DISEÑO DE UN ALGORITMO PARA PREDECIR EL ÉXITO EN LAS PRUEBAS SABER PRO

Sebastian Castaño Orozco	Dennis Castrillón Sepúlveda
Universidad Eafit	Universidad Eafit
Colombia	Colombia
scasta31@eafit.edu.co	dcastri9@eafit.edu.co

Miguel Correa	Mauricio Toro
Universidad Eafit	Universidad Eafit
Colombia	Colombia
mcorrea@eafit.edu.co	mtorobe@eafit.edu.co

RESUMEN

El siguiente trabajo tiene como fin la construcción de un algoritmo basado en árboles de decisiones, que permita predecir el éxito de un estudiante de pregrado en las pruebas Saber Pro, teniendo como definición de éxito, la obtención de una nota superior al promedio de su cohorte.

El trabajo se enmarca en la necesidad de asignar una importancia cuantitativa respecto a los aspectos para tener en cuenta en la preparación en torno a la prueba y la necesidad de brindar una herramienta de análisis en torno al proceso global de preparación, presentación y obtención de resultados de la prueba Saber Pro.

A continuación, se detallan algunos trabajos relacionados y alternativas de solución basadas en árboles de decisión.

1. INTRODUCCIÓN

Un reto continuo en la metodología de enseñanza y aprendizaje en los pregrados de las distintas universidades es ofrecer modelos de educación que permitan la obtención de resultados positivos por parte de los estudiantes en las pruebas realizadas por el estado en los últimos semestres de la carrera.

Por tanto, predecir qué factores pueden influir en mayor proporción en la probabilidad de tener éxito en los resultados de las pruebas Saber Pro (en el caso de Colombia) puede ser un mecanismo óptimo de toma de decisiones relacionadas con el sistema educativo.

1.1. Problema

Se requiere predecir el éxito de un estudiante en las pruebas Saber Pro de acuerdo con sus características sociodemográficas y académicas, a saber, edad, ingresos de sus padres, pregrado, resultados en la prueba Saber 11, género, estrato, entre otras, a través de un algoritmo de árboles de decisión.

La resolución de este problema significaría un avance que impactaría directamente en la toma de decisiones relativas al proceso social y educativo por el que atraviesa un estudiante previo a enfrentarse a las pruebas Saber Pro en aras de mejorar los resultados y la probabilidad de éxito en el futuro.

2. TRABAJOS RELACIONADOS

2.1 Predicción del desempeño de un estudiante a través de Machine Learning

Este trabajo, realizado como parte de la tesis doctoral del estudiante Murat Pojon de la universidad de Tampere en Finlandia consistió en predecir los resultados en un examen para una base de datos de 480 estudiantes pertenecientes a la universidad de Jordania, teniendo como referencia 17 características de cada estudiante, entre las que se incluía género, nacionalidad y veces que el estudiante alzó la mano durante clase, entre otras. [1]

La predicción del desempeño por árboles de decisión se realizó con el algoritmo de particionamiento recursivo (CART) y obtuvo una precisión del 91.7%. Es pertinente resaltar que Pojon realiza una comparación entre varios algoritmos, para los cuales el algoritmo CART obtiene el segundo valor mas cercano a los resultados reales, por debajo del algoritmo de clasificación de Naïve Bayes.

2.2 Predicción de resultados mediante minería de datos y métodos de clasificación

Dorina Kababchieva de la universidad de Sofía en Bulgaria, procesa los datos de 10330 estudiantes descritos por 20 parámetros entre los que se encuentra género, puntaje en el examen de admisión y semestre, entre otros, con el fin de predecir los resultados en un examen de acuerdo con estos parámetros. La predicción se realiza a través del algoritmo C4.5 y se obtiene una precisión del 63.1%. [2]

Cabe destacar que la precisión alcanzada presenta un valor menor, en parte debido a la gran cantidad de estudiantes analizados.

2.3 Arboles de decisión para predecir la aprobación académica de estudiantes

Josip Mesarić y Dario Šebalj crearon un modelo que permitiera la clasificación de estudiantes en una de dos categorías, dependiendo del desempeño que tuvieran al final del curso académico, además de identificar qué factores influían su desempeño. Este modelo se basa en la información extraída de los estudiantes después de que completaran el primer curso académico y la recolección de datos a partir de estudiantes que cursaban su segundo año académico.

Se compararon diferentes algoritmos para la construcción de los árboles de decisión, y a partir de un modelo estadístico se determinó el algoritmo que tenía una mayor precisión, que en este caso fue usando REPTree que tuvo una tasa de clasificación del 79%, pero este árbol no tuvo tanto éxito clasificando ambas clases, por lo que se hizo un promedio de dos modelos, de esta forma se aumentó la tasa de clasificación apoyándose del modelo de algoritmo J48. [3]

2.4 Predecir el rendimiento de los estudiantes usando algoritmo de árboles de clasificación y regresión

.El rendimiento académico de los estudiantes siempre va a ser una preocupación de las partes que están interesadas en la educación y en el desarrollo de las personas, por lo que se busca innovar y establecer métodos que permitan a la persona mejorar más en dichos ámbitos. Teniendo como base toda la información recolectada a partir de los datos que se almacenan en los sistemas de registro, de esta forma se identifican patrones de comportamiento

En este estudio se utilizó un algoritmo de clasificación de minería de datos CART, para analizar los datos de la actividad de los estudiantes y predecir su desempeño académico.

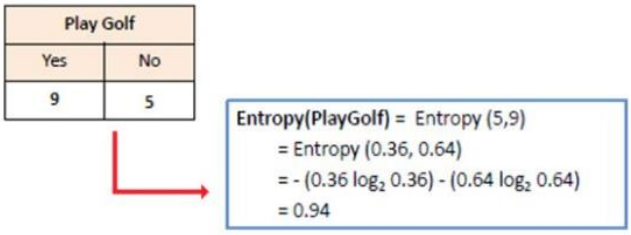
El modelo CART clasificó correctamente 167 estudiantes que reprobaron el curso, pero clasificaron erróneamente a otros 3 que no aprobaron la clase (clasificó correctamente 98,2% de los casos). El modelo también clasificó correctamente 182 estudiantes que no fueron reprobados (clasificó correctamente 100.0% de los casos). La precisión general de la clasificación es, por tanto, el promedio ponderado de estos dos valores (99,1%).[4]

3. ALTERNATIVAS DE ALGORITMOS DE ÁRBOL DE DECISIÓN

3.1 Algoritmo ID3

El algoritmo ID3, inventado por Ross Quinlan, se basa en la búsqueda de hipótesis o reglas dado un conjunto de ejemplos, el cual deberá estar conformado por una serie de tuplas de valores, cada uno de ellos denominados atributos, en el que uno de ellos (de tipo binario) será el atributo por clasificar. Los elementos de este algoritmo son los nodos que contienen los atributos, arcos que contienen valores posibles del nodo padre y hojas que clasifican el ejemplo como positivo o negativo.[5]

ID3 comienza con el conjunto original como el nodo raíz. En cada iteración del algoritmo, se itera a través de todos sin usar el atributo del conjunto y calcula la entropía de ese atributo. A continuación, selecciona el atributo que tiene la entropía más pequeña para posteriormente dividir el conjunto entre este atributo y dar lugar a los subconjuntos de datos.[6]

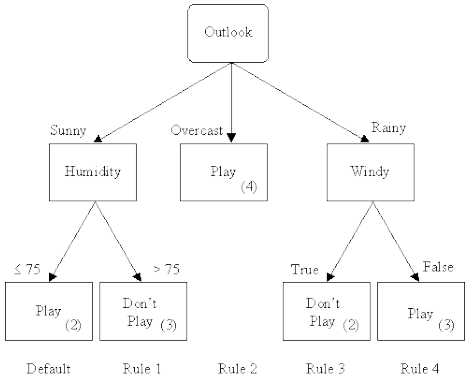


Fígura 1. Ejemplo de algoritmo ID3

3.2 Algoritmo C4.5

C4.5 es un algoritmo usado para generar un árbol de decisión. Fue desarrollado por Ross Quinlan en 1993 y es una extensión del algoritmo ID3, tiene como ventajas el manejo de datos perdidos y la posibilidad de trabajar con datos continuos.

El algoritmo considera todas las pruebas posibles que pueden dividir el conjunto de datos y selecciona la prueba con la mayor ganancia de información. Para cada atributo discreto, se considera una prueba con n resultados, siendo n el número de valores posibles que puede tomar el atributo. Para cada atributo continuo, se realiza una prueba binaria. En cada nodo, el sistema debe decidir qué prueba escoge para dividir los datos. [7]



Fígura 2. Ejemplo de algoritmo C4.5

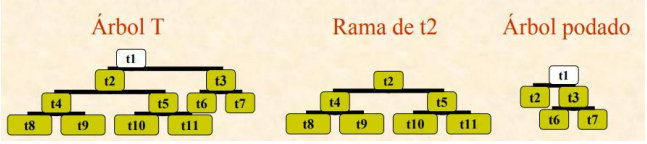
3.3 Algoritmo CART

El algoritmo CART fue diseñado por Breiman, con este algoritmo, se generan árboles de decisión binarios, lo que quiere decir que cada nodo se divide en exactamente dos ramas.

Este modelo admite variables de entrada y de salida nominal, ordinal y continua, por lo que se pueden resolver tanto problemas de clasificación como de regresión. [9]

Se basa en la idea de impureza. CART selecciona el corte que conduce al mayor decrecimiento de la impureza. Así se consiguen descendientes homogéneos en la variable respuesta Y.

CART propone segmentar la base de datos hasta obtener una estructura de árbol lo más compleja posible. Un nodo se declara como terminal sólo si su tamaño es inferior a un umbral preestablecido (normalmente muy pequeño). La complejidad de un árbol se mide por el número de nodos terminales. A continuación, se poda la estructura de árbol maximal que se ha obtenido. Una rama del nodo t de un árbol T está formada por él y todos sus descendientes. Podar la rama en t consiste en eliminar todos los descendientes del nodo t.



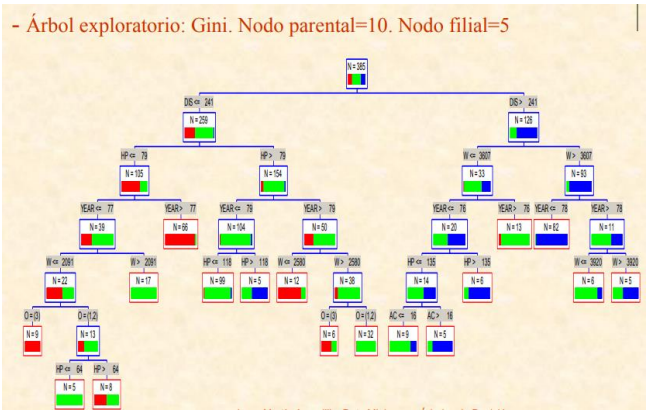
Fígura 3. Representación proceso CART

El proceso de poda se apoya en la siguiente medida: Combina el riesgo o coste de predicción y la complejidad. El primer sumando mide el riesgo de T (tasa de error si el problema es de clasificación o la suma de las varianzas residuales si es de regresión). El segundo sumando penaliza las estructuras de árbol complejas. El parámetro $\alpha \geq 0$ se denomina parámetro de complejidad

$$R_\alpha(T) = R(T) + \alpha|\tilde{T}|$$

Fígura 4. Medida para la poda maximal

Se realiza de una manera inteligente, eliminando las ramas más débiles. La idea es encontrar subárboles que minimicen Rα (T). [8]



Fígura 5. Ejemplo consumo vehículos con CART

3.4 Algoritmo CHAID

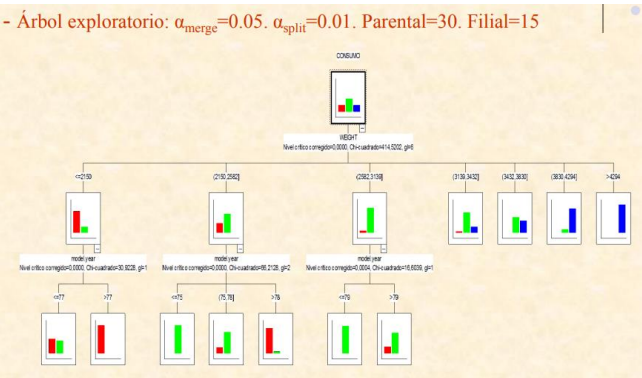
Procede del ámbito de la Inteligencia artificial. Desarrollado por Kass a principios de los años 80. CHAID considera todos los cortes posibles en todas las variables. Selecciona el corte que da el menor p-valor asociado a una medida de contraste estadístico. Si la variable criterio es categórica la medida es la χ 2 de Pearson. Si es continua la medida es la del test de la F. La búsqueda de la variable y el corte óptimo se lleva a cabo en dos fases: merge (fusión de categorías) y split (selección de la variable de corte).

Fase merge; Agrupa estados o valores de las variables explicativas. Para cada variable, agrupará los estados de cuya unión se obtenga el de menor significación estadística del contraste; siempre que ésta supere un umbral αmerge, fijado de antemano

Fase Split: De la fase merge se toma la agrupación en la variable con contraste más significativo (menor p-valor ajustado). Si la significación estadística es inferior a un mínimo αsplit prefijado, se toma dicha agrupación como partición del nodo.

El criterio de parada CHAID: Se fija de antemano por el experimentador. Depende de: El nivel Split, número de

niveles de la estructura de árbol, un umbral mínimo para el tamaño de los nodos descendientes. [8]



Fígura 6. Ejemplo consumo vehículos con CHAID

REFERENCIAS

1. Pojon, M. Using Machine Learning to Predict Student Performance. Retrieved August 14, 2020, from University of Tampere: https://trepo.tuni.fi/bitstream/handle/10024/101646/GRADU-1498472565.pdf?sequence=1.
2. Kabakchieva, D. Predicting Student Performance by Using Data Mining Methods for Classification, *Sciendo*. Retrieved August 14, 2020, from Versita, Bulgarian Academy of Sciences: https://content.sciendo.com/view/journals/cait/13/1/article-p61.xml
3. Mesaric, J. Sebalj, D. Decision trees for predicting the academic success of students. Faculty of Economics in Osijek, University of Josip Juraj Strossmayer in Osijek. Retrieved CRORR 7(2016): http://bib.irb.hr/datoteka/853222.clanak.pdf.
4. M Krishna, Bandlamudi S B P Rani, G Kalyan Chakravarthi, B Madhavrao, S M B Chowdary. Predicting Student Performance using Classification and Regression Trees Algorithm. Retrieved January 2020: http://www.ijitee.org/wp-content/uploads/papers/v9i3/C8964019320.pdf.
5. Anónimo. ¿Qué es el algoritmo ID3. Retrieved August 14, 2020, from Quora: https://es.quora.com/Qu%C3%A9-es-el-algoritmo-ID3.
6. Wikipedia. Algoritmo ID3 - ID3 algorithm. Retrieved August 14, 2020, from Qwe: https://es.qwe.wiki/wiki/ID3_algorithm.
7. Blogspot. Minería de Datos - C4.5. Retrieved August 14, 2020, from Blogspot: https://mineriac45.blogspot.com/
8. Martin J. M. Data Mining con Árboles de Decisión. Dpto. Estadística, Investigación Operativa y Cálculo Numérico Universidad Nacional Educación a Distancia. https://web.fdi.ucm.es/posgrado/conferencias/JorgeMartin-slides.pdf.
9. Parra F. Estadística y Machine Learning con R. Retrieved January 25, 2019. https://bookdown.org/content/2274/agrupacion-de-la-informacion.html.
10. Wikipedia. Chi-square automatic interaction detection. Retrieved January 5, 2019. https://en.wikipedia.org/wiki/Chisquare_automatic_interaction_detection.