Predicting student's performance in the test "Saber PRO"

Julian Gomez Benitez	Juan Pablo Rincon Usma	Miguel Correa	Mauricio Toro
University Eafit	University Eafit	Universidad Eafit	Universidad Eafit
Colombia	Colombia	Colombia	Colombia
jgomezb11@eafit.edu.co	jprinconu@eafit.edu.co	macorream@eafit.edu.co	mtorobe@eafit.edu.co

ABSTRACT

Today technology gives us a proportion of data that we can use to predict results about everyday things that happen around us, we want to focus on predicting academic success through academic and sociodemographic variables, the idea is to analyze these data and predict whether students will be able to do well on the "Saber pro" tests. Similar studies have been carried out which allow us to determine how the students will do, and they have given good results, taking this into account, help can be offered to students who have difficulties and thus avoid that their grades decrease considerably.

Keywords

Decision trees, machine learning, academic success, standardized student scores, test-score prediction

1. INTRODUCTION

in a future, the technology is going to be the new key for the education using the digital transformation process of education in Colombia.

In the past has been studied the causes and motivations that influence of academic desertion, using algorithm to predict the academic desertion, even so, few have managed to predict the academic success in the higher education, the academic success can be consider like the employability of the graduate, the salary of the graduate or even the happiness in their jobs.

For this project we will consider the academic success like the probability of that one student gets a score above average of another students. The test "Saber Pro" are the Colombian´s government test that is design for the university students, that are about to finish the college.

1.1. Problem

We want to design an algorithm base on decision tree and the information of "Saber 11" to predict the if a student will get a score above average. For this we have the dates of his age, his degree, his parent's salary, his gender, his stratum, etc.

1.2 Solution

In this work, we focused on decision trees because they provide great explainability “Simpler models such as linear regression and decision trees on the other hand provide less predictive capacity and are not always capable of modelling the inherent complexity of the dataset (i.e. feature interactions). They are however significantly easier to explain and interpret. “ . We avoid black-box methods such as neural networks, support-vector machines and random forests because they lack explainability. “Black-box models such as neural networks, gradient boosting models or complicated ensembles often provide great accuracy. The inner workings of these models are harder to understand and they don’t provide an estimate of the importance of each feature on the model predictions, nor is it easy to understand how the different features interact.”

We opt to use a decision tree, specifically a Cart, because it has some great advantages, they provide most model interpretability because they are simply series of if-else conditions, missing values in the data also do NOT affect the process of building a decision tree to any considerable extent and a decision tree does not require normalization of data.

1.3 Article structure

In what follows, in Section 2, we present related work to the problem. Later, in Section 3 we present the datasets and methods used in this research. In Section 4, we present the algorithm design. After, in Section 5, we present the results. Finally, in Section 6, we discuss the results and we propose some future work directions.

2. RELATED WORK

Explain four (4) articles related to the problem described in Section 1.1. You may find the related problems in scientific journals. Consider Google Scholar for your search. *(In this semester, related work is research on decision trees to predict student-test scores or academic success)*

3.1 PREDICTING STUDENTS’ PERFORMANCE USING ID3 AND C4.5 CLASSIFICATION ALGORITHMS

An educational institution needs to have an approximate prior knowledge of enrolled students to predict their performance in future academics. This helps them to identify promising students and also provides them an opportunity to pay attention to and improve those who would probably get lower grades.

we have developed a system which can predict the performance of students from their previous performances using concepts of data mining techniques under Classification. We have analyzed the data set containing information about students, such as gender, marks scored in the board examinations of classes X and XII. By applying the ID3 (Iterative Dichotomiser 3) and C4.5 classification algorithms on this data. for a total of 182 students, the

average percentage of accuracy achieved in Bulk and Singular Evaluations is approximately 75.275.

PREDICTING STUDENTS' PERFORMANCE USING ID3 AND C4.5 CLASSIFICATION ALGORITHMS (N.o 5).(2013). https://arxiv.org/ftp/arxiv/papers/1310/1310.2071.pdf

3.2 Mining Student Data Using Decision Trees

Student performance in university courses is of great concern to the higher education managements where several factors may affect the performance. We use the data mining processes, particularly classification, to help in enhancing the quality of the higher educational system by evaluating student data. From the obtained results, we can notice that the classification accuracy algorithms are not so high. This can indicate that the collected samples and attributes are not sufficient to generate a classification model of high quality.

Mining Student Data Using Decision Trees. (2006). https://www.acit2k.org/ACIT2006/Proceeding/131.pdf

3.3 Predicting Student Performance using Classification and Regression Trees Algorithm.

Student academic achievement is always a matter of great concern to education stakeholders, especially in today's fastpaced, web-enabled classrooms. High quality teaching stuff, well-designed curriculum, student-centered learning and academic support are heavily impact on student success and help to equalize education background differences. Classification and Regression Trees (CART) decision tree algorithm was used to classify students and predict those at risk, based on the impact of four online activities: message exchanging, group wiki content creation, course files opening and online quiz taking. In this study, the CART technique achieved very high accuracy (99.1 %) in classifying students into those who successfully passed the class and those who failed to do so.

Predicting Student Performance using Classification and Regression Trees Algorithm .(2020). http://www.ijitee.org/wpcontent/uploads/papers/v9i3/C8964019320.pdf

3.4 Decision trees for predicting the academic success of students

The aim of this paper is to create a model that successfully classifies students into one of two categories, depending on their success at the end of their first academic year, and finding meaningful variables affecting their success. The most significant variables were total points in the state exam, points from high school and points in the Croatian language exam. The highest classification rate of 79% was produced using the REPTree decision tree algorithm, but the tree was not as successful in classifying both classes. Therefore, the average rate of classification was calculated for two models that gave the highest total rate of classification, where a higher percentage is achieved using the model relying on the algorithm J48.

Decision trees for predicting the academic success of students. (2016). http://bib.irb.hr/datoteka/853222.c**lanak.pdf**

3. MATERIALS AND METHODS

In this section, we explain how the data was collected and processed and, after, different solution alternatives considered to choose a decision-tree algorithm.

3.1 Data Collection and Processing

We collected data from the *Colombian Institute for the Promotion of Higher Education* (ICFES), which is available online at ftp.icfes.gov.co. Such data includes anonymized Saber 11 and Saber Pro results. Saber 11 scores of all Colombian high schools graduated from 2008 to 2014 and Saber Pro scores of all Colombian bachelor-degree graduates from 2012 to 2018 were obtained. There were 864,000 records for Saber 11 and records 430,000 for Saber Pro. Both Saber 11 and Saber Pro, included, not only the scores but also socio-economic data from the students, gathered by ICFES, before the test.

In the next step, both datasets were merged using the unique identifier assigned to each student. Therefore, a new dataset that included students that made both standardized tests was created. The size of this new dataset is 212,010 students. After, the binary predictor variable was defined as follows: Does the student score in Saber Pro is higher than the national average of the period?

It was found out that the datasets were not balanced. There were 95,741 students above average and 101,332 students below average. We performed undersampling to balance the dataset to a 50%-50% ratio. After undersampling, the final dataset had 191,412 students.

Finally, to analyze the efficiency and learning rates of our implementation, we randomly created subsets of the main dataset, as shown in Table 1. The dataset was divided into 70% for training and 30% for testing. Datasets are available at https://github.com/mauriciotoro/ST0245-Eafit/tree/master/proyecto/datasets .

	Dataset 1	Dataset 2	Dataset 3	Dataset 4	Dataset 5
Train	15,000	45,000	75,000	105,000	135,000
Test	5,000	15,000	25,000	35,000	45,000

Table 1. Number of students in each dataset used for training and testing.

3.2 Decision-tree algorithm alternatives

In what follows, we present different algorithms to solve to automatically build a binary decision tree. *(In this semester, examples of such algorithms are ID3, C4.5, Hierarchical clustering, Decision tree learning).*

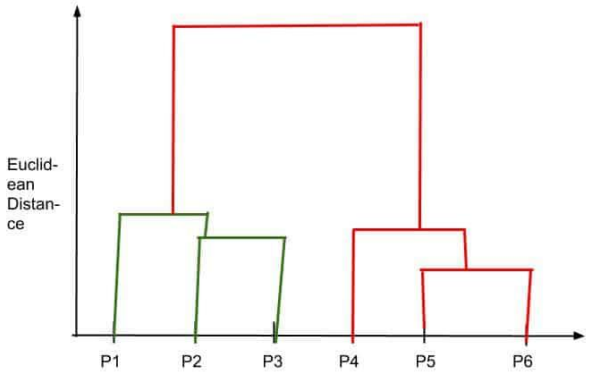
3.2.1 Hierarchical clustering

hierarchical clustering is a method of cluster analysis which seeks to build a hierarchy of clusters. Strategies for hierarchical clustering generally fall into two types:

Agglomerative: This is a "bottom-up" approach: each observation starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy.

Divisive: This is a "top-down" approach: all observations start in one cluster, and splits are performed recursively as one moves down the hierarchy.

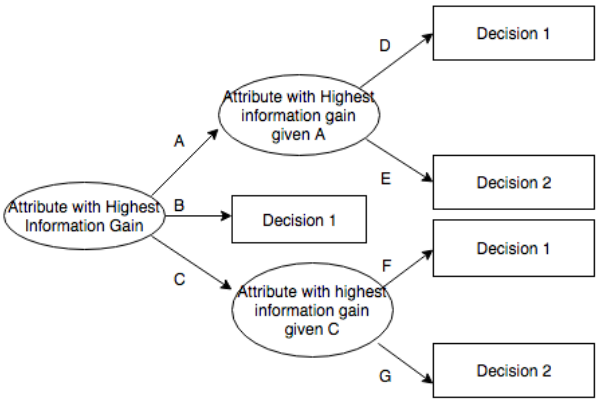
This algorithm has a time complexity of O(n^3) and requires O(n^2) memory



3.2. ID3 algorithm

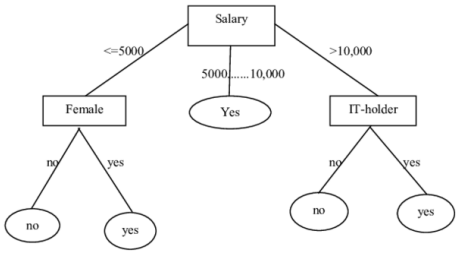
The ID3 algorithm begins with the original set S as the root node. On each iteration of the algorithm, it iterates through every unused attribute of the set S and calculates the entropy or the information gain IG(S) of that attribute. It then selects the attribute which has the smallest entropy (or largest information gain) value. The set S is then split or partitioned by the selected attribute to produce subsets of the data. The algorithm continues to recurse on each subset, considering only attributes never selected before.

This algorithm has a time complexity of O(m · n), where m is the size of the training data and n is the number of attributes.



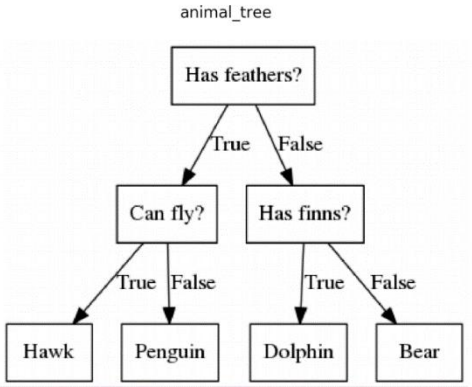
3.2.3 C4.5 algorithm

C4.5 builds decision trees from a set of training data in the same way as ID3, using the concept of information entropy. At each node of the tree, C4.5 chooses the attribute of the data that most effectively splits its set of samples into subsets enriched in one class or the other. The splitting criterion is the normalized information gain (difference in entropy). The attribute with the highest normalized information gain is chosen to make the decision. The time complexity O(m · n2) Where m is the size of the training data and n is the number of attributes



3.2.4 Decision tree learning

Decision tree learning is a method commonly used in data mining. The goal is to create a model that predicts the value of a target variable based on several input variables. A tree is built by splitting the source set, constituting the root node of the tree, into subsets—which constitute the successor children. The splitting is based on a set of splitting rules based on classification features. This process is repeated on each derived subset in a recursive manner called recursive partitioning. The standard decision-tree learning algorithm has a time complexity of O(m · n2).



.4. ALGORITHM DESIGN AND IMPLEMENTATION

In what follows, we explain the data structure and the algorithms used in this work. The implementation of the data structure and algorithm is available at GitHub[1].

4.1 Data Structure

We choose a vector, cause in terms of a time is a good option, because his insertion time is O(1) and his access time is O(1), well in this vector we are storing the data from students through a column and every student is in a row.

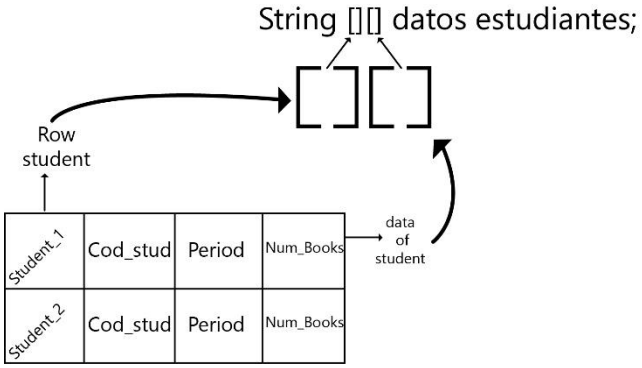


Figure 1: A vector to store the data of the students of Saber Pro based on the conditions of each one. The row refers to a student, meanwhile the column data represents those data that have been recopilated from every single student.

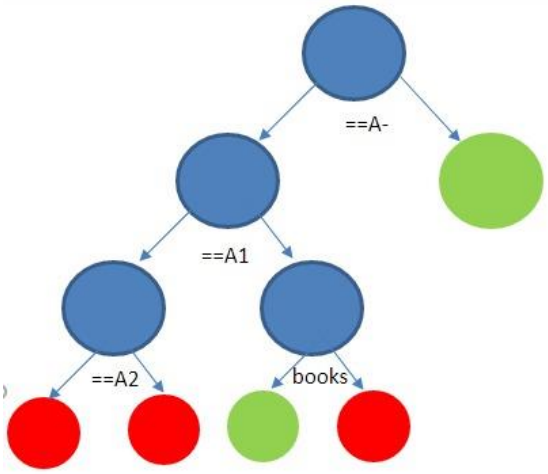
4.2 Algorithms

We created a Cart tree, for this we divided a group of students depending on a condition, the students that fulfilled this condition would be the node on the right while the rest would be the one on the left, the idea is that the condition would be the best to know if a group of students would do well on the test, for this we used the Gini impurity for each condition and thus know the best, being able to predict the students that they would do well.

[1]https://github.com/jgomezb11/ST0245-002

4.2.1 Training the model

This tree is created, first of all dividing a group of students by means of a condition, in this case the best condition that reduces the impurity of Gini, then 2 nodes are created, with the students that fulfill and those that don't, then it looks for the best condition for the following groups of students and it continues dividing, this continues making it until the leaves(final nodes), have a smaller size equal to 33% of the initial group of students.



4.2.2 Testing algorithm

Explain, briefly, how did you test the model: This is equivalent to explain how does your algorithm classifies new data after the tree is built.

4.3 Complexity analysis of the algorithms

The worst case is that all the data they give us from the students is useful to calculate the Gini impurity.

We calculate the complexities taking into account how the tree is generated (recursion) and how the best variable is sought using the Gini impurity.

The N represents the number of students.
The M represents the number of questions to the students.

Algorithm	Time Complexity
Train the decision tree	O(M* N^3*log(N))
Test the decision tree	O(N*log(N))

Table 2:
The N represents the number of students.
The M represents the number of questions to the students.

Algorithm	Memory Complexity
Train the decision tree	O(N*M)

Test the decision tree	O(N*M)

Table 3:
The N represents the number of rows.
The M represents the number of columns.

4.4 Design criteria of the algorithm

For the design of this algorithm we focused in which the tree was dynamic for each independent case, that is to say that it was able to process any type of data and was able to generate the tree from the algorithm, thus we give certainty that this algorithm will be able to be used for any topic.

5. RESULTS

5.1 Model evaluation

In this section, we present some metrics to evaluate the model. Accuracy is the ratio of number of correct predictions to the total number of input samples. Precision. is the ratio of successful students identified correctly by the model to successful students identified by the model. Finally, Recall is the ratio of successful students identified correctly by the model to successful students in the dataset.

5.1.1 Evaluation on training datasets

In what follows, we present the evaluation metrics for the training datasets in Table 3.

	Dataset 1	***Dataset 2***	***...Dataset n***
Accuracy	0.70	0.71	0.72
Precision	0.80	0.81	0.82
Recall	0.43	0.46	0.50

Table 3. Model evaluation on the training datasets.

5.1.2 Evaluation on test datasets

In what follows, we present the evaluation metrics for the test datasets in Table 4.

	Dataset 1	***Dataset 2***	***...Dataset n***
Accuracy	0.68	0.67	0.70
Precision	0.78	0.55	0.7
Recall	0.79	0.55	0.8

Table 4. Model evaluation on the test datasets.

5.2 Execution times

	Dataset 1	***Dataset 2***	***...Dataset n***
Training time	4 s	25 s	50 s
Testing time	1.16 s	8.3 s	16.56 s

Table 5: Execution time of the *Cart* algorithm for different datasets.

5.3 Memory consumption

We present memory consumption of the binary decision tree, for different datasets, in Table 6.

	Dataset 1	***Dataset 2***	***...Dataset n***
Memory consumption	591 MB	624 MB	659 MB

Table 6: Memory consumption of the binary decision tree for different datasets.

6. DISCUSSION OF THE RESULTS

In conclusion, through the data obtained, we can see that the tree is very effective in predicting data. It can be said that our algorithm has the capacity to create CART type trees, in terms of accuracy the tree presents good results, in terms of precision the tree is quite accurate when predicting. It seems to us that the complexity is fair, because the tree must be trained reviewing several data and it is the heaviest thing. It seems to us that these trees of decision are quite right at the time of so complex decisions as it is it to give scholastic scholarships, therefore we believe that if they are useful in these fields.

6.1 Future work

In the future we could improve the complexity in time because it is a quite heavy time, also we could predict better results if we used several types of trees (forests), because we would have several methods predicting the data they use.

REFERENCES

1. Kalpesh Adhatrao, Aditya Gaykar, Amiraj Dhawan, Rohit Jha and Vipul Honrao. PREDICTING STUDENTS' PERFORMANCE USING ID3 AND C4.5 CLASSIFICATION ALGORITHMS. Navi Mumbai, Maharashtra, India. September 2013. 14 pages.

2. Qasem A. Al-Radaideh, Emad M. Al-Shawakfa, and Mustafa I. Al-Najjar. Mining Student Data Using Decision Trees. Jordan. 2006. 5 pages.

3. M Krishna, Bandlamudi S B P Rani, G Kalyan Chakravarthi, B Madhavrao, S M B Chowdary.

Predicting Student Performance using Classification and Regression Trees Algorithm. 2020. 8 pages.

4. Josip Mesarić, and Dario Šebalj. Decision trees for predicting the academic success of students. Osijek, Croatia. 2016. 22 pages.