Received: 27 December 2023 | Revised: 4 May 2024 | Accepted: 9 May 2024

DOI: 10.1002/cae.22767

RESEARCH ARTICLE

WILEY

Predictive modeling of on-time graduation in computing engineering programs: A case study from Northern Chile

Aldo Quelopana[1] | Brian Keith[1] | Ricardo Pizarro[1,2]

[1]Department of Computing & Systems Engineering, Universidad Católica del Norte, Antofagasta, Chile

[2]Departamento de Electrónica in Universidad de Alcalá (UAH), Alcalá de Henares, Madrid, Spain

Correspondence
Brian Keith, Department of Computing & Systems Engineering, Universidad Católica del Norte, Av. Angamos 0610, Antofagasta, Chile.
Email: brian.keith@ucn.cl

Abstract

In the ever-evolving landscape of 21st-century education, this research seeks to understand the challenges of on-time graduation for students in two related computing majors. In particular, we focus on the Universidad Católica del Norte computing engineering programs in Chile, specifically the "Computing and Informatics Civil Engineering" (ICCI) and "Computing and Informatics Execution Engineering" (IECI) programs. We developed a machine-learning-based model using random forests to predict delays in submissions of the final report of graduation projects, the key step in the graduation pipeline of the analyzed students. We had access to a data set comprised of 209 students in the period from 2013 to 2017, out of these students, only 111 completed all their graduation requirements. Thus, we focused on this subset of students for the analysis. Our analyses of results indicate that individual advisors minimally contribute to predicting timely or late submissions, emphasizing the need for a holistic approach. In contrast, the specific major, graduation modality, and time in the program play crucial roles, with GPA emerging as the most influential factor (24.06%). Notably, the "Professional Work" modality exhibits a moderate positive correlation with late submissions, contextualized by students' employment commitments. The study's predictive model offers actionable insights for educators and administrators, identifying at-risk students and advocating for personalized support strategies. This research contributes to the ongoing dialogue on enhancing educational outcomes by integrating data-driven approaches tailored to diverse student profiles.

KEYWORDS

graduation project, predictive model, random forest, variable importance

1 | INTRODUCTION

In a context of profound challenges and transformation, education has been entrusted with departing from the traditional "banking" model of education, where passive students are merely filled with whatever the teachers deposit in them [47]. Instead, it must strive to achieve a balance between instinct, intellect, and emotion, which are responsible for thinking, feeling, and doing. This is the basis of the mission imposed by UNESCO on 21st-century education, which consists of learning to know, learning to do, learning to be, and learning to live together [17]. The Universidad Católica del Norte (UCN) is located in northern Chile, in the cities of Antofagasta

https://doi.org/10.1002/cae.22767

and Coquimbo. It is situated in strategically geographic sectors for the country where industries such as mining [34], astronomy [46], volcanology [16], and aquaculture [7] are developed. The UCN seeks to address the challenges indicated by UNESCO by incorporating these dimensions into its educational approach.

The Department of Systems and Computing Engineering (DISC) offers two undergraduate degrees, which, like the rest of the university's programs, pursue the goals delineated by the UCN. The first program is the "Computing and Informatics Civil Engineering" program (Ingeniería Civil en Computación e Informática in Spanish, abbreviated ICCI),[1] a 6-year program in computer science and engineering that emphasizes a strong background in fundamental scientific fields and management topics. The second program is the "Computing and Informatics Execution Engineering" program (Ingeniería Ejecución en Computación e Informática in Spanish, abbreviated IECI),[2] a 4-year program in computer science and engineering, focusing more on technical aspects of the field and less on general scientific knowledge or management topics.

Despite efforts to enhance students' chances of graduating on time from the university's computing engineering programs, there is a strong need to identify key factors that influence on-time graduation rates to ensure effective targeting of resources and efforts. Previous analysis has shown that the critical period is the thesis phase, spanning from the end of coursework to the submission of the final thesis version.

This present study focuses on developing a machine-learning-based model [39] to predict whether students in the aforementioned programs (ICCI and IECI) will delay their thesis submissions between the end of coursework and the final deadline. It aims not only to identify critical factors that contribute to reducing submission time but also to provide deeper insights into the behavior of students within the Latin American cultural context [40]. By focusing on a specific academic context, we aim to extend the existing literature and provide actionable insights for educators and administrators. In particular, in this study, we aim to address the following research questions:

- First, what are the relative contributions of individual student characteristics (e.g., GPA, program duration), institutional factors (e.g., graduation modality, advisor), and academic program (ICCI vs. IECI) in *predicting the timely submission* of final graduation projects among computing engineering students, as determined by a random forest machine learning model?
- Second, how does the choice of graduation modality (Capstone Project, Research Thesis, Professional Work, or Industry Project) *influence the likelihood of on-time graduation* among computing engineering students, and what insights can be derived from the observed correlations between modality and timely submission to inform program design and student support strategies?

To answer these questions, we employ a data-driven approach, leveraging machine learning techniques to analyze a data set of 209 computing engineering students from a university in northern Chile, spanning data from the years 2013 to 2017. The following sections provide a detailed description of the study context, methodology, and findings.

2 | RELATED WORK

A growing body of research has investigated the application of data mining and machine learning techniques to predict student academic outcomes and identify key factors influencing student success in higher education. However, the use of predictive models and data analysis in the academic environment is still a relatively new field [48]. Higher education institutions can leverage student data to construct statistical and machine-learning models to predict various outcomes. We present an overview of relevant literature in the rest of this section.

For instance, [42] developed a predictive model to identify students at risk of failing, based on several critical variables. Although their findings are noteworthy, the model specifically addresses issues in online learning, which is distinct from the traditional classroom setting under discussion here.

2.1 | Predictive modeling in engineering education

In an in-person context, [1] discuss a study at a Nigerian University that employs educational data mining to predict the academic success of engineering students. This study, focusing on the foundational first 3 years, used a data mining model based on the Konstanz Information Miner—KNIME [9]. Key predictors identified include the GPA of the first 3 years, with the third

[1] https://admision.ucn.cl/carreras/tecnologia-computacion/ingenieria-civil-en-computacion-e-informatica/

[2] https://admision.ucn.cl/carreras/tecnologia-computacion/ingenieria-en-computacion-e-informatica/

year being the most influential. Al-Alawi et al. [4] present a study from a major public university in Oman, exploring factors impacting academic performance among students on academic probation. Using supervised machine learning algorithms, this study underscores the impact of study duration and previous secondary school performance on academic success. Even though these studies are relevant, the insights obtained are specific to the mentioned countries, whose cultures differ from those in Latin America.

Osmanbegović and Suljić [35] explored the efficacy of data mining algorithms, including Naïve Bayes (NB), decision trees, and neural networks, in predicting student performance at the University of Tuzla. Their findings underscore the potential of these techniques to identify students at risk of underperforming, enabling targeted interventions. Similarly, Kaur et al. [28] employed decision trees, NB, and multilayer perceptron networks to predict student performance, demonstrating the utility of these methods in educational contexts.

Focusing on engineering education, Huang and Fang [27] utilized multiple linear regression, multilayer perceptron network, radial basis function (RBF) network models, and support vector machines (SVM) to predict the academic performance of engineering students. Their study highlights the importance of prior academic achievement and performance in foundational courses as predictors of success. In a similar vein, Marbouti et al. [32] used predictive modeling to identify at-risk students in engineering courses, underscoring the potential for early intervention. In a similar vein, Akçapınar et al. [3] explored the use of gradient-boosting trees to predict students' academic performance in a digital learning environment to develop an early-warning system for at-risk students. Their study demonstrates the effectiveness of the random forest model and the NB model when considering online learning behaviors in predictive modeling.

2.2 | Large-scale studies, dropout prediction, and early prediction of student success

Aulck et al. [6] conducted a large-scale study using machine learning to predict student dropout rates at a public university in the United States. Their work employed regularized logistic regression, k-nearest neighbors, and random forest models to identify key predictors in drop-out rates. In particular, this study highlights the importance of grades in drop-out rates. Furthermore, this study demonstrates the scalability of machine-learning approaches in educational contexts.

Helal et al. [24] proposed a machine learning framework for predicting at-risk students in higher education, incorporating data from various sources such as student demographics, academic performance, and learning management system interactions. Their approach achieved high accuracy and provided insights into the most influential features for prediction. Similarly, Berens et al. [8] focused on the early prediction of student success by employing a range of machine learning algorithms, including random forests and an ensemble approach with AdaBoost, to predict academic achievement in introductory programming courses. Their findings underscore the predictive power of early assignment scores and highlight the potential for timely interventions.

2.3 | Predictive modeling of graduation times and our study

An important work at a large American research university analyzed data from 160,933 students, exploring the efficacy of gradient-boosted logistic models in predicting graduation times, a methodological innovation compared to traditional logistic models [2]. This research draws on Tinto's Theory of Drop Out [45], which suggests that a student's decision to continue or discontinue their education is influenced by their educational and institutional commitments, both of which are dynamic and affected by various factors such as academic performance and social integration. The study employed a discrete-time hazard modeling framework, apt for handling time-to-event data where events are discrete and concurrent across individuals. Notably, the study found that the gradient-boosted model, particularly the xgboost algorithm, was superior in predicting the graduation semester, especially for female and minority students, highlighting enrollment factors and cumulative grades as key predictors. This underscores the importance of academic and social integration in influencing graduation timelines, aligning with Tinto's theoretical framework, and demonstrates the advantage of advanced statistical techniques in educational research. However, the study is broader than what is needed to analyze specifically in the proposed work.

Our study builds upon this foundation, applying machine learning techniques to predict on-time graduation in computing engineering programs. Moreover, we note that most of these previous studies have not focused on the issue of on-time graduation, but rather on general student performance and drop-out rates. By focusing on a specific academic context and leveraging the unique strengths of the random forest algorithm, we aim to extend the existing literature and provide actionable insights for educators and administrators.

3 | CONTEXT

After completing their coursework, students select one of the various modalities to graduate. Subsequently, a supervising advisor is assigned from the pool of available professors within the DISC. This assignment takes into account the student's potential research area or the specific domain of application of their project.

There are four graduation modalities:

1. **Capstone Project**: This modality involves forming a group of students (assigned randomly from the pool of students seeking graduation through this modality) who will work with a company to solve a problem while being mentored by an advisor. It requires submitting a detailed report of the work done and usually includes a demonstration of the implemented project. The formal duration of this modality is one semester (4 months) for both degrees.
2. **Research Thesis**: This modality entails working on an undergraduate-level research problem under the guidance of a professor who serves as the student's advisor. It is typically individual work, though occasionally it involves groups of two. A detailed report of the research must be submitted. The formal duration is two semesters (8 months) for ICCI and one semester for IECI.
3. **Professional Work**: This modality involves working in a real-world environment for at least 10 months. It does not require working on a specific project but consists of performing a particular role within the organization. A detailed report of the work done must be submitted. This modality is usually reserved for returning students who have been working for several years. The formal duration is two semesters for ICCI and one semester for IECI.
4. **Regular Project**: This modality consists of working on a specific project within an organization. It is usually done individually but sometimes involves groups of two. A detailed report of the work done must be submitted, and it often includes a demonstration of the implemented project. The formal duration is two semesters for ICCI and one semester for IECI.

We note that regardless of the choice of modality, students are required to submit a final report about their work. This final report must be approved by their advisor. In this aspect, certain differences may arise; for example, some modalities might be easier from the perspective that they require less effort in writing the document, such as “Professional Work,” because it mainly relies on the experience acquired by the student. Additionally, certain advisors might have stricter requirements, necessitating more work from the students before submission, potentially causing delays. However, such claims remain speculative unless substantiated by the data itself, which is one of the motivations behind this project.

4 | MATERIALS AND METHODS

In this section, we present the materials and methods used in our study. We begin by describing the data set and its characteristics, followed by an overview of our methodology. We then detail the specific steps involved in data preprocessing, exploratory data analysis, model building, and validation.

4.1 | Data set

The data set comprises information from 209 students who successfully completed all their coursework and were expected to graduate between 2013 and 2017 from ICCI and IECI. Each student is associated with various details, including an advisor, program, graduation modality, proposed submission date for the final report, and the actual submission date when the final report was officially submitted to the evaluation committee. The primary focus of this work centers on the last two features, as they enable us to assess whether students submitted their final reports late.

In particular, each record in the data set represents an individual student and includes the following attributes: a unique identifier for the student, their assigned graduation project advisor, the program they are enrolled in (either ICCI or IECI), their chosen graduation modality (Capstone Project, Research Thesis, Professional Work, or Industry Project), the proposed submission date for their final report, and the actual submission date of the final report. Table 1 provides an example of a single record from the data set.

TABLE 1 Example record from the student data set.

Attribute	Value
Student ID	12,345
Advisor ID	a
Program	ICCI
Graduation Modality	Research thesis
Proposed Submission Date	2017-05-15
Actual Submission Date	2017-06-01

TABLE 2 Example rows from the academic history data set for a single student.

Student ID	Admission year	Admission semester	Graduation date	Course ID	Grade
12345	2011	1	2017-06-15	DAIS-01000	7.0
12345	2011	1	2017-06-15	DAMA-01000	5.5
12345	2011	1	2017-06-15	DAIS-02000	6.0
12345	2011	1	2023-06-15	DAIS-03000	4.5

In addition to this primary data set, we also have access to a secondary data set containing the complete academic history of each student. Each row in this data set includes the student identifier, the year and semester of admission, the graduation date, a course identifier representing a course taken by the student during the program, and the associated grade obtained in that course. Due to the nature of this data, there are multiple rows for each student, representing the various courses they completed throughout their academic journey, resulting in redundant data. Table 2 provides an example of a few rows from this secondary data set for a single student.

This secondary data set allows for a more comprehensive analysis of each student's academic performance throughout their entire program, providing valuable insights into factors that may influence their ability to complete their graduation requirements in a timely manner.

4.2 | Methodology overview

The objective of this study is to predict whether a student will submit their final report late, thereby delaying their graduation, and to identify the variables that influence the timeliness of their submission. We present an overview of the methodology in Figure 1.

Our methodology can be summarized in the following steps.

1. **Preprocessing**: In this phase, the available data is preprocessed. In particular, this means doing cleaning tasks, correcting possible errors in the data, and storing it in an easy-to-use repository.
2. **Exploratory data analysis**: During this phase, we gather general insights about the data, such as its temporal distribution and class balance. This information informs our subsequent phases.
3. **Model building**: In this phase, we create a predictive model using the preprocessed data. It involves selecting an appropriate model and optimizing its parameters.
4. **Validation**: In this phase, we assess the performance of the final models and draw meaningful conclusions. Our primary focus is on detecting the significance and impact of specific variables.

4.3 | Data preprocessing

All the preprocessing was conducted using the Python programming language[3] along with various libraries. Specifically, the data set was stored in a *pandas* DataFrame[4] for convenient manipulation. We utilized a unique number associated with each student to identify them.

Since the primary focus of this project is to determine whether a student will submit their final report late, we removed rows containing missing or invalid values in the columns for the proposed date of submission and the actual submission date. This step resulted in the removal of 98 students out of the initial 209 rows. Consequently, we were left with only 111 students who successfully completed their coursework and graduated. The removed rows corresponded to students who either failed to graduate because they never submitted their final report or students who had not yet graduated at the time of the analysis and had not submitted their report. In both cases, as there was no official submission, it was not meaningful to calculate the delay between the proposed date of submission and the actual submission date.

Using this information, for each student who submitted their work, we generated an indicator variable to determine whether they submitted it late or not. This column serves as our response variable. Therefore, due to the binary nature of this variable, we defined a classification problem and constructed a predictive classifier model using machine learning techniques.

Furthermore, utilizing data from the academic history data set, we calculate the GPA of each student using the official formula provided by the university,

[3] https://www.python.org/

[4] https://pandas.pydata.org/docs/index.html

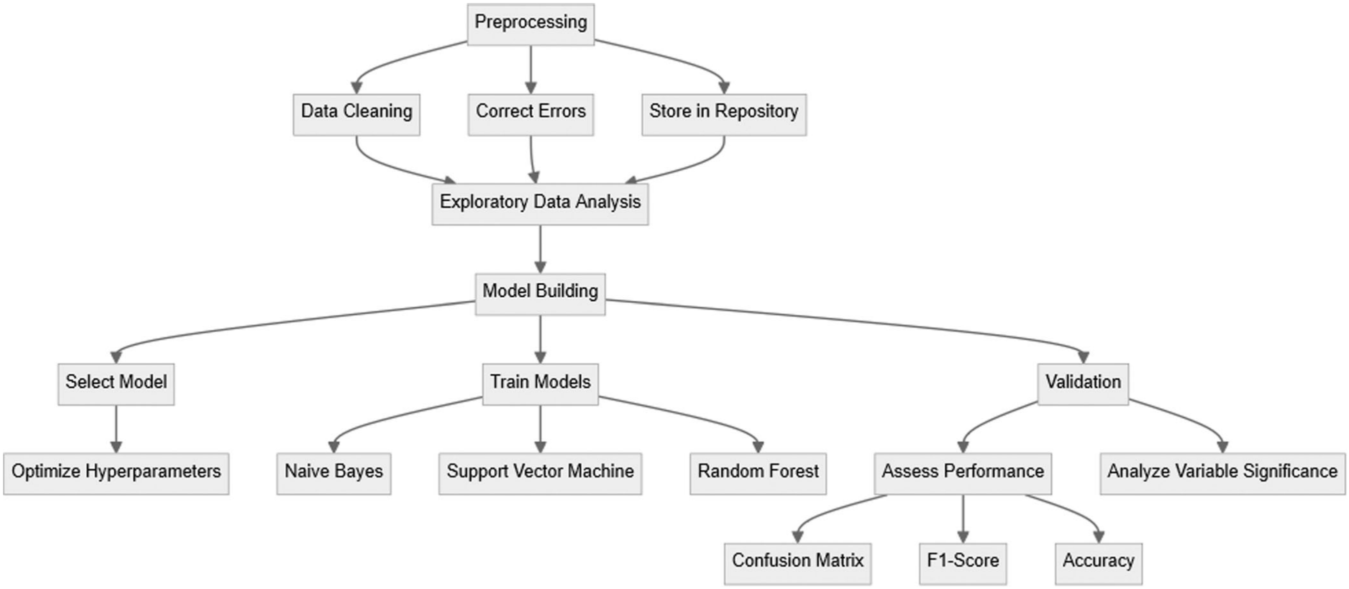

FIGURE 1 Workflow diagram of the predictive modeling process. The process consists of four main stages: (1) Preprocessing; (2) Exploratory Data Analysis; (3) Model Building, involving model selection, parameter optimization, and training of machine learning models; and (4) Validation, which assesses model performance.

which involves a weighted average of all successfully completed courses. It is important to note that failed courses are excluded from this calculation. Additionally, we assume that students cannot submit their final thesis report unless they have successfully completed all their courses; hence, all courses should be approved by the time this submission occurs.

We also possessed information about the year when the students entered their program, allowing us to calculate the duration, in years, it took each student to complete their degree. However, it is important to note that this file does not contain every student in the original data set, resulting in only 49 rows of student data. Consequently, we encounter 62 cases where we lack information on the time it took for these students to graduate from their program. This missing data is random due to variations in the data sources. To address these missing values, we handle them by imputing the mean calculated from the available data of the 49 students.

Given that the columns for graduation modality, program, and advisor contained string data, we converted them using dummy coding. We adopted this approach because the total number of distinct categories was relatively low. This method helps avoid issues related to numerical labeling (as discussed in Eye and Clogg [19]), such as imposing a hierarchical order between the classes indirectly (e.g., implying that advisor A is superior to advisor B because they are listed first). With the data now preprocessed, we have the flexibility to utilize most classifiers available in the *scikit-learn* library [36].

4.4 | Classifier models

We utilized the data set to train various classifiers and assessed their performance on our test set. This evaluation involved constructing a confusion matrix and analyzing different metrics, such as the F1-score and accuracy. Our initial classifier was a NB [23], chosen for its simplicity, which served as a baseline performance benchmark and provided insights into what we could expect from more complex models. Furthermore, we implemented a SVM [15] classifier, chosen for its ability to construct a robust classifier model from small data sets [11]. Finally, we constructed a Random Forest (RF) [25, 44] classifier, which was chosen due to its versatility and the capability to compute feature importance scores directly from the model, as demonstrated by the work of Gutiérrez et al. [22].

The three algorithms—NB, SVM, and RF—were chosen for this study due to their distinct characteristics and suitability for the task at hand. NB was selected as a baseline model due to its simplicity and ability to provide a benchmark for comparison with more complex models. SVMs were chosen for their ability to effectively handle high-dimensional data and their reputation for achieving good performance on classification tasks, particularly with limited samples. Random Forests were included due to their robustness, ability to handle overfitting, and their inherent ability to estimate feature importance, which aligns with our objective of identifying the most influential variables in predicting timely graduation.

4.4.1 | NB

NB is a probabilistic machine learning algorithm based on applying Bayes' theorem with the assumption of conditional independence between features [23]. Despite its simplicity, NB often performs well in real-world applications, particularly in text classification and spam filtering tasks [18, 33]. The model learns the joint probability distribution of the features and the class labels from the training data. During prediction, it uses Bayes' theorem to calculate the posterior probability of each class given the input features and assigns the class with the highest probability [38]. The key advantage of NB is its efficiency in terms of training and prediction time, as well as its ability to handle high-dimensional data. However, the assumption of conditional independence between features is often violated in practice, which can limit the model's performance [37].

4.4.2 | SVM

SVM are a supervised learning algorithm used for both classification and regression tasks [15]. In particular, for binary classification, SVM seeks to find the optimal hyperplane that separates the classes in the feature space [13]. The optimal hyperplane is chosen to maximize the margin of separation between the classes [49].

SVM has several advantages, such as the ability to handle high-dimensional data, its effectiveness in cases where the number of features is greater than the number of samples, and its robustness to outliers [14]. However, the training time of SVM can be longer compared to simpler models, and the choice of kernel function and hyperparameters can impact the performance of the model [26].

SVM can handle nonlinearly separable data using the kernel trick, which involves transforming the input data into a higher-dimensional space where a separating hyperplane can be found [41]. Common kernel functions include linear, polynomial, and RBF kernels [29]. In this study, we test these three alternatives during the hyperparameter optimization phase, allowing the model to handle nonlinear relationships between features.

4.4.3 | Random forests

Random forests are an ensemble learning method that relies on multiple smaller decision trees to create a better model in terms of robustness and accuracy [25, 44]. The algorithm builds a large number of decision trees on bootstrapped samples of the training data, using a random subset of features at each split. This process, known as feature bagging, helps to reduce the correlation between the trees and improves the model's ability to generalize [12].

During prediction, each decision tree in the forest independently predicts the class label for the input sample, and the final prediction is obtained by aggregating the individual predictions [10]. Random forests are known for their ability to handle high-dimensional data. Furthermore, random forests are robust to outliers and noise and provide the ability to estimate feature importance [50].

One of the key advantages of Random Forests is their ability to estimate the importance of each feature in the model. This is typically done by measuring the average decrease in impurity (e.g., Gini impurity or information gain) across all the trees in the forest when a particular feature is used for splitting [31]. This feature importance estimation aligns well with our objective of identifying the most influential variables in predicting timely graduation [22].

4.5 | Model training

We employed a holdout approach, dividing the data into training and test sets using a 70/30 split with stratified classes. This stratification helps prevent issues related to unbalanced classes that could arise from random partitioning. Specifically, the data set consists of 80 instances for training and 31 for testing. However, it is important to note that this data set contains missing data in the column detailing the duration of each student's program completion. To address this missing data, we imputed the missing values using the mean of the available real values.

To adjust the hyperparameters of both classifiers, we conducted a randomized search with stratified K-fold cross-validation [21] comprising 21 folds across 600 iterations using the training set. The scoring metric employed was the weighted F1 score [43], as defined in the *scikit-learn* library [36], to account for label imbalance. Subsequently, we evaluated the performance of the resulting models on the test set.

Finally, we show the 24 features defined as input for our classifier model based on our data set in Table 3.

5 | EVALUATION

In this section, we present the results of our evaluation of the machine learning models developed for predicting on-time graduation among computing engineering

Index	Feature
1	GPA
2	Number of years it took the student to finish their degree
3–16	Advisors with dummy coding
17–20	Graduation modality (“Capstone Project,” “Research Thesis,” “Industry Project,” “Professional Work”) with dummy coding.
21–22	The major of each student (ICCI or IECI) with dummy coding.

TABLE 3 Input features of the classifier models.

students. We begin with an exploratory data analysis to gain insights into the data set, followed by a comparison of the performance of different classifier models. Finally, we examine the variable importance within the best-performing model to identify the key factors influencing timely graduation.

5.1 | Exploratory data analysis

Having preprocessed the data, we performed an exploratory data analysis. Specifically, we examined the class distribution of timely and late submissions, as well as the overall temporal distribution.

Out of the 111 students remaining in the data set, only 34 cases involved students who submitted their final reports late. This indicates an imbalance in the classes, with the class of interest representing only 30% of the data. Consequently, it was essential to employ techniques such as stratification and class weighting to address this imbalance.

Given the data imbalance, we addressed it in our classification model. To do so, we utilized the *compute_class_weight* function from the *scikit-learn* library to determine the weights for each class using the training data set. This yields a weight of 0.7291 for the negative class and 1.590 for the positive class. These weights are then applied to the classifiers to enhance their performance in the minority class.

We conducted a detailed analysis of the extent of delay among students who submitted their final reports later than the originally proposed dates. For this analysis, we generated an Empirical Cumulative Distribution Function (ECDF) plot [20], as illustrated in Figure 2. The ECDF reveals that 50% of the students experienced a delay of 5.07 weeks or less, and 75% experienced delays of 17.53 weeks or less. The average delay, represented by the mean, is 13.096 weeks. It is worth noting that this mean differs from our median value of 5 weeks, indicating the presence of outliers in the data that contribute to an increased average time.

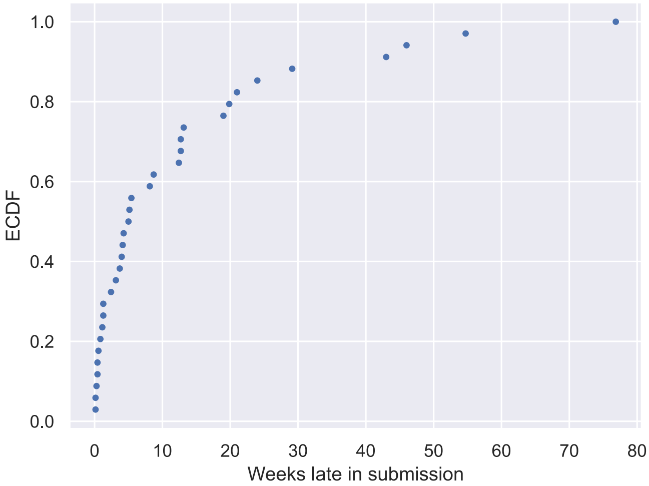

FIGURE 2 Empirical Cumulative Distribution Function of late students.

5.2 | Finding the best classifier

We provide the optimal hyperparameters in Tables 4 and 5, which were determined through cross-validation using the training data. The mean cross-validated F1 score for the best Random Forest model is 0.7183, while for SVMs, it is 0.6873. Note that we exclude NB from the hyperparameter optimization procedures due to its simplicity [5].

The final models are trained using the optimal hyperparameters and subsequently evaluated on the test datasets. The average results are presented in Table 6.

In Table 6, the highest F1 score (0.73) is achieved by our Random Forest classifier, closely followed by the NB model with just a one-point difference. These results are notably significant compared to the null classifier, which assigns every item to the majority class (not late), as they are 18 points higher than the baseline. It is worth noting that the SVM does not perform as well as the other two classifiers on our data set, even after hyperparameter optimization.

We observe marginally better results with our Random Forest classifier compared to the baseline NB

TABLE 4 Random forest hyperparameters found by cross-validation.

Hyperparameter	Value
n_estimators	4
max_depth	2658
max_features	Log2
Bootstrap	True

TABLE 5 Support Vector Machine hyperparameters found by cross-validation.

Hyperparameter	Value
Kernel	Rbf
Gamma	0.6989
Degree	2
C	7.951

TABLE 6 Metrics obtained on the test set.

Classifier	Precision	Recall	F1-score
Naïve Bayes	0.73	0.74	0.72
Random Forest	0.73	0.74	0.73
SVM	0.65	0.65	0.65
All false	0.46	0.68	0.55

classifier, as depicted in the confusion matrices in Figure 3. The key difference lies in the Random Forest model assigning a higher weight to the positive class, resulting in one additional correctly classified true case and one misclassified case. Conversely, the SVM model exhibits inferior performance compared to the NB classifier but still outperforms the null classifier.

5.3 | Variable importance

The Random Forest model allows us to assess the importance of various variables within the model. We performed this analysis using the implementation from the *scikit-learn* library [36], which calculates the importance of each variable based on the mean decrease of impurity [30].

In Table 7, we present the raw results from our permutation importance analysis for all variables in the RF model. However, interpreting these values in this format can be challenging. To simplify the interpretation of the results, we have created Table 8, which provides summarized feature importance by averaging the feature importance scores of similar features into macro features.

Following this approach, we find that the choice of advisor has the lowest mean importance, with only 1.84%. In contrast, the other macro features have higher mean importance. In particular, the modality has a mean importance of 5.34%, and the major has a mean importance of 5.93%. Regarding the continuous variables, we find that the total number of years studying that degree had an importance of 6.94%. Finally, the highest importance was associated with GPA, with 24.05% of the importance.

We conducted a follow-up analysis with permutation importance, which yielded similar results, indicating that, in most cases, the advisor variable does not have a significant influence (multiple instances with zero importance associated with the advisor variables). Combining information from the impurity decrease importance and the follow-up permutation importance analysis, we can conclude that GPA is the most crucial variable in terms of its contribution to our model's predictive ability, followed by years in the degree, and, lastly, majors and graduation modality.

6 | DISCUSSION

In this section, we discuss the key findings of our study and their implications for understanding and promoting on-time graduation among computing engineering students. We also consider the limitations of our work and potential avenues for future research.

6.1 | Impact of advisors and major on submission timeliness

Table 8 displays the importance of each variable in the model. In particular, these results highlight that, individually, each advisor does not significantly contribute to the accurate prediction of late or timely submissions. The highest importance of an advisor is approximately 5.5%. In terms of their correlation with the late submission class, in most cases, there is minimal to no discernible correlation between the advisor and the target class.

Regarding the major in which the student is enrolled, two important observations stand out. Both majors exhibit nontrivial importance in the model, collectively accounting for an average of 5.93% of the feature importance. Specifically, ICCI and IECI majors contribute 4.37% and 6.31%, respectively. Notably, for ICCI, there is a slight negative correlation with late

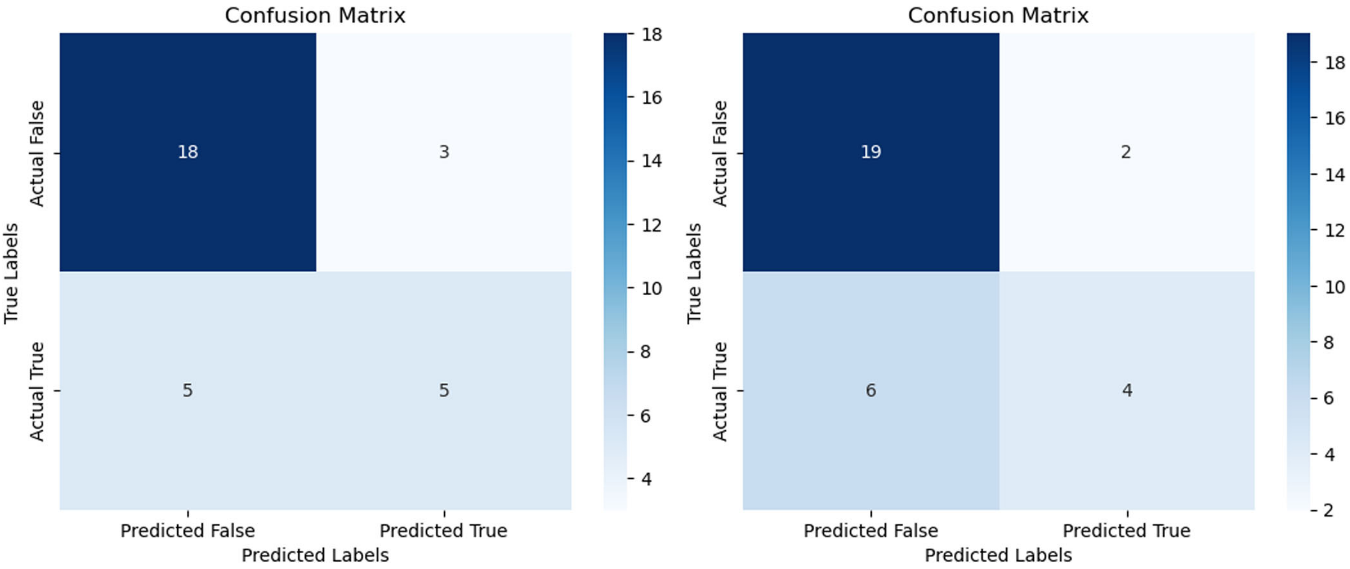

FIGURE 3 Left: Random Forest classifier confusion matrix. Right: Naïve Bayes confusion matrix.

submissions, while for IECI, there is a slight positive correlation with late submissions. This suggests that students in the ICCI major are more likely to submit their work on time, whereas students in the IECI major tend to have associations with late submissions.

6.2 | Role of graduation modalities in submission timeliness

The modalities are also presented in Table 8, which hold a mean importance of 5.34% in the model, similar to the significance of the majors. Individually, they are ranked as follows: Professional Work has the most significant influence on the model outcome at 12.04%, followed by Research Thesis at 5.06%, Capstone Projects at 3.78%, and Industry Projects at 2.85%. Notably, both Capstone Projects and Research Thesis correlate slightly negatively with late submissions. Conversely, Professional Work displays a moderate positive correlation with late submissions, while Regular Projects show no correlation with the target variable.

The correlation results for the modalities align with the specific circumstances faced by students in each case. Capstone Projects were designed to expedite graduation, and a negative correlation, albeit small, with late submissions suggests that their original purpose is being realized. Additionally, students pursuing the Research Thesis modality typically receive more hands-on guidance from their advisors, as their goal often involves producing significant results for publication, potentially increasing the pressure on students to meet deadlines.

On the other hand, Industry Projects represent the most common modality and typically involve off-campus work in an industry-specific project. As a result, there is less direct interaction with advisors and more pressure to meet workplace requirements. Given its prevalence and diverse student and project profiles, it is reasonable to expect that other variables may influence the outcome. Therefore, the nearly zero correlation for this feature is contextually justified. Furthermore, this scenario can serve as a baseline for evaluating the effectiveness of reducing graduation times.

In contrast to the previous examples, Professional Work exhibits a clear, moderate positive correlation. This correlation is easily explained by the context in which these students find themselves. Typically, students who opt for Professional Work are either returning students or individuals who are simultaneously employed and pursuing their degrees. Therefore, it is logical that these students might require more time to submit their work, given their existing commitments to their workplace, professional responsibilities, and, in some cases, additional life responsibilities.

6.3 | Impact of study duration and GPA on submission timeliness

Another crucial variable, according to the model, is the number of years the student has spent studying for the degree. This feature holds a 6.94% importance in our model, but it exhibits virtually no positive or negative correlation with the target class. Therefore, while information about the number of years in the degree contributes to improving our model's results, it has minimal influence on whether the student will submit their work late or on time.

TABLE 7 Feature importance obtained from the best Random Forest classifier ordered from highest to lowest importance using the mean decrease in impurity.

Variable	% Importance	Feature correlation with target
GPA	24.0569	−0.113559
Modality = Professional Work	12.0433	0.35578
Years in Degree	6.9412	0.004653
Major = IECI	6.3079	0.284268
Advisor i	5.5607	−0.166667
Modality = Research Thesis	5.0559	−0.278557
Major = ICCI	4.3694	−0.243562
Advisor d	3.9156	0.232415
Modality = Capstone Project	3.7754	−0.207289
Advisor c	3.2351	−0.097363
Advisor g	3.1406	−0.034091
Modality = Industry Project	2.8538	−0.012563
Advisor a	2.5847	−0.085916
Advisor b	2.3204	−0.012563
Advisor h	1.7791	0.012563
Advisor m	1.7143	0.25332
Advisor j	1.3291	−0.166667
Advisor k	0.9982	0.068608
Advisor f	0.8042	−0.034091
Advisor l	0.6572	0.008682
Advisor e	0.5714	0.312559
Advisor p	0.5248	−0.081502
Advisor o	0.3182	−0.116105
Advisor n	0	N/A * Not enough data.

TABLE 8 Summary of the scores for feature importance.

Summarized Features	% Importance
GPA	24.06
Years in Degree	6.94
Mean for Advisors	1.84
Mean for Majors	5.93
Mean for Modalities	5.34

Finally, the results of GPA show an importance of 24.06% in our model. However, it should be noted that the GPA is a continuous variable, and our method for determining the importance of variables tends to overstate the importance of continuous features. In this context, both the GPA and number of years must be interpreted while considering this aspect. With respect to the correlation analysis, GPA shows a small negative correlation with the positive class, suggesting that a higher GPA is associated with timely submissions. However, the correlation is not strong enough, and thus GPA alone would not be a good predictor. Even then, considering the high importance given to GPA in the model, this finding suggests that this variable should be considered when analyzing the potential for late submissions of future students.

6.4 | Sample size and study scope limitations

One of the main limitations of this study is the relatively small sample size. Due to the limited number of graduates in the ICCI and IECI programs during the period from 2013 to 2017, our data set consisted of only 209 students. After preprocessing and removing records with missing or invalid values, the final sample size was further reduced to 111 students who successfully completed all their graduation requirements. This small sample size may limit the generalizability of our findings to other cohorts or institutions. It is essential to validate the predictive model on larger and more diverse data sets to assess its robustness and applicability in different contexts. Future research should aim to collect data from a broader range of students, programs, and institutions to enhance the external validity of the results and provide more comprehensive insights into the factors influencing on-time graduation in computing engineering programs.

Furthermore, we note that incorporating additional output values, such as predicting the final average grade based on early academic performance or entrance exam scores, could potentially enhance the model's predictive capabilities. However, such extensions are beyond the scope of the current study. Our research primarily focuses on identifying the factors influencing the timely submission of the final graduation project report, and exploring additional output variables would introduce new dimensions to the analysis that would require a substantial expansion of the study's objectives and methodology. To maintain a clear and focused research narrative, we have chosen to concentrate on the core question of predicting on-time graduation based on the available data. By delimiting the scope of our study in this manner, we aim to provide a more targeted and in-depth analysis of the key factors influencing timely graduation in computing engineering programs. Future research could build upon our findings by investigating the predictive power of additional output values, such as the final average grade or entrance exam scores, to further enhance the understanding of student success factors in higher education.

6.5 | Contextual limitations

Another limitation of this study is that the data used for analysis comes from a single institution located in the north of Chile. The specific characteristics of this institution, such as its educational approach, student demographics, and cultural context, may limit the applicability of our findings to other institutions, regions, or countries. It is crucial to recognize that the factors influencing on-time graduation in computing engineering programs may vary across different educational settings and cultural backgrounds. Therefore, the results of this study should be interpreted with caution when considering their relevance to other contexts.

Finally, we note that our study does not validate the predictive model using an external data set from a different institution or region. This lack of external validation may limit the robustness and generalizability of the results. To ensure the model's performance and applicability in diverse settings, future research should aim to validate the findings using data from multiple institutions and compare the results across different contexts. This approach would help to identify the common factors influencing on-time graduation and assess the model's ability to generalize beyond the specific institution studied in this research.

7 | CONCLUSIONS AND FUTURE RESEARCH LINES

The present study has successfully developed a predictive machine-learning-based model for on-time graduation among computing engineering students at a university in the north of Chile, covering the period between the end of coursework and the final thesis submission. Our findings highlight the importance of factors such as GPA, graduation modality, and program duration in influencing the timely completion of graduation theses. This model not only identifies at-risk students but also provides insights for educators and administrators to implement targeted interventions. It underscores the necessity for personalized support strategies that consider individual student profiles and academic trajectories.

The adoption of Random Forests as the preferred methodology for constructing the predictive machine-learning model in this study is based on its general effectiveness as an ensemble-based technique. In general, Random Forests models provide a robust and versatile framework for predictive modeling. Furthermore, the model's inherent ability to rank the importance of features allows for a focused examination of key determinants impacting on-time graduation.

The authors acknowledge the specific context in which the model was developed. Therefore, they emphasize the need for future research to validate the model by incorporating recent student data. Additionally, it is recommended to explore its applicability to other disciplines and institutions to understand its broader relevance and effectiveness.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available on request from the corresponding author. The data are not publicly available due to privacy or ethical restrictions.

ORCID

Brian Keith http://orcid.org/0000-0001-5734-8962
Ricardo Pizarro http://orcid.org/0000-0001-7894-0236

REFERENCES

1. A. I. Adekitan and O. Salau, *The impact of engineering students' performance in the first three years on their graduation result using educational data mining*, Heliyon **5** (2019), no. 2, e01250. https://doi.org/10.1016/j.heliyon.2019.e01250
2. J. M. Aiken, R. De Bin, M. Hjorth-Jensen, and M. D. Caballero, *Predicting time to graduation at a large enrollment American university*, PLoS One **15** (2020), no. 11, e0242334. https://doi.org/10.1371/journal.pone.0242334
3. G. Akçapınar, M. N. Hasnine, R. Majumdar, B. Flanaganand H. Ogata, *Developing an early-warning system for spotting at-risk students by using eBook interaction logs*, Smart Learn. Environ.**6** (2019), 4.
4. L. Al-Alawi, J. Al-Shaqsi, A. Tarhini, and A. S. Al-Busaidi, *Using machine learning to predict factors affecting academic performance: the case of college students on academic probation*, Educ. Inform. Technol. **28** (2023), 12407–12432. https://doi.org/10.1007/s10639-023-11700-0
5. M. V. Albert, K. Kording, M. Herrmann, and A. Jayaraman, *Fall classification by machine learning using mobile phones*, PLoS One **7** (2012), no. 5, e36556. https://doi.org/10.1371/journal.pone.0036556

6. L. Aulck, N. Velagapudi, J. Blumenstock, and J. West, *Predicting student dropout in higher education, Proceedings of the ICML Workshop #Data4Good: Machine Learning in Social Good Applications*, New York, NY, USA, (2016).
7. M. Avendaño, M. Cantillánez, and J. E. González, *Implementation of seed collection programs for the recovery of Argopecten purpuratus populations in the La Rinconada marine reserve (Antofagasta, Chile)*, Aquat. Conserv. Marine Freshwater Ecosyst. **33** (2023), no. 2, 215–225. https://doi.org/10.1002/aqc.3910
8. J. Berens, K. Schneider, S. Görtz, S. Oster, and J. Burghoff, *Early detection of students at risk–predicting student dropouts using administrative student data and machine learning methods*, J. Educ. Data Min. **11** (2018), 1–41. https://doi.org/10.5281/zenodo.3594771
9. M. R. Berthold, N. Cebron, F. Dill, T. R. Gabriel, T. Kötter, T. Meinl, P. Ohl, C. Sieb, K. Thiel, and B. Wiswedel, *KNIME: The Konstanz Information Miner*, Data analysis, machine learning and applications. studies in classification, data analysis, and knowledge organization (C. Preisach, H. Burkhardt, L. Schmidt-Thieme, and R. Decker, eds.), Springer, Berlin, Heidelberg, 2008. https://doi.org/10.1007/978-3-540-78246-9_38
10. G. Biau and E. Scornet, *A random forest guided tour*, Test **25** (2016), 197–227.
11. O. Boursalie, R. Samavi, and T. E. Doyle, *M4CVD: mobile machine learning model for monitoring cardiovascular disease*, Proc. Comput. Sci. **63** (2015), 384–391. https://doi.org/10.1016/j.procs.2015.08.357
12. L. Breiman, *Random forests*, Mach. Learn. **45** (2001), 5–32.
13. C. J. C. Burges, *A tutorial on support vector machines for pattern recognition*, Data Mining Knowl. Discovery **2** (1998), no. 2, 121–167.
14. E. Byvatov and G. Schneider, *Support vector machine applications in bioinformatics*, Appl. Bioinformatics. **2** (2003), no. 2, 67–77.
15. C. Cortes and V. Vapnik, *Support-vector networks*, Mach. Learn. **20** (1995), 273–297. https://doi.org/10.1007/BF00994018
16. S. L. de Silva and P. W. Francis, Volcanoes of the Central Andes, Springer-Verlag, Berlin, 1991, p. 216.
17. J. Delors, I. Al Mufti, I. Amagi, R. Carneiro, F. Chung, B. Geremek, W. Gorham, A. Kornhauser, M. Manley, M. Padrón, M. A. Savané, K. Singh, R. Stavenhagen, M. Won, and Z. Nanzhao,*Learning: the treasure within*. Report to UNESCO of the International Commission on Education for the Twenty-first Century (highlights), 1996.
18. V. P. Deshpande, R. F. Erbacher, and C. Harris, An evaluation of Naïve Bayesian anti-spam filtering techniques. In 2007 IEEE SMC Inform. Assurance Secur. Worksh., IEEE, 2007, 333–340.
19. A. Eye and C. C. Clogg, Categorical variables in developmental research: methods of analysis, Elsevier, San Diego, CA, USA, 1996.
20. B. Flem, C. Reimann, K. Fabian, M. Birke, P. Filzmoser, and D. Banks, *Graphical statistics to explore the natural and anthropogenic processes influencing the inorganic quality of drinking water, ground water and surface water*, Appl. Geochem. **88** (2018), 133–148. https://doi.org/10.1016/j.apgeochem.2017.09.006
21. T. Fushiki, *Estimation of prediction error by using K-fold cross-validation*, Stat. Comput. **21** (2011), 137–146. https://doi.org/10.1007/s11222-009-9153-8
22. L. Gutiérrez, V. Flores, B. Keith, and A. Quelopana, *Using the Belbin method and models for predicting the academic performance of engineering students*, Comput. Appl. Eng. Educ. **27** (2019), no. 2, 500–509. https://doi.org/10.1002/cae.22092
23. D. J. Hand and K. Yu, *Idiot's Bayes: not so stupid after all?* Int. Stat. Rev./Revue Internationale de Statistique **69** (2001), no. 3, 385–398. https://doi.org/10.2307/1403452
24. S. Helal, J. Li, L. Liu, E. Ebrahimie, S. Dawson, D. J. Murray, and Q. Long, *Predicting academic performance by considering student heterogeneity*, Knowl.-Based Syst. **161** (2018), 134–146.
25. T. K. Ho, Random decision forests, in Proc. 3rd Int. Conf. Document Anal. Recogn., Montreal, QC, Canada, vol. **1**, 1995, 278–282. https://doi.org/10.1109/ICDAR.1995.598994.
26. C.-W. Hsu, C.-C. Chang, and C.-J. Lin, *A practical guide to support vector classification*, Data Sci. Assoc. (2003), 1396–1400.
27. S. Huang and N. Fang, *Predicting student academic performance in an engineering dynamics course: a comparison of four types of predictive mathematical models*, Comput. Educ. **61** (2013), 133–145.
28. P. Kaur, M. Singh, and G. S. Josan, *Classification and prediction based data mining algorithms to predict slow learners in education sector*, Proc. Comput. Sci. **57** (2015), 500–508.
29. S. S. Keerthi and C. J. Lin, *Asymptotic behaviors of support vector machines with Gaussian kernel*, Neural Comput. **15** (2003), no. 7, 1667–1689.
30. X. Li, Y. Wang, S. Basu, K. Kumbier, and B. Yu, A debiased MDI feature importance measure for random forests, *Proceedings of the 33rd International Conference on Neural Information Processing Systems (NIPS'19)*. Curran Associates Inc., Red Hook, NY, USA, (2019), 8049–8059.
31. G. Louppe, L. Wehenkel, A. Sutera, and P. Geurts, Understanding variable importances in forests of randomized trees, *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1 (NIPS'13)*. Curran Associates Inc., Red Hook, NY, USA, (2013), 431–439.
32. F. Marbouti, H. A. Diefes-Dux, and K. Madhavan, *Models for early prediction of at-risk students in a course using standards-based grading*, Comput. Educ. **103** (2016), 1–15.
33. A. McCallum and K. Nigam, A comparison of event models for naive Bayes text classification. In AAAI-98 Worksh. Learn. Text Categorization, vol. **752**, 1988, 41–48.
34. OECD, Mining regions and cities in the region of Antofagasta, Chile: towards a regional mining strategy, OECD Rural Studies, OECD Publishing, Paris, 2023. https://doi.org/10.1787/336e2d2f-en
35. E. Osmanbegovic and M. Suljic, *Data mining approach for predicting student performance*, Econ. Rev. J. Econ. Bus. **10** (2012), no. 1, 3–12.
36. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, and J. Vanderplas, *Scikit-learn: machine learning in python*, J. Mach. Learn. Res. **12** (2011), 2825–2830.
37. J. D. Rennie, L. Shih, J. Teevan, and D. R. Karger. Tackling the poor assumptions of naive Bayes text classifiers. Proc. 20th Int. Conf. Mach. Learn. (ICML-03), 2003, 616–623.
38. I. Rish, An empirical study of the naive Bayes classifier. In IJCAI 2001 Worksh. Empirical Methods Artif. Intell., vol. **3**, 2001, 41–46.
39. S. Russell and P. Norvig, Artificial intelligence: a modern approach, 4th edition., Pearson Education Inc., Hoboken, NJ, USA, 2021.

40. S. Z. Salas-Pilco and Y. Yang, *Artificial intelligence applications in Latin American higher education: a systematic review*, Int. J. Educ. Technol. High. Educ. **19** (2022), 21. https://doi.org/10.1186/s41239-022-00326-w
41. B. Schölkopf, A. Smola, and K. R. Müller, *Nonlinear component analysis as a kernel eigenvalue problem*, Neural Comput. **10** (1998), no. 5, 1299–1319.
42. V. C. Smith, A. Lange, and D. R. Huston, *Predictive modeling to forecast student outcomes and drive effective interventions in online community college courses*, Online Learn. **16** (2012), no. 3, 51–61.
43. A. A. Taha and A. Hanbury, *Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool*, BMC Med. Imaging **15** (2015), no. 29, 29. https://doi.org/10.1186/s12880-015-0068-x
44. H. Tin Kam, *The random subspace method for constructing decision forests*, IEEE. Trans. Pattern. Anal. Mach. Intell. **20** (1998), no. 8, 832–844. https://doi.org/10.1109/34.709601
45. V. Tinto, *Dropout from higher education: a theoretical synthesis of recent research*, Rev. Educ. Res. **45** (1975), 89–125. https://doi.org/10.3102/00346543045001089
46. E. Unda-Sanzana, *Antofagasta: astronomy education on the shoulders of giants*, Proc. Int. Astron. Union **15** (2019), S367, 419–420. https://doi.org/10.1017/S174392132100051X
47. R. Valdés-Cotera, *UNESCO's utopia of lifelong learning: an intellectual history*, Int. Rev. Educ. **65** (2019), 667–669. https://doi.org/10.1007/s11159-019-09797-y
48. A. Van Barneveld, K. E. Arnold, and J. P. Campbell, *Analytics in higher education: establishing a common language*, EDUCAUSE Learn. Initiative **1** (2012), no. 1, l–ll.
49. V. Vapnik, The nature of statistical learning theory, Springer Science & Business Media, New York, NY, USA, 2013.
50. A. Verikas, A. Gelzinis, and M. Bacauskiene, *Mining data with random forests: a survey and results of new tests*, Pattern Recogn. **44** (2011), no. 2, 330–349.

AUTHOR BIOGRAPHIES

Aldo Quelopana is the Chair of the Department of Computing and Systems Engineering at the Universidad Católica del Norte, having previously served as the Program Director of the computing and informatics civil engineering program. He earned his BSc and MSc degrees in computer engineering from the same institution (2008), a master's in project management from the University of Melbourne, Australia (2015), and a PhD in Mining Engineering from McGill University, Canada (2024). He currently teaches courses related to IT Project Management and Information Technology for Mining Processes.

Brian Keith is an assistant professor at the Department of Computing and Systems Engineering at the Universidad Católica del Norte. He obtained his PhD in computer science and applications at Virginia Tech in 2023. His research areas include visualization, artificial intelligence, and computational narratives. Brian obtained a degree in civil engineering in computing and informatics in 2016, a BA degree in mathematics in 2017, and an MA degree in computer engineering in 2017 from Catholic University of the North.

Ricardo Pizarro is a current PhD student in the electronics department of the University of Alcalá. He received his bachelor's degree in civil engineering in computing and informatics and master's degree in informatics engineering from the Universidad Católica del Norte, Chile. His current research interests include human action recognition and detection in videos.

How to cite this article: A. Quelopana, B. Keith, and R. Pizarro, *Predictive modeling of on-time graduation in computing engineering programs: A case study from Northern Chile*, Comput. Appl. Eng. Educ. (2024);e22767. https://doi.org/10.1002/cae.22767