ALGORITHM FOR THE PREDICTION ON THE ICFES SABER PRO EXAM RESULTS

Samuel Ceballos Posada
Universidad Eafit
Colombia
sceballosp@eafit.edu.co

Pedro Botero Aristizábal
Universidad Eafit
Colombia
pboteroa@eafit.edu.co

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

SUMMARY

Para escribirlo pueden dar respuesta a estas preguntas: ¿Cuál es el problema?, ¿Por qué es importante el problema?, ¿Qué problemas relacionados hay?

The Icfes Saber Pro Exam is a test that every student who is part of a university has to do in order to graduate. This exam, although not helping much for the professional future of the student, is a way for universities to understand their weaknesses and look for the points I which they should focus their academic programs.

The algorithm uses previous information and exams to predict how each student will do on the Saber Pro, by analyzing their result on the Icfes exam made at the end of High School and other information collected including family wealthness, time using technology devices, city of living, etc. and putting the students on certain group using classification that determine how good he/she will do on the exam. This algorithm has previously been used by Universities in other countries to look for further ways of improvement on their curriculums.

Keywords

Exams; Predictions; Data Structures; Array of Arrays; Complexity.

ACM Classification Keywords

•Applied computing~ Law, social and behavioral sciences ~ Sociology

•Software and its engineering ~ Software organization and properties ~ Contextual software domains ~ Operating systems ~ File systems management

1. INTRODUCTION

Universities operate in a very competitive environment, they are constantly confronting other institutions in order to attract the best students and making sure they stay. Also, the Saber Pro exam results play an important factor in university rankings, so they care a lot about student performance.

It is a fact that universities collect a lot of data from each student when they are applying. More specifically, universities in Colombia ask for the results of the ICFES exam, which is never used again after the student is accepted in the institution. With the implementation of a decision tree algorithm, universities could use the data from the ICFES exam to determine how a student will perform in the Saber Pro exam. This way universities could see what are the areas in which students struggle the most, thus make adjustments to their curriculum in order to improve performance in the exam.

2. PROBLEM

The main objective of this project is to establish an algorithm that is able to predict how a person will do on the ICFES Saber Pro exam based on information that includes habits, results on previous exams and basic information.

The solution of this project will help universities to discover where their weaknesses are at and try to implement new techniques to help their students grow better on those aspects.

3. SIMILAR WORK

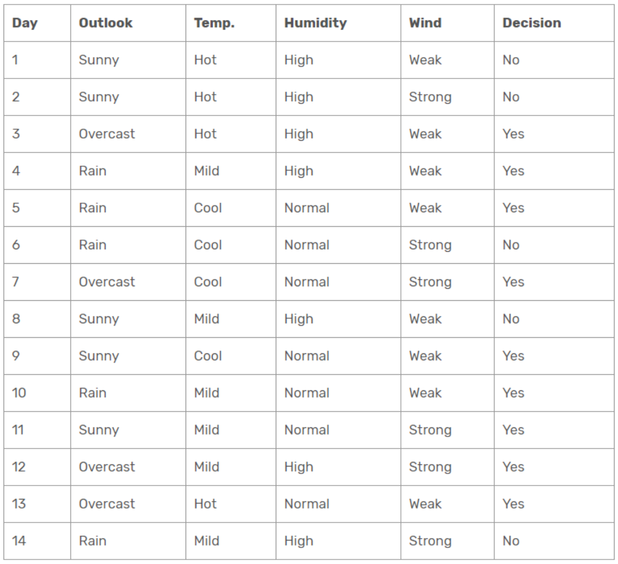
3.1 ID3

The ID3 algorithm, also known as Iterative Dichotomiser 3 is a recursive algorithm invented by Ross Quinlan in 1975, used to generate a decision tree from a fixed set of examples attempting to create the smallest tree possible. The resulting tree is used to classify future samples. It is most commonly used in machine learning and natural language processing.

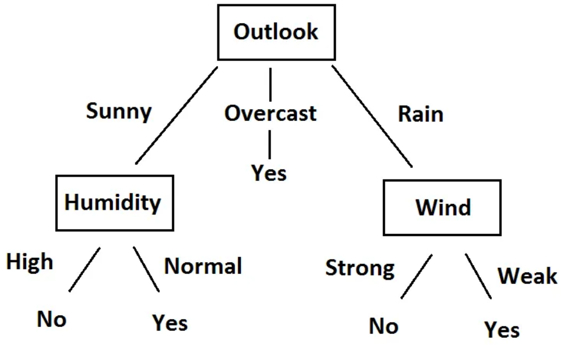
ID3 works with the following steps:

1. Calculate the entropy of the attributes.
2. Calculate the Information Gain for the attributes, then split the set into subsets using the attribute with the minimum entropy.
3. Find the attribute with the maximum information gain.
4. Recourse until the desired tree is found.

For example, the table is a data set of factors to play tennis outside, which produces the following tree:



Day	Outlook	Temp.	Humidity	Wind	Decision
1	Sunny	Hot	High	Weak	No
2	Sunny	Hot	High	Strong	No
3	Overcast	Hot	High	Weak	Yes
4	Rain	Mild	High	Weak	Yes
5	Rain	Cool	Normal	Weak	Yes
6	Rain	Cool	Normal	Strong	No
7	Overcast	Cool	Normal	Strong	Yes
8	Sunny	Mild	High	Weak	No
9	Sunny	Cool	Normal	Weak	Yes
10	Rain	Mild	Normal	Weak	Yes
11	Sunny	Mild	Normal	Strong	Yes
12	Overcast	Mild	High	Strong	Yes
13	Overcast	Hot	Normal	Weak	Yes
14	Rain	Mild	High	Strong	No



Even though ID3 is the most common decision tree algorithm, it has some disadvantages. For example, all attributes must be nominal values, the data set cannot have missing data and the algorithm tends to fall into overfitting.

3.2 C4.5

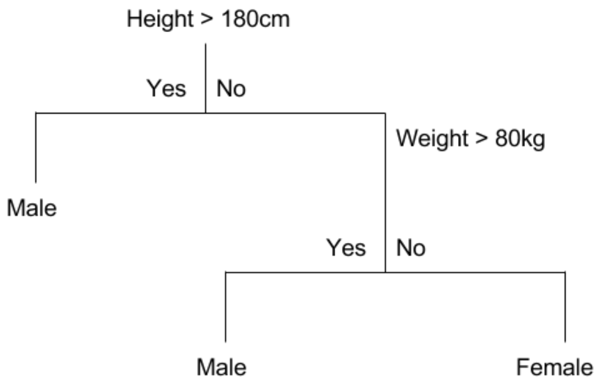
C4.5 is another algorithm created by Ross Quinlan based on ID3. It generates a decision tree where each node splits the classes based on the Information Gain. The attribute with the highest Information gain is used as the splitting criteria, just like ID3.

The main difference between the two is that C4.5 can use discrete and continuous attributes and it can handle missing attribute values.

3.3 CART

First introduced by Leo Breiman during the 1980's. The CART algorithm, also known as Classification And Regression Algorithm, is used to predict results based on different types of modeling problems. Due to its simplicity, it is commonly known as a decision tree, but recently many platforms have started to use the term CART when referring to it. This algorithm is commonly used as a foundation for more advanced algorithms, like bagged decision trees, random forest and boosted decision trees.

The CART model is represented by a binary tree. Where each root node holds an input variable which splits into boolean values (True or False). Then, the leaf node contains output variables which are the ones used to make a prediction on the data. The purity of the model can be tested by using the Gini Index Function, which uses the probabilities to determine how effective the model is. The lower the index, the purest the model and vice versa. [1]

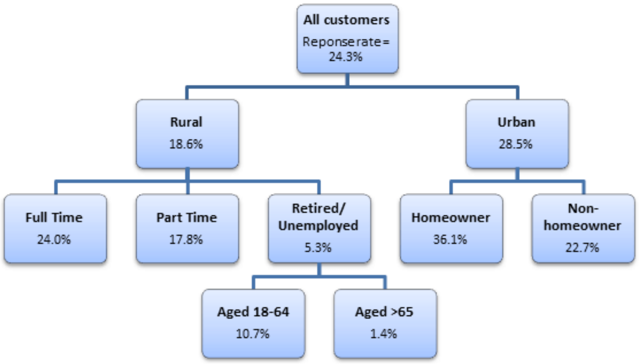


One example of the CART algorithm being used is an investigation made 10 years ago by the DG Vaishnav College in Tamil Nadu, India. Where researchers made a CART model to determine and classify blood donors according to the amount of times they have donated blood and how much time passes between donations. This showed that most of the people that donate higher amounts of blood are Regular Voluntary Donors, while Lapsed Voluntary Donors and New Voluntary Donors made smaller donations in terms of quantity. [2]

3.4 CHAID

The CHAID algorithm (Chi-square Automatic Interaction Detector) was first brought into attention by Gordon V. Kass in 1980. This method is used to determine and discover relationships between categorical response and predictor variables, in order to discover patterns that may be difficult to visualize in an unorganized manner. It then uses the chi-square test to measure and analyze how different the model data is compared to the expectations it had.

This method is commonly used on direct marketing to determine how a certain group of customers may respond to a campaign or advertisement made by a specific enterprise.



4. ARRAY OF ARRAYS

An array of arrays is a very simple way to organize large quantities of information, its operations are not very complicated and follow a logic that is easy to understand. Also, they permit to append information at any memory space inside the array making it easier to control.

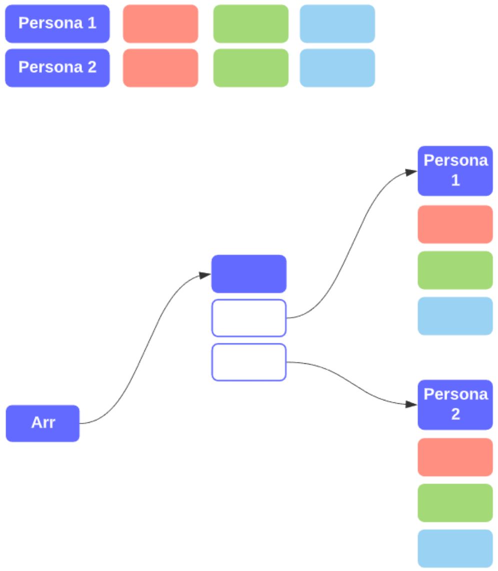


Figure 1: Array of arrays. Each inner array contains all the info needed for one person. Each color represents a different variable.

4.1 Operations of the data structure

4.1.1. Data Append

By appending data into an array, you can insert information at whatever point you desire and the structure will change according to the position of the new information.

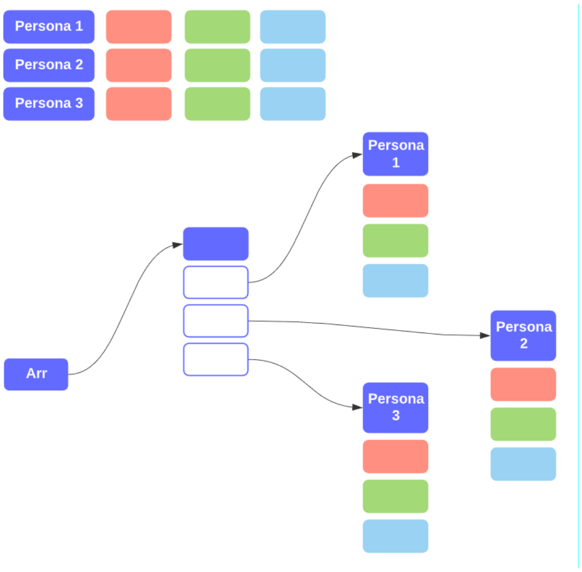


Figure 2: Data Append operation on the previous array.

4.2 Design criteria of the data structure

An array of arrays can be quite difficult to work on in terms of complexity, memory and time as it has to analyze each array inside the big array to determine whether something is actually inside that array. On the other hand, it is very easy to understand how it works and all its uses.

4.3 Complexity analysis

Method	Complexity
Data Append	O(n)

Table 1: Report of the complexity analysis

4.4 Execution time

Create	
Dataset	Runtime
1	234.0805 ms
2	709.5405 ms
3	3596.085 ms

Table 2: Time taken for each operation of the data structure

Dataset	Best time (ms)	Worst time (ms)	Average time (ms)
1	227.04	248.95	234.0805
2	663.41	1117.52	709.5405
3	3510.87	3859.73	3596.085

Table 3: Best, worst and average runtime for each dataset.

4.5 Memory used

Create	
Dataset	Memory use
1	13.1855 mb
2	13.274 mb
3	13.4595 mb

Table 4: Memory used for each operation of the data structure and for each data set data sets.

Dataset	Best memory (mb)	Worst memory (mb)	Average memory (mb)
1	13.12	13.3	13.1855
2	13.25	13.31	13.274
3	13.32	13.53	13.4595

Table 5: Best, worst and average memory use for each dataset.

4.6 Result analysis

In terms of efficiency, the data structure chosen for this project works on what would be called a good manner. It is very easy to understand and every aspect in terms of memory and runtime isn’t bad. For this kind of project, the Array of arrays is a very good data structure.

Dataset	Memory use	Runtime
1	13.1855 mb	234.0805 ms
2	13.274 mb	709.5405 ms
3	13.4595 mb	3596.085 ms

Table 6: Analysis of the memory used and the runtime.

REFERENCIAS

1. Jason Brownlee. 2019. Classification And Regression Trees for Machine Learning. (August 2019). Retrieved February 9, 2020 from https://machinelearningmastery.com/classification-and-regression-trees-for-machine-learning/

2. T. Santhanam and Shyam Sundaram. Application of CART Algorithm in Blood Donors Classification. Journal of Computer Science 6, Tamil Nadu, 2010 from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.165.8749&rep=rep1&type=pdf

3. Sarah Littler. 2018. CHAID (Chi-square Automatic Interaction Detector). (May 2018). Retrieved February 9, 2020 from https://select-statistics.co.uk/blog/chaid-chi-square-automatic-interaction-detector/

4. Sefik Serengil. 2017. A Step by Step ID3 Decision Tree Example. (November 2017). Retrieved February 9, 2020 from https://sefiks.com/2017/11/20/a-step-by-step-id3-decision-tree-example/

5. Sefik Serengil. 2018. A Step By Step C4.5 Decision Tree Example. (May 2018). Retrieved February 9, 2020 from https://sefiks.com/2018/05/13/a-step-by-step-c4-5-decision-tree-example/