Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
Predicting student performance using sequence classification with time-based windows

Galina Deeva\textsuperscript{a,}\textsuperscript{*}, Johannes De Smedt\textsuperscript{a}, Cecilia Saint-Pierre\textsuperscript{b}, Richard Weber\textsuperscript{c}, Jochen De Weerdt\textsuperscript{a}

\textsuperscript{a} Research Centre for Information Systems Engineering, KU Leuven, Belgium
\textsuperscript{b} Online Learning Office, VTI, Universidad de Chile, Chile
\textsuperscript{c} Department of Industrial Engineering, FCFM, Instituto Sistemas Complejos de Ingeniería, Universidad de Chile, Chile

\textbf{A R T I C L E I N F O}

\textbf{Keywords:}
Machine learning
Sequence mining
Feature engineering
Success prediction
Behavioral patterns

\textbf{A B S T R A C T}

A growing number of universities worldwide use various forms of online and blended learning as part of their academic curricula. Furthermore, the recent changes caused by the COVID-19 pandemic have led to a drastic increase in importance and ubiquity of online education. Among the major advantages of e-learning is not only improving students’ learning experience and widening their educational prospects, but also an opportunity to gain insights into students’ learning processes with learning analytics. This study contributes to the topic of improving and understanding e-learning processes in the following ways. First, we demonstrate that accurate predictive models can be built based on sequential patterns derived from students’ behavioral data, which are able to identify underperforming students early in the course. Second, we investigate the specificity-generalizability trade-off in building such predictive models by investigating whether predictive models should be built for every course individually based on course-specific sequential patterns, or across several courses based on more general behavioral patterns. Finally, we present a methodology for capturing temporal aspects in behavioral data and analyze its influence on the predictive performance of the models.

The results of our improved sequence classification technique are capable to predict student performance with high levels of accuracy, reaching 90% for course-specific models.

\section{Introduction}

With the recent growth of available educational technologies, online and blended learning have become crucial components of educational processes worldwide. The COVID-19 pandemic has strengthened this tendency significantly. More and more universities implement online assignments and even complete course modules as part of their academic curricula. Besides improving students’ learning experience and widening their educational prospects, such online courses also offer an opportunity to gain insights into students’ learning processes applying learning analytics (Viberg, Hatakka, Bälter, & Mavroudi, 2018).

Different combinations of online and offline learning become more common within educational institutions, such as blended learning and flipped classroom approaches, in which different parts of the learning process are happening both online and offline, and hybrid learning approaches, where some students are present in-class, while others are joining the class online. Analyzing data from such courses is a challenging task, due to dealing with rather incomplete datasets in comparison with purely online education. The incompleteness of the data captured by blended learning platforms is attributed to the fact that a large portion of learning processes happens outside of the learning platform and thus is invisible to it. Nevertheless, a recent study of a blended learning course has indicated that it is possible to predict student performance in such a challenging setting using just behavioral data (Van Goidsenhoven et al., 2020). However, the accuracy of these predictions remains relatively low and can still be further improved.

Since it is not always feasible to address the problem of such a challenging setup of incomplete datasets with a few data points on the data level (for example, by obtaining more data), the solution can come on the algorithm level — by improving and tailoring existing algorithms to extract more information from the available data.

The main goal of this paper is to demonstrate that sequential pattern mining can be used to build models that can detect which students are likely to perform poorly early-on in the course, both in online and...
blended learning setups. In particular, this paper puts forward an extension to a standard sequence classification approach by allowing to tailor the derived sequential patterns to be discovered from absolute time windows. More specifically, we present a flexible method to pinpoint the timing of sequential patterns within a course’s time span, which can be flexibly selected by teachers and thus can facilitate interpretability. Given the importance of supporting teachers to organize their guidance more effectively and proactively, predicting student performance early-on is valuable, and with our method, we can enhance teacher control over the analysis and modeling, which is an interesting feature when specific information about the course is known.

Another key contribution of this work is the fact that, without the explicit use of sensitive socio-demographical variables including gender, ethnicity, age, social status, etc., we are capable to build accurate models for failure prediction. By only working with behavioral data, we can maximally avoid strongly undesired biases related to these socio-demographical variables and/or self-reported data.

The experimental evaluation is performed using data from nine SPOCs hosted at the Learning Management System (LMS) platform at Universidad de Chile (UCHile), called EOL. Using data from SPOCs allows us to enlarge the understanding of how incomplete data of the blended learning setting can be used for obtaining accurate and useful predictions of student performance. Moreover, another reason for focusing on private online courses as opposed to MOOCs is the additional costs involved in the case of on-campus education. Given such costs, it is generally more valuable to predict the performance of students enrolled in the university and bounded to graduate in comparison with the students who freely enroll in a MOOC without always having a clear intention to complete it.

The rest of the paper is organized as follows. First, Section 2 discusses the current state of predictive learning analytics and its two important aspects: types of data and algorithms used. Then, in Section 3, the proposed approach is compared to other sequence mining methods available in the literature, highlighting their differences and similarities, and outlining the contribution of this work. Next, Section 4 describes how sequence classification can be used in the learning analytics context. Then, in Section 5, the data and the results of the predictions are presented. Finally, Section 6 contains information regarding possible implications of our findings, and Section 7 summarizes the study.

2. Related work

Learning analytics (LA) is an important area of research mainly addressing the problem of revealing valuable and applicable knowledge from educational data (Daud et al., 2017). Its main purpose is to understand how students learn, as well as to potentially improve their learning outcomes (Viberg et al., 2018). A recent systematic literature review of LA in higher education identified the prediction of student performance as a common task (Viberg et al., 2018). This task is typically approached with classification and regression methods (Viberg et al., 2018): to predict a category of the learning outcome such as ‘passed’ and ‘failed’, or to predict a final score, respectively.

The studies that are focused on predicting student performance can be analyzed from different angles, but generally, two main dimensions can be utilized for their categorization. The first dimension deals with the kind of data and, consequently, features being used. Some authors aim to predict student performance based on students’ demographic data, their personality, or other survey-based data; see, e.g., Daud et al. (2017). Among other data which contributed to predicting student success are cumulative grade point average (GPA), assignments and quiz marks (Papamitsiou, Terzis, & Economides, 2014), past performances in previous assessments and entry tests, and demographics (Leitner, Khalil, & Ebner, 2017). Shahiri, Husain, et al. (2015) broadly discussed the factors that contribute to the prediction of students’ performance.

Recently, data that represents the interactions of students with a learning system have been used in predictive analytics. Agudo-Peregrina, Iglesias-Pradas, Conde-González, and Hernández-García (2014) analyzed six online and two face-to-face courses to investigate whether students’ interaction data can be exploited to predict academic performance. Not unexpectedly, they show that this is possible only in online and not in face-to-face courses. Other studies that use behavioral data from online learning platforms can be found in, e.g., Waheed et al. (2020), Romero, Espejo, Zafra, Romero, and Ventura (2013) and Popescu and Leon (2018).

The second dimension for a categorization of predictive learning analytics research is the data analysis technique being used. As discussed above, various supervised machine learning approaches, such as classification and regression, are widely used, but other techniques are also observed in the literature. For instance, Daud et al. (2017) built five predictive models based on different algorithms, namely SVMs, CART, C4.5, Bayesian Networks, and Naive Bayes. Additionally, the authors of this study provide an overview of the most typical data mining methods used in predictive learning analytics. Besides the aforementioned techniques, the methods that are listed are neural network models, induction rules, association rules, and genetic algorithms, among others.

The actual process of feature engineering based on behavioral data is rarely discussed in detail, even though it is an important and challenging step. In the literature, it is common to observe relatively handcrafted and generally expert-driven features, i.e., proposed by the teacher and manually calculated rather than automatically discovered. As a result, such features could be rather simplistic in nature. For example, some of the features extracted from behavioral data by Minaei-Bidgoli, Kashy, Kortemeyer, and Punch (2003) are the total number of correct answers, the total number of tries for doing homework, the total time spent on the problem, etc. Similarly, Waheed et al. (2020) provide another example of constructing handcrafted features, such as the total number of clicks on a particular module, the number of attempts, and the number of assignments submitted.

The common approach to featurize behavioral data with activity (click) counts is, however, not always capable to capture the information regarding the order between the clicks and other temporal aspects. Sequence mining presents itself as a capable technique for featurization of behavioral data because of its ability to capture the ordinal relationship between items in a sequence. Examples of sequence mining used in education can be found in studies by Kinnebrew, Loretz, and Biswas (2013), Kotsiantis, Pierrakeas, and Pintelas (2004), and Deeva, De Smedt, De Koninck, and De Weerdt (2017). This study continues the work on this topic to discover better and smarter predictive and featurization algorithms for predicting student performance.

3. Comparative analysis and contributions

In this section, we provide an overview of the most common sequence mining techniques as well as their key differences so as to position our approach. Table 1 depicts a comparative assessment in terms of four key dimensions which are both common in the sequence mining literature, as well as indicate the points of differentiation of the technique used in this work: (1) the type of algorithm, (2) the algorithm being support-based, (3) the pattern type used, and (4) the use of windows. We denote our approach as iBCM*, given that it is an adaptation of the original iBCM algorithm (De Smedt, Deeva, & De Weerdt, 2019).

First, sequence classification can be achieved in two ways: indirect or direct. For the former, a regular unsupervised sequential pattern mining algorithm is applied with the derived patterns subsequently being provided to a classification algorithm. The latter techniques make use of the label information to discover the most discriminating sequential patterns first, before similarly applying a classification algorithm of choice. iBCM* is an indirect algorithm, however, as pointed out.
by De Smedt et al. (2019), it is outperforming direct algorithms such as BIDE(DC) (Fradkin & Mörchen, 2015), MiSeRe (Egho, Gay, Bouillé, Voisine, & Clérot, 2015), and ISM (Masseglia, Poncelet, & Teisseire, 2003), in terms of classification performance on a wide range of sequence classification datasets.

Second, Table 1 shows a strong consistency in the support-based nature of a large majority of techniques. MiSeRe is the only exception, as it is essentially a parameter-free algorithm relying on a Bayesian reconstruction approach. The support-based nature of iBCM is important because in the application to behavioral learning data, we require a flexible and tunable technique, given that the sequence data can be of very different granularity levels and that differentiating patterns in terms of student success can be complex. This is demonstrated in our experimental evaluation in Section 5.

More relevant for pinpointing the contribution of this work, is the comparison in terms of pattern types and windows. A majority of sequence mining techniques rely on partial orders (e.g. GoKrimp (Lam, Mörchen, Fradkin, & Calders, 2014), SCIS (Zhou, Cule, & Goethals, 2015), BIDE (Wang & Han, 2004), and SPADE (Zaki, 2001)). However, constraint-based approaches such as cSPADE (Zaki, 2000) and iBCM allow for more flexible and richer patterns to be derived from the sequences. While cSPADE allows for typical constraints such as gap, item, and time constraints, iBCM relies on a comprehensive set of declarative behavioral templates, as further detailed in Section 4.2, which intrinsically cover gap and item constraints. In De Smedt et al. (2019), it was shown that this richness in derived sequential patterns has a positive impact on classification performance as well as the conciseness of the retrieved pattern base. As for student success prediction, the increased diversity in possible sequential patterns is desirable, given that a typical dataset usually contains relatively few observations, especially compared to sequence databases in other fields such as biology, physics, web mining, or text mining. Nevertheless, iBCM has also been used and benchmarked on smaller sequence database (De Smedt et al., 2019) as well as in the context of session stitching and web mining of large web logs (De Smedt, Lacka, Nita, Kohls, & Paton, 2021), indicating that these constraints serve predictive purposes in a multitude of contexts.

Finally, the last comparison dimension relates to window configuration and time. While many sequence mining algorithms could be configured to do something similar, iBCM is the only technique that can natively work with windows. This entails that iBCM can be restricted to discover sequential patterns from subsequences defined based on their relative length. For instance, an original complete sequence with 20 items, can be subdivided into 4 subsequences of 5 items, with patterns only derived within each window. The core contribution of this paper is that we extend this window-based approach to absolute time windows. This is particularly relevant for characterizing sequences pertaining to learning processes by students, given that time is a crucial factor in the characterization of how students engage and learn.

In conclusion, the core contribution of this work is a learning analytics-oriented extension of iBCM to allow for the construction of student success prediction models based on behavioral data. By combining the comprehensive set of constraint types of iBCM with custom absolute time-based windows, we can better capture the temporal aspects affecting student success.

4. Predicting student performance using sequence classification: methodology

In this section, the core methodology for building predictive models based on sequence classification is presented. We use the iBCM (interesting Behavioral Constraint Miner) sequence classification algorithm introduced by De Smedt et al. (2019) for deriving sequential patterns. iBCM is a state-of-the-art sequence classification technique supporting a wide range of pattern types, which makes it a particularly suitable technique for adapting it to a specific prediction task. iBCM can be used for featurizing learners’ activity sequences and requires to be paired with a classification algorithm to construct a predictive model.

In what follows, behavioral data in the context of success prediction is discussed. Then, the way sequential patterns are extracted is outlined and followed by an explanation of how to build a predictive model. Finally, an adaptation of the baseline sequence classification technique is introduced that can capture more specific learning-driven temporal aspects in sequences through time-based windows.

4.1. Analyzing behavioral data from an e-learning platform

In the context of online learning, one of the most easily available and unambiguous sources of student data is the student activity recorded by an e-learning platform. Other types of data exist as well and could potentially provide valuable information for predicting student performance, e.g., demographic data such as age and previous education, students’ affective states and their transitions over time, students’ personality characteristics such as level of neuroticism and achievement striving. However, these types of data are not always available for all students and/or might suffer from reporting bias. Moreover, demographic data can easily lead to biased models in terms of gender, ethnical characteristics, etc. By only using behavioral data, we aim at building models that are as unbiased as possible with respect to personal characteristics.

Tracking data, also often referred to as behavioral data, represents student behavior captured by an e-learning platform. In this paper, we analyze the data from the EOL platform, the edX-based LMS used in UChile to run courses for undergraduate and graduate schools as well as professional education and MOOCs. This LMS provides tracking data in a streaming newline-delimited JSON (NDJSON) format. More specifically, the data in NDJSON format is a collection of lines, each of which is a valid nested JSON structure. These data can be transformed into a sequence as illustrated in Fig. 1. Firstly, it can be flattened and unnested to obtain tabular data, containing timestamp, username, event, and its context, i.e. chapter and section of the course. Generally, there are several possible ways to obtain a sequence from such data. As shown by Deeva et al. (2017), three levels of event granularity can be considered: the most general level with event information only, an intermediate level in which the event name is combined with the chapter information and, finally, the most fine-granular level in which event names are combined with chapter and section.

Based on a chosen activity granularity level, one can obtain sequences of events, in sequence mining also called as sequences of items. In the remainder of this paper, “event” and “item” are used to refer to the recording of a student action within the platform, characterized by a particular timestamp.
4.2. Mining sequential patterns from behavioral data using iBCM

The iBCM algorithm uses label information to derive the most discriminative patterns for a certain class of sequences. These patterns are then used as binary features indicating a presence or absence of a pattern, which allows to transform a set of sequences into a feature matrix, as shown in Fig. 2. iBCM has been previously shown to achieve high levels of performance in terms of model accuracy in a relatively short computational time (De Smedt et al., 2019).

One of the key advantages of iBCM is the richness of the sequential patterns that it supports, as it combines higher expressiveness compared to partial orders with a hierarchy of sequential behavior ranging from a simple order to a next-in-sequence and alternating behavior. Some of the other sequence mining algorithms, e.g. BIDE and PrefixSpan, use partial orders, while iBCM is capable to exploit a whole hierarchy of sequential patterns. The sequential patterns that can be discovered by iBCM are based on a set of Declare language templates (Di Ciccio & Mecella, 2013). These patterns are widely used for identifying not only sequential, but also overall behavioral characteristics of programs and processes. The Declare template base consists of a number of patterns for modeling flexible business processes, and it is extensive. The set of templates that is used by the current version of iBCM is listed in Table 2.

| Type          | Template | Regular Expression |
|---------------|----------|--------------------|
| Unary         | Existence(A,n) | .*(A.*)(n)         |
|              | Absence(A,n) | [ˆA] *(A[A] *) *(n-1) |
|              | Exactly(A,n) | [ˆA] *[A][A] * *(n) |
|              | Init(A)     | (A.*)?             |
|              | Last(A)     | .A                 |
| Unordered     | Responded existence(A,B) | [ˆA] *(A.*B.*) *[B.*A.*) ? |
|              | Co-existence(A,B) | [B.*A.*] *[A.*B.*) ? |
| Simple ordered| Response(A,B) | [ˆA] *[A.*B.*) *[A.*] |
|              | Precedence(A,B) | [B.*A.*] *[B.*] |
|              | Succession(A,B) | [B.*][A][B.*] |
| Alternating reordered | Alternate response(A,B) | [ˆA] *[A][A][B][A] * |
|              | Alternate precedence(A,B) | [B.*][A][B][B][B] * |
|              | Alternate succession(A,B) | [B][A][B][B][B][B] * |
| Chain ordered | Chain response(A,B) | [A][AB][A] * |
|              | Chain precedence(A,B) | [B][AB][B] * |
|              | Chain succession(A,B) | [AB][AB][AB][AB] * |
| Negative      | Not co-existence(A,B) | [ˆB][AB] *[B][A] * |
|              | Not succession(A,B) | [ˆA][AB][B][A] * |
|              | Not chain succession(A,B) | [A][AB][AB][AB][AB] * |
| Choice        | Choice(A,B) | .*(AB) |
|              | Exclusive choice(A,B) | *(B)[A][B][A] * |

As illustrated in Fig. 2, after sequences are featurized, i.e., transformed into a feature matrix using the iBCM-derived patterns as features, a classification algorithm can be applied. There are several aspects that need to be considered when doing so. First of all, to prevent potential data leakage and to ensure actionability of the model, only the data (sequences) up to a certain time $T$ can be used to predict student performance (Deeva et al., 2017). In terms of data leakage, using the data from the whole course might potentially contain cases of students that already dropped out from the course or simply engaged

| Course-specific |  |
|-----------------|---|
| Sequence        | Label |
| ade             | Pass  |
| bc              | Fail  |

Possible interpretation:
a=watch video  
b=answer question  
c=read explanation  
d=pause video  
e=close page

Fig. 1. An example of behavioral data and its transformations needed to perform a sequential data analysis.
in much less active participation on the platform, which might bias the algorithm. In terms of actionability of the prediction, it is more relevant and practical to pinpoint potential students at-risk as early as possible during the course, as this could help the teachers to support these students before the course is finished. On the other hand, there is a trade-off between early prediction and the amount of data available for building a model. Therefore, it is important to choose a suitable moment $T$, which would be an optimal point within the course for making the prediction.

### 4.4. Capturing temporal aspects

The advantage of sequence classification for analyzing temporal data in comparison with simpler featurization methods, such as for example the bag-of-words approach where number of occurrences of each item is used as a feature, is that it can take into account not only the type of items in the sequences, but also their order. However, in case of educational data, there could be situations in which the order of items is not enough to differentiate between students with different performance levels. Fig. 3 shows an example that illustrates this idea using a four-week course. Each week could be considered as a time window. Three students have performed the same sequence of activities, namely $\langle abcd\rangle$, and therefore will be treated way by a sequence classification algorithm and will be predicted to achieve the same level of performance. However, as is clear from the figure, the first student has been working consistently throughout the semester, whereas the second student has completed all the work at the last possible moment, i.e. within the last week/window. Given that previous research in learning analytics has shown that the way the students approach deadlines is correlated with their final results (Celis, Moreno, Poblete, Villanueva, & Weber, 2015), we could potentially expect the first student to perform better than the second.

To capture this difference, we introduce a notion of time-based windows that can be implemented with iBCM. Currently iBCM already has a functionality that allows to pinpoint patterns that are found in a particular window. However, these windows are defined based on the number of items. More specifically, a complete sequence is divided into $n$ windows of equal size, i.e., equifrequent windows, where $n$ is a user-defined parameter. This means that with $n = 4$, a sequence $\langle abcd\rangle$, which consists of eight items, will be divided into 4 sub-sequences of 2 items each. In case of educational data, this abstraction will facilitate yielding more fine-granular patterns, but still will not capture the difference between students that demonstrated different attitude in terms of time. Therefore, in this paper we propose to tailor iBCM to discover patterns based on equidistant windows by taking into account absolute times. As such, iBCM will be capable of picking up the temporal differences between students in the discovered patterns. While the equifrequent windows version of iBCM would discover the same set of patterns in the sequences of three students shown in Fig. 3, the equidistant windows version would yield three different sets of behavioral patterns.

### 5. Experimental evaluation

#### 5.1. Data description

This study analyzes behavioral data from nine SPOCs, as summarized in Table 2. The models for three courses with the highest number of students are built, namely FCS-001, ENFGP243, and ENMAN303.

In the scoring system used by these courses, 7 is the best score and 4 the worst score to pass. The relatively low percentage of students who failed a course (roughly 7% for all courses combined, and 0% for some of the courses) makes the prediction of student success even more
challenging, given the data imbalance and small number of observations (i.e. students). To surmount this challenge, we aim at predicting the students who performed in the bottom 20 percent.

For each of these three courses, predictive models for general and course-specific patterns are constructed, with the goal of comparing their performance and thus investigating whether there is a significant difference between these two cases. Moreover, to assess the possibility of generalizing the patterns across multiple courses, another set of predictive models is built for all nine courses combined. This setting can only be applied in case of general patterns since course-specific patterns cannot be generalized for a combination of all the datasets. The results of the algorithm performance for all courses combined allow to draw conclusions not only regarding the usability of general patterns, but also whether such indicative patterns are common across different courses and can be generalizable.

Furthermore, the performance of iBCM with temporal windows is studied by comparing three setups. First, iBCM is applied in its default mode, i.e., patterns are captured in the whole sequence. Secondly, the time-based window approach is applied to obtain subsequences based on half of the analyzed timeline, so resulting in two equidistant windows. Finally, the approach is also applied so that subsequences are formed that are covering one fourth of the analyzed timeline, so based on half of the analyzed timeline, so resulting in two equidistant windows.

The data characteristics at course level are summarized in Table 3. The first column represents the code of the course and its duration; the second column shows the total number of students ($N_f$). The distribution of the course grade is summarized in the third column, where the mean and the median of the courses’ grades are given. $Q_{20}$ is the grade of the student who scored in the 20th percentile, which means that all the students who scored below this number are marked as the predicted target variable. $N_{20}$ is the number of students whose grade is below $Q_{20}$. Finally, the fourth column provides the total numbers of events per course (Total), as well as alphabet sizes for the general and course-specific event granularity ($N_{items}$).

All experiments have been performed using four classifiers: Support Vector Machines (SVM), logistic regression, Random Forest, and Naïve Bayes. Random minority class oversampling is applied as a method for handling class imbalance for all the experiments. The evaluation of the models is performed using 10-fold cross-validation. Given that SVM has achieved the highest level of performance on average, all results are reported using SVM.

Finally, iBCM uses the hyperparameter called “support”, which is defined as a proportion of analyzed sequences which needs to contain the pattern for it to be considered frequent. For example, a support value of 0.6 implies that 60% of sequences must contain a certain pattern to call it frequent. To eliminate potential influences of the support level on other investigated parameters, all results are calculated for three different support levels: 0.4, 0.6, and 0.8.

### 5.2. Results

The results of the experimental evaluation are summarized in Tables 4 and 5, using the following three performance metrics:

1. Accuracy, i.e. the percentage of correct classifications;
2. AUC (area under the ROC curve), which measures the trade-off between the true and false positive rates over different classification thresholds;
3. Top decile lift (from here on: lift), which measures how much better a classification model is, compared to a random selection, considering the top 10% predicted instances with the highest prediction probability.

Overall, sequence classification with iBCM is able to achieve good performance, with the average accuracy, AUC, and lift for all experiments equal to 0.7, 0.65, and 2.46, respectively. Moreover, some of the constructed models achieve very high levels of performance, e.g., the

### Tables

#### Table 3

Overview of the courses analyzed in this study, including their duration, score distributions, total number of events and alphabet sizes for the abstract and course-specific event granularity.

| Course Code | Duration | # of students | Score distribution 20th percentile | # of events Alphabet size |
|-------------|----------|---------------|-----------------------------------|--------------------------|
| FCS-001     | 22 weeks | $N_f = 247$   | mean = 5.63, median = 5.90, $Q_{20} = 5.00, N_{20} = 47$ | Total = 49,151, $N_{items} = 23 (201)$ |
| ENGF243     | 16 weeks | $N_f = 214$   | mean = 6.24, median = 6.33, $Q_{20} = 6.04, N_{20} = 39$ | Total = 110,758, $N_{items} = 24 (179)$ |
| ENMAN303    | 17 weeks | $N_f = 147$   | mean = 5.78, median = 6.40, $Q_{20} = 5.50, N_{20} = 28$ | Total = 62,329, $N_{items} = 28 (228)$ |
| ENENG288    | 15 weeks | $N_f = 94$    | mean = 5.26, median = 5.70, $Q_{20} = 4.80, N_{20} = 17$ | Total = 44,809, $N_{items} = 28$ |
| CFG-TUT-V1  | 20 weeks | $N_f = 75$    | mean = 4.80, median = 5.90, $Q_{20} = 4.60, N_{20} = 14$ | Total = 8,818, $N_{items} = 22$ |
| CIE-115.418 | 16 weeks | $N_f = 60$    | mean = 5.43, median = 5.40, $Q_{20} = 5.00, N_{20} = 11$ | Total = 15,853, $N_{items} = 19$ |
| TUT-CFP-DMC | 10 weeks | $N_f = 24$    | Pax/Fail: $N_{20} = 4$ | Total = 1,556, $N_{items} = 23$ |
| ENGF106     | 17 weeks | $N_f = 23$    | mean = 6.22, median = 6.20, $Q_{20} = 5.90, N_{20} = 4$ | Total = 13,696, $N_{items} = 23$ |
| CIE-FC720   | 7 weeks  | $N_f = 14$    | mean = 5.49, median = 5.50, $Q_{20} = 4.80, N_{20} = 2$ | Total = 4,406, $N_{items} = 19$ |
| All courses | $N_f = 895$ | $N_{20} = 163$ | Total = 311,386, $N_{items} = 30$ |

---

Fig. 3. An example of the situation where the information regarding temporal aspect could be substantial for predicting student performance.
model for course-specific patterns for the ENMAN303 course, where the metrics are $\text{accuracy} = 0.91$, $\text{AUC} = 0.9$, and $\text{lift} = 5.03$ (see Table 4). The reason the models built for this course were capable of achieving the best predictive performance among all the analyzed courses is that this is the only course in which the learning process occurs completely online, unlike the others which utilized a flipped classroom approach. Overall, these results provide additional evidence that sequence classification in general, and iBCM in particular, form a suitable and efficient tool for mining educational data.

A few additional trends can be observed. First, the models built based on course-specific patterns outperform general patterns with an average AUC score of 0.7 and a lift of 3.22 compared to an average AUC equal to 0.63 and lift of 2.43, respectively. This observation is not entirely unexpected; however, what is interesting is that the performance of the general-pattern-based models is still relatively good, especially if the complexity of the blended learning setting is considered. For instance, the performance metrics for the models built for ENFGF243 reach an accuracy of 0.76, AUC of 0.71 and lift of 3.24, which means that using more coarse-granular patterns can sometimes be sufficient for predicting student success. It can be concluded that a larger alphabet size with more detailed information will most likely lead to better predictions; however, if generalizability of the results is important, the coarser level of granularity could be used as well, even at the levels when no information about the tackled object is known and only the action information is provided to the algorithm.

Similarly to the results based on general patterns for the other courses, the predictive models for all courses combined do not achieve very strong predictive performance, but still are able to outperform a random model with the highest AUC equal to 0.64. One of the reasons for the complexity of predicting students scoring in the bottom 20% is the differences in scoring across different courses.

Moreover, as can be seen in Table 3, the score distributions for some courses are characterized by low variability, with e.g., the lowest scores of 5 and the highest score of 6. Given such low variability, the behavior of students whose score is in the bottom 20%, might not differ sufficiently from those obtaining a higher score. Naturally, predicting groups with more distinct behavior is a less challenging task, and therefore, for courses with a higher number of low-scoring cases, the prediction performance of sequence classification models based on general patterns will likely be higher.

The models with support $= 0.6$ achieve the best results for all three metrics analyzed (with $\text{accuracy} = 0.72$, $\text{AUC} = 0.66$, and $\text{lift} = 2.77$ on average). The lower the support, the more patterns are discovered by iBCM, which leads to better predictions when support is lowered from 80% to 60% as more patterns are correlated with better predictions. Conversely, when support is further lowered to 0.4, more of the discovered patterns become irrelevant, and the predictive performance decreases.

The influence of time windows on predictive performance can be observed in Tables 4 and 5. In two-thirds (14 out of 21) of the combinations of the course, pattern type and support, the performance metrics are higher for the cases of 2 and 4 windows when compared with 1 window. For example, for ENMAN303 with general patterns, the AUC score increases from roughly 0.6 to 0.8 when the number of
windows is increased from 1 to 2. However, this trend is not present for all the cases. For example, for the two-windows setup, the models with four windows tend to perform worse than those with two windows. There are several factors that can play a role in such a finding. First, it is to be expected that temporal information allows to grasp more specific and informative patterns when compared to when no windows are used. However, the way the windows were defined in the current set of experiments – i.e., by dividing the course timeline in a set of equal time intervals – might not correspond to a meaningful time division within a course. Mining patterns based on when the course content was released might be more beneficial in this context. Second, it is important to note that discovering frequent patterns in the whole sequence is easier for iBCM when compared to mining patterns in a subsequence limited to a certain window. As such, lowering the support level might allow to ease the requirements for the pattern to be considered frequent, and could facilitate discovering more patterns in the time-based subsequences.

### 5.3. Pattern interpretation

As shown above, sequence classification techniques are capable of discovering a large number – sometimes thousands – of meaningful patterns, which can be used to build accurate predictive models. However, given a large number of generated patterns, a possible usage of feature space transformation techniques and the fact that classification algorithms often serve as black boxes, it is not always easy to interpret the results of the models at the pattern level. Moreover, it is important to be cautious when interpreting causal relationship between patterns and student performance, since correlation between a desirable learning outcome and the pattern does not always imply that following this pattern is recommended. Nevertheless, relative importance of sequential patterns in a predictive model could provide the educators with an indication towards which student actions might have had the largest influence to the student performance, and a direction for further investigation of the possible reasons of this influence. Ideally, such an investigation could lead to either supporting the discovered patterns by existing educational theories, or to emerging of new educational theories.

This section provides examples of important sequential patterns. Since it is even more conceptually interesting to investigate which abstract patterns are important, as it provides more opportunities to generalize, we illustrate the types of sequential patterns based on the results of the best-performing model for the abstract pattern types, which is the model built for ENMAN303, with support equal to 0.8 and with 2 time windows.

As discussed in Section 4, iBCM is in essence a feature engineering technique, which extracts binary features from learners’ activity sequences, and as such, it can be combined with any classification algorithm, such as SVM, random forest, logistic regression, etc. Although we use SVM as the models for calculating the results in the previous subsection due to its overall higher performance with sequential patterns, random forest can provide a better tool for visualizing feature importance in a machine learning model. Therefore, in this subsection, random forest is used for illustration of important patterns. The AUC of the models built by random forest and SVM has been equal to 0.8 for both algorithms for the selected case (ENMAN303, abstract patterns, support of 0.8, and w2).

The 15 most important sequential patterns discovered by iBCM are shown in Fig. 4 (the interpretation of the numerical encoding of the events is provided in the caption of the figure). Given the nature of a random forest model, gaining a full understanding of the decision process based on this features is infeasible. Nevertheless, the potential meaning of why these behavioral patterns are important for
The results reveal that, although a larger alphabet size leads to discovering more informative pattern, more general non-course specific explanations that could improve their learning outcomes. Nonetheless, irrespectively of whether an increase in performance metrics is present for temporal windows, being able to pinpoint the exact timing of a pattern in a sequence is in itself an interesting feature. Such functionality can provide better interpretation of the results and facilitate the translation of the discovered patterns to the course context. Moreover, if the knowledge of the teachers about the desired pattern timing is considered, the time window functionality could be used to observe whether the students follow the learning path foreseen by the teacher.

When it comes to event granularity, a larger alphabet size containing more information about the student actions will most likely improve the predictive performance of the models. Notwithstanding, this not only comes at the price of the results’ generalizability, but also requires more computational time, as the running times of the experiments have been increasing proportionally to the number of sequential patterns discovered. Besides, when a 10-fold cross validation setup is utilized, the presence of such an increased number of patterns (e.g. over 10,000 patterns for the ENMAN303 models) need to be discovered for each fold separately, which is computationally even more expensive.

7. Conclusion and future work

This paper has demonstrated the construction of predictive models for detecting underperforming students based on behavioral data. Hereto, sequence classification is used to derive behavioral patterns correlated with student performance, which can be transformed into recommendations for improving the quality of online education at UCChile and beyond. The possibility to identify the students that will score in the bottom 20% of the class will allow teachers to potentially target those students providing better support and offer them additional explanations that could improve their learning outcomes.

The results reveal that, although a larger alphabet size leads to discovering more informative pattern, more general non-course specific information could be used for constructing predictive models as well. In addition, the utility of the proposed time window functionality is discussed in terms of both predictive performance as well as explainability. Future research could focus on exploring this topic in more depth, by comparing different windowing setups in iBCM. For example,
the performance of the equifrequent vs. equidistant approach can be compared. Moreover, different ways of dividing the course timeline could be investigated.

**CRediT authorship contribution statement**

**Galina Deeva**: Conceptualization, Methodology, Software, Formal analysis, Writing – original draft, Visualization. **Johannes De Smedt**: Software, Validation, Writing – original draft, Writing – review & editing. **Cecilia Saint-Pierre**: Investigation, Resources, Data curation, Writing – original draft. **Richard Weber**: Investigation, Resources, Data curation, Writing – original draft. **Jochen De Weerdt**: Conceptualization, Methodology, Writing – original draft, Writing – review & editing, Supervision.

**Declaration of competing interest**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Acknowledgments**

RW gratefully acknowledges financial support from ANID, Chile PIA/BASAL AFB180003, Fondecyt, Chile 1181036, Fondecyt, Chile 1221562, and FONDEF, Chile ID18110216: “Desarrollo de tecnologías de big data para aumentar la retención y el éxito de estudiantes universitarios”.

All authors acknowledge financial support from NeEDS Secondents “Research and Innovation Staff Exchange Network of European Data Scientists” (NeEDS); H2020-MSCA-RISE-2018, Project ID 822214.

**References**

Agudo-Peregrina, A. F., Iglesias-Pradas, S., Conde-González, M. A., & Hernández-García, A. (2014). Can we predict success from log data in VLEs? Classification of learning interactions for learning analytics and their relation with performance in VLE-supported F2F and online learning. *Computers in Human Behavior*, 31, 542–550.

Celis, S., Moreno, L., Poblete, P., Villanueva, J., & Weber, R. (2015). Un modelo analítico para la predicción del rendimiento académico de estudiantes de ingeniería. *Revista Ingeniería de Sistemas Volumen XXIX*.

Daud, A., Aljohani, N. R., Abbasi, R. A., Lytras, M. D., Abbasi, F., & Alowibdi, J. S. (2017). Predicting student performance using advanced learning analytics. In *Proceedings of the 26th international conference on world wide web companion* (pp. 415–421).

De Smedt, J., Deeva, G., & De Weerdt, J. (2019). Mining behavioral sequence constraints for classification. *IEEE Transactions on Knowledge and Data Engineering*, 32(6), 1130–1142.

De Smedt, J., Lacka, E., Nita, S., Kohls, H.-H., & Paton, R. (2021). Session stitching using sequence fingerprinting for web page visits. *Decision Support Systems*, 150, Article 113579.