# **Decision trees to predict a student's performance on Saber Pro**

|  |  |  |  |
| --- | --- | --- | --- |
| Kevin Daniel Torres Parra  Universidad Eafit  Colombia  Ktorres2@eafit.edu.co | Vladlen Shatunov  Universidad Eafit  Colombia  vshatunov@eafit.edu.co | Miguel Correa Universidad Eafit Colombia macorream@eafit.edu.co | Mauricio Toro  Universidad Eafit  Colombia  mtorobe@eafit.edu.co |

**For each version of this report: 1. Detele all text in red. 2. Adjust spaces among words and paragraphs. 3. Change the color of all the texts to black.**

**Red text =** Comments

**Black text =** Miguel and Mauricio’s contribution

**Green text** = To complete for the 1st deliverable

**Blue text**  = To complete for the 2nd deliverable

**Violet text** = To complete for the 3rd deliverable

# **ABSTRACT**

Soon, technology will play a big role in the digital transition for education to achieve the so called Education 4.0, often mentioned as the fourth industrial revolution which will be dependent of new technologies. The main problem that we are facing is student dropouts since it’s a concern for our country´s because of the waste of resources from the institutions as they lose students. Not only that, but it also affects the country´s amount of professional/capable people for jobs.

Which is the algorithm you proposed?, What results did you achieve? , What are the conclusions of this work? Abstract should have **at most 200 words**. (*In this semester, you should summarize here execution times, memory consumption, accurracy, precision and sensibility*)

## **Keywords**

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

# **1. INTRODUCTION**

Student dropouts have reasons to do what they did, the reasons are usually a combination of factors that are generated from within the system in a social context, familiar, individual and the setting. Student dropouts involve some problems for educational institutions, because of students deciding to leave the institution, the institution will face a waste of resources.

# **1.1. Problem**

The main problem is how student dropouts affects the economy of our country by negatively affecting the development of human capital, thus generating high social costs because the human capital is less qualified. Its important to resolve this problem because of the reasons previously mentioned.

**1.2 Solution**

In this work, we focused on decision trees because they provide great explainability (*A* c*itation for this argument is missing!*). We avoid black-box methods such as neural networks, support-vector machines and random forests because they lack explainability. (*Another* c*itation for this argument is missing!*)

Explain, briefly, your solution to the problem *(In this semester, the solution is an implementation of a decision-tree algorithm to predict academic success. Which algorithm did you choose? Why?)*

**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**

**2.1 Decision trees for predicting the academic success of students**the algorithm predicts, based on information from grades in elementary and high school, whether a student will attend college successfully or not.  
Mesarić, J. (2016, 30 December). Decision trees for predicting the academic success of students. hrcak. <https://hrcak.srce.hr/index.php?id_clanak_jezik=257113&show=clanak>

**2.2 Predicting Students Final GPA Using Decision Trees**  
a educational data mining is used to predict students' final GPA based on their grades in previous courses. After pre-processing the data, they applied the J48 decision tree algorithm to discover classification rules.  
Al-Barrak, M. A., & Al-Razgan, M. (2016). Predicting student’s final GPA using decision trees: a case study. International journal of information and education technology, 6(7), 528.

**2.3 Australia-wide predictions of soil properties using decision trees**  
An algorithm predicts from data taken from the soil the properties of the soil throughout Australia. Properties considered include pH, organic carbon, total phosphorus, total nitrogen, thickness, texture and clay content.  
Henderson, B. L., Bui, E. N., Moran, C. J., & Simon, D. A. P. (2005). Australia-wide predictions of soil properties using decision trees. Geoderma, 124(3-4), 383-398.

## **2.4 Predicting Students' Performance Using ID3 And C4.5 Classification Algorithms** the algorithm predicts whether a student will drop out of a college based on his or her previous semester's grades and the college entrance exam Adhatrao, K., Gaykar, A., Dhawan, A., Jha, R., & Honrao, V. (2013). Predicting students' performance using ID3 and C4. 5 classification algorithms. *arXiv preprint arXiv:1310.2071*.

## **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/dataset](https://github.com/mauriciotoro/ST0245-Eafit/tree/master/proyecto/datasets)s .

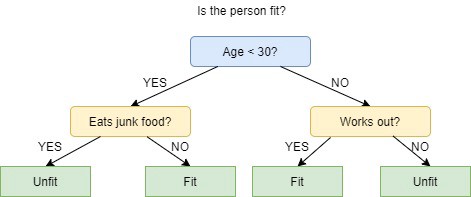
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Dataset 1** | **Dataset 2** | **Dataset 3** | **Dataset 4** | **Dataset 5** |
| **Train** | 15,000 | 45,000 | 75,000 | 105,000 | 135,000 |
| **Test** | 5,000 | 15,000 | 25,000 | 35,000 | 45,000 |

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

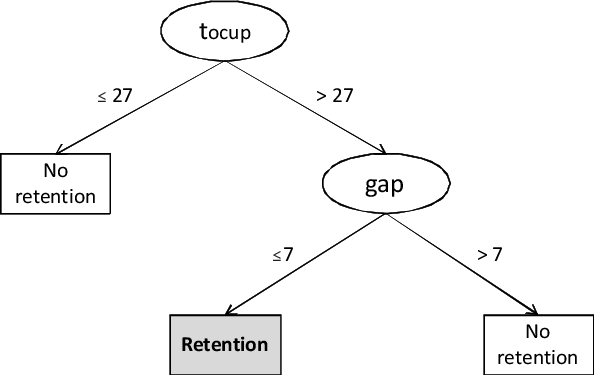
## **3.2 Decision-tree algorithm alternatives**

## 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 and CART).*

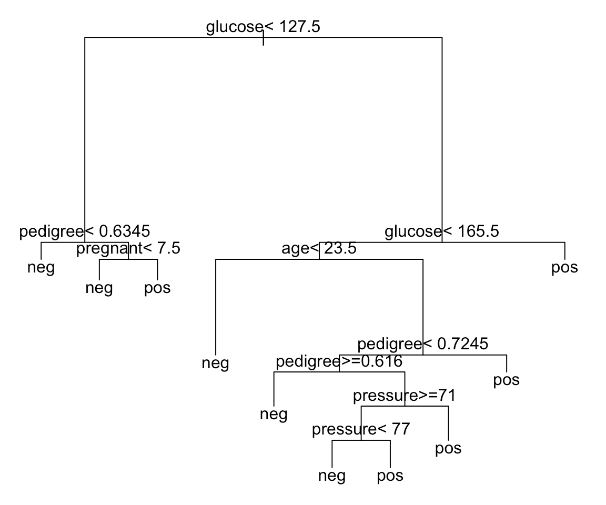
**3.2.1 ID3**  
ID3, also known as Iterative Dichotomiser 3. ID3 uses a top-down greedy approach to build a decision tree. This means that we start building the tree from the top and “greedy” means that at each iteration we select the better feature to create a node.

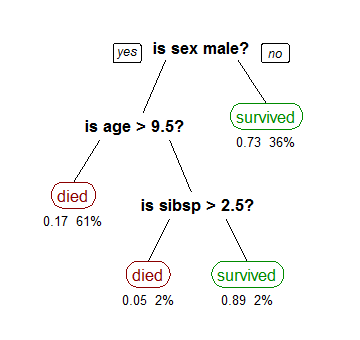


**3.2.2 C4.5**  
The C4.5 algorithm is used in Data Mining as a decision tree classifier and can later on be employed to generate a decision, all based on data. This algorithm can work with discrete and continuous data, and also it inherently employs a process named single pass pruning process which mitigates overfitting.



**3.2.3 CART**  
Please explain the algorithm, its complexity and include a vectorized Figure.CART algorithm is based on classification and regression trees by Breiman. CART consists of a tree that is constructed by splitting a node in two child nodes repeatedly.



**3.2.4 CHAID**  
CHAID makes predictions similar to regression analysis (XAID) and can also be used in detecting and identifying the interaction of variables.  


## **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]](#footnote-2).

## **4.1 Data Structure**

## Explain the data structure used to make the prediction and make a figure explaining it. Do not use figures from the Internet. *(In this semester, the data structure is a binary decision tree)*

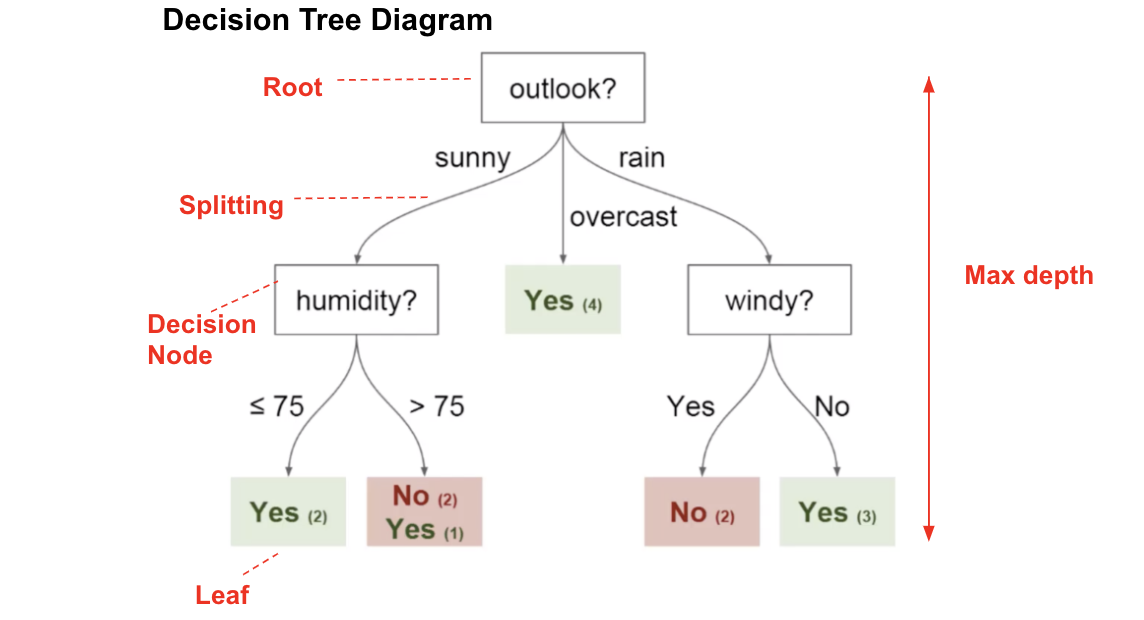
**Figure 1:** A binary decision tree to predict Saber Pro based on the results of Saber 11. Violet nodes represent those with a high probability of success, green medium probability and red a low probability of success.

**4.2 Algorithms**

Explain the design of the algorithm to solve the problem and make a figure. Do not use figures from the Internet, make your own. *(In this semester, one algorithm must be an algorithm to train a decision-tree algorithm such as ID3, C4.5, CART and the second algorithm must be an algorithm to classify new data using such a tree).*

**4.2.1 Training the model**

Explain, briefly, how did you train the model: This is equivalent to explain how does your algorithm build automatically a binary decision tree.



**Figure 2:** Training a binary decision tree using *(In this semester, one could be CART, ID3, C4.5… please choose)*. In this example, we show a model to predict whether or not to play Golf, according to weather.

**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**

Explain in your own words the analysis for the worst case using O notation. How did you calculate such complexities.

|  |  |
| --- | --- |
| **Algorithm** | **Time Complexity** |
| Train the decision tree | O(N2\*M2) |
| Test the decision tree | O(N3\*M\*2N) |

**Table 2:** Time Complexity of the training and testing algorithms. *(Please explain what do N and M mean in this problem.)*

|  |  |
| --- | --- |
| **Algorithm** | **Memory Complexity** |
| Train the decision tree | O(N\*M\*2N ) |
| Test the decision tree | O(1) |

**Table 3:** Memory Complexity of the training and testing algorithms. *(Please explain what do N and M mean in this problem.)*

**4.4 Design criteria of the algorithm**

Explain why the algorithm was designed that way. Use objective criteria. Objective criteria are based on efficiency, which is measured in terms of time and memory consumption. Examples of non-objective criteria are: “I was sick”, “it was the first data structure that I found on the Internet”, “I did it on the last day before deadline”, etc. Remember: This is 40% of the project grading.

**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* | 0.7 | 0.75 | 0.9 |
| *Precision* | 0.7 | 0.75 | 0.9 |
| *Recall* | 0.7 | 0.75 | 0.9 |

## **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* | 0.5 | 0.55 | 0.7 |
| *Precision* | 0.5 | 0.55 | 0.7 |
| *Recall* | 0.5 | 0.55 | 0.8 |

## **Table 4.** Model evaluation on the test datasets.

**5.2 Execution times**

Compute execution time for each dataset in github. Measure execution time 100 times for each dataset and report average execution time for each dataset.

## 

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Dataset 1*** | ***Dataset 2*** | ***...Dataset n*** |
| *Training time* | 10.2 s | 20.4 s | 5.1 s |
| *Testing time* | 1.1 s | 1.3 s | 3.3 s |

## **Table 5:** Execution time of the *(Please write the name of the algorithm, C4.5, ID3)* 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 | 10 MB | 20 MB | 5 MB |

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

## To measure memory consumption, you should use a profiler. An very good one for Java is VisualVM, developed by Oracle, <http://docs.oracle.com/javase/7/docs/technotes/guides/visualvm/profiler.html> For Python, use C Profiler.

## **6. DISCUSSION OF THE RESULTS**

Explain the results obtained. Is precision, accuracy and sensibility appropriate for this problem? Is the model over-fitting? Is memory consumption and time consumption appropriate? *(In this semester, according to the results, can this be applied to give scholarships or to help students with low probability of success? For which one is better?)*

**6.1 Future work**

Answer, what would you like to improve in the future? How would you like to improve your algorithm and its implementation? What about using random forest?

# **ACKNOWLEDGEMENTS**

Identify the kind of acknowledgment you want to write: for a person or for an institution. Consider the following guidelines: 1. Name of teacher is not mentioned because he is an author. 2. You should not mention websites of authors of articles that you have not contacted. 3. You should mention students, teachers from other courses that helped you.

As an example: This research was supported/partially supported by [Name of Foundation, Grant maker, Donor].

We thank for assistance with [particular technique, methodology] to [Name Surname, position, institution name] for comments that greatly improved the manuscript.

# **REFERENCES**

Reference sourced using ACM reference format. Read ACM guidelines in <http://bit.ly/2pZnE5g>

As an example, consider this two references:

1.Adobe Acrobat Reader 7, Be sure that the references sections text is Ragged Right, Not Justified. <http://www.adobe.com/products/acrobat/>.

2. Fischer, G. and Nakakoji, K. Amplifying designers’ creativity with domainoriented design environments. in Dartnall, T. ed. Artificial Intelligence and Creativity: An Interdisciplinary Approach, Kluwer Academic Publishers, Dordrecht, 1994, 343-364.

3. Sakkaf, Y. (2020, April 17). Decision Trees: ID3 Algorithm Explained | Towards Data Science. Medium. <https://towardsdatascience.com/decision-trees-for-classification-id3-algorithm-explained-89df76e72df1>

4. Saha, S. (2018, November 16). What is the C4.5 algorithm and how does it work? - Towards Data Science.

Medium. <https://towardsdatascience.com/what-is-the-c4-5-algorithm-and-how-does-it-work-2b971a9e7db0>

5. Data Mining for Predicting Traffic Congestion and Its Application to Spanish Data - Scientific Figure on ResearchGate. Available from: https://www.researchgate.net/figure/Tree-generated-by-the-C45-algorithm-for-the-case-of-60-minutes-of-anticipation\_fig1\_274700559 [accessed 16 Aug, 2020]

6. (2018, March 11). CART Model: Decision Tree Essentials. Articles - STHDA. <http://www.sthda.com/english/articles/35-statistical-machine-learning-essentials/141-cart-model-decision-tree-essentials/>

7. Wikipedia contributors. (2019, 5 enero). Chi-square automatic interaction detection. Wikipedia. https://en.wikipedia.org/wiki/Chi-square\_automatic\_interaction\_detection

1. <http://www.github.com/> ????????? /proyecto/ [↑](#footnote-ref-2)