again, with a support threshold of 5%. Figure 10 shows the maximal frequent 2-itemsets for the courses which students failed in their first semester, for each degree. Figure 11 depicts the same information but for the second semester.

From Fig. 10, observe that for the two degrees and the set of common courses, there is a pair of courses (marked with "*") that students frequently fail, when taken in the

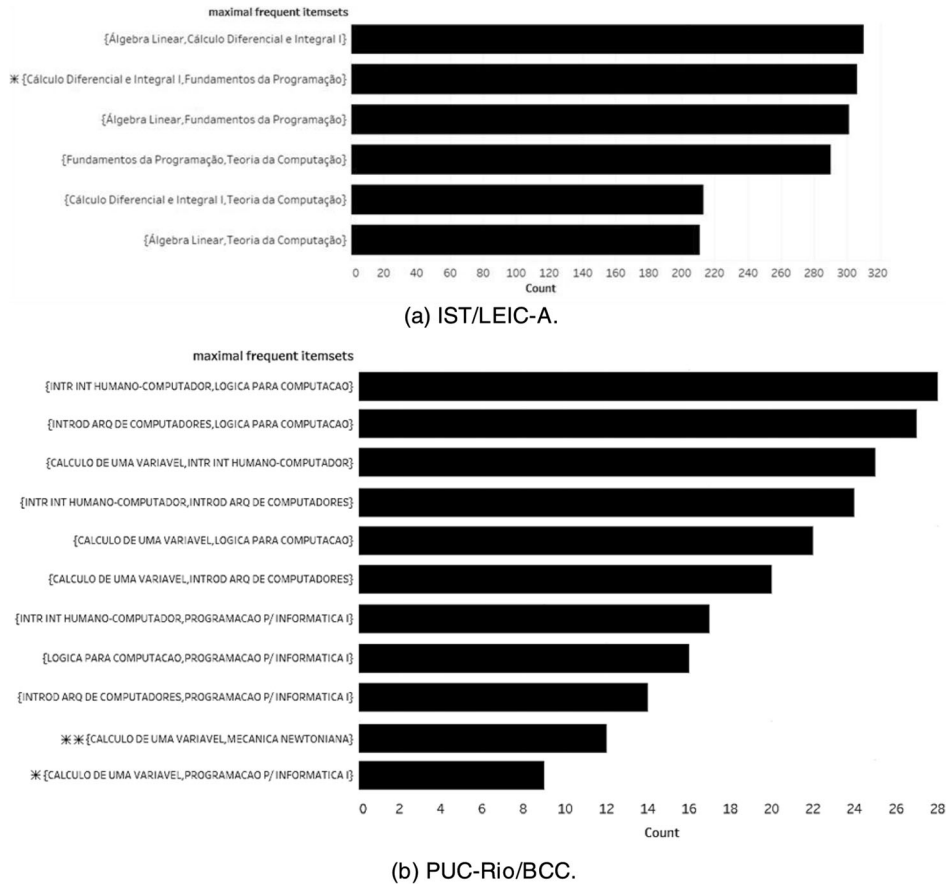

Fig. 10 Maximal frequent itemsets of cardinality 2 considering the failed attempts of students in the 1st semester with support of 5%. **a** IST/LEIC-A. **b** PUC-Rio/BCC

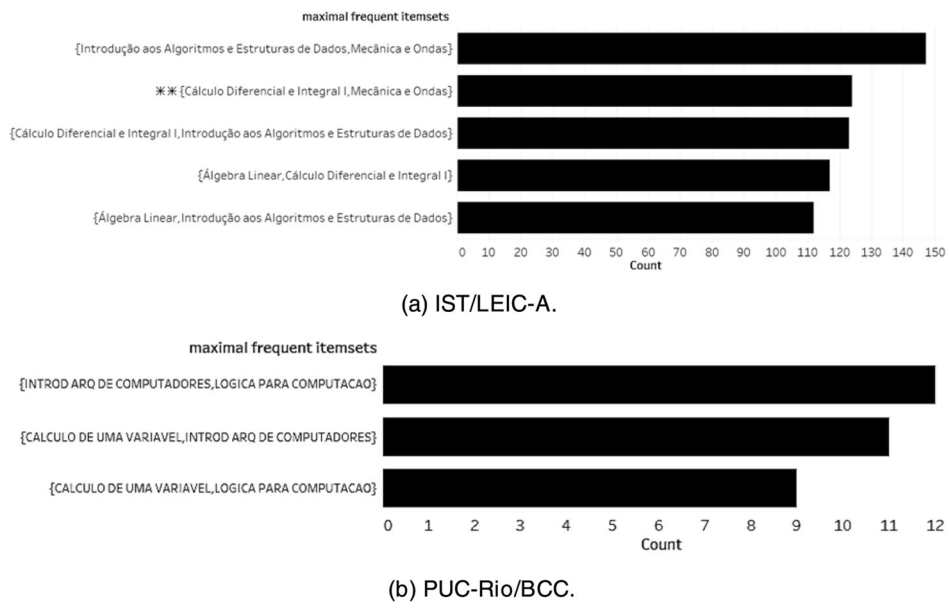

Fig. 11 Maximal frequent itemsets of cardinality 2 considering the failed attempts of students in the 2nd semester with support of 5%. **a** IST/LEIC-A. **b** PUC-Rio/BCC

1st semester, namely, {"Cálculo Diferencial e Integral I," "Fundamentos de Programação"} for IST/LEIC-A, which is equivalent to {"Cálculo de uma Variável," "Programação para Informática I"} for PUC-Rio/BCC.

Furthermore, from Figs. 10 and 11, we can conclude that there are problematic pairs of courses that frequently appear in both semesters, in the first and the second semesters. For IST/LEIC-A, this is the case with {"Álgebra Linear," "Cálculo Diferencial e Integral I"}. For PUC-Rio/BCC, this is far worse since all three 2-itemsets that are frequent in the second semester (Fig. 11b) are also frequent in the first semester (Fig. 10b). Hence, students can be warned not to enroll in these pairs of courses in the second semester, if they have already failed in both courses in the first semester. In the case of IST/LEIC-A, interviewing the degree coordinators, we conclude that after the curriculum revision made in the academic year of 2014/2015, it is possible to expect improvements in several problematic pairs of courses, which were better distributed along the curriculum.

Lastly, note that in Fig. 11a, the pair {"Cálculo Diferencial e Integral I," "Mecânica e Ondas"} (marked with "**") is a maximal frequent itemset in the second semester of IST/LEIC-A. From Fig. 10b, also notice that the equivalent pair {"Cálculo de Uma Variável," "Mecânica Newtoniana"} is a maximal frequent itemset in the first semester of PUC-Rio/BCC. This indicates that students tend to struggle with this pair of courses whenever they are taken together.

Conclusions and future work

In this article, we compared two bachelor CS degrees, the Bologna BSc in Information Systems and Computer Engineering (IST/LEIC-A), Portugal, and the BSc in Computer Science (PUC-Rio/BCC), Brazil, which have similar curricula but differ in other aspects,

among which: PUC-Rio is a private university, while IST is public; IST/LEIC-A is much older than PUC-Rio/BCC; and IST/LEIC-A attracts, per year, over ten times more students than PUC-Rio/BCC.

The analysis was based on student transcripts collected from the academic systems of both institutions over the past years. For IST/LEIC-A, the data cover from 2006 until 2016 and encompass 2367 different students. From this total, 38% graduated, 25% did not graduate, and 37% are still enrolled. As for the non-graduated students, the dropout rate is 80%. For PUC-Rio/BCC, the data cover from 2009 until 2017 and encompass 304 different students, of which almost 5% graduated, 53% did not graduate, and 42% are still enrolled. Among the non-graduated students, the dropout rate is 82%. The semesters in which students frequently drop out are 2nd (45%), 4th (16%), and 1st (13%) for IST/LEIC-A, and 1st (26%), 2nd (12%), and 3rd (14%) for PUC-Rio/BCC. With respect to student retention, these are the problematic target semesters to be monitored. Indeed, regardless of the institution being public or private, high dropout rates have consequences for the students and the institution, since high dropout rates affect educational costs (or the academic fee in a private institution).

The time spent until graduation for IST/LEIC-A is mostly 6 semesters (31%), 8 semesters (17%), and 10 semesters (13%). For PUC-Rio/BCC, students spent between 9 and 13 semesters, but we have to keep in mind the very low rate of graduated students, only 5% overall. The low percentage for PUC-Rio/BCC should not be taken prima facie, as very few students were admitted when the degree started to be offered, but that number has increased significantly in recent years. This aspect distorts the graduation rate.

The adherence indexes indicate a mismatch between the semesters that the curriculum recommends for the courses and the semesters that students actually enroll in those courses. Furthermore, a visualization of course advances and delays indicates that students are not being approved in the semester that the curriculum recommends for the courses. Therefore, the often long discussions to decide the serialization of the curriculum turns out to benefit only the students that rarely fail in a course, which is a small percentage. For the majority of the students that frequently fail in one course or another, the serialization is just an indication that becomes less useful as the student progresses along the degree or starts to fail in one or more courses repeatedly.

We highlight the difficulty or apathy of the students with the math courses. This happens at both degrees, and there are two possible reasons: (1) although students know a priori that math courses are part of the curriculum of the computer science degrees, they usually underestimate the necessary effort to succeed in the math courses; and (2) students give priority to conclude CS-specific courses, which are more attractive to them. Therefore, institutions ought to publicize, with due emphasis, the structure of the curricula before students enroll for a degree. Furthermore, effort should be made to better contextualize math courses to computer science students, which is often overlooked.

We were able to find sets of courses that students are struggling with, when they take the courses in the same semester, at both degrees. In this case, the action of the degree coordinator could be to reconsider the set of courses suggested for each semester of the recommended curriculum. Depending on the number of students, the coordinator may even fine-tune the course sequence to specific students (if few students fail in a

course, the degree staff may offer extra support to those few), or help students identify other courses, or even other degrees, that they may be better prepared to follow. This may involve considerable manual work and may not be implementable, due to restrictions that the academic rules impose, such as those in effect at IST/University of Lisbon.

As future work, we suggest the development of curriculum guidelines with explicit recommendations that courses that demand more effort should not be simultaneously taken—an obvious point that is often overlooked. We are also working on a recommendation system "with memory," i.e., based on the students' number of attempts in a course. Then, if a student is struggling with a single course, the system will suggest concluding this course; however, if s/he is struggling with two (or more) courses, the system will suggest giving preference to the course with the lowest failure rate, for example. Computing off-line the maximal frequent itemsets from the students' transcripts of each degree poses some challenges, but it is worthwhile for the reasons already pointed out. What would be more difficult is to incorporate a recommendation system into a real-time, main-stream enrollment system. A better approach would be to create a "virtual advisor" tool that incorporates such a recommendation system and which students would use to plan his/her courses before the actual enrollment.

Appendix

The Apriori algorithm

An *itemset* is a set of items, and a *transaction* is characterized by an itemset. The *support* of an itemset S is the number (or percentage) of transactions that contain S. The *support threshold* τ indicates the minimum support that must be considered, that is, any itemset whose support is less than τ is discarded. An itemset is *frequent* if its support is above τ. The goal is to find all frequent itemsets M such that no superset of M is also frequent. Such itemsets are called *maximal frequent itemsets*. The definition of τ requires some domain knowledge and considerable experimentation. If τ is set too high, one may end up with very few frequent itemsets. By contrast, if τ is set too low, one may end up with too many frequent itemsets of little significance. The Apriori algorithm [24–26] mines frequent itemsets and explores the fact that, if an itemset I is frequent, then any subsets J of I must also be frequent.

Table 4 Example of a (fictitious) set of students and the courses they failed to pass in a given semester

Student ID	List of courses
1	{"Álgebra Linear," "Cálculo Diferencial e Integral I," "Elementos de Programação," "Matemática Experimental"}
2	{"Álgebra Linear," "Elementos de Programação," "Elementos de Matemática Finita," "Matemática Experimental"}
3	{"Álgebra Linear," "Cálculo Diferencial e Integral I," "Elementos de Programação"}
4	{"Álgebra Linear," "Elementos de Programação," "Matemática Experimental"}
5	{"Cálculo Diferencial e Integral I," "Elementos de Programação," "Elementos de Matemática Finita," "Matemática Experimental"}
...	...
44	{"Cálculo Diferencial e Integral I," "Elementos de Programação," "Elementos de Matemática Finita"}

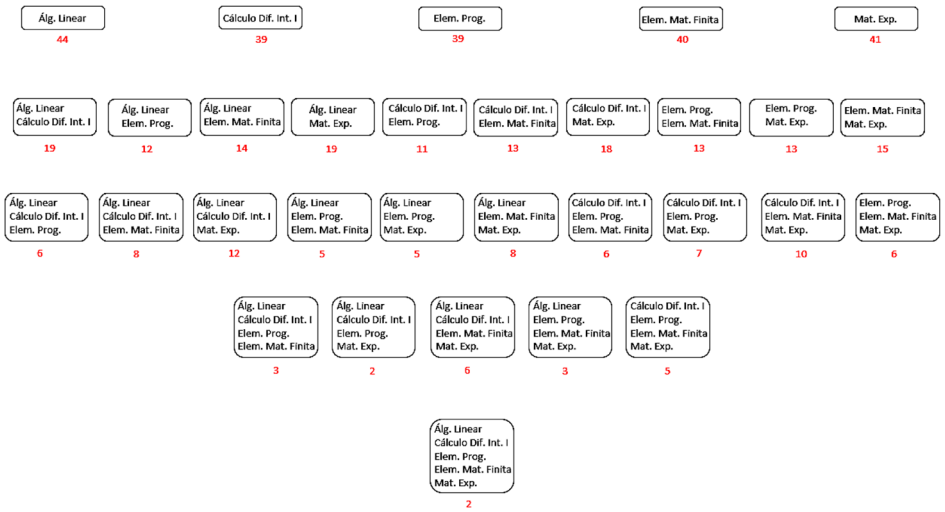

Fig. 12 All possible sets of courses that students failed in the semester under consideration, with the number of students that failed in all courses in the set

In our application domain, an itemset is a set of courses. A transaction is the set of courses that a student failed to pass in the semester under consideration. The support of a set of courses C is the number (or percentage) of students that failed to pass all courses in C. For example, consider Table 4. Each line of the table represents the courses that a student failed to pass in the semester under consideration (note that Table 4 is just a partial listing of the 44 transactions). Figure 12 shows all subsets of the set of courses considered (partially listed in Table 4). The integer below each set indicates the number of students that failed to pass all courses in the set.

Consider a threshold of 50%. Since we have 44 transactions, the minimum absolute frequency of an itemset is $m = 0.5 \times 44 = 22$. Figure 13 illustrates the execution of the Apriori algorithm for the transactions in Table 4. The first step counts the absolute frequencies of all 1-itemset and keeps only the 1-itemsets whose support is greater than $m = 22$. The next step constructs all 2-itemsets and counts their frequency, based on

1-itemset candidates	Abs. Frequency
{"Álgebra Linear"}	44
{"Cálculo Diferencial e Integral I"}	39
{"Elementos de Programação"}	39
{"Elementos de Matemática Finita"}	40
{"Matemática Experimental"}	41

Comparing with the support threshold of 5%

Frequent 1-itemset	Abs. Frequency
{"Álgebra Linear"}	44
{"Cálculo Diferencial e Integral I"}	39
{"Elementos de Programação"}	39
{"Elementos de Matemática Finita"}	40
{"Matemática Experimental"}	41

2-itemset candidates	Abs. Frequency
{"Álg. Linear", "Cálculo Dif. Int. I"}	19
{"Álg. Linear", "Elem. Prog."}	12
{"Álg. Linear", "Elem. Mat. Finita"}	14
{"Álg. Linear", "Mat. Exp."}	19
{"Cálculo Dif. Int. I", "Elem. Prog."}	11
{"Cálculo Dif. Int. I", "Elem. Mat. Finita"}	13
{"Cálculo Dif. Int. I", "Mat. Exp."}	18
{"Elem. Prog.", "Elem. Mat. Finita"}	13
{"Elem. Prog.", "Mat. Exp."}	13
{"Elem. Mat. Finita", "Mat. Exp."}	15

Comparing with the support threshold of 5%

Frequent 2-itemset	Abs. Frequency

Fig. 13 Example of executing the Apriori algorithm for the transactions in Table 4

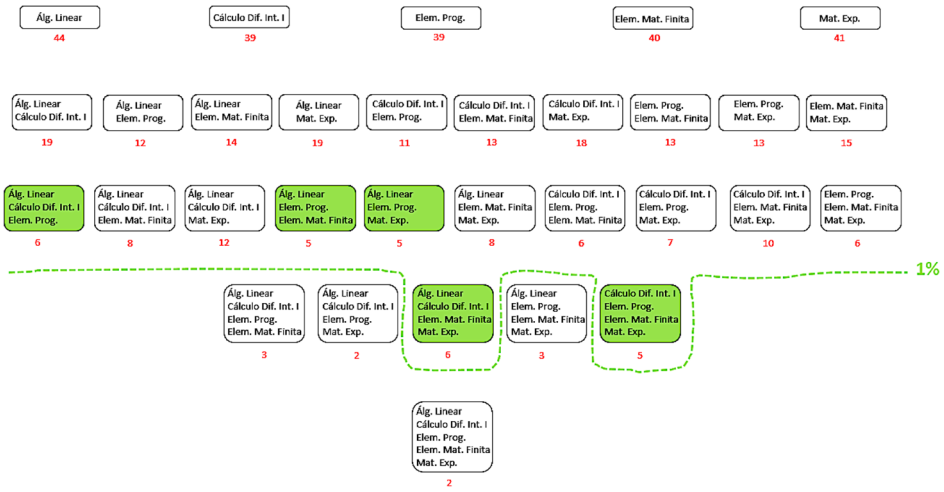

Fig. 14 Application of Market Basket Analysis for a set of students and the courses they failed to pass in a given semester, considering a support threshold of 10%

the frequent 1-itemset. In this example, there is no 2-itemsets whose support is greater than $m = 22$. Thus, the algorithm stops and returns all frequent 1-itemset with support greater than $m = 22$.

Suppose now we choose a support threshold of 10%, that is $m = 0.1 \times 44 = 4.4$. In this case, the Apriori algorithm finds frequent 1-itemset up to frequent 4-itemsets shown above of the dashed line in Fig. 14.

Abbreviations

LEIC-A: Bologna BSc in Information Systems and Computer Engineering; IST: Instituto Superior Técnico; BCC: BSc in Computer Science; PUC-Rio: Pontifical Catholic University of Rio de Janeiro; EDM: Educational Data Mining; AP: Approved; FA: Failed; NE: Non-evaluated

Acknowledgements

This work was partly funded by CNPq under grant 302303/2017-0 and by FAPERJ under grant E-26-202.818/2017.

Authors' contributions

All authors contributed to the writing of this article, read, and approved the final manuscript.

Funding

Not applicable.

Availability of data and materials

An anonymized version of the data can be made available upon request.

Competing interests

The author(s) declare(s) that they have no competing interests.

Author details

[1]CEMAT and INESC-ID–Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal. [2]Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ, Brasil. [3]Departamento de Informática Aplicada, Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, RJ, Brasil. [4]CEMAT–Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal. [5]CEAFEL–Instituto Superior Técnico, Universidade de Lisboa, Lisbon, Portugal.

Received: 25 September 2019 Accepted: 1 May 2020
Published online: 18 May 2020

References

1. Psaromiligkos Y, Orfanidou M, Kytagias C, Zafiri E (2011) Mining log data for the analysis of learners' behaviour in web-based learning management systems. Oper Res 11(2):187–200. https://doi.org/10.1007/s12351-008-0032-4
2. Sanjeev A P, Zytkow J M (1995) Discovering enrollment knowledge in University Databases. In: Proceedings of the 1st International Conference on knowledge discovery and data Mining, pp 246-251.

3. Dutt A, Ismail MA, Herawan T (2017) A systematic review on educational data mining. IEEE Access 5:15991–16005. https://doi.org/10.1109/ACCESS.2017.2654247
4. Nasiri M, Minaei B (2012) Predicting GPA and academic dismissal in LMS using educational data mining: a case mining. In: Proceedings of the 3rd International Conference on e-learning and e-teaching, pp 53-58. doi:https://doi.org/10.1109/ICELET.2012.6333365
5. Ochoa X (2016) Simple metrics for curricular analytics. In: Proceedings of the 1st learning analytics for curriculum and program quality improvement workshop, CEUR Workshop Proceedings, 1590, p. 20-26.
6. Pechenizkiy M, Trcka N, De Bra P, Toledo P (2012). CurriM: curriculum mining. In: Proceedings of the 5th International Conference on educational data mining, pp. 216-217.
7. Barbosa A, Araujo N, Pordeus J P, Santos E. (2017) Using learning analytics and visualization techniques to evaluate the structure of higher education curricula. In: Proceedings of the XXVIII Brazilian Symposium on computers in education 28(1): 1297. doi:https://doi.org/10.5753/cbie.sbie.2017.1297
8. Beck J, Woolf B (2000) High-level student modeling with machine learning. Intelligent tutoring systems - ITS 2000. Lecture Notes in Computer Science, vol 1839. Springer, Berlin, Heidelberg, pp 584-593. doi:https://doi.org/10.1007/3-540-45108-0_62
9. Ha S H, Bae S M, Park S C (2000) Web mining for distance education. In: Proceedings of the 2000 IEEE International Conference on management of innovation and technology vol 2, pp 715-719. doi:https://doi.org/10.1109/ICMIT.2000.916789
10. Luan J. (2002) Data mining and knowledge management in higher education-potential applications. ERIC ED474143.
11. Ma Y, Liu B, Wong C K, Yu P S, Lee S M (2000) Targeting the right students using data mining. In: Proceedings of the 6th ACM SIGKDD International Conference on knowledge discovery and data mining, pp 457-464. doi:https://doi.org/10.1145/347090.347184
12. Romero C, Ventura S (2013) Data mining in education. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 3(1):12–27. https://doi.org/10.1002/widm.1075
13. Olaniyi AS, Abiola HM, TaofeekatTosin SI, Kayode SY, Babatunde AN (2017) Knowledge discovery from educational database using Apriori algorithm. Comput Sci Telecommun 51:1
14. Tan P N, Steinbach M, Kumar V (2005) Introduction to data mining. Boston: Pearson Addison Wesley. ISBN-13:978-0321321367
15. Van Der Aalst W, Adriansyah A, De Medeiro A K A, Arcieri F, Baier T, Blickle T, Burattin A. (2011) Process mining manifesto. In: Business process management workshops. BPM 2011. Lecture Notes in Business Information Processing, vol 99. Springer, Berlin, Heidelberg, pp 169-194. doi:https://doi.org/10.1007/978-3-642-28108-2_19
16. Campagni R, Merlini D, Sprugnoli R, Verri MC (2015) Data mining models for student careers. Expert Syst Appl 42(13): 5508–5521. https://doi.org/10.1016/j.eswa.2015.02.052
17. Asif R, Merceron A, Pathan M K (2014) Investigating performances' progress of student. In: Proceedings of the DeLFI Workshops, pp 116-123.
18. Kumar V, Chadha A (2012) Mining association rules in student's assessment data. International Journal of Computer Science Issues 9(5):211–216
19. Buldu A, Üçgün K (2010) Data mining application on students' data. Procedia Soc Behav Sci 2(2):5251–5259. https://doi.org/10.1016/j.sbspro.2010.03.855
20. Chandra E, Nandhini K (2010) Knowledge mining from student data. Eur J Sci Res 47(1):156–163
21. Oladokun VO, Adebanjo AT, Charles-Owaba OE (2008) Predicting students' academic performance using artificial neural network: a case study of an engineering course. Pac J Sci Technol 9(1):72–79
22. Gottin V, Jiménez H, Finamore A C, Casanova M A, Furtado A L, Nunes B P (2017) An analysis of degree curricula through mining student records. In: Proceedings of the IEEE 17th International Conference on advanced learning technologies, pp 276-280. doi:https://doi.org/10.1109/ICALT.2017.54
23. Leskovec J, Rajaraman A, Ullman J D (2014) Mining of massive datasets. Cambridge University Press. ISBN-13:978-1107015357
24. Agrawal R, Imieliński T, Swami A (1993) Mining association rules between sets of items in large databases. ACM SIGMOD Rec 22(2):207–216
25. Han J, Pei J, Kamber M (2011) Data mining: concepts and techniques. Elsevier.
26. Siemens G, Baker R S (2012) Learning analytics and educational data mining: towards communication and collaboration. In: Proceedings of the 2nd International Conference on learning analytics and knowledge, pp 252-254. doi:https://doi.org/10.1145/2330601.2330661

Publisher's Note