INSTITUTO
UNIVERSITÁRIO
DE LISBOA

Department of Quantitative Methods for Management and Economics/ Department of Information Science and Technology

PREDICTION OF STUDENT SUCCESS: A SMART DATA-DRIVEN APPROACH

Master in Data Science

By

ANA ROSA ALMEIDA PINTO

Supervisor:

PhD, Anabela Ribeiro Dias da Costa, Assistant Professor

ISCTE-IUL

Co-Supervisor:

PhD, Diana Elisabeta Aldea Mendes, Associate Professor

ISCTE-IUL

October, 2022

Department of Quantitative Methods for Management and Economics/ Department of Information Science and Technology

PREDICTION OF STUDENT SUCCESS: A SMART DATA-DRIVEN APPROACH

Master in Data Science

By

ANA ROSA ALMEIDA PINTO

Supervisor:

PhD, Anabela Ribeiro Dias da Costa, Assistant Professor

ISCTE-IUL

Co-Supervisor:

PhD, Diana Elisabeta Aldea Mendes, Associate Professor

ISCTE-IUL

October, 2022

Acknowledgments

I would like to thank my two amazing supervisors, professors Anabela Costa and Diana Aldea Mendes, for their outstanding availability and understanding, and especially for encouraging me to finish the thesis even when it seemed impossible to do so.

Special acknowledgments to my parents and sister who always supported me and were my safety net throughout the master. Thank you for always believing in me, words cannot describe how lucky I am for having such a supporting and caring family.

Abstract

Predicting student's academic performance is one of the subjects related to the Educational Data Mining process, which intends to extract useful information and new patterns from educational data. Understanding the drivers of student success may assist educators in developing pedagogical methods providing a tool for personalized feedback and advice.

In order to improve the academic performance of students and create a decision support solution for higher education institutes, this dissertation proposed a methodology that uses educational data mining to compare prediction models for the students' success. Data belongs to ISCTE master students, a Portuguese university, during 2012 to 2022 academic years. In addition, it was studied which factors are the strongest predictors of the student's success. PyCaret library was used to compare the performance of several algorithms. Factors that were proposed to influence the success include, for example, the student's gender, previous educational background, the existence of a special statute, and the parents' educational degree.

The analysis revealed that the Light Gradient Boosting Machine Classifier had the best performance with an accuracy of 87.37%, followed by Gradient Boosting Classifier (accuracy = 85.11%) and Adaptive Boosting Classifier (accuracy = 83.37%). Hyperparameter tunning improved the performance of all the algorithms. Feature importance analysis revealed that the factors that impacted the student's success most were the average grade, master time, and the gap between degrees, *i.e.*, the number of years between the last degree and the start of the master.

Keywords: Student's success; predicting; modelling; educational data mining.

Resumo

A previsão do sucesso académico de estudantes é um dos tópicos relacionados com a mineração de dados educacionais, a qual pretende extrair informação útil e encontrar padrões a partir de dados académicos. Compreender que fatores afetam o sucesso dos estudantes pode ajudar, as instituições de educação, no desenvolvimento de métodos pedagógicos, dando uma ferramenta de feedback e aconselhamento personalizado.

Com o fim de melhorar o desempenho académico dos estudantes e criar uma solução de apoio à decisão, para instituições de ensino superior, este artigo propõe uma metodologia que usa mineração de dados para comparar modelos de previsão para o sucesso dos alunos. Os dados pertencem a alunos de mestrado que frequentaram o ISCTE, uma universidade portuguesa, durante os anos letivos de 2012 a 2022. Além disso, foram estudados quais os fatores que mais afetam o sucesso do aluno. Os vários algoritmos foram comparados pela biblioteca PyCaret. Alguns dos fatores que foram propostos como relevantes para o sucesso incluem, o género do aluno, a formação educacional anterior, a existência de um estatuto especial e o grau de escolaridade dos pais.

A análise dos resultados demonstrou que o classificador *Light Gradient Boosting Machine* (LGBMC) é o que tem o melhor desempenho com uma *accuracy* de 87.37%, seguindo-se o classificador *Gradient Boosting Classifier* (*accuracy*=85.11%) e o classificador *Adaptive Boosting* (*accuracy*=83.37%). A afinação de hiperparâmetros melhorou o desempenho de todos os algoritmos. As variáveis que demonstraram ter maior impacto foram a média dos estudantes, a duração do mestrado e o intervalo entre estudos.

Palavras-Chave: Sucesso académico; previsão; modelação; mineração de dados educacionais.

Index

Tables Index

Figures Index

Abbreviations

ACC - Accuracy

ADA - Adaptive Boosting Classifier

AGCN - Attention-based Graph Convolutional Networks

ANN - Artificial Neural Network

AUC - Area Under The Curve

CGPA - Cumulative Grade Point Average

CRISP-DM - Cross Industry Standard Process for Data Mining

DGES - Direção Geral do Ensino Superior

DT - Decision Tree

ECTS - European Credit Transfer and Accumulation System

EDM - Educational Data Mining

FN - False Negatives

FP - False Positives

GBC - Gradient Boosting Classifier

GPA - Grade Point Average

HE - Higher Education

HEI - Higher Education Institutions

KNN - K-nearest neighbor

LGBMC/ LightGBMC/ LGBM - Light Gradient Boosting Machine Classifier

LSTM - Long short term memory

MAE - Mean Absolute Error

MLP - Multilayer Perceptron

NB - Naïve Bayes

OECD - Organization for Economic Co-operation and Development

RF - Random Forest

RMSE - Root Mean Squared Error

RNN - Recurrent neural networks

SHAP - SHapley Additive exPlanations

TN - True Negatives

TP - True positives

TT - Training Time

Introduction

Accordingly to *Direção Geral do Ensino Superior* (DGES), Portugal registered in 2021 a historic number of enrolments in higher education, 412,000 students, reaching the highest rate in the last decade (DGES, 2021). Masters' enrolment concentrates 16% of those enroled and grew 4% compared to the previous year. In 2022 the trend continued, and Portugal had a new pike, 433,217 students enroled in Higher Education (HE). On the other hand, grade repetition has been identified by the Organization for Economic Co-operation and Development (OECD) as one of the main problems of the Portuguese education system. Liebowitz *et al.* (2018) reported that the share of early school leavers is substantial and many of those fail to pursue additional training – 13 out of 100 18-24 year-olds have not completed upper secondary education and are not enroled in any further training or education, in Portugal. One of the main goals set by OCDE for Portugal is the reduction of student dropout and year repetition rates and the need for metrics to measure success in improving equity, performance, and school dropout rates.

Dropout, termination of studies at a premature level, or high retention time are problems faced by Higher Education Institutions (HEI). Those problems affect students, their families, institutions, and the government. Finding ways to prevent and unveil the reasons behind those issues remains challenging and is of utmost importance. Drop out is not a novelty but continues to be a major topic for researchers' attention due to its impact that can ultimately influence prospective students to lose their opportunity to study in higher education (Hutagaol & Suharjito, 2019). The increase in Portuguese enrolment students shows the importance of correctly allocating the institution's resources to best serve the highest number of students.

Among many other solutions, to control the student dropout rate, one is the creation of a prediction mechanism whereby students and institutions can be warned about their potentially poor performance (Sultana *et al.*, 2017).

Educational Data Mining (EDM) is an interdisciplinary area related to methods designed to explore and extract information from education data. Generally, EDM is applied to develop computational approaches that combine theory and data to assist with and enhance the quality of academic performance of students and graduates, and faculty information of these institutions. EDM uses several techniques, such as Decision Trees (DT), K-Nearest Neighbor (KNN), and Naïve Bayes (NB) (Hashim *et al.*, 2020).

The student's performance prediction is associated with different features. The most frequently used features are the following: the grade point average (GPA) and internal assessments (such as exam marks, assignment marks, and quizzes), followed by student demographic data (such as gender, age, and residence) and external assessments (such as final exam mark for specific a subject). Moreover, high school background, scholarship, and extra-curricular activities are also used by researchers (Alshdaifat *et al.*, 2020).

EDM techniques and approaches rely on the data type and the study context; thus, having a methodology developed for a specific Portuguese institution will provide new insights for those institutions and help it create different target actions to increase student success.

The work presented in this document is contextualized in this research track. This study presents a data-driven methodology to create a model that predicts the master student's success in ISCTE, a Portuguese university established in 1972, which currently has approximately 10000 students enroled in undergraduate and postgraduate programs. In addition, the study aims to understand the variability factors that most impact the student's success.

The current dissertation is organized as follows. The *Literature Review* in Chapter 1 contextualizes the reader about the topic and sums up the previous studies developed in this area. *Methodology* in Chapter 2 describes the methodology followed, models, and performance metrics selected – explaining the employed data-driven approach. In Chapter 3 (*Results and Discussion*) are presented the Machine Learning experiments, the results, and the evaluation of the results obtained in this study. Finally, *Conclusions and Future Work* present the main conclusions of the work.[1]

[1] Code could be provided if requested to the authors.

CHAPTER 1

Literature Review

A success indicator of an educational institution is students' learning outcomes, which could be positive, related to high Grade Point Average (GPA) and rate of graduated students, or negative, for high dropouts or long study periods. Therefore, one of the most important duties of educational organizations and administrators is to improve student success (Karpicke & Murphy, 1996, cited by (Altun *et al.*, 2022).

These institutes collect lots of potentially valuable data as they have data related to students' admission, progression, and graduation. Analysing student performance, benefits not only the institutions but also the students, parents, government, and financiers (Muchuchuti *et al.*, 2020).

Despite the huge amount of data available in higher education institutions (HEI), most institutions have not been able to analyse this data and transform it into valuable information (Miguéis *et al.*, 2018).

Educational Data Mining is the process of applying data mining tools and techniques to analyse the data at educational institutions (Al-Mahmoud & Al-Razgan, 2015). EDM emerged to take advantage of the vast amounts of data generated from the educational ecosystem (Al-Barrak & Al-Razgan, 2016). Educational institutions use educational data mining to gain deep and thorough knowledge to enhance the assessment, evaluation, planning, and decision-making in their educational programs. Predicting and investigating the performance/success of the students is essential to assist educators in identifying weaknesses and enhancing academic scores (Baashar *et al.*, 2021).

Understanding the drivers of student success may assist educators in developing pedagogical methods providing a tool for personalized feedback and advice. Early detection of student susceptibility to academic failure could also serve as a filter system for enroling first-year students into universities since there is increasing competition for students to get admitted to universities, especially in engineering universities (Sultana *et al.*, 2017).

Predicting student performance and the factors that impact that performance may help organizations create different target actions considering the types of students, which may also result in a more efficient allocation of the institutions' resources (Miguéis *et al.*, 2018).

By reviewing the literature related to student performance prediction, two main factors are highlighted: attributes and techniques. The most commonly used attributes are Cumulative Grade Point Average (CGPA) and internal assessment (Yaacob *et al.*, 2019). In addition, demographic data (age, gender, student's area, parent's income, parent's education level) is also commonly used (Hutagaol & Suharjito, 2019).

Several EDM techniques for student performance prediction have been reported, such as regression analysis, Decision Trees, Naive Bayes, and artificial neural networks (ANN) (Sultana *et al.*, 2017).

A systematic literature review was conducted to identify the most relevant machine learning methodologies to predict students' academic performance. In order to achieve so, the review followed the approach used by Baashar *et al.* (2021), which involves creating the subsequent methodologic steps: (1) research questions, (2) inclusion criteria, (3) information source and search strategy, and (4) study selection.

The review examined the academic literature from 2013 to October 2022 and later grouped and unfolded the most relevant research in student performance forecasting. A set of articles was scrutinized under the following criteria: i) the methodology used for analysing data; ii) data source regarding the education level; iii) metrics; iv) study goal; v) the publication year. Therefore, the following research questions have been addressed:

Q1: What is the students' academic level used to predict students' academic performance?

Q2: What attributes/features are used to predict students' academic performance?

Q3: What machine learning approaches are used for the prediction?

Q4: Which model performs best?

The articles included were i) directly relevant to forecast/predict/classify the students-performance/academic-performance/students-success ii) using machine learning approaches; iii) belonging to students from a bachelor, university, or under/graduated data; iv) focused on educational data.

The Scopus search engine was used to gather the relevant literature, and a cast around was performed for the time period between 18th October 2021 and 10th October 2022.

The following keywords and terms were combined: ("student* performance" OR "academic performance" OR "student* success") AND (forecast OR prediction OR classification) AND (accuracy OR "performance metric*") AND ("machine learning" OR "data mining") AND "educational data" AND ("grade*" OR "score") AND (universit* OR bachelor OR graduat*). Articles were selected based on title, abstract, and keywords, using the inclusion criteria described above. The query presented was achieved through an iterative process that incrementally reduced the number of results by increasing the number of filters used.

Query started with synonyms terms for students' success, once this is the thesis focus. We observed the large demand for this research topic, which may be justified by the effects of Covid-19 on learning/grade evaluation process. Subsequently, (forecast OR prediction OR classification) was added to the previous query since the present analysis is concentrated on predicting the student's success. The models used by different authors can be compared by analysing the accuracy and other performance metrics. Therefore (accuracy OR "performance metric*") was added to the query.

Further, to obtain articles related to machine learning and data mining, both filters were added to the query. The next filter added was "educational data" since it relates to academic performance studies (Prada *et al.*, 2020). Subsequently, ("grade*" OR "score") filter was added since the current study aims to analyse the models that predict the student's performance by using mainly grades or scores. Lastly, (universit* OR bachelor OR graduat*) was attached to the query to filter the articles that use data belonging to bachelor, university, or graduate level students.

After applying the previously mentioned inclusion/exclusion criteria, the number of articles was reduced to 40 relevant articles; 36 were further analysed in the second phase. The four excluded articles belonged to: one article that was not freely available, two whose theme was out of scope, and one that was a systematic review.

The 36 collected articles were scrutinized to unveil information to answer the research questions. Figure 1 illustrates the distribution of the selected articles per year and the number of times the papers were cited. As shown in Figure 1, most of the included articles were published between 2019 and 2020. As expected, the number of citations is lower in 2021 and 2022 since the corresponding articles were the last published, so the time to be cited is also minor. The articles published in 2018 were the most cited by other papers.

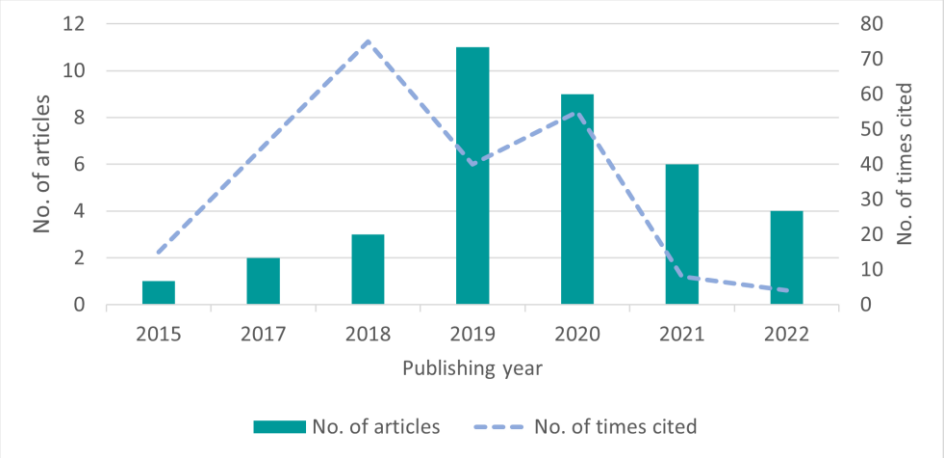

Figure 1. Number of articles and the number of times the articles were mentioned by year.

To investigate the students' academic level - used in the different studies - the articles were grouped by student's study field and dataset origin (**Error! Reference source not found.**). Articles that used data about students from different subject areas, *i.e.,* for instance, mathematics and biology, were included in both categories; thus, the same article may be part of more than one group. Another category comprises subject fields that were mentioned only in one article. Regarding the study field, Mathematics and Engineering pupils are the main focus of the articles. Engineering disciplines and degrees have been strongly correlated with low GPA, very long study periods, and high rate of dropout (Bayer, Bydžovská, Géryk, Obšívač, & Popelínský, 2012; Márquez-Vera *et al.*, 2016 cited by Dewantoro & Ardisa, 2020). Therefore, most studies are performed in engineering fields, as shown in **Error! Reference source not found.**. Only 6% of the papers were centred on high school data (Magbag & Raga, 2020; Nudelman *et al.*, 2019; Rosado *et al.*, 2019). Summing up findings regarding Q1, most studies use data from computer science masters.

About the dataset origin, 35 articles used data created after the student enrolment (*e.g.*, GPA, score, number of attempts), and 14 used data available in admission (*e.g.*, type of previous education, admission score, student status).

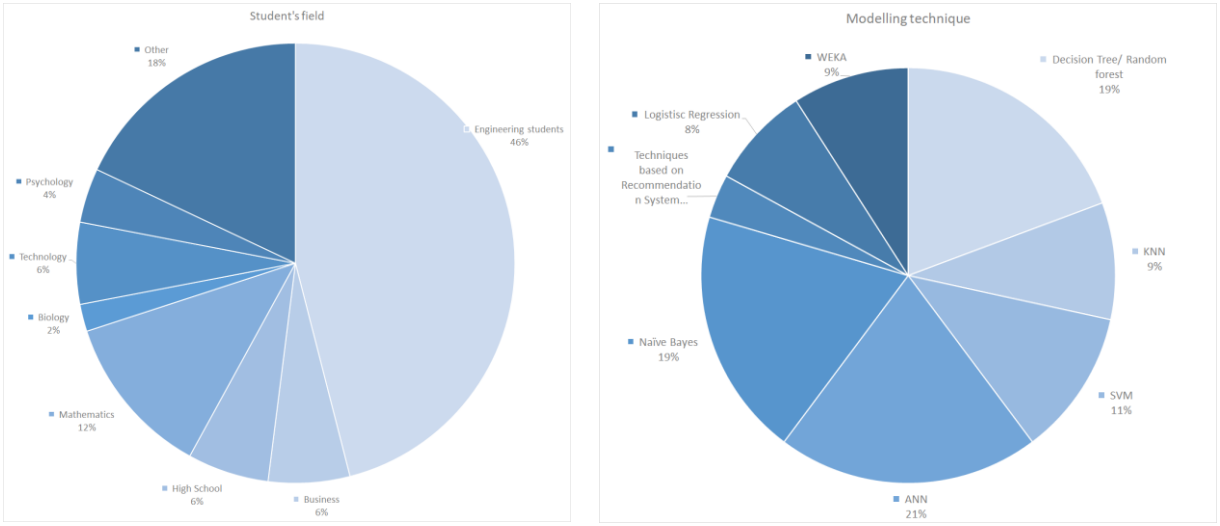

Figure 2. Article's datasets origin, data is from prior admission to university or after the student's enrolment (left). Article's division by Modelling Technique (right).

A broad range of features is selected for predicting the student's performance, which may vary with the final purpose of the studies, the techniques that will be applied, and so on. While some studies select features such as student demographic data (age, gender, country of residence, citizenship), others may include the courses' final grades (Prada *et al.*, 2020). As mentioned, feature/attribute selection can vary according to the study goal; for example, Aydoğdu (2020) used the number of attendings to live sessions, total time spent in live sessions, and time spent on the content for predicting final performance in online learning environments. Due to the wide variety of features selected for each study, the current review created three feature categories: student explanatory information, degree information, and student performance, following the methodology used by Prada *et al.*(2020). Student descriptive information includes data related to the student's parents' education, demographic data, and surveys. Degree information includes data about the subject/degree student it is taking. Student's performance category integrates facts related to subjects scores, GPA, number of attempts, and all the academic data. Articles that selected one or more examples that belonged to one category were included in that category. Thus, the same article may be part of more than one group. The results are presented in Figure 6 – Appendix A.

In Q2, features related to the student's performance and subject/course information, including knowledge area, average score, failure rate, mobility status, and GPA - are most eligible by the articles (57.8%).

The machine learning technique selection depends on the study's primary objective; thus, it is important to group the collected articles by the goal. The article's primary objectives were fractionated into three groups: a) articles that aim to study the prediction of student's success, student's dropout or struggling (binary – *e.g.*, will succeed / won't succeed); b) student classification (*e.g.*, Excellent; Low-performance and Average (Prada *et al.*, 2020)) and, c) prediction of student's grade. The main focus of the collected articles is based on the student's success/dropout prediction and the prediction of the student's score (87%), as shown in Figure 6 – Appendix A.

To answer research question 3 - *What machine learning approaches are used for the prediction?* – the collected articles were grouped by the modelling technique that was applied. Artificial Neural Network comprises Graph Convolutional Network, Multilayer Perceptron (MLP), and more complex ANN. Mai, Do, Chung, Le, & Thoai (2019) employed techniques based on Recommendation Systems such as Collaborative Filtering and Matrix Factorization. WEKA is not a technique but a tool used to apply classification techniques and predictions, allowing the use of different techniques simultaneously. For that reason, a category with the articles that used WEKA was created, corresponding to 9% (Figure 2).

From Figure 2, it can be observed that, as reported by Shahiri, Husain, & Rashid (2015) - cited by Miguéis, Freitas, Garcia, & Silva (2018) - the most frequently used classification and regression techniques for student's academic success prediction are: Decision Trees (17.5%), artificial neural networks (18.6%), naive bayes (17.5%), k-nearest neighbor (KNN, 8.2%) and support vector machines (SVM, 10.3%).

Decision Trees are used, for example, by Sivasakthi (2018) to predict the introductory programming performance of first-year bachelor students in Computer Applications. Regarding Neural Networks, Dewantoro & Ardisa (2020) proposed a solution to identify first-year students at risk of failing during their studies. Naive Bayes was used by Muchuchuti, Narasimhan, & Sidume (2020) on the WEKA data mining workbench to predict the final performance of a set of 124 observations about students in Computer Systems Engineering. Regarding Support Vector Machines, Yousafzai *et al.* (2021) utilized this model to predict the student's grades from the given student performance data in 1044 records.

Comparing different studies is difficult because the features used and the approach to the selected models may vary. Most of the studies used accuracy as the chosen metric to evaluate the models performance (Adekitan & Salau, 2020; Nudelman *et al.*, 2019; Prada *et al.*, 2020; Rosado *et al.*, 2019; Sultana *et al.*, 2017) For this review, the accuracy results of each paper were grouped by modelling technique. Table 10 sums up the data extracted from the analysis of the 36 articles. The metric was not included in the table for the articles that used metrics different from accuracy and mix modelling techniques.

In the cases where articles present different accuracies for the same technique, due to feature selection, the higher accuracy was the one considered in Table 10. The higher accuracy of a Decision Tree was achieved by a random forest model employed by Miguéis *et al.* (2018) to predict the students' performance level in a dataset of approximately 7000 students. The model reached an accuracy of 96.1%, superior to the other classification techniques considered by the authors (Decision Trees, support vector machines, naive bayes, bagged trees, and boosted trees). In the same study, support vector machine model had an accuracy of 93.9%.

Regarding the KNN model, Dewantoro & Ardisa (2020) successfully predicted that 85.71% of the students are at risk of failing their studies. However, in their research, the Artificial Neural Network was the model that accomplished the best accuracy (92.85%).

The higher accuracy (92.37%) with Naïve Bayes was obtained by Rosado *et al.* (2019) in a collection of 4250 data accumulated over two academic years regarding the basic information of the Grade 7 Junior High School students. The main goals were: to identify the general average of the students when grouped according to gender; to identify who performs better between male and female; to identify the subject on which the students excel most; identify the subject on which students have difficulty; identify who performs best when the students are grouped according to the last school attended; identify the academic performance of the students based on their parent's occupation and provide predictive analysis to help the decision makers create a marketing strategy for those schools where only a few students enroled (Rosado *et al.*, 2019).

Regarding Logistic Regression, Jorda & Raqueno (2019) were able to predict the student's academic risk of dropout with an accuracy of 85.53%.

Although it was not catched by our Scopus search, Gil *et al.* (2021) also studied the prediction of academic success in a dataset similar to ours, students from ISCTE. Gil *et al.* (2021) proposed a data-driven approach to predict the success of first year students. Though analysing similar data - with the difference that our thesis focus on master student's while Gil *et al.* (2021) studied bachelor students - the approach to predict the academic success and the definition of academic success differs. Thus, this thesis aims to provide an innovative approach and methodology to predict master student's academic success.

Chapter 2 will explain the step-by-step methodology followed in this dissertation. In addition, it will include a brief description of the models and metrics adopted.

CHAPTER 2

Methodology

The present study adopted a Cross Industry Standard Process for Data Mining (CRISP-DM) methodology, which consists of an interactive process of six stages: Business Understanding, Data Understanding, Data Preparation, Modelling, Evaluation, and Deployment. The CRISP-DM cycle aims to tune the final result, *i.e.*, the capability to adequately model a problem according to evaluation metrics (Gil *et al.*, 2021). The methodology isn't restricted; some phases can be dismissed, depending on the context of the study. In research, the deployment phase is usually replaced by knowledge extraction to understand a given problem (Moro *et al.,* 2011).

All the experiments were implemented using Python. The following subsections detail the main CRISP-DM phases and the applied tasks.

2.1. Business Understanding

This study aims to understand the variability factors that impact the master student's success in ISCTE, a Portuguese university established in 1972 which currently has approximately 10000 students enroled in undergraduate and postgraduate programs.

The dataset was provided by ISCTE academic services. ISCTE comprises four schools: ISCTE Business School, the School of Sociology and Public Policy, the School of Technology and Architecture, and the School of Social Sciences and Humanities (ISCTE, 2021).

The information regarding students' candidacy, demographic information (gender, address, etc.), grades, etc., are saved on Fénix@ISCTE-IUL (Fénix) - the information system adopted by ISCTE-IUL to manage educational processes.

The data was anonymized by the Information Systems Department to ensure the students safety and ISCTE compliance with RGPD. This study defines that "success" is to finish the master's degree, *i.e.*, a student that has 120 or more European Credit Transfer and Accumulation System (ECTS) in the second academic year (last year of master) since 120 is the number of ECTS defined to graduate from the master. Thus, the primary purpose of the present analysis is to predict if a student will finish the master's degree or not (regardless of the years it takes). This study created a predictive model to classify a data record into one of two predefined classes: "Success" and "Failure".

One of the objectives of this dissertation is to understand to what extent this methodology presents a scientific contribution, which may serve as basic information in student projects to define specific study paths or student support programs that may eventually be applied to other projects. The main goal is to find the best methodology to predict student success to guarantee competitive advantage to universities that may create specific programs to increase student success which may impact the university ranking.

The application of this project on the ISCTE dataset, will be evaluated based on previously defined criteria, such as:

- Is a student's success related to the age of the student?
- Is a student's success related to the father's and mother's degrees?

2.2. Data Understanding and Data Preparation

This study considered data related to the masters between 2012 and 2022 (10 years timeframe) belonging to all 4 ISCTE schools, corresponding to 77 masters.

Initially, we had 33 variables belonging to the following seven tables: Personal Data; Candidacy Data; Statute Data; Mobility Data; Enrolment Data; Registration Data, and Final Average Data. Personal Data gathers information about the socio-demographic features, *i.e.*, student's birth year, gender, nationality, occupation, mother and father occupation, father and mother educational qualifications, ISCTE admission year, and city of residence. Candidacy data save the data concerning the features: Master's admission year, Master's id, Master's entrance grade, Prior education degree, course before Master's, year of completion of the previous degree, School/University before ISCTE and enrolment status. Statute and Mobility owned the data related to regular and mobility statutes, respectively. It features the statute's start and end date and its type. Enrolment Data saves the data regarding each curricular Course Code the student takes and has the following features: curricularCourseCode, Grade, ECTS, Semester and Status. Registration Data table keeps the features Academic year, Master Year of study and enrolment status. Lastly, Final Average Data owns the Grade and the number of approved ECTS.

Data Preparation is the process of modifying raw data so it can be further processed and analysed, including the tasks of data cleaning, data transformation, and data reduction described by Kochański (2003). Data cleaning mainly replaces missing and empty values, eliminates erroneous data and removes inconsistency. Data transformation implies the modifications that allow it to be in the form that makes its exploration possible, such as normalisation or data aggregation. Finally, data reduction involves the tasks of attribute selection, dimension reduction, discretization, numerosity reduction and aggregation (Kochański, 2003).

Figure 3 it's a representation of the most important steps followed in the data preparation and modelling.

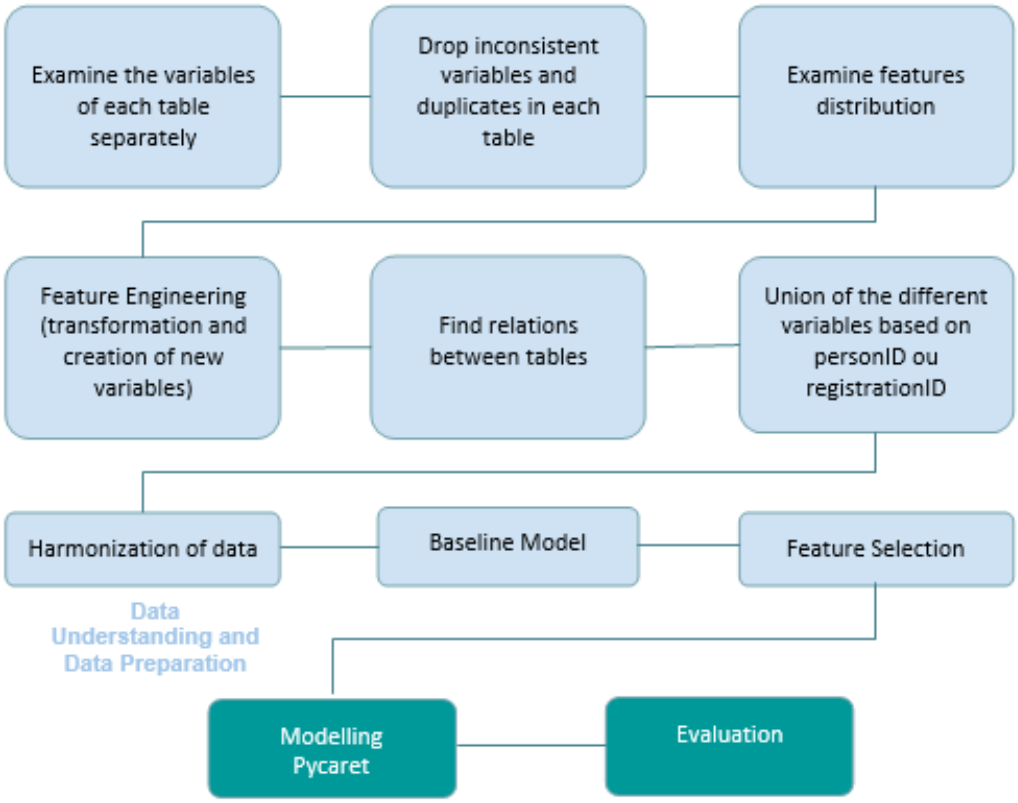

Figure 3. Schema of the methodology followed from the data preparation to modelling.

The methodology was not sequential; it was an iterative process that went back and forward, *i.e.*, for example, feature engineering and treatment of null values were performed several times during the data preparation step. Feature engineering was performed in each table separately but also in the combined dataset arising from merging multiple tables.

One of the most important steps of the methodological process was analysing how the seven tables (Personal Data; Candidacy Data; Statute Data; Mobility Data; Enrolment Data; Registration Data and Final Average Data) provided by ISCTE were related. The tables were connected through the personID and registrationID variables. The number of unique personID and registrationID values differed between tables; the ones that only existed in one table were ignored.

As represented in Figure 3, tables were examined separately and then grouped in one merged dataset based on personID and registrationID variables. The most relevant steps followed during the data preparation are listed below. The steps are divided by table/dataset where they were performed.

Personal Data table

- Remove duplicates from personal data: 835 values/rows;
- Drop features: students, father and mother occupation since more than 14000 observations had those features as null;
- Drop 1260 observations with null residence, father's and mother's educational level and student's nationality;
- Remove 12 entries that had non-sense information in the Residence city feature, *i.e.*, special characters, etc.;
- Feature transformation
    - Transformation of nationality feature in Portuguese or Other since these variables had 108 different nationalities
    - Group father, mother and student educational level in with higher studies (With HE), without higher studies (Without HE) and Unknown
        - Without:
            - Don't know how to read or write
            - Knows how to read but doesn't have 4th grade
            - Middle school
            - Elementary school - 4th grade
            - Elementary school - 6th grade
            - Elementary school - 9th grade
            - High school - 12th grade
            - Higher professional technical course diploma
        - With:
            - Technological specialisation course - post higher school
            - Higher education - bachelor (Bologna)
            - Higher education - Integrated master
            - Higher education - bachelor (pre-Bologna)
            - Higher education - bachelor
            - Postgraduate - 2º cycle (Bologna)
            - Postgraduate - PhD (Bologna)
            - Postgraduate - Master (pre-Bologna)
            - Postgraduate - PhD (pre-Bologna)

- Group city of residence in Residence feature. For that purpose, six dictionaries were created to address all the cases, 992 cities were grouped in 7 categories:
  - Lisbon: all the cities that belong to Lisbon metropolitan region
  - Portuguese islands: cities that belong to Azores or Madeira islands
  - Portugal Continental: Portuguese cities that aren't part of Lisbon metropolitan region or Azores or Madeira islands
  - Africa: African cities
  - Asia: Asian cities
  - China: Chinese cities
  - Other: cities that don't belong to the other groups, for example, in Mexico.

Mobility and Statute tables

- Concatenation between mobility and statute table so that the features could be examined together - the creation of statutes table;
- Remove duplicates from the statutes data table: 17 values/rows;
- Feature Engineering
  - Creation of Statute, Mobility, and Covid categorical features based on statutes and mobility tables, the variables assume the values of 'Yes' and 'No'':
    - If the student has a mobility statute (Erasmus+ and other International Cooperation Agreements - studies, Double Degree Programme, Protocol/Partnership with a foreign institution, Erasmus, Erasmus+ Programme - training (internship), Ibero-America Santander Grants), the feature mobility is settled to 'Yes' otherwise is settled to 'No';
    - If the student doesn't have a statute different from mobility or covid statute (Student worker, SAS Scholarship, International Student, Part-time Student, PALOP International Student - Master, Class Delegate, Class Deputy, FCT fellow, AEISCTE-IUL athlete, Master's student from IBS degree, Young Association Leader, Group of risk, Student with Temporary Disability, Special educational needs, Pregnant / Parents with children aged < or = 3 years, Online Student, Top 15 IBS, Monitor, Military, Firefighter, Student who professes religious confession, High-Performance Sportsman, Death of Spouse or Relative), the feature statute is settled to 'Yes' otherwise is settled to 'No';
    - If the student has a covid statute (temporary disability - Covid-19), the feature covid is settled to 'Yes'; otherwise is settled to 'No';

- For students with more than one statute, only the most recent was considered, *i.e.*, if a student had a Student worker statute in the first semester of 2020/2021 but had the covid statute after that, only the Covid feature is settled to 'Yes'; features Mobility and Statute are settled to 'No';

Enrolment Data

- Creation of no. failed courses feature based on the sum of courses with enrolment status equal to failed;
- The other variables (curricularCourseCode, Grade, ECT) were ignored since they were out of the study goals of this project which didn't include the study of the impact of each curricular Course on the Student's success;

Registration data and final average tables

- Concatenation between Registration data and final average tables based on RegistrationId - the creation of "regist" table;
- For students with more than one registration id, *i.e.*, that registered in more than one master, the last register was the one considered;
- Drop 140 observations that had enrolment status equal to zero;
- Drop 2 observations that had Master Year of Study equal to zero;

Combined datasets

- Remove 142 observations that didn't have registration data;
- Drop 14 rows that had mismatched information between candidacy and registration data about enrolment status;
- Drop 94 observations that only existed on the personal data table;
- Delete the 1298 observations that didn't exist in the personal data table;
- Creation of feature master years which is a measure of the years the master took (results from the difference between the year under analysis and the master's admission year);
- Drop 52 rows that had candidacy id equal to zero
- Drop one row that had the year of birth equal to 2011, which is non-sense
- Drop 10 duplicated rows
- Creation of an Admission age feature based on the difference between the master's admission year and the year of birth
- Drop one row that had the year of birth equal to 2011, which is non-sense
- Drop course before Master's variable due to its dispersion (more than 3000 distinct values)
- Drop 3 rows that corresponded to a non-sense year of the last degree completion values

- Creation of feature gap between degrees based on the variables year of the last degree completion and master's enrolment year
- Drop 14 rows that belong to students that had a non-sense gap between degrees values
- Drop 16 rows that lack Previous school/university data
- Transformation of School/university before ISCTE values into ISCTE or Other
- Creation of an Abandon feature for students that had the following enrolment status: annulled, withdrawal request, prescribed, withdrawal, Temporary interruption of studies, internal abandonment
- Select only the registers belonging to the Master Year of study equal to two (the last year of the master's degree)
- Creation of a Target feature based on the number of Approved ECTS, students with 120 or more ECTS are considered successful, while students with fewer are considered failed.
- Drop Master's admission grade since the criteria differ between masters
- Drop the number of Approved ECTS variable since it is highly correlated with the target feature;
- Drop birth date since it's highly correlated with the gap between degrees feature.

Besides all the decisions mentioned above, several features showed differences between uppercase/lowercase, miswritten word or/and words with empty spaces or special characters fixed and harmonised. For example, Agua Grande city, in St. Time, was written in three different ways - Agua Grande, Água Grande, and Água-Grande.

The final step was the feature selection which improves model performance and reduces training time since fewer data is being examined. For this purpose, correlations between variables were studied, and the highly correlated variables were removed. The remaining part of feature selection was done in the modelling step. Table 1 summarizes the features obtained from data preparation. All the variables that suffered a transformation were considered derived. The variables that remain the same as the ones provided in the tables were considered extracted.

The final dataset comprises a total of 12418 records belonging to 71 master degrees between 2012 and 2022. Regarding the categorical features: 53.52% of the students didn't have a statute, and 56.35% had fathers with higher education. Considering the residence, 74.46% of the students are from Lisbon, and 19.41% are from other parts of continental Portugal. The average grade of the students is 15.25, with a standard deviation of 1.38 points. The master's duration was, on average, 1.29 years with a standard deviation of 0.93 years; the average number of failed courses of the students is 0.29, with a standard deviation of 0.98, and the gap between degrees it's 2.89 years, with a standard deviation of 5.25 years. Finally, 59.40% of the students were considered successful (finished the master's), and 40.60% were not.

Table 1. Variables selected after data preparation. Features highlighted in blue were excluded due to high correlations with other features. Features highlighted in grey were used as input for the baseline model.

Variables	Description	Source Table	Origin	Classes/Distribution
Master id	Id of the master	Personal Data	Extracted	B113; B16; B55; B4; B24; B01; 0219; B103; 039; B104; B27; B114; 0186; B42; B119; 064; 013; 6421; B12; B15; 0115; B107; B30; 0239; 0177; 012; B110; 079; 0210; B31; B34; B68; B66; B008; B72; B54; 0117; B121; B76; B23; 027; 0116; B57; B115; 0321; 0218; 081; 0176; 0277; 014; 0127
Prior education degree	Prior level of studies that the student had before applying to the master's (*e.g.* high school, other master, bachelor, etc)	Candidacy Data	Derived	With Higher Education (HE), Without HE
Enrolment status	Master's enrolment status; if the student completed all the courses; quit the master etc.	Registration Data	Extracted	Complete; Withdrawal; Active; Awaiting final clearance; Transited; Academic part concluded
Mobility	Describes if the student made was part of a mobility program (*e.g.,* Erasmus)	Student Mobility Data	Derived	No; Yes
Statute	Describes if the student had one special statute (*e.g.,* military, working student, special needs)	Student Statute Data	Derived	No; Yes
Covid	Describes if the student had covid while attending the	Student Statute Data	Derived	No; Yes

	master.			
No. failed courses	Measures the number of curricular units that the student failed	Enrolment Data	Derived	-
Academic year under analysis	Academics' year that is related to the master average grade we are considering	Registration Data	Derived	2012/2013; 2013/2014; 2014/2015; 2015/2016; 2016/2017; 2017/2018; 2018/2019; 2019/2020; 2020/2021; 2021/2022
Average grade	The average grade that the student has in the second curricular year (last year of the master's)	Final Average Data	Extracted	-
Gender	Student gender	Personal Data	Extracted	Female; Male
Nationality	Student's nationality	Personal Data	Derived	Portuguese; Other
Father educational qualification (Father EQ)	Measures the father's education level (master, bachelor, high school, etc.)	Personal Data	Derived	With Higher Education (HE); Without HE
Mother educational qualifications (Mother EQ)	Measures the mother's education level (master, bachelor, high school, etc.)	Personal Data	Derived	With Higher Education (HE); Without HE
Residence	Residence of the student	Personal Data	Derived	Lisbon; Continental Portugal; PT islands (Azores and Madeira); Brazil; Europe; Africa; Asia; China; Other

Master’s admission year	Year of Master's admission	Candidacy Data	Extracted	-
Admission age	Student's age when the master began	Personal Data	Derived	-
Gap between degrees	Number of years between the year when the student took the last degree and the start of this master	Candidacy Data	Derived	-
Previous school/uni	Previous school or university that the master attend	Candidacy Data	Derived	ISCTE; Other
Abandon	Describes if the students quit the master	Registration Data	Derived	No; Yes
Target	Measures the success of the student. Students with success had 120 ECTS, and failure students had less than 120	Final Average Data	Derived	Failure; Success
Master years	Measures of the years that the master's last (results from the difference between the year under analysis and the master’s admission year);			

2.3. Modelling and Evaluation

As previously mentioned, the last part of feature selection was included in the modelling stage. The variables were used as input for creating a Random Forest baseline model, which returned as output the feature importance, and the optimal number of features for the data (9, in this case). The model considered that the most important features were: Master's admission year, Master id, Statute, No. failed courses, Average grade, Master years, Father EQ, Residence, and Gap between degrees. Thus, a total of 9 features are represented in the final dataset, 2 socio-demographic features (Father EQ, Residence) and 7 education path features (Master's admission year). The baseline model was created to choose the input variables for PyCaret.

For this thesis, PyCaret - an open-source Python library - was used in order to compare several classification algorithms. PyCaret automates machine learning workflows and allows the training and comparison of several machine learning models simultaneously. In addition, PyCaret verifies if the algorithms need hot encoding data and automatically transforms the data ((Ali & Moreno, 2022a)). Another ability of PyCaret is that choose a train/test split of 70/30 automatically. In order to demonstrate the use of the predict_model() function on unseen data and to infer the model's performance on new data, a sample of 10% of the records was withheld from the original dataset to be used for predictions at the end (validation set).

Setup() function initializes the training environment, automatically deduces the data types and extracts information for all the features (for example, the existence of duplicates and null values). In the setup() function, the mandatory arguments are the target variable (Target feature - in our case) and the dataset; thus, PyCaret's inference algorithm understands if there is a classification or a regression problem. After setup, the compare_model() function is called to train and evaluate the performance of all the estimators available in the model library using 10-fold cross-validation. The function’s outputs print a scoring grid that shows average Accuracy, Area Under the Curve (AUC), Recall, Precision, and F1-scores, along with training times (TT). The metrics are computed based on the value of the following rates:

- True positives (TP): proportion of positive cases that are correctly identified.
- False Positives (FP): proportion of positive cases that are incorrectly identified.
- False Negatives (FN): proportion of negative cases that are incorrectly identified.
- True Negatives (TN): proportion of negative cases that are correctly identified.

Accuracy measures the overall effectiveness of the algorithm and answers the question: How many students did we correctly label out of all students? It is defined by the following formula: (TP+TN)/(TP+FP+FN+TN). Recall gives the proportion of positive cases that are correctly identified, that is: TP/(TP+FN). Precision is the ratio of correctly labelled data to all the labelled data; answers to *How many of those we labelled as successes are actually successes?* It is given by: TP/(TP+FP). Recall/Sensitivity is the true positive (TP) rate which is the probability of detecting a true outcome, and is defined by: TP/(TP+FN). F1-score considers precision and recall, and it's given by 2*(Recall * Precision) / (Recall + Precision). Lastly, AUC is a classification performance measure that indicates the model's accuracy. It measures how much capable of distinguishing between classes, the model is. The higher the AUC, the better the model predicts the classes (Yaacob *et al.*, 2019).

For this thesis, five models were applied: Light Gradient Boosting Machine Classifier (LGBMC), Gradient Boosting Classifier (GBC), Random Forest (RF), Adaptive Boosting Classifier (ADA), and Decision Tree (DT).

Gradient boosting describes a class of ensemble machine learning algorithms for classification or regression predictive modelling problems. Ensemble techniques are divided into Bagging and Boosting.

Boosting is an ensemble method that does not make the predictors separately but sequentially, thus, the following predictors learn from the prior predictors' errors (Pandey *et al.*, 2020). Trees are added one at a time to the ensemble and fit to correct the prediction errors made by prior models. Light Gradient Boosting Machine, Gradient Boosting Classifier, and Ada Boost Classifier are examples of boosting algorithms.

LGBM uses tree-based learning algorithms, and it's designed to be distributed and efficient with faster training speed and higher efficiency once it supports parallel and GPU learning. LightGBM grows trees vertically, while other tree-based learning algorithms grow trees horizontally. It means that LightGBM grows tree leaf-wise while other algorithms grow level-wise (Zhang, 2022).

Adaptive Boosting Classifier was the first successful boosting algorithm to be created, and it is commonly used for binary classification. Using very short (one-level) Decision Trees as weak learners that are added sequentially to the ensemble results in an improved performance compared to the classification by one single tree or another tree base-learner. ADA minimises the exponential loss function, which can make the algorithm sensitive to outliers. (Shanbehzadeh *et al.*, 2022).

Gradient Boosting Classifier it's a generic algorithm that is more flexible than ADA once any differentiable loss function can be utilised. This method trains the learners and depends on reducing the loss functions of that weak learner by training the residual of the model (Stankoski *et al.*, 2019).

Bagging or boosting aggregation combines multiple ensemble learners by varying the training dataset. Instead of training a model on the entire dataset, bagging creates several weak learners trained on a subset of the original dataset. Random Forest is an example of a bagging ensemble.

Random Forest creates multiple Decision Trees that are trained on different and independent subsets of the data. Once numerous trees have been constructed, it takes the prediction from each tree and returns the final output based on the class chosen by the majority of the trees. Training time is relatively low compared to other algorithms (Nudelman *et al.*, 2019).

Decision Trees are one of the most popular classification techniques once they are easy to understand, they can be applied to different types of attributes, nominal or numerical, and can classify new examples fast. Decision Trees represent a group of classification rules in a tree form, and each rule's result can be denominated by a node. Nodes are recursively divided into descendants. The model prediction is given by a node with no descendants. Thus, the higher an attribute appears in the tree, the more influential it is for data division (Al-Barrak & Al-Razgan, 2016; Nudelman *et al.*, 2019).

CHAPTER 3

Results and Discussion

3.1. Input Models

The PyCaret library has eighteen classifiers. Based on the initial analyses performed by the setup() function, 14 classifiers were chosen to be trained. The resulting metrics are summed up in Table 2. The models were trained to optimize the accuracy metric.

Table 2. Classification models returned by compare_models PyCaret function (baseline models), only the 9 most important features were used to train the models.

Model		Accuracy	AUC	Recall	Prec.	F1	TT(Sec)
lightgbm	Light Gradient Boosting Machine	**0.8737**	**0.9402**	**0.8582**	**0.8330**	**0.8453**	**0.188**
gbc	Gradient Boosting Classifier	0.8511	0.9263	0.8219	0.8107	0.8161	1.092
ada	Ada Boost Classifier	0.8337	0.9128	0.8168	0.7803	0.7980	0.435
rf	Random Forest Classifier	0.8318	0.9131	0.7739	0.8010	0.7871	2.732
dt	Decision Tree Classifier	0.8250	0.8229	0.7863	0.7806	0.7833	0.070
et	Extra Trees Classifier	0.7873	0.8681	0.7313	0.7375	0.7343	3.149
knn	K Neighbors Classifier	0.7743	0.8433	0.7440	0.7091	0.7260	1.152
lr	Logistic Regression	0.7534	0.8323	0.7090	0.6875	0.6979	1.411
lda	Linear Discriminant Analysis	0.7516	0.8272	0.7249	0.6791	0.7011	0.266
ridge	Ridge Classifier	0.7515	0.0000	0.7186	0.6811	0.6991	0.073
svm	SVM - Linear Kernel	0.7212	0.0000	0.6095	0.7062	0.6117	0.289
dummy	Dummy Classifier	0.5980	0.5000	0.0000	0.0000	0.0000	0.014
nb	Naive Bayes	0.5701	0.7077	0.9564	0.4826	0.6415	0.030
qda	Quadratic Discriminant Analysis	0.5566	0.4995	0.2083	0.3726	0.2402	0.126

To understand if and how the number of features influenced the chosen models and their performance, compare_models() was also performed on the dataset with all the features that arise from the data preparation step (Master's admission year, Master id, Prior education degree, Mobility, Statute, Covid, No. failed courses, Average grade, Master years, Gender, Nationality, Father EQ, Residence, Gap between degrees, Previous school/uni, Target).

Table 3. Classification models returned by compare_models PyCaret function (baseline models), all features selected in the end of the data preparation step.

Model		Accuracy	AUC	Recall	Prec.	F1	TT(Sec)
lightgbm	Light Gradient Boosting Machine	**0.8832**	**0.9508**	**0.8567**	**0.8521**	**0.8542**	**0.196**
gbc	Gradient Boosting Classifier	0.8557	0.9339	0.8302	0.8131	0.8213	1.473
rf	Random Forest Classifier	0.8483	0.9265	0.7890	0.8241	0.8059	2.364
ada	Ada Boost Classifier	0.8323	0.9181	0.8161	0.7767	0.7954	0.512
dt	Decision Tree Classifier	0.8296	0.8248	0.7976	0.7812	0.7890	0.080
et	Extra Trees Classifier	0.7978	0.8848	0.7439	0.7493	0.7463	3.090
knn	K Neighbors Classifier	0.7782	0.8470	0.7458	0.7129	0.7289	1.296
lr	Logistic Regression	0.7583	0.8401	0.7103	0.6931	0.7014	2.020
ridge	Ridge Classifier	0.7538	0.0000	0.7183	0.6828	0.6999	0.076
lda	Linear Discriminant Analysis	0.7530	0.8343	0.7218	0.6804	0.7003	0.265
svm	SVM - Linear Kernel	0.7325	0.0000	0.6040	0.7279	0.6307	0.353
dummy	Dummy Classifier	0.6003	0.5000	0.0000	0.0000	0.0000	0.016
qda	Quadratic Discriminant Analysis	0.5793	0.5064	0.1430	0.4191	0.2003	0.136
nb	Naive Bayes	0.5636	0.6654	0.9668	0.4774	0.6391	0.035

By comparing Table 2 and Table 3, it's possible to verify that an increased number of features does not reflect a significant accuracy increase (0.95%), thus not reimbursing the increased training time and computer power needed in the process. Therefore, the further analysis only considers the 9 most significant features previously mentioned.

3.2. Tune Models with hyperparameters selection

Based on the accuracy metric, the Light Gradient Boosting Machine, Gradient Boosting Classifier, Random Forest, Ada Boost Classifier, and Decision Tree models, the five with the highest accuracy, were chosen to be further analysed and tuned (Table 2). The five models were firstly automatically tuned by PyCaret tune_model function. By default, as previously mentioned, the function will optimize the Accuracy and evaluates the performance of each configuration using a 10-fold cross-validation. Furthermore, for each model, a set of hyperparameters were automatically tuned. The list of hyperparameters used for each model is available in Table 11. All the tuned models were set to automatically choose the better model, *i.e.*, PyCaret has an option (choose_better = True) that guarantees that the better performing model will be returned, meaning that if hyperparameter tuning doesn't improve the performance, it will return the input model instead (Ali & Moreno, 2022b). A simple description of the tuned parameters follows:

- Learning Rate - Defines how much to change the model in response to the estimated error each time the model weights are updated;
- Max_depth - The maximum depth of the tree;
- n_estimators - The number of trees in the forest;
- Max_features - The number of features to consider when looking for the best split;
- Min_samples_split - The minimum number of samples required to split an internal node of a tree;
- Min_samples_leaf - The minimum number of samples required at a leaf node.
- Bootstrap - Whether bootstrap samples are used when building trees, *i.e.*, taking a sample of a population by drawing with replacement. If False, the whole dataset is used to build each tree.
- Max_leaf_nodes – Number of leaves a node can have

The comparison of the model's performance before and after the tuning for each algorithm is summed up in the following tables (from Table 4 to Table 8).

Table 4. Comparison of the three PyCaret LGBMC models, before and after auto-tuning and custom tuning.

Model	Accuracy	AUC	Recall	Prec.	F1
LGBMC	0.8737	0.9402	0.8582	0.8330	0.8453
LGBMC_tuned	0.8649	0.9341	0.8547	0.8180	0.8357
LGBMC_custom_tuned	0.8749	0.9384	0.8630	0.8323	0.8472

Table 5. Comparison of the three PyCaret GBC models, before and after auto-tuning and custom tuning.

Model	Accuracy	AUC	Recall	Prec.	F1
GBC	0.8512	0.9263	0.8223	0.8108	0.8163
GBC_tuned	0.8736	0.9393	0.8499	0.8382	0.8439
GBC_custom_tuned	0.8512	0.9261	0.8213	0.8112	0.8161

Table 6. Comparison of the three PyCaret RF models, before and after auto-tuning and custom tuning.

Model	Accuracy	AUC	Recall	Prec.	F1
RF	0.8346	0.9159	0.7758	0.8057	0.7903
RF_tuned	0.8596	0.9294	0.8569	0.8067	0.8308
RF_custom_tuned	0.8153	0.8985	0.7199	0.8012	0.7581

Table 7. Comparison of the three PyCaret ADA models, before and after auto-tuning and custom tuning.

Model	Accuracy	AUC	Recall	Prec.	F1
ADA	0.8337	0.9128	0.8168	0.7803	0.7980
ADA_tuned	0.8455	0.9234	0.8258	0.7975	0.8112
ADA_custom_tuned	0.8483	0.9237	0.8273	0.8019	0.8142

For the Decision Tree, besides auto-tune and tuning with defined hyperparameters, ensemble models were applied through ensemble_model() PyCaret function. As previously mentioned, two techniques exist for ensembling: boosting and bagging. Both techniques were applied, and the boosting ensemble corresponds to an AdaBoostClassifier applied to the Decision Tree.

Table 8. Comparison of the three PyCaret Decision Tree models, before and after auto-tuning and custom tuning (custom tune and bagging method).

Model	Accuracy	AUC	Recall	Prec.	F1
DT	0.8269	0.8241	0.7854	0.7848	0.7849
DT_tuned	0.8439	0.9047	0.8305	0.7923	0.8107
DT_custom_tuned	0.8498	0.9116	0.8302	0.8035	0.8165
DT bagging	0.8559	0.9290	0.8238	0.8191	0.8213

DT boosted	0.8314	0.8963	0.7889	0.7913	0.7899

For LGBMC and ADA models, the tuning with hyperparameters improved the model's accuracy, AUC, Recall, Precision, and F1-score. Regarding the LGBMC model, the hyperparameters that differed from the auto-tuned and input models were the learning_rate=0.05 (lower than the auto-tuned and input models = 0.1) and the max_depth=6 (higher than the auto-tuned and input models = -1). Compared with the other ADA models, the ADA custom tuned had 500 estimators, while the other only used 200.

On the other hand, Table 5 and 6, show that for the GBC and RF, auto-tuned models performed better than the input and tuned with defined hyperparameters. The hyperparameters that made a difference in auto-tune models were max_depth=5 and n_estimators=200 while for RF, were max_depth=11,max_features=1.0, min_samples_split=10 and n_estimators=140.

Decision Tree performance improved with custom hyperparameter tuning and bagging ensemble.

The best models within each algorithm type were then applied to test and validation sets, using the predict_model PyCaret function, reminding that the validation set was created to examine the model performance with unseen data. The results are displayed in Table 9.

Table 9. Comparison between metrics in training, test and validation sets for the best models.

Model	Accuracy	AUC	Recall	Prec.	F1
LGBMC_custom_tuned (Training)	**0.8749**	0.9384	**0.8630**	0.8323	**0.8472**
LGBMC_custom_tuned (Test)	0.8732	0.945	0.8486	0.8443	0.8464
LGBMC_custom_tuned (Validation)	0.8833	0.8814	0.8704	0.8523	0.8612
GBC_tuned (Training)	0.8736	**0.9393**	0.8499	**0.8382**	0.8439
GBC_tuned (Test)	0.8795	0.9473	0.8464	0.8588	0.8526
GBC_tuned (Validation)	0.8849	0.8808	0.8569	0.86520	0.8610
RF_tuned (Training)	0.8596	0.9294	0.8569	0.806700	0.8308
RF_tuned (Test)	0.8673	0.9369	0.8681	0.82	0.8434
RF_tuned (Validation)	0.8800	0.8800	0.8801	0.8395	0.8593
ADA_custom_tuned (Training)	0.8483	0.9237	0.8273	0.8019	0.8142
ADA_custom_tuned (Test)	0.8467	0.9289	0.8312	0.8032	0.817

ADA_custom_tuned (Validation)	0.8559	0.8507	0.8201	0.8314	0.8257
DT_custom_tuned (Training)	0.8498	0.9116	0.8302	0.8035	0.8165
DT_custom_tuned (Test)	0.8527	0.9174	0.8333	0.8133	0.8232
DT_custom_tuned (Validation)	0.8543	0.8519	0.8375	0.817	0.8271
DT bagging (Training)	0.8559	0.9290	0.8238	0.8191	0.8213
DT bagging (Test)	0.8628	0.9346	0.8261	0.8382	0.8321
DT bagging (Validation)	0.8768	0.8728	0.8491	0.8541	0.8516

As shown in Table 9, all the models perform well on test sets since the difference between the training and test accuracies is very low, which can be a sign that overfitting was prevented. The same its' found for the validation sets once the AUC value is similar between the training, test, and validation sets. Overall, the accuracy is always slightly higher in the test set, but that may be related to the sample data present in the set and the sample size which is lower than in the training set.

3.3. Knowledge Extraction

The main goal of this thesis was to identify the best model to predict the master students' success. From the literature review, it was observed that the most frequently used classification and regression techniques for student's academic success prediction are: Decision Trees (17.5%), artificial neural networks (18.6%), naive bayes (17.5%), k-nearest neighbor (KNN, 8.2%) and support vector machines (SVM, 10.3%). Our results from Table 2 and Table 3 show that the Decision Tree algorithms outperforms the other techniques which can be the reason why there are the most adopted.
Comparing all the models after the tuning (Table 9), LGBMC remains the model with better accuracy, recall, and F1-score; thus the best fitted model to our data.

Figure 7 – Appendix C shows the confusion matrix for all the models within each category that performed best. Analysing precision, AUC, recall, and F1, in addition to accuracy, it's important once accuracy it's more suitable to assess balanced datasets (Nabil *et al.*, 2021). In this case, we are working with an imbalance dataset, 59.40% of the students were considered successful (finished the master), and 40.60% were not; thus, the other metrics give a more faithful insight. The F1-score is useful for different class distributions once it is considered the average value between recall and precision (Nabil *et al.*, 2021). Based on the accuracy and F1 metrics, the LGBMC model outperforms the other techniques in practically all explored performance metrics.

Nabil *et al.* (2021), while studying the prediction of Students' Academic Performance, given their grades in the previous courses of the first academic year, found that gradient boosting outperformed deep neural network (DNN), Decision Tree, Random Forest, Logistic Regression, Support Vector Classifier, and K-Nearest Neighbor with an accuracy of 89%.

Miguéis *et al.* (2018) compare several models to predict the students' performance level from a European Engineering School of a public research University. The author reported that random forests were superior to the other classification techniques considered in the analysis (decision trees, support vector machines, naive bayes, bagged trees, and boosted trees), presenting an accuracy of 96.1%. Our results contrast with the ones reported by Miguéis *et al.* (2018), since the boosted trees show higher performance than random forests. However, we should keep in mind that comparing the results is not linear, since the considered features (Socio-economic status, High school background, Enrolment process, First year assessment, First year performance, Socio-demographic) and dataset size (2459 observations) are very distinct.

Muchuchuti *et al.*, (2020) reported an accuracy of 53.54% with an ADA classifier while studying five classifiers for student performance amelioration prediction on a dataset of 124-observation. Our results show a much higher accuracy; the authors used Academic progression data and the final degree classification as features, while our study used a broad range of features that influenced the results.

One of the main goals of this thesis was to identify the drivers of student success and to find a model that could predict that success. The target variable was divided into Success and Failure, encoding the labelled success as 1 and the failure as 0. To analyse the feature's importance and the reasons behind the model prediction; decision boundary, feature importance and SHAP plots were created for the models with those options. Plots are displayed in Figure 4, Figure 5, and Figure 8 (Appendix C).

According to Figure 4, which displays the feature importance plots for all the algorithms, it is found that Average grade is the most important and has the most significant influence on all the algorithms being the most relevant variable for student success prediction. The gap between degrees and master years is in the top four of the most important features of all five models. Thus, we can consider those the most important success drivers for students. For RF, GBC, and DT master admission year of 2020/2021 is the third most important feature. In line with our results, Gil (2019), while studying the drivers for academic success, found that the gap between degrees and admission year is among the most important factors for academic success. Feature importance plots give the more important features to predict the student's success but do not provide information if they impact the prediction of success or failure, *i.e.*, master years – master durability - could impact the success or the failure.

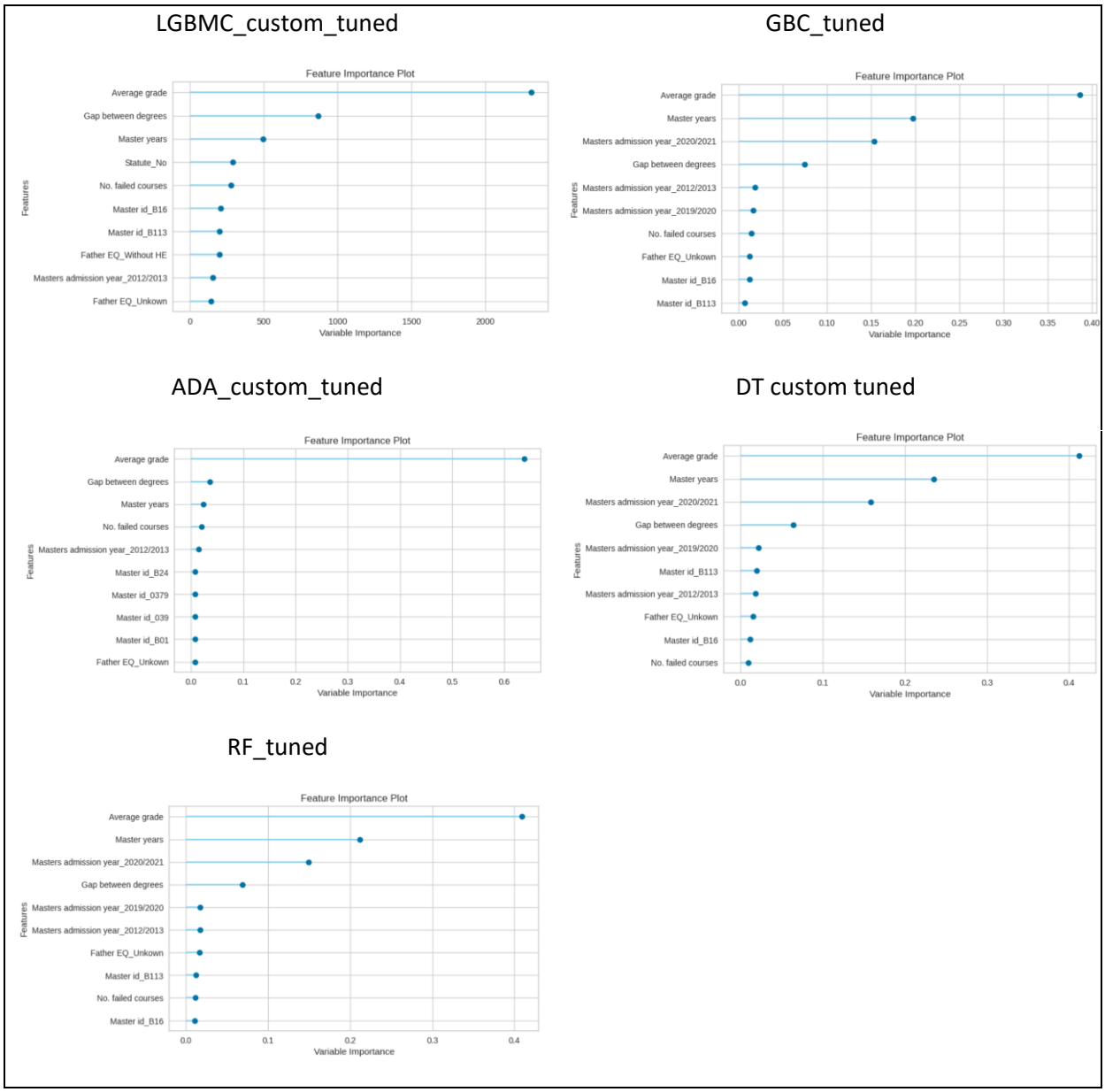

Figure 4. Feature Importance given by LGBMC, GBC, ADA, DT and RF models (tuned or custom tuned, the plot concerns the model with highest performance within each algorithm type).

Decision boundary plots are used as a diagnostic tool to visualize the classification of the data-points in Feature Space. The Decision Boundary plot contains a Scatter Plot within which contains the data-points belonging to different classes (denoted by colour and shape). Usually, there is a single-line decision boundary, *i.e.*, one line that divides the feature space into two subspaces. In our case, that division doesn't exist (Figure 8), which highlights the complexity of the algorithms used and how hard is to understand the motives behind the black-box models.

The Decision Tree is the most understandable model of the five selected, not being a black-box. With that in mind, the tree created by the model was plotted and analysed. Figure 9 and Figure 10 (Appendix C) show the Decision Tree created by DT_custom_tuned model. The root node cuts the tree based on the master admission year of 2020/2021, classifying 4678 samples as failures and 3145 as successes. Observations with less than 0.5 in master admission year of 2020/2021 (reminding that categorial features suffered hot encoding) were automatically identified as a failure. The tree has 100 leaf nodes, a depth of 12, and two main branches. The two main branches were created based on the Average grade being equal to or lower than 14.975, where 2843 samples were classified as failure and 3143 as success.

SHapley Additive exPlanations (SHAP) is a game theoretic approach to explain the output of any machine learning model. In PyCaret, SHAP can show the global contribution by using the feature importance and only supports tree-based models for binary classification: RF, DT, LGBM.

All variables are shown in the order of global feature importance, the first being the most important and the last being the least important. Features with values near the axis have a low impact on the model decision; failure corresponds to 0 and success to 1; thus, in the plot, low indicates failure, and high indicates success. From Figure 5, we can extract that:

- For RF and DT: the average grade of a student has the strongest effect on whether that student has success or not;
- As the average grade increases, the student is more likely to have success;
- For the three models: as the master years increase, *i.e.*, the time that a student takes to finish the master, the student is more likely to have success;
- For the three models: as the gap between degrees increases, the student is less likely to have success;
- For LGBMC model: the master admission year of 2020/2021 has the most substantial effect on prediction, and as this feature increases, the student is less likely to have success.

The information obtained from the SHAP plot is in line with the ones provided by feature importance plots, as master years, admission years, average grades, and the gap between degrees was selected as the most important features. In addition, SHAP plots show many features related with master id's, suggesting that this can also be a feature to consider.

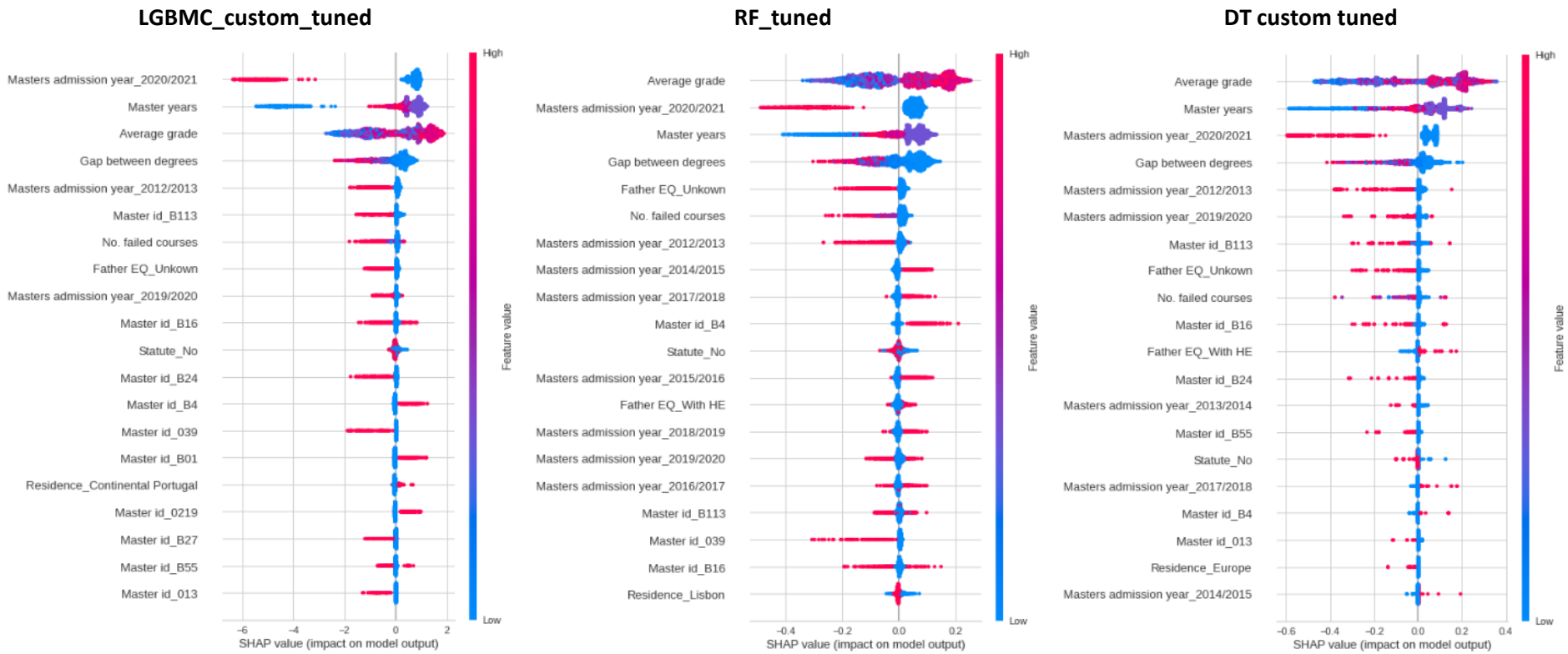

Figure 5. SHAP plot for LGBMC, RF and DT models (tuned or custom tuned, the plot concerns the model with the highest performance within each algorithm type).

Summing up the results: Light Gradient Boosting Machine Classifier had the best performance, followed by Gradient Boosting Classifier, Adaptive Boosting Classifier, Random Forest, and Decision Trees. The feature importance analysis identified average grades, the gap between degrees and master years as the essential features. The methodology developed allowed the prediction of student success with an accuracy close to 88%.

Besides the promising results, this study had some limitations. The following topics are a summary of those limitations as well as suggestions that should be used in future studies:

- A huge amount of time was spent in harmonizing data due to miswritten words or/and words with empty spaces or special characters;
- Some variables were discarded due to the massive variability of values; others had to be grouped in much smaller groups which may lead to a loss of information. This could be fixed if ISCTE changes the way students register their data, *i.e.*, instead of the student writing, for example, the specific bachelor's name, it should have a limit amount of options;
- It wasn't possible to discover the match between the master id's and the master real names, thus limiting the information that could be extracted from this variable. In future studies, it would be interesting to group students based on master areas and see the impacts on the success prediction;
- This study only encompassed variables that did not include any feedback from students or teachers; in future studies could be interesting to study the influence of non-cognitive features such as time management, leadership, or extracurricular activities;
- Student success was described as the ability of the student to finish the master degree; thus, taking into consideration the total ECTS of the master, future studies should apply the same methodology to ECTS related only to the completion of the master's first year.

Conclusions and Future Work

A success indicator of an educational institution is students' learning outcomes, which could be positive, related to high GPAS and high rate of graduated students, or negative with a high rate of dropouts or long study periods. Therefore, one of the most important duties of educational organizations and administrators is to improve student success (Karpicke & Murphy, 1996, cited by (Altun *et al.*, 2022).

Educational Data Mining (EDM) is applying data mining tools and techniques to analyse the data at educational institutions (Al-Mahmoud & Al-Razgan, 2015). Educational institutions use educational data mining to gain deep and thorough knowledge to enhance the assessment, evaluation, planning, and decision-making in their educational programs.

Understanding the drivers of student success may assist educators in developing pedagogical methods providing a tool for personalized feedback and advice. Predicting student performance and the factors that impact that performance may help organizations to create different target actions considering the types of students, which may also result in a more efficient allocation of the institutions' resources (Miguéis *et al.*, 2018).

This study aimed to develop a data-driven methodology, understand the variability factors that impact the master student's success, and identify the most suited algorithm to predict the student's success. We adopted a Cross Industry Standard Process for Data Mining methodology, which consists of an interactive process of six stages: Business Understanding, Data Understanding, Data Preparation, Modelling, Evaluation, and Deployment. Data belongs to ISCTE master students, a Portuguese university, during 2012 to 2022 academic years. The PyCaret library was used to compare the performance of several algorithms. The factors that influence the success include the student's gender, previous educational background, the existence of a special statute, and the parents' educational degree.

From the initial PyCaret setup, Light Gradient Boosting Machine Classifier had the best accuracy (87.37%), followed by Gradient Boosting Classifier (85.11%), Adaptive Boosting Classifier (83.37%), Random Forest (83.18%), and Decision Tree (82.5%). The five models were then tuned, and although the performance increased for all the models, the order of the models' performance remained the same: LGBMC (87.49%) followed by GBC (87.36%), ADA (84.83%), RF (85.96) and DT (84.98%). Two ensemble techniques, bagging and boosting, were applied to Decision Tree models where bagging was revealed to enhance model performance (85.59%) compared to the input model (84.98%) and the boosting model (83.14%).

In order to understand the drivers for student success - feature importance, decision boundary, and SHAP plots - were created. The analysis of feature importance plots revealed that Average grade is the most important and has the greatest influence on all the algorithms, being the most relevant variable for student's success prediction. The gap between degrees and master years is in the top four of the most important features of all five models; thus, we can consider that those are the most important success drivers for students. For RF, GBC, and DT master admission year of 2020/2021 is the third most important feature. In line with our results, Gil (2019) while studying the drivers for academic success, found that the gap between degrees and admission year is in the list of the most important factors for academic success.

Feature importance plots give the more important features to predict the student's success but do not provide information if impact the prediction of success or failure, *i.e.*, master years – master durability - could impact the success or the failure. For that purpose, SHAP plots were computed for LGBMC, RF, and DT models. The plots exposed that: the average grade of a student has the strongest effect on whether that student has success or not; as the average grade increases, the student is more likely to have success; as the master years increase, *i.e.*, the time that student takes to finish the master, the student is more likely to have success; as the gap between degrees increase, the student is less likely to have success; master admission year of 2020/2021 has the most substantial effect on the prediction of LGBMC model, and as this feature increase, the student is less likely to have success.

The present thesis developed a data-driven methodology to predict ISCTE master students' success and unveil the drivers of this success. The methodology developed may serve as a basic decision support tool for defining specific study paths or student support programs that may eventually be applied. Decision Tree based models were identified as the most suited, and LGBMC had the best performance; thus, it best fitted our data. Average grades and the gap between degrees and master's years drove student success.

There are still many shortcomings in this study, a few were mention on the previous chapter. For further work, we suggest applying other methods for feature selection, so each feature is more significant and optimal for prediction modelling. Apply the same methodology to ECTS related only to the completion of the master's first year, group students based on the master area to identify if the master type influences students' success. Lastly, the methodology could be applied to a broader range of features, including student, parents, and teacher feedback.

References

Abu-dalbouh, H. M. (2021). Application of decision tree algorithm for predicting students' performance via online learning during coronavirus pandemic. *Journal of Theoretical and Applied Information Technology*, *99*(19), 4546–4556.

Adekitan, A. I., & Salau, O. (2020). Toward an improved learning process: the relevance of ethnicity to data mining prediction of students' performance. *SN Applied Sciences*, *2*(1), 1–15. https://doi.org/10.1007/s42452-019-1752-1

Al-Azawei, A., & Al-Masoudy, M. A. A. (2020). Predicting learners' performance in Virtual Learning Environment (VLE) based on demographic, behavioral and engagement antecedents. *International Journal of Emerging Technologies in Learning*, *15*(9), 60–75. https://doi.org/10.3991/ijet.v15i09.12691

Al-Barrak, M. A., & Al-Razgan, M. (2016). Predicting Students Final GPA Using Decision Trees: A Case Study. *International Journal of Information and Education Technology*, *6*(7), 528–533. https://doi.org/10.7763/ijiet.2016.v6.745

Al-Barrak, M. A., & Al-Razgan, M. S. (2015). Predicting students' performance through classification: A case study. *Journal of Theoretical and Applied Information Technology*, *75*(2), 167–175.

Al-Mahmoud, H., & Al-Razgan, M. (2015). Arabic Text Mining: A Systematic Review of the Published Literature 2002-2014. *2015 International Conference on Cloud Computing, ICCC 2015*, *October 2015*. https://doi.org/10.1109/CLOUDCOMP.2015.7149632

Ali, M., & Moreno, P. (2022a). *Data Preparation*. https://pycaret.gitbook.io/docs/get-started/preprocessing/data-preparation#one-hot-encoding

Ali, M., & Moreno, P. (2022b). *Optimization functions in PyCaret*. https://pycaret.gitbook.io/docs/get-started/functions/optimize

Almutairi, F. M., Sidiropoulos, N. D., & Karypis, G. (2017). Context-Aware Recommendation-Based Learning Analytics Using Tensor and Coupled Matrix Factorization. *IEEE Journal on Selected Topics in Signal Processing*, *11*(5), 729–741. https://doi.org/10.1109/JSTSP.2017.2705581

Alshdaifat, E., Al-Shdaifat, A., Zaid, A., & Aloqaily, A. (2020). The impact of data normalization on predicting student performance: A case study from hashemite university. *International Journal of Advanced Trends in Computer Science and Engineering*, *9*(4), 4580–4588. https://doi.org/10.30534/ijatcse/2020/57942020

Altun, M., Kayikçi, K., & Irmak, S. (2022). A Model Proposal for Predicting Students' Academic Performances Based on Data Mining*. *Hacettepe Egitim Dergisi*, *37*(3), 1080–1098. https://doi.org/10.16986/HUJE.2021068491

Arun, D. K., Namratha, V., Ramyashree, B. V., Jain, Y. P., & Roy Choudhury, A. (2021). Student

academic performance prediction using educational data mining. *2021 International Conference on Computer Communication and Informatics, ICCCI 2021*, *2021-Janua*, 1–9. https://doi.org/10.1109/ICCCI50826.2021.9457021

Aydoğdu, Ş. (2020). Predicting student final performance using artificial neural networks in online learning environments. *Education and Information Technologies*, *25*(3), 1913–1927. https://doi.org/10.1007/s10639-019-10053-x

Baashar, Y., Alkawsi, G., Ali, N., Alhussian, H., & Bahbouh, H. T. (2021). Predicting student's performance using machine learning methods: A systematic literature review. *Proceedings - International Conference on Computer and Information Sciences: Sustaining Tomorrow with Digital Innovation, ICCOINS 2021*, *July*, 357–362. https://doi.org/10.1109/ICCOINS49721.2021.9497185

Bayer, J., Bydžovská, H., Géryk, J., Obšívač, T., & Popelínský, L. (2012). Predicting drop-out from social behaviour of students. *Proceedings of the 5th International Conference on Educational Data Mining, EDM 2012*, *Dm*, 103–109.

Dake, D. K., Essel, D. D., & Agbodaze, J. E. (2021). Using Machine Learning To Predict Students' Academic Performance During Covid-19. *Proceedings - 2021 International Conference on Computing, Computational Modelling and Applications, ICCMA 2021*, *Dm*, 9–15. https://doi.org/10.1109/ICCMA53594.2021.00010

Das, A. K., & Rodriguez-Marek, E. (2019). A predictive analytics system for forecasting student academic performance: Insights from a pilot project at eastern Washington university. *2019 Joint 8th International Conference on Informatics, Electronics and Vision, ICIEV 2019 and 3rd International Conference on Imaging, Vision and Pattern Recognition, IcIVPR 2019 with International Conference on Activity and Behavior Computing, ABC 2019*, 255–262. https://doi.org/10.1109/ICIEV.2019.8858523

Dewantoro, G., & Ardisa, N. (2020). A Decision Support System for Undergraduate Students Admissions using Educational Data Mining. *In Proceedings of the 7th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, 105–109. https://doi.org/10.1109/ICITACEE50144.2020.9239244

DGES. (2021). Número de inscritos no ensino superior atinge máximo anual mais elevada da última década. *Ministério Da Ciência, Tecnologia e Ensino Superior - Gabinete Do Ministro*, *1677–7042*, 45708–45716.

Fida, S., Masood, N., Tariq, N., & Qayyum, F. (2022). A Novel Hybrid Ensemble Clustering Technique for Student Performance Prediction. *Journal of Universal Computer Science*, *28*(8), 777–798. https://doi.org/10.3897/jucs.73427

Gil, P. A. V. D. F. (2019). *Unfolding the drivers for academic success: The case of ISCTE-IUL* [ISCTE].

https://repositorio.iscte-iul.pt/handle/10071/20069%0Ahttps://repositorio.iscte-iul.pt/bitstream/10071/20069/4/master_paulo_ferreira_gil.pdf

Gil, P. D., Martins, S. da C., Moro, S., & Costa, J. M. (2021). A data-driven approach to predict first-year students' academic success in higher education institutions. *Education and Information Technologies*, *26*(2), 2165–2190. https://doi.org/10.1007/s10639-020-10346-6

Hashim, A. S., Awadh, W. A., & Hamoud, A. K. (2020). Student Performance Prediction Model based on Supervised Machine Learning Algorithms. *IOP Conference Series: Materials Science and Engineering*, *928*(3). https://doi.org/10.1088/1757-899X/928/3/032019

Hu, Q., & Rangwala, H. (n.d.). Reliable deep grade prediction with uncertainty estimation. *The 9th International LearningAnalytics &Knowl- Edge Conference (LAK19)*, 76–85. https://doi.org/10.1145/3303772.3303802

Hu, Q., & Rangwala, H. (2019). Academic Performance Estimation with Attention-based Graph Convolutional Networks. *In Proceedings of the 12th International Conference on Educational Data Mining (EDM 2019)*, *168*, 69–78.

Hutagaol, N., & Suharjito. (2019). Predictive modelling of student dropout using ensemble classifier method in higher education. *Advances in Science, Technology and Engineering Systems*, *4*(4), 206–211. https://doi.org/10.25046/aj040425

ISCTE. (2021). *About ISCTE*. https://www.iscte-iul.pt/contents/iscte/about-us/541/about-iscte

Jorda, E. R., & Raqueno, A. R. (2019). Predictive model for the academic performance of the engineering students using CHAID and C 5.0 algorithm. *International Journal of Engineering Research and Technology*, *12*(6), 917–928.

Kochański, A. (2003). Data preparation. *Informatyka w Technologii Materiałów*, *10*(January 2010), 124–137. https://doi.org/10.1017/9781107051386.003

Liebowitz, D., González, P., Hooge, E., & Lima, G. (2018). OECD Reviews of School Resources: Portugal 2018. In *OECD Reviews of School Resources*. OECD Publishing. http://dx.doi.org/10.1787/9789264265530-en.%0Ahttp://search.ebscohost.com/login.aspx?direct=true&db=eric&AN=ED565715&site=ehost-live%0Ahttp://dx.doi.org/10.1787/9789264251731-en

Liu, M., Li, Z., Sun, R., & Zhang, N. (2021). Predicting Course Score for Undergrade Students Using Neural Networks. *Intelligent Computing Theories and Application*, *12837*, 732–744. https://doi.org/10.1007/978-3-030-84529-2

Magbag, A., & Raga, R. (2020). Prediction of college academic performance of senior high school graduates using classification techniques. *International Journal of Scientific and Technology Research*, *9*(4), 2104–2109.

Mai, T. Le, Do, P. T., Chung, M. T., Le, V. T., & Thoai, N. (2019). Adapting the Score Prediction to

Characteristics of Undergraduate Student Data. *In Proceedings of the 2019 International Conference on Advanced Computing and Applications (ACOMP). Nha Trang, Vietnam*, 70–77. https://doi.org/10.1109/ACOMP.2019.00018

Mai, T. Le, Do, P. T., Chung, M. T., & Thoai, N. (2019). An apache spark-based platform for predicting the performance of undergraduate students. *Proceedings - 21st IEEE International Conference on High Performance Computing and Communications, 17th IEEE International Conference on Smart City and 5th IEEE International Conference on Data Science and Systems, HPCC/SmartCity/DSS 2019*, 191–199. https://doi.org/10.1109/HPCC/SmartCity/DSS.2019.00041

Maitra, S., Eshrak, S., Bari, M. A., Al-Sakin, A., Munia, R. H., Akter, N., & Haque, Z. (2019). Prediction of academic performance applying NNs: A focus on statistical feature-shedding and lifestyle. *International Journal of Advanced Computer Science and Applications*, *10*(9), 561–570. https://doi.org/10.14569/ijacsa.2019.0100974

Márquez-Vera, C., Cano, A., Romero, C., Noaman, A. Y. M., Mousa Fardoun, H., & Ventura, S. (2016). Early dropout prediction using data mining: A case study with high school students. *Expert Systems*, *33*(1), 107–124. https://doi.org/10.1111/exsy.12135

Mengash, H. A. (2020). Using data mining techniques to predict student performance to support decision making in university admission systems. *IEEE Access*, *8*, 55462–55470. https://doi.org/10.1109/ACCESS.2020.2981905

Miguéis, V. L., Freitas, A., Garcia, P. J. V., & Silva, A. (2018). Early segmentation of students according to their academic performance: A predictive modelling approach. *Decision Support Systems*, *115*, 36–51. https://doi.org/10.1016/j.dss.2018.09.001

Muchuchuti, S., Narasimhan, L., & Sidume, F. (2020). Classification model for student performance amelioration. *Lecture Notes in Networks and Systems*, *69*(January), 742–755. https://doi.org/10.1007/978-3-030-12388-8_51

Nabil, A., Seyam, M., & Abou-Elfetouh, A. (2021). Prediction of Students' Academic Performance Based on Courses' Grades Using Deep Neural Networks. *IEEE Access*, *9*, 140731–140746. https://doi.org/10.1109/ACCESS.2021.3119596

Nudelman, Z., Moodley, D., & Berman, S. (2019). Using bayesian networks and machine learning to predict computer science success. *Communications in Computer and Information Science*, *963*(January), 207–222. https://doi.org/10.1007/978-3-030-05813-5_14

Pandey, A., Shukla, S., & Mohbey, K. K. (2020). Comparative Analysis of a Deep Learning Approach with Various Classification Techniques for Credit Score Computation. *Recent Advances in Computer Science and Communications*, *14*(9), 2785–2799. https://doi.org/10.2174/2666255813999200721004720

Prada, M. A., Dominguez, M., Vicario, J. L., Alves, P. A. V., Barbu, M., Podpora, M., Spagnolini, U.,

Pereira, M. J. V., & Vilanova, R. (2020). Educational Data Mining for Tutoring Support in Higher Education: A Web-Based Tool Case Study in Engineering Degrees. *IEEE Access*, *8*, 212818–212836. https://doi.org/10.1109/ACCESS.2020.3040858

Putpuek, N., Rojanaprasert, N., Atchariyachanvanich, K., & Thamrongthanyawong, T. (2018). Comparative Study of Prediction Models for Final GPA Score: A Case Study of Rajabhat Rajanagarindra University. *Proceedings - 17th IEEE/ACIS International Conference on Computer and Information Science, ICIS 2018*, 92–97. https://doi.org/10.1109/ICIS.2018.8466475

Rosado, J. T., Payne, A. P., & Rebong, C. B. (2019). EMineProve: Educational Data Mining for Predicting Performance Improvement Using Classification Method. *IOP Conference Series: Materials Science and Engineering*, *649*(1). https://doi.org/10.1088/1757-899X/649/1/012018

Shahiri, A. M., Husain, W., & Rashid, N. A. (2015). A Review on Predicting Student's Performance Using Data Mining Techniques. *Procedia Computer Science*, *72*(December), 414–422. https://doi.org/10.1016/j.procs.2015.12.157

Shanbehzadeh, M., Nopour, R., Mashoufi, M., Erfannia, L., Amraei, M., & Mehrabi, N. (2022). Comparing Data Mining Algorithms for Breast Cancer Diagnosis. *Shiraz E Medical Journal*, *23*(7). https://doi.org/10.5812/semj-120140

Sivasakthi, M. (2018). Classification and prediction based data mining algorithms to predict students' introductory programming performance. *Proceedings of the International Conference on Inventive Computing and Informatics, ICICI 2017*, *Icici*, 346–350. https://doi.org/10.1109/ICICI.2017.8365371

Stankoski, S., Kiprijanovska, I., Ilievski, I., Slobodan, J., & Gjoreski, H. (2019). Electrical Energy Consumption Prediction Using Machine Learning Simon. In S. Gievska & G. Madjarov (Eds.), *Communications in Computer and Information Science* (Vol. 1110, pp. 72–82). https://doi.org/10.1007/978-3-030-33110-8

Suleiman, R., & Anane, R. (2022). Institutional Data Analysis and Machine Learning Prediction of Student Performance. *2022 IEEE 25th International Conference on Computer Supported Cooperative Work in Design, CSCWD 2022*, *May*, 1480–1485. https://doi.org/10.1109/CSCWD54268.2022.9776102

Sultana, S., Khan, S., & Abbas, M. A. (2017). Predicting performance of electrical engineering students using cognitive and non-cognitive features for identification of potential dropouts. *International Journal of Electrical Engineering and Education*, *54*(2), 105–118. https://doi.org/10.1177/0020720916688484

Sunday, K., Ocheja, P., Hussain, S., Oyelere, S. S., Balogun, O. S., & Agbo, F. J. (2020). Analysing student performance in programming education using classification techniques. *International Journal of Emerging Technologies in Learning*, *15*(2), 127–144.

https://doi.org/10.3991/ijet.v15i02.11527

Suvon, M. N. I., Siam, S. C., Ferdous, M., Alam, M., & Khan, R. (2022). Masters and Doctor of Philosophy admission prediction of Bangladeshi students into different classes of universities. *IAES International Journal of Artificial Intelligence*, *11*(4), 1545–1553. https://doi.org/10.11591/ijai.v11.i4.pp1545-1553

Tsiakmaki, M., Pierrakeas, C., Kostopoulos, G., Kotsiantis, S., Koutsonikos, G., & Ragos, O. (2019). Predicting university students' grades based on previous academic achievements. *2018 9th International Conference on Information, Intelligence, Systems and Applications, IISA 2018*, 1–6. https://doi.org/10.1109/IISA.2018.8633618

Yaacob, W. F. W., Nasir, S. A. M., Yaacob, W. F. W., & Sobri, N. M. (2019). Supervised data mining approach for predicting student performance. *Indonesian Journal of Electrical Engineering and Computer Science*, *16*(3), 1584–1592. https://doi.org/10.11591/ijeecs.v16.i3.pp1584-1592

Yağcı, M. (2022). Educational data mining: prediction of students' academic performance using machine learning algorithms. *Smart Learning Environments*, *9*(1). https://doi.org/10.1186/s40561-022-00192-z

Yousafzai, B. K., Afzal, S., Rahman, T., Khan, I., Ullah, I., Rehman, A. U., Baz, M., Hamam, H., & Cheikhrouhou, O. (2021). Student-performulator: Student academic performance using hybrid deep neural network. *Sustainability (Switzerland)*, *13*(17), 1–21. https://doi.org/10.3390/su13179775

Zhang, H. (2022). *LightGMB documentation*. https://lightgbm.readthedocs.io/en/v3.3.2/Features.html

Appendixes

A – Literature Review details

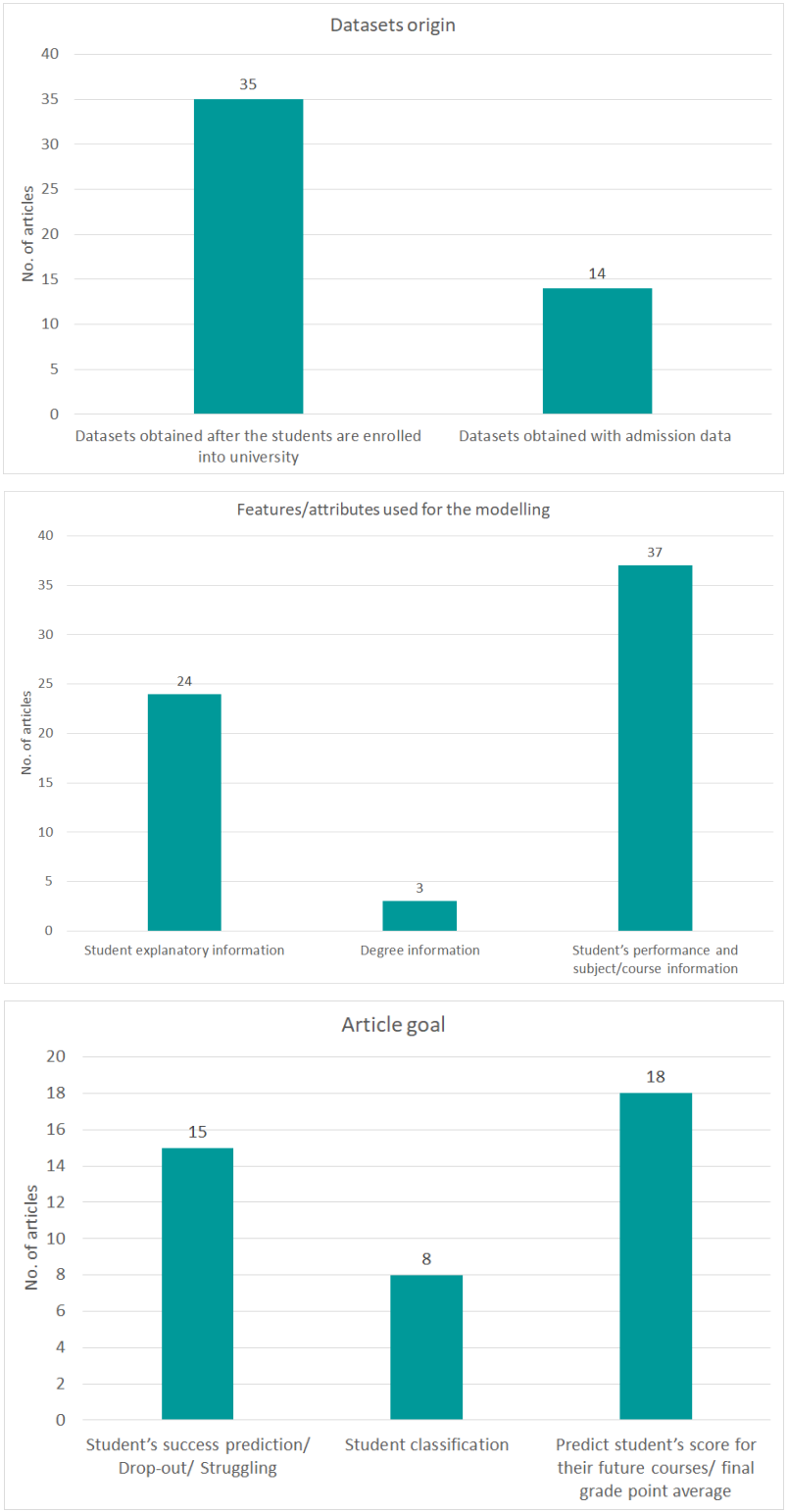

Figure 6. Article's datasets origin, with data obtained with admission or after the enrolment (first plot). Features/attributes selected in the articles collection (middle plot). Articles division by main study goal (last plot).

Table 10. Summary of the 36 reviewed studies.

Authors	Metrics	Features used	Nr. Features	Dataset volume	Goal	Models							
						knn	ANN	MLP	DT	RF	NB	SVM	Mixed/ Other
(Al-Barrak & Al-Razgan, 2015)	ND	Student name, student ID, final GPA, semester of graduation, major, nationality, campus, and all the courses taken by the student including the course' grade	8	236	Predicting Students Final GPA and identify the most important courses in the students' study plan				J48				WEKA
(Almutairi *et al.*, 2017)	RMSE and MAE	Grade of all courses	ND	10245	Predict student performance at the course-level in terms of final grades in classes students have not yet taken								Two CMF models and one Low-Rank Tensor Factorization (LRTF) model.
(Sultana *et al.*, 2017)	Acc	Demographic features: Gender, Mother's education, Student employment status, Guardian, Parent's cohabitation status Cognitive features: Previous result, Sessional marks, Quizzes, Assignments, Projects, Absenteeism, First sessional, Second sessional Non-cognitive features: Time management, Self-concept, Self-appraisal,	25	113	Predicting performance of electrical engineering students for identification of potential dropouts		82%		84%		845		Logistic regression - 84%

		Leadership, Community support, Study preference, Independence, Proximity to college, Go out, School support, Plan for future studies											
(Sivasakthi, 2018)		Students' sex, Higher secondary studied, Medium of Instruction, Private coaching, Area of School, Grade in introductory programming at at college, Grade in introductory programming at test at Test	7	300	Predicting Bachelor of Computer Application student's introductory programming performance			0,9323	J48 - 92,03% REPTree - 91,03%		0,8446	SMO - 90,03%	WEKA
(Putpuek *et al.*, 2018)	Acc	Gender, Graduated accumulative GPA,Property of loan status, Approved loan status, admission type, talent and province of high school.	7	2281	Compare prediction models for the level of the final grade point average (GPA) score of graduated students	43.05%			C4.5 - 42.88% ID3 - 41.65%		43.18%		
(Miguéis *et al.*, 2018)	Acc	Socio-economic status, High school background, Enrolment process, First year assessment, First year performance, Socio-demographic	6	2459	Predict overall academic performance.				DT - 91.5% Bagging - Decision Trees - 88.7% Adaptive Boosting - Decision Trees - 95.7%	96.1%	75.9%	93.9%	
(Das & Rodriguez-Marek,	Acc	Demographic: gender, whether first generation student, family income.	11	227	Determining whether any correlation	70.62%							

2019)		Academic: SAT/ACT score, high school GPA, Math I/II/III grades, Physics I/II/III grades.			exists between preparatory classes taken by Electrical Engineering (EE) students at Eastern Washington University (EWU) early in their academic careers and their departmental GPAs upon graduation.								
(Hu & Rangwala, 2019)	Acc	Courses taken and respective grades	94	21688	Students course grade prediction and detection at-risk students		Graph Convolutional Network (GCN) - 92.45% LSTM - 86.79%	89.62%					
(Mai, Do, Chung, Le, *et al.*, 2019)	RMSE, MAE and MSE	Course and grade	39	61271	Predict student's score for their future courses using various techniques based on Recommendation System								User-based, Item-based Collaborative Filtering and Matrix Factorization (using Alternative Least Square) and Non-

													negative Matrix Factorization
(Mai, Do, Chung, & Thoai, 2019)	RMSE, MAE and MSE	Course and grade	39	61271	Predict The Performance of Undergraduate Student								Combination of Collaborative Filtering & Matrix Factorization WEKA
(Rosado *et al.*, 2019)	Acc	Subjects, parent's occupations, grade, general average of students, final academic performance of students, last school attended, marketing	8	4250	Predict the performance improvement of students						92.37 %		
(Tsiakmaki *et al.*, 2019)	MAE	Gender, Way that students entered the Department, Grades of each course, Unsuccessful attempts to pass the corresponding course in previous semesters' examinations	18	592	Predicting University Students' Grades	1.285				1.198		1.207	Linear Regression M5 WEKA
(Maitra *et al.*, 2019)	Acc	lifestyle-factors, living with family, attendance in class, gender, class note-taking tendency etc	103	Not defined	Prediction of Academic Performance		92.77%						
(Jorda & Raqueno, 2019)	Acc	Courses and respective grades	13	3765	Predict Academic Performance of Engineering Students		82.92%		CHAID - 83.68% C5.0 - 85.93% C&R Tree - 83.03%		82.05 %		Discriminant - 63.03% Decision List - 43.57% Logistic regression -

													83.538%
(Hu & Rangwala, n.d.)	Acc	Courses taken and respective grades	91	17652	Grade prediction		LSTM - 92.07%	90.24%					
(Yaacob *et al.*, 2019)	Acc	ID, CGPA every semester and the course grade student	12	631	Predicting student performance	84.80%			82.15% - Information Gain 80.99% - Gini		89.26 %		Logistic regression - 85.28%
(Nudelman *et al.*, 2019)	Acc	Student's scores for Mathematics, English, Physical sciences, Quantitative literacy, Academic literacy, Courses Registered, Courses passed, Cumulative GPA, Financial aid	17	783	Predict Computer Science Student's Success				J48 Sensitivity 63.61%	Sensitivity 59.70%	Sensitivity 64.17 %		Bayesian Networks Sensitivity 90.64%
(Dewantoro & Ardisa, 2020)	Acc	Students' admission data: Numerical grade earned on high school exam(HS Grade), High school major(HS Major), High school origin(HS Origin) , Grade Point Average upon graduation(GPA) , Length of study, Length of study period until graduation	5	145	Predict GPA and Length of study period until graduation	85.714%	92.857%				85.714%		
(Sunday *et al.*, 2020)	Acc	Class, Test Score, Assignment Completed, Class Lab Work, Class Attendance	5	239	Classification of students				ID3 - 85.355% J48 - 87.02%				WEKA
(Muchuchuti *et al.*, 2020)	Acc	Academic progression data and the final degree classification.	10	124	Predict the final performance of students					0,5792	53.47 %		IBK - 57.16% Adaboost - 53.54%

													OneR - 60.19% WEKA
(Prada *et al.*, 2020)	Acc	Student explanatory information: gender, age, age of access to studies, nationality, type of previous education, admission score, student status, parents' education level and residence. Degree information: institution, nature and length. Student's performance and subject information: name, type and length of the subject, score, number of attempts, semester, year, knowledge area, language, nature, average score, failure rate and mobility status.	26	21000	Student classification and dropout prediction							85%	
(Al-Azawei & Al-Masoudy, 2020)	Acc	Demographics, performance and behavioral features.	>10	38239	Predicting students' performance in a Virtual Learning Environment (VLE) based on four time periods of the examined online course in order to provide an early prediction model.								WEKA
(Aydoğdu, 2020)	Acc	Gender, content score, time spent on the content,	10	3518	Predict student		80.47%						

		number of entries to content, homework score, number of attendance to live sessions, total time spent in live sessions, number of attendance to archived courses and total time spent in archived courses			performances								
(Magbag & Raga, 2020)	Acc	Gender, SHS grade and strand, entrance exam performance as pre-enrolment data and freshmen details such as college, course and the number of units enroled.	17	4762	Prediction Of College Academic Performance Of Senior High School Graduates		64.02%						Logistic regression - 60,97%
(Adekitan & Salau, 2020)	Acc	Year of graduation, the geopolitical zone, the college, preadmission scores and the Cumulative Grade Point Average	5	2413	Identify the relationship, if any, between the admission criteria scores and the graduation grades, and to examine the infuence of ethnicity using the geopolitical zone of origin of the student on the predictive accuracy of the models		52.8%		43.8%	49.6%	51.7%		
(Mengash, 2020)	Acc	Pre-admission criteria (high school grade average, Scholastic, Achievement Admission Test score, and General Aptitude Test	4	2039	Predict Student Performance		79.22%		75.91%		73.61 %	75.2 8%	

		score)											
(Abu-dalbouh, 2021)	Acc	Student: Student age, Student level, residence, Gender, Study Type, GPA, High school grade Course: course name, Suitableness of course in e-learning, influence of assessment, course name Lecturer: gender, lecture degree, influence of lecturer Infrastructure: hardware and internet spead, recording lectures, availability of online learning tools	18	1062	Predicting student's performance				J48 95.06 %		87.80 %		Bayes Net-D 90.24%
(Liu *et al.*, 2021)	Acc	Student' history semester course information: grades in required history semester courses, GPA, and failure rate in required courses Teacher' information set of the target course: teacher name, class mean and standard deviation of the course taught by the teacher	10	51498	Predict undergraduate course scores for a new semester based on students' previous academic performance and provide early risk-warning in specific course for high risk students who maybe fail in this course.		94%						
(Nabil *et al.*, 2021)	Acc	Academic features related to students' grades in the courses of the first academic year.	12	4266	Prediction of Students' Academic Performance	91%	Deep neural network - 89%		87%	91%		91%	gradient boosting 91% logistic

													regressio n - 91%
(Arun *et al.*, 2021)	Acc		13	Not defined	Student Academic Performance Prediction								Group 1: NavieBay esUpdate able, Hoeffidin g Tree, Random forest, Logistic Regressio n,Classific ationViaR egression . - 86.09 Group 2: Hoeffding Tree, Logistic Regressio n, NavieBay esUpdate able, JRip, IterativeC lassifierO ptimizer. - 86.01 Group 3: Classificat ionViaReg ression, Random Forest, AdaBoost , Decision Table, LMT. - 87.38

(Yousafzai *et al.*, 2021)	Acc	Name of student's school, gender, age, Home address , Size of family, Cohabitation status of parent, mother's education, father's education, mother's job, father's job, Why school was chosen,Guardian of student, Travel time from home to school, Study time in a week, Number of class failures in past, Additional educational support, Educational support from family, Extra paid classes within the course subject, Extra-curricular activities, Attending nursery school, Desire of taking higher education, Internet facility at home, Have a romantic relationship, Family relationships quality, After school free time, Going outside with friends, Alcohol usage at daytime, Alcohol usage at weekend, Recent health status, Absences from school, Grade of the first period, Grade of the second period, Grade of final period	30	1044	Student Academic Performance	84.55%				85.33%	82.21 %	70.79%	Bidirectional Long Short-Term Memorynetwork - 90.16% Logistic regression - 81.67%
(Dake *et al.*, 2021)	Acc	Siblings Disturbing, COVID-19 and State of Mind, Overall Lecturers Support at Home, General Family Support, Facilities for Academic Work at Home, Personal Computer,	13	536	Predict student's academic performance				J48 - 86.56%	83.95%	85.63 %		Random Tree - 73.12%

		Learning Condition (At Home), Internet Connectivity at Home, Region During COVID-19 Period, Age, Gender											
(Altun *et al.*, 2022)	Acc	Exam scores, final grades, weighted averages of semester grades, and graduation grades	4	1570	Predicting Students Academic Performances		94%		87%		85.714		Multiple Linear Regression - 97% Logistic Regression - 72%
(Fida *et al.*, 2022)	Acc	Demographic, Academic background, Behavioral and other extra features.	17	400	Student Performance Prediction								Ensemble Meta-Based Tree Model - 93%
(Yağcı, 2022)	Acc	Midterm exam grades, Department data and Faculty data	>20	1854 students from state University in Turkey	Prediction of students' academic performance	69.9%	74.6%			74.6%	71.3%	73.5%	Logistic Regression - 71.7%

(Suleiman & Anane, 2022)	RMSE	Demographic and academic data: an index , gender of the student, marital status, age, pre-entry score, mode of entry, year of entry, cumulative grade point average for year 1 , year 2, year 3 and year 4	11	1769 students from Nigerian University	Prediction of student performance								Linear Regression
(Suvon *et al.*, 2022)	Acc	Student's name, admitted university name with its state and country, admitted department, intended research area, types of funding (fellowship, assistantship, external scholarship), intended semester, undergraduate university name with CGPA and department, IELTS/TOEFL score, GRE score, publications, job experience, research experience, application method, and funding source.	18	230	Admission prediction of Bangladeshi students				89%	86%			

ND – Not defined; Acc - accuracy

B – Hyperparameters

Table 11. Hyperparameters choose to tune LGBMC, GBC, RF, ADA, DT models.

Model	Learning Rate	Max_depth	n_estimators	Max_features	Min_samples_split	Min_samples_leaf	bootstrap	Max_leaf_nodes
LGBMC	0.01, 0.05, 0.1, 0.2	5, 6, 7, 8	100, 300, 400, 500					
GBC	0.01, 0.05, 0.1, 0.2	3, 5, 7, 8, 9, 10, 12	100, 120, 150, 200, 250, 300	np.random.randint(1, 9,20)				
RF		5, 6, 7, 8, 10, 12	100, 300, 400, 500		2, 5, 7, 9, 10	1, 2, 4, 5, 6, 7	'True', 'False'	
ADA	0.01, 0.05, 0.1, 0.2, 0.5, 0.7		100, 200, 250, 300, 400, 500					
DT		5, 6, 7, 8, 10, 12			2, 5, 7, 9, 10	1, 2, 4, 6		10,40,70, 100

Table 12. Hyperparameters of the three PyCaret LGBMC models, before and after auto tuning and custom tuning.

Model	Hyperparameters
LGBMC	`LGBMClassifier(boosting_type='gbdt',class_weight=None,colsample_bytree=1.0, importance_type='split',learning_rate=0.1,max_depth=-1, min_child_samples=20,min_child_weight=0.001,min_split_gain=0.0, n_estimators=100,n_jobs=-1,num_leaves=31,objective=None, random_state=5916,reg_alpha=0.0,reg_lambda=0.0,silent='warn', subsample=1.0,subsample_for_bin=200000,subsample_freq=0)`

LGBMC_tuned	LGBMClassifier(boosting_type='gbdt',class_weight=None,colsample_bytree=1.0, importance_type='split',learning_rate=0.1,max_depth=-1, min_child_samples=20,min_child_weight=0.001,min_split_gain=0.0, n_estimators=100,n_jobs=-1,num_leaves=31,objective=None, random_state=5916,reg_alpha=0.0,reg_lambda=0.0,silent='warn', subsample=1.0,subsample_for_bin=200000,subsample_freq=0)
LGBMC_custom_tuned	LGBMClassifier(boosting_type='gbdt',class_weight=None,colsample_bytree=1.0, importance_type='split',learning_rate=0.05,max_depth=6, min_child_samples=20,min_child_weight=0.001,min_split_gain=0.0, n_estimators=300,n_jobs=-1,num_leaves=31,objective=None, random_state=5916,reg_alpha=0.0,reg_lambda=0.0,silent='warn', subsample=1.0,subsample_for_bin=200000,subsample_freq=0)

Table 13. Hyperparameters of the three PyCaret GBC models, before and after auto tuning and custom tuning.

Model	Hyperparameters
GBC	GradientBoostingClassifier(ccp_alpha=0.0,criterion='friedman_mse',init=None, learning_rate=0.1,loss='deviance',max_depth=3, max_features=None,max_leaf_nodes=None, min_impurity_decrease=0.0,min_impurity_split=None, min_samples_leaf=1,min_samples_split=2, min_weight_fraction_leaf=0.0,n_estimators=100, n_iter_no_change=None,presort='deprecated', random_state=5916,subsample=1.0,tol=0.0001, validation_fraction=0.1,verbose=0, warm_start=False)
GBC_tuned	GradientBoostingClassifier(ccp_alpha=0.0,criterion='friedman_mse',init=None, learning_rate=0.15,loss='deviance',max_depth=5, max_features=1.0,max_leaf_nodes=None, min_impurity_decrease=0.002,min_impurity_split=None, min_samples_leaf=2,min_samples_split=10, min_weight_fraction_leaf=0.0,n_estimators=200, n_iter_no_change=None,presort='deprecated', random_state=5916,subsample=1.0,tol=0.0001, validation_fraction=0.1,verbose=0, warm_start=False)
GBC_custom_tuned	GradientBoostingClassifier(ccp_alpha=0.0,criterion='friedman_mse',init=None, learning_rate=0.1,loss='deviance',max_depth=3, max_features=None,max_leaf_nodes=None, min_impurity_decrease=0.0,min_impurity_split=None, min_samples_leaf=1,min_samples_split=2, min_weight_fraction_leaf=0.0,n_estimators=100, n_iter_no_change=None,presort='deprecated',

	random_state=5916,subsample=1.0,tol=0.0001, validation_fraction=0.1,verbose=0, warm_start=False)

Table 14. Hyperparameters of the three PyCaret ADA models, before and after auto tuning and custom tuning.

Model	Hyperparameters
ADA	AdaBoostClassifier(algorithm='SAMME.R', base_estimator=None, learning_rate=1.0,n_estimators=50, random_state=5916)
ADA_tuned	AdaBoostClassifier(algorithm='SAMME.R', base_estimator=None, learning_rate=0.5, n_estimators=200, random_state=5916)
ADA_custom_tuned	AdaBoostClassifier(algorithm='SAMME.R',base_estimator=None,learning_rate=0.5,n_estimators=500, random_state=5916)

Table 15. Hyperparameters of the three PyCaret RF models, before and after auto tuning and custom tuning.

Model	Hyperparameters
RF	RandomForestClassifier(bootstrap=True,ccp_alpha=0.0,class_weight=None, criterion='gini',max_depth=None,max_features='auto', max_leaf_nodes=None,max_samples=None, min_impurity_decrease=0.0,min_impurity_split=None, min_samples_leaf=1,min_samples_split=2, min_weight_fraction_leaf=0.0,n_estimators=100, n_jobs=-1,oob_score=False,random_state=5916,verbose=0, warm_start=False)
RF_tuned	RandomForestClassifier(bootstrap=True,ccp_alpha=0.0,class_weight={}, criterion='gini',max_depth=11,max_features=1.0, max_leaf_nodes=None,max_samples=None, min_impurity_decrease=0,min_impurity_split=None, min_samples_leaf=4,min_samples_split=10, min_weight_fraction_leaf=0.0,n_estimators=140, n_jobs=-1,oob_score=False,random_state=5916,verbose=0, warm_start=False)
RF_custom_tuned	RandomForestClassifier(bootstrap=True,ccp_alpha=0.0,class_weight=None, criterion='gini',max_depth=None,max_features='auto', max_leaf_nodes=None,max_samples=None, min_impurity_decrease=0.0,min_impurity_split=None, min_samples_leaf=1,min_samples_split=2, min_weight_fraction_leaf=0.0,n_estimators=100,

	n_jobs=-1,oob_score=False,random_state=5916,verbose=0, warm_start=False)

Table 16. Hyperparameters of the three PyCaret Decision Tree models, before and after auto tuning and custom tuning (custom tune and bagging method).

Model	Hyperparameters
DT	DecisionTreeClassifier(ccp_alpha=0.0,class_weight=None,criterion='gini', max_depth=None,max_features=None,max_leaf_nodes=None, min_impurity_decrease=0.0,min_impurity_split=None, min_samples_leaf=1,min_samples_split=2, min_weight_fraction_leaf=0.0,presort='deprecated', random_state=5916,splitter='best')
DT_tuned	DecisionTreeClassifier(ccp_alpha=0.0,class_weight=None,criterion='gini', max_depth=12,max_features=1.0,max_leaf_nodes=None, min_impurity_decrease=0,min_impurity_split=None, min_samples_leaf=5,min_samples_split=10, min_weight_fraction_leaf=0.0,presort='deprecated', random_state=5916,splitter='best')
DT_custom_tuned	DecisionTreeClassifier(ccp_alpha=0.0,class_weight=None,criterion='gini', max_depth=12,max_features=None,max_leaf_nodes=100, min_impurity_decrease=0.0,min_impurity_split=None, min_samples_leaf=6,min_samples_split=5, min_weight_fraction_leaf=0.0,presort='deprecated', random_state=5916,splitter='best')
DT bagging	BaggingClassifier(base_estimator=DecisionTreeClassifier(ccp_alpha=0.0, class_weight=None, criterion='gini', max_depth=None, max_features=None, max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split=None, min_samples_leaf=1, min_samples_split=2, min_weight_fraction_leaf=0.0, presort='deprecated', random_state=5916, splitter='best'), bootstrap=True,bootstrap_features=False,max_features=1.0, max_samples=1.0,n_estimators=100,n_jobs=1,oob_score=False, random_state=123,verbose=0,warm_start=False)
DT boosted	AdaBoostClassifier(algorithm='SAMME.R', base_estimator=DecisionTreeClassifier(ccp_alpha=0.0, class_weight=None,

	criterion='gini', max_depth=None, max_features=None, max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split=None, min_samples_leaf=1, min_samples_split=2, min_weight_fraction_leaf=0.0, presort='deprecated', random_state=5916, splitter='best'), learning_rate=1.0,n_estimators=100,random_state=123)

C – Plots

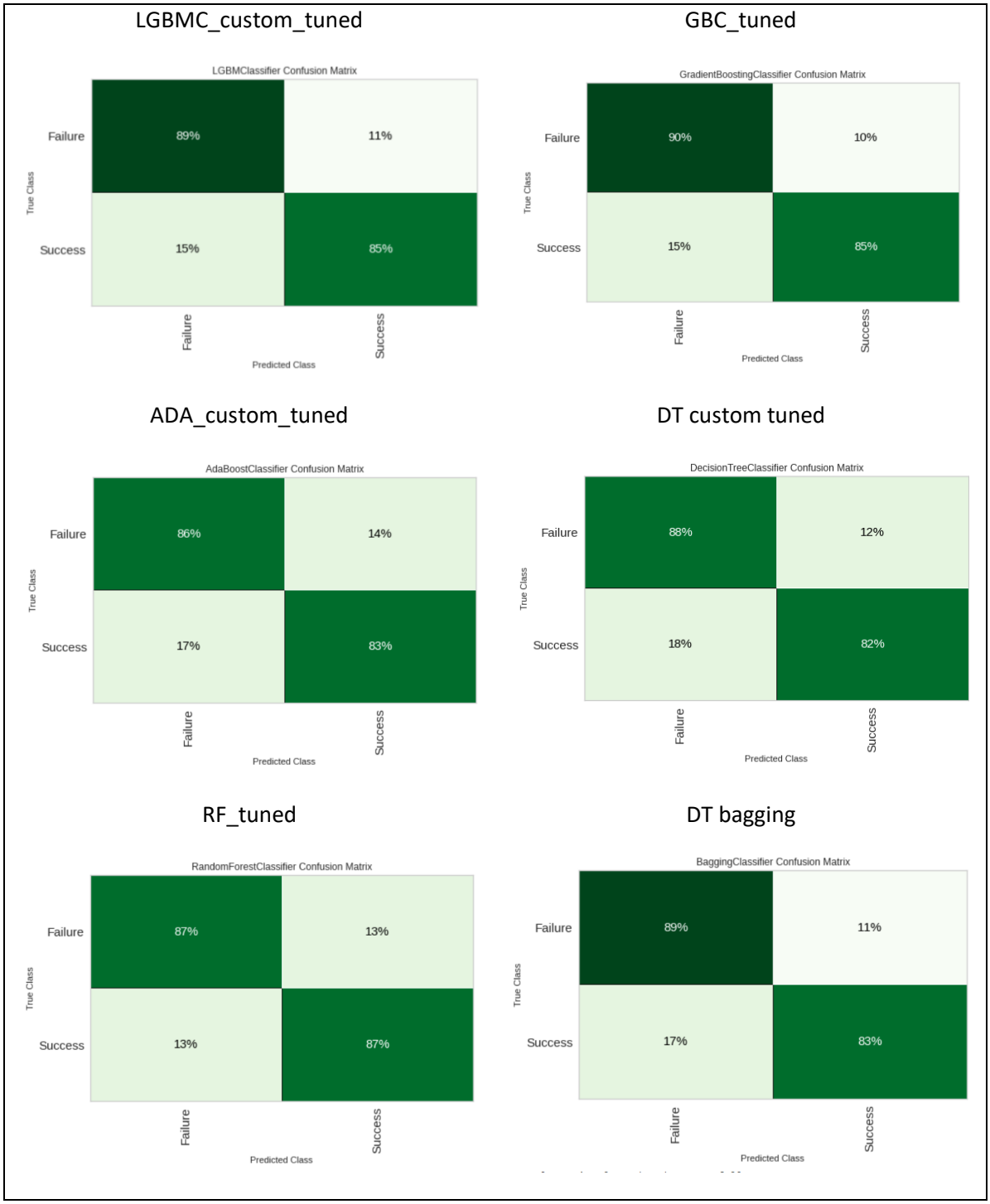

Figure 7. Confusion matrix for LGBMC, GBC, ADA, DT and RF models (tuned or custom tuned, the plot concerns the model with highest performance within each algorithm type).

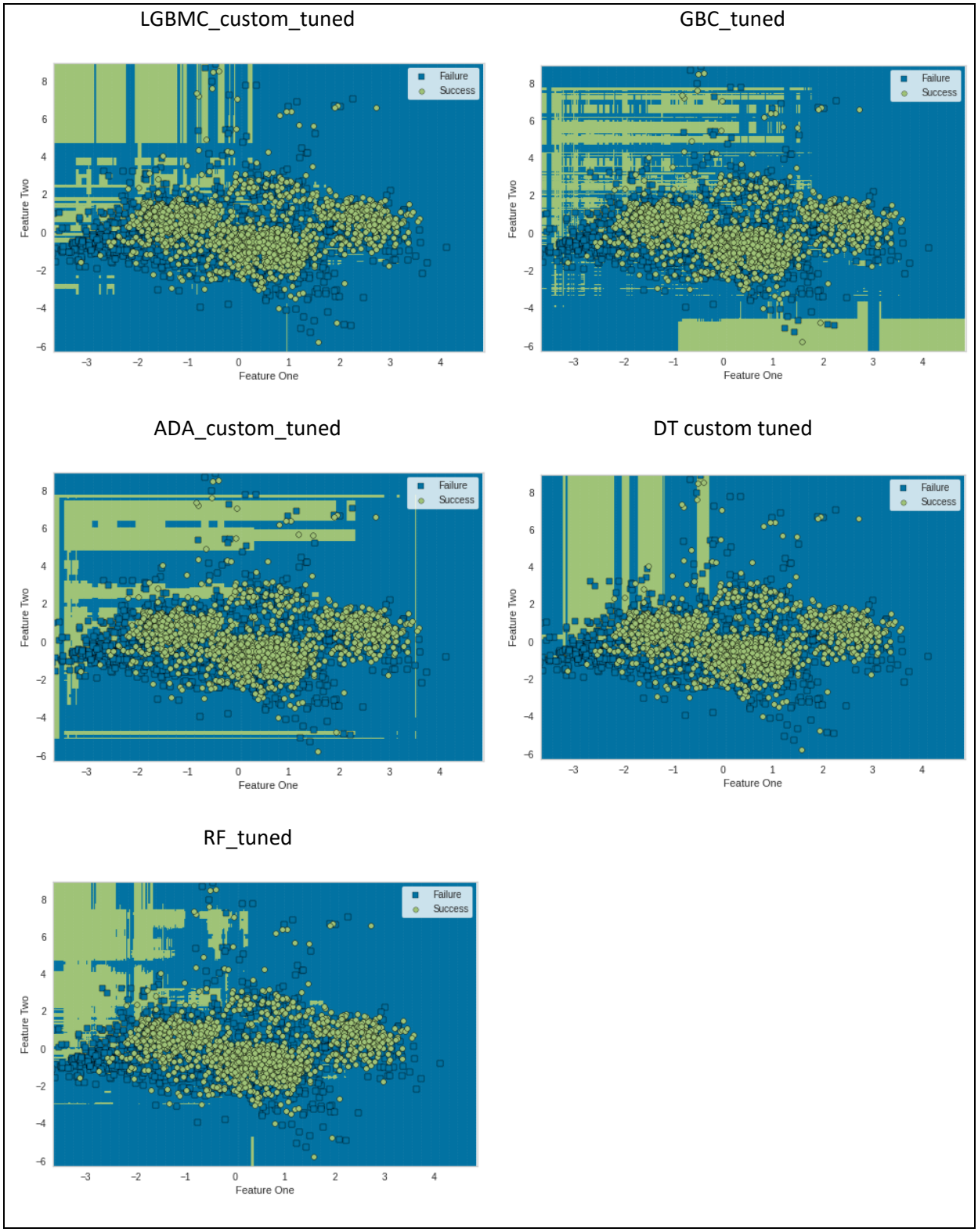

Figure 8. Boundary plot for LGBMC, GBC, ADA, DT and RF models (tuned or custom tuned, the plot concerns the model with highest performance within each algorithm type).

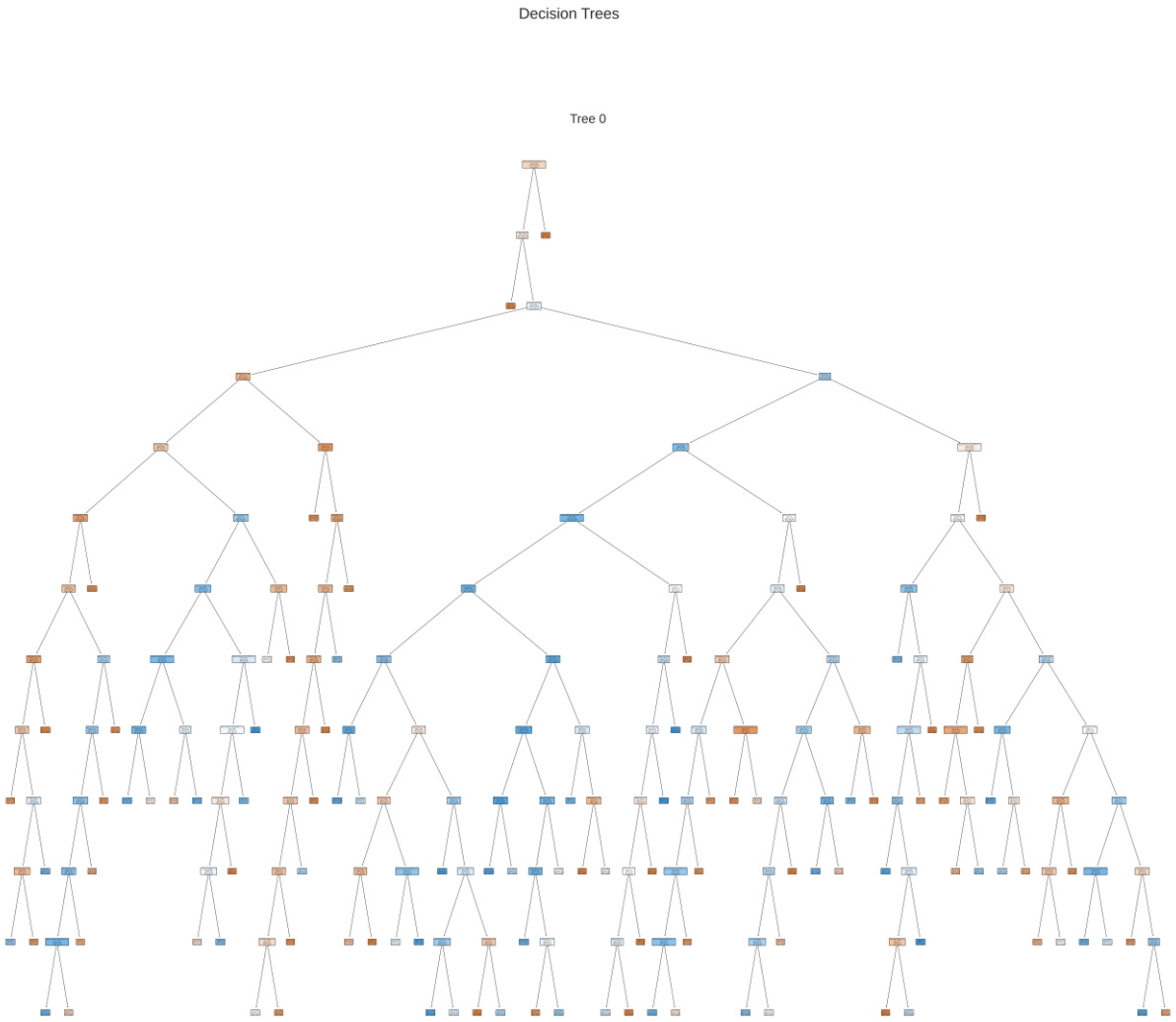

Figure 9. Tree created by Custom Tuned Decision Tree.

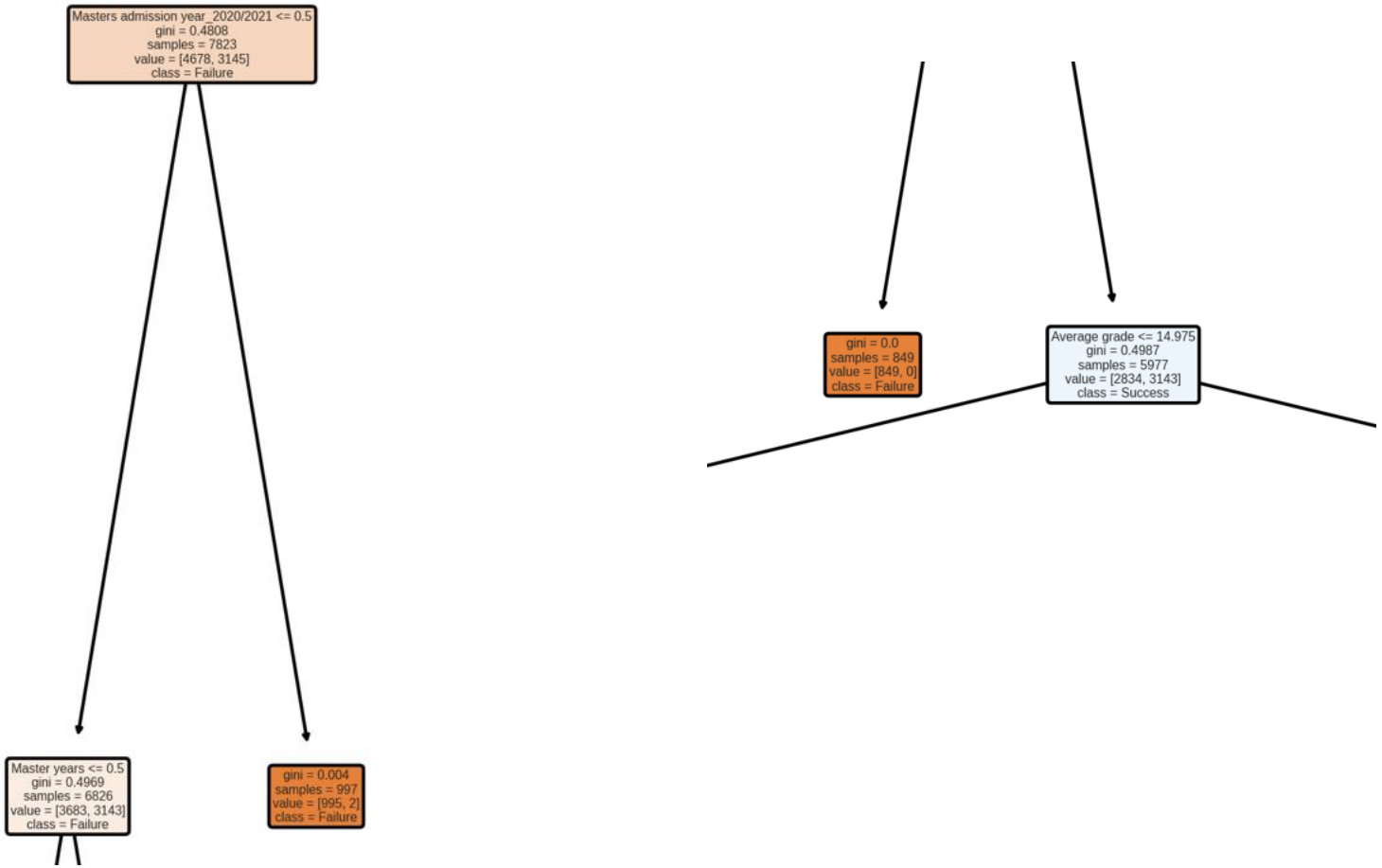

Figure 10. Cuts of the Decision Tree created by the Decision Tree tuned model. Left – root node of the tree. Right – node that created the two main branche