PREDICTION OF COLOMBIAN STUDENTS' ACADEMIC SUCCESS USING DECISION TREES ALGORITHMS

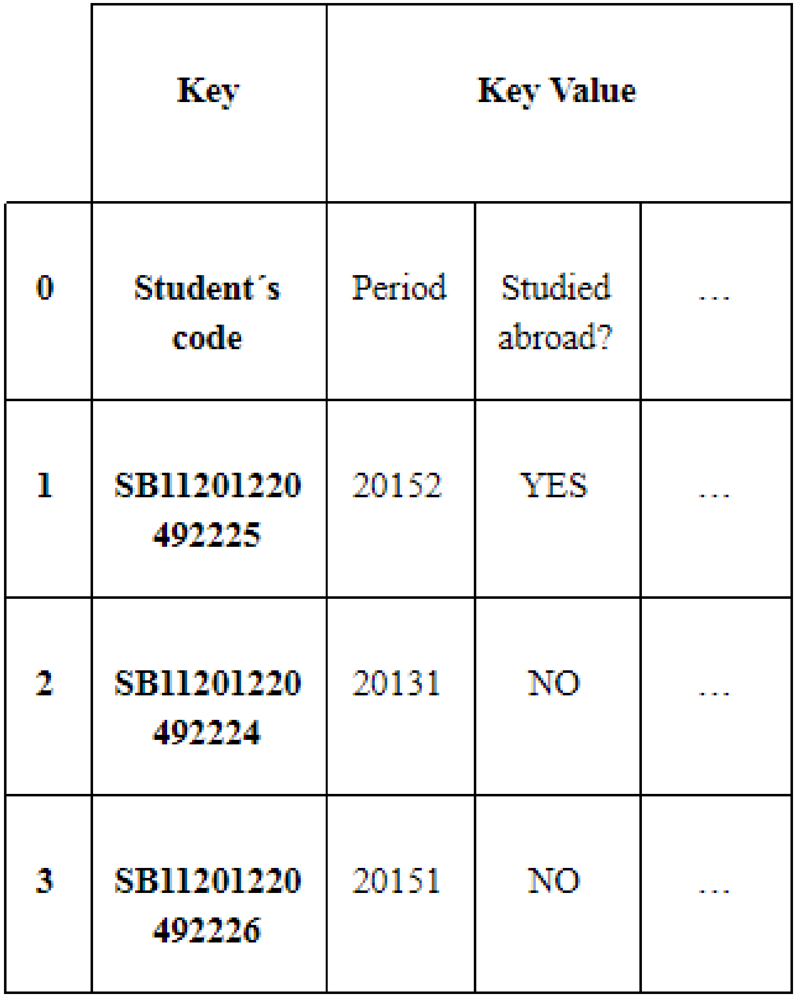
Juliana Restrepo Tobar

Juliana Restrepo Tobar
David Vergara Patiño
Andrea Carvajal Maldonado

Medellín, June 2020

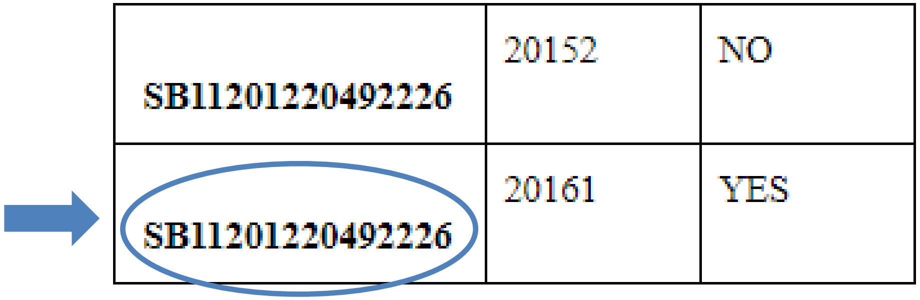
Designed Data Structure
1. Data Frame

Figure 1: Data Frame of students and their information. The student’s code is the key and all information such as ID, age, genre is they key value.



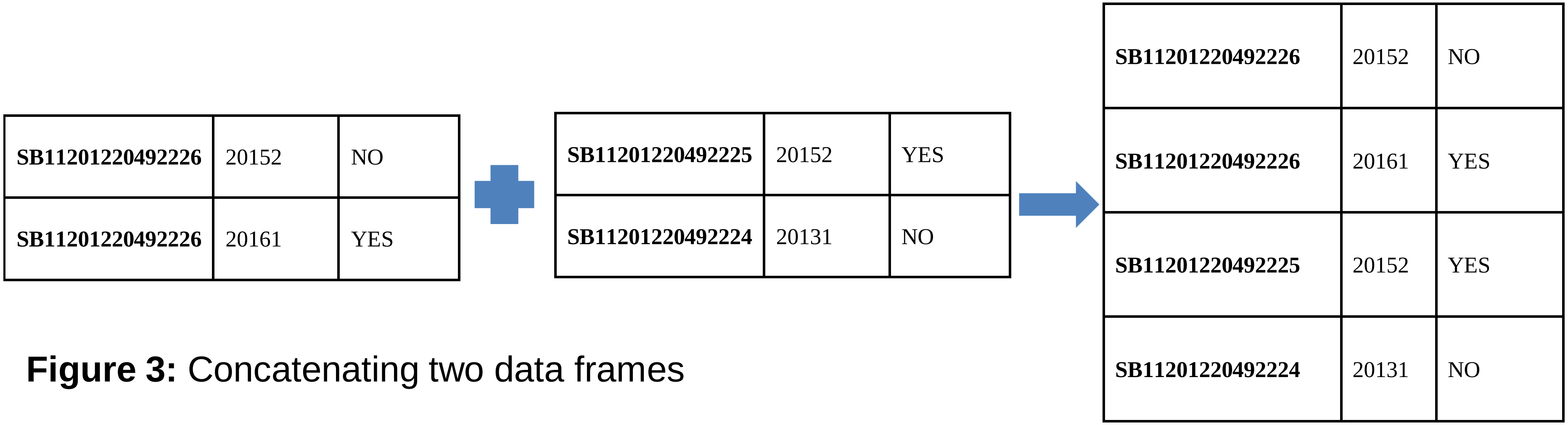
	Key	**Key Value**		
0	**Student´s code**	Period	Studied abroad?	…
1	**SB11201220 492225**	20152	YES	…
2	**SB11201220 492224**	20131	NO	…
3	**SB11201220 492226**	20151	NO	…

Operations



SB11201220492226	20152	NO
SB11201220492226	20161	YES

Figure 2: Information access operation



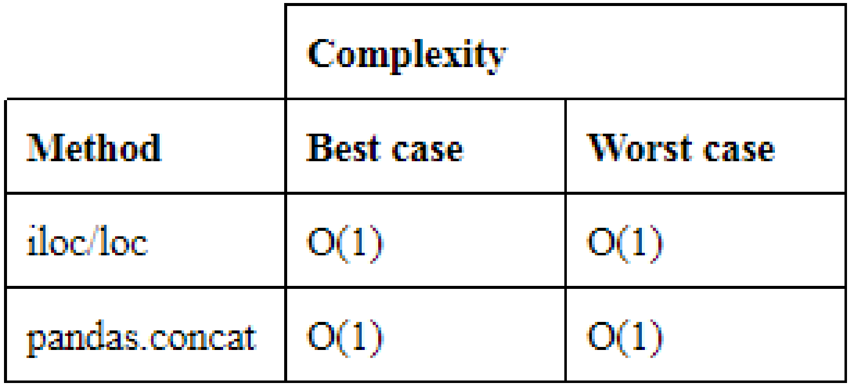
SB11201220492226	20152	NO
SB11201220492226	20161	YES

SB11201220492225	20152	YES
SB11201220492224	20131	NO

SB11201220492226	20152	NO
SB11201220492226	20161	YES
SB11201220492225	20152	YES
SB11201220492224	20131	NO

Figure 3: Concatenating two data frames

Time and Complexity Analysis



	Complexity	
Method	Best case	Worst case
iloc/loc	O(1)	O(1)
pandas.concat	O(1)	O(1)

Table 1: Complexity of some operations in Data Frame

Structure	Data Frame
Space	59869502 bytes-539116653 bytes
Time of creation	0.41s-4.43s
Time of access	0.000216s-0.000648s
Time of concatenation	0.75s-0.80s

Table 2: Results analysis

Design Criteria

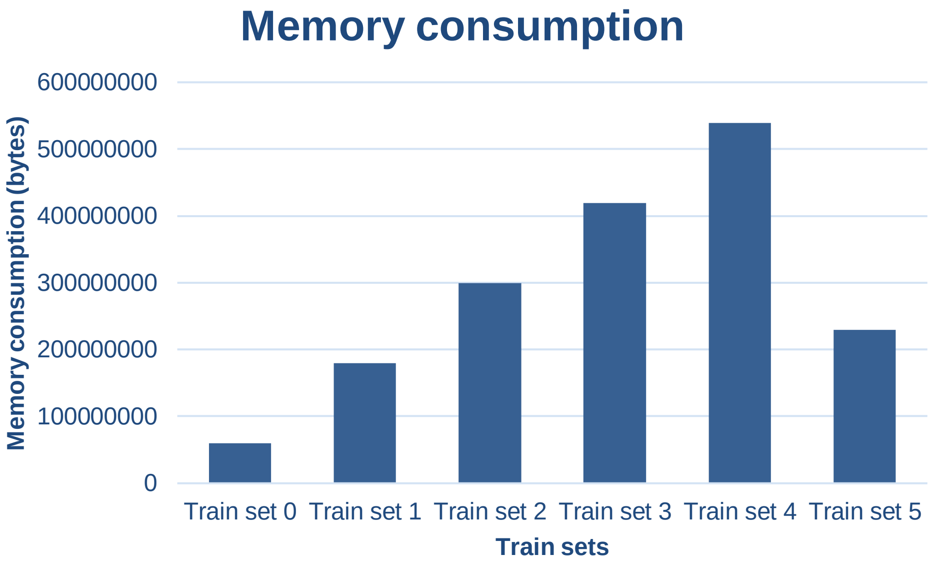
- Since the format of the input data is a matrix, it is very convenient to store it in a table-like structure.

- Data frames allow storing different data types.

- One of the most used data structures when working with big data and AI.

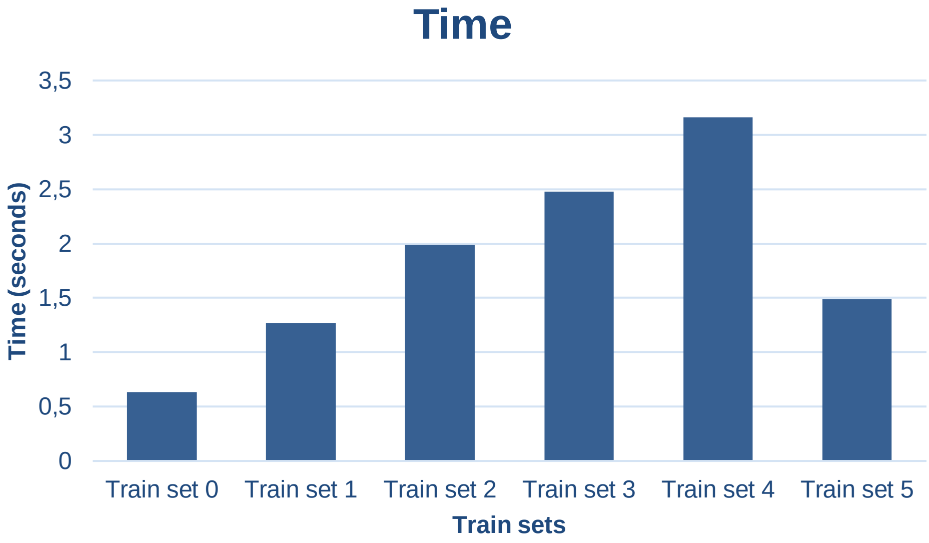
- Efficient and easy way of organizing and managing a large volume of information.

Time and Memory Consumption



Graph 2: Memory consumption of the data

Graph 1: Time taken to create the data frame



Designed Data Structure
2. Binary Decision Tree

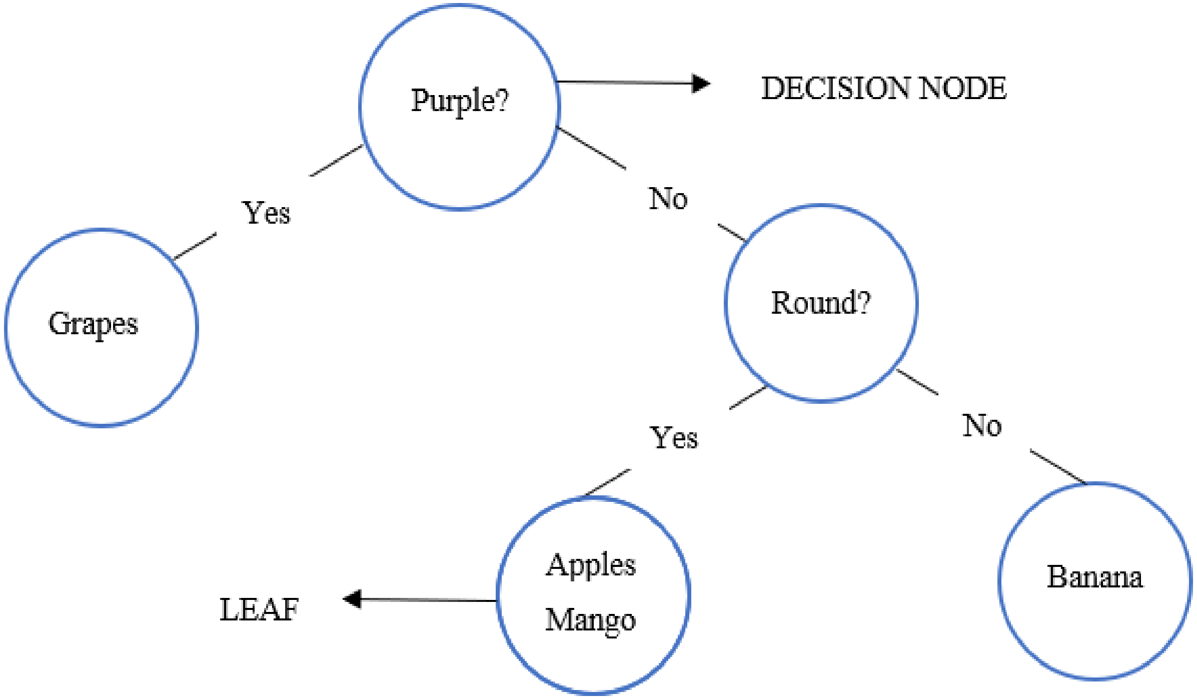


Figure 3: Binary Decision Tree.

Data Structure Operations

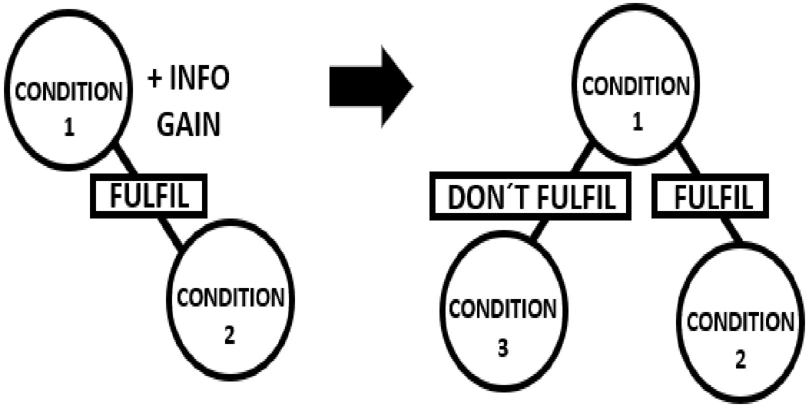


Figure 3: Creation of a binary decision tree

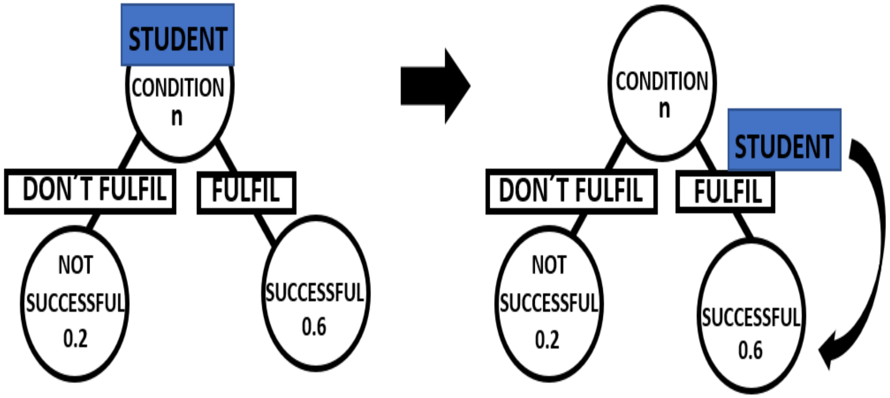


Figure 4: Classifying a student

Time and Complexity Analysis

	Complexity	
Operation	**Average case**	**Worst case**
Creation	$O(2^x * n * m)$	$O(2^x * n * m)$
Classify	$O(\log_2 n)$	$O(\log_2 n)$

Table 3: Complexity of the decision tree's operations.

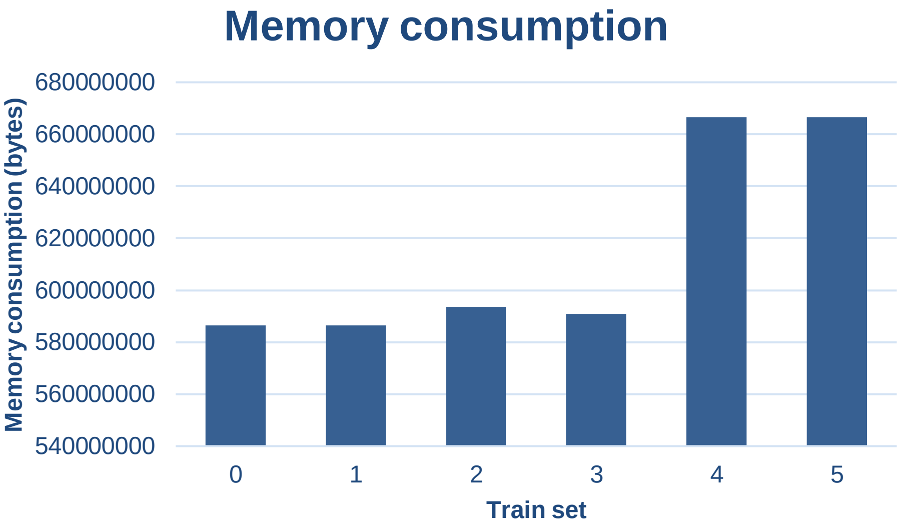
Structure	**Binary Decision Tree**
Space	586452992-666603520 bytes
Time of creation	253.796-813.418 s
Time of classifying	0.706-2.148s

Table 4: Results analysis.

Design Criteria

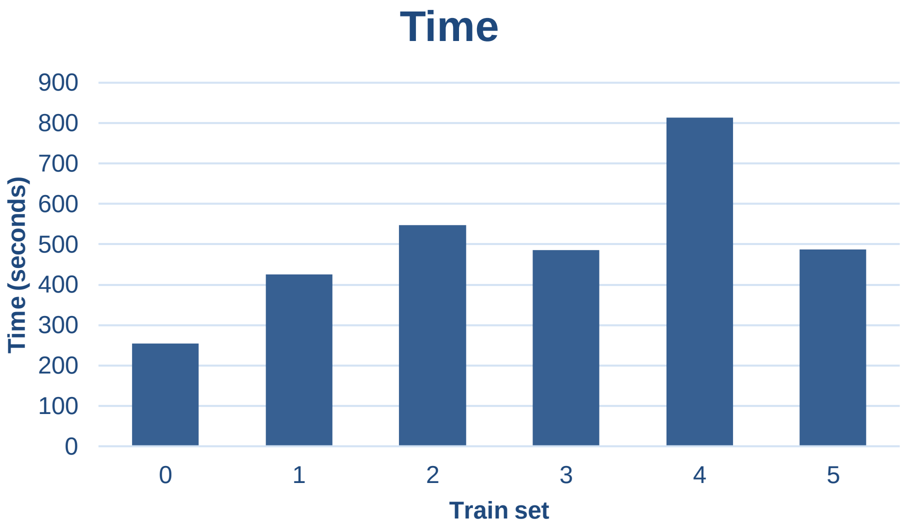
- CART is the latest, most used decision tree algorithm with C5.
- Consumes less memory and has a lower probability of misclassification than C5.0.

Time and Memory Consumption

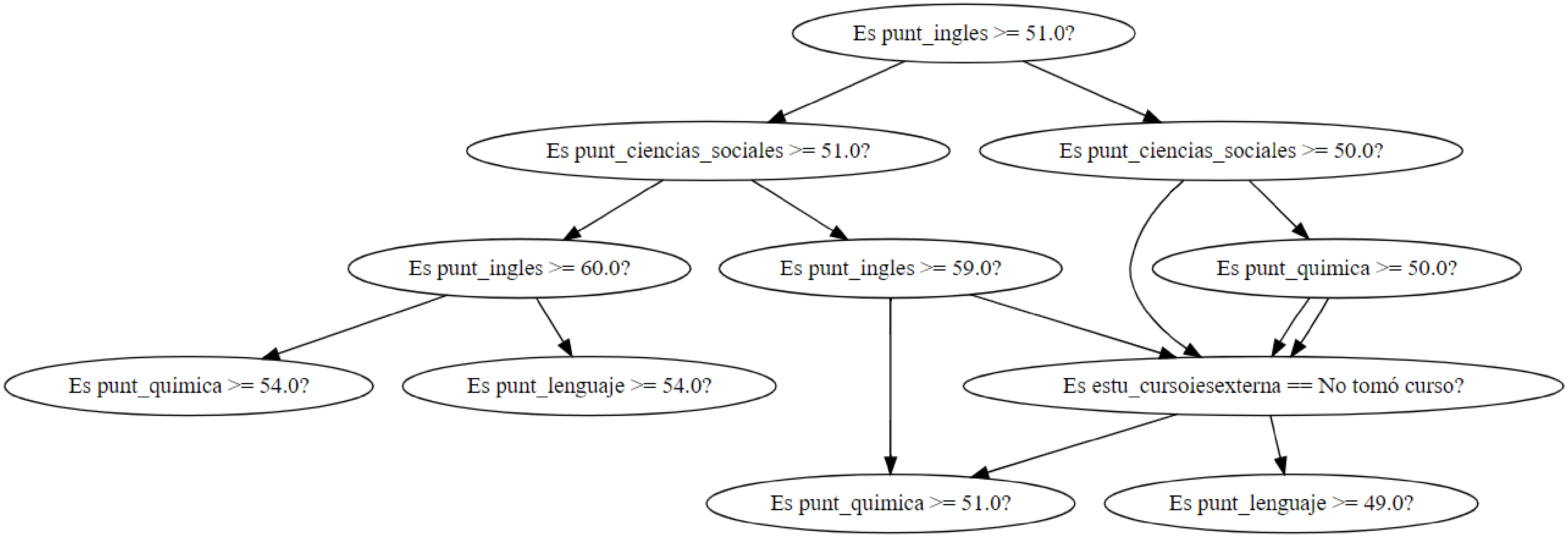


Graph 4: Memory consumption of the data

Graph 3: Time taken to create the decision tree



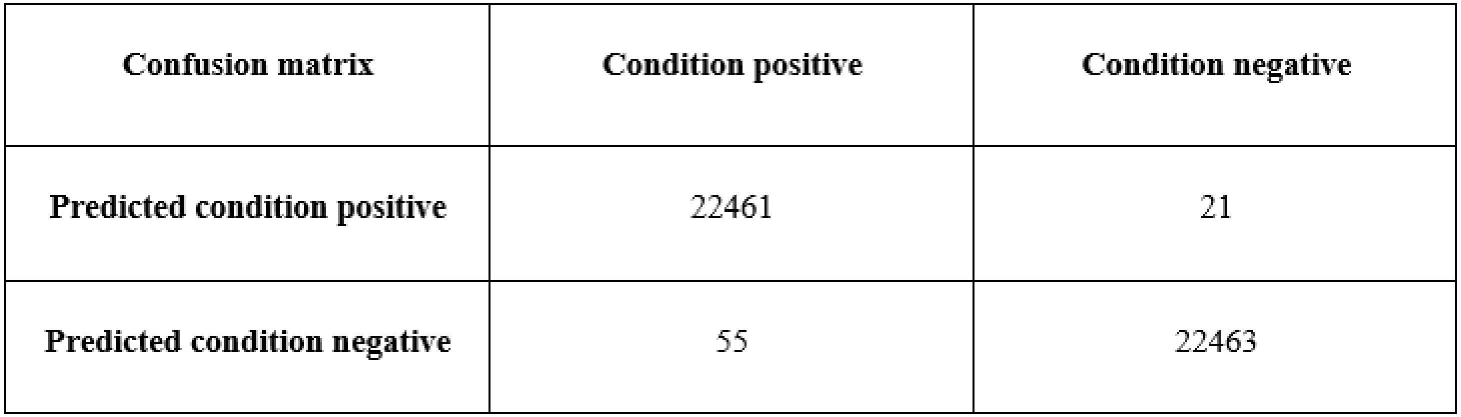
Implementation


Es punt_ingles >= 51.0?
Es punt_ciencias_sociales >= 51.0?
Es punt_ciencias_sociales >= 50.0?
Es punt_ingles >= 60.0?
Es punt_ingles >= 59.0?
Es punt_quimica >= 50.0?
Es punt_quimica >= 54.0?
Es punt_lenguaje >= 54.0?
Es estu_cursoiesexterna == No tomó curso?
Es punt_quimica >= 51.0?
Es punt_lenguaje >= 49.0?

Implementation

GitHub link: https://github.com/jrestrepot/ST0245-032/blob/master/proyecto/codigo/proyecto_final_datos1.py

Accuracy percentage: 99.8%



Confusion matrix	Condition positive	Condition negative
Predicted condition positive	22461	21
Predicted condition negative	55	22463

Table 5 : Time taken to create the decision tree

Applications

- Universities can use it to easily recognize excellent students and contact them for admission or scholarship processes.

- Also can be used to predict which active students are more likely to have a low score on their exam so they can provide them with special attention.

- Can be applied by the government to reinforce aspects in the educational system.