Universidad Politécnica
de Madrid

Escuela Técnica Superior de Ingenieros Informáticos

Máster Universitario en Inteligencia Artificial

Trabajo Fin de Máster

A tool for human evaluation of interpretability

Autor(a): Adrian Vargas Rangel
Tutor(a): Bojan Mihaljevic

Madrid, julio 2024

Este Trabajo Fin de Máster se ha depositado en la ETSI Informáticos de la Universidad Politécnica de Madrid para su defensa.

Trabajo Fin de Máster
Máster Universitario en Inteligencia Artificial

Título: A tool for human evaluation of interpretability

julio 2024

Autor(a): Adrian Vargas Rangel
Tutor(a): Bojan Mihaljevic
Inteligencia Artificial
ETSI Informáticos
Universidad Politécnica de Madrid

Resumen

A medida que los sistemas de inteligencia artificial (IA) se integran en diversos sectores de la sociedad, la necesidad de modelos de aprendizaje automático interpretables se vuelve crucial para su aceptación y uso ético. En áreas críticas como la medicina, donde los resultados de los algoritmos utilizados para diagnósticos deben ser precisos y comprensibles, es esencial desarrollar nuevas técnicas de explicabilidad que faciliten la toma de decisiones y garanticen la fiabilidad de estas tecnologías.

Este Trabajo de Fin de Máster se inspira en la metodología del estudio *"Interpretable Decision Sets: A Joint Framework for Description and Prediction"* [1] y desarrolla un cuestionario para evaluar la interpretabilidad de modelos transparentes, como los árboles de decisión y el modelo de Interpretable Decision Sets (IDS) propuesto en dicho estudio. Para la evaluación, se emplean datos sobre el rendimiento académico en matemáticas de estudiantes, proporcionados por Paulo Cortez y disponibles en el *UCI Machine Learning Repository* [2].

El cuestionario está diseñado para extraer las reglas subyacentes de cada modelo, clasificando a los estudiantes como aprobados o no aprobados, y evaluando la capacidad de los usuarios para interpretar correctamente estas reglas, incluso en situaciones ambiguas. Esta evaluación incluye tanto la exactitud de las predicciones como la habilidad de los usuarios para identificar y comprender errores, lo cual es fundamental para fomentar la confianza y el uso efectivo de la IA en decisiones reales.

Con este enfoque, se busca contribuir al campo de la inteligencia artificial explicable proporcionando una base sólida para la construcción de herramientas de investigación que exploren cómo los humanos interpretan las decisiones de modelos transparentes.

El código fuente del cuestionario desarrollado está disponible en el repositorio de GitHub: https://github.com/adrian-vargas/survey-xai.

Abstract

As artificial intelligence (AI) systems become increasingly integrated into various sectors of society, the need for interpretable machine learning models is crucial for their acceptance and ethical use. In critical fields such as medicine, where the results of algorithms used for diagnostics must be both accurate and understandable, it is essential to develop new explainability techniques that facilitate decision-making and ensure the reliability of these technologies.

This Master's Thesis is inspired by the methodology from the study "Interpretable Decision Sets: A Joint Framework for Description and Prediction" [1] and develops a questionnaire to assess the interpretability of transparent models, such as decision trees and the Interpretable Decision Sets (IDS) model proposed in that study. For the evaluation, data on students' academic performance in mathematics, provided by Paulo Cortez and available in the UCI Machine Learning Repository [2], were used.

The questionnaire is designed to extract the underlying rules of each model, classifying students as passing or failing, and evaluating the ability of users to correctly interpret these rules, even in ambiguous situations. This evaluation includes both the accuracy of the predictions and the ability of users to identify and understand errors, which is fundamental to fostering trust and effectively using AI in real-world decisions.

This approach aims to contribute to the field of explainable artificial intelligence by providing a solid foundation for building research tools that explore how humans interpret the decisions of transparent models.

The source code for the developed questionnaire is available on the GitHub repository: https://github.com/adrian-vargas/survey-xai.

Tabla de contenidos

Capítulo 1

Introducción

En este capítulo se ofrece una visión general del contexto de la investigación. En la Sección 1.1, se aborda la motivación que ha llevado a la realización de este proyecto, explicando las razones que justifican su relevancia. La Sección 1.2 describe el objetivo principal y los objetivos específicos que se persiguen, mientras que la Sección 1.3 plantea la hipótesis que se pretende validar con este estudio. Finalmente, en la Sección 1.4 se presenta la estructura del documento como guía sobre el contenido de cada uno de los capítulos.

1.1. Motivación

Al cursar la asignatura Ïnteligencia Artificial Explicable.[el] inicio de mi carrera en inteligencia artificial,

Mi interés por la inteligencia artificial explicable se confirmó al cursar dicha asignatura en el Máster de Inteligencia Artificial, impartida por los profesores Bojan Mihaljevic y Esteban García Cuesta, ya que ha surgido una creciente preocupación dentro de la comunidad científica y de la sociedad en general por la falta de transparencia en muchos sistemas de IA, especialmente en sectores críticos como la medicina, la educación y la justicia. En estos ámbitos, las decisiones automatizadas pueden tener consecuencias significativas y, a veces, irreversibles, lo que hace imperativa la necesidad de modelos más comprensibles y justos [3].

La urgencia de desarrollar modelos interpretables se hace evidente a la luz de casos recientes como el escándalo de los subsidios para el cuidado infantil en los Países Bajos en 2018. Un sistema automatizado de detección de fraudes, implementado por la Agencia de Impuestos neerlandesa, etiquetó incorrectamente a más de 26,000 familias, en su mayoría de origen migrante, como fraudulentas [4, 5]. Estas acusaciones erróneas dieron lugar a demandas de devolución de sumas de dinero que a veces alcanzaron hasta 100.000 euros [6], provocando graves consecuencias sociales y económicas, como la pérdida de empleo y vivienda, así como un aumento de problemas de salud mental entre los afectados.

Otro ejemplo que subraya la importancia de la interpretabilidad es el caso Loomis [7]. En este caso, el uso del software COMPAS (Correctional Offender Management Profiling for Alternative Sanctions) en el sistema judicial de Estados Unidos, diseñado para predecir la probabilidad de reincidencia de un acusado, fue criticado por su falta

de transparencia y explicabilidad. Un estudio realizado por ProPublica en 2016 reveló que COMPAS no solo era menos preciso de lo que se afirmaba, sino que también presentaba sesgos raciales significativos, clasificando a personas negras como de mayor riesgo de reincidencia en comparación con personas blancas con historiales similares [8].

Estos incidentes destacan la necesidad de modelos de inteligencia artificial que sean no solo técnicamente eficientes, sino también transparentes e interpretables para evitar decisiones injustas.

Basándose en el estudio “Interpretable Decision Sets (IDS)"de Lakkaraju et al. [1], esta investigación se centra en evaluar la interpretabilidad de modelos explicables desde la perspectiva de los usuarios finales.

1.2. Objetivo

El objetivo de este Trabajo de Fin de Máster (TFM) es desarrollar un cuestionario para evaluar la interpretabilidad de modelos de decisión, específicamente de los modelos *Interpretable Decision Sets* (IDS) y los árboles de decisión (DT), en el contexto de la inteligencia artificial. Este cuestionario, inspirado en el estudio “Interpretable Decision Sets: A Joint Framework for Description and Prediction” [1], será utilizado para investigar cómo diversos factores, como la estructura del modelo y la presentación visual de los resultados, influyen en la percepción de interpretabilidad y en la capacidad de los usuarios para detectar errores.

1.3. Objetivos Específicos

1. Diseñar un cuestionario para evaluar la percepción de interpretabilidad de los modelos *Interpretable Decision Sets* (IDS) y los árboles de decisión (DT).
2. Implementar el cuestionario utilizando un conjunto de datos sobre el rendimiento académico de estudiantes en matemáticas [2].
3. Analizar cómo factores como la estructura del modelo, la ambigüedad de la información y la confianza en las visualizaciones afectan la percepción de interpretabilidad.
4. Desarrollar una herramienta que sirva como base para futuros estudios en el campo de la inteligencia artificial explicable.

1.4. Hipótesis

Se plantea que los modelos IDS serán percibidos como más interpretables que los árboles de decisión, especialmente en situaciones de ambigüedad. No obstante, la confianza excesiva en las visualizaciones puede llevar a una comprensión superficial, afectando la capacidad de los usuarios para identificar errores correctamente. Por tanto, se sugiere que tanto la estructura del modelo como la presentación de resultados influyen en la percepción de interpretabilidad y precisión.

1.5. Estructura del Documento

El presente trabajo se organiza en los siguientes capítulos:

- Capítulo 1: Introducción. Este capítulo proporciona una introducción general al tema del TFM, incluyendo el contexto, la motivación del estudio, el objetivo principal, los objetivos específicos, la hipótesis, y la estructura general del documento.
- Capítulo 2: Fundamentos Teóricos. Se revisan los conceptos teóricos clave relacionados con la interpretabilidad en la inteligencia artificial, así como una descripción de los modelos que se evaluarán, como los Interpretable Decision Sets (IDS) y los Árboles de Decisión (DT). Incluye definiciones de términos, explicaciones de técnicas y metodologías relevantes, y una discusión sobre la importancia de la interpretabilidad en la inteligencia artificial.
- Capítulo 3: Estado del Arte. Este capítulo presenta una revisión de los estudios relevantes en el campo, destacando investigaciones previas relacionadas con la evaluación de la interpretabilidad de modelos de IA, los desafíos asociados y los enfoques utilizados en estudios similares. Se discute cómo el presente trabajo se sitúa en el contexto de la literatura existente.
- Capítulo 4: Metodología. Se describe el diseño del cuestionario, la selección del conjunto de datos (rendimiento académico de los estudiantes en matemáticas), y el proceso de desarrollo de las herramientas utilizadas. También se incluyen los métodos de análisis de datos empleados para evaluar la interpretabilidad de los modelos.
- Capítulo 5: Resultados. En este capítulo se presentan los resultados obtenidos del cuestionario, incluyendo el análisis de las respuestas recopiladas, la interpretación de los resultados y los hallazgos significativos relacionados con la hipótesis.
- Capítulo 6: Discusión. Se interpretan los resultados en el contexto de los objetivos e hipótesis planteados al inicio. Se discute la importancia de los hallazgos, se exploran las limitaciones del estudio y se sugieren posibles mejoras o direcciones para futuras investigaciones.
- Capítulo 7: Conclusiones. Este capítulo ofrece un resumen de las principales conclusiones del estudio, destacando las contribuciones del TFM al campo de la inteligencia artificial explicable y la evaluación de modelos interpretables. También se incluyen recomendaciones para futuras investigaciones.

Capítulo 2

Fundamentos teóricos

Este capítulo presenta los conceptos esenciales sobre la interpretabilidad en inteligencia artificial y su relevancia en aplicaciones críticas donde la transparencia es fundamental. Se describen los enfoques intrínsecos para mejorar la interpretabilidad, como los árboles de decisión (DT) y los Interpretable Decision Sets (IDS), junto con técnicas como la sparsidad, simulabilidad, modularidad y parsimonia. También se exploran métodos basados en reglas y se discuten los factores que influyen en la percepción de interpretabilidad y la confianza del usuario, proporcionando el marco teórico necesario para este TFM.

2.1. Introducción a la Interpretabilidad

La interpretabilidad en el campo de la inteligencia artificial (IA) se define como la capacidad de explicar o presentar los resultados de un sistema de aprendizaje automático de manera comprensible para los seres humanos. Es fundamental para garantizar que se cumplan criterios como la equidad, la privacidad, la fiabilidad, la robustez, la causalidad, la usabilidad y la confianza. De este modo, permite a los usuarios entender cómo funcionan los algoritmos y verificar si estos cumplen con los objetivos esperados [9, 10].

A diferencia de la explicabilidad, que se enfoca en proporcionar explicaciones posteriores al desarrollo de los modelos (post-hoc), la interpretabilidad basada en modelos implica un diseño intencional desde la etapa de modelado para que sean comprensibles para los usuarios. Esto puede lograrse mediante la construcción de modelos intrínsecamente interpretables, como los árboles de decisión, que ofrecen claridad sobre las relaciones aprendidas a partir de los datos. Sin embargo, este enfoque presenta el reto de equilibrar la simplicidad del modelo con su precisión predictiva, además de lidiar con posibles sesgos en los datos o en los métodos explicativos utilizados [11, 12].

Existen dos enfoques principales para abordar la interpretabilidad:

- *Interpretabilidad local:* Se centra en explicar la predicción del modelo para una instancia específica. Responde preguntas como: "¿Por qué se tomó esta decisión para este dato?". Técnicas como LIME (Local Interpretable Model-agnostic Explanations) muestran qué características del dato son más relevantes para una

predicción individual [9, 13]. Es útil en situaciones donde se necesita entender decisiones individuales, como en diagnósticos médicos o aprobaciones de crédito.

- *Interpretabilidad global:* Busca comprender el comportamiento general del modelo en todo el conjunto de datos. Responde a preguntas como: "¿Cómo toma decisiones el modelo en general?.º "¿Qué patrones ha aprendido el modelo?". Métodos como los árboles de decisión y los modelos aditivos generalizados (GAMs) ofrecen una visión global, permitiendo observar cómo las predicciones varían según las características de los datos [9, 12]. Es crucial en aplicaciones que requieren entender las reglas generales del modelo, como en políticas públicas o investigaciones científicas.

Ambos enfoques son complementarios y se aplican en diferentes contextos. Mientras que la *interpretabilidad local* facilita la comprensión de decisiones individuales, la *interpretabilidad global* proporciona una visión más amplia del funcionamiento del modelo, lo cual es esencial para su validación y aceptación en aplicaciones críticas [13, 9, 12].

En este TFM, la interpretabilidad se analizará en el contexto de modelos de decisión interpretables, específicamente en los *Interpretable Decision Sets* (IDS) y los árboles de decisión (DT). Se explorará cómo la estructura del modelo, la presentación de resultados y otros factores influyen en la percepción de interpretabilidad y en la capacidad de los usuarios para detectar errores.

2.2. Criterios de interpretabilidad

Esta sección se basa en los trabajos de Murdoch et al. (2019) [11] y Rudin (2019) [14], quienes destacan, desde diferentes perspectivas, la importancia de utilizar modelos de decisión que sean inherentemente interpretables en aplicaciones donde la transparencia es crucial, como la medicina, la justicia o la toma de decisiones críticas en entornos regulatorios.

Los modelos de caja negra, utilizados en decisiones de alto riesgo, pueden tener consecuencias negativas significativas debido a la falta de transparencia y a la dificultad para interpretar sus resultados. Por ello, ambos autores recomiendan el uso de modelos interpretables que proporcionen explicaciones fieles y comprensibles, en lugar de depender de explicaciones post hoc.

Según estos estudios, es más efectivo diseñar modelos que sean interpretables desde el principio, ya que esto evita las complicaciones asociadas con la explicación de modelos complejos. Para alcanzar esta interpretabilidad, se pueden utilizar varios enfoques intrínsecos, tales como la sparsidad, la simulabilidad, y la modularidad. A continuación, se exploran estos tres enfoques y sus beneficios en el diseño de modelos interpretables.

2.2.1. Sparsidad

La sparsidad implica reducir el número de parámetros no nulos en un modelo, lo que facilita la comprensión del papel específico de cada variable en las predicciones. Este enfoque es particularmente útil en casos donde se sabe que la relación sub-

yacente depende de un conjunto limitado de señales significativas. Por ejemplo, los modelos lineales, como la regresión LASSO (Least Absolute Shrinkage and Selection Operator), o métodos como el sparse coding, aplican técnicas de penalización para mantener la simplicidad del modelo sin comprometer significativamente la precisión predictiva [15]. Esta reducción en complejidad no solo mejora la interpretabilidad, sino que también puede ayudar a evitar el sobreajuste al destacar características verdaderamente relevantes.

La regresión LASSO se define mediante la siguiente ecuación [15]:

$$\min_{\theta} \left(\frac{1}{2} \|Z - \Phi\theta\|_2^2 + \lambda \|\theta\|_1 \right), \tag{2.1}$$

donde Z es el vector de las salidas observadas, Φ es la matriz de regresores (características), θ es el vector de parámetros desconocidos del modelo, $\|\cdot\|_2$ denota la norma L_2, y $\|\cdot\|_1$ denota la norma L_1.

La sparsidad se promueve a través del término de penalización $\lambda\|\theta\|_1$, que reduce algunos de los coeficientes θ a cero. Este enfoque favorece modelos más simples y parsimoniosos, mejorando la interpretabilidad al centrarse únicamente en las características más relevantes.

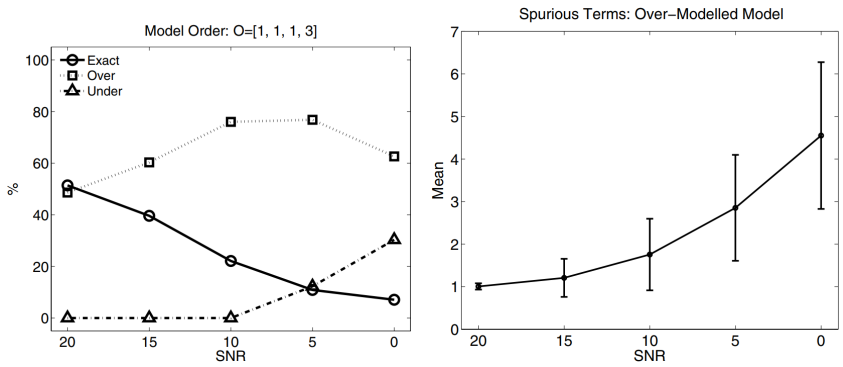


Figura 2.1: Relación entre sparsidad y selección de modelos utilizando LASSO. El gráfico de la izquierda muestra la tasa de selección de modelos exactos (círculos), sobre-modelados (cuadrados) y sub-modelados (triángulos) en función del nivel de ruido en la señal (SNR). Un modelo exacto tiene solo las variables relevantes, mientras que un sobre-modelado incluye variables adicionales innecesarias (términos espurios) y un sub-modelado omite variables relevantes. El gráfico de la derecha muestra la media y desviación estándar del número de términos espurios seleccionados en modelos sobre-modelados, destacando cómo LASSO minimiza la inclusión de estas variables irrelevantes, promoviendo así la sparsidad y la interpretabilidad del modelo. Adaptado de [15].

2.2.2. Simulabilidad

Por otro lado, la simulabilidad se refiere a la capacidad de un modelo para ser reproducido y comprendido fácilmente por un ser humano. Modelos como los árboles de decisión y las listas de reglas "si-entonces" ejemplifican esta característica, ya que permiten seguir paso a paso el proceso de toma de decisiones del modelo. Este

enfoque es especialmente valioso en contextos donde las decisiones deben ser comprensibles para personas no expertas, como en aplicaciones médicas, donde tanto los pacientes como los profesionales de la salud necesitan entender las recomendaciones del modelo.

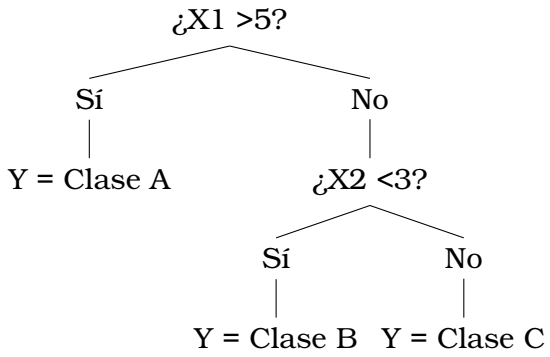


Figura 2.2: Árbol de decisión basado en [11] que ejemplifica la simulabilidad al dividir el espacio de características en regiones de decisión claras mediante reglas "si-entonces". Cada nodo representa una pregunta sobre las características (X_1 y X_2), y las ramas llevan a decisiones simples ("Sí.º "No"), facilitando que cualquier usuario pueda seguir y reproducir el proceso de toma de decisiones del modelo.

2.2.3. Modularidad

La modularidad es un enfoque clave en modelos interpretables, ya que permite descomponer el modelo en partes significativas que se pueden interpretar de forma independiente. Este enfoque es particularmente útil en modelos complejos, donde los subcomponentes o módulos tienen un significado interpretativo claro.

Por ejemplo, los Modelos Aditivos Generalizados (GAMs)[16], como el Explainable Boosting Machine (EBM) [11], restringen las relaciones entre las variables a una forma aditiva, lo que facilita la interpretación de cada término individual. Esto significa que cada función en el modelo representa el efecto de un predictor específico en la predicción final, y estos efectos se pueden analizar de manera aislada para comprender su contribución individual.

La ecuación de un GAM se puede expresar como:

$$g(\mu) = \beta + f_1(x_1) + \cdots + f_m(x_m) \tag{2.2}$$

donde $g(\mu)$ es la función de enlace, β es el término independiente, y $f_i(x_i)$ son funciones suaves que representan el efecto de cada predictor x_i.

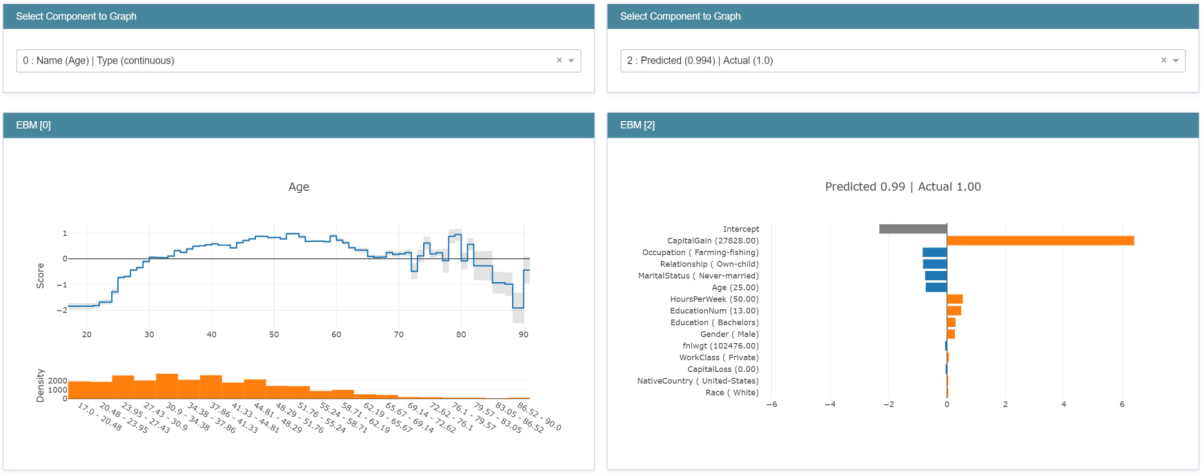


Figura 2.3: Ejemplo de modularidad en un modelo aditivo como el EBM. Izquierda: La función f_{Age} muestra cómo la característica 'Edad' afecta la predicción final. Derecha: Descomposición de una predicción individual, donde se destaca cómo cada característica contribuye de manera independiente, con 'CapitalGain' dominando la predicción. Adaptado de [11].

2.2.4. Parsimonía

La parsimonia se refiere a la simplicidad de un modelo al utilizar el mínimo número de parámetros o términos necesarios para capturar la dinámica de un sistema. Esto facilita la interpretabilidad y la generalización del modelo, ya que reduce la complejidad y ayuda a evitar sobreajustes. Según Kutz y Brunton (2022), promover la parsimonia en el aprendizaje automático, especialmente en modelos informados por la física, resulta en modelos más interpretables y físicamente coherentes, permitiendo una mejor generalización a nuevos escenarios.

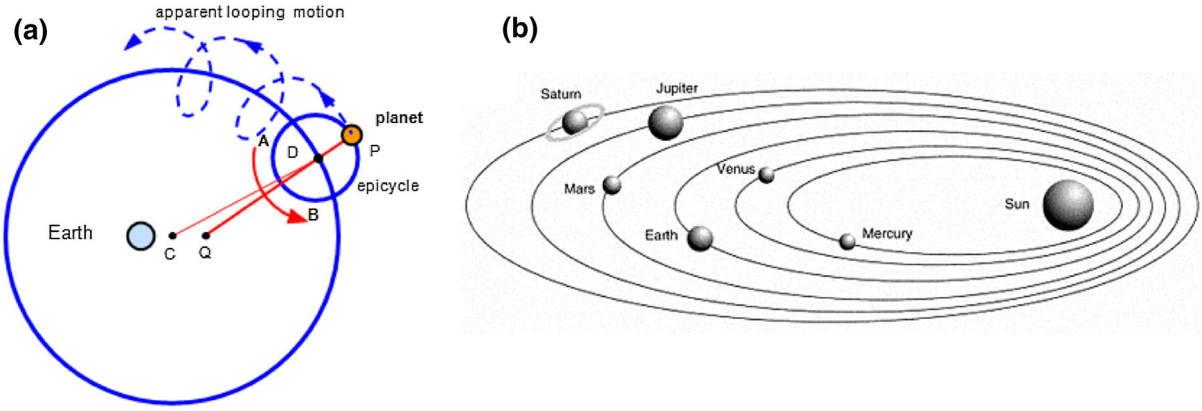


Figura 2.4: Comparación entre dos modelos astronómicos que ilustran el principio de parsimonía: (a) El modelo ptolemaico describe el movimiento planetario con epiciclos y deferentes, donde el ángulo azimutal del planeta se calcula mediante $\alpha(t) = \Omega t - \sin^{-1}\left(\frac{CE}{R}\sin(\Omega t)\right)$ [17], lo que implica una mayor complejidad debido a la necesidad de múltiples parámetros; (b) El modelo heliocéntrico de Copérnico, refinado por Newton, utiliza una ley de gravitación universal más simple, $F = G\frac{m_1 m_2}{r^2}$, que requiere menos supuestos y es más parsimonioso. Imagen adaptada de [18].

2.2.5. Subgrupos: Métodos Contrastivos y Emergentes

Además de los enfoques intrínsecos como la sparsidad, la simulabilidad, la modularidad y la parsimonia, los métodos basados en reglas también emplean técnicas específicas para mejorar la interpretabilidad mediante la identificación de patrones diferenciados en los datos. Un enfoque destacado es el *descubrimiento de subgrupos*, que busca encontrar segmentos de datos que sean relevantes o interesantes con respecto a una variable de interés.

A diferencia de métodos como las Máquinas de Soporte Vectorial o Redes Neuronales, que se enfocan en maximizar la precisión de la clasificación de ejemplos individuales, los métodos basados en reglas, como el descubrimiento de subgrupos, se centran en caracterizar las clases a través de sus relaciones con otras entidades presentes en los datos [19]. Estos métodos no solo se preocupan por la precisión de la predicción, sino también por la claridad y comprensibilidad de las relaciones aprendidas, lo cual es esencial para la interpretabilidad [20].

El *descubrimiento de subgrupos*, un concepto ampliamente utilizado en minería de datos, consiste en identificar segmentos de la población que sean estadísticamente relevantes según ciertos criterios. Estos segmentos se definen mediante reglas del tipo “si-condición(es)-entonces-clase”, que permiten describir de forma comprensible cómo se agrupan los datos respecto a una propiedad específica [21, 20]. Este paradigma incluye tres enfoques principales:

1. *Descubrimiento de subgrupos*: Este método busca identificar subgrupos dentro del conjunto de datos que tengan una alta probabilidad de cumplir con una determinada característica o condición de interés. Por ejemplo, en un estudio médico, puede identificar pacientes con síntomas similares que tienen una alta probabilidad de padecer una enfermedad específica.

2. *Conjunto de contraste*: Este enfoque se centra en encontrar pares de atributos y valores que sean únicos para cada grupo o clase. Su objetivo es maximizar la diferenciación entre clases, destacando atributos específicos que caracterizan exclusivamente a cada clase. Por ejemplo, en los árboles de decisión, se busca que cada nodo defina claramente las fronteras entre las diferentes clases (ver Figura 2.2).

3. *Patrón emergente*: Se enfoca en extraer subgrupos en los que las frecuencias relativas de la variable objetivo varían de manera significativa entre los distintos grupos. Utiliza listas de decisión y métodos centrados en la çobertura"de reglas para identificar los atributos que son más representativos o influyentes para una clase específica. Este enfoque es útil para descubrir reglas o patrones que puedan ser menos evidentes pero importantes para la comprensión del modelo (ver Figura 2.3).

En general, estos algoritmos utilizan heurísticas para aproximar un conjunto óptimo de reglas, lo que significa que introducen ciertos supuestos sobre la estructura de los datos para hacer el problema manejable en tiempo polinomial. Estas técnicas ayudan a construir modelos que no solo sean precisos, sino también comprensibles para los usuarios finales, permitiendo una mejor toma de decisiones basada en las interpretaciones claras proporcionadas por las reglas generadas.

2.2.6. Contexto y Audiencia

Finalmente, la elección del enfoque interpretativo depende en gran medida del contexto del problema y de la audiencia objetivo. En aplicaciones donde la precisión predictiva es menos crítica que la interpretabilidad, como en auditorías de modelos para asegurar la equidad, se prefieren modelos más simples y transparentes. En cambio, en situaciones que exigen alta precisión, se pueden considerar métodos post hoc para interpretar modelos más complejos, asegurando así un equilibrio adecuado entre interpretabilidad y precisión.

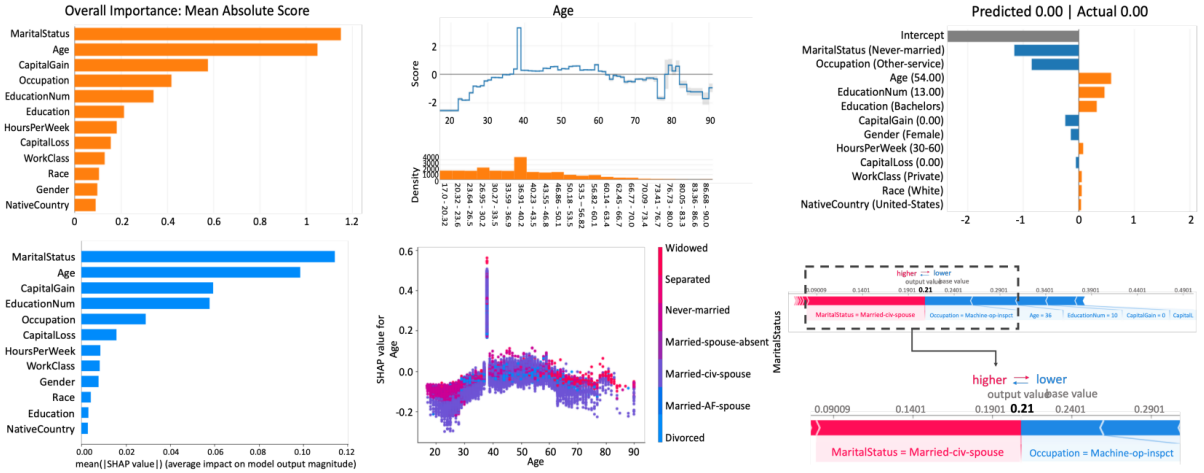


Figura 2.5: Las visualizaciones de GAM (arriba) y SHAP (abajo), adaptadas de [12] y generadas por InterpretML [22], muestran cómo diferentes enfoques interpretativos se ajustan a diversas audiencias y contextos. Las explicaciones globales (columna izquierda) ayudan a científicos de datos a identificar las variables más influyentes en el modelo; los gráficos de componentes o de dependencia (columna central) son útiles para que analistas de negocio interpreten el impacto de factores específicos, como edad o ingreso, en la puntuación crediticia; y las explicaciones locales (columna derecha) son cruciales en medicina personalizada para justificar decisiones como la recomendación de un tratamiento específico.

2.3. Modelos de Decisión Interpretables

A continuación, se presentan los tres modelos interpretables con los que evaluaremos la interpretabilidad en este TFM.

2.3.1. Árboles de Decisión (DT)

Esta sección se basa en los trabajos de Mienye et al. (2024) [23] y Duda et al. (2000) [19], quienes proporcionan un análisis exhaustivo de los conceptos, algoritmos y aplicaciones de los Árboles de Decisión (DT). Los DT son modelos de aprendizaje supervisado utilizados tanto para tareas de clasificación como de regresión. Un árbol de decisión se construye dividiendo iterativamente un conjunto de datos en subconjuntos más pequeños basados en reglas de decisión derivadas de las características de los datos. Cada nodo interno del árbol representa una característica o atributo, mientras que cada rama representa una regla de decisión y cada hoja final una decisión o resultado.

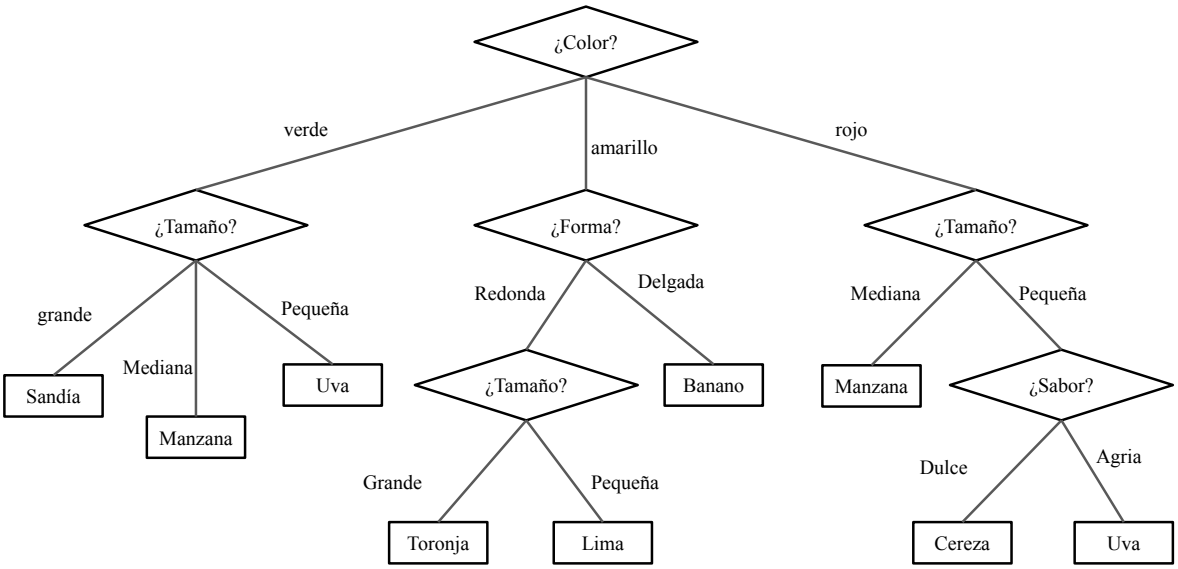


Figura 2.6: Ilustración de un árbol de decisión basada en técnicas de clasificación no paramétricas. Adaptada de [19], Capítulo 4. Los árboles de decisión son fáciles de interpretar cuando son pequeños, pero pueden volverse complejos y difíciles de validar a medida que crecen [24].

si color = verde **y** tamaño = grande **entonces** Sandía
si color = verde **y** tamaño = mediana **entonces** Manzana
si color = verde **y** tamaño = pequeña **entonces** Uva
si color = amarillo **y** forma = redonda **y** tamaño = grande **entonces** Toronja
si color = amarillo **y** forma = redonda **y** tamaño = pequeña **entonces** Lima
si color = amarillo **y** forma = delgada **entonces** Banano
si color = rojo **y** tamaño = mediana **entonces** Manzana
si color = rojo **y** tamaño = pequeña **y** sabor = dulce **entonces** Cereza
si color = rojo **y** tamaño = pequeña **y** sabor = agrio **entonces** Uva

Figura 2.7: Reglas de decisión basadas en el árbol de decisión de la Figura 2.6.

El entrenamiento de un árbol es un proceso recursivo que comienza con todos los datos en la raíz y divide sucesivamente los nodos según la mejor característica, repitiendo el proceso en cada nodo hijo [19]. Uno de los algoritmos más utilizados para construir árboles de decisión es el ID3, que selecciona la característica que proporciona la mayor ganancia de información. A continuación se presenta el algoritmo:

Algorithm 1: Algoritmo ID3 para Árbol de Decisión

Input: Conjunto de datos de entrenamiento $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$
Output: Árbol de decisión T

Función `ID3(`D`)`**:**
  if *D está vacío* **then**
    return *un nodo terminal con clase por defecto* $c_{default}$
  if *todas las instancias en D tienen la misma etiqueta de clase y* **then**
    return *un nodo terminal con clase y*
  if *el conjunto de atributos J está vacío* **then**
    return *un nodo terminal con la clase más frecuente en D*
  Seleccionar el atributo f que mejor divide los datos usando ganancia de información;
  Crear un nodo de decisión para f;
  for *cada valor posible b_i de f* **do**
    Crear una rama para b_i;
    Sea D_i el subconjunto de D donde $x_i = b_i$;
    Recursivamente construir el subárbol para D_i;
    Adjuntar el subárbol a la rama para b_i;
  return *el nodo de decisión*

Uno de los criterios más comunes para dividir los nodos es la ganancia de información, definida como:

$$\text{Ganancia de Información} = I(p) - \sum_i \frac{|p_i|}{|p|} I(p_i), \tag{2.3}$$

donde $I(p)$ es la entropía del conjunto de datos original, y $I(p_i)$ es la entropía del subconjunto resultante después de la división.

Otro criterio utilizado es el índice de Gini:

$$\text{Índice de Gini} = 1 - \sum_i p_i^2, \tag{2.4}$$

donde p_i representa la proporción de observaciones de la clase i en el nodo.

2.3.2. Listas de decisión

Las listas de decisión son una representación para funciones Booleanas construidas como secuencias ordenadas de predicados lógicos del tipo "si [condición(es)] entonces [clase]" seguida de otras condiciones adicionales (e.g., "más si [condición(es)] entonces [clase]"). Las condiciones corresponden a predicados sobre las características del problema de clasificación (los *features*) [24, 25, 26].

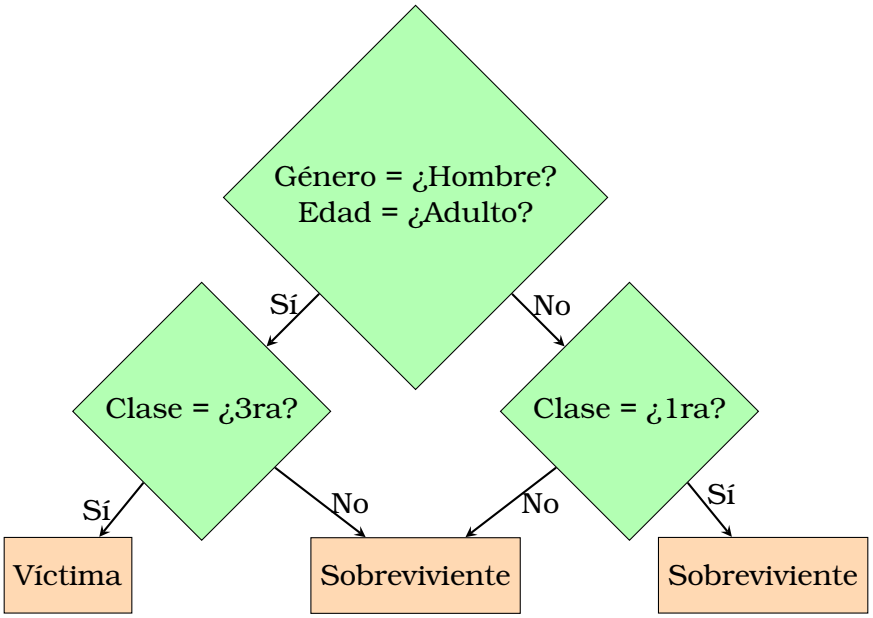


Figura 2.8: Ejemplo de lista de decisión. Adaptado de [25]. Se muestra entrenado con el conjunto de datos del Titanic

si género = hombre **y** edad = adulto **entonces** víctima
si no, si clase = 3ra **entonces** víctima
si no, si clase = 1ra **entonces** sobreviviente
si no, sobreviviente

Figura 2.9: Reglas de decisión basadas en el árbol de decisión de la Figura 2.6.

A diferencia de otros métodos de aprendizaje automático como Máquinas de Soporte Vectorial, Bosques Aleatorios o Redes Neuronales, las listas de decisión ofrecen una explicación natural y sencilla para cada predicción, permitiendo que un experto entienda el mecanismo de decisión y razone sobre el resultado. Esto se considera una ventaja significativa de las listas de decisión [25].

Además, las listas de decisión también ofrecen ventajas durante el proceso de entrenamiento cuando se comparan con otros métodos interpretables, como los árboles de decisión. Mientras que estos métodos utilizan algoritmos avaros, lo cual permite tiempos de entrenamiento cortos pero a costa de una reducción en el desempeño, las listas de decisión construyen modelos secuenciales más robustos, manteniendo una mayor capacidad explicativa [24].

Gracias a su naturaleza "emergente," las listas de decisión se construyen mediante reglas que particionan el espacio de características de manera más significativa. Por ejemplo, una lista de decisión de tamaño k (k-DL) utiliza reglas en forma de cláusulas conjuntas, donde cada regla r_i puede representarse matemáticamente como:

$$r_i : \bigwedge_{j=1}^{m_i} (x_j = a_{ij}) \rightarrow y = c_i,$$

donde x_j son los atributos, a_{ij} son los valores específicos de dichos atributos, y c_i

es la clase asignada. Este enfoque garantiza la tratabilidad computacional del problema, explorando un mayor número de combinaciones posibles sin comprometer la eficiencia [25].

En particular, basándonos en el trabajo de Rivest (1987), las listas de decisión son polinómicamente aprendibles. Esto significa que pueden ser aprendidas de manera eficiente usando algoritmos de aprendizaje, como se define en el marco teórico de Valiant (1984). Rivest demuestra que las listas de decisión de tamaño k (k-DL) pueden ser identificadas en tiempo polinómico mediante un algoritmo codicioso que selecciona en cada paso la regla que maximiza la ganancia de información:

$$\Delta I = I(S) - I(S|A),$$

donde $I(S)$ es la entropía del conjunto de datos S y $I(S|A)$ es la entropía después de dividir S usando el atributo A que mejor explica los datos restantes [26].

El algoritmo encuentra iterativamente reglas que cubren el mayor número de ejemplos de una misma clase, organizándolas en una lista secuencial. Cada regla se aplica en orden hasta que todos los ejemplos han sido clasificados.

Algorithm 2: Algoritmo para Listas de Decisión

Input: Conjunto de datos de entrenamiento $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$
Output: Lista de decisión L

Función `DecisionList(`D`)`**:**
    $L \leftarrow \emptyset$;
    while D *no está vacío* **do**
        Encontrar la regla $r = (c, v)$ que cubra más ejemplos en D;
        $L \leftarrow L \cup \{r\}$;
        Eliminar de D los ejemplos cubiertos por r;
    return L

2.3.3. Conjuntos Interpretables de Decisión (IDS)

Esta sección se basa en el trabajo de Lakkaraju et al. (2016) [1], quienes proponen un marco para construir modelos predictivos que son altamente precisos y al mismo tiempo altamente interpretables. Un conjunto de decisión es un conjunto de predicados lógicos asociados a una etiqueta de clase. Las variables de los predicados corresponden a los atributos del problema de clasificación (los *features*). El predicado sigue la forma "si [condición(es)] entonces [clase]". Cada predicado está compuesto por ítems (predicados más pequeños) unidos por conjunción. Se dice que el conjunto asigna una clase a un ejemplo si el predicado es verdadero. En caso de empates (varios predicados son verdaderos, pero asignan una clase diferente), se utiliza alguna regla de desempate, generalmente *ad-hoc* para cada aplicación, como dar prioridad a la regla más específica o a la primera regla que se cumpla. La figura 2.10 muestra un ejemplo de un conjunto de decisión, adaptado de este trabajo.

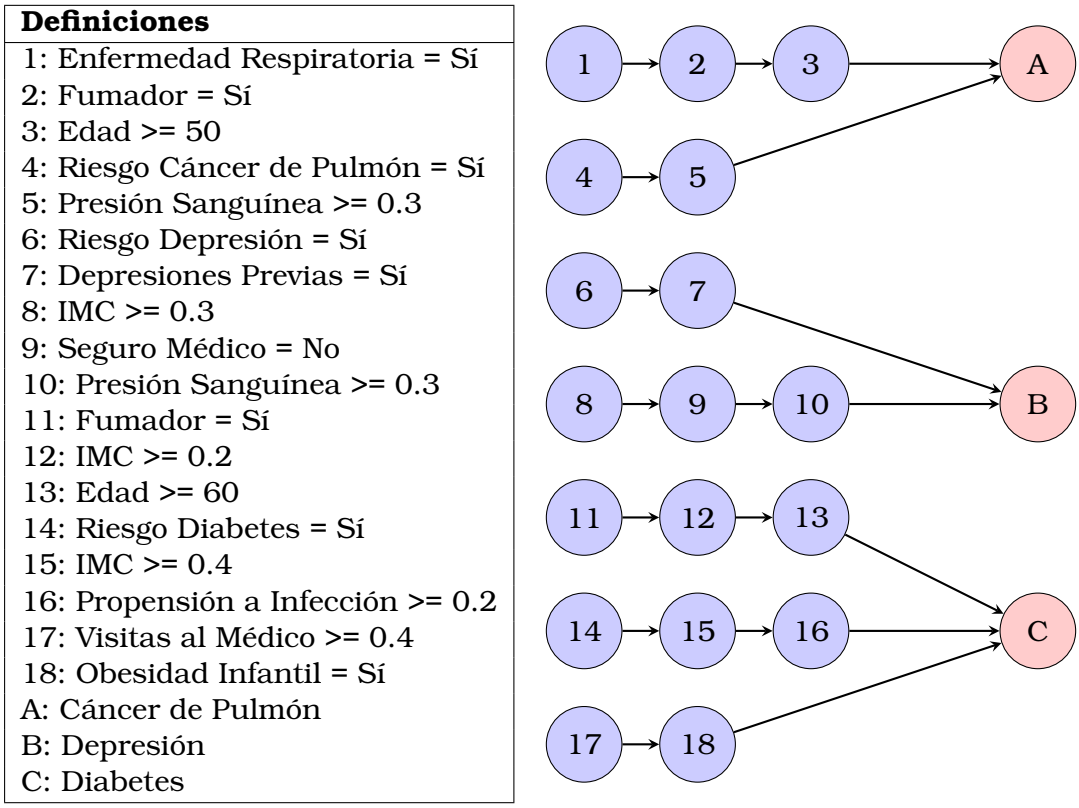


Figura 2.10: Ejemplo de un conjunto interpretable de decisión. Adaptado de [1].

si enfermedad respiratoria = sí **y** fumador = sí **y** edad $\geq$ 50 **entonces** cáncer de pulmón
si riesgo cáncer de pulmón = sí **y** presión sanguínea $\geq$ 0.3 **entonces** cáncer de pulmón
si riesgo depresión = sí **y** depresiones previas = sí **entonces** depresión
si IMC $\geq$ 0.3 **y** seguro médico = no **y** presión sanguínea $\geq$ 0.3 **entonces** depresión
si fumador = sí **y** IMC $\geq$ 0.2 **y** edad $\geq$ 60 **entonces** diabetes
si riesgo diabetes = sí **y** IMC $\geq$ 0.4 **y** propensión a infección $\geq$ 0.2 **entonces** diabetes
si visitas al médico $\geq$ 0.4 **y** obesidad infantil = sí **entonces** diabetes

Figura 2.11: Reglas de decisión basadas en el conjunto interpretable de decisión de la Figura 2.10.

Al igual que las listas de decisión, los conjuntos de decisión ofrecen ventajas sobre otros métodos de aprendizaje automático debido a su sencillez de inferencia e interpretabilidad [27, 24]. Como se mencionó anteriormente, la interpretabilidad se refiere a la facilidad con que un experto humano puede entender y razonar sobre el mecanismo utilizado por el modelo para asignar una clase [14]. Sin embargo, a diferencia de las listas de decisión, que por su naturaleza anidada pueden requerir razonamiento sobre múltiples características y recordar decisiones previas, los conjuntos de decisión solo requieren evaluar predicados simples de forma independiente, lo que los hace aún más interpretables.

Dicha "facilidad de interpretación" es difícil de cuantificar. Aunque la estructura de los conjuntos de decisión se considera interpretable por construcción, es posible construir conjuntos tan grandes y con predicados tan largos que se vuelven difíciles de manejar para un humano, de forma similar a lo que sucede con árboles de

decisión muy grandes. Para abordar este problema, Lakkaraju et al. proponen un marco para construir modelos predictivos que optimizan tanto la precisión como la interpretabilidad. Este marco define una serie de métricas que cuantifican propiedades inherentes a cualquier conjunto de decisión interpretable, tales como:

- *Tamaño del conjunto:* Esta métrica se refiere al número total de reglas en el conjunto de decisión, denotado como $|R|$. Un conjunto más pequeño ($|R|$) tiende a ser más fácil de interpretar, ya que requiere evaluar menos reglas para comprender el modelo completo.
- *Longitud de los predicados:* Mide el número de condiciones dentro de cada regla ïf-then". Si consideramos una regla $r \in R$, la longitud de r, denotada como len(r), se define como el número de condiciones lógicas en esa regla. Predicados más cortos (len(r) más bajos) simplifican el proceso de interpretación, ya que reducen la complejidad cognitiva al momento de entender cómo se toma una decisión.
- *Superposición entre reglas:* Evalúa la cantidad de solapamiento entre reglas. Sea R_i y R_j dos reglas distintas en el conjunto R. La superposición entre estas reglas puede definirse como:

$$\text{Overlap}(R_i, R_j) = \frac{|R_i \cap R_j|}{\text{mín}(|R_i|, |R_j|)}, \tag{2.5}$$

  donde $|R_i|$ y $|R_j|$ representan el número de instancias cubiertas por las reglas R_i y R_j, respectivamente. Minimizar la superposición es crucial para asegurar que las reglas no entren en conflicto y que cada regla cubra una parte distinta del espacio de datos, facilitando la claridad y precisión de la interpretación.

Estas métricas son integradas en una 'función objetivo' dentro del algoritmo de construcción de IDS. El objetivo del algoritmo no es solo maximizar la precisión predictiva, sino también encontrar un equilibrio que minimice la complejidad del modelo, resultando en conjuntos de decisión más simples y comprensibles. La función objetivo pondera cada una de estas métricas para guiar el proceso de selección de reglas durante el entrenamiento del modelo. Así, se priorizan reglas que ofrecen un buen rendimiento predictivo, pero que al mismo tiempo mantienen la simplicidad y claridad necesarias para ser interpretadas fácilmente por expertos humanos.

Este enfoque garantiza que el modelo resultante no solo sea preciso, sino también interpretable, haciendo que las decisiones derivadas del modelo sean más transparentes y confiables.

El algoritmo de construcción de IDS, denominado *Smooth Local Search* (SLS), optimiza la precisión del modelo y la interpretabilidad de las reglas generadas al buscar el mejor conjunto de reglas bajo ciertas restricciones. El algoritmo funciona de la siguiente manera: minimiza el solapamiento entre las reglas y asegura que cada regla cubra la mayor cantidad de datos posible sin aumentar significativamente la complejidad del modelo. A diferencia de otros métodos de búsqueda, SLS explora de manera eficiente el espacio de soluciones utilizando una búsqueda local suave que evita quedarse atrapado en óptimos locales al suavizar la función objetivo.

El procedimiento del algoritmo SLS se puede describir en los siguientes pasos:

- *Inicialización:* Se inicia con un conjunto vacío de reglas, $R = \emptyset$.

- *Generación de Candidatos:* Se generan candidatos de reglas para cada clase utilizando el conjunto de datos de entrenamiento D. Estas reglas son evaluadas en términos de precisión y cobertura, definidos como:

$$\text{Precisión}(r) = \frac{\text{Número de ejemplos correctamente clasificados por la regla } r}{\text{Número total de ejemplos cubiertos por la regla } r} \tag{2.6}$$

$$\text{Cobertura}(r) = \frac{\text{Número de ejemplos de la clase que cubre la regla } r}{\text{Número total de ejemplos de esa clase en el conjunto de datos } D} \tag{2.7}$$

- *Evaluación y Selección de Reglas:* Se selecciona la regla candidata que maximiza la función objetivo:

$$\text{Función Objetivo}(r) = \alpha \cdot \text{Precisión}(r) + \beta \cdot \text{Cobertura}(r) - \gamma \cdot \text{Solapamiento}(r, R), \tag{2.8}$$

  donde α, β, γ son coeficientes de ponderación que ajustan la importancia relativa de cada término.

- *Verificación de Interpretabilidad:* Antes de añadir una regla al conjunto R, se verifica si cumple con el umbral de interpretabilidad definido por el parámetro θ:

$$\text{Complejidad}(r) \leq \theta. \tag{2.9}$$

- *Optimización de la Complejidad del Modelo:* Después de añadir todas las reglas que cumplen con los criterios anteriores, el conjunto de reglas R es ordenado para optimizar su aplicabilidad y minimizar la complejidad del modelo.

El uso del algoritmo SLS permite construir un conjunto de reglas interpretables de alta calidad que no solo logra una precisión predictiva comparable a otros modelos más complejos, sino que también facilita la comprensión por parte de usuarios no expertos. A continuación, se presenta el algoritmo en pseudocódigo:

Algorithm 3: Algoritmo SLS para Conjuntos Interpretables de Decisión

Input : Conjunto de datos de entrenamiento D, umbral de interpretabilidad θ
Output: Conjunto de reglas R

Función `SLS` (D, θ)**:**
  Inicializar el conjunto de reglas $R \leftarrow \emptyset$;
  while *exista alguna clase sin cubrir en* D **do**
    Generar candidatos de reglas para cada clase utilizando el conjunto de datos D;
    Evaluar cada regla candidata basada en precisión y cobertura;
    Seleccionar la regla que maximiza la precisión y cobertura y minimiza el solapamiento con reglas existentes en R;
    if *la regla cumple con el umbral de interpretabilidad* θ **then**
      Añadir la regla seleccionada a R;
    end
  end
  Ordenar el conjunto de reglas R para optimizar la aplicabilidad y minimizar la complejidad del modelo;
  return R

2.4. Factores que Afectan la Interpretabilidad

La interpretabilidad de un modelo de aprendizaje automático depende de dos factores principales: la transparencia del modelo y la presentación de los resultados. La transparencia implica la facilidad con la que los usuarios pueden entender los mecanismos internos del modelo, que van desde la capacidad de simular mentalmente sus operaciones hasta comprender completamente los algoritmos subyacentes [14]. Por otra parte, las visualizaciones desempeñan un papel crucial en cómo los usuarios perciben la precisión y la confiabilidad del modelo. No obstante, si estas visualizaciones no se diseñan adecuadamente, pueden inducir una confianza excesiva o una comprensión superficial de los resultados [3].

2.4.1. Transparencia del Modelo

La transparencia en los modelos de aprendizaje automático se refiere a la facilidad con la que se pueden entender los mecanismos internos del modelo y cómo estos conducen a una predicción específica. Un modelo se considera transparente cuando su funcionamiento puede ser descrito y comprendido por los usuarios sin necesidad de conocimientos técnicos avanzados [14].

Existen diferentes niveles de transparencia que influyen en la interpretabilidad de un modelo:

- *Transparencia simulable:* Implica que una persona pueda simular mentalmente las operaciones del modelo. Por ejemplo, los árboles de decisión pequeños o los conjuntos de reglas simples, como los *Interpretable Decision Sets* (IDS), permiten seguir cada paso de su cálculo en un tiempo razonable [12]. (Véase Figura 2.6).
- *Transparencia descriptiva:* Se refiere a la capacidad de explicar cómo el modelo toma decisiones en términos comprensibles para el usuario, descomponiendo

el modelo en reglas más pequeñas y manejables. Esto es particularmente útil en los IDS, donde las reglas se presentan de manera independiente y clara [1]. (Véase Figura 2.10).

- *Transparencia algorítmica:* Se centra en la comprensión de los algoritmos que rigen el comportamiento del modelo y cómo estos afectan sus resultados. Modelos como los árboles de decisión (DT), las listas de decisiones (DL), y los IDS son considerados algorítmicamente transparentes porque se basan en estructuras claramente definidas [28]. (Véase Figura 2.8).

La falta de transparencia puede generar problemas significativos en la interpretabilidad. Los modelos de caja negra, como las redes neuronales profundas, suelen ser complejos y opacos, dificultando su comprensión y aceptación en aplicaciones críticas [3]. En contraste, los modelos más simples y explicables, como los árboles de decisión y los IDS, proporcionan una mayor transparencia, lo que es esencial para garantizar decisiones confiables y justificables.

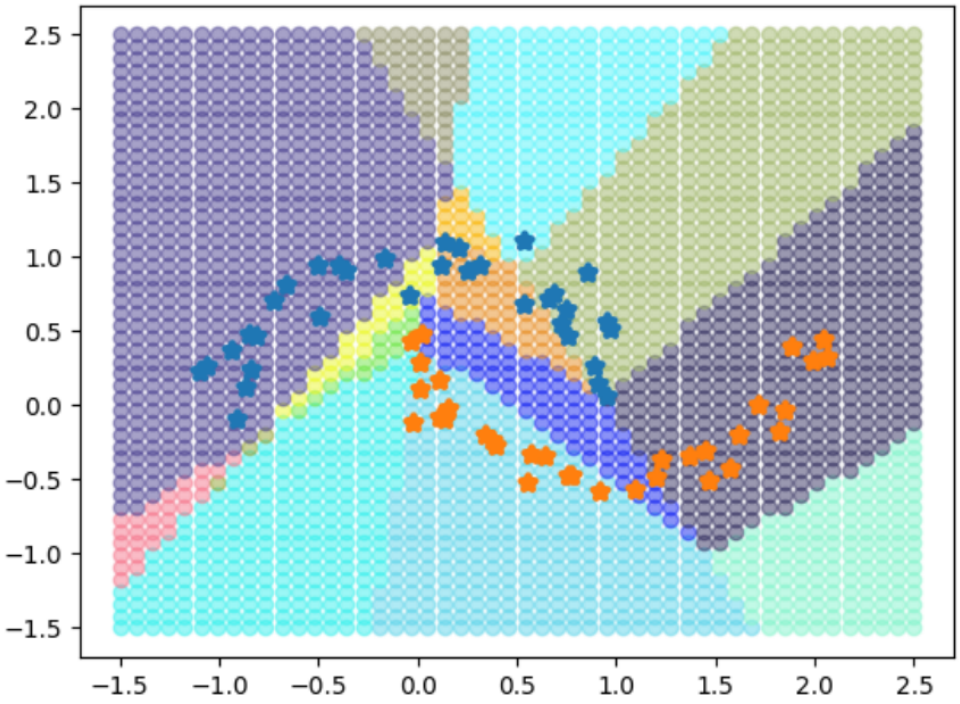


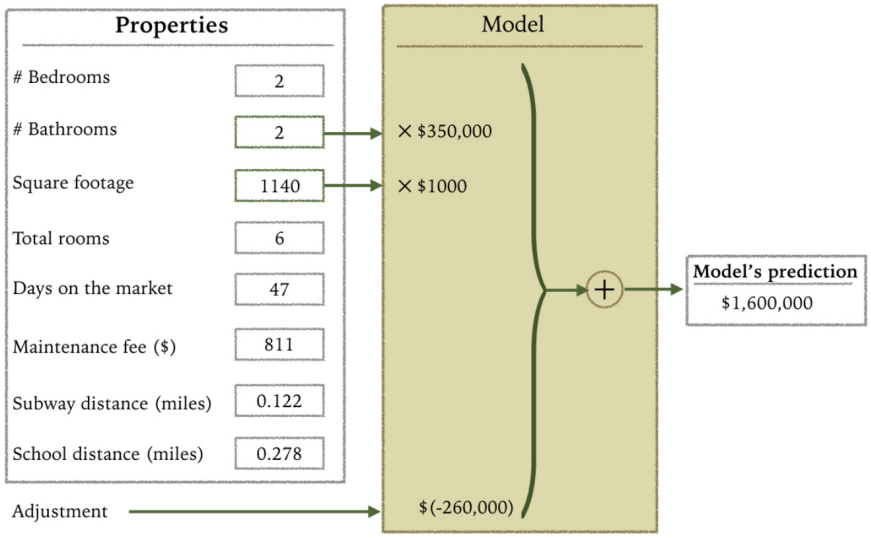
Figura 2.12: En esta gráfica de [29], se muestra cómo un árbol de decisión clasifica un conjunto de datos dividiendo el espacio de características en regiones distintas, representadas por diferentes colores. Esto ilustra cómo los árboles de decisión pueden mejorar la interpretabilidad de modelos complejos, como las redes neuronales profundas, al simplificar sus decisiones en reglas comprensibles. Los puntos indican las muestras de datos, mientras que las fronteras de color reflejan los límites de decisión del modelo, facilitando la comprensión de sus predicciones.

2.4.2. Confianza en Visualizaciones

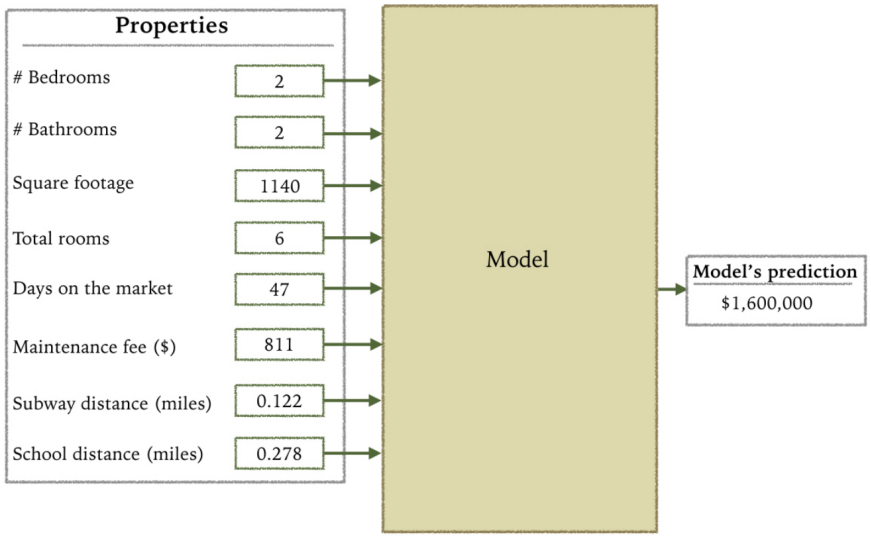
La visualización de datos juega un papel fundamental en la forma en que los usuarios interpretan y confían en los resultados de los modelos de aprendizaje automático. Estas visualizaciones pueden hacer que los modelos complejos sean más comprensibles

al proporcionar una representación gráfica de los datos y las decisiones del modelo. Sin embargo, esta simplificación visual también conlleva ciertos riesgos que pueden afectar significativamente la confianza del usuario [30, 12].

Por un lado, las visualizaciones eficaces pueden mejorar la transparencia y facilitar la comprensión de las predicciones del modelo. Diagramas de árboles de decisión, gráficos de reglas y otros tipos de visualizaciones pueden ayudar a los usuarios a seguir el proceso de toma de decisiones de un modelo, aumentando así su confianza en la precisión y confiabilidad de las predicciones. (Véase Figura 2.13).



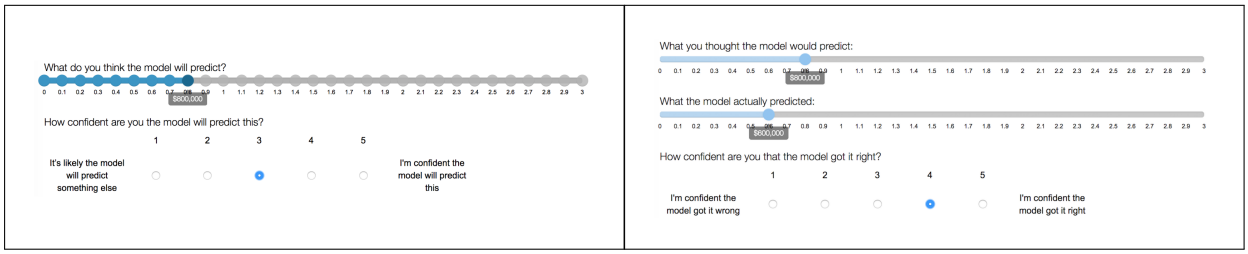
(a) Modelo Transparente (CLEAR-2): Muestra los cálculos internos, lo que puede aumentar la confianza del usuario, pero también sobrecargar de información.



(b) Modelo de Caja Negra (BB-8): Oculta los cálculos internos, lo que reduce la transparencia, pero evita la sobrecarga cognitiva.

Figura 2.13: Adaptación de las condiciones experimentales de Poursabzi et al. [30], que muestran cómo la presentación del modelo afecta la confianza del usuario.

Por otro lado, la confianza en las visualizaciones puede llevar a una comprensión superficial o a una confianza excesiva en los resultados. Investigaciones han demostrado que los usuarios tienden a confiar más en modelos cuyos resultados están respaldados por visualizaciones atractivas, incluso si no comprenden completamente los algoritmos subyacentes [12]. Por ejemplo, en la Figura 2.14, se ilustra cómo la presentación de los resultados puede influir en la percepción de los usuarios. Este fenómeno puede llevar a que los usuarios acepten las decisiones de los modelos sin cuestionarlas o sin detectar posibles errores, simplemente porque la visualización es persuasiva.



(a) Paso 1: Los participantes adivinan lo que el modelo predecirá y expresan su confianza en esa suposición.

(b) Paso 2: Los participantes ajustan su confianza en la predicción del modelo después de conocer el resultado.

Figura 2.14: Adaptación de las fases del experimento de Poursabzi et al. [30].

Además, las visualizaciones pueden ocultar la complejidad o ambigüedad de los datos, lo que resulta en una representación simplificada que no refleja completamente las incertidumbres o limitaciones del modelo. Como se observa en la Figura 2.15, es crucial que las visualizaciones se diseñen con cuidado para evitar inducir una falsa sensación de confianza o comprensión.

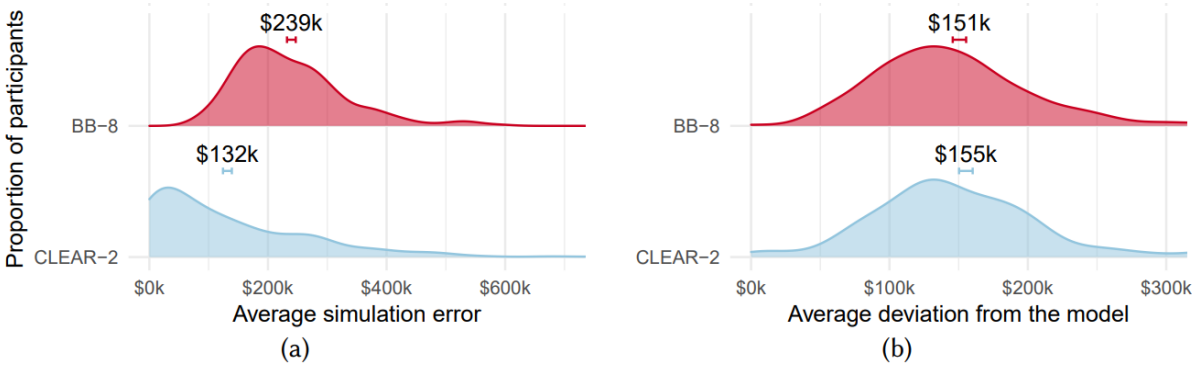


Figura 2.15: Resultados del Experimento 1 que comparan la percepción de los usuarios sobre dos tipos de modelos: un modelo de caja negra (BB-8) y un modelo transparente (CLEAR-2). En (a), se muestra el error promedio de simulación, donde los participantes tienden a percibir mayores errores en el modelo de caja negra en comparación con el modelo transparente. En (b), se presenta la desviación promedio de las predicciones del modelo respecto a los valores reales, indicando cómo los participantes evalúan la precisión del modelo según su nivel de transparencia. Adaptado de Poursabzi et al. [30].

2.5. Evaluación de la Interpretabilidad: Métodos y Métricas

Para garantizar que los usuarios finales comprendan cómo funcionan los modelos de IA, es crucial llevar a cabo estudios empíricos que evalúen su interpretabilidad [9].

2.5.1. Necesidad de Validación Empírica

La interpretabilidad no es un concepto absoluto; varía según el contexto, el modelo y las expectativas de los usuarios finales [31]. Por ello, no se puede asumir que un modelo es interpretativo sin estudios empíricos que lo validen, los cuales permiten evaluar cómo los usuarios interactúan con los modelos, comprenden sus decisiones y cómo esta comprensión afecta su confianza y capacidad para detectar errores o sesgos [12].

Estos estudios también identifican las características del modelo más relevantes para los usuarios y cómo estas influyen en su toma de decisiones, proporcionando información valiosa para adaptar las explicaciones del modelo a diferentes necesidades [32]. Las metodologías empíricas, como experimentos con usuarios, encuestas, entrevistas o evaluación de tareas, permiten medir la capacidad de los usuarios para entender y utilizar las salidas del modelo de manera efectiva, desarrollando así herramientas adaptadas a diversos contextos [9].

2.5.2. Métodos de Evaluación

Existen varios métodos para evaluar la interpretabilidad de los modelos de aprendizaje automático. Algunos de los enfoques más comunes incluyen:

- *Estudios de Usuario:* involucran experimentos con usuarios reales para evaluar cómo entienden y perciben la interpretabilidad de un modelo. Por ejemplo, se puede medir el tiempo que los usuarios tardan en comprender una predicción o su precisión al identificar errores en las predicciones del modelo. Este tipo de estudios es particularmente útil en entornos donde la decisión basada en el modelo tiene un impacto crítico, como en diagnósticos médicos o aplicaciones financieras [12].

- *Métricas Cuantitativas:* estas métricas miden características específicas de los modelos que se consideran proxies de interpretabilidad, tales como la simplicidad, la consistencia y la cobertura de las reglas en los modelos basados en reglas como los *Interpretable Decision Sets* (IDS) y los árboles de decisión (DT) [13]. Ejemplos incluyen el número de nodos en un árbol de decisión o la longitud de una regla en un conjunto de reglas. Las métricas cuantitativas permiten una evaluación objetiva de la interpretabilidad, aunque no siempre capturan completamente la percepción del usuario.

- *Evaluación Basada en Tareas:* evalúa cómo los usuarios utilizan el modelo para completar tareas específicas. Por ejemplo, se podría evaluar si los usuarios pueden hacer predicciones más precisas con la ayuda del modelo interpretativo o si pueden identificar correctamente los errores cuando se les presenta una visualización del modelo. Este enfoque es útil para validar la efectividad práctica de un modelo en escenarios del mundo real [9].

- *Encuestas y Cuestionarios:* herramientas como cuestionarios estandarizados pueden medir la percepción de los usuarios sobre la transparencia, la facilidad de comprensión y la confianza en el modelo [32]. Los cuestionarios pueden incluir preguntas sobre cuán claro es el modelo, cuán fácil es de entender, y cuánta confianza inspira en sus decisiones [12]. Estas herramientas son esenciales para capturar las percepciones subjetivas de los usuarios y ajustar los modelos interpretativos a sus necesidades específicas.

Combinar estos métodos proporciona una visión más completa y precisa de la interpretabilidad de los modelos de IA, adaptada a las necesidades y expectativas de diferentes usuarios y aplicaciones. Una evaluación integral de la interpretabilidad debe tener en cuenta tanto las métricas cuantitativas como los estudios cualitativos, para capturar tanto la objetividad de las características del modelo como las percepciones subjetivas de los usuarios [9, 12].

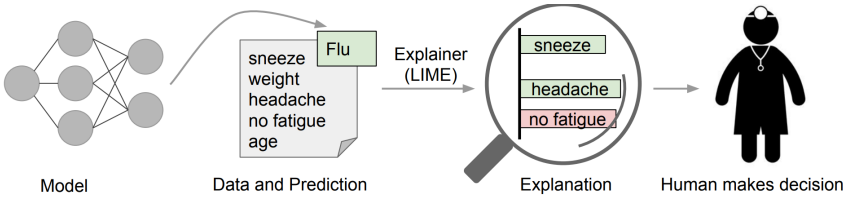


Figura 2.16: Explicación de predicciones individuales usando LIME. Un modelo predice que un paciente tiene gripe, y LIME resalta los síntomas en la historia del paciente que llevaron a la predicción. "Sneeze"(estornudo) y "headache"(dolor de cabeza) contribuyen a la predicción de "flu"(gripe), mientras que "no fatigue"(sin fatiga) es evidencia en contra de ella. Con esta información, un médico puede tomar una decisión informada sobre si confiar o no en la predicción del modelo.

Como se puede observar en la Figura 2.5, las diferentes herramientas de interpretabilidad, como SHAP y GAMs, ofrecen visualizaciones que varían en términos de claridad y nivel de detalle proporcionado al usuario.

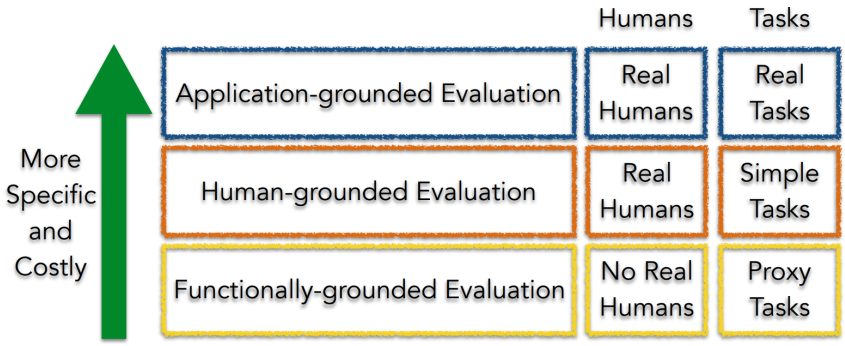


Figura 2.17: Taxonomía de métodos de evaluación de la interpretabilidad, según Doshi-Velez y Kim (2017). La figura clasifica los métodos de evaluación en tres categorías: evaluación basada en aplicaciones, evaluación basada en humanos y evaluación basada en funciones. Cada categoría varía en términos de especificidad y costo, siendo las evaluaciones basadas en aplicaciones las más específicas y costosas, ya que implican tareas reales y usuarios humanos reales, mientras que las evaluaciones basadas en funciones son las menos costosas y específicas, al no requerir la participación de usuarios reales y emplear tareas proxy.

Método	**Modelo**	**Interpretabilidad**	**Usabilidad**	**Aplicación**
Explicaciones de Saliencia	Caja Negra	Local	Media	Diagnóstico Médico
Gráficos de Dependencia Parcial	Caja Blanca/Negra	Global	Alta	Análisis Económico
Modelos de Reglas	Caja Blanca	Local/Global	Alta	Aplicaciones Regulatorias

Cuadro 2.1: Comparación simplificada de métodos de explicabilidad de modelos de aprendizaje automático. La tabla clasifica tres métodos comunes de explicabilidad: Explicaciones de Saliencia, Gráficos de Dependencia Parcial y Modelos de Reglas, según el tipo de modelo al que se aplican (caja negra o blanca), el nivel de explicabilidad que proporcionan (local o global), su usabilidad, y sus aplicaciones típicas. Esta clasificación se basa en estudios previos [12, 9, 13] que discuten la efectividad de cada método en diferentes contextos, permitiendo seleccionar el enfoque más adecuado según las necesidades de los usuarios finales y el tipo de modelo utilizado.

2.6. Resumen

En este capítulo, se han abordado los conceptos clave relacionados con la interpretabilidad en inteligencia artificial, destacando su importancia en aplicaciones críticas donde la transparencia y la confianza en los modelos son esenciales. Se han explorado enfoques intrínsecos para mejorar la interpretabilidad, como la *sparsidad*, la *simulabilidad*, la *modularidad* y la *parsimonia*, cada uno facilitando una mejor comprensión del proceso de toma de decisiones de los modelos.

También se han examinado métodos basados en reglas, como el *descubrimiento de subgrupos*, los *conjuntos de contraste* y los *patrones emergentes*, que identifican patrones diferenciados en los datos y proporcionan explicaciones claras de las decisiones del modelo, fundamentales para caracterizar las clases de manera comprensible para los usuarios finales.

Estos fundamentos teóricos proporcionan un marco sólido para el diseño del cuestionario de este TFM.

Capítulo 3

Estado del Arte

La Inteligencia Artificial Explicable (XAI, por sus siglas en inglés de *eXplainable Artificial Intelligence*) se refiere a un conjunto de métodos y técnicas diseñados para hacer que los modelos de aprendizaje automático sean comprensibles e interpretables para los usuarios humanos. El objetivo es permitir a los usuarios confiar y gestionar de manera efectiva los sistemas basados en aprendizaje automático [10].

En los últimos años, la comunidad científica se ha interesado en desarrollar estos métodos debido a la creciente utilización de la inteligencia artificial, lo que incrementa la necesidad de asegurar su uso ético y seguro. En este contexto, XAI se enfoca en cuatro objetivos principales [33]:

- *Confianza y aceptación*. Para que los sistemas de IA sean ampliamente aceptados, es fundamental que los usuarios confíen en sus decisiones. Explicar cómo y por qué un modelo toma una decisión específica es crucial para construir esta confianza.

- *Transparencia*. En muchas aplicaciones, especialmente en dominios sensibles como la medicina, la justicia y las finanzas, es esencial que los modelos de IA sean transparentes. Esto facilita no solo la detección y corrección de errores, sino también asegura que las decisiones sean justas y no discriminatorias.

- *Cumplimiento normativo*. Diversas regulaciones, como el Reglamento General de Protección de Datos (GDPR) en Europa, exigen que las decisiones automatizadas sean explicables. Esto garantiza que los usuarios tengan derecho a una explicación clara y comprensible de cómo se toman las decisiones que les afectan.

- *Mejora del modelo*. Comprender cómo funciona un modelo permite a los desarrolladores identificar áreas de mejora y ajustar los modelos para obtener mayor rendimiento y precisión.

Modelo	Enfoque	Características
Caja Negra	Modelos Subrogados	Uso de modelos más simples para aproximar las predicciones
	Métricas del Efecto	Evaluación del impacto de variables (locales y globales)
	Explicabilidad basada en Ejemplos	Uso de ejemplos específicos para explicar predicciones
Transparentes	Intrínsecamente Interpretables	Diseñados para ser comprensibles desde su construcción
	Optimización para Interpretabilidad	Ajustados específicamente para mejorar la claridad y explicación

Cuadro 3.1: Comparación de técnicas en XAI según el tipo de modelo.

Los métodos y técnicas desarrollados en XAI se pueden agrupar en dos categorías principales según el tipo de modelo: modelos de caja negra y modelos transparentes [33]. Estas categorías se subdividen en técnicas específicas, como se muestra en el Cuadro 3.1.

3.1. XAI para Modelos de Caja Negra

Aunque este trabajo no se enfoca en métodos aplicables a modelos de caja negra, es importante mencionarlos para brindar un panorama completo de la Inteligencia Artificial Explicable (XAI). En general, un *modelo de caja negra* es aquel cuyo funcionamiento interno es desconocido o no es interpretable. Esto puede deberse a que su acceso está restringido (por propiedad intelectual) o porque su mecanismo de predicción es inherentemente complejo, como en el caso de las redes neuronales [33, 14].

Existen tres tipos principales de técnicas para abordar la explicabilidad en modelos de caja negra [33]:

- *Modelos subrogados*: Se refiere a la creación de un modelo más sencillo e interpretable que imita las predicciones del modelo de caja negra. Este modelo subrogado puede aproximar el comportamiento del modelo original de manera *local* (para un subconjunto específico de datos de entrada) o *global* (para el conjunto completo de datos).
- *Explicabilidad basada en ejemplos*: Busca explicar una predicción específica proporcionando ejemplos similares y contraejemplos. Esto ayuda al usuario a entender cómo diferentes características influyen en una predicción en particular.
- *Métricas del efecto de las variables sobre la predicción*: Estas métricas buscan cuantificar la influencia de cada variable de entrada en las predicciones del modelo. Algunas técnicas destacadas en esta categoría son:
  - *LIME (Local Interpretable Model-agnostic Explanations)*: Genera explicaciones locales al identificar cuáles características influyen en una predicción específica y su efecto (positivo o negativo) [32].

- *SHAP (SHapley Additive exPlanations)*: Calcula la contribución de cada característica a la predicción utilizando conceptos de la Teoría de Juegos [34].
- *Eliminación de características*: Consiste en eliminar una característica del modelo (generalmente, reentrenando el modelo) para observar cómo cambian las predicciones sin dicha característica.
- *Ocultamiento de características*: Similar a la eliminación, pero se ocultan parcial o totalmente algunas características, utilizado principalmente en redes neuronales convolucionales para entender qué partes de los datos son más relevantes.

3.2. XAI para Modelos Transparentes

Los modelos transparentes son aquellos que no solo exponen el mecanismo mediante el cual generan las predicciones, sino que este proceso es fácilmente comprensible para un usuario humano. Un ejemplo clásico es la regresión lineal: es fácil entender cómo este modelo genera una predicción (mediante multiplicaciones por coeficientes y una suma), y también es fácil interpretar el significado de estos coeficientes (el valor de cada coeficiente indica el efecto de la variable correspondiente sobre la predicción).

En principio, se presume que los modelos transparentes son interpretables, es decir, que un humano puede entender el significado de los elementos del proceso de decisión y sus relaciones, como en el ejemplo de la regresión lineal. Sin embargo, en la práctica, un modelo transparente puede ser tan complejo que sobrepase la capacidad cognitiva de una persona. Por ejemplo, una regresión lineal con mil variables resulta difícil de interpretar, ya que es complejo entender el impacto de tantos factores en una predicción.

Para mejorar la interpretabilidad de estos modelos, se pueden aplicar técnicas de regularización como la *Lasso*, que promueven la *sparsidad* al reducir el número de variables que tienen un impacto significativo en las predicciones. Esta técnica puede observarse en la Figura 2.1, donde se ilustra cómo Lasso minimiza la inclusión de variables irrelevantes, mejorando así la simplicidad y la claridad del modelo.

De esta forma, se facilita que el usuario enfoque su atención solo en las características más importantes, lo que mejora la comprensión general del modelo.

3.3. Métricas de Evaluación de la Explicabilidad en Modelos Transparentes

Esta sección discute brevemente métricas cuantitativas propuestas en la literatura para evaluar y mejorar la explicabilidad de modelos transparentes. Aunque estas métricas se justifican intuitivamente, es importante destacar que su efectividad puede depender del contexto específico del modelo y del usuario, como se ha señalado en los fundamentos teóricos (ver Figuras 2.1, 2.2, y 2.4).

3.3.1. Número de Características

El *número de características* utilizadas por un modelo es una de las métricas más comúnmente aplicadas para medir la explicabilidad [35, 1, 30]. Este concepto se re-

laciona estrechamente con la *sparsidad*, ilustrada en la Figura 2.1, que busca reducir el número de parámetros no nulos en un modelo para facilitar la comprensión del papel específico de cada variable en las predicciones. En modelos como las regresiones lineales (y otros modelos aditivos, como los *Generalized Additive Models*), es posible promover la *sparsidad* introduciendo un término de regularización en la función de pérdida, tal como se explicó en los fundamentos teóricos.

Esta idea también se extiende a otros tipos de modelos, como los árboles de decisión. Por ejemplo, [36, 35] proponen el uso de técnicas de regularización para favorecer árboles de decisión *esparsos* (del inglés *sparse*), utilizando programación dinámica y teoremas que garantizan un desempeño mínimo. Este enfoque promueve la simplicidad y la *parsimonia*, representada en la Figura 2.4, utilizando la menor cantidad de variables posible sin comprometer el rendimiento.

En el caso de conjuntos de decisión, [1] proponen tres métricas: una para penalizar la función de pérdida basada en el número de reglas individuales en el conjunto de decisiones, otra para penalizar la longitud de las reglas, y una última para maximizar la cobertura de una regla (es decir, aplicarla al mayor número posible de datos). Este enfoque de minimización también se alinea con los principios de *parsimonia* y *modularidad* (ver Figura 2.4), al reducir la complejidad del modelo mientras se mantiene la efectividad de la predicción.

3.3.2. Complejidad de la Estructura del Modelo

La *complejidad de la estructura* de un modelo transparente también afecta su interpretabilidad, como se discutió en los fundamentos teóricos en relación con la *simulabilidad* (ver Figura 2.2). Diferentes configuraciones estructurales pueden hacer que un modelo sea más o menos fácil de explicar.

Por ejemplo, en los árboles de decisión, tanto el número de ramas por nodo como la profundidad del árbol influyen en su interpretabilidad. Un árbol con muchas ramas o de gran profundidad puede ser difícil de entender, contraviniendo el principio de *simulabilidad* (ver Figura 2.2), que sugiere que los modelos deben ser fáciles de reproducir y comprender por humanos. Para mitigar esto, [35] recomienda el uso de árboles binarios (cada nodo tiene solo dos ramas), y la programación dinámica para limitar la profundidad del árbol, controlando así la complejidad del modelo.

Para los conjuntos de decisión, [1] sugiere el uso de listas simples de reglas, sin anidación, ya que se considera que son más fáciles de entender que las secuencias de reglas anidadas. Esta preferencia por la simplicidad y la claridad se alinea con la noción de *parsimonia* (ver Figura 2.4), donde se favorecen las soluciones más simples y directas.

3.4. Métricas para la Evaluación Humana de la Interpretabilidad

Esta sección presenta las métricas propuestas en la literatura para la evaluación humana de la interpretabilidad. Dichas métricas son esenciales para determinar si los usuarios pueden entender un modelo de aprendizaje automático, especialmente cuando se trata de modelos transparentes, como los discutidos en la Figura 2.3 sobre la modularidad y en la Figura 2.2 sobre la simulabilidad. Estas métricas se dividen

en dos categorías principales: cualitativas y cuantitativas, dependiendo de su enfoque para medir el entendimiento.

3.4.1. Métricas cualitativas

Las métricas cualitativas buscan evaluar la capacidad del usuario para comprender el modelo mediante la recopilación de datos no numéricos. Generalmente, estos datos se obtienen a través de entrevistas semi-estructuradas o preguntas contextualizadas [12]. Tal como se ilustra en la Figura 2.2, los modelos que favorecen una estructura simple, como los árboles de decisión, facilitan la comprensión del proceso de decisión por parte del usuario. En este contexto, se puede emplear una metodología en la que el usuario tenga acceso al modelo o a visualizaciones relacionadas, y un investigador formule preguntas, asigne tareas o simplemente escuche al usuario para determinar su grado de entendimiento.

Por ejemplo, [12] realizó entrevistas semi-estructuradas para identificar las principales dificultades que enfrentan los científicos de datos al interpretar modelos complejos. Basándose en estos resultados, se diseñaron entrevistas adicionales, utilizando el mismo enfoque mostrado en la Figura 2.5, para determinar cómo los diferentes grupos de usuarios perciben la interpretabilidad de los modelos en función de su contexto y necesidades específicas.

3.4.2. Métricas cuantitativas

Las métricas cuantitativas, a diferencia de las cualitativas, se enfocan en la recopilación de datos numéricos que puedan ser analizados de manera estadística. Similar a cómo se evaluó la interpretabilidad mediante la reducción de características en la Figura 2.1, estas métricas intentan medir la capacidad del usuario para comprender un modelo a través de la exactitud de sus predicciones o la eficiencia con la que completan tareas basadas en el modelo.

La métrica más común es la exactitud o el error de desviación [1, 30], que consiste en solicitar al usuario que realice predicciones utilizando el modelo, como se describe en la Figura 2.3, donde los Modelos Aditivos Generalizados (GAMs) descomponen las predicciones en componentes interpretables. Esta métrica permite determinar si un usuario es capaz de seguir el razonamiento del modelo y entender cómo se llega a una predicción específica.

Adicionalmente, se emplean otras métricas como el *error de simulación* [30], en las que se muestra al usuario una serie de ejemplos y se le pide realizar predicciones para evaluar su capacidad de aprendizaje sin ayuda del modelo. De manera complementaria, el uso de cuestionarios como el NASA-TLX (*Task Load Index*) [37], que puede ayudar a cuantificar la carga cognitiva que experimentan los usuarios al interactuar con el modelo, proporcionando una medida indirecta de la interpretabilidad.

En conjunto, estas métricas ofrecen una visión integral de cómo los usuarios perciben la interpretabilidad de los modelos de aprendizaje automático, conectando las técnicas teóricas presentadas en la sección de fundamentos con la evaluación práctica de la experiencia del usuario.

3.5. Herramientas de Interpretabilidad para Modelos Transparentes

A diferencia de los modelos de caja negra, que requieren explicaciones post-hoc, los modelos transparentes permiten interpretar directamente sus decisiones debido a su estructura lógica. Sin embargo, aunque estos modelos, como los árboles de decisión (DT) y los Interpretable Decision Sets (IDS), ya son comprensibles, la claridad en la presentación y las visualizaciones influyen en cómo los usuarios perciben su interpretabilidad [33, 14]. La manera en que se presentan los resultados puede facilitar o dificultar la identificación de patrones, la detección de errores y la confianza en las decisiones.

Para abordar este reto, se han desarrollado herramientas que facilitan tanto el análisis de los modelos transparentes como la exploración de cómo las visualizaciones y la estructura del modelo afectan la experiencia del usuario. Estas herramientas combinan enfoques visuales y técnicos que mejoran la comprensión y fomentan la confianza en las predicciones.

A continuación, se describen herramientas clave en el campo de la inteligencia artificial explicable que facilitan el análisis y optimización de la interpretabilidad de modelos transparentes. Estas herramientas permiten evaluar cómo la presentación visual y la estructura de un modelo influyen en la percepción de interpretabilidad y la confianza del usuario en los sistemas de IA. Además, proporcionan una base sólida para estudiar los modelos transparentes de este trabajo, aplicando metodologías prácticas y enfoques visuales que mejoran tanto la claridad como la comprensión de los modelos de decisión.

3.5.1. InterpretML

InterpretML, propuesto por Nori et al. [22], es un marco de código abierto diseñado para proporcionar interpretabilidad tanto en modelos de caja blanca como en técnicas de explicabilidad post-hoc para modelos de caja negra. Este marco unifica diversas técnicas y permite a los investigadores y profesionales comparar diferentes enfoques de interpretabilidad a través de una API integrada y visualizaciones interactivas.

InterpretML ofrece dos tipos principales de interpretabilidad:

- Modelos de caja blanca: Estos modelos son transparentes desde su construcción, permitiendo que los usuarios comprendan directamente cómo cada característica influye en las predicciones. Ejemplos incluyen los modelos aditivos generalizados (GAM) y los modelos lineales. En la Figura 2.3 se muestra un ejemplo de cómo la característica 'Edad' afecta la predicción en un modelo EBM, proporcionando una visualización clara de las relaciones entre características y resultados.
- Explicabilidad para modelos de caja negra: Utilizando técnicas post-hoc como dependencia parcial, LIME y SHAP, InterpretML proporciona interpretaciones locales y globales de modelos más complejos, sin necesidad de conocer su estructura interna. La Figura 2.5 compara visualizaciones generadas por GAM (arriba) y SHAP (abajo), mostrando cómo diferentes enfoques interpretativos pueden ajustarse a audiencias y contextos específicos.

Una de las contribuciones clave de InterpretML es el Explainable Boosting Machine (EBM), un modelo de caja blanca basado en un modelo aditivo generalizado (GAM) que utiliza técnicas modernas como bagging y boosting para ajustar cada característica iterativamente, reduciendo la correlación entre ellas. Esto permite que el EBM logre niveles de precisión comparables a los de modelos de caja negra como los bosques aleatorios o XGBoost, pero manteniendo la interpretabilidad total.

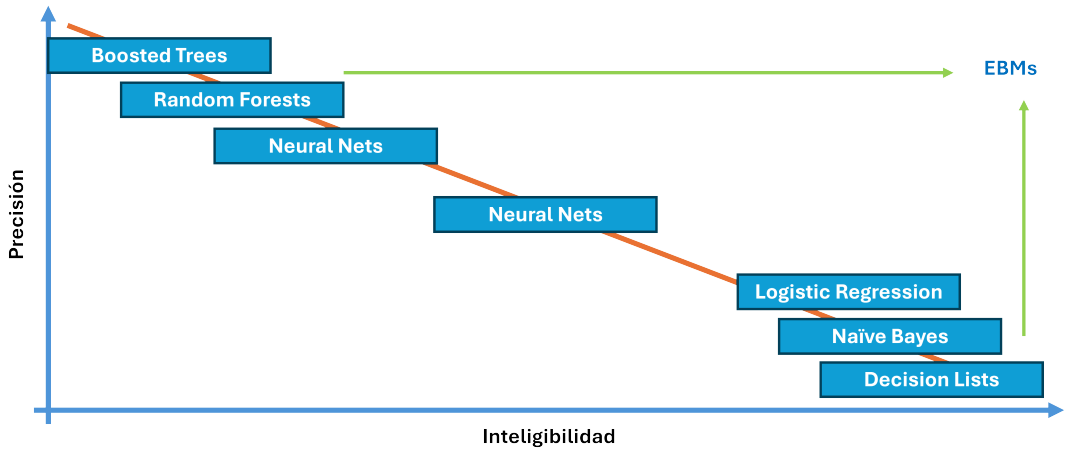


Figura 3.1: Gráfico comparativo de precisión e inteligibilidad de modelos de aprendizaje automático, inspirado en un gráfico presentado por [38].

El EBM es modular y permite visualizar cómo cada característica influye en las predicciones, facilitando la comprensión para los usuarios mediante gráficos claros, como los mostrados en la Figura 2.3, que ilustran la descomposición de una predicción individual y cómo cada característica contribuye de forma independiente.

InterpretML presenta varias ventajas:

- Facilidad de comparación: La API unificada facilita la comparación de diferentes algoritmos de interpretabilidad, con integración sencilla a frameworks como scikit-learn.
- Interoperabilidad: Es compatible con herramientas como Jupyter Notebook y plotly, permitiendo una interacción intuitiva con los modelos y sus explicaciones.
- Visualización interactiva: InterpretML ofrece un panel de control interactivo que permite explorar visualmente las explicaciones generadas, ayudando a los usuarios a entender fácilmente las contribuciones de cada característica.

InterpretML ha demostrado ser útil en áreas como la salud, las finanzas y la justicia, donde la interpretabilidad es crítica. Su capacidad para combinar precisión y transparencia lo convierte en una herramienta esencial en sistemas de alta responsabilidad.

3.5.2. Yellowbrick

Yellowbrick es una biblioteca de visualización diseñada para facilitar la evaluación e interpretación de modelos de aprendizaje automático creados con scikit-learn. Esta

herramienta ofrece una amplia gama de visualizaciones que permiten a los usuarios entender mejor el comportamiento de los modelos, ayudando en la toma de decisiones y mejorando la interpretabilidad de los mismos [39].

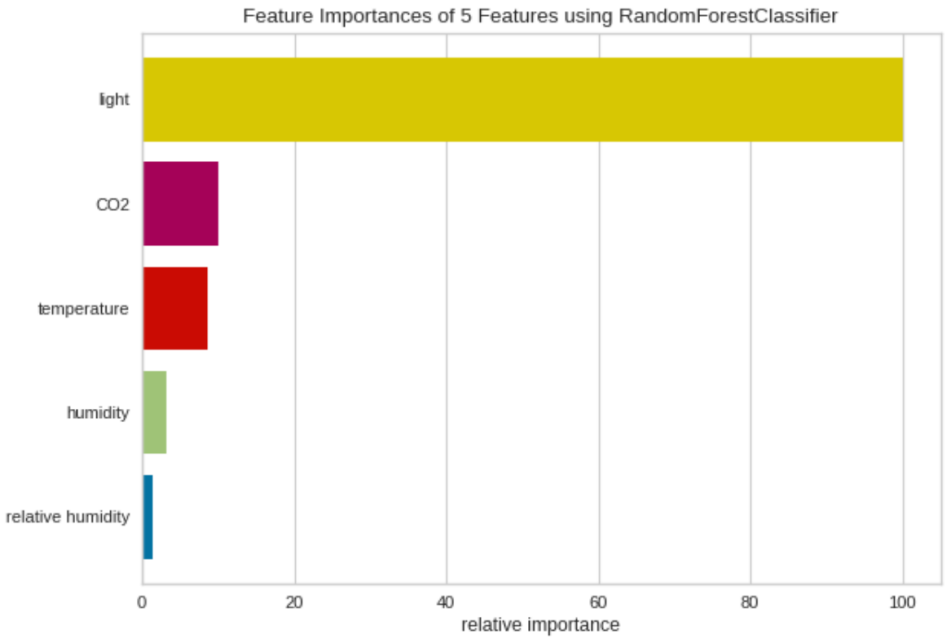


Figura 3.2: Visualización de la importancia de características generada por Yellowbrick utilizando un modelo *RandomForestClassifier*. El gráfico muestra la influencia relativa de cinco características sobre las predicciones del modelo. La característica 'light' tiene la mayor importancia relativa, seguida de 'CO2' y 'temperature', lo que indica su fuerte contribución a las decisiones del modelo [39].

Yellowbrick se enfoca en generar gráficos que muestran cómo las características y el rendimiento de los modelos afectan los resultados, lo cual es esencial para los modelos transparentes como los Árboles de Decisión (DT) y los Conjuntos de Decisión Interpretables (IDS) utilizados en este trabajo. Entre sus funcionalidades más destacadas se incluyen:

- *Visualización de la importancia de características*: Yellowbrick permite representar gráficamente la importancia relativa de las características dentro de un modelo, como se muestra en la Figura 3.2. En este TFM, esta herramienta es fundamental para comprender el impacto de las distintas características en las predicciones de los modelos de decisión. En particular, se ha empleado para identificar las características más influyentes dentro del dataset de matemáticas utilizado en este proyecto.
- *Curvas de validación y curvas de aprendizaje*: Estas curvas permiten observar cómo el rendimiento del modelo varía según los valores de los hiperparámetros y cómo mejora a medida que recibe más datos de entrenamiento. Este tipo de visualización es útil para diagnosticar problemas de sobreajuste o subajuste, los cuales pueden afectar la interpretabilidad del modelo en este trabajo.
- *Matrices de confusión y reportes de clasificación*: Facilitan la visualización de cómo el modelo clasifica correctamente e incorrectamente las instancias, lo que

ayuda a comprender mejor el comportamiento del modelo en cada clase. Esta herramienta ha sido utilizada en este proyecto para analizar los resultados de los modelos DT y evaluar sus predicciones en relación con los datos de rendimiento académico en matemáticas.

- *Curvas ROC y AUC*: Permiten evaluar el rendimiento de los modelos de clasificación, mostrando el compromiso entre la tasa de verdaderos positivos y la tasa de falsos positivos en diferentes umbrales de decisión. Estas curvas son especialmente útiles para evaluar la capacidad de los modelos para manejar errores de clasificación.

Yellowbrick ofrece ventajas significativas, entre las que destaca su integración perfecta con scikit-learn, lo que permite a los usuarios generar visualizaciones sin una configuración adicional compleja. Además, proporciona una interfaz fácil de usar que se adapta tanto a principiantes como a expertos, facilitando la evaluación visual de los modelos de manera clara y concisa.

3.5.3. Anchors

Anchors, propuestos por Ribeiro, Singh y Guestrin [40], son reglas locales que explican predicciones individuales de un modelo a través de condiciones suficientes. Estas reglas aseguran que si las condiciones de un *anchor* se cumplen, la predicción del modelo será la misma con alta probabilidad. Los *anchors* son especialmente útiles en modelos de caja negra o en tareas donde la interpretabilidad es fundamental. Aunque en este trabajo no implementamos *Anchors*, su relevancia para el campo de la IA explicable es notable, ya que proporcionan explicaciones simples y de alta precisión que son fácilmente comprensibles por los usuarios.

En la siguiente tabla se muestra un ejemplo adaptado del artículo original, donde se utiliza *Anchors* para etiquetar la parte del discurso de la palabra "play.[en] diferentes contextos.

Instancia	**Condición**	**Predicción**
I want to play(V) ball.	La palabra previa es **PARTICLE**	play es **VERBO**.
I went to a play(N) yesterday.	La palabra previa es **DETERMINANTE**	play es **SUSTANTIVO**.
I play(V) ball on Mondays.	La palabra previa es **PRONOMBRE**	play es **VERBO**.

Cuadro 3.2: Ejemplo de Anchors para la etiqueta de parte del discurso de la palabra "play"(adaptado de Ribeiro, Singh y Guestrin [40]).

En este ejemplo, el modelo predice si la palabra “play” es un verbo o un sustantivo dependiendo de la palabra que la precede, generando reglas locales que son fáciles de comprender y aplicar. Estos *anchors* proporcionan una explicación clara sobre el comportamiento del modelo en casos específicos, lo que mejora la confianza y la interpretabilidad para los usuarios finales.

3.5.4. DiCE: Explicaciones Contrafactuales Diversas

DiCE (*Diverse Counterfactual Explanations*) es un marco propuesto por Mothilal, Sharma y Tan [41] para generar explicaciones contrafactuales que ayuden a los usuarios a

comprender el comportamiento de modelos de aprendizaje automático. DiCE se centra en proporcionar múltiples explicaciones contrafactuales diversas, que exploran varias formas en las que se podría modificar una instancia para obtener una predicción diferente. Este enfoque permite a los usuarios entender qué cambios mínimos en los atributos de entrada podrían alterar la decisión del modelo, haciendo que este sea más interpretable.

Una *explicación contrafactual* responde a la pregunta: "¿Qué habría que cambiar en esta instancia para obtener un resultado diferente?". DiCE es capaz de generar varias explicaciones contrafactuales que muestran diferentes caminos posibles para lograr un cambio en la predicción. Esto es particularmente útil cuando hay varias combinaciones de características que pueden influir en el resultado, ya que proporciona al usuario un conjunto diverso de escenarios.

DiCE se basa en los siguientes principios:

- *Diversidad*: DiCE no genera una única explicación contrafactual, sino varias. Cada una ofrece una manera diferente de alterar la predicción de un modelo. Esta diversidad es útil para evitar que el usuario se enfoque solo en una explicación que podría no ser la mejor o más factible.
- *Flexibilidad*: Los usuarios pueden especificar qué características se pueden cambiar y cuáles no. Esto es útil en casos donde ciertos atributos son fijos (por ejemplo, la edad de una persona) y no pueden ser modificados en las explicaciones contrafactuales.
- *Compatibilidad con modelos de caja negra*: DiCE es un enfoque agnóstico al modelo, lo que significa que puede ser aplicado a una variedad de modelos, incluidos los modelos de caja negra.
- *Optimización*: DiCE utiliza un algoritmo de optimización para generar explicaciones que minimizan el número de cambios necesarios en la instancia original. De esta forma, se realizan interpretaciones más realistas y fáciles de entender.

DiCE emplea un enfoque basado en la búsqueda de ejemplos contrafactuales cercanos que resulten en un cambio en la predicción del modelo. Dados los inputs de una instancia x y una predicción del modelo $f(x)$, DiCE busca generar nuevos ejemplos x' que pertenezcan a una clase diferente y que estén lo más cerca posible de la instancia original. Para lograr esto, DiCE resuelve el siguiente problema de optimización:

$$\min_{x'} d(x, x') \quad \text{sujeto a} \quad f(x') \neq f(x)$$

donde $d(x, x')$ es una métrica de distancia que asegura que los ejemplos contrafactuales sean similares a la instancia original. El objetivo es generar instancias contrafactuales x' que difieran lo menos posible de x y que den lugar a una predicción diferente por parte del modelo f.

Como ejemplo, consideremos un modelo de predicción de aprobación de préstamos. Si el modelo predice que un cliente no será aprobado para un préstamo, DiCE puede generar múltiples explicaciones contrafactuales mostrando qué atributos del cliente deben cambiarse para que la solicitud sea aprobada. Por ejemplo, una explicación podría sugerir que aumentando los ingresos anuales y reduciendo la deuda actual, el

cliente podría obtener una aprobación. Otra explicación podría sugerir que reducir el número de créditos activos mejoraría las probabilidades de aprobación.

Atributo Original	Atributo Modificado (Contrafactual 1)	Atributo Modificado (Contrafactual 2)
Ingresos anuales: $35,000	Ingresos anuales: $50,000	Ingresos anuales: $35,000
Deuda actual: $10,000	Deuda actual: $5,000	Deuda actual: $10,000
Créditos activos: 3	Créditos activos: 3	Créditos activos: 1

Cuadro 3.3: Ejemplo de explicaciones contrafactuales generadas por DiCE para un modelo de aprobación de préstamos. Se muestran múltiples combinaciones de atributos que podrían llevar a una predicción diferente.

Aunque DiCE es eficaz para generar explicaciones contrafactuales diversas, no garantiza que todas las explicaciones sean factibles o realistas en un contexto del mundo real. Por ejemplo, en el caso de la predicción de préstamos, aumentar los ingresos anuales puede no ser una opción viable a corto plazo para muchos individuos. Por lo tanto, es crucial que los usuarios de DiCE evalúen la plausibilidad de las explicaciones generadas.

DiCE es una herramienta poderosa para la IA explicable, permitiendo a los usuarios explorar cómo pequeños cambios en las entradas pueden alterar los resultados del modelo, y proporcionando una visión más completa del comportamiento del modelo a través de explicaciones diversas.

3.6. Resumen

En este estado del arte se han revisado las principales herramientas y enfoques de XAI, con especial énfasis en los modelos transparentes. También, se han explorado herramientas clave como InterpretML y Yellowbrick, que permiten visualizar y analizar la interpretabilidad de estos modelos, facilitando su comprensión por parte de los usuarios.

Asimismo, se han discutido métodos como Anchors, que proporciona reglas locales explicativas, y DiCE, que genera explicaciones contrafactuales diversas, destacando cómo estas técnicas ayudan a mejorar la transparencia y confianza en las predicciones de los modelos.

Esta revisión proporciona una base teórica sólida para el desarrollo del cuestionario de evaluación de interpretabilidad en este TFM, identificando los enfoques más relevantes para medir cómo los usuarios perciben la interpretabilidad de los modelos utilizados.

Capítulo 4

Metodología

En esta sección se describe la propuesta de este TFM, que consiste en el desarrollo de una herramienta de evaluación de la interpretabilidad basada en dos modelos: Árboles de Decisión (DT) y Conjuntos Interpretables de Decisión (IDS). La evaluación se realiza a través de un cuestionario implementado en una aplicación web diseñada para recolectar las respuestas de los usuarios y su percepción sobre la interpretabilidad de los modelos. Se explica el dataset utilizado para el entrenamiento de los modelos y generación de las preguntas del cuestionario.

El código de implementación de los modelos, el preprocesamiento de datos y el análisis de interpretabilidad se encuentra en un notebook de Google Colab disponible en [42], donde se detallan todos los pasos involucrados en el proceso.

4.1. Datos utilizados

El conjunto de datos utilizado proviene del *UCI Machine Learning Repository* bajo el nombre *Student Performance Dataset* [43]. Este dataset recoge información académica, social y demográfica de estudiantes de secundaria en dos escuelas portuguesas, centrado en las asignaturas de Matemáticas y Lengua Portuguesa. En este trabajo, solo se considera la asignatura de Matemáticas.

Este conjunto de datos es adecuado para evaluar la interpretabilidad de los modelos de clasificación debido a la variedad de características sociales, demográficas y académicas que pueden influir en el rendimiento académico. Además, la estructura del problema como una tarea de clasificación binaria permite una evaluación clara de los modelos propuestos en términos de su capacidad para predecir el éxito o fracaso de los estudiantes en la asignatura de Matemáticas.

Para la selección de las variables clave utilizadas en la construcción de los modelos, se realizó un análisis exploratorio tanto basado en la literatura [2] como en un análisis propio del dataset. El conjunto de datos original contiene un total de 33 características, que incluyen variables demográficas, sociales y académicas de los estudiantes. Inicialmente, se consideraron variables relacionadas con el desempeño académico, como *G1*, *G2* y *G3*, que corresponden a las calificaciones parciales y final en la asignatura de Matemáticas. Sin embargo, se decidió excluirlas por las siguientes razones:

- *G3*: Esta variable representa la calificación final en Matemáticas y fue utilizada como la variable objetivo o dependiente en los modelos predictivos. Por lo tanto, no puede ser considerada como una variable predictora, ya que su valor es lo que se pretende predecir.

- *G2 y G1*: Aunque estas variables corresponden a las calificaciones parciales en Matemáticas, se decidió excluirlas debido a su alta correlación con la variable *G3*. Incluir estas variables podría sesgar el modelo hacia una predicción directa basada en estas notas, lo que no aporta valor en términos de interpretabilidad. Además, el objetivo de este trabajo es evaluar la interpretabilidad de los modelos basados en características externas, como las demográficas y sociales, y no únicamente en las calificaciones previas.

Para evaluar la importancia de las características seleccionadas, se utilizó un modelo de Árbol de Decisión (Decision Tree, DT), implementado mediante la librería *Yellowbrick*, que proporciona gráficos de *Feature Importance* para visualizar la contribución de cada característica al rendimiento del modelo. Esta herramienta facilitó la validación de la selección de variables basada en su relevancia predictiva. En la Figura 4.1 se muestra la importancia relativa de cada característica dentro del modelo DT.

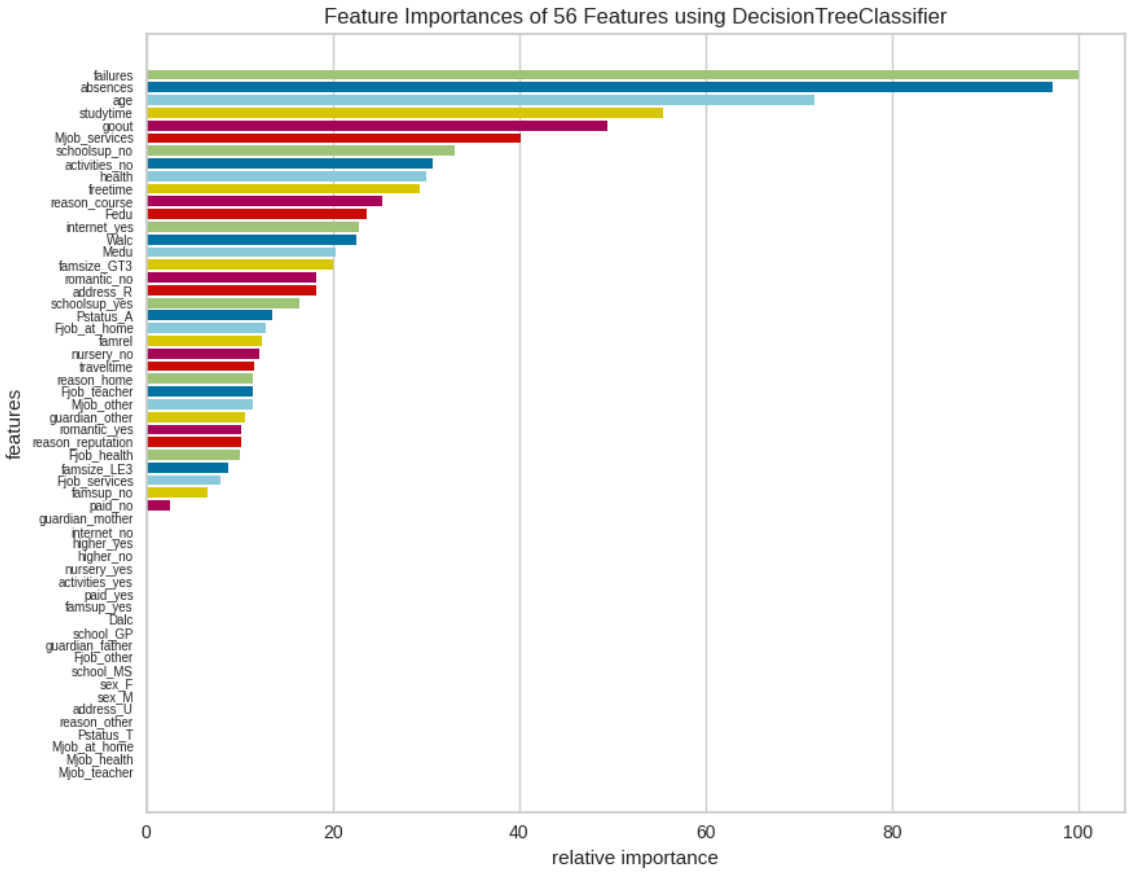


Figura 4.1: Importancia de las características calculada con *Yellowbrick* para el modelo de clasificación utilizando Árboles de Decisión (DT). Las variables seleccionadas fueron validadas en base a su impacto en la predicción de la calificación final en matemáticas (*G3*).

Para los modelos, se seleccionaron cuatro variables clave por su interpretabilidad y relevancia predictiva, basadas en el estudio de [2] y un análisis propio. Estas variables se detallan en el Cuadro 4.1.

Característica	**Descripción**
absences	Ausencias del estudiante en el año académico.
failures	Materias reprobadas en cursos previos.
studytime	Tiempo de estudio semanal: 1: Menos de 2 horas 2: 2 a 5 horas 3: 5 a 10 horas 4: Más de 10 horas
age	Edad del estudiante.
G3	Calificación final en Matemáticas.

Cuadro 4.1: Características seleccionadas para construir los modelos de predicción de rendimiento en Matemáticas, basadas en [2] y análisis propio. *G3* es la variable dependiente.

El objetivo del modelo es predecir si un estudiante aprobará la asignatura de Matemáticas. Para esto, la variable *G3* se convirtió en una variable categórica binaria, donde se considera *Aprobado* cuando el estudiante obtiene una calificación final mayor o igual a 10 (en una escala de 0 a 20). La Tabla 4.2 presenta la categorización de *G3*.

Clase	**Definición**
Aprobado	$G3 \geq 10$
Reprobado	$G3 < 10$

Cuadro 4.2: Clases creadas a partir de la calificación final (*G3*) en Matemáticas, con un umbral de 10 puntos para definir Aprobado y Reprobado [43].

4.1.1. Implementación de los Modelos

Para la evaluación de la interpretabilidad, se implementaron dos tipos de modelos:

- *Árboles de Decisión (DT):* Se utilizan por su facilidad de interpretación. Los Árboles de Decisión dividen el espacio de características mediante reglas simples, permitiendo a los usuarios visualizar cómo se toman las decisiones. Para su evaluación se emplean tanto métricas de precisión como análisis visual utilizando Yellowbrick.

- *Conjuntos Interpretables de Decisión (IDS):* Los IDS se componen de reglas conjuntas que facilitan una explicación comprensible del modelo. En este trabajo, se evalúa su capacidad para generar decisiones claras y fáciles de entender para los usuarios.

4.1.2. Diseño del Cuestionario

El cuestionario fue diseñado para recoger la percepción de los usuarios sobre la interpretabilidad de los modelos descritos anteriormente. Las preguntas fueron clasificadas en las siguientes categorías:

- *Preguntas de Precisión:* Los usuarios deben decidir si el modelo ha realizado una predicción correcta con base en las observaciones proporcionadas.
- *Detección de Errores:* Los usuarios deben identificar si el modelo puede haber cometido un error en la predicción.
- *Descripciones Generales:* Se pide a los usuarios que describan en sus propias palabras las reglas que los modelos han generado, permitiendo evaluar si el conjunto de reglas es comprensible.

4.1.3. Desarrollo de la Aplicación Web

La aplicación web fue desarrollada utilizando *Flask*, un microframework de Python, y se aloja en la plataforma Railway. El flujo de trabajo de la aplicación es el siguiente:

1. *Carga de preguntas:* Las preguntas se almacenan en un archivo JSON, que es cargado y mostrado dinámicamente en la interfaz web.
2. *Recolecta de respuestas:* Cada respuesta del usuario se almacena en una base de datos MongoDB, junto con el tiempo que toma el usuario en responder cada pregunta.
3. *Procesamiento de resultados:* La aplicación genera un ID único para cada usuario y registra las respuestas para su posterior análisis.

4.1.4. Herramientas de Interpretabilidad

Para facilitar el análisis de la interpretabilidad de los modelos, se emplearon dos herramientas principales:

- *InterpretML:* Se utilizó para generar explicaciones tanto locales como globales sobre las predicciones de los modelos implementados. El *Explainable Boosting Machine (EBM)* también fue considerado para proporcionar un modelo más preciso y altamente interpretable.
- *Yellowbrick:* Esta herramienta permitió visualizar la importancia de las características y generar gráficos como la curva ROC y la matriz de confusión, útiles para evaluar el rendimiento y la claridad de los modelos implementados.

Capítulo 5

Resultados

5.1. Desafíos en la Implementación

Durante la realización de este Trabajo Fin de Máster (TFM), se encontraron varios desafíos significativos que impactaron en el desarrollo y la implementación del proyecto. Uno de los mayores retos fue la implementación del modelo Interpretable Decision Sets (IDS). La librería pyIDS no ofrecía una clasificación adecuada ya que clasificaba todos los casos como no aprobados, lo que llevó a la decisión de desarrollar el modelo IDS desde cero para manejar este problema. La falta de documentación y recursos disponibles sobre la implementación de IDS dificultó este proceso, requiriendo un esfuerzo considerable para lograr una implementación funcional y precisa del algoritmo.

Debido a lo anterior, no fue posible completar a tiempo la preparación del cuestionario para su implementación dentro del entorno universitario. Este problema se agravó por el hecho de que la mayoría de los estudiantes estaban en fechas próximas a terminar su curso, lo que impidió la realización de pruebas con un número suficiente de participantes. Sin embargo, en un principio se desarrolló un pequeño script de Python para Moodle localmente, con la intención de facilitar su posible implementación futura.

Otro desafío importante fue la limitación de recursos computacionales disponibles para el entrenamiento de los modelos involucrados, específicamente para IDS. El procesamiento de datos y el entrenamiento de múltiples modelos interpretables y no interpretables requería una capacidad de cómputo considerable. Estas limitaciones afectaron tanto la velocidad del desarrollo como la posibilidad de realizar experimentos más amplios y exhaustivos.

5.2. Implementación del Cuestionario en Moodle

Aunque no fue posible implementar el cuestionario en el entorno de Moodle de la Universidad, se desarrolló un script en Moodle localmente para su posible implementación en el futuro. Moodle fue elegido por varias razones:

- Plataforma Robusta: Moodle ofrece una plataforma robusta y bien documentada para la administración de cuestionarios, lo que facilita el desarrollo e implemen-

tación.

- Escalabilidad: Permite la recopilación y gestión eficiente de grandes volúmenes de datos, lo cual es crucial para estudios con numerosos participantes.
- Accesibilidad: La plataforma es accesible para los participantes desde cualquier dispositivo con conexión a internet, aumentando la probabilidad de participación.

5.3. Diseño del Experimento

El experimento fue diseñado para evaluar la interpretabilidad de los modelos Decision Tree e Interpretable Decision Sets utilizando el dataset de matemáticas del UCI Machine Learning Repository. Para evaluar la interpretabilidad, se desarrolló un cuestionario para su futura implementación que los participantes deberán completar, proporcionando sus percepciones sobre la interpretabilidad de cada modelo.

- Prueba Estadística: La prueba t se puede aplicará a los datos del cuestionario para determinar si hay una diferencia significativa en la interpretabilidad percibida entre los diferentes modelos utilizados. Esto se hará comparando las medias de las calificaciones de interpretabilidad entre los modelos. Esta propuesta se basa en las siguientes razones:
  - Evaluación de Diferencias de Medias: La prueba t es adecuada para comparar las medias de dos grupos, en este caso, la interpretabilidad percibida de dos modelos diferentes (en nuestro caso, Decision Tree e Interpretable Decision Sets).
  - Simplicidad y Eficiencia: La prueba t es relativamente sencilla de implementar y entender, lo que facilita su aplicación y la interpretación de los resultados.
- Hipótesis Nula: La hipótesis nula del experimento es que no hay diferencia significativa en la interpretabilidad percibida entre los diferentes modelos de inteligencia artificial evaluados. Esto se formuló para probar si la percepción de interpretabilidad es independiente del modelo utilizado.
- Tamaño de la Muestra: Para asegurar la validez estadística del experimento, se decidió utilizar el mismo tamaño de muestra que Lakkaraju, es decir, 47 participantes. Esta decisión se basó en las siguientes razones:
  - Utilizar el mismo tamaño de muestra permite una comparabilidad directa con el estudio de Lakkaraju, lo cual es relevante para validar y contrastar los resultados obtenidos en este TFM.
  - Dada la disponibilidad limitada de participantes, un tamaño de muestra de 47 estudiantes es factible y suficiente para obtener resultados estadísticamente significativos dentro del contexto de este estudio.

5.4. Cuestionario Utilizado en el Experimento

El cuestionario diseñado incluyó los siguientes tipos de preguntas para evaluar la interpretabilidad de los modelos:

- Preguntas de Exactitud: Se pidió a los participantes que realizaran predicciones basadas en las observaciones proporcionadas y compararan las predicciones de ambos modelos (Decision Tree e IDS).
- Preguntas de Detección de Error: Se presentaron observaciones y predicciones realizadas por los modelos, y se pidió a los participantes que evaluaran si las predicciones eran correctas.
- Métricas de Interpretabilidad Indirectas: Se incluyó el cuestionario NASA-TLX para evaluar la carga cognitiva y se midió el tiempo utilizado por los participantes para responder cada pregunta.

El cuestionario consistió en un total de 20 preguntas distribuidas de la siguiente manera:

- 12 Preguntas de Exactitud:
  - 6 preguntas resueltas por ambos modelos (DT e IDS).
  - 3 preguntas ambiguas para evaluar la capacidad de manejo de incertidumbre.
  - 3 preguntas exclusivas para cada modelo para destacar diferencias en la estructura y enfoque.
- 8 Preguntas de Detección de Error:
  - 4 preguntas resueltas por ambos modelos.
  - 2 preguntas ambiguas para evaluar cómo los modelos manejan la incertidumbre y errores potenciales.
  - 2 preguntas exclusivas para cada modelo.

Cada sección del cuestionario fue diseñada para obtener una visión integral de la interpretabilidad y precisión de los modelos desde diferentes ángulos, proporcionando así una evaluación completa y detallada.

5.5. Conclusiones

Esta experiencia resalta la importancia de disponer de recursos computacionales adecuados y una planificación detallada para la preparación de experimentos que involucren participación humana. Asimismo, la creación de soluciones personalizadas, como el desarrollo propio del algoritmo IDS, que puede ser necesaria cuando las herramientas disponibles no cumplen con los requisitos específicos del proyecto.

A pesar de los desafíos mencionados, se lograron objetivos importantes como el desarrollo del cuestionario para evaluar la interpretabilidad. Además, se propone una implementación del cuestionario en la plataforma Moodle para su futura aplicación en la Universidad Politécnica de Madrid.

El código de este proyecto se encuentra en el siguiente repositorio:

https://github.com/adrian-vargas/MasterThesis-IDS

Bibliografía

[1] Himabindu Lakkaraju, Stephen H. Bach y Jure Leskovec. «Interpretable Decision Sets: A Joint Framework for Description and Prediction». En: *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. San Francisco, California, USA: Association for Computing Machinery, 2016, págs. 1675-1684. ISBN: 9781450342322. DOI: 10.1145/2939672.2939874. URL: https://doi.org/10.1145/2939672.2939874.

[2] P. Cortez y A. M. Gonçalves Silva. «Using data mining to predict secondary school student performance». En: 2008. URL: https://api.semanticscholar.org/CorpusID:16621299.

[3] W Samek. *Explainable AI: interpreting, explaining and visualizing deep learning*. Springer Nature, 2019.

[4] Amnesty International. *El escándalo de los subsidios para el cuidado infantil en Países Bajos, una alerta urgente para prohibir los algoritmos racistas*. Accessed: 2024-08-28. 2021. URL: https://www.amnesty.org/es/latest/news/2021/10/xenophobic-machines-dutch-child-benefit-scandal/.

[5] BBC News Mundo. *"Le arruinaron la vida a gente inocente": el escándalo que hizo dimitir en bloque al gobierno de Países Bajos*. Accessed: 2024-08-28. 2021. URL: https://www.bbc.com/mundo/noticias-internacional-55683795.

[6] Euronews. *El escándalo por la discriminación racial en las ayudas familiares cerca al Gobierno de Rutte*. https://es.euronews.com/2021/01/13/el-escandalo-por-la-discriminacion-racial-en-las-ayudas-familiares-cerca-al-gobierno-de-ru. Último acceso: 28 de agosto de 2024. 2021.

[7] BBC Mundo. *¿Cómo en Estados Unidos las matemáticas te pueden meter en prisión?* Último acceso: agosto 28, 2024. 2016. URL: https://www.bbc.com/mundo/noticias-37679463.

[8] Angwin, Julia and Larson, Jeff and Mattu, Surya and Kirchner, Lauren. *Machine Bias*. https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing. Último acceso: 28 de agosto de 2024. 2016.

[9] Finale Doshi-Velez y Been Kim. «Towards a rigorous science of interpretable machine learning». En: *arXiv preprint arXiv:1702.08608* (2017).

[10] David Gunning et al. «DARPA's explainable AI (XAI) program: A retrospective». En: *Applied AI Letters* 2.4 (2021), e61.

[11] W James Murdoch et al. «Interpretable machine learning: definitions, methods, and applications». En: *arXiv preprint arXiv:1901.04592* (2019).

[12] Harmanpreet Kaur et al. «Interpreting Interpretability: Understanding Data Scientists' Use of Interpretability Tools for Machine Learning». En: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. CHI '20. New

York, NY, USA: Association for Computing Machinery, 2020, págs. 1-14. ISBN: 9781450367080.

[13] Leilani H Gilpin et al. «Explaining explanations: An overview of interpretability of machine learning». En: *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*. IEEE. 2018, págs. 80-89.

[14] Cynthia Rudin. «Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead». En: *Nature Machine Intelligence* (2019).

[15] Sunil L Kukreja, Johan Löfberg y Martin J Brenner. «A least absolute shrinkage and selection operator (LASSO) for nonlinear system identification». En: *IFAC proceedings volumes* 39.1 (2006), págs. 814-819.

[16] Trevor J Hastie. «Generalized additive models». En: *Statistical models in S*. Routledge, 2017, págs. 249-307.

[17] Ilia Rushkin. «Optimizing the Ptolemaic Model of Planetary and Solar Motion». En: *arXiv preprint arXiv:1502.01967* (2015).

[18] J Nathan Kutz y Steven L Brunton. «Parsimony as the ultimate regularizer for physics-informed machine learning». En: *Nonlinear Dynamics* 107.3 (2022), págs. 1801-1817.

[19] Richard O. Duda, Peter E. Hart y David G. Stork. *Pattern Classification*. 2nd Edition. Wiley, 2000.

[20] Franciso Herrera et al. «An overview on subgroup discovery: foundations and applications». En: *Knowledge and Information Systems* 29.3 (2010).

[21] Stefan Wrobel. «Relational Data Mining». En: Springer, 2001. Cap. Inductive Logic Programming for Knowledge Discovery in Databases.

[22] Harsha Nori et al. «Interpretml: A unified framework for machine learning interpretability». En: *arXiv preprint arXiv:1909.09223* (2019).

[23] Ibomoiye Domor Mienye y Nobert Jere. «A Survey of Decision Trees: Concepts, Algorithms, and Applications». En: *IEEE Access* (2024).

[24] Han Liu, Alexander Gegov y Mihaela Cocea. *Rule Based Systems for Big Data*. Springer, 2015.

[25] Benjamin Letham et al. «Interpretable Classifiers Using Rules and Bayesian Analysis: Building a Better Stroke Prediction Model». En: *The Annals of Applied Statistics* 9.3 (2015).

[26] Ronald L Rivest. «Learning decision lists». En: *Machine learning* 2 (1987), págs. 229-246.

[27] Alexey Ignatiev et al. «A SAT-Based Approach to Learn Explainable Decision Sets». En: *Automated Reasoning*. Ed. por Didier Galmiche, Stephan Schulz y Roberto Sebastiani. Cham: Springer International Publishing, 2018, págs. 627-645. ISBN: 978-3-319-94205-6.

[28] Leo Breiman et al. *Classification and Regression Trees*. Chapman y Hall, 1984.

[29] Caglar Aytekin. «Neural networks are decision trees». En: *arXiv preprint arXiv:2210.05189* (2022).

[30] Forough Poursabzi-Sangdeh et al. «Manipulating and Measuring Model Interpretability». En: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. CHI '21. New York, NY, USA: Association for Computing Machinery, 2021. ISBN: 9781450380966.

[31] Zachary C. Lipton. *The mythos of model interpretability*. Disponible en arXiv. 2016. arXiv: 1606.03490 [stat.ML]. URL: https://arxiv.org/abs/1606.03490.

[32] Marco Tulio Ribeiro, Sameer Singh y Carlos Guestrin. «"Why should i trust you?.Explaining the predictions of any classifier». En: *Proceedings of the 22nd*

ACM SIGKDD international conference on knowledge discovery and data mining. 2016, págs. 1135-1144.

[33] Bojan Mihaljevic y Esteban García. *Notas del curso XAI. Universidad Politécnica de Madrid*. 2024.

[34] Scott Lundberg y Su-In Lee. *A Unified Approach to Interpreting Model Predictions*. 2017. arXiv: 1705.07874 [cs.AI]. URL: https://arxiv.org/abs/1705.07874.

[35] Jimmy Lin et al. «Generalized and Scalable Optimal Sparse Decision Trees». En: *Proceedings of the 37th International Conference on Machine Learning*. Ed. por Hal Daumé III y Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, 13–18 Jul de 2020, págs. 6150-6160.

[36] Xiyang Hu, Cynthia Rudin y Margo Seltzer. «Optimal Sparse Decision Trees». En: *Advances in Neural Information Processing Systems*. Ed. por H. Wallach et al. Vol. 32. Curran Associates, Inc., 2019.

[37] Sandra G. Hart y Lowell E. Staveland. «Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research». En: *Human Mental Workload*. Ed. por Peter A. Hancock y Najmedin Meshkati. Vol. 52. Advances in Psychology. North-Holland, 1988, págs. 139-183. DOI: https://doi.org/10.1016/S0166-4115(08)62386-9.

[38] Microsoft Developer. *The Science Behind InterpretML: Explainable Boosting Machine*. Accedido el: 20 de septiembre de 2024. 2020. URL: https://www.youtube.com/watch?v=MREiHgHgl0k.

[39] Benjamin Bengfort et al. *Yellowbrick*. Ver. 0.9.1. 14 de nov. de 2018. DOI: 10.5281/zenodo.1206264. URL: http://www.scikit-yb.org/en/latest/.

[40] Marco Tulio Ribeiro, Sameer Singh y Carlos Guestrin. «Anchors: High-precision model-agnostic explanations». En: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1. 2018.

[41] Ramaravind K Mothilal, Amit Sharma y Chenhao Tan. «Explaining machine learning classifiers through diverse counterfactual explanations». En: *Proceedings of the 2020 conference on fairness, accountability, and transparency*. 2020, págs. 607-617.

[42] Adrian Vargas Rangel. *Una herramienta para la evaluación de la interpretabilidad*. Último acceso: septiembre 2024. 2024. URL: https://colab.research.google.com/drive/1_VD8UgamgW4jdyxzdepSxo6fJLQLNkL1?usp=sharing.

[43] Paulo Cortez. «Student performance». En: *UCI machine learning repository* (2014).

Apéndice A

Anexo

A.1. Cuestionario para Evaluar la Interpretabilidad de los Modelos

El cuestionario incluye los siguientes tipos de preguntas:

- Preguntas de Exactitud: Pedir al usuario que realice una predicción basada en el modelo.
- Preguntas de Detección de Error: Presentar al usuario una observación y una predicción realizada por el modelo y preguntar si es correcta.
- Métricas de Interpretabilidad Indirectas: Se incluirán métricas como el cuestionario NASA-TLX para evaluar la carga cognitiva y se medirá el tiempo utilizado por el usuario para responder cada pregunta.

Para garantizar una evaluación completa y detallada, se propone un cuestionario con un total de 20 preguntas:

- 12 Preguntas de Exactitud:
  - 6 preguntas resueltas por ambos modelos (DT e IDS).
  - 3 preguntas ambiguas para evaluar la capacidad de manejo de incertidumbre.
  - 3 preguntas exclusivas para cada modelo para destacar diferencias en la estructura y enfoque.
- 8 Preguntas de Detección de Error:
  - 4 preguntas resueltas por ambos modelos.
  - 2 preguntas ambiguas para evaluar cómo los modelos manejan la incertidumbre y errores potenciales.
  - 2 preguntas exclusivas para cada modelo.

A.1.1. Preguntas de Exactitud

A.1.1.1. Preguntas resueltas por ambos modelos

- Observación: `absences = 1, failures = 1, studytime = 2, age = 18`
  Predicción:
  - a) Aprobado
  - b) No aprobado
  - c) No estoy seguro

  Modelo: Ambos modelos predicen No Aprobado
- Observación: `absences = 4, failures = 0, studytime = 3, age = 16`
  Predicción:
  - a) Aprobado
  - b) No aprobado
  - c) No estoy seguro

  Modelo: Ambos modelos predicen No Aprobado
- Observación: `absences = 3, failures = 1, studytime = 1, age = 15`
  Predicción:
  - a) Aprobado
  - b) No aprobado
  - c) No estoy seguro

  Modelo: Ambos modelos predicen Aprobado
- Observación: `absences = 0, failures = 0, studytime = 1, age = 18`
  Predicción:
  - a) Aprobado
  - b) No aprobado
  - c) No estoy seguro

  Modelo: Ambos modelos predicen Aprobado
- Observación: `absences = 1, failures = 0, studytime = 2, age = 15`
  Predicción:
  - a) Aprobado
  - b) No aprobado
  - c) No estoy seguro

  Modelo: Ambos modelos predicen Aprobado
- Observación: `absences = 0, failures = 1, studytime = 3, age = 18`
  Predicción:
  - a) Aprobado

- b) No aprobado
- c) No estoy seguro

Modelo: Ambos modelos predicen No Aprobado

A.1.1.2. Preguntas ambiguas

- Observación: `absences = 3, failures = 1, studytime = 2, age = 17`
  Predicción:
  - a) Aprobado
  - b) No aprobado
  - c) No estoy seguro

  Modelo: La combinación de valores puede no estar claramente definida en las reglas de los modelos generando incertidumbre.
- Observación: `absences = 2, failures = 2, studytime = 3, age = 16`
  Predicción:
  - a) Aprobado
  - b) No aprobado
  - c) No estoy seguro

  Modelo: Los modelos pueden no tener reglas claras para `failures = 2` y `studytime = 3` simultáneamente lo que genera ambigüedad.
- Observación: `absences = 1, failures = 0, studytime = 4, age = 18`
  Predicción:
  - a) Aprobado
  - b) No aprobado
  - c) No estoy seguro

  Modelo: La alta variabilidad en `studytime (4)` podría no estar claramente definida en las reglas de los modelos causando incertidumbre.

A.1.1.3. Preguntas exclusivas para cada modelo

- Observación: `absences = 2, failures = 1, studytime = 2, age = 16`
  Predicción:
  - a) Aprobado
  - b) No aprobado
  - c) No estoy seguro

  Modelo: DT predice Aprobado; IDS predice No Aprobado.
- Observación: `absences = 5, failures = 2, studytime = 2, age = 17`
  Predicción:
  - a) Aprobado

- b) No aprobado
- c) No estoy seguro

Modelo: DT predice No Aprobado; IDS no tiene reglas específicas.

- Observación: `absences = 1, failures = 0, studytime = 2, age = 16`
  Predicción:
  - a) Aprobado
  - b) No aprobado
  - c) No estoy seguro

  Modelo: IDS predice No Aprobado; DT predice Aprobado.

A.1.2. Preguntas de Detección de Error

A.1.2.1. Preguntas resueltas por ambos modelos

- Observación: `absences = 2, failures = 0, studytime = 1, age = 16`
  Predicción del modelo: No aprobado
  ¿Es correcta la predicción?:
  - a) Sí
  - b) No
  - c) No estoy seguro

  Modelo: Ambos modelos predicen No Aprobado
- Observación: `absences = 0, failures = 0, studytime = 3, age = 17`
  Predicción del modelo: Aprobado
  ¿Es correcta la predicción?:
  - a) Sí
  - b) No
  - c) No estoy seguro

  Modelo: Ambos modelos predicen Aprobado
- Observación: `absences = 1, failures = 1, studytime = 2, age = 18`
  Predicción del modelo: No aprobado
  ¿Es correcta la predicción?:
  - a) Sí
  - b) No
  - c) No estoy seguro

  Modelo: Ambos modelos predicen No Aprobado
- Observación: `absences = 4, failures = 0, studytime = 3, age = 16`
  Predicción del modelo: No aprobado
  ¿Es correcta la predicción?:

- a) Sí
- b) No
- c) No estoy seguro

Modelo: Ambos modelos predicen No Aprobado

A.1.2.2. Preguntas ambiguas

- Observación: `absences = 4, failures = 1, studytime = 2, age = 17`
  Predicción del modelo: No aprobado
  ¿Es correcta la predicción?:
  - a) Sí
  - b) No
  - c) No estoy seguro

  Modelo: La combinación de ausencias moderadas y fallos podría no estar claramente cubierta por las reglas.
- Observación: `absences = 0, failures = 2, studytime = 3, age = 15`
  Predicción del modelo: No aprobado
  ¿Es correcta la predicción?:
  - a) Sí
  - b) No
  - c) No estoy seguro

  Modelo: La ausencia de fallos pero con alto número de fallos previos y tiempo de estudio moderado puede no estar claramente definida en las reglas.
- Observación: `absences = 1, failures = 1, studytime = 2, age = 16`
  Predicción del modelo: No aprobado
  ¿Es correcta la predicción?:
  - a) Sí
  - b) No
  - c) No estoy seguro

  Modelo: La combinación de valores puede no estar claramente definida en las reglas de los modelos generando incertidumbre.

A.1.2.3. Preguntas exclusivas para cada modelo

- Observación: `absences = 3, failures = 0, studytime = 1, age = 18`
  Predicción del modelo: No aprobado
  ¿Es correcta la predicción?:
  - a) Sí
  - b) No
  - c) No estoy seguro

Modelo: DT predice No Aprobado; IDS no tiene reglas específicas.

- Observación: `absences = 2, failures = 1, studytime = 2, age = 16`
  Predicción del modelo: No aprobado
  ¿Es correcta la predicción?:
  - a) Sí
  - b) No
  - c) No estoy seguro

  Modelo: DT predice Aprobado; IDS predice No Aprobado.

- Observación: `absences = 5, failures = 2, studytime = 2, age = 17`
  Predicción del modelo: No aprobado
  ¿Es correcta la predicción?:
  - a) Sí
  - b) No
  - c) No estoy seguro

  Modelo: DT predice No Aprobado; IDS no tiene reglas específicas.

- Observación: `absences = 0, failures = 0, studytime = 1, age = 16`
  Predicción del modelo: No aprobado
  ¿Es correcta la predicción?:
  - a) Sí
  - b) No
  - c) No estoy seguro

  Modelo: IDS predice No Aprobado; DT predice Aprobado.

A.2. Implementación del Cuestionario en Moodle

Aunque no fue posible implementar el cuestionario completo en el entorno de Moodle de la universidad debido al acercamiento del fin de curso y a restricciones de políticas internas, se desarrolló un script en Moodle ya que no se encontraron soluciones en GitHub ni en Moodle para que el cálculo se realizara de forma automatizada y se encontró que Moodle ofrece la posibilidad de revisar el intento de cada estudiante mediante una tabla debajo de cada pregunta que brinda el historial de la interacción del estudiante con esa pregunta y que incluye marcas de tiempo para cada acción, lo que lo convierte en una opción accesible frente a otras soluciones de pago.

Con la información que nos proporciona Mooodle, se puede calcular manualmente el tiempo que el usuario tarda en responder cada pregunta.

A continuación, se describe el proceso de uso del script de webscrapping para la extracción y cálculo de los tiempos de respuesta:

1. Descargar Python 3.12.0 o superior
2. Descargar la carpeta `web_scraping_response_times`

Figura A.1: Consulta de resultados de cuestionario

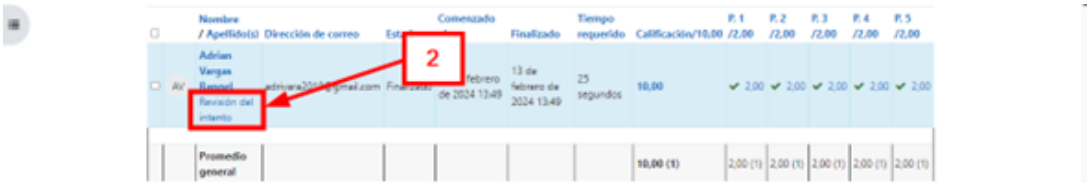


Figura A.2: Revisión del intento

3. Guardar como html la revisión del intento de cada participante dentro de la carpeta descargada
4. Abrir una terminal en la carpeta descargada
5. Ejecutar la línea: `python extractor.py`
6. Se creará el archivo de Excel "`all_response_times`" con los tiempos de respuesta calculados en segundos

Las pruebas con las preguntas dummy confirmaron que el sistema de Moodle puede ser utilizado con el script de webscrapping adjunto en el repositorio de github de esta memoria para extraer la hora de inicio y fin de cada pregunta del cuestionario y calcular el tiempo de respuesta del participante, lo que facilita la futura implementación del cuestionario completo.

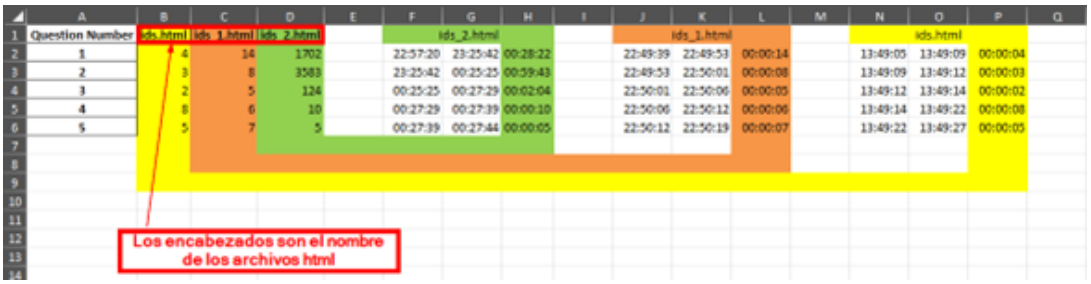


Figura A.3: EJemplo de registro de tiempo de respuesta por pregunta. Cada columna de B a D representa el cuestionario de un participante individual

Este proceso asegura que el cuestionario pueda ser desplegado en el entorno de Moodle de la universidad en futuras implementaciones, proporcionando una herramienta adecuada para evaluar la interpretabilidad de los modelos de inteligencia artificial.