DESARROLLO DE MODELOS PARA PREDECIR EL RENDIMIENTO ACADÉMICO MEDIANTE INTELIGENCIA ARTIFICIAL

DEVELOPMENT OF MODELS FOR PREDICTING ACADEMIC ACHIEVEMENTS USING ARTIFICIAL INTELLIGENCE

TRABAJO FIN DE GRADO
CURSO 2021-2022

AUTOR
GORKA SILVA RAMÓN

DIRECTOR
JOSÉ IGNACIO HIDALGO PÉREZ

GRADO EN INGENIERÍA INFORMÁTICA
FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID

DESARROLLO DE MODELOS PARA PREDECIR EL RENDIMIENTO ACADÉMICO MEDIANTE INTELIGENCIA ARTIFICIAL

DEVELOPMENT OF MODELS FOR PREDICTING ACADEMIC ACHIEVEMENTS USING ARTIFICIAL INTELLIGENCE

TRABAJO DE FIN DE GRADO EN INGENIERÍA INFORMÁTICA
DEPARTAMENTO DE ARQUITECTURA DE COMPUTADORES

AUTOR
GORKA SILVA RAMÓN

DIRECTOR
JOSÉ IGNACIO HIDALGO PÉREZ

CONVOCATORIA: JUNIO 2022

GRADO EN INGENIERÍA INFORMÁTICA
FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID

JUNIO DE 2022

DEDICATORIA

A Iñaki por haberme apoyado durante
mis años de estudio. Sin él habría sido
todo mucho más complicado.

AGRADECIMIENTOS

A mis padres por haber tenido que lidiar con muchas de mis indecisiones académicas y haberme apoyado en todo momento.

A los profesores y compañeros de la Facultad de Informática que me han enseñado lo asombrosa que puede ser la informática.

A José Ignacio Hidalgo Pérez por haber dirigido este TFG. Ha estado siempre disponible para que pudiese realizar el trabajo lo mejor posible. Es un gran profesor y una excelente persona.

A Alberto y a Daniel por haberme ayudado a comprender las librerías de Python necesarias para este trabajo.

RESUMEN

El desarrollo de la inteligencia artificial está consiguiendo mejoras en todos los campos de aplicación. Una de las funciones más utilizadas en inteligencia artificial es el análisis de datos y la generación de modelos predictivos. Gracias al análisis de datos con inteligencia artificial se obtienen resultados reveladores que ayudan a mejorar los servicios en muchas instituciones. Como no podía ser menos, el campo de la educación también se puede beneficiar de esta tecnología.

Es gracias a la educación que se consigue desarrollar a las personas y, por lo tanto, cualquier mejora en este ámbito es provechoso para la sociedad. En este trabajo de fin de grado se ha diseñado una herramienta web capaz de realizar predicciones de rendimiento académico sobre alumnos universitarios. Estas predicciones nos pueden brindar información muy valiosa acerca de los factores que afectan al rendimiento de los estudiantes. No solo analiza las calificaciones del alumno sino, también, los datos socioeconómicos, lo que permite obtener conclusiones que de otra manera no habrían sido posibles con un análisis convencional. A parte de dar otra perspectiva, esta herramienta tiene otra ventaja y es que puede analizar gran cantidad de datos de una forma muy rápida, y, ya que se puede aplicar en cualquier titulación universitaria, se espera que este software pueda aportar su granito de arena a la mejora en la educación.

Palabras clave

Educación, inteligencia artificial, rendimiento académico, predicción, software.

ABSTRACT

The development of artificial intelligence is a leading to improvements in all fields of application. One of the most widely used functions in artificial itelligence is data analysis and generating predictive models. Thanks to data analysis with artificial intelligence, revealing results are obtained that help improve services in many institutions. Of course, the field of education can also benefit from this technology.

It is thanks to education that people are able to develop and, therefore, any improvement in this field is beneficial to society. In this final degree project, a web tool has been designed that is capable of making academic performance predictions for university students. These predictions can provide us with valuable information about the factors that affect student performance. It analyses not only the student's grades but also socio-economic data, which allows conclusions to be drawn that would not otherwise have been possible with conventional analysis. Apart from giving another perspective, this tool has another advantage in that it can analyse large amounts of data very quickly, and since it can be applied to any university degree, it is hoped that this software can contribute to the improvement of education.

Keywords

Education, artificial intelligence, academic performance, prediction, software.

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

Capítulo 1 - Introducción

1.1 El problema de los costes en la formación del alumnado

El rendimiento académico depende de muchos factores que habitualmente no se pueden medir de forma sencilla. Sin embargo, existen ciertos datos que nos pueden proporcionar información muy valiosa acerca de la situación socioeconómica del estudiante. Por ejemplo, cuando un alumno realiza la matrícula en la universidad, proporciona información sobre su situación social, el nivel de estudios de sus padres, ingresos, calificaciones de etapas anteriores, etc. Analizando este tipo de datos se podrían obtener conclusiones que ayuden a mejorar el sistema educativo. Si gracias a este análisis se tomasen decisiones que disminuyesen el número de estudiantes suspensos, se podrían obtener numerosos beneficios, como el ahorro de costes y la optimización de recursos, entre otros.

Actualmente, en una universidad pública, el coste aproximado de formación de un alumno es de 7500€ por curso. El coste de una primera matrícula en Ingeniería Informática es de 1473€ por curso. Si un alumno suspendiese 60 créditos el coste de la segunda matrícula sería 2715€. El precio a pagar de la segunda matrícula no alcanza ni la mitad del coste de la formación.

Durante el curso 2019-2020, en España, se matricularon una media de 54,6 créditos por alumno en las universidades públicas presenciales. Se superó una media de 46,6 créditos, luego se suspendieron una media de 8 créditos por alumno. Si suponemos que se suspende esta media de créditos en la Facultad de Informática estos supondrían un coste de formación aproximado de 1000€, de los cuales el alumno pagaría 362€ si fuesen créditos de segunda matrícula. Por lo tanto, por cada alumno de la facultad, se estaría desaprovechando una media de 638€. En el curso 2019-2020 estuvieron matriculados 2127 alumnos, esto implicaría una media de 1.357.026€ desaprovechados.

1.2 Algoritmos de predicción

El Machine Learning es la ciencia de programar computadoras que puedan aprender a partir de ciertos datos. Un ejemplo de un algoritmo sencillo de Machine

Learning es un programa que clasifique los emails como *spam* o *no-spam*. Este software aprende a discernir entre estas dos clases únicamente a partir de datos proporcionados anteriormente, como por ejemplo, los e-mails que otros usuarios hayan reportado como spam.

Uno de los campos más utilizados de la inteligencia artificial es el análisis de datos. Los algoritmos de Machine Learning producen unos resultados que no sería posible obtener de otra forma, y que pueden brindar una información muy valiosa en cualquier campo. A continuación se muestran algunos ejemplos de aplicaciones del análisis de datos con inteligencia artificial.

COMPAS: Es una herramienta utilizada por los tribunales de EE. UU. que ayuda a la toma de decisiones. Evalúa con inteligencia artificial la probabilidad de que el acusado vuelva a reincidir en un futuro.

ZAML: Es un software diseñado para evaluar a un cliente que solicita un préstamo a una entidad financiera. Utiliza todo tipo de datos para conseguir mejor información sobre el cliente y realiza una predicción de la rentabilidad de conceder el préstamo a ese cliente.

Dentro de la inteligencia artificial existen los algoritmos de Machine Learning que son utilizados frecuentemente para el análisis de datos. Los algoritmos de clasificación, que pertenecen al grupo de algoritmos de predicción, tienen la finalidad de asignar una clase al objeto de la predicción. Este tipo de algoritmo es el que se usará en este trabajo de fin de grado. Algunos ejemplos sobre algoritmos de clasificación son:

Predicción de tráfico en *Google Maps*: clasifica los tramos de carretera en función de lo congestionados que estén. Tiene 3 tipos de congestión: ninguna (azul), leve (amarillo), severa (rojo).

Software de *Vesta Corporation* para la predicción de fraudes en las transacciones de los clientes: utiliza datos sobre cómo los clientes realizan las transacciones para poder detectar con mayor facilidad posibles fraudes. Este algoritmo de clasificación sólo clasifica una transacción en dos tipos de clases: fraudulenta o no fraudulenta.

1.3 Objetivos

En este TFG hemos desarrollado la herramienta *Academic Performance Predictor*. para hacer predicciones de rendimiento académico de tal forma que, con los resultados obtenidos, se ayude al desempeño de los estudiantes y a la toma de decisiones de gestión académica. Estas predicciones se realizan con varios algoritmos de inteligencia artificial, lo que permitirá analizar diversos tipos de datos y obtener resultados que no podrían ser obtenidos con un análisis convencional.

En este proyecto se busca desarrollar una herramienta web que, a partir de unos datos socioeconómicos y calificaciones en otras asignaturas, pueda realizar una predicción sobre si un alumno puede, o no, superar una calificación (cota) en una asignatura.

Este servicio se podrá usar para todo tipo de titulación universitaria. Será necesario proporcionar datos suficientes para entrenar el modelo de inteligencia artificial para que pueda hacer predicciones reveladoras. Con cada ejecución se realizarán tres predicciones, una por cada uno de los diferentes algoritmos predictivos.

1.4 Organización de la memoria

A continuación, podemos encontrar la organización de la memoria:

En el Capítulo uno, se encuentra una introducción sobre qué es un algoritmo de predicción y los objetivos de usar estos algoritmos en el proyecto para ayudar a la mejora de la educación en las universidades públicas.

Capítulo dos, se explican cada uno de los tres algoritmos de clasificación usados en este proyecto.

Capítulo tres, se muestra cómo funcionan las herramientas de análisis de los modelos de predicción.

Capítulo cuatro, se explica cada una de las funcionalidades de la herramienta web .

Capítulo cinco, se comentan los resultados obtenidos después de haber realizado varias predicciones.

Capítulo seis, se expone una conclusión y las posibles mejoras futuras de este proyecto.

Capítulo 2 - Algoritmos de clasificación

Los algoritmos de Machine Learning pueden dividirse en dos tipos: aprendizaje supervisado y aprendizaje no supervisado. Este software utiliza tres algoritmos con aprendizaje supervisado y aprendizaje por refuerzo. En los algoritmos de aprendizaje supervisado los datos con los que se entrena el sistema incluyen la solución buscada por este. Normalmente se utiliza el aprendizaje supervisado para problemas de clasificación, como es este caso.

Un algoritmo de clasificación recibe los datos de un objeto a predecir y, a partir de esa información, le asigna una de las diferentes clases posibles. En el capítulo anterior se explicado el ejemplo de un algoritmo de clasificación que distingue entre e-mails *spam* y e-mails *no-spam*. Spam y no-spam son las clases del problema. Los datos de entrenamiento incluirán varios e-mails clasificados en alguna de las dos clases. Gracias a esta información se entrenará al *modelo*. El modelo se podría definir como un conjunto de reglas que permite realizar la predicción de la clase a la que pertenece a partir de unos datos iniciales. De esta forma, teniendo el modelo creado, el algoritmo podrá predecir la clase de un e-mail cualquiera. En este capítulo se explica el funcionamiento de los diferentes algoritmos de clasificación usados para las predicciones.

Sierra da una definición formal del funcionamiento general de los algoritmos de clasificación. "Un clasificador puede ser definido como: una partición del espacio de clasificación *X* en *M* subconjuntos disjuntos A_1, A_2, …, A_M, siendo *X* la unión de todos ellos y para todo x perteneciente a A_m la clase predicha es C_m." (1)

2.1 K-Nearest Neighbors Classifier

Uno de los tipos de algoritmos de clasificación supervisada más conocidos son los basados en criterios de vecindad. Estos métodos de clasificación necesitan definir un espacio donde sitúan a todos los patrones reconocidos y una métrica para medir las distancias entre patrones. Una de las ventajas de estas técnicas es su simplicidad.

El funcionamiento es el siguiente: se sitúa un patrón que se quiera clasificar en el espacio definido y se le asigna una clase en función de la clase de los patrones más

próximos a él. Este método se basa en que los elementos de una misma clase se encontrarán próximos en el espacio. Esto puede no ser aplicable para todos los problemas de clasificación, lo que hace que no resuelvan correctamente ciertos problemas.

Un problema de clasificación abordado con un enfoque basado en criterios de vecindad se puede caracterizar del siguiente modo:

1. Se dispone de un conjunto N de muestras ya clasificadas llamado 'conjunto de entrenamiento'
2. Se clasifica un caso X, no perteneciente al conjunto de entrenamiento.
3. El algoritmo define una métrica entre los diferentes objetos del espacio de representación. Para hacer la predicción se utilizará únicamente esta métrica. (1)

Esta definición de Sierra da una perspectiva más general del funcionamiento de un algoritmo basado en criterios de vecindad.

A diferencia de otros métodos de clasificación, los algoritmos de clasificación por vecindad no requieren un paso de inducción del modelo, ya que el modelo se halla implícito en los datos. (1)

Para realizar la predicción, este algoritmo sitúa a todos los patrones en un espacio con tantas dimensiones como variables a analizar. Para saber cuál es la clase a la que pertenece el patrón que queremos predecir, se seleccionan los *k* patrones más cercanos a éste en el espacio designado. La clase del patrón es aquella a la que pertenecen la mayoría de los *k* vecinos. En la Figura 1 se muestra un ejemplo de clasificación para las clases *A* y *B* en un espacio de dos dimensiones. Si k = 3 se le asignaría la clase B, mientras que si k = 6 se asignaría la clase A.

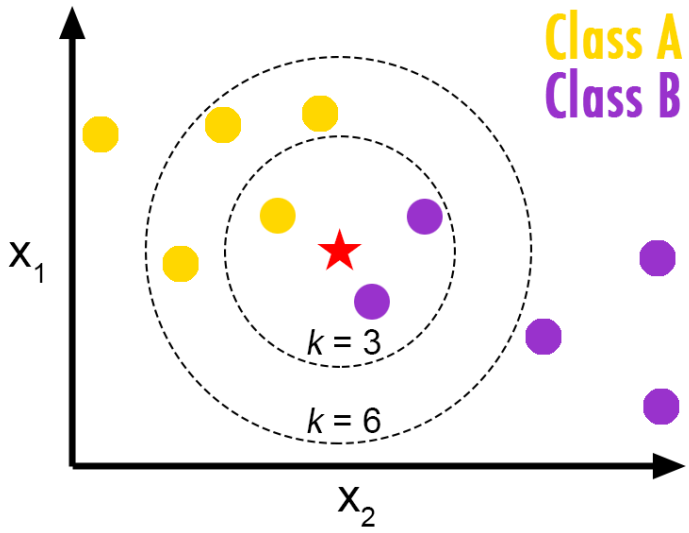

Figura 1. Representación del funcionamiento del algoritmo K-NN Classifier (2)

En la Figura 2 se muestra el comportamiento esperado (en % de bien clasificados) del algoritmo K-NN respecto al valor del parámetro k. Como se puede observar con los valores muy pequeños o muy grandes no se pueden obtener buenos resultados.

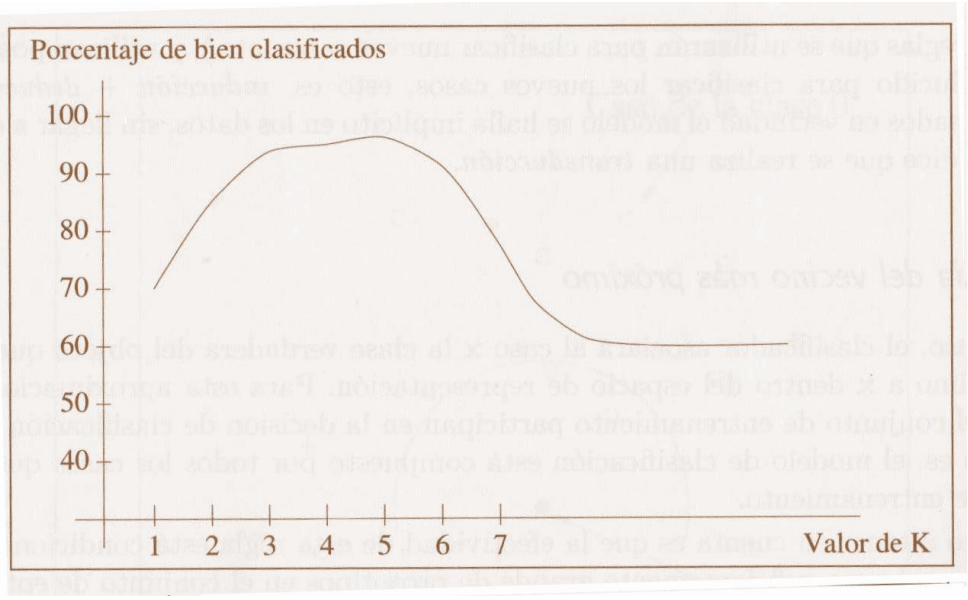

Figura 2. Comportamiento esperado del algoritmo K-NN respecto al valor del parámetro k. (1)

2.2 MultiLayer Perceptron Classifier

Una red neuronal intenta simular el aprendizaje del cerebro humano de una forma simplificada. Se basa en interconectar varios elementos de cómputo simples (neuronas) para obtener unas salidas determinadas por las diferentes entradas.

Estas neuronas tienen una plasticidad en su estructura que hace que se puedan adaptar para conseguir el comportamiento deseado. Poseen un peso asociado que es el que van modificando según las necesidades de la red. Por lo tanto, la salida de cada neurona dependerá de las entradas y de su peso. Estos pesos se van ajustando según los patrones de entrenamiento introducidos.

Para que la red neuronal se entrene es necesario que se conozcan las clases de los patrones de entrenamiento, lo que significa que es un tipo de aprendizaje supervisado. En este proyecto se usa una red neuronal para resolver un problema de clasificación, luego la salida de la red indicará la clase a la que pertenece el patrón a analizar.

MLP Classifier es una red neuronal de tipo *feedforward* con varias capas ocultas y una única neurona de salida. En esta neurona de salida se realiza la suma ponderada de los valores obtenidos en la última capa oculta. La salida de la red será 0 o 1 en función de la predicción de la clase a la que pertenezca el individuo, siendo 0 la clase *No alcanzará* y 1 la clase *Alcanzará.* La capa de entrada tendrá una neurona por cada variable considerada en la predicción. En la Figura 3 se muestra un diagrama de ejemplo de una red neuronal con tres entradas y una salida.

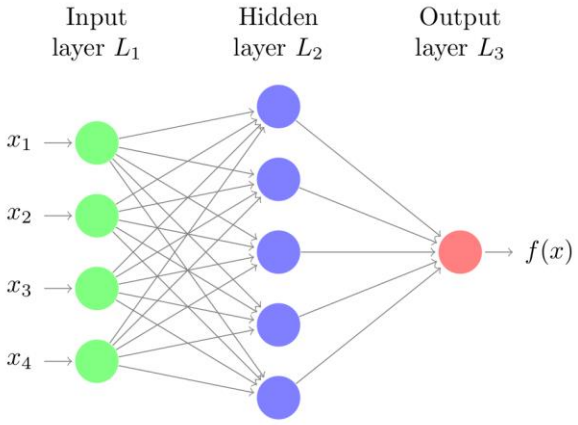

Figura 3. Representación simplificada de una red neuronal con una capa oculta. (3)

2.3 Gradient Boosting

Un árbol de decisión representa un conjunto de restricciones o condiciones que se organizan jerárquicamente y que se aplican de forma sucesiva desde la raíz hasta llegar a un nodo terminal o nodo hoja (4). Los árboles de clasificación empiezan por un nodo *raíz* al que pertenecen todos los patrones. El resto de nodos pueden ser nodos *intermedios* o nodos *hoja*. En los nodos hoja es donde encontramos las diferentes zonas a las que pertenecen los patrones.

Para llegar hasta un nodo hoja, los patrones siguen caminos diferentes según las reglas definidas en los nodos intermedios. En cada nodo intermedio se decide a qué nodo hijo se asignará el patrón según la regla definida. Esta asignación depende del valor de las variables de un patrón. La clase a predecir del patrón depende del nodo hoja al que haya sido asignado.

En este proyecto se usa el algoritmo Gradient Boosting para hacer una predicción de clasificación con árboles de decisión. El funcionamiento se basa en la unión de las predicciones de varios árboles de decisión que se van creando según las predicciones de árboles anteriores. Su funcionamiento de generación es el siguiente:

1. El primer árbol solo tiene una hoja. Esta hoja contiene el valor de la media entre los alumnos que superan la cota en esa asignatura, con valor 1, y los que no, con valor 0. La predicción de este árbol es que el valor de todos los patrones es la media.

Figura 4. Ejemplo de árbol inicial donde la media de todos los individuos es 0,68.

2. El segundo árbol se crea realizando las predicciones de cada uno de los patrones de entrenamiento, pero no se hace intentando predecir las clases del problema, que son 0 o 1, sino intentando predecir los errores del primer árbol, llamados *errores residuales*. Esto quiere decir que si un alumno ha superado la cota, el error residual del primer árbol es 1 – 0.32 = 0.68. En otro caso el error será 0.32 en este

ejemplo. Por lo tanto, los nodos hojas solo podrán tener esos dos valores. Utilizando las variables se crean diferentes reglas para los nodos intermedios del árbol de decisión.

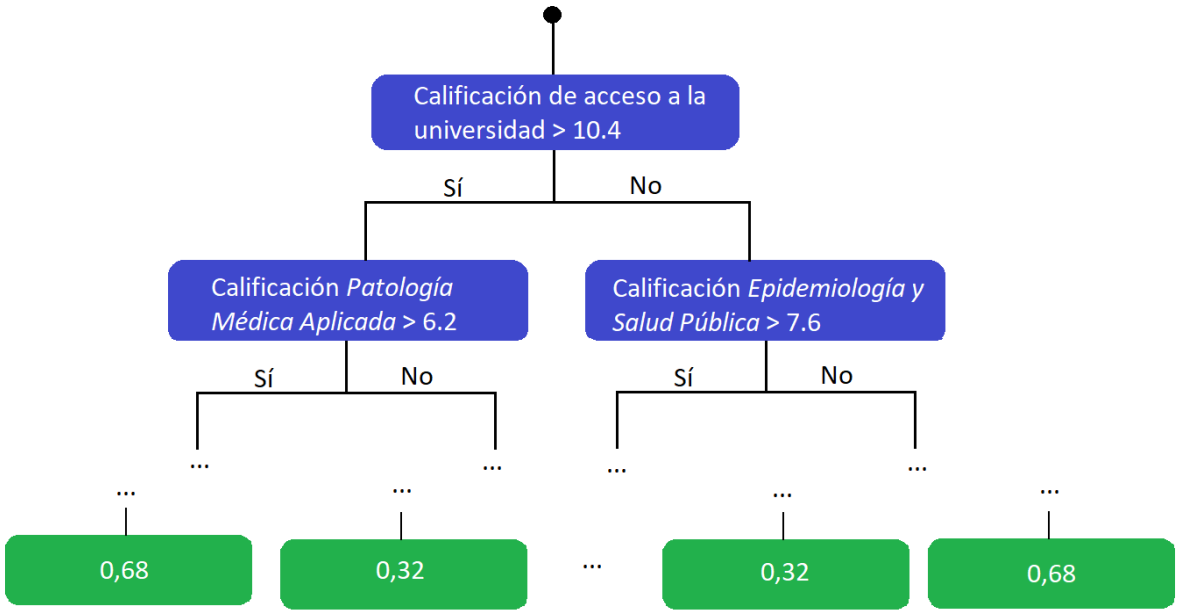

Figura 5. Ejemplo segundo árbol generado.

3. El error de predicción para el tercer árbol se calcula sumando la predicción del primer árbol junto con la predicción del segundo árbol multiplicada por el Learning Rate. Por ejemplo, si Learning Rate = 0.1 para un alumno que ha superado la cota, la predicción sería:

$$0.68 + (0.1*0.68) = 0.748$$

Una predicción más acertada que el primer árbol. Los árboles siguientes se crean como este segundo árbol, intentando predecir los errores de su respectivo árbol anterior.

Capítulo 3 - Herramientas de análisis de modelos

Para interpretar los resultados de la predicción se utilizan funciones de la librería *Scikit-learn* (5) y la librería *SHAP* (6). Scikit-learn es una librería de Python (7) que proporciona varios algoritmos de predicción supervisada y no supervisada. Para el análisis de los resultados de estos algoritmos también proporciona numeras funciones que pueden generar gráficos. SHAP (SHapley Additive exPlanations) es una herramienta de visualización que se usa para dar transparencia a las predicciones de cualquier modelo mostrando la contribución de cada variable en la predicción. Las imágenes y la tabla mostradas para interpretar los resultados se obtienen del modelo Gradient Boosting.

3.1 Análisis de las características del modelo

3.1.1 Métricas de Gradient Boosting

	precision	recall	f1-score	support
No alcanzará	0.43	0.38	0.40	8
Alcanzará	0.80	0.83	0.82	24
accuracy			0.72	32
macro avg	0.61	0.60	0.61	32
weighted avg	0.71	0.72	0.71	32

Figura 6. Métricas de Gradient Boosting.

Precision: es el ratio de las predicciones correctas respecto al total de predicciones de esa clase.

En el caso de la clase Alcanzará (positivos): Precision = TP / TP + FP

Recall: es el ratio de las predicciones correctas respecto al total de individuos de esa clase.

En el caso de la clase Alcanzará (positivos): Recall = TP / TP + FN

F1-score: es la media armónica de *precision* y *recall*. Esto nos da una información más equilibrada sobre el rendimiento del modelo.

F1 Score = 2*(Recall * Precision) / (Recall + Precision)

Support: número de patrones pertenecientes a esa clase.

Accuracy: Es el ratio de las predicciones correctas de todas las predicciones realizadas.

Accuracy = TP + TN / TP + FP + FN + TN

Macro avg: Es la media de los valores precision, recall y f1-score, de las clases.

Weighted avg: Calcula la media de los valores precision, recall y f1-score de las clases, teniendo en cuenta el desbalance entre el número de patrones de cada clase.

3.1.2 Matriz de confusión

TrueNegative	FalsePositive	FalseNegative	TruePositive
3	5	4	20

Figura 7. Matriz de confusión.

True Negative: Número de patrones pertenecientes a la clase No alcanzará que han sido predichos correctamente.

False Positive: Número de patrones pertenecientes a la clase No alcanzará que no han sido predichos correctamente.

False Negative: Número de patrones pertenecientes a la clase Alcanzará que no han sido predichos correctamente.

True Positive: Número de patrones pertenecientes a la clase Alcanzará que han sido predichos correctamente.

3.1.3 Gráfico Beeswarm

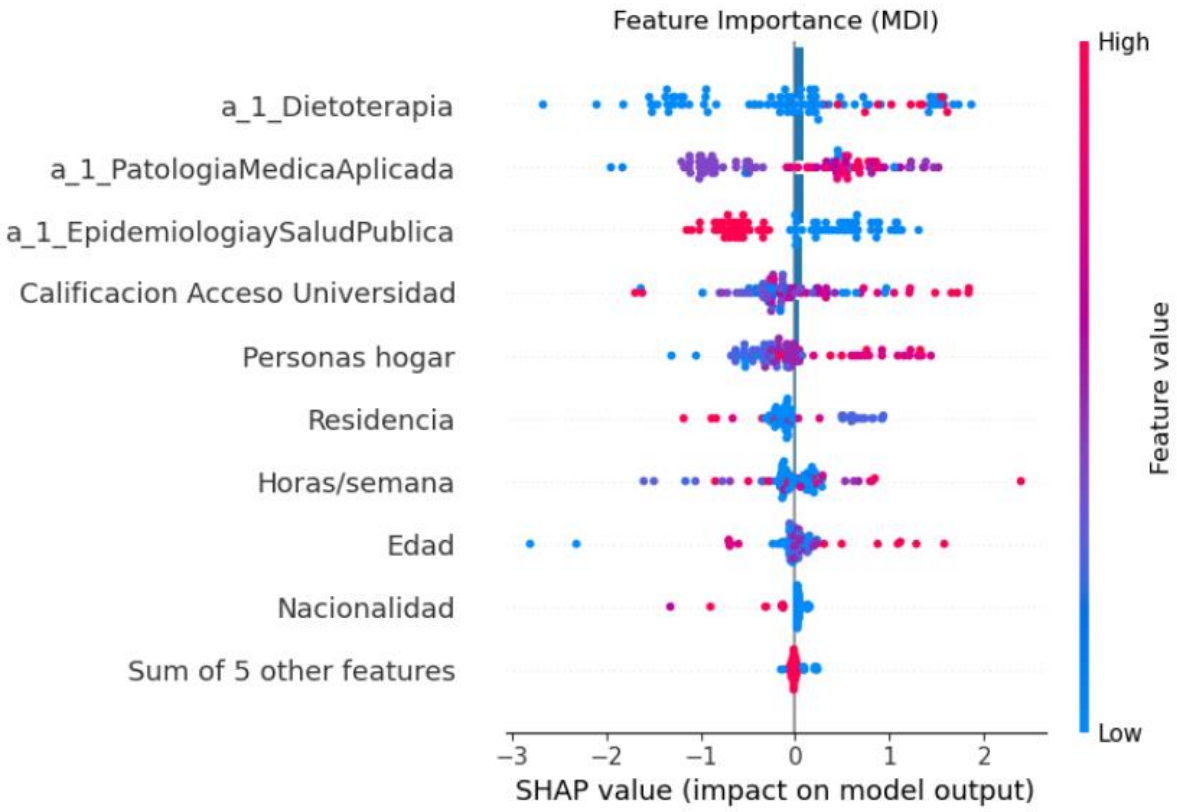

Figura 8. Ejemplo gráfico Beeswarm. (5)

La disposición de los puntos en el eje x indica cómo influyen los datos individuales en la predicción. Aquellos con valores positivos favorecen la predicción de que superará la cota. Por ejemplo, en la Figura 8 vemos una clara separación entre los puntos azules y los puntos rojos en la calificación de la asignatura Epidemiología y Salud Pública. Los puntos rojos, al tener una puntuación negativa, indican que no son favorables para la predicción de que superará la cota. Cuanto más rojo sea un punto, mayor valor (calificación, número de horas...) tiene respecto a los demás individuos. En este ejemplo las calificaciones altas en Epidemiología y Salud Pública influyen negativante para superar la cota. En cambio, las calificaciones bajas en esta asignatura, influyen positivamente.

3.1.4 Gráfico Feature Importance

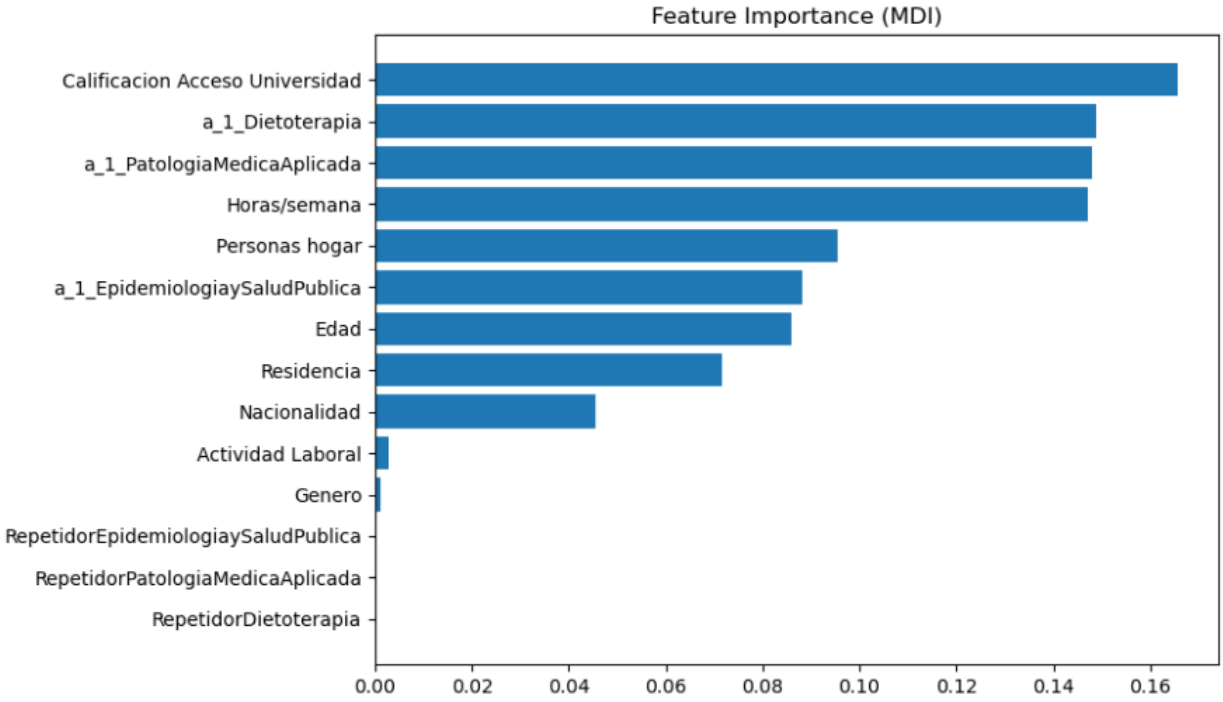

Figura 9. Ejemplo gráfico Feature Importance. (5)

Este gráfico muestra la contribución de cada una de las variables en la decisión de la predicción. El valor se expresa en ratio de influencia. En el gráfico de la Figura 9 se observa que la variable con mayor importancia en la decisión de las predicciones ha sido la Calificación de Acceso a la Universidad. Este valor es una media de todas las predicciones realizadas sobre los datos de test. Habrá casos individuales en los que otras variables tengan mayor importancia.

3.2 Análisis de las características de una predicción individual

3.2.1 Gráfico en cascada de SHAP

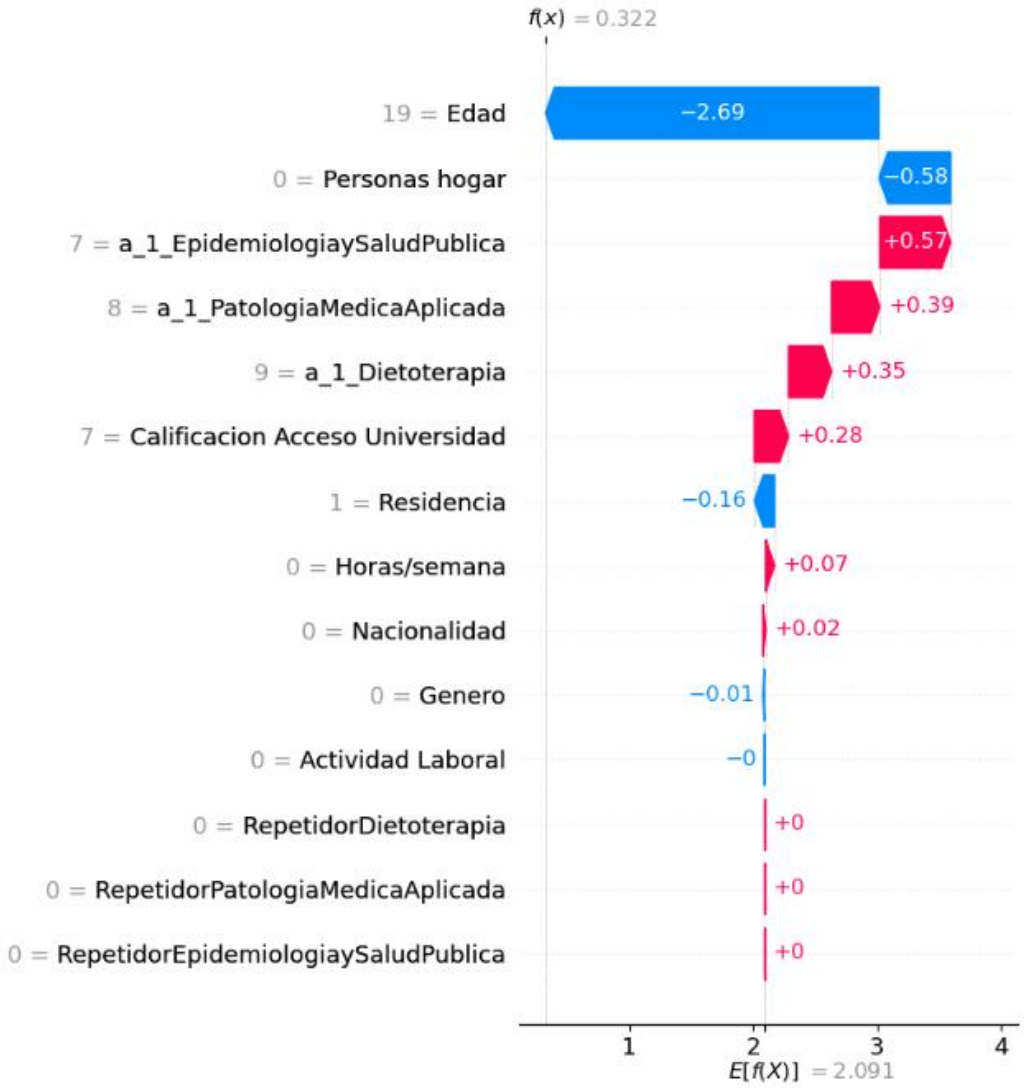

Figura 10. Gráfico en cascada de SHAP. (6)

En la parte inferior del gráfico se indica en qué valor esperado empieza la predicción antes de considerar los datos introducidos del alumno a predecir. Este valor esperado se obtiene por los datos usados durante el entrenamiento del modelo. En este caso el valor es E[f(X)] = 2.091. A partir de este valor se indica cómo contribuye cada variable en la predicción, siendo las contribuciones positivas las que favorecen la predicción de superar la cota. Si el valor final de f(X) es positivo el algoritmo predice que el alumno alcanzará la cota.

Este gráfico se diferencia del anterior porque muestra la importancia de las variables en una predicción individual, mientras que el gráfico Feature Importance mostraba la media de todas las predicciones individuales. Como se puede observar en la Figura 10, la variable más decisiva ha sido la edad. Lo que significa que una edad de 19 años influye negativamente en superar la cota, de una forma tan importante que casi hace que una cota que en general se supera en la mayoría de casos, en este caso se haya quedado con un valor de f(X) cercano a 0.

Capítulo 4 - Descripción del sistema

4.1 Funcionamiento de la herramienta web

Este software hace una predicción de si un alumno superará, o no, una calificación determinada en una asignatura en concreto. Para conseguir esto se tendrá que entrenar un modelo con datos de otros alumnos de esa titulación. Se seleccionarán las variables que se quieren tener en cuenta en la predicción y, en base a esas variables, se entrenará el modelo. Una vez entrenado el modelo se introducen los datos del alumno a predecir y los diferentes algoritmos indicarán si el alumno superará, o no, la calificación establecida. La Figura 11 contiene el diagrama de flujo de la herramienta web. El diagrama indica todos los pasos que se han de seguir para hacer las predicciones. Cada paso es una pantalla diferente en la que se pide una acción al usuario.

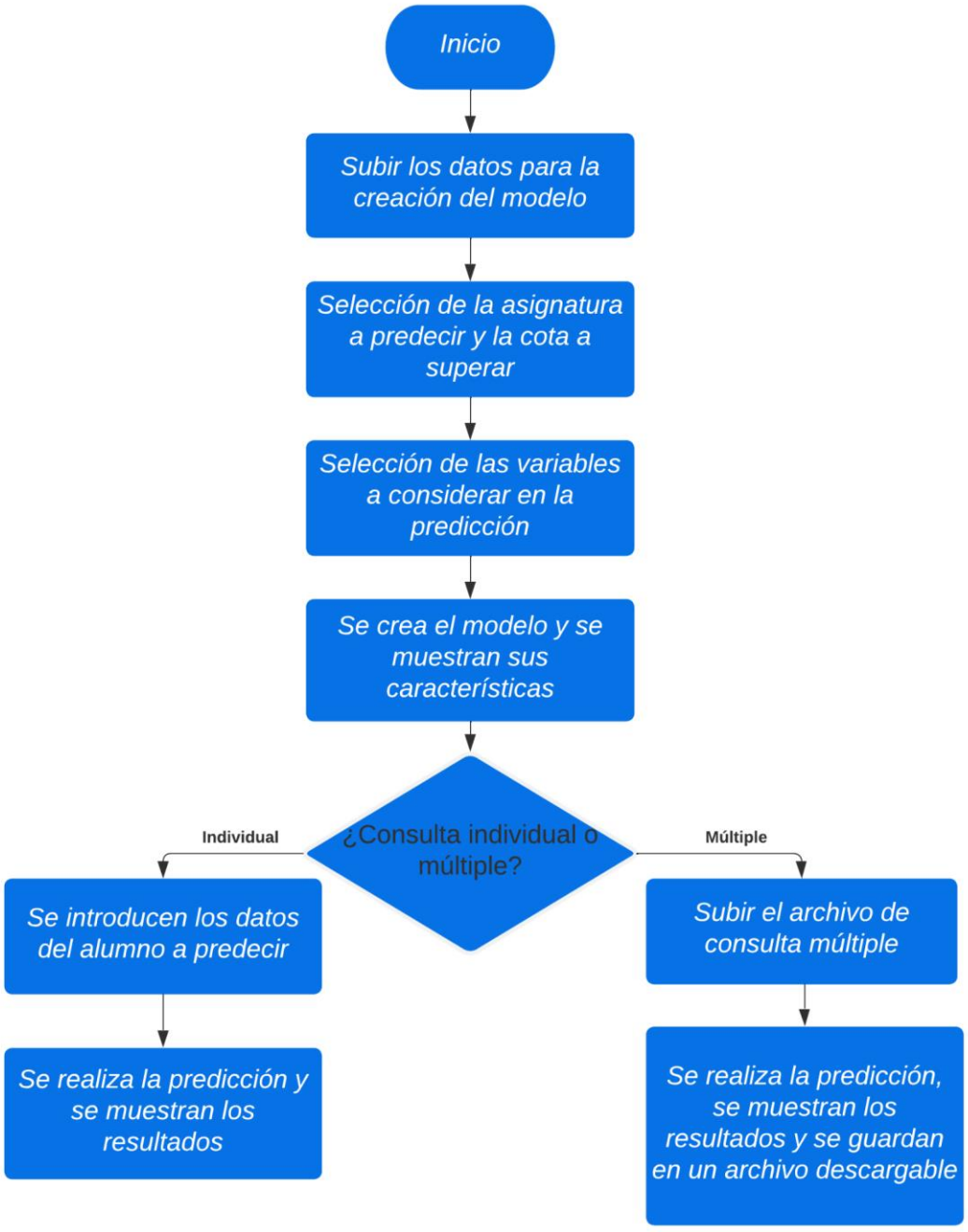

Figura 11. Diagrama de flujo.

4.2 Inicio

Se informa al usuario con un texto explicativo sobre la herramienta Academic Performance Predictor.

Figura 12. Inicio

4.3 Subir archivo de creación del modelo

Para realizar la predicción es necesario la creación de un modelo entrenado. Para ello, el usuario debe subir un archivo con datos de varios alumnos. Este archivo tiene que tener un formato en específico. Para informarse sobre este formato el usuario puede descargar un archivo PDF con las instrucciones sobre el formato haciendo click en el botón "Descargar instrucciones". También puede descargar un archivo de ejemplo en el botón "Descargar archivo de ejemplo".

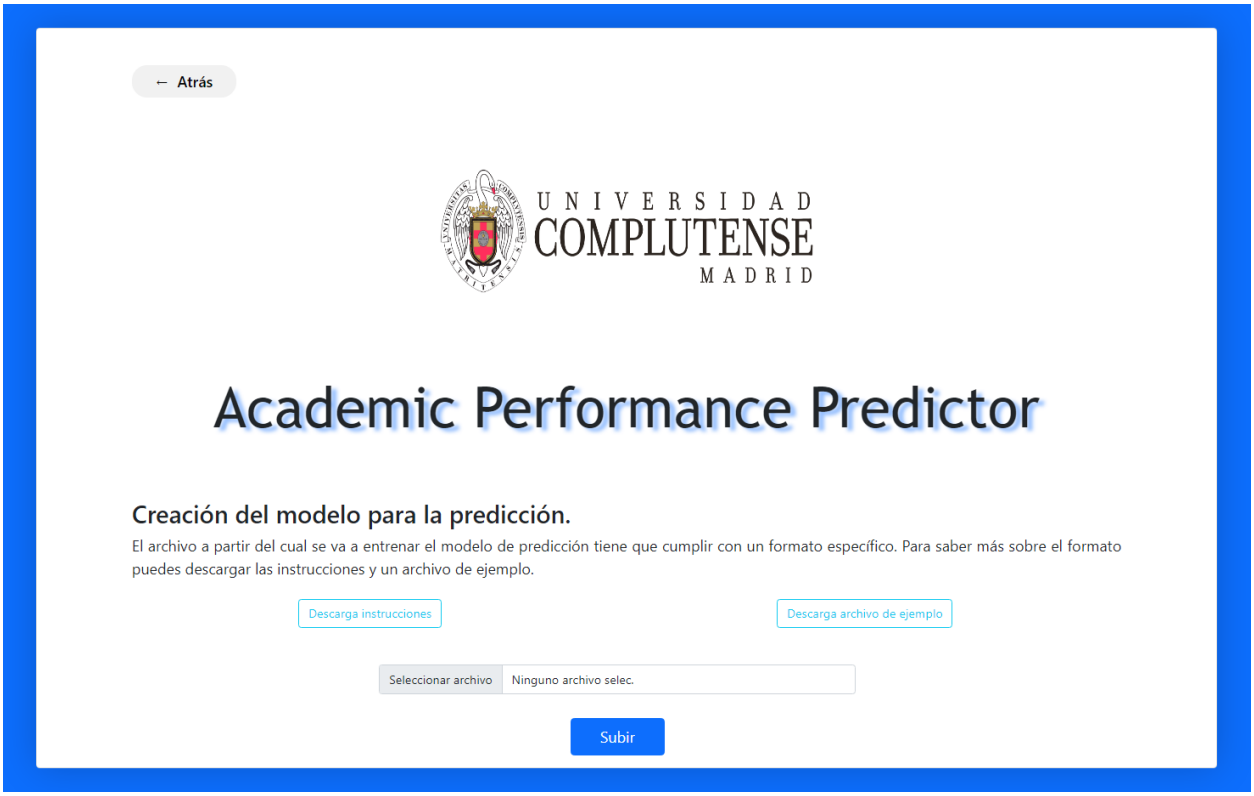

Figura 13. Subir archivo de creación del modelo.

4.4 Seleccionar asignatura y cota

El usuario deberá elegir una de las asignaturas incluidas en el archivo subido anteriormente para realizar la predicción. Seguidamente tendrá que definir una calificación de cota a predecir. La predicción indicará si el alumno, o alumnos, superan dicha calificación en la asignatura seleccionada.

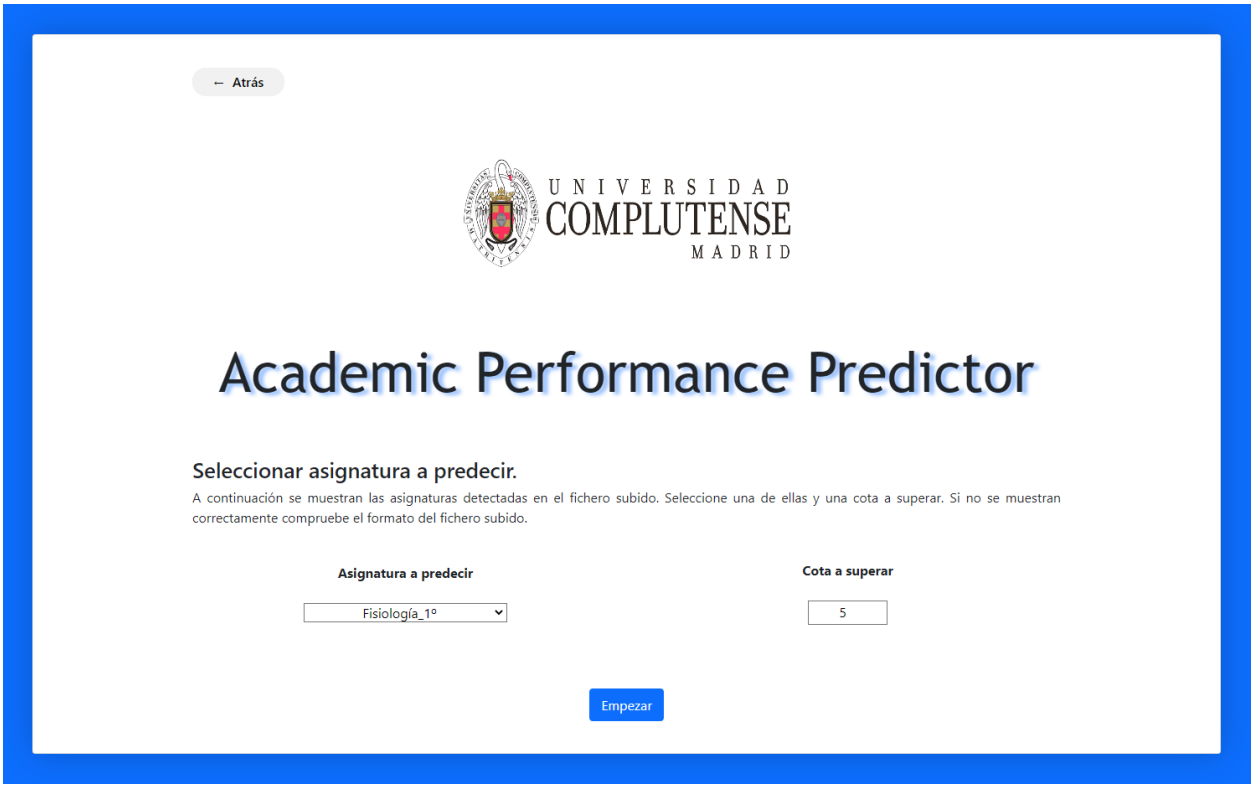

Figura 14. Seleccionar asignatura y cota.

4.5 Seleccionar las variables a considerar en la predicción

El usuario elegirá los distintos datos a tener en cuenta en la predicción. Estos datos son los que están incluidos en el archivo subido para el entrenamiento. Si no quiere tener en cuenta alguno de los datos deberá desmarcar su casilla. Estos datos pueden ser datos socioeconómicos o datos sobre asignaturas. Sólo se podrán considerar las asignaturas de un curso anterior o igual a la asignatura seleccionada para la predicción.

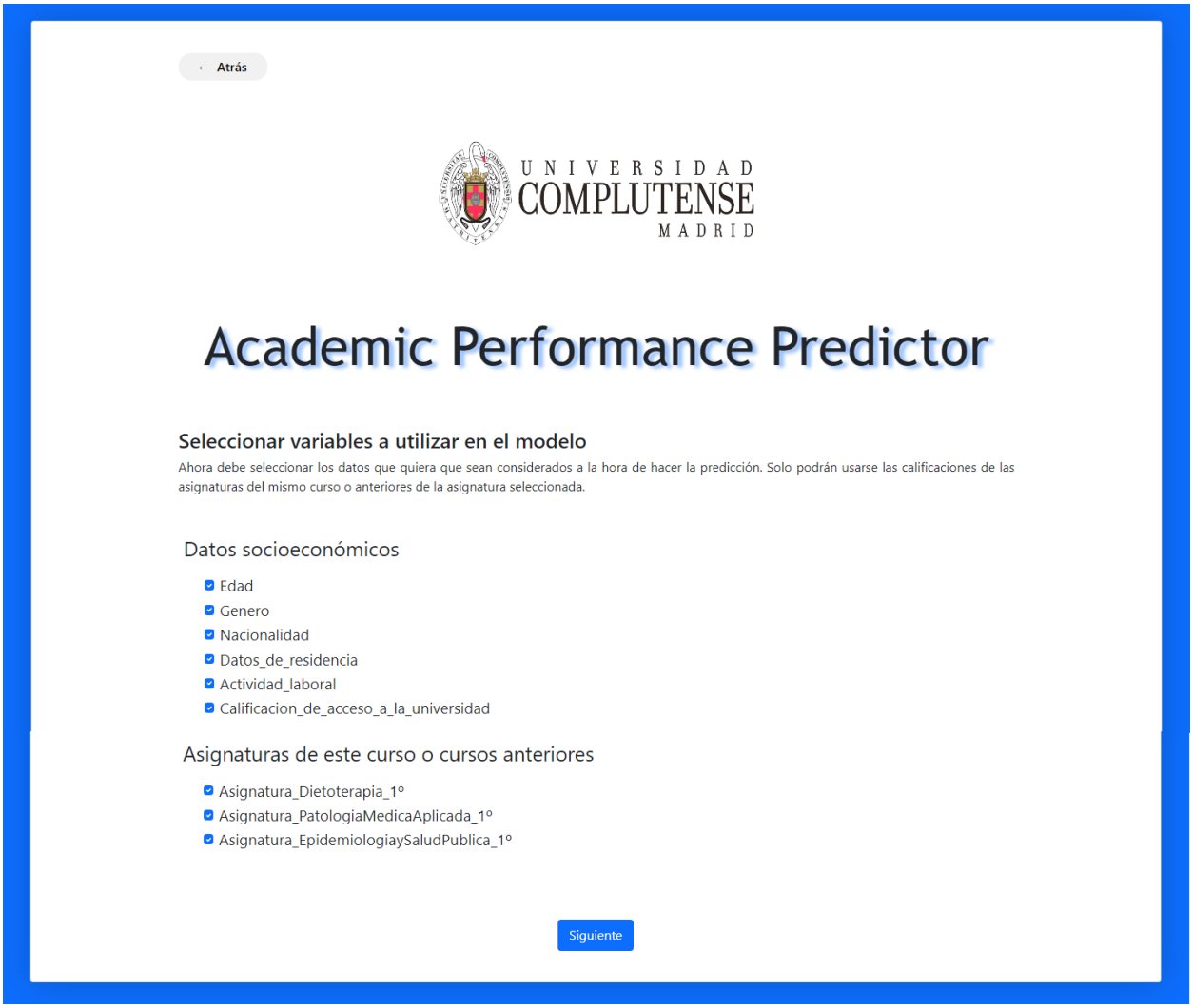

Figura 15. Seleccionar las variables a considerar en la predicción.

4.6 Creación del modelo

Una vez creado el modelo la herramienta muestra sus características con las diferentes imágenes para su análisis. Después de considerar si el modelo es correcto, el usuario puede elegir entre realizar predicciones con ese modelo o volver atrás para crear otro.

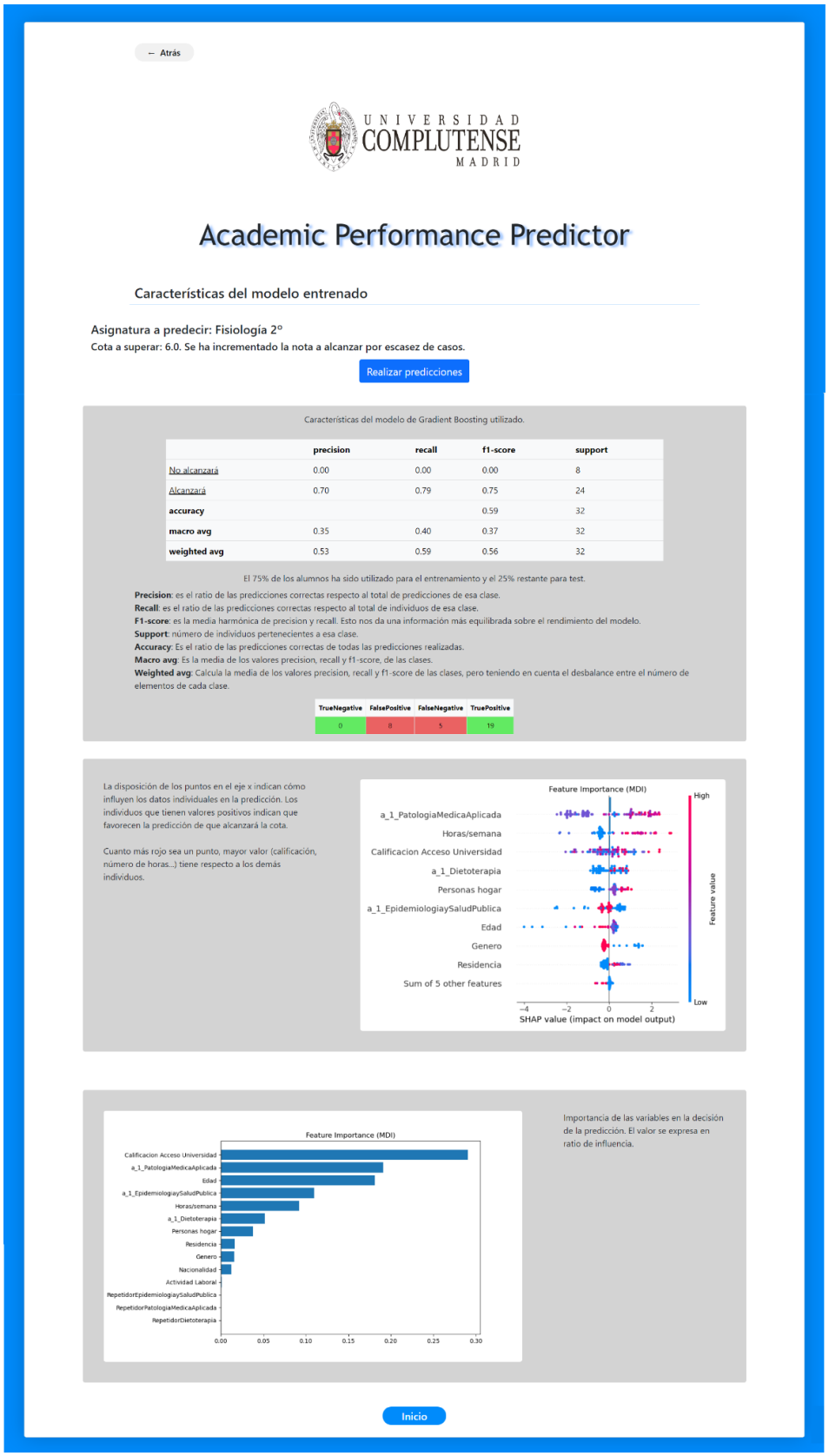

Figura 16. Características del modelo de predicción.

4.7 Tipo de consulta

Una vez definido el modelo, se muestran dos opciones: *predicción individual* o *predicción múltiple*. El usuario debe elegir cuál quiere realizar.

Figura 17. Tipo de consulta.

4.8 Consulta individual

Para hacer una consulta individual se deben introducir los datos del alumno a predecir. Sólo se podrá introducir información sobre los datos seleccionados previamente para la creación del modelo. En la Figura 18 se muestra un ejemplo que ha seleccionado todos los datos socioeconómicos disponibles y tres asignaturas.

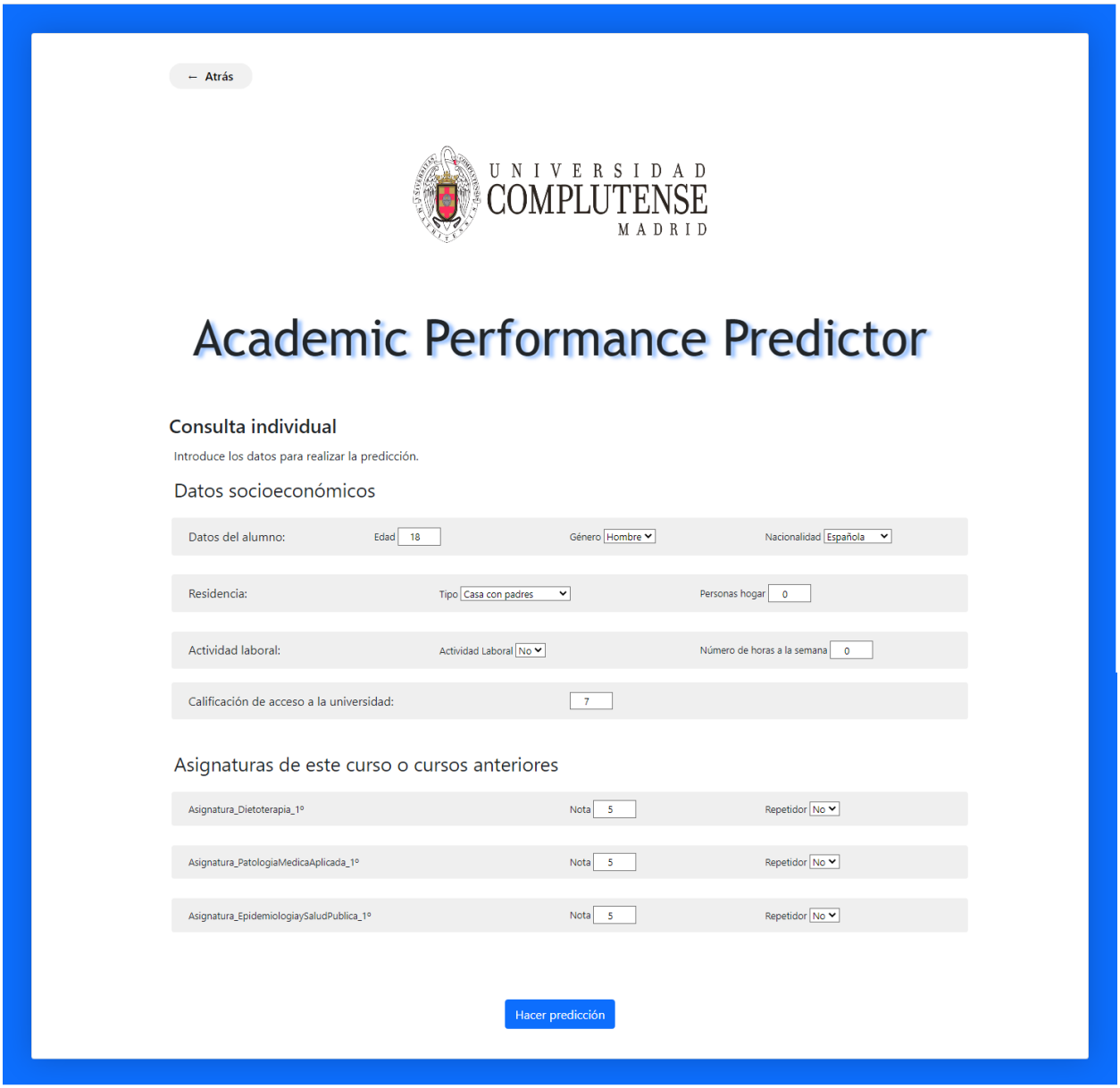

Figura 18. Consulta individual.

4.8.1 Resultados consulta individual

Se muestran los resultados obtenidos de la predicción. La primera tabla indica los resultados de los tres algoritmos. Seguidamente se muestra el gráfico de cascada SHAP que muestra las características de las variables en esa predicción individual. Posteriormente se vuelven a mostrar las características del modelo.

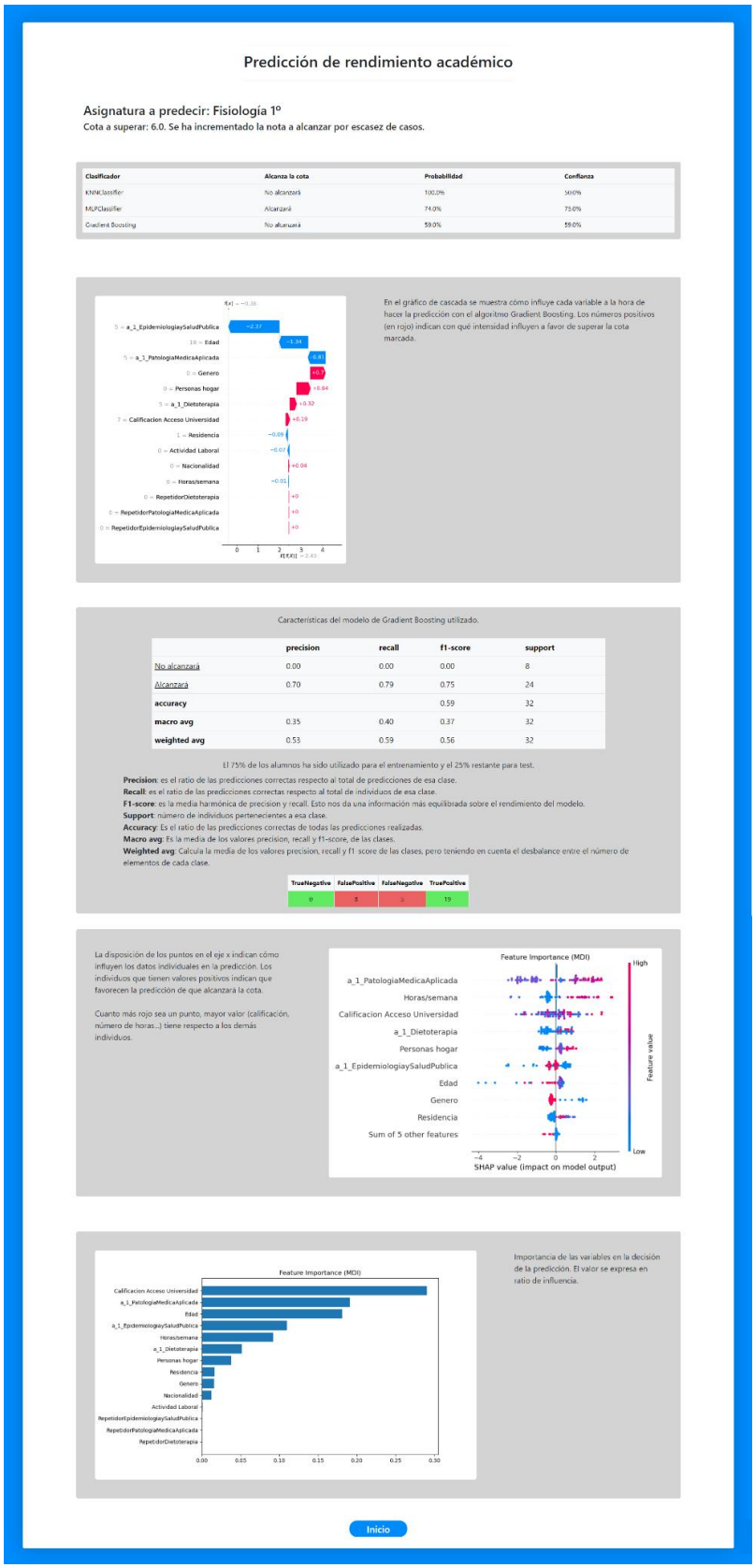

Figura 19. Resultados de la consulta individual.

4.9 Consulta múltiple

Para realizar una consulta múltiple es necesario subir un archivo que contenga todas las consultas individuales. Se debe seleccionar un archivo con el mismo formato que el subido para el modelo. Para ayudar al usuario existen las mismas funcionalidades de "Descarga de instrucciones" y "Descarga de archivo de ejemplo".

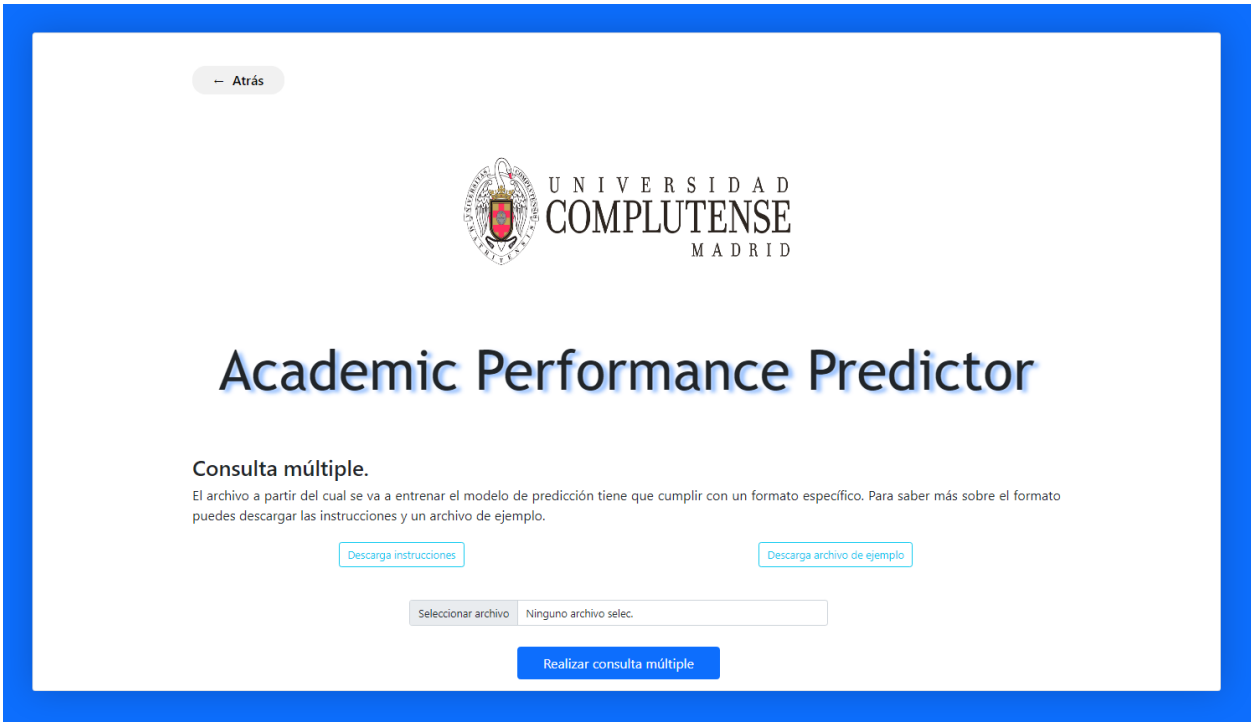

Figura 20. Consulta múltiple.

4.9.1 Resultados consulta múltiple

Los resultados de la predicción se pueden descargar de un archivo que contiene la predicción de los tres algoritmos para cada una de las consultas individuales. En esta página se vuelven a mostrar las características del modelo utilizado.

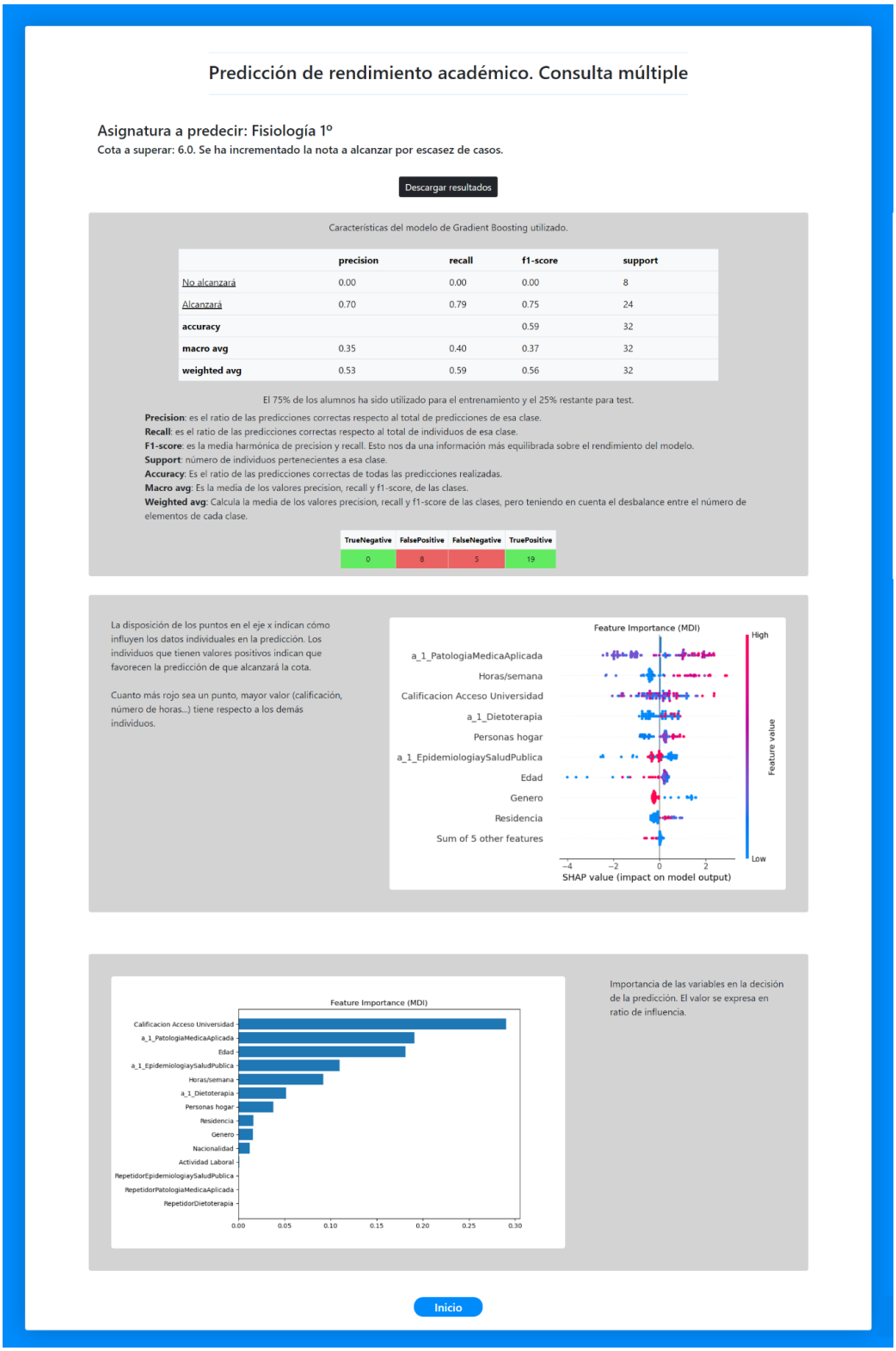

Figura 21. Resultados consulta múltiple.

Capítulo 5 - Resultados obtenidos

En esta sección se muestran los resultados de varias predicciones realizadas con esta herramienta. Para estas predicciones se han utilizado datos del Grado en Medicina de la UCM. Estos datos contienen todos los posibles datos socioeconómicos a analizar, que son:

- Edad
- Género
- Nacionalidad
- Actividad laboral con el número de horas semanales
- Tipo de residencia y número de personas en el hogar

También se incluye la Calificación de Acceso a la Universidad y cuatro asignaturas, dos pertenecen al segundo curso y dos al tercer curso.

El usuario que utilice esta herramienta de predicción deberá tener conocimientos básicos sobre el funcionamiento de los modelos de predicción. Con estos conocimientos debería ser capaz de distinguir entre un modelo adecuado y otro que no lo sea, observando sus características. También será necesario revisar el conjunto de datos introducidos para detectar reglas en el modelo inadecuadas. Esto suele ocurrir cuando hay poca información sobre una o varias variables en el conjunto de entrenamiento.

5.1 Creación de un modelo adecuado

Una parte esencial para obtener resultados significativos es crear modelos de predicción adecuados. Los modelos de predicción adecuados son aquellos que están entrenados correctamente, es decir, tienen una cantidad significativa de diferentes datos que hacen posible realizar buenas predicciones para todo tipo de consultas. Si un algoritmo fuese entrenado con datos insuficientes no podría obtener reglas de predicción que clasificasen correctamente todo tipo de consulta. Sólo podría acertar en una pequeña cantidad de consultas.

Esta herramienta utiliza un 75% de los datos introducidos para entrenamiento y guarda un 25% para realizar test y comprobar el funcionamiento del modelo. De esta forma, las características del modelo se obtienen de realizar predicciones con este 25% de los alumnos introducidos.

Uno de los requisitos para que sea un modelo adecuado es que tenga un porcentaje pequeño de fallos en la matriz de confusión. Aunque, como veremos más adelante, puede cumplir este requisito y no ser adecuado. Es importante considerar que el conjunto de datos del Grado en Medicina tiene un número reducido de estudiantes. Debido a esto hay que seleccionar cuidadosamente la asignatura a predecir, la cota y las variables para poder crear un modelo que sea lo suficientemente adecuado como para hacer predicciones significativas. Si no se eligen correctamente estos tres parámetros nos encontraremos con modelos entrenados con una gran diferencia entre el número de estudiantes pertenecientes a una clase respecto a la otra. Esto implicaría que el modelo predijese con una frecuencia muy alta los de una clase y con una frecuencia baja o nula los de la otra. A continuación se exponen las características de un modelo inadecuado.

Asignatura a predecir: PatologiaMedicaAplicada 3º

Cota a superar: 5.0

Características del modelo de Gradient Boosting utilizado.

	precision	recall	f1-score	support
No alcanzará	0.00	0.00	0.00	5
Alcanzará	0.83	0.93	0.88	27
accuracy			0.78	32
macro avg	0.42	0.46	0.44	32
weighted avg	0.70	0.78	0.74	32

El 75% de los alumnos ha sido utilizado para el entrenamiento y el 25% restante para test.

Precision: es el ratio de las predicciones correctas respecto al total de predicciones de esa clase.
Recall: es el ratio de las predicciones correctas respecto al total de individuos de esa clase.
F1-score: es la media harmónica de precision y recall. Esto nos da una información más equilibrada sobre el rendimiento del modelo.
Support: número de individuos pertenecientes a esa clase.
Accuracy: Es el ratio de las predicciones correctas de todas las predicciones realizadas.
Macro avg: Es la media de los valores precision, recall y f1-score, de las clases.
Weighted avg: Calcula la media de los valores precision, recall y f1-score de las clases, pero teniendo en cuenta el desbalance entre el número de elementos de cada clase.

TrueNegative	FalsePositive	FalseNegative	TruePositive
0	5	2	25

Figura 22. Modelo inadecuado.

En este modelo se cumple el requisito de tener un porcentaje de fallos pequeño en la matriz de confusión (78,12% de aciertos frente al 21,88% de fallos). Este dato no nos debe inducir a pensar que es un modelo adecuado ya que hay que tener en cuenta otro factor, y es que el número de estudiantes pertenecientes a las diferentes clases está desbalanceado. Si observamos el conjunto completo, incluyendo el conjunto de entrenamiento y el de test, solo un 8,66% de los alumnos pertenece a la clase No alcanzará. Debido a esto, los clasificadores asignarán a la gran mayoría de alumnos a la clase Alcanzará. Esto conseguirá un gran porcentaje de aciertos sobre el conjunto de datos introducido, pero, si se quiere generalizar, obtendrá unos resultados muy desfavorables para todos los casos que perteneciesen a la clase No alcanzará, y por lo tanto, no se podrían sacar resultados reveladores de las predicciones.

En la matriz de confusión se puede observar que el clasificador únicamente ha asignado a 2 de 32 estudiantes a la clase No alcanzará. En ambos casos ha fallado. Y de los 5 que realmente pertenecían a la clase No alcanzará no ha acertado la predicción de ninguno de ellos.

5.2 Resultados de las predicciones considerando todas las variables del conjunto de datos

Como se ha comentado anteriormente, un buen indicativo de que un modelo es adecuado es aquel que cumple que los porcentajes de aciertos son altos y el número de individuos de las dos clases es parejo. Debido a que el conjunto de datos era pequeño ha sido algo complejo encontrar una asignatura y una cota que cumpla los requisitos anteriores, pero se ha conseguido con la asignatura *Dietoterapia* y la cota 6.7.

Asignatura a predecir: Dietoterapia 3º

Cota a superar: 6.7

Características del modelo de Gradient Boosting utilizado.

	precision	recall	f1-score	support
No alcanzará	0.73	0.79	0.76	14
Alcanzará	0.77	0.71	0.74	14
accuracy			0.75	28
macro avg	0.75	0.75	0.75	28
weighted avg	0.75	0.75	0.75	28

El 75% de los alumnos ha sido utilizado para el entrenamiento y el 25% restante para test.

Precision: es el ratio de las predicciones correctas respecto al total de predicciones de esa clase.
Recall: es el ratio de las predicciones correctas respecto al total de individuos de esa clase.
F1-score: es la media harmónica de precision y recall. Esto nos da una información más equilibrada sobre el rendimiento del modelo.
Support: número de individuos pertenecientes a esa clase.
Accuracy: Es el ratio de las predicciones correctas de todas las predicciones realizadas.
Macro avg: Es la media de los valores precision, recall y f1-score, de las clases.
Weighted avg: Calcula la media de los valores precision, recall y f1-score de las clases, pero teniendo en cuenta el desbalance entre el número de elementos de cada clase.

TrueNegative	FalsePositive	FalseNegative	TruePositive
11	3	4	10

Figura 23. Características de un modelo adecuado.

En el conjunto de test hay el mismo número de individuos pertenecientes a cada una de las dos clases. En el total de individuos del conjunto completo hay 53 pertenecientes a la clase Alcanzará y 59 a la clase No alcanzará. El porcentaje de acierto en los datos de test ha sido un 75%, que, aunque sea menor que el modelo anterior, es un resultado aceptable. En este modelo se han seleccionado todas las variables posibles a tener en consideración: los 5 datos socioeconómicos, la Calificación de Acceso a la Universidad y las 3 asignaturas restantes. Para poder explicar de una forma más ilustrativa se ha decidido hacer las comparaciones entre dos predicciones individuales.

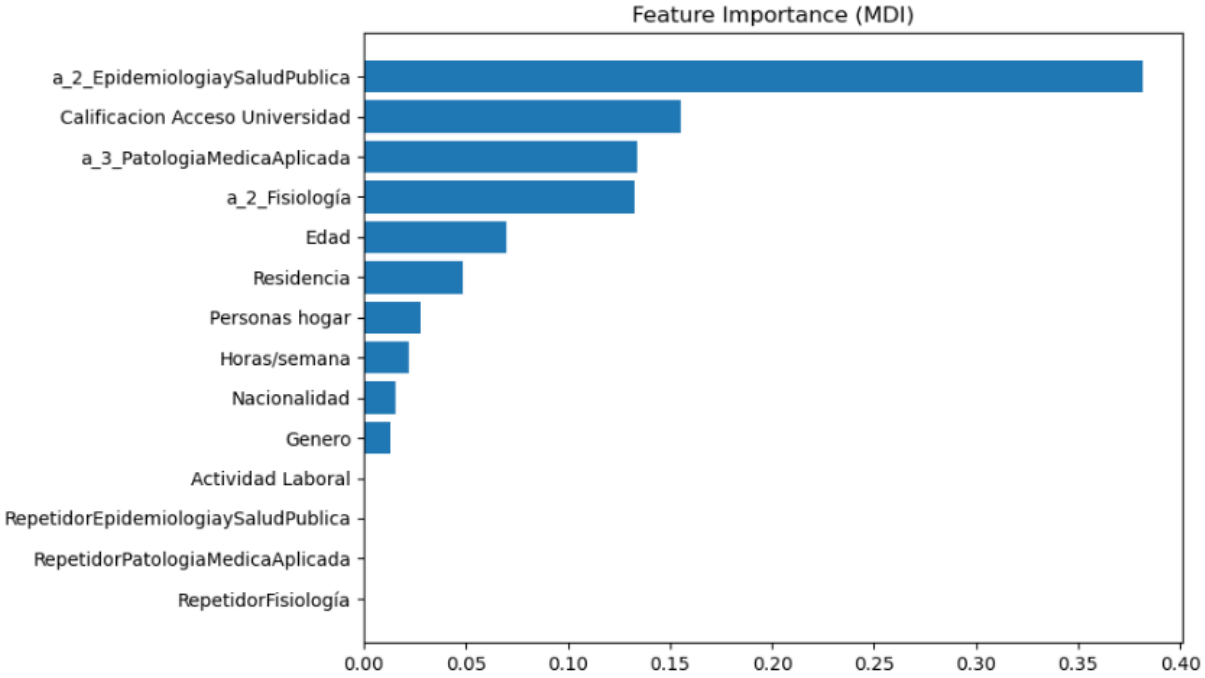

Figura 24. Gráfico Feature Importance.

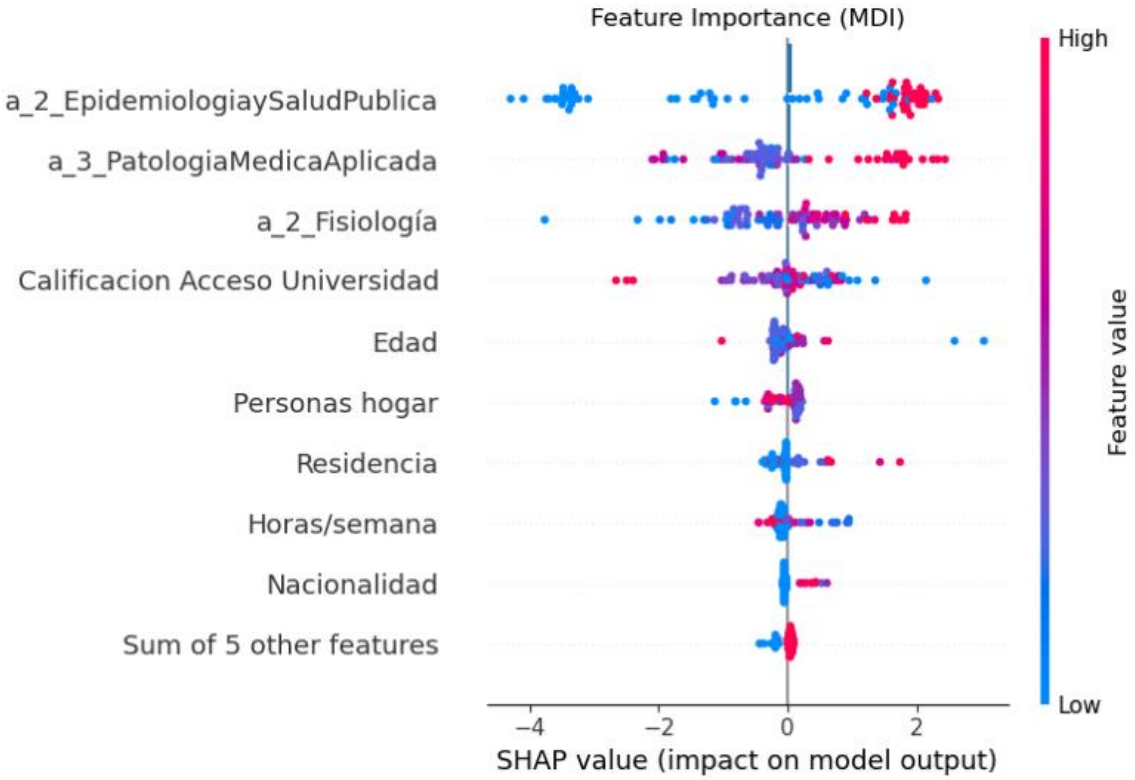

Figura 25. Gráfico Beeswarm

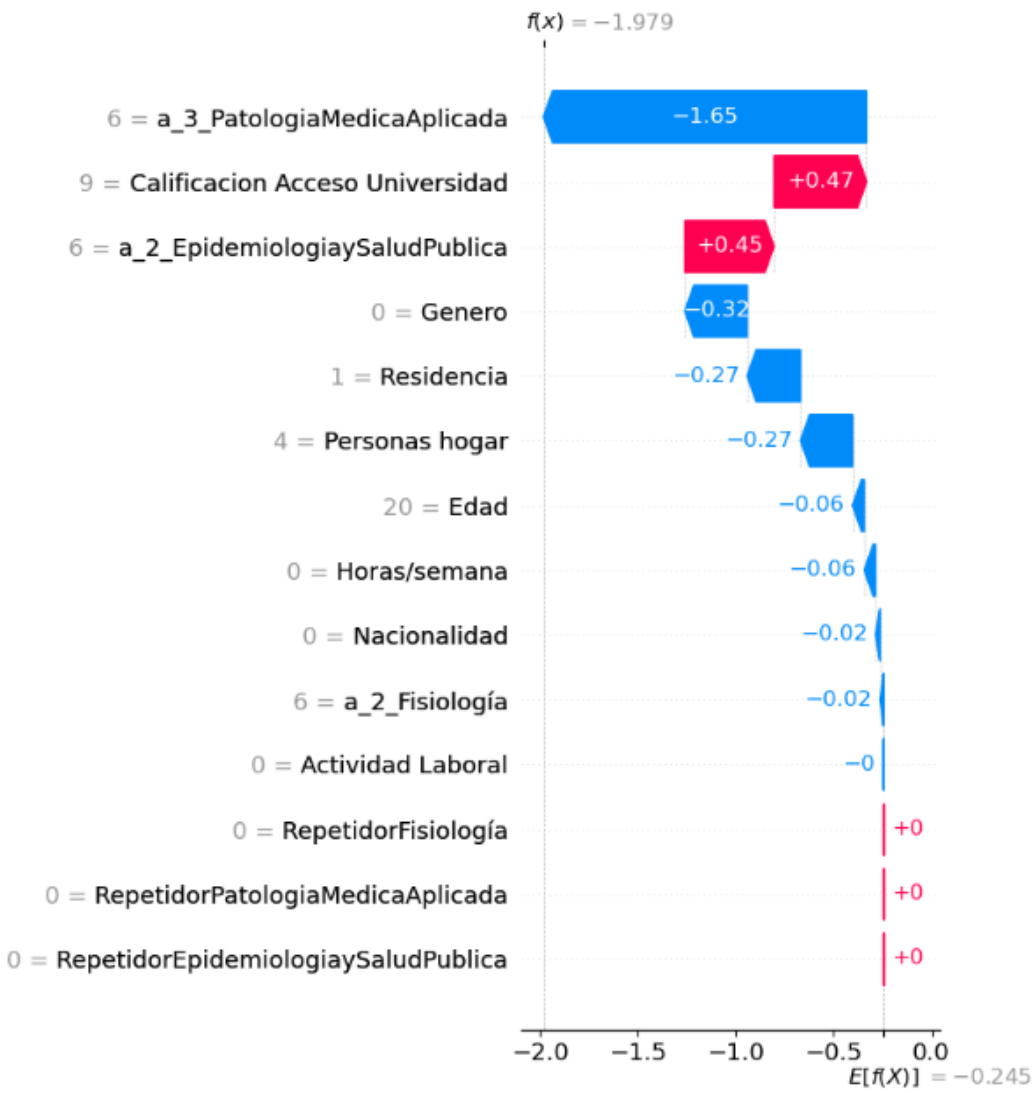

Figura 27. Gráfico en cascada de la predicción 1.

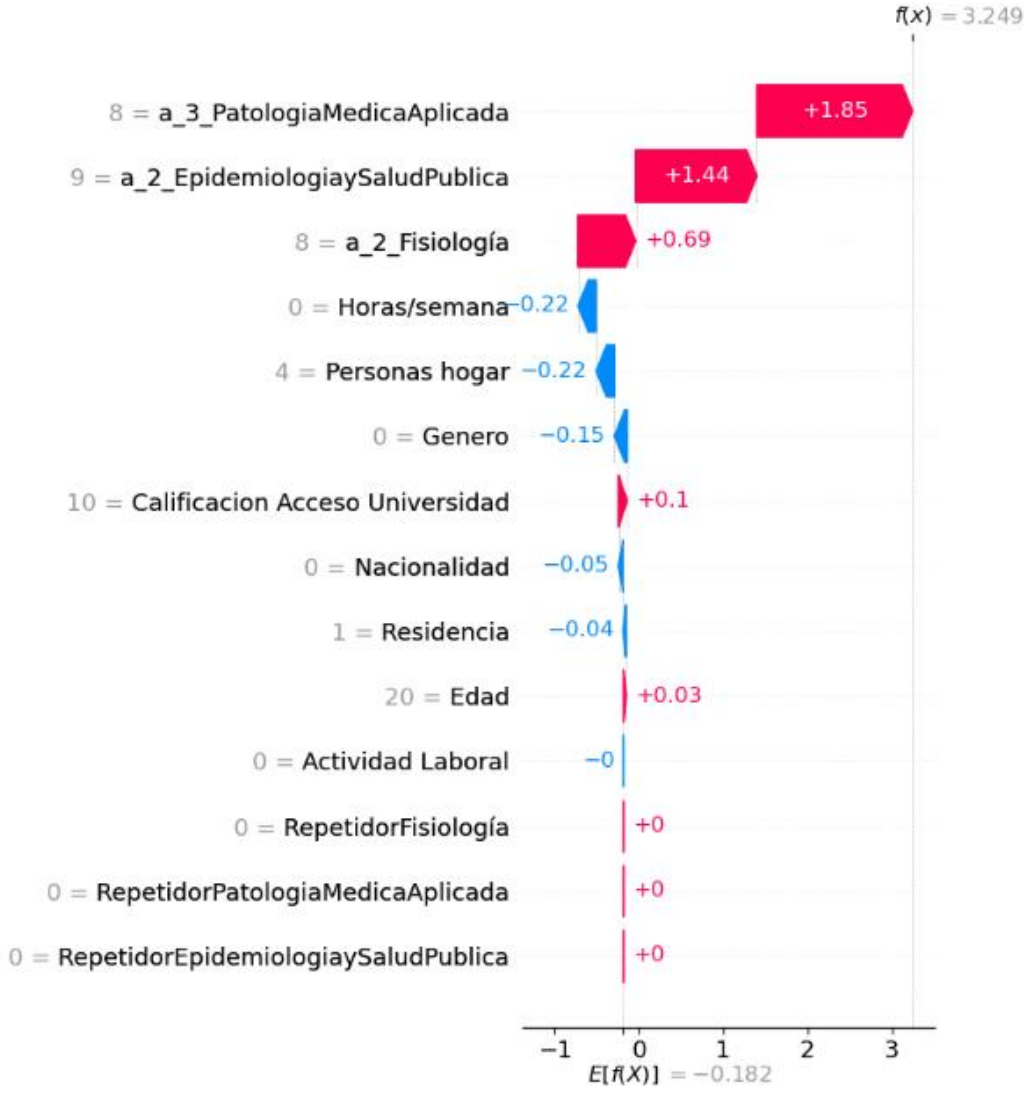

Figura 26. Gráfico en cascada de la predicción 2.

Después de varias ejecuciones se ha observado que, en este modelo, las variables que cobran mayor importancia son las calificaciones de las 3 asignaturas y la Calificación de Acceso a la Universidad. Debido a esto se ha decidido hacer una comparación entre predicciones que tuviesen los mismos datos socioeconómicos.

Según el gráfico Feature Importance, la variable más reveladora es la calificación en Epidemiología, pero, en los casos seleccionados para la comparación, se considera más importante la calificación en Patología Médica Aplicada.

Esta calificación es directamente proporcional a la posibilidad de superar la cota, algo que no pasa con la Calificación de Acceso a la Universidad. Esto puede ser contraintuitivo ya que los alumnos con mayor nota de acceso suelen ser mejores estudiantes. En este caso una calificación de 9 suma 0,47 a favor de superar la cota, y una calificación de 10 suma únicamente 0,1. El gráfico Beeswarm de la Figura 25 nos muestra que los puntos con calificaciones altas se quedan centrados, es decir, no influyen ni positiva, ni negativamente en la superación de la cota. Sin embargo, se puede observar que unas menores calificaciones influyen positivamente en la superación de la cota.

Uno de los resultados que debe llamar la atención, es que la calificación de 6 en Epidemiología sume 0,45, cuando, en teoría, una calificación así debería ser favorable para la clase negativa (No alcanzará). Esto se debe a una escasez de datos que ha producido un modelo que, para las calificaciones menores de 7 en Epidemiología, muestra unos resultados inadecuados. Aunque en el gráfico Beeswarm los puntos de los alumnos que sacan una nota menor de 7 en Epidemiología están bastante repartidos a lo largo del eje x, lo que sucede es que solo hay dos alumnos con notas 6 y 6.1 respectivamente en Epidemiología, y estos dos alumnos pertenecen a la clase Alcanzará. Los demás alumnos tiene una calificación mayor o igual a 6.6 en Epidemiología. Luego son estos dos casos los que producen el desbalance en la predicción 1.

5.3 Resultados de las predicciones considerando únicamente los datos socioeconómicos del conjunto de datos

Se ha generado un nuevo modelo de predicción con la misma asignatura y la misma cota, debido a que, tras varias ejecuciones, se ha demostrado que es lo que obtiene mejores resultados. En este caso se seleccionan únicamente datos socioeconómicos, lo que hace que los resultados de las predicciones no sean tan buenos como con el modelo anterior.

Características del modelo de Gradient Boosting utilizado.

	precision	recall	f1-score	support
No alcanzará	0.71	0.80	0.75	15
Alcanzará	0.73	0.62	0.67	13
accuracy			0.71	28
macro avg	0.72	0.71	0.71	28
weighted avg	0.72	0.71	0.71	28

El 75% de los alumnos ha sido utilizado para el entrenamiento y el 25% restante para test.

Precision: es el ratio de las predicciones correctas respecto al total de predicciones de esa clase.
Recall: es el ratio de las predicciones correctas respecto al total de individuos de esa clase.
F1-score: es la media harmónica de precision y recall. Esto nos da una información más equilibrada sobre el rendimiento del modelo.
Support: número de individuos pertenecientes a esa clase.
Accuracy: Es el ratio de las predicciones correctas de todas las predicciones realizadas.
Macro avg: Es la media de los valores precision, recall y f1-score, de las clases.
Weighted avg: Calcula la media de los valores precision, recall y f1-score de las clases, pero teniendo en cuenta el desbalance entre el número de elementos de cada clase.

TrueNegative	FalsePositive	FalseNegative	TruePositive
12	3	5	8

Figura 28. Características del modelo.

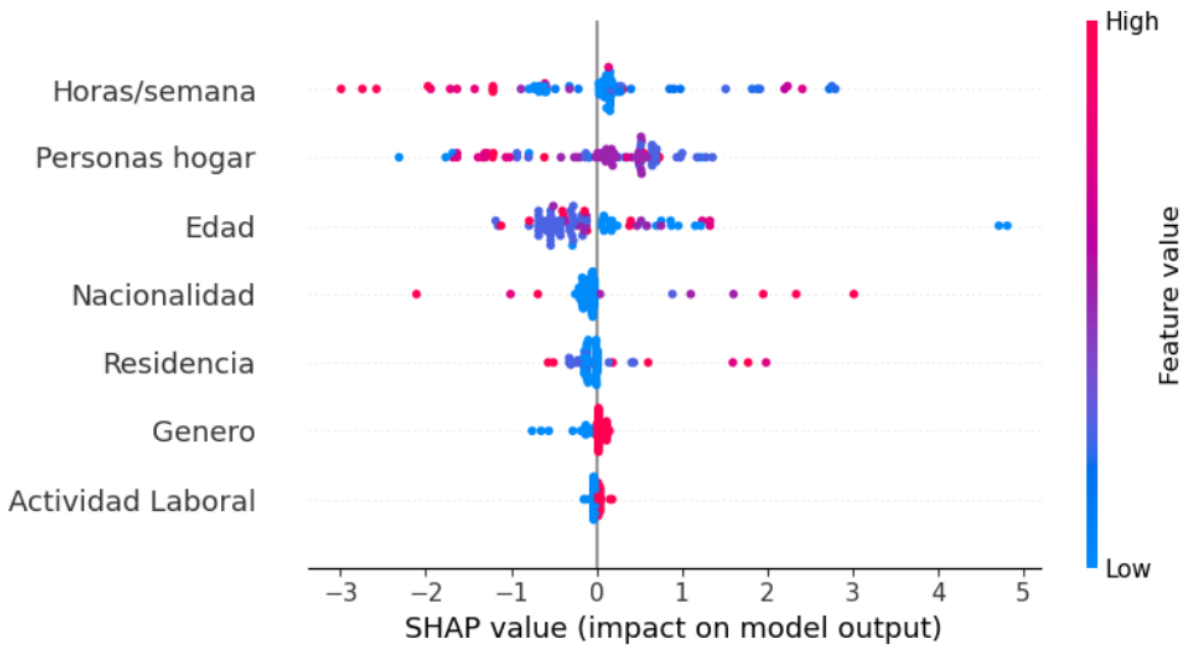

Figura 29. Gráfico Beeswarm segundo modelo.

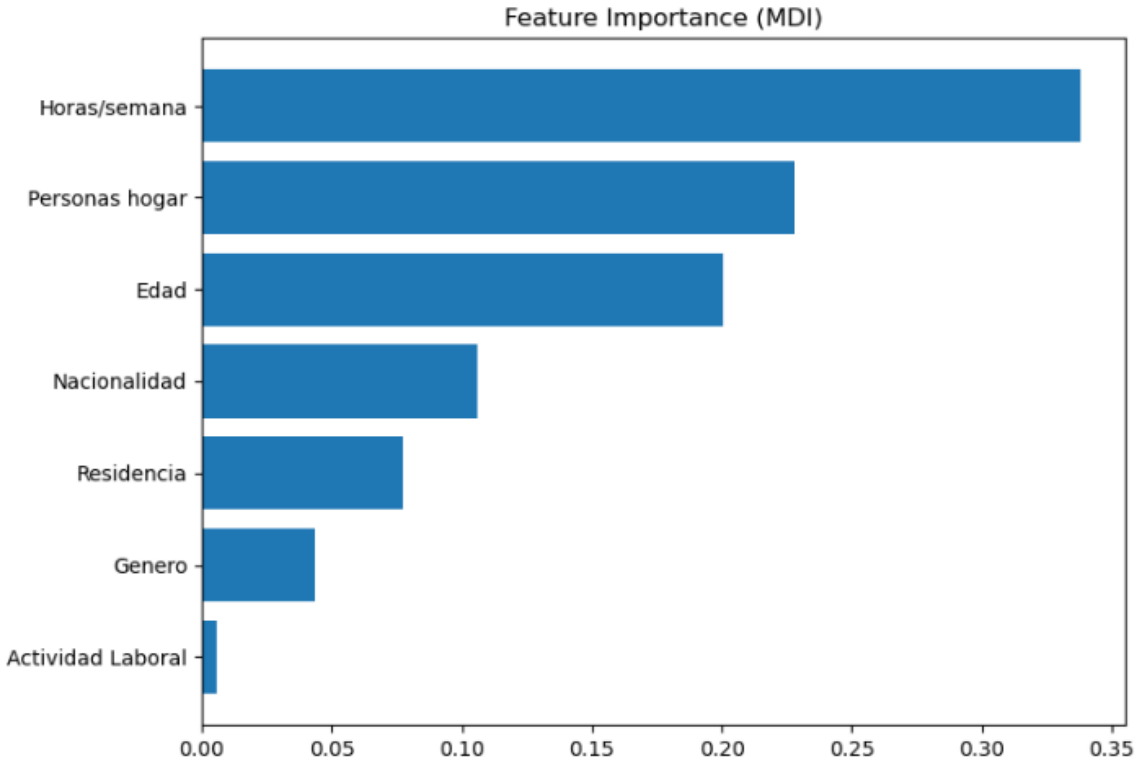

Figura 30. Gráfico Feature Importance segundo modelo

Para sacar conclusiones reveladoras se han realizado numerosas predicciones individuales. Hay dos variables que no se pueden tener en cuenta en las predicciones porque no se tienen datos suficientes: nacionalidad y tipo de residencia. Debido a que la mayoría de los alumnos pertenecen a un solo tipo de todos los posibles. De las demás variables se pueden sacar conclusiones importantes.

- Horas de trabajo a la semana: de 1 a 7 suman para superar la cota. A pesar de que es un grupo reducido puede ser significativo. El grupo mayor es el de 0 horas y, en general, también suma. A partir de las 8 horas ya empieza a restar, siendo el grupo más grande el de 20 horas y, por lo tanto, el más representativo.

- Número de personas en el hogar: en este conjunto de datos el mayor número de personas ha sido 8. El mayor grupo es el de 3 personas en el hogar. En general, el algoritmo considera negativo para alcanzar la cota un número mayor o igual que 4. Y, aunque no se observe de manera clara en el gráfico Beeswarm, de 0 a 3 influye positivamente. A partir de un número mayor o igual a 6 no se pueden sacar conclusiones porque son muy pocos casos.

- Edad: sacar conclusiones de las edades menores o iguales a 19 sería un error ya que sólo existen 3 casos que tienen una calificación en Dietoterapia. En general, la edad de 20 años influye positivamente pero de forma ligera. Las edades de 21 años y 22 años influyen negativamente y es en la edad de 23 años o superior cuando influye positivamente de forma decisiva.

- Género: en el caso de las mujeres influye positivamente de forma tenue pero homogénea. Aunque sólo hay un 22% de hombres en el conjunto de datos, esta variable es una representación significativa.

Capítulo 6 - Conclusiones y trabajo futuro

La tecnología ha transformado nuestro estilo de vida en los últimos años. La forma en que nos entretenemos, comunicamos y trabajamos ha cambiado debido a la incorporación de las nuevas tecnologías. En el campo de la educación se ha mejorado mucho el sistema educativo, pero no se ha llevado a cabo una transformación tan radical como en otros ámbitos. Se sigue el mismo modelo de hace 50 años: un profesor enseñando materias de la misma manera que se hace actualmente.

En mi opinión, el desarrollo de la inteligencia artificial puede traer numerosos beneficios para la educación. Esta herramienta de software hace un análisis del cual se pueden sacar conclusiones que ayuden a la mejora educativa. Lo primero que nos viene a la mente es una mejora en la optimización de recursos, pero esto solo es una pequeña parte de todos los cambios que podría traer la inteligencia artificial en la educación. Softwares como este son sólo un inicio de todos los posibles cambios que se pueden llevar a cabo para mejorar la educación, y por lo tanto, nuestras vidas.

Este software puede mejorarse añadiendo cualquier tipo de dato socioeconómico, lo que puede traer resultados reveladores. A la hora de hacer la matrícula en la universidad, el alumno proporciona muchos datos socioeconómicos que no se tienen en cuenta en el software actual, pero que igualmente nos podría brindar una información muy valiosa. Aunque este software está diseñado para estudiantes universitarios, se podría ampliar fácilmente para poderse aplicar en cualquier institución educativa.

La educación no trata solo de formar personas en el ámbito laboral. Su objetivo debe ser formar personas completas para conseguir que, el hijo del ignorante no tenga por qué ser ignorante, o que el hijo del pobre no tenga que ser pobre. (8)

Chapter - Conclusions and future work

Technology has transformed our lifestyles in recent years. The way we entertain ourselves, communicate and work has changed due to the incorporation of new technologies. In the field of education, the education system has been greatly improved, but there has not been a radical transformation as in other areas. It is still the same model as 50 years ago: a teacher teaching subjects in the same way as it is done today.

In my opinion, the development of artificial intelligence can bring many benefits to education. This software tool makes an analysis from which conclusions can be drawn to help improve education. The first thing that comes to mind is an improvement in the optimization of resources, but this is only a small part of all the changes that artificial intelligence could bring to education. Software like this is just a start of all the possible changes that can be made to improve education, and therefore, our lives.

This software can be enhanced by adding any kind of socio-economic data, which can bring revealing results. When registering for university, the student provides many socio-economic data that are not taken into account in the current software, but which could still provide valuable information. Although this software is designed for university students, it could easily be extended to any educational institution.

Education is not just about training people for the workplace. Its aim should be to train complete people so that the child of the ignorant does not have to be ignorant, or the child of the poor does not have to be poor (8).

BIBLIOGRAFÍA

1. **Sierra, Basilio.** *Aprendizaje automático: conceptos básicos y avanzados.* Madrid : PEARSON PRENTICE HALL, 2006.

2. **Mi Diario Python.** Introducción al Machine Learning #9 - K Vecinos más cercanos (Clasificación y Regresión). [En línea] Mi Diario Python. https://pythondiario.com/2018/01/introduccion-al-machine-learning-9-k.html.

3. **cienciadedatos.net.** Redes neuronales con Python. [En línea] cienciadedatos.net. https://www.cienciadedatos.net/documentos/py35-redes-neuronales-python.html.

4. **García, Jesús Manuel de la Cruz.** *Aprendizaje automático : un enfoque práctico.* Madrid : Ra-Ma, 2010.

5. **Scikit-learn.** Scikit-learn. [En línea] Scikit-learn. https://scikit-learn.org/.

6. **SHAP.** Librería SHAP. [En línea] https://github.com/slundberg/shap.

7. **Van Rossum, G., & Drake Jr, F. L.** *Python reference manual.* Centrum voor Wiskunde en Informatica Amsterdam : CreateSpace, 1995.

8. **Savater, Fernando.** La importancia de la educación. [En línea] 2015. https://www.youtube.com/watch?v=OuSpnCMncN4&ab_channel=JulianJurth.

9. **Aurelien, Géron.** *Hands-on machine learning with scikit-learn and tensorflow : concepts, tools, and techniques to build intelligent systems.* Sebastopol : O'Reilly Media, 2017.

10. **Serrano, Alberto García.** *Inteligencia artificial : fundamentos, práctica y aplicaciones.* Madrid : RC Libros, D.L., 2016.

APÉNDICES

Apéndice A - Instrucciones para configurar un archivo de datos correcto para el uso del sistema de predicción Academic Performance Predictor

El archivo a partir del cual se va a crear el modelo de predicción tiene que cumplir las siguientes características:

1. **El formato del archivo tiene que ser .xlsx.**
2. **No puede haber columnas sin título.**
3. **En las calificaciones no puede haber datos en blanco. En caso de desconocer una calificación se tiene que escribir el número 99.**
4. **Debe existir una columna "ID" en la cual se indique un número de alumno.**
5. **Tiene que existir una columna para cada uno de los siguientes datos socioeconómicos.** Cada dato tiene que estar configurado como se indica a continuación. Algunos de estos datos tienen que estar codificados numéricamente. Estas columnas pueden estar vacías.
   - Edad: Número entero entre 16 y 98.
   - Genero:
     - Masculino: 0
     - Femenino: 1
   - Nacionalidad:
     - Española: 0
     - Italiana: 1
     - Venezolana: 2
     - Polaca: 3
     - China: 4
     - Alemana: 5
     - Rumana: 6
     - Inglesa: 7
     - Búlgara: 8
     - Peruana: 9
     - Colombiana: 10

    - Marroquí: 11
    - Portuguesa: 12
    - Otras: 13
- Residencia:
    - Casa con padres: 1
    - Casa con compañeros: 2
    - Colegio Mayor: 3
    - Casa individual: 4
    - Otros: 5
- Personas hogar: Número entero entre 0 y 98.
- Actividad Laboral:
    - No: 0
    - Sí: 1
- Horas/semana: Número entero entre 0 y 40.
- Calificación Acceso Universidad: Número entero entre 0 y 14.

6. **El formato del título de las columnas de calificación de una asignatura tiene que cumplir el siguiente estilo**: a_*curso*_*Nombre*. Por ejemplo: a_1_Fisiología. La calificación debe estar comprendida entre 0 y 10.
7. **Por cada columna de asignatura debe existir una columna para indicar si el alumno ha repetido esa asignatura. El formato del título de esa columna es**: Repetidor*Nombre*. Donde el valor 1 indica que el alumno ha repetido y el valor 0 indica que no he repetido dicha asignatura. Ejemplo: RepetidorFisiología.