


Chapter 15

Best Practices in Dropout Prediction: Experience-Based Recommendations for Institutional Implementation

Juan J. Alcolea

 <https://orcid.org/0000-0003-1593-9064>

DIMETRICAL, The Analytics Lab, Spain

Alvaro Ortigosa

Universidad Autonoma de Madrid, Spain

Rosa M. Carro

 <https://orcid.org/0000-0001-9684-5179>

Universidad Autonoma de Madrid, Spain

Oscar J. Blanco

 <https://orcid.org/0000-0002-6166-907X>

DIMETRICAL, The Analytics Lab, Spain

ABSTRACT

This chapter focuses on the key practical aspects to be considered when facing the task of developing predictive models for student learning outcomes. It is based on the authors' experience building and delivering dropout prediction models within higher education contexts. The chapter presents the information used to generate the predictive models, how this information is treated, how the models are fed, which types of algorithms have been used, and why and how the obtained results have been evaluated. It recommends best practices for building, training, and evaluating predictive models. It is hoped that readers will find these recommendations useful for the design, development, deployment, and use of early warning systems.

DOI: 10.4018/978-1-7998-5074-8.ch015

INTRODUCTION

Over the last few years, the authors of this chapter have led the design of an innovative early warning system. The client has been a large online university, in which student retention is crucial for business success. In this context, the authors have learned some valuable lessons about the requirements for implementing a system in a higher education setting, which goes beyond the scope of research and pilot testing. According to the authors' experience, when dealing with real users and real needs, there is a necessity to focus not only on the parameters that are important in the research stage (such as accuracy), but also on much wider user-oriented, design-related issues.

The purpose of this chapter is to take the reader through a set of experience-based recommendations for the design, development, and implementation of a system which predicts student dropout in higher education contexts. The information available in universities regarding their students is usually rich and diverse (e.g., academic systems, learning management systems, CRM systems, etc.) and can be used for risk-prediction and retention purposes.

Research has shown that it is possible to predict student dropout risk as early as day one (Berens, Schneider, Görtz, Oster, & Burghoff, 2019). Research has also shown that student-activity logs are a key resource for gaining insight into student behavior in online courses. Analysis of observed behavior patterns is a necessary step to calculate a frequent and up-to-date dropout risk. Either way, the goal is the same: to intervene as early as possible to prevent students from dropping out and to help them to complete the course successfully.

Dropout risk should be considered a dynamic variable that should be recalculated over time and should evolve according to the information available at each milestone. For a new student who has just enrolled in a program, information is limited to data collected during student registration. However, throughout the academic period, this information is enriched by the student's interactions with the learning environment, their progress, logins, average test scores, etc. Dropout risk may also be predicted using archived data of students enrolled in a 2- or 4-year college program.

The purpose of this chapter, therefore, is to provide a set of recommendations to build predictive dropout systems. These recommendations are based on the authors' experience in using pre-course data, student activity logs, and archived data. The chapter is organized as follows: The Background section contextualizes this work and provides a brief background on design-related issues which need attention. The Design and Development Framework section is organized into four subsections: First, the theoretical rationale behind this work is presented. Then we discuss critical questions to be addressed at the planning stage in order to reveal possible underlying assumptions that if not made explicit, may hinder the project at later stages. Next, the authors discuss various techniques related to feature engineering in dropout prediction contexts. Finally, based on their experience, the authors focus on issues related to the evaluation of early warning systems.

BACKGROUND

With digitalization and the rise of e-learning, a range of computational tools and approaches have emerged, which allow educators to better support the learner's experience in schools, colleges, and universities (Freitas et al., 2015). One of the key benefits of digital tools is that a large amount of student data can help course administrators gain insight into student online-learning behavior, thus enabling administra-

Best Practices in Dropout Prediction

tors to answer important questions about students' learning habits, effective and poor teaching practices, etc. (Nunn, Avella, Kanai, & Kebritchi, 2016).

In this context, three new research fields, which are all very closely related, have emerged: educational data mining (EDM), academic analytics (AA) and learning analytics (LA) (Bienkowski, Feng, & Means, 2012; Elias, 2011). EDM is the use of data mining algorithms to solve educational issues (Baker & Yacef, 2009; Romero & Ventura, 2010). AA refers to the application of the principles and tools of business intelligence to academia to improve educational institutions' decision-making and performance (Campbell, De Blois, & Oblinger, 2007). Finally, LA (also known as predictive learning analytics or PLA) uses predictive models that provide actionable information. It is a multi-disciplinary approach based on data processing, technology-learning enhancement, EDM and visualization (Scheffel, Drachsler, Stoyanov, & Specht, 2014). Course data is used to improve learning and teaching (Ferguson, Clow, Griffiths, & Brasher, 2019). While LA focuses on the application of known methods and models to address issues affecting student learning, EDM focuses on the development of new computational data analysis methods (Bienkowski, Feng, & Means, 2012).

In this context, one of the main uses of LA is to predict student performance (Alyahyan & Düşteğör, 2020; Helal et al., 2018; Polyzou & Karypis, 2019) and predict dropout risk (Hung, Shelton, Yang, & Du, 2019): mainly detecting students at risk of failing the course (Herodotou, Rienties, Verdin, & Boroowa, 2019; Olivé, Huynh, Reynolds, Dougiamas, & Wiese, 2019) or dropping out of the institution (Badr, Algobail, Almutairi, & Almutery, 2016; Dalipi, Imran, & Kastrati, 2018). These dropout-prediction systems evolved to dropout early warning systems (Knowles, 2015; Marquez-Vera, 2016).

While early warning systems are considered to be an important mechanism to reduce student dropout (Badr et al., 2016; Knowles, 2015; Romero & Ventura, 2019), research in the area of LA best practices and recommendations for the integration of predictive LA solutions into higher education institutions is scarce (Herodotou et al., 2019). Research has mainly focused on relatively small samples, or it has used variables expensive to collect, which limits its generalizability (Sandoval, Gonzalez, Alarcon, & Pichara, 2018).

One of the main problems is that large-scale application of predictive models faces challenges which are not present in small-scale research projects. Further, project stakeholders often do not anticipate these challenges when transitioning from theoretical to practical contexts (discussed in Ortigosa et al., 2019). Consequently, this chapter focuses on explaining the rationale behind design decisions when building predictive models, which often involve large-scale e-learning scenarios. The goal is to take advantage of technology to deliver usable and actionable systems, which are able to overcome resistance to adoption, and which can produce useful and timely predictions, as well as support effective retention actions.

DESIGN AND DEVELOPMENT FRAMEWORK

Contextualization

The guidelines offered in this text emerge from the authors' four-year experience designing, developing, testing, deploying, evaluating, and maintaining a dropout prevention system based on several predictive models in a distance-learning, higher education context. After conducting a feasibility analysis, the system was built and deployed in 2017 and is currently in active use. It is actively maintained and evolves continuously.

Best Practices in Dropout Prediction

The project team consists of a combination of four experts with different profiles: two from industry and two from academia. Each of them has more than twenty years of practice or research experience in analytics for the higher education sector.

Bearing the above in mind, the guidelines outlined in this chapter, rather than being the result of a research process, are more the result of a retrospective analysis carried out by the team members as to what was or wasn't successful, where they experienced major difficulties, and how they coped with them.

Design: Making Assumptions Explicit

The first step when designing predictive analytics systems is to properly define the specific business problem/solution pair: a complete understanding of all the aspects involved is essential for developing an adequate predictive-modelling strategy. The risk of not investing time in a proper and shared business problem/solution definition is that different stakeholders silently take for granted personal assumptions, which, at a later stage, may reveal themselves as divergent, posing a serious issue with coherence and expectations. That is why an initial effort is needed to define the business problem/solution properly and make all the assumptions explicit. The following questions represent, in the authors' experience, a bare minimum to achieve this goal.

Which Event(s) Will Be Predicted?

This question may sound simple and straightforward, but it is not. While the answer can typically be expressed in a single simple sentence, the concepts used in that sentence are key and usually need to be carefully defined from functional and technical points of view. Typically, at this point, hidden complexity and ambiguity start to arise. The following sample answers illustrate this fact:

“We want to predict student dropout.” A simple statement, but:

- In this problem's context, what does “student” mean? All the students? Only freshmen? Only undergraduates?
- In this problem's context, what does “dropout” mean? If a student quits a degree and enrolls in another, is that a dropout? If a student leaves for a semester and then returns, is that a dropout? If a student is expelled, is that a dropout?
- Can the situations described as “dropout” be identified in the databases available? How?

“We want to predict course success.” Again, a simple statement, but:

- In this problem's context, what is a “course”? All the courses?
- In this problem's context, what is “course success”? Is it equal for all courses?
- Are “success” and “failure” the only possible outcomes, or would it be more interesting to get something more detailed (e.g., predicting course grades)?
- Is it possible to identify “courses” and “course success” (as defined in the previous questions) in the available databases? How?

Best Practices in Dropout Prediction

Furthermore, the selection of the event(s) to predict has profound implications for the initiative, since it implicitly defines the granularity of the context in which that target event occurs, and this granularity may severely impact the core areas of the predictive effort (see Table 1).

Table 1. Effect of the granularity of the predicted event context

Granularity	Example event (context)	Type of Features	Available Training Data	Model Fragility	Number of Models Needed
Higher	Success (Course)	Detailed	Fewer	Higher	Higher
Lower	Dropout (Institution)	Aggregated	More	Lower	Lower

Granularity affects model development as follows. First, in general, more training data allows the generation of more accurate models (Abdulraheem, Haimovich, Ham, & Vazquez, 2015). Second, detailed features are more meaningful, easier to interpret, and retain more information than aggregated features. Third, fragile models are easily invalidated by subtle changes in context (see the last point of this section for more about the most fundamental assumption: immutability of context). Fourth, the number of models needed to predict the event directly influences the complexity of developing, deploying, and maintaining the initiative.

To be more specific, consider the following scenario as an example: an online university in existence for 10 years offers 25 degrees, with each degree offering an average of 25 courses. Each course has on average 20 enrolled students, and each course requires, on average, the completion of 10 different evaluable individual tasks.

For the sake of simplicity, let us suppose that all the predictive power lies in the grades each student gets on the evaluable tasks. Table 2 shows the impact of selecting different target events to predict, with their corresponding context granularity.

Table 2. Impact of selecting different predicted events with different context granularities

Event to Predict	Number of Required Models	Number of Historic Task Results Available for Training each Model	Example Feature	Example Invalidating Change	Probability of Invalidating Change
Course Failure	625 (one per course)	2,000	Grade obtained in task “n”	<ul style="list-style-type: none"> Task “n” is modified. The order of the tasks is changed. 	HIGH
Degree Dropout	25 (one per degree)	50,000	Average of grades obtained in all tasks from course “x”	<ul style="list-style-type: none"> The content of course “x” is severely changed Course “x” disappears, and two new courses are created 	MEDIUM
University Dropout	1 (one for the university)	1,250,000	Average percentile of average grades obtained in all tasks of all enrolled courses during 1 st quarter	<ul style="list-style-type: none"> A university-wide methodological change eliminates individual evaluable tasks in favour of group-based full-semester projects 	LOW

Best Practices in Dropout Prediction**What is the Desired System Output?**

Predictive models can be chosen and trained to generate different types of answers about the event(s). The following questions should be answered to clearly define the type of output to be obtained from the system:

- Are the desired predictions categorical or numerical? For example, “Predicted grade = A+” is a categorical prediction, while “Predicted semesters-to-graduate = 10” is a numerical prediction.
- Is the event binary? That is, can the answers be expressed as yes/no? For example, Dropout = yes or Dropout = no.
- Are more than two possible answers needed? For example, Dropout = no, Dropout = yes, early or Dropout = yes, late.
- Should the answers be expressed in terms of probability? For example, Dropout probability = 47%.

In the authors’ experience, expressing the predictions in terms of probability is often the best option, due to the following reasons:

1. Predictions are naturally uncertain and, in most cases, the models themselves use probabilities in their internal logic, even if the results are expressed in categorical terms. Therefore, expressing the predictions in terms of probability often suits the inherent uncertainty of predictions better. Furthermore, in terms of user-perception, if a prediction is “NO” and the reality turns out to be “YES”, the users tend to consider this a much bigger error than when the prediction says, e.g., “DROPOUT – 45% probability”, and the student finally drops out.
2. The most important reason for using probability-based answers is that they allow prioritization. Behind most dropout-prediction efforts lies the problem of assigning scarce counselling or tutoring resources in the best possible way. Probability-based dropout predictions allow the prioritization of students by “level of risk”, while a simple “yes/no” prediction generates a set of indistinguishable students to be addressed, with no further information to decide who needs to be supported first.

Who will get the Predictive Results?

Focusing solely on the predictive models, without thinking about how they will be used, in which user-contexts, and by whom, may result in a short-sighted approach that neglects important usability and organizational aspects of the solution.

Depending on the type of information displayed along with the prediction, and on the relationship between the students and the users of the predictive system, legal implications related to personal-data privacy may arise.

Another important issue to consider in advance is the internal organizational model to support the retention effort and, hence, the primary addressees of the predictions. Is it a centralized or decentralized model? Is it single- or multi-tiered? The type and number of models to develop, as well as the interfaces and channel(s) to create for delivering the prediction, may be heavily impacted by the answers to these questions.

Best Practices in Dropout Prediction**What do we want to Achieve by Knowing in Advance that the Event is Likely to Occur?**

Predictive models aim raise an alert regarding a certain event before it happens. There is an obvious underlying assumption here that is easily overlooked but may need to be explicitly stated and validated: a certain event can be prevented if its probable occurrence is known in advance. While this assumption is at the core of nearly every predictive modelling effort, it is worth asking this explicit question: can something be done to prevent the occurrence of the predicted event? What? Is this expectation just a hypothesis, wishful thinking, or is there evidence? What is the expected success rate, i.e., in how many correct predictions is the predicted and unwanted event expected to be avoided? Is that enough? The answers to these questions are the key to setting realistic expectations and may be important enough to challenge the entire predictive effort.

Why are we Doing this? what are we Trying to Achieve by Preventing the Predicted Event?

In considering the answer to the second question in the heading above, it is necessary to define, in a sentence and institutional terms, the outcome/s desired. These would include:

- Increasing retention rates.
- Shortening the time to graduate.
- Increasing student satisfaction.
- Increasing course success rate.
- Increasing income.

Once the expected outcomes have been defined, they should be quantified. What exactly should be achieved to consider this initiative successful? Which institutional metrics should be affected and to what extent? What are their values now? What are the minimum target values in a successful scenario? What is the timeframe to achieve them?

When Are Predictions Needed?

Time is a critical dimension in predictive modelling. The amount of available data to be used, the accuracy of the model, the set of possible measures to be taken, and their expiry date depend on the time at which the prediction is made (see Table 3).

Table 3. Impact of time in predictive modelling

Prediction Timing	Available Data	Model Performance	Window of Opportunity to Intervene
Early	Scarce	Limited	Large
Late	Abundant	Optimum	Small

Best Practices in Dropout Prediction

In the authors' opinion, the best way to tackle the dilemma of deciding whether to provide early or late predictions is to provide both. The recommended approach is to provide periodic and updated predictions from the beginning of the academic period to its end. The key decision in this case is the frequency of the predictions, as it directly affects the number of models to generate: a specific date-specialized model has to be generated for each periodic milestone. This will result in more - and more complex - models but will inform the decision makers both as early as possible and as accurately as possible, taking into account all the new information available at each period (Ortigosa et al., 2019). In the "Feature Engineering" section below, recommendations are presented on how to represent the evolution of the features for the models over time.

Does the Problem Context Comply with the "Fundamental Assumption" of Predictive Modelling?

It is very easy to forget that predictive modelling is completely based on a rigid assumption: that the events that correlated in the past will correlate in the future. This assumption is sometimes treated as if it were an axiom and therefore implicitly overlooked. However, in the authors' experience, this is far from being an axiom and, in fact, there may be several reasons that render that core assumption false in most realistic scenarios. The most common one is the inherent mutability of the reality that surrounds the events to correlate, that is, their context. A changing context can easily spoil a predictive model that, theoretically (and after being trained and evaluated under laboratory conditions), was considered valid. The model was perfectly valid in the context in which it was trained, but it is no longer valid once it is applied in the real world, where the context has changed.

The following is an example taken from the authors' experience: when training a predictive model to serve as an early warning system for preventing first-year student dropout, several features related to the early **completion** of the user profile appeared as very relevant (that is, they had a high predictive power). After being trained with LMS data from the previous five years, the models detected a clear positive correlation among "poor profiles" and student dropout, and a negative correlation among "rich profiles" and student persistence. The students who devoted time to completing their profile (wrote a self-description, filled-in their **personal details**, uploaded a photograph, etc.) were less likely to drop out than those who left the profile empty or provided little information. They were good predictors and made sense to everyone. But then the context changed and rendered all the profile-related features (and hence, the models) completely useless: specific training on the LMS was included as part of a new initial pre-course for freshmen, and one of the graded tasks was precisely to fill in the user profile in the LMS. Completing the student profile, which had been an optional task for several years, turned into compulsory homework. The context in which the correlated events happened in the past changed dramatically and, therefore, the assumption made by predictive modelling was no longer valid.

The conclusion is clear: the fundamental assumptions of predictive modelling must be explicitly challenged before model training and before putting the trained models into production, and there should be a periodical auditing task. The most common scenario will probably be this one: the context changed from the time that training data began, and the context will change during the period of time in which trained models will be used. The former must be considered during feature-engineering processes, in the model training stage; and the latter should be detected once models are put into production to periodically adapt them accordingly.

Best Practices in Dropout Prediction

This constant adaptation to changing contexts is a source of complexity during model training, and is probably the biggest threat to otherwise perfectly trained models, and the major cause of re-work in production scenarios. Current educational contexts are evolving very fast and under increasing competitive pressure, so changes may come from very different aspects of educational processes: changes in methodology, pedagogical changes, administrative changes, organizational changes, technical changes in the tools used by the students, etc.

Development: Feature Engineering

A large part of the effort in any machine learning initiative is invested in the feature-engineering stage. There is a good reason for this: feature selection and quality are essential for model performance. In this section, the authors review the type of features that have proven to be useful (with some predictive power) and are commonly used in educational environments, providing practical recommendations about feature modelling.

In many cases, academic grades and attendance have been considered to build predictive models (Knowles, 2015). Information about the student background, his interactions within the LMS and the results obtained in continuous assessment are used in (Howard, Meehan, & Parnell, 2018). In (Najdi & Er-Raha, 2016), data about the students' age, gender, distance from home, and pre-enrolment and first-term performances are used. In (Lacave, Molina, & Cruz-Lemus, 2018), both academic and social data are combined with predictive purposes. Most of these works make use of information generated while the students are taking the courses, which may not be available for earlier predictions.

In other cases, the models do not include such information, but basic administrative data along with additional data to improve the quality of prediction (e.g., periodic national exams for primary school students, or household surveys and census data for older ones) (Adelman, Haimovich, Ham, & Vazquez, 2018). In (Conijn, Snijders, Kleingeld, & Matzat, 2017), the authors stress the need to consider other sources of data beyond the LMS records to improve early predictions, such as personality features (Shum & Crick, 2012), learning styles, or motivation (Tempelaar, Rienties, & Giesbergs, 2015). They analyzed data from 17 blended courses; the inconsistencies found on the results obtained made it difficult to draw general conclusions about the online behavior of potential students at risk (Conijn et al., 2017).

In the work done by the authors of this chapter, information from different sources is combined, including all the data available in administrative databases from the very beginning, along with all the interactions registered within the LMS (Ortigosa et al., 2019).

A Classification of Features Based on their Domain

One of the main questions to be addressed is which attributes or features are used to generate the models. Table 4 shows several types of features from different domains to be considered.

Extracting Relevant Features from Basic Events

When generating event-based features, it is particularly important to try to measure all the potentially relevant aspects of that event. For example, for a single and simple event such as the completion of an online test, how many base features can be derived from its recording? To answer this, the following questions should be asked:

Best Practices in Dropout Prediction*Table 4. Types of features classified by domain*

Domain	Description	Examples
Socio-demographic	Personal student characteristics and context	<ul style="list-style-type: none"> • Age • Gender • Ethnicity • Marital status • Employment • Locations
Academic	Quantitative & qualitative descriptions of a general academic context	<ul style="list-style-type: none"> • Educational level • University access mode • Grades obtained in previous studies • Existence of previous “gaps” or “sabbatical” periods • Current enrolled study • Progress level in current study • Performance metrics in previous academic periods (i.e., GPA, performance rate, presentation rate, etc.) • Number of courses/credits enrolled in current study • Date of enrolment (early/late?) • Performance metrics in current enrolled courses by type of content (tests, tasks, exams, etc.)
Economic	Student’s economic context in general and to his studies in particular	<ul style="list-style-type: none"> • Economic level (household income) • Grant/scholarship awarded? • Special discounts applied? • Loan? • Payment type (single, fractioned, etc.) • Pending payments?
Behavioral	Student’s behavior registered in the institution systems	<ul style="list-style-type: none"> • Frequency of activity in LMS • Daily activity patterns in LMS • Weekly activity patterns in LMS • Workday/holiday activity ratio in LMS • Delay/advance in due dates in LMS time-limited events or homework
Interaction with other stakeholders	Student’s interactions with peers, faculty, and administrative staff	<ul style="list-style-type: none"> • Frequency of personal interactions with peers (messaging, online chats, etc.) • Frequency of personal interactions with teachers (messaging, videoconferences, etc.) • Frequency of personal interactions with staff (helpdesk tickets, meetings with a counsellor, etc.) • Frequency of participation in broadcast-channels (i.e., forums)
Interaction with LMS & content	Student’s interaction with teaching content, usually in the LMS	<ul style="list-style-type: none"> • Type of access device • Level of completeness of LMS user profile • Number of accesses to the LMS • Number and type of resources accessed • Number of completed tests & number of attempts • Number of delivered tasks

- Which are the relevant properties of the event from an academic point of view? In the case of test completion, apart from the test score, there are many other interesting properties, such as: When was the test completed, related to its due date? How long did the student take to complete it?
- Is there a meaningful context that can be considered as a reference for that event? How do the properties of the event compare with that context? In the case of an online test in a course, an interesting context is composed of the rest of the students enrolled in the same course. Relating the properties of a student’s event to the same event from his peers contextualizes that event in a meaningful and informative way.

Best Practices in Dropout Prediction

Table 5 below shows some examples of base features derived from the recording of this event.

Table 5. Features derived from a simple event (test completion)

Feature	Description	Example Value
Timing of the event	If the test had a due date, what was the difference between the due date and the date in which the student completed it?	-2 days
Timing of the event compared to peers	How does the date compare to their peers' one?	25 th percentile
Quality of the event	What score did the student get?	90%
Quality compared to peers	How does the score relate to their peers' one?	85 th percentile
Duration of the event	How long did it take the student to answer the test?	720 secs.
Duration of the event related to peers	How long did it take to their peers to complete the test? How does the student compare to them?	20 th percentile

By putting the event into context, the amount of information available for the model increases. The direct interpretation of the example included in Table 5 is the following: The student completed the test 2 days before the due date; only 25% of his peers completed it earlier. He obtained a score of 90%, which is better than 85% of his peers. It took him 720 seconds and only 20% of his peers spent less time completing it. In essence, the model is being informed by the fact that, in relative terms, the student completed the test early, correctly and quickly. This information cannot be obtained if the context is taken out of the equation.

Aggregating Basic Features

Most of the events that constitute the basic features in educational contexts usually occur within the courses; that is, at a detailed level. Some authors recommend the use of fine-grained attributes to improve predictions (Azcona & Casey, 2015). However, in higher-level models the direct use of low-level features is impractical, mainly for two reasons:

The number of features grows too quickly. For example, imagine a model to predict dropout at degree level. Focusing on the “test completion” event and the “score” property, if the score of each test in every course is translated into one feature (e.g. score-of-test-1-course-A, score-of-test-2-course-A, and so on), and based on the sample scenario described in the previous section, one would have 25 courses x 10 tests per course = 250 features, just to describe the scores of every possible test in the target degree. When multiplying this by the number of other possible events and by the number of possible properties per event, one easily ends up with training sets containing tens of thousands of features, which, in most cases, supposes an infeasible scenario.

Base, detailed features are usually too weak. As explained in the previous section, the base, detailed features (not its number) is too dependent on the rigid assumption of the immutability of very specific contexts, and for that reason, their validity over time is fragile. For example, in the case described, any change in the number or nature of the tests in any course of the degree will likely require the complete reconstruction of the model. This seems too rigid, and the violation of the “most fundamental assumption” revealed previously is likely to be guaranteed sooner rather than later (if it has not already happened since the training data were obtained).

Best Practices in Dropout Prediction

Therefore, the usual way of proceeding is to aggregate the base, detailed features into higher-level features, and get rid of the former. This is usually done through aggregation and calculation of common descriptive statistics: count, maximum value, minimum value, average value, median value, and standard deviation (Ortigosa et al., 2019). Of course, this operation is a trade-off: a manageable number of features and greater robustness to changes in low-level contexts are obtained, but at the expense of losing some detailed information in the process. An example of processing base, low-level features and generating aggregated ones is shown in Table 6.

Table 6. A simple example of base and aggregated features for one event (test completion)

Course	Test	Result (0-100)	Count	Minimum	Maximum	Average	Median	Standard Deviation
A	A1	70	7	70	100	85	85	10.8
	A2	75						
	A3	85						
	A4	80						
	A5	90						
	A6	95						
	A7	100						
B	B1	50	5	25	50	36	35	9.6
	B2	40						
	B3	30						
	B4	35						
	B5	25						
	B6	-						
C	C1	60	6	45	60	50	47.5	6.3
	C2	55						
	C3	45						
	C4	50						
	C5	45						
	C6	45						
D	D1	90	11	10	90	55	55	24.2
	D2	45						
	D3	70						
	D4	40						
	D5	65						
	D6	55						
	D7	50						
	D8	80						
	D9	25						
	D10	75						
	D11	10						

Best Practices in Dropout Prediction

Note the following:

- The number of detailed base features is 30, and the number of aggregated features is 24. The ratio in this simple example is not that impressive but, in real contexts, the cardinality of the base feature set is much higher.
- The number of aggregated features does not depend on the number of detailed base features, but on the number of aggregating entities (in this example, the number of courses). If each course had 50 tests, the number of detailed features would increase to $50 \times 40 = 200$, but the number of aggregated features would remain constant in 24.
- The aggregated features do a fairly decent job describing the underlying situation to the models:
 - In general terms, in course A the student is performing consistently well in the tests
 - In general terms, in course B, the student is performing consistently poorly in the tests
 - In general terms, in course C, the student's performance in the tests is consistently average
 - In general terms, in course D, the student's performance is chaotic, with a mixed bag of high and low scores
- Inevitably, some information is lost by aggregating:
 - Thanks to the “count” aggregated feature, the model knows that in course B the student has not completed one of the tests, but it does not know which one, and it might be relevant.
 - Thanks to the “minimum” and “maximum” aggregated features, the model knows that in course D the student has scored 90 and 10 at least once, but it does not know in which tests, and this might be relevant.

In the authors' experience, models often tend to prefer “extreme” aggregated features (minimum or maximum) to develop their internal logic over central tendency aggregated features (average, median) or dispersion ones (standard deviation), and they use it to test threshold values (e.g., testing if the minimum score in tests is over X, or if the maximum score in tests is below Y).

Capturing the Time Dimension

As mentioned previously, the authors recommended a predictive strategy that produces periodical, updated predictions based on the ever-growing set of available data as the academic period unfolds. While more complex, this strategy has several benefits already explained. Additionally, it allows encoding how the “time” dimension affects the features: at the end of each period, the models can be provided, not only with a snapshot of the values of every feature in the current milestone, but also with information on how each feature has evolved over time. Yet, how can the change over time be measured into a single, numeric feature to be interpreted by models, and how can it be encoded? Table 7 summarizes this.

While short-term tendency and long-term global values are straightforward, the method to calculate a feature that captures the long-term tendency is a little more elaborate. For that, the regression line of all the ordered values from each known period is calculated and the slope is extracted. While there are cases in which the regression line fits the data points poorly due to highly variable data, in the authors' experience, it is still useful and good enough to represent the long-term tendency in most cases. In the example shown, the slope is the general rate of change in the number of logins over the months and captures two aspects simultaneously:

Best Practices in Dropout Prediction*Table 7. Time-related features*

Time	Description	Calculation
Short term: tendency	What has happened since last time?	The difference from the previous period
Long term: tendency	Since the start of the academic period, what is the general tendency (increasing, decreasing or stationary)?	The slope of the regression line
Long term: global	Since the start of the academic period, which is the current global value of the feature?	Accumulated value

- The **direction** of the global tendency:
 - Positive slope = “increasing” global tendency
 - Negative slope = “decreasing” global tendency
 - Zero or near-zero slope value = “stationary” or “chaotic” global tendency
- The **frequency** of the global tendency:
 - Higher absolute values = bigger changes

For example, suppose one wants to generate dropout predictions throughout a semester, with a monthly periodicity, and one of the features to feed the models with is the number of LMS logins in each period. Table 8 shows a possible set of values for one student.

Table 8. Simple recording of an event

Month	1	2	3	4	5	6
Number of logins in this month	80	35	54	37	29	18

In this example, all the model knows is, for example, that the student logged in 54 times in month 3. While this feature may be relevant (and, in the authors’ experience, it is), much more information can be provided to the models when considering the time dimension and enriching the “current snapshot” view by capturing the **behavior** of that feature along the time, as shown in Table 9.

Table 9. Time-aware enriched event data

Month	1	2	3	4	5	6
Number of logins in this month	80	35	54	37	29	18
Accumulated value	80	115	169	206	235	253
Difference with previous month	0	-45	19	-17	-8	-11
Short term tendency	N/A	Decrease	Increase	Decrease	Decrease	Decrease
Slope of regression line (see Fig. 1 for plots)	0	-45	-13	-11	-10	-11.7
Long term tendency	N/A	Decrease	Decrease	Decrease	Decrease	Decrease

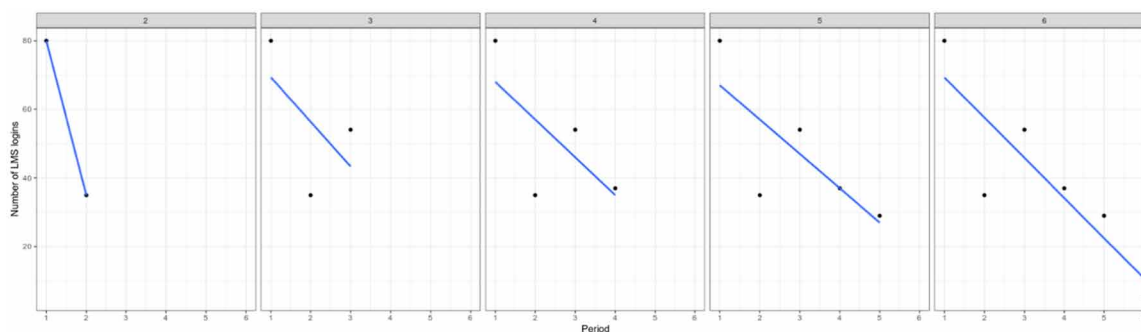
Best Practices in Dropout Prediction

What the model now knows when using this enriched version can be expressed like this:

The student logged in 54 times in month 3, for 169 logins since the beginning of the course. While the general tendency is still a decrease in the number of logins each month, this month's value contradicts that general tendency with an increase of 19 more logins than in the previous month.

This is, obviously, a much more informative and complete version of the situation about a single metric. Figure 1 shows the plots of linear regression lines for the “number of logins in last month” variable from the previous table, representing the “decreasing” global tendency calculated every month.

Figure 1. Linear regression lines for “number of logins in last month”



Pre and Post Evaluation: Will it Work? did it Work?

In this section, the authors present some ideas related to model training and evaluation, during both model development and effective system usage.

Model Interpretability is Fundamental

In the authors' experience, general model interpretability and specific case-by-case scoring interpretability are essential requirements for the sake of user adoption and the capability of generating useful insights. This restricts the models eligible for predictions to white-box ones, whose internal logic is both visible and understandable in terms of the problem's domain. In the authors' experience, the benefits derived from an open and interpretable model logic outweigh, by far, the limited loss of model performance due to the use of white-box models (Ortigosa et al., 2019). While not suitable for production-use, black-box models are still useful, due to their usual better results, in the training stage, as a reference upper bound of the obtainable performance when tuning the white-box models.

If Possible, Mimic the Production Scenario when Estimating Performance

As in any other domain, in dropout-prediction modelling, parameter tuning and model evaluation requires the prior estimation of model performance through the usual techniques, K-fold cross-validation and Holdout method being the most common ones. In the case of the Holdout method, the authors recom-

Best Practices in Dropout Prediction

mend selecting the data from the most recent academic period available in the training set as the holdout set, instead of choosing a random set with samples taken from different academic periods. The reason for this is that this setting better resembles the real scenario of model application and captures one of the basic assumptions: that, by using previous academic periods' data, it is possible to predict new academic period's results. In the authors' experience, the use of completely random holdout sets is more prone to optimistic model performance estimations.

You will Probably Have to Deal with an Unbalanced Training Dataset

Dropout is usually a highly unbalanced classification problem. That is, the prevalence of one of the classes (usually, “dropout”) is proportionately much smaller than the other (usually, “persistence”). Specific techniques will likely be needed to address this unbalanced nature of the problem. In the authors' experience, the use of penalized classification algorithms (in which the data scientist can adjust the “cost” of errors such as false positives or false negatives) gives better results and is easier to tune than the use of under or oversampling techniques. These observations are in line with some research reports, which prove that modified support vector machine and cost-sensitive classifiers outperform oversampling and under sampling methods (Ganganwar, 2012).

Focus on Sensitivity and Specificity, and Consider the Different Institutional Costs of False Positives and False Negatives

Due to the highly unbalanced nature of the dropout phenomenon, “accuracy” is not a reliable performance measure. In an institution with a 10% dropout prevalence, a trivial model that predicts that every student will not be a dropout yields a 90% precision, and is, obviously, completely useless. “Sensitivity” and “Specificity” are usually the two performance measures to focus on. As in most cases, in dropout prediction, there is usually an inverse relationship between these two magnitudes and tuning the model to increase one usually implies a decrease in the other, so a trade-off must be found. Receiver operating characteristic (ROC) curves are a well-known technique to represent this dependency graphically and are especially useful in this task. Nevertheless, keep in mind that the typical optimum threshold cut-off value for ROC curves (the value that maximizes the sum “specificity + sensitivity”) may not be the best option in dropout prediction problems, due to the different “institutional cost” of false positives and false negatives. Usually, a false positive (labelling a student as “at-risk”, when he is not) is judged to be a less-critical error than a false negative (labelling a student as “not-at-risk”, when in fact he was), and a certain tolerance of more false positives for the sake of reducing the false negative errors is usually accepted.

The authors recommend the use of example scenarios to communicate the estimated model performance to business users and the avoidance of technical jargon. Instead of informing about sensitivity or specificity, they suggest describing performance in business terms and using absolute numbers. For example, “with the current model performance estimations, in a degree with X students and a dropout rate of Y%, N students would be correctly labelled as at-risk (targeting a P% of the total number of at-risk students) at the cost of incorrectly labelling M, not at-risk students as at-risk. This gives a total of T students to address retention actions.”

Best Practices in Dropout Prediction**After Model Generation, Stop and Estimate Return on Investment (ROI)**

Once the models are in place and their probable performances have been estimated, it is advisable to define different possible scenarios before proceeding. The economic aspect is one of the many ways dropout reduction benefits an institution, and in this regard, different ROI scenarios can be estimated based on the values of the following variables, before proceeding with the development of the rest of the system:

- The average annual income produced by an enrolled student
- The current dropout prevalence
- The estimated sensitivity and specificity of the model/s
- The estimated annual cost of the retention actions for one student
- The expected success rate of retention actions
- The expected number of students that will enrol in the next year
- The estimated annual cost of the predictive models

The insights gained during this exercise may produce relevant changes in the dropout prevention strategy and the development of the rest of the project.

Demonstrate Value: Evaluate and Quantify the Real Performance in Production

Once the system is implemented and generating predictions that lead to prioritization and execution of retention actions, a periodic evaluation (usually at the end of each academic cycle) is needed to calculate the real system performance. According to (Herodotou et al., 2019), when adopting a predictive analytics system, it is advisable to provide evidence about whether suggested interventions work through robust evaluation. Among other benefits, it will help to increase system credibility and decrease resistance to system adoption. The main purpose of this evaluation is to answer these two essential questions:

- How well did the models predict dropout?
- Did the retention actions have any effect?

Answering these two questions is tricky, and some factors need to be considered:

- The scientific method for demonstrating whether the retention actions had any effect is to compare the dropout prevalence in an intervention group of at-risk students to a control group, calculating the difference between them and its statistical significance. However, choosing that control group is no easy task: for the comparison to be valid, the population of the control group must be carefully chosen to be similar to that of the intervention group. To achieve this, the authors of this chapter clustered the students based on their academic contexts and dropout-risk profiles, and then identified the individuals in each cluster that, although at-risk, were not recipients of retention actions due to the inherent limitation of resources, using them as the control group in each cluster. A positive impact (a decrease in dropout rates) was observed in all clusters, but that impact greatly varied from one cluster to another: from a decrease of 0.7 percentage points in the cluster where retention actions were less effective, to a decrease of 7.6 percentage points in the cluster where

Best Practices in Dropout Prediction

they were more effective, suggesting that certain types of students, or students in certain situations, may be more likely to be receptive retention actions than others.

- Oddly enough, a well-functioning system, with good predictions and well-crafted and effective retention actions, will cause a decrease in the perceived performance of the predictive models, by turning predicted dropouts into persistent students due to the success of the retention actions. For this reason, the calculated real sensitivity of the models should be considered a lower bound, and the false negative rate an upper bound.

For models that do not produce yes/no answers but probabilities (as recommended in this chapter), metrics such as sensitivity and specificity can be difficult to calculate and share with stakeholders. In the authors' experience, ROC curves are difficult to understand and interpret for business users. A better option may be to group students in intervals based on the predicted risk, calculate the real prevalence of the predicted event on each group, and plot the results. In an ideal scenario with perfect predictions, effective retention actions, and a control group, two plots (one for the intervention students and one for the control group), representing the predicted risk on the x-axis and the real prevalence on the y-axis, can be generated. In the plot corresponding to the intervention students, a less-than 45 degree straight line should appear, meaning that the originally predicted risk was perfect, but due to the effective retention actions taken, the final occurrence of the event was reduced by a certain amount. In the plot corresponding to the control group, a perfect 45-degree straight line should appear, meaning that the dropout occurred exactly in the predicted proportions without retention actions.

FUTURE DIRECTIONS

The authors have started working on four lines of further research, all of them oriented towards improving dropout prediction and prevention. The first focuses on analyzing the effects of different types of retention actions on effective dropout risk reduction. The second one consists of analyzing the potential predictive power of sentiment analysis inside e-learning tools and student's social networks for dropout risk. The authors already have experience in sentiment analysis in e-learning and social network contexts (Ortigosa, Martin, & Carro, 2014; Rodriguez, Ortigosa, & Carro, 2014). The third one deals with analyzing the potential predictive power of personality traits for dropout risk. From a psychological point of view, there is evidence that personality differences would predict differences in student performance (MacCann et al., 2020). However, it is not clear yet what should be measured (van der Linden et al. 2017), how the correlations found in psychological contexts can be used in a predictive system, or the ethical implications of such a prediction. The authors, who have some experience inferring the user personality by mining social interactions (Ortigosa, Carro, & Quiroga, 2014) intend to explore this further. Finally, the last one focuses on generating automated, event-triggered and agent-based retention actions, using chatbot technology.

CONCLUSION

Predictive models are the core of dropout prevention systems, and their importance is indisputable. While essential, good predictive models are not enough for a successful deployment and exploitation

Best Practices in Dropout Prediction

of such systems in large-scale production scenarios, and many other additional aspects should be considered: expected results, user context and organization, available information, institutional alignment, user confidence in the system, etc.

Dropout prevention should not be a delimited project, but a sustained effort in which at least three key tasks should be addressed recurrently. First, periodic auditing should be done to detect contextual changes and adapt the models accordingly. Second, the obtained results after each academic cycle should be evaluated, analyzed and shared with all stakeholders, and used to build users' confidence as well as to improve the models. Third, user feedback and concerns should be addressed as soon as they arise. It takes a lot of effort to gain the users' trust and, therefore, it is worth doing everything to avoid losing it.

As stated previously, the aim of this chapter is to address the lack of industry-oriented analyses of the current approaches to dropout prevention. To this end, it aims to contribute to filling this gap by presenting some common challenges from an empirical and production-oriented perspective. It describes some of the lessons learned when developing and implementing a dropout prevention system within a higher education context. It highlights some common pitfalls and recommends best practices for building, training, and evaluating predictive models, according to the authors' experience (Ortigosa et al., 2019). It is hoped that data scientists, project managers and users will find the reflections and recommendations in this chapter useful for the design, development, deployment, and use of early warning systems in general and dropout prediction systems in particular.

ACKNOWLEDGMENT

We want to express our sincere thanks to Josefina Alcolea Jiménez-Clover for reviewing this text.

This research was partially supported by the Regional Government of Madrid [e-Madrid project P2018/TCS-4307].

REFERENCES

- Abdulraheem, A., Abdullah Arshah, R., & Qin, H. (2015). Evaluating the effect of dataset size on predictive model using supervised learning technique. *International Journal of Software Engineering & Computer Sciences*, 1, 75–84. doi:10.15282/ijsecs.1.2015.6.0006
- Adelman, M., Haimovich, F., Ham, A., & Vazquez, E. (2018). Predicting school dropout with administrative data: New evidence from Guatemala and Honduras. *Education Economics*, 26(4), 356–372. doi:10.1080/09645292.2018.1433127
- Alyahyan, E., & Düşteğör, D. (2020). Predicting academic success in higher education: Literature review and best practice. *International Journal of Educational Technology in Higher Education*, 17(1), 3. doi:10.118641239-020-0177-7
- Azcona, D., & Casey, K. (2015). Micro-analytics for student performance prediction leveraging fine-grained learning analytics to predict performance. *International Journal of Computer Science and Software Engineering*, 4(8), 218–223.

Best Practices in Dropout Prediction

Badr, G., Algobail, A., Almutairi, H., & Almutery, M. (2016). Predicting students' performance in university courses: A case study and tool in KSU Mathematics Department. *Procedia Computer Science*, 82, 80–89. doi:10.1016/j.procs.2016.04.012

Baker, R., & Yacef, K. (2009). The state of educational data mining in 2009: A review and future visions. *J. Educ. Data Mining*, 1(1), 3–17.

Berens, J., Schneider, K., Görtz, S., Oster, S., & Burghoff, J. (2019). Early detection of students at risk - Predicting student dropouts using administrative student data from German Universities and machine learning methods. *Journal of Educational Data Mining*, 11(3), 1–41. doi:10.5281/zenodo.3594771

Bienkowski, M., Feng, M., & Means, B. (2012). *Enhancing teaching and learning through educational data mining and learning analytics: An issue brief*. U.S. Department of Education, Office of Educational Technology. Retrieved from <https://www.ed.gov/technology>

Campbell, J., De Blois, P., & Oblinger, D. (2007). Academic analytics: A new tool for a new era. *Educause Review*, 42(4), 40–57. Retrieved from <http://www.educause.edu/ero/article/academic-analytics-new-tool-new-era>

Conijn, R., Snijders, C., Kleingeld, A., & Matzat, U. (2017). Predicting Student Performance from LMS Data: A Comparison of 17 Blended Courses Using Moodle LMS. *IEEE Transactions on Learning Technologies*, 10(1), 17–29. doi:10.1109/TLT.2016.2616312

Dalipi, F., Imran, A., & Kastrati, Z. (2018). MOOC dropout prediction using machine learning techniques: Review and research challenges. *2018 IEEE Global Engineering Education Conference (EDUCON)*, 1007–1014. 10.1109/EDUCON.2018.8363340

Ferguson, R., Clow, D., Griffiths, D., & Brasher, A. (2019). Moving Forward with Learning Analytics: Expert Views. *Journal of Learning Analytics*, 6(3), 43–59. doi:10.18608/jla.2019.63.8

Freitas, S., Gibson, D., Du Plessis, C., Halloran, P., Williams, E., Ambrose, M., ... Arnab, S. (2015). Foundations of dynamic learning analytics: Using university student data to increase retention. *British Journal of Educational Technology*, 46(6), 1175–1188. doi:10.1111/bjet.12212

Ganganwar, V. (2012). An overview of classification algorithms for imbalanced datasets. *International Journal of Emerging Technology and Advanced Engineering*, 2(4), 42–47.

Helal, S., Li, J., Liu, L., Ebrahimie, E., Dawson, S., Murray, D. J., & Long, Q. (2018). Predicting academic performance by considering student heterogeneity. *Knowledge-Based Systems*, 161(1), 134–146. doi:10.1016/j.knosys.2018.07.042

Herodotou, C., Rienties, B., Verdin, B., & Boroowa, A. (2019). Predictive Learning Analytics 'At Scale': Guidelines to Successful Implementation in Higher Education. *Journal of Learning Analytics*, 6(1), 85–95. doi:10.18608/jla.2019.61.5

Howard, E., Meehan, M., & Parnell, A. (2018). Contrasting Prediction Methods for Early Warning Systems at Undergraduate Level. *The Internet and Higher Education*, 37, 66–75. doi:10.1016/j.iheduc.2018.02.001

Best Practices in Dropout Prediction

Hung, J., Shelton, B., Yang, J., & Du, X. (2019). Improving Predictive modelling for At-Risk Student Identification: A Multistage Approach. *IEEE Transactions on Learning Technologies*, 12(2), 148–157. doi:10.1109/TLT.2019.2911072

Knowles, J. (2015). Of Needles and Haystacks: Building and Accurate Statewide Dropout Early Warning System in Wisconsin. *J. Educ. Data Mining*, 7(3), 18–67.

Lacave, C., Molina, A., & Cruz-Lemus, J. (2018). Learning Analytics to identify dropout factors of Computer Science studies through Bayesian networks. *Behaviour & Information Technology*, 37(10-11), 993–1007. doi:10.1080/0144929X.2018.1485053

MacCann, C., Jiang, Y., Brown, L., Double, K., Bucich, M., & Minbashian, A. (2020). Emotional intelligence predicts academic performance: A meta-analysis. *Psychological Bulletin*, 146(2), 150–186. doi:10.1037/bul0000219 PMID:31829667

Marquez-Vera, C., Cano, A., Romero, C., Noaman, A., Fardoun, H., & Ventura, S. (2016). Early dropout prediction using data mining: A case study with high school students. *Expert Systems: International Journal of Knowledge Engineering and Neural Networks*, 33(1), 107–124. doi:10.1111/exsy.12135

Najdi, L., & Er-Raha, B. (2016). A Novel Predictive modelling System to Analyze Students at Risk of Academic Failure. *International Journal of Computers and Applications*, 156(6), 25–30. doi:10.5120/ijca2016912482

Nunn, S., Avella, J. T., Kanai, T., & Kebritchi, M. (2016). Learning analytics methods, benefits, and challenges in higher education: A systematic literature review. *Online Learning*, 20(2), 13-29.

Olivé, D., Huynh, D., Reynolds, M., Dougiamas, M., & Wiese, D. (2019). A Quest for a One-Size-Fits-All Neural Network: Early Prediction of Students at Risk in Online Courses. *IEEE Transactions on Learning Technologies*, 12(2), 171–183. doi:10.1109/TLT.2019.2911068

Ortigosa, A., Carro, R., Bravo-Agapito, J., Lizcano, D., Alcolea, J., & Blanco, O. (2019). From Lab to Production: Lessons Learnt and Real-Life Challenges of an Early Student-Dropout Prevention System. *IEEE Transactions on Learning Technologies*, 12(2), 264–277. doi:10.1109/TLT.2019.2911608

Ortigosa, A., Carro, R. M., & Quiroga, J. I. (2014). Predicting user personality by mining social interactions in Facebook. *Journal of Computer and System Sciences*, 80(1), 57–71. doi:10.1016/j.jcss.2013.03.008

Ortigosa, A., Martín, J. M., & Carro, R. M. (2014). Sentiment analysis in Facebook and its application to e-learning. *Computers in Human Behavior*, 31, 527–541. doi:10.1016/j.chb.2013.05.024

Polyzou, A., & Karypis, G. (2019). Feature Extraction for Next-Term Prediction of Poor Student Performance. *IEEE Transactions on Learning Technologies*, 12(2), 237–248. doi:10.1109/TLT.2019.2913358

Rodriguez, P., Ortigosa, A., & Carro, R. M. (2014). Detecting and making use of emotions to enhance student motivation in e-learning environments. *Int. Journal of Continuing Engineering Education and Life Long Learning*, 24(2), 168-183.

Romero, C., & Ventura, S. (2010). Educational data mining: A review of the state of the art. *IEEE Transactions on Systems, Man and Cybernetics. Part C, Applications and Reviews*, 40(6), 601–618. doi:10.1109/TSMCC.2010.2053532

Best Practices in Dropout Prediction

- Romero, C., & Ventura, S. (2019). Guest Editorial: Special Issue on Early Prediction and Supporting of Learning Performance. *IEEE Transactions on Learning Technologies*, 12(2), 145-147.
- Sandoval, A., Gonzalez, C., Alarcon, R., Pichara, K., & Montenegro, M. (2018). Centralized student performance prediction in large courses based on low-cost variables in an institutional context. *The Internet and Higher Education*, 37, 76-89. doi:10.1016/j.iheeduc.2018.02.002
- Scheffel, M., Drachler, H., Stoyanov, S., & Specht, M. (2014). Quality indicators for learning analytics. *Journal of Educational Technology & Society*, 17(4), 117-132.
- Shum, S., & Crick, R. (2012). Learning dispositions and transferable competencies: Pedagogy, modelling and learning analytics. In *Proc* (pp. 92-101). LAK. doi:10.1145/2330601.2330629
- Tempelaar, D., Rienties, B., & Giesbers, B. (2015). In search for the most informative data for feedback generation: Learning analytics in a data-rich context. *Computers in Human Behavior*, 47, 157-167. doi:10.1016/j.chb.2014.05.038
- van der Linden, D., Pekaar, K., Bakker, A., Schermer, J., Vernon, P., Dunkel, C., & Petrides, K. (2017). Overlap between the general factor of personality and emotional intelligence: A meta-analysis. *Psychological Bulletin*, 143(1), 36-52. doi:10.1037/bul0000078 PMID:27841449

ADDITIONAL READING

- Baneres, D., Rodriguez, M. E., & Serra, M. (2019). An early feedback prediction system for learners at-risk within a first-year higher education course. *IEEE Transactions on Learning Technologies*, 12(2), 249-263. doi:10.1109/TLT.2019.2912167
- Berens, J., Schneider, K., Görtz, S., Oster, S., & Burghoff, J. (2019). Early detection of students at risk - Predicting student dropouts using administrative student data from German Universities and machine learning methods. *Journal of Educational Data Mining*, 11(3), 1-41. doi:10.5281/zenodo.3594771
- Ferguson, R., Clow, D., Griffiths, D., & Brasher, A. (2019). Moving forward with learning analytics: Expert views. *Journal of Learning Analytics*, 6(3), 43-59. doi:10.18608/jla.2019.63.8
- Herodotou, C., Rienties, B., Verdin, B., & Boroowa, A. (2019). Predictive learning analytics 'at scale': Guidelines to successful implementation in higher education. *Journal of Learning Analytics*, 6(1), 85-95. doi:10.18608/jla.2019.61.5
- Lang, C., Siemens, G., Wise, A., & Gasevic, D. (2017). Handbook of Learning Analytics. <https://www.solaresearch.org/hla-17/>
- Ortigosa, A., Carro, R., Bravo-Agapito, J., Lizcano, D., Alcolea, J., & Blanco, O. (2019). From lab to production: Lessons learnt and real-life challenges of an early student-dropout prevention system. *IEEE Transactions on Learning Technologies*, 12(2), 264-277. doi:10.1109/TLT.2019.2911608
- Romero, C., & Ventura, S. (2013). Data mining in education. *Wiley Interdisciplinary Reviews. Data Mining and Knowledge Discovery*, 3(1), 12-27. doi:10.1002/widm.1075

Best Practices in Dropout Prediction

Sønderlund, A. L., Hughes E., & Smith, J. (2018). The efficacy of learning analytics interventions in higher education: A systematic review. *British J. Edu.Technol.*, 1–25.

KEY TERMS AND DEFINITIONS

Accuracy: The performance metric for classifiers indicating the percentage of correctly classified cases regardless of the class to which they belong.

Holdout Method: The simplest kind of cross-validation, in which the data set is separated into two sets, the training set and the testing set, which are not swapped.

K-Fold Cross-Validation: A statistical method for cross-validation, used to estimate the skill of machine learning models.

Over/Under Sampling Techniques: Techniques used to adjust the class distribution of a data set (i.e., the ratio between the different classes/categories represented), in which new data points are added/removed.

Receiver Operating Characteristic (ROC) Curve: A graphical plot that illustrates the predictive capacity of a binary classifier for distinguishing between classes at various thresholds settings.

Return on Investment (ROI): The profit from an activity for a particular period compared with the amount invested in it.

Sensitivity: The performance metric for binary classifiers indicating the percentage of true positive cases correctly labelled as positive by the system.

Specificity: The performance metric for binary classifiers indicating the percentage of true negative cases correctly labelled as negative by the system.