DECISION TREES TO PREACH IF A STUDENT WILL HAVE A SCORE IN THE "SABER PRO" TEST, ABOUT THE AVERAGE.

Juan Pablo Cortes Gonzalez
Universidad Eafit
Colombia
jpcortesg@eafit.edu.co

Yhilmar Andres Chaverra Castaño
Universidad Eafit
Colombia
yachaverrc@eafit.edu.co

Mauricio Toro
Universidad Eafit
Colombia
mtorobe@eafit.edu.co

ABSTRACT

It is no secret that in the future technology will be the most solid and important branch in Colombia's educational growth. Currently, only cases of academic dropout have been studied and the factors that drive it, with the help of technology, have found algorithms that lead to conclusions about dropping out.

In comparison with these studies there are few answers that are found for the academic success of higher education, there is no unit of measure to know the magnitude of academic success, it can be relative in each person. That is why in this work we will define academic success as the possibility that a student has to obtain a higher than average score on the "saber pro" exam.

1. INTRODUCTION

every year, a number of students do the "know pro" tests but the background is not very good with respect to what is expected of the students in the country's universities, there is a large database with which you can do a prediction of the possibility of success in these tests, this would be a great tool to know your chances of success, depending on a series of factors or variables which subject the student to a result, this would help to discover according to the predictive model which These are the factors that most affect the test score and improve the educational performance of many students.

It is important to clarify that many variables that are derogatory with race, gender, belief etc. are excluded so that the result is not exclusive or generates social gaps or discrimination.

2. PROBLEM

The problem to be solved is to create an algorithm through decision trees and based on the results of the "ICFES" tests to predict whether a student will have a good grade on the "saber pro" tests.

The solution will work with different variables, for example, age, ICFES score, ETC university degree. With these variables, you can create the decision tree to know the future of "know pro" tests. In addition, there is a variable that communicates whether the student obtained a good score or not in the "know pro" tests.

3. RELATED WORK

3.1 Algorithm ID3

Is a greedy constructive algorithm to get decision trees.

Place in the root node of the tree the attribute that by itself better.

This algorithm places a root node in the tree as a starting mode which, in case it is better to classify the training set, creates a child node for each value of the attribute, assigns an example value to the corresponding node. repeat the steps with the example nodes associated with each of the nodes.

It has a linear complexity and grows with the number of instances, and exponentially with the attributes and produces trees that overfit training instances. [1]

Pseudocode:

node = DecisionTreeNode(examples)

handle target attributes with arbitrary labels

dictionary = summarizeExamples(examples, targetAttribute)

for key in dictionary:

if dictionary[key] == total number of examples

node.label = key

return node

test for number of examples to avoid overfitting

if attributes is empty or number of examples < minimum allowed per branch:

node.label = most common value in examples

return node

bestA = the attribute with the most information gain

node.decision = bestA

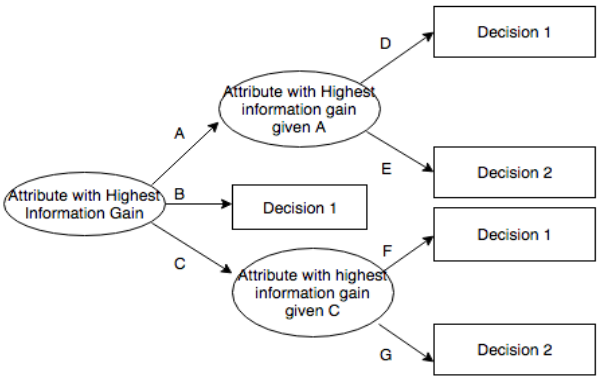
for each possible value v of bestA:

subset = the subset of examples that have value v for bestA

if subset is not empty:

node.addBranch(id3(subset, targetAttribute, attributes-bestA))

return node [4]



3.2 Algorithm C4.5

Induction algorithm that uses a gain radius which works to select the attribute, which in the algorithm are discrete and continuous.

continuous: they are evaluated considering the threshold and a set of examples is divided into 2 subsets, the first one contains less than or equal to the threshold and the second the higher values. [3]

Pseudocode:

1. Check for base case.

2. For each attribute a.

Find the normalized information gain.

3. Let a_best be the attribute with the

Highest normalized information gain

4. Create a decision node that splits on a_best.

5. Recursion the sublist obtained by splitting on a_best and and those nodes as children of node.[5]

example C4.5 decision tree with four condition nodes:

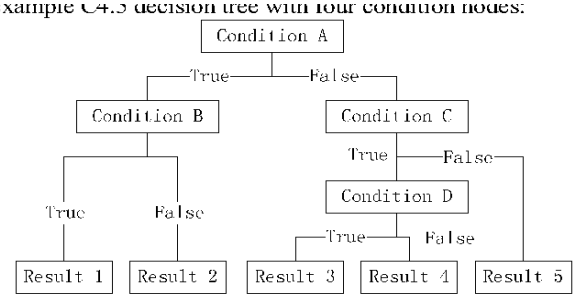


Figure 2. An example of C4.5 decision tree with four condition node

3.3 REPTree

This algorithm prunes the trees based on using back-fitting method and reduced-error pruning. As in C4.5, this algorithm can also work with missing or uncompleting values by splitting the identical instances into pieces. [5]

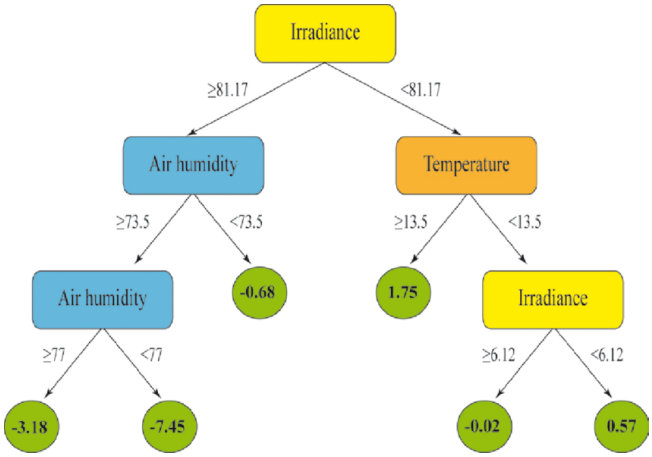
This is a fast learning decision tree; the operation could explain how:

- Build the decision tree.
- List the variables using the information found.
- reduce the error that arises from the variance.

This process generates several trees, then the algorithm selects among all the trees made, in addition to this the algorithm is designed to prune the tree.

The reduced error pruning tree (REP) as a learning algorithm of the decision tree can be considered a fast classifier. [6]

This algorithm also works with lack of values, or incomplete
values



3.4 Random Tree

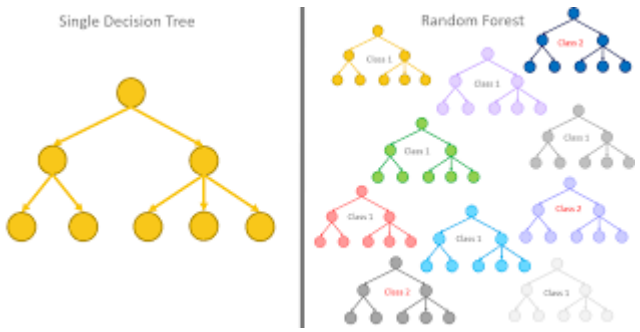
The random tree algorithm selects a test on the basis of a specific number of random features at each node without pruning. [7]

Only use subsets of random attributes, for each division.

For each division of the tree only the subset of attributes is available. It works similarly to C4.5 or CART, but select the set of attributes you will use.

It has the advantage of being easy to interpret. each inner node of the tree is an attribute of the subset of the division.

The merit of building a random tree is the efficiency of training and minimum memory requirements. [8]



REFERENCES

[1]Arboles de decisión Christopher Expósito Izquierdo Airam Expósito Márquez Israel López Plata Belén Melian Batista J. Marcos Moreno Vega 30 – 41.

[2] ID3 Algorithm Abbas Rizvi CS157 B Spring 2010

[3] What is the C4.5 algorithm and how does it work? Sumit Saha Aug 20, 2018 4. Decision Tree Pseudocode www.cs.swarthmore.edu.

[4] Implementation of Decision Tree Classifier using WEKA tool September 19, 2017

[5] S. K. Jayanthi and S. Sasikala. "REPTree Classifier For Identifying LinkSpam in Web Search Engines." IJSC 3.2 (2013): pp. 498-505.

[6] I. H. Witten, E. Frank, and M.A. Hall. "Data Mining: Practical MachineLearning Tools and Techniques." Morgan Kaufmann, 2016.

[7] I. H. Witten, E. Frank, and M.A. Hall. "Data Mining: Practical MachineLearning Tools and Techniques." Morgan Kaufmann, 2016.

[8] G. Jagannathan, K. Pillaipakkamnatt, and R. N. Wright. "A practical Differentially Private Random Decision Tree Classifier." Data Mining Workshops, 2009. ICDMW'09. IEEE International Conference on. IEEE, 2009

.