

LA PREDICCIÓN DE UN ÁRBOL SOBRE TU ÉXITO



Presentación del Equipo



Daniel
Arango



Jean Paul
Cano



Miguel
Correa

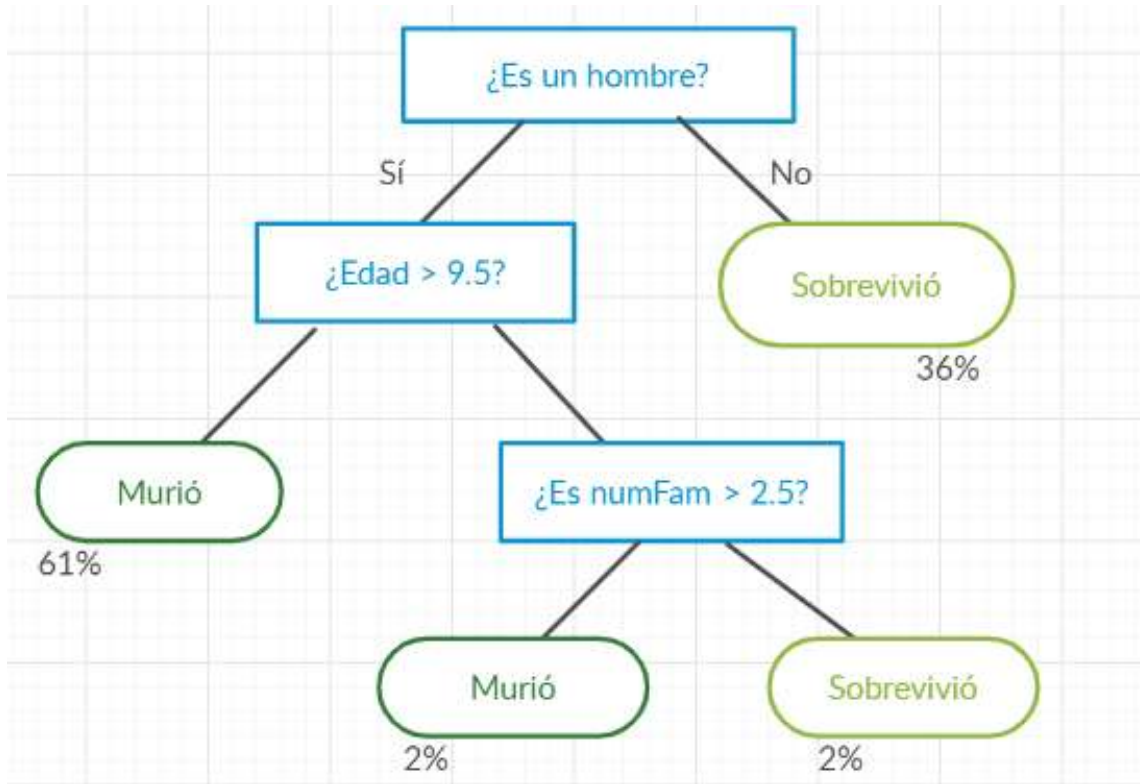


Mauricio
Toro



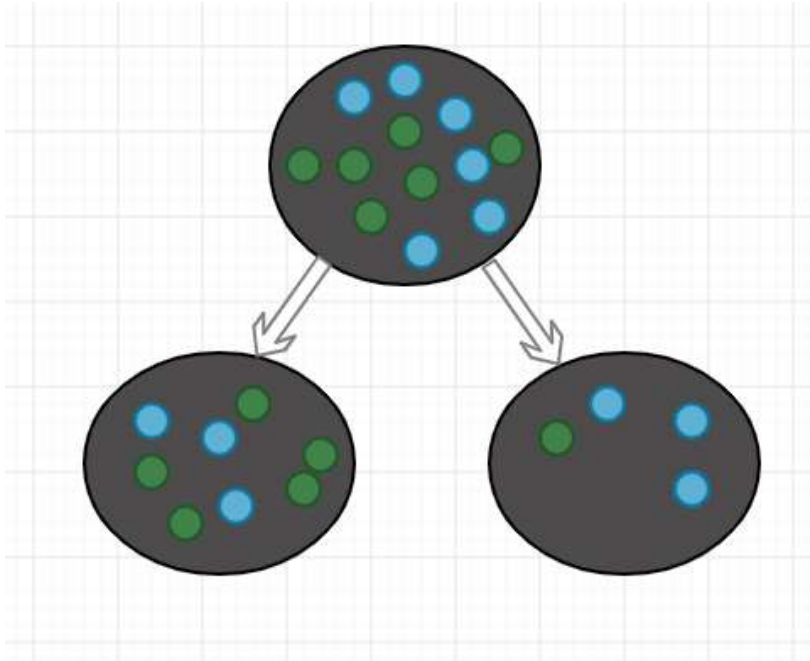
<http://github.com/Drealm-bot/ST0245-002/proyecto/>

Diseño del Algoritmo

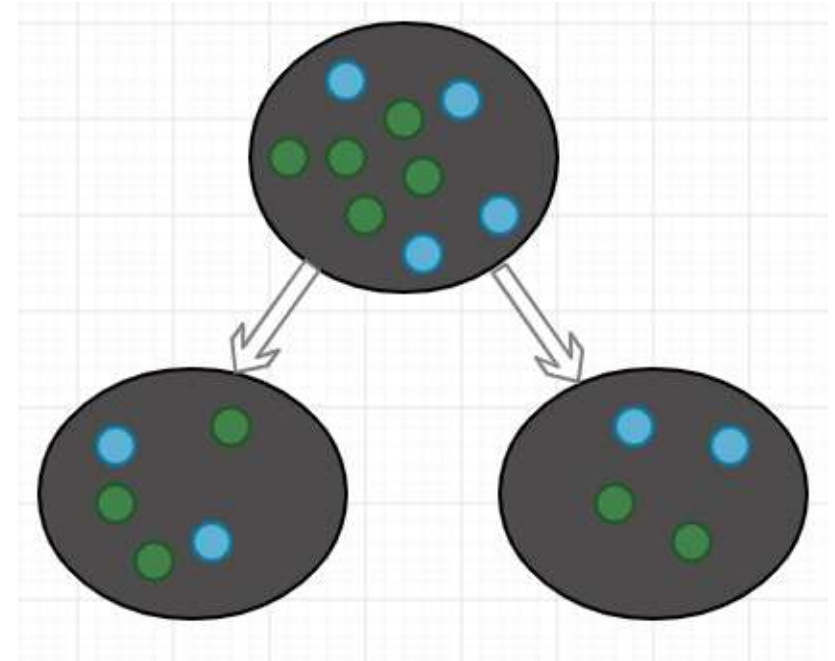


Algoritmo para construir un árbol binario de decisión CART. En este ejemplo, mostramos un modelo para predecir si un pasajero del titanic sobrevivió o no a su hundimiento.

División de un nodo



Esta división está basada en la condición “NotaLenguaje > 45.” Para este caso, la impureza Gini de la izquierda es 0.47, la impureza Gini de la derecha es 0.37 y la impureza ponderada es de 0.43.



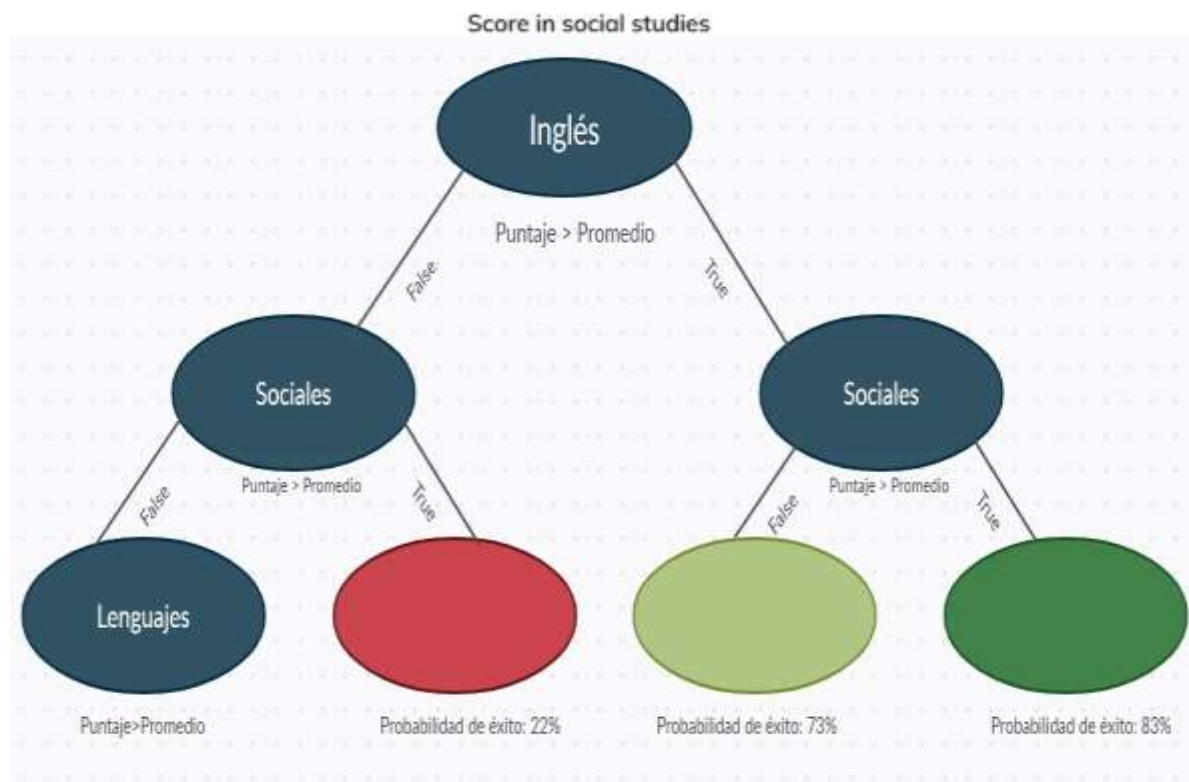
Esta división está basada en la condición “NotaMath == 5.” Para este caso, la impureza Gini de la izquierda es 0.48, la impureza Gini de la derecha es 0.5 y la impureza ponderada es 0.48.

	Complejidad en tiempo	Complejidad en memoria
Entrenamiento del modelo	$O(N*M)$	$O(N*M)$
Validación del modelo	$O(N*M)$	$O(N)$

Complejidad en tiempo y memoria del algoritmo CART, donde N es el número de datos que recibe el programa y M , que será constante, es el número de nodos en árbol de decisiones.



Modelo de Árbol de Decisión



Características Más Relevantes



Inglés



Ciencias Sociales



Lenguajes

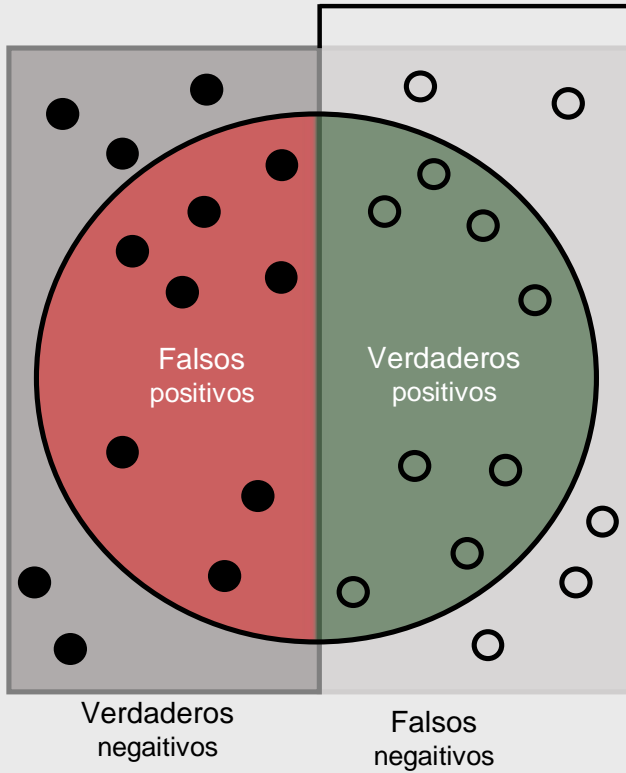
Un árbol de decisión para predecir el resultado del Saber Pro usando los resultados del Saber 11. El verde representa nodos con alta probabilidad de éxito; verde lima media probabilidad; y rojo baja probabilidad.

Métricas de Evaluación



*Elementos
Seleccionados*

*Elementos
Relevantes*



Precisión:



¿Cuántos datos
seleccionados son
relevantes?

Sensibilidad:



¿Cuántos datos
relevantes son
seleccionados?

Exactitud:



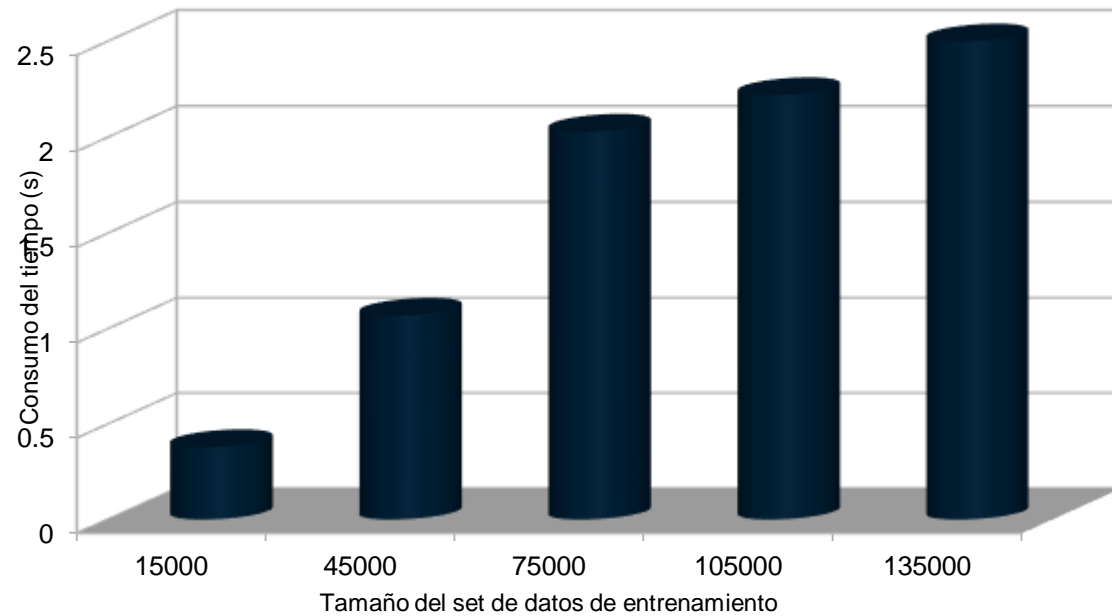
¿Cuántos datos se
filtraron
correctamente?

	Conjunto de entrenamiento	Conjunto de validación
Exactitud	0.74	0.74
Precisión	0.74	0.73
Sensibilidad	0.76	0.76

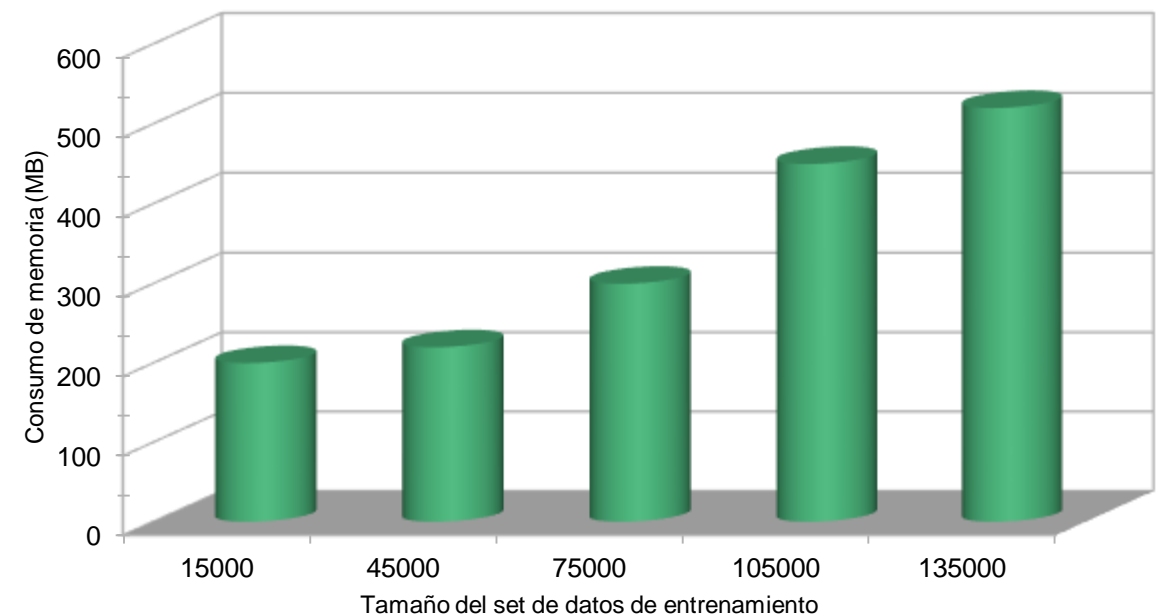
En esta tabla se logra apreciar que es un algoritmo que es un algoritmo eficiente y que en tanto en el entrenamiento como en la validación conserva los mismos porcentajes para las métricas de evaluación.



Consumo de tiempo y memoria



Consumo de tiempo



Consumo de memoria



Incluyan la citación del reporte en arXiv y
su vínculo comose muestra abajo

C. Patiño-Forero, M. Agudelo-Toro, and M. Toro. Planning
system for deliveries in Medellín. ArXiv e-prints, Nov. 2016.
Available at: <https://arxiv.org/abs/1611.04156>

Incluyan un
pantallazo

Cornell University

arXiv.org > cs > arXiv:1611.04156

Computer Science > Data Structures and Algorithms

[Submitted on 13 Nov 2016]

Planning system for deliveries in Medellín

Catalina Patiño-Forero, Mateo Agudelo-Toro, Mauricio Toro

Here we present the implementation of an application capable of planning the shortest delivery route in the city of Medellín, Colombia. We discuss the different approaches to this problem which is similar to the famous Traveling Salesman Problem (TSP), but differs in the fact that, in our problem, we can visit each place (or vertex) more than once. Solving this problem is important since it would help people, especially stores with delivering services, to save time and money spent in fuel, because they can plan any route in an efficient way.

Comments: 5 pages, 9 figures

Subjects: **Data Structures and Algorithms (cs.DS)**

ACM classes: F.2.0; G.2.2

Cite as: arXiv:1611.04156 [cs.DS]
(or arXiv:1611.04156v1 [cs.DS] for this version)



¡GRACIAS POR ESCUCHARNOS!