Universidad Politécnica
de Madrid

Escuela Técnica Superior de Ingenieros Informáticos

Máster Universitario en Inteligencia Artificial

Trabajo Fin de Máster

Generación de Muestras Contrafactuales Basada en la Importancia de las Características

Autor(a): Ismael Beviá Ballesteros
Tutor(a): Esteban García Cuesta

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: Generación de Muestras Contrafactuales Basada en la Importancia de las Características

Julio 2024

Autor(a): Ismael Beviá Ballesteros
Tutor(a): Esteban García Cuesta
Departamento de Inteligencia Artificial
ETSI Informáticos
Universidad Politécnica de Madrid

Resumen

Las explicaciones contrafactuales tratan de describir un posible escenario alternativo que no ha tenido lugar y contradice los hechos mostrando cómo habría sido el resultado bajo condiciones distintas, un aspecto esencial para que los usuarios entiendan y actúen sobre las predicciones. Al formular este tipo de explicaciones, que se presentan como contraejemplos de la clase objetivo elaborados a partir de una instancia inicial, es crucial considerar la alta probabilidad de generar muestras poco útiles o incluso imposibles. Esto puede ocurrir por ejemplo si se proponen modificaciones sobre demasiadas características, un cambio desproporcional al necesario o cuando el resultado es inalcanzable desde la instancia sometida a estudio.

Con el fin de producir explicaciones adecuadas ante desafíos como los mencionados, en el presente trabajo se propone el diseño y desarrollo de dos propuestas de algoritmos generadores de contrafactuales independientes del modelo con los que se ha buscado abordar los aspectos más deseables al proporcionar este tipo de explicaciones y obtener resultados competitivos. Ambos procesos se basan en una estrategia de mejora centrada en otra forma de explicación ampliamente adoptada: la relevancia de las características para guiar el cambio mínimo, priorizando su modificación.

La eficacia de estos métodos ha sido demostrada sobre cinco conjuntos de datos, evaluando tanto su coherencia lógica en las explicaciones como su rendimiento práctico. El primer algoritmo es un mecanismo de formulación de contrafactuales sintéticos que sigue una estrategia híbrida, combinando enfoques basados en optimización y búsqueda heurística. Por otro lado, el segundo algoritmo sigue la idea de construir un camino incremental de ejemplos para guiar y asegurar la factibilidad de un cambio hacia una situación contrafactual. Finalmente, también se ha llevado a cabo un estudio comparativo sobre la primera propuesta con varias técnicas relevantes actuales, consiguiendo superarlas en diversos campos.

Abstract

Counterfactual explanations aim to describe a potential alternative scenario that did not occur, contradicting the initial facts by showing how the outcome would have been under different conditions. This is essential for users to understand and act upon predictions. When formulating such explanations, presented as counterexamples to the target class, derived from an initial instance, it is crucial to consider the high probability of generating samples that are either unhelpful or even impossible. This may happen, for example, if modifications are proposed on too many features, if the change is disproportionate to what is necessary, or if the result is unattainable from the instance under study.

To produce suitable explanations in the face of challenges like those mentioned, this work proposes the design and development of two algorithmic proposals for generating counterfactuals independent of the model. These proposals aim to address the most desirable aspects when providing such explanations and to achieve competitive results. Both processes are based on an improvement strategy focused on another widely adopted form of explanation: the relevance of features to guide minimal change, prioritizing their modification.

The effectiveness of these methods has been demonstrated on five datasets, evaluating both their logical coherence in explanations and their practical performance. The first algorithm is a mechanism for formulating synthetic counterfactuals that follows a hybrid strategy, combining optimization-based approaches with heuristic search. On the other hand, the second algorithm follows the idea of constructing an incremental path of examples to guide and ensure the feasibility of a change towards a counterfactual situation. Finally, a comparative study has been conducted on the first proposal with various relevant current techniques, surpassing them in diverse fields.

Tabla de contenidos

Capítulo 1

Introducción

En los últimos años, la Inteligencia Artificial ha experimentado un impresionante avance lo cual ha llevado a que los algoritmos actuales sean sumamente poderosos y se utilicen en una amplia variedad de aplicaciones. Sin embargo, a medida que el mundo se adapta y se vuelve más dependiente de esta tecnología, también aumenta la necesidad de entender el funcionamiento interno detrás de estas herramientas. Aunque cada vez ofrecen mejores resultados, también se están volviendo más opacas.

Con el fin de abordar esta preocupación emerge la Inteligencia Artificial Explicable, un área de investigación que busca resultados comprensibles y transparentes para los usuarios sin afectar a la precisión de los sistemas. Por ejemplo, las extremadamente populares redes neuronales artificiales son reconocidas por su eficacia y capacidad de obtener los mejores resultados en multitud de tareas pero, presentan una estructura y un funcionamiento interno muy complejo. En definitiva, ahora más que nunca se destaca la necesidad de dar explicaciones y justificaciones detrás de las decisiones de los algoritmos. Tanto es así que en 2018 entró en vigor la Ley del derecho a la explicación como parte del Reglamente General de Protección de Datos y además, en 2024 se ha aprobado la Ley de Inteligencia Artificial propuesta por la Comisión Europea, la cual se enfoca, entre otros aspectos, en la explicabilidad de los sistemas.

Dentro de la Inteligencia Artificial Explicable existen cada vez más tendencias. Entre ellas, se destaca la estimación como pesos de la importancia de las características de entrada, como pesos, que cuantifican la influencia en la predicción, Esta práctica ampliamente popular y estudiada, no solo se utiliza para justificar la toma de decisiones de un modelo, sino también para alimentar otros procesos.

Por otra parte, una técnica más reciente introducida en 2018 por Wachter et al. [1] es la generación de contrafactuales. Esta técnica se basa en formar ejemplos que presentan ajustes mínimos sobre una instancia sometida a estudio con el fin de simular un escenario donde el modelo clasificaría la muestra con una predicción diferente. El interés en esta última tendencia se fundamenta en su flexibilidad con el método subyacente y en su capacidad para ser fácilmente entendible e informativa para los seres humanos ya que resaltan información contextualmente relevante, lo que las hace muy similares a las explicaciones humanas.

Se puede señalar una clara relación entre ambos tipos de explicaciones. Mientras que la primera destaca cuáles son las características más importantes para el cambio, la segunda formula ejemplos de una categoría diferente ajustando las características.

Esta afirmación motiva el trabajo realizado en este documento, donde se presenta el estudio y la aplicación de ambas técnicas combinadas en dos propuestas distintas de generación de contrafactuales.

1.1. Objetivos

El objetivo principal del presente trabajo es diseñar y desarrollar propuestas de formulación de contrafactuales basadas en la importancia de las características y que aborden los aspectos deseables de este tipo de explicaciones. Más específicamente, se busca construir algoritmos que generen ejemplos contrafactuales en problemas de datos tabulares que consigan resultados que sean competitivos con las herramientas empleadas en la actualidad y planteen una buena estrategia. Para lograrlo, es crucial analizar la metodología del estado del arte, proponer y validar un nuevo enfoque y abrir nuevas vías para futuras investigaciones. Este objetivo general se puede dividir en los siguientes componentes:

- **Análisis de las explicaciones y revisión de técnicas**: Estudiar la explicabilidad y, más concretamente, las necesidades y principales líneas de investigación en el desarrollo de contrafactuales. También, conlleva hacer una revisión de la literatura con el fin de conocer los sistemas que siguen un enfoque similar al abordar esta problemática.
- **Diseño e implementación de propuestas**: Plantear y desarrollar herramientas adecuadas y novedosas que aborden las necesidades destacadas.
- **Evaluación de las explicaciones**: Realizar una demostración lógica y técnica sobre la resolución de problemas, así como una comparativa con trabajos populares disponibles en la actualidad.

1.2. Estructura del documento

El trabajo realizado se ha organizado en seis capítulos incluyendo este primero que pretende ofrecer una visión general del problema, justificar la importancia de la investigación y establecer los objetivos. En el segundo y tercer capítulo se abordan los aspectos más teóricos del trabajo, contextualizando el tema, desarrollando puntos fundamentales de este tipo de explicaciones y, de seguido, exponiendo una revisión del trabajo previo sobre este enfoque.

Luego, en base el conocimiento previo, en el capítulo 4 se detalla el diseño y la metodología de las dos propuestas elaboradas, así como su motivación y, en el capítulo 5, se detalla el entorno de experimentación y los resultados obtenidos en las pruebas para validar las propuestas, evaluar su eficacia y garantizar que se han cumplido los objetivos. Finalmente, en el capitulo 6 se comentan las conclusiones finales del trabajo realizado y se plantean posibles vías de trabajo futuro.

Capítulo 2

Fundamentos Teóricos

En esta sección se va a profundizar en varios conceptos relacionados con el trabajo con el fin de dar contexto al mismo y describir aspectos fundamentales y destacables para su comprensión.

2.1. Introducción a la Inteligencia Artificial Explicable

Nos hemos acostumbrado a que la Inteligencia Artificial (IA) tome decisiones por nosotros como por ejemplo sobre recomendaciones y sugerencias de películas o fuentes de información. Estos casos expuestos puede que no requieran de una explicación para el usuario, pero, en decisiones criticas como el diagnostico, si es importante entender las razones detrás de la resolución. En general, aunque los algoritmos actuales son poderosos en términos de resultados y predicciones, sufren de opacidad por lo que es difícil entender su trabajo interno. Para abordar esta problemática surge la Inteligencia Artificial Explicable (IAX), un término utilizado por primera vez por Van Lent et al. en 2004 [2] aunque se considera que existe desde la aparición de los sistemas expertos [3].

La IAX es un campo de investigación que pretende hacer los resultados de la IA más comprensibles tendiendo a referirse a las iniciativas y los esfuerzos realizados sobre la transparencia y la confianza en la IA [4]. El énfasis actual sobre este tema de investigación es el resultado de hacer uso de este tipo de sistemas en procesos de toma de decisiones críticas por lo que la presión social, ética y legal exige que sean capaces de ser explicados y comprensibles. Por lo general, la IAX se centra en desmitificar las cajas negras que son aquellos modelos donde los funcionamientos internos no son fácilmente accesibles y, por lo tanto, no son transparentes. Un ejemplo de estos algoritmos opacos son las actualmente muy exitosas redes neuronales que se están empleando en campos como la atención médica o la economía donde la transparencia y explicabilidad es reconocida como críticamente importante. En definitiva, la IAX pretende a ayudar a comprender estos sistemas sin afectar a la precisión, aunque a menudo se ha de buscar un equilibrio entre esta y la interpretabilidad. La Figura 2.1 [5] muestra en una representación gráfica la relación entre estos aspectos en algunos de los modelos más exitosos donde se demuestra que los más precisos generalmente no son muy explicables y los modelos más interpretables suelen ser los menos precisos.

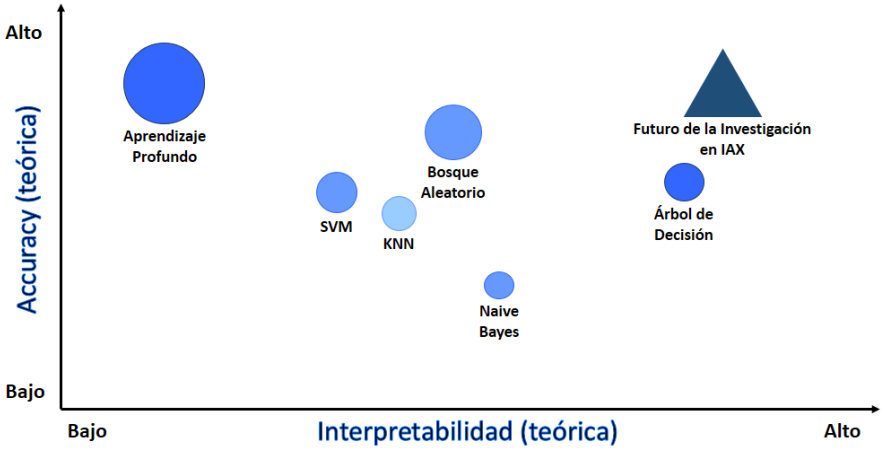


Figura 2.1: Accuracy frente interpretabilidad en diferentes modelos de aprendizaje automático [5].

En el trabajo desarrollado por Phillips et al. [6] se han definido cuatro principios que deben ser respetados por un sistema para ser considerado explicable:

- Explicación: presentar evidencia o razonamientos sobre las salidas o procesos.
- Significativo: las explicaciones deben ser comprensibles por los usuarios.
- Exactitud: las explicaciones deben reflejar con precisión la razón para generar una salida o el proceso de un sistema.
- Límites del conocimiento: un sistema solo debe emplearse bajo las condiciones para las que fue diseñado y cuando consigue suficiente confianza en su resultado.

La necesidad de dar explicaciones surge de pretender resolver aspectos como poder justificar los resultados para demostrar así no estar sesgados, no ser discriminatorios y ser comprobables para defender las decisiones automáticas como justas y éticas [4].

Además, es importante destacar la tendencia de desarrollo de legislación en este ámbito la cual se ha preocupado en los últimos años por que la IA proporcione una justificación exigiendo que sea explicativa bajo la ley del "derecho a la explicación" incluida en el Reglamento General de Protección de Datos (GDPR) que entró en vigor en la Unión Europea el 25 de mayo de 2018 [7]. Asimismo, la reciente aprobación en 2024 de la Ley de Inteligencia Artificial propuesta por la Comisión Europea, refuerza este enfoque como uno de sus aspectos principales. Por otra parte, comprender más sobre un sistema proporciona mayor visibilidad a los defectos de este por lo que se puede utilizar para "arreglarlo" y mejorarlo de forma continúa consiguiendo así solucionar los sesgos del modelo y promover la equidad [8]. En los casos que el modelo de aprendizaje automático se ofrece como producto, la interpretabilidad de las decisiones suele ser un aspecto determinante.

Los sistemas modernos de aprendizaje automático trabajan a partir de las observaciones y crean representaciones sobre esos datos considerando intrínsicamente interacciones de alto grado entre las características. Estas extraordinarias capacidades han revolucionado la elaboración de herramientas predictivas, pero, hacen que

los resultados sean aún más difíciles de entender debido a la estructura de los algoritmos y la forma en la que funcionan. Por ejemplo, las redes neuronales artificiales están compuestas de capas de neuronas artificiales donde cada una aplica una función de activación a la suma ponderada de sus entradas, introduciendo no linealidad. Esta arquitectura logra producir predicciones de muy alto nivel, pero múltiples capas con interacciones no lineales implican una estructura también muy complicada. Además, otra problemática de estos algoritmos complejos erradica en la robustez que se refiere a varias medidas distintas pero relacionadas de cuanto pueden cambiar las explicaciones bajo modificaciones en el sistema. Un método de explicación robusto es aquel en el que las explicaciones permanecerán similares en ciertos escenarios.

Aparte de estas dificultades más técnicas, también existen diversos estudios que intentan abordar la interpretación desde un enfoque que mantenga a los humanos involucrados por lo que se recomienda que los sistemas de IA se adhieran a los siguientes principios [7]: generar interés y motivación en los usuarios, involucrarlos en el desarrollo garantizando así la participación y empoderamiento sobre la lógica del algoritmo, implementar técnicas colaborativas desde la experiencia de múltiples dominios para mejorar las explicaciones, evaluar los métodos de forma cualitativa y cuantitativa, valorar el uso de contrafactuales y considerar como de importantes son las explicaciones dependiendo del contexto. Tener en cuenta y profundizar en estos principios de diseño además de en el aspecto técnico es fundamental para avanzar en el estado del arte de la explicabilidad.

En definitiva, las explicaciones deben conseguir que un modelo sea expresivo con el fin de mejorar su comprensión, la confianza sobre este y promover decisiones imparciales y justas.

2.1.1. Categorización de Explicabilidad de Algoritmos

En este subapartado se van a distinguir de forma general las diferentes perspectivas de explicación siguiendo como base el marco y la taxonomía propuesta por Hassija et al. [7], representada en la Figura 2.2. En posteriores apartados se van a profundizar y explicar en detalle aquellos apartados más relevantes en el presente trabajo.

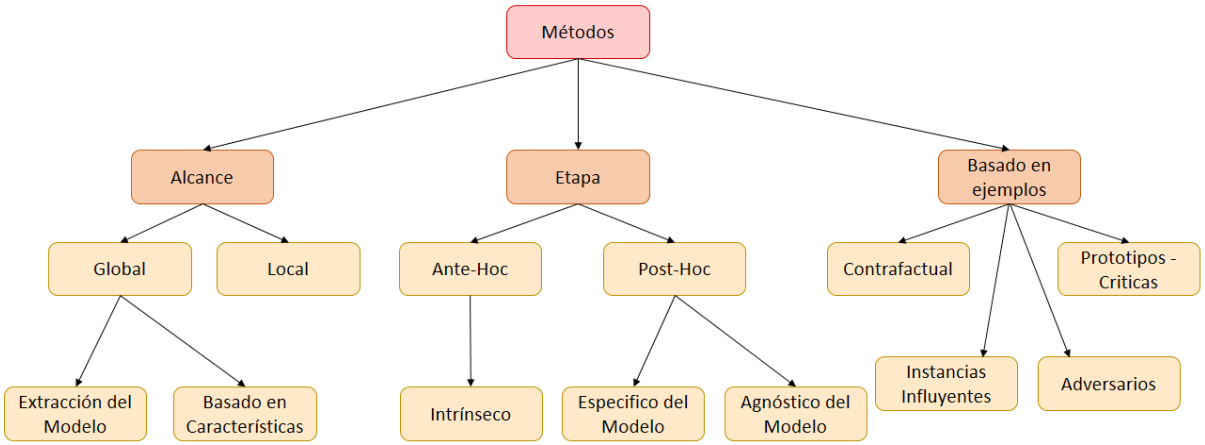


Figura 2.2: Categorización de explicabilidad [7].

Según el alcance del método, lo que se refiere al enfoque o perspectiva desde la cual se

derivan las explicaciones, este puede ser global, local o en algunos casos, extenderse a ambas.

Una perspectiva global tiene como fin dar una compresión de la lógica general y completa del modelo y es utilizada para tareas como ayudar a decisiones a gran escala o facilitar una compresión completa a un experto. Dentro de este enfoque se han diferenciado dos subclases: la extracción de modelos y los basados en características. La extracción de modelos implica formar un sustituto interpretable por sí mismo con el fin de explicar las predicciones y obtener información de otro opaco. Esto se hace frecuentemente siguiendo una estrategia de extracción de reglas que describen las relaciones entre las características de entrada y las salidas o utilizando modelos transparentes lo que implica transferir el conocimiento de uno complejo a uno más simple que sea competente en términos de predicción. Estos enfoques consiguen obtener explicaciones globales, pero al simplificar el modelo, también se ve comprometida la precisión por lo que se ha optado por investigar sobre métodos basados en la importancia de las características.

Con respecto a la interpretación local, en este enfoque se pretende dar explicaciones separadas a las predicciones dando como resultado la importancia de las características al observar su peso en la decisión. Las explicaciones formadas por los métodos locales pueden ayudar a generar confianza o a tomar decisiones para obtener resultados más deseables y suelen ser de mayor interés para un usuario final. Hasta la fecha, se han propuesto multitud de técnicas de entre las cuales se pueden destacar por ser actuales y ampliamente utilizadas Local Interpretable Model-Agnostic (LIME) [9] y los valores Shapley [10]. LIME trabaja generando modelos sustitutos simples e interpretables de alta fidelidad local que imitan el comportamiento de modelos complejos en las cercanías de un caso con el fin de aproximar y dar una explicación a la predicción. Por otro lado, los valores Shapley son una forma de calcular la contribución marginal promedio de cada característica considerando todas las coaliciones posibles para conseguir una distribución justa.

Es importante resaltar que este tipo de métodos de explicabilidad local son destacadamente utilizados en el contexto de redes neuronales.

Por otro lado, según la etapa de interpretabilidad (lo que hace referencia al momento donde se analiza el modelo) se puede distinguir entre aquellos donde se desarrolla antes de su entrenamiento (Ante Hoc) o después (Post Hoc). La interpretabilidad Ante Hoc está relacionada con la intrínseca involucrando principalmente el manejo de los datos y siendo las técnicas más utilizadas algunas prácticas tradicionales transparentes como la regresión lineal, la regresión logística, algoritmos de aprendizaje basados en reglas o K-NNs. Por otra parte, la interpretabilidad Post Hoc involucra técnicas que se aplican de forma posterior a los modelos pudiendo ser métodos específicos para un modelo o métodos agnósticos que son aplicables sobre distintos tipos de algoritmos y trabajan realizando un análisis simultaneo de la entrada y salida, aunque se limita la información obtenida sobre el modelo. En este punto, es importante destacar un principal interés en el desarrollo de metodologías Post Hoc independientes del modelo.

Por último, se distinguen los métodos basados en ejemplos hacen uso de instancias específicas de un conjunto de datos para explicar las predicciones de los modelos. Entre las técnicas más destacadas se encuentran:

1. Los prototipos, los cuales constituyen un conjunto de ejemplos seleccionados directamente de los datos que representan al resto con precisión y se utilizan para proporcionar explicaciones intuitivas sobre cómo funciona el modelo de forma general.
2. Las críticas que, por otro lado, son aquellas instancias mal representadas por los prototipos y cuestionan o señalan las deficiencias en el rendimiento del modelo. Estas son mucho más difíciles de localizar.
3. Las instancias de entrenamiento influyentes que son aquellas que tienen un peso considerable al determinar los parámetros y las decisiones del modelo. Son cruciales en la tarea de depuración y análisis de un modelo y se pueden obtener a partir de la comparar el modelo eliminando instancias.
4. Los contrafactuales que se distinguen de los prototipos en que no son ejemplos tomados directamente del conjunto de datos, sino que buscan estar formados por ajustes mínimos en los valores de las instancias de entrada con el fin de simular escenarios potenciales distintos al de la instancia de estudio. Al emplear y desarrollar esta técnica, es crucial considerar la posibilidad de que se generen opciones o resultados inadecuados o incluso imposibles en la realidad.
5. Los ejemplos adversarios, los cuales son simplemente instancias perturbadas con el fin de confundir al modelo. Son útiles para encontrar vulnerabilidades, mejorar la interpretación y formar modelos más robustos que resistan estas perturbaciones.

Como se ha comentado al introducir el punto, en los siguientes apartados se va a profundizar sobre los tipos metodologías que son de mayor interés para el desarrollo del trabajo. Estas son la importancia de las características y las explicaciones contrafácticas, dos enfoques de métodos de explicabilidad local post-hoc.

2.2. Importancia de las Características

Los métodos basados en la importancia de las características tienden a funcionar asignando un puntaje a cada característica de entrada que indica su influencia en la predicción. Este tipo de explicaciones se consideran las más populares y han sido aplicadas en una gran variedad de dominios, como las finanzas o la salud. Además, es, con diferencia, la técnica más utilizada y estudiada [11]. Por ejemplo, en un ámbito como el de la medicina no solo es importante el resultado, si no también conocer que características son las que condicionan una situación de forma explicativa y cuantificable. Esta información puede resultar útil para identificar qué aspectos se deben mejorar o identificar riesgos [12].

Estos métodos logran medir la contribución individual en un clasificador específico, sin tener en cuenta la relación entre los datos o el efecto que conlleven. Cabe destacar que la importancia de las características varía según el modelo, ya que cada sistema puede interpretar y utilizar las características de manera distinta.

Como se ha comentado al desarrollar la categorización de técnicas de explicabilidad, existen métodos de este tipo con enfoques globales que miden la importancia para todo el modelo y locales que se centran en una contribución determinada. Las características locales obtenidas como importantes para casos distintos pueden no ser las

mismas y, de la misma forma, pueden no corresponderse con las obtenidas globalmente. Para el desarrollo del presente trabajo, se han considerado las características locales y agnósticas al modelo como más relevantes.

Esta metodología se suele confundir o "mezclar" con las técnicas de selección de características y, aunque en algunos casos son utilizados para resolver un problema similar, se pueden destacar claras diferencias en su propósito y momento de aplicación [12]. La selección de características es una técnica de preprocesamiento y se refiere al proceso de identificar y descartar características que se aplica principalmente antes de entrenar un modelo mientras que, la importancia de las características, se realiza durante o después del entrenamiento para explicar el modelo aprendido.

Las funciones de explicación que extraen la importancia se pueden dividir en dos categorías [11]: las técnicas basadas en perturbaciones y las técnicas basadas en gradientes, aunque estas últimas se pueden ver como un caso particular de las anteriores con un tamaño de permutación infinitesimal. Un ejemplo basado en perturbaciones podrían ser los ya presentados valores Shapley. Por otro lado, también se pueden destacar los mapas de calor como una forma de explicar la importancia de regiones o conjuntos de características mostrando visualmente que datos están teniendo un mayor impacto en la decisión.

En definitiva, los métodos basados en la importancia de las características son una poderosa herramienta de explicabilidad que puede ser presentada directamente a los usuarios como información sobre la predicción o puede ser utilizada por los ingenieros para validar la coherencia de un modelo. Además, también pueden utilizarse como en el presente trabajo para conseguir información sobre la predicción y emplearla en otro proceso.

2.3. Explicaciones Contrafácticas

Las explicaciones contrafactuales (CFEs) son una técnica emergente de explicabilidad local introducida por Wachter et al. [1] que puede ser utilizada sobre modelos complejos como bosques aleatorios o redes neuronales. Estas explicaciones se basan en hacer un cálculo normalmente mínimo sobre un punto de los datos de entrada lo que llevaría al modelo de aprendizaje automático a clasificarlo de una manera diferente y deseada [13] proporcionando explicaciones del tipo "que pasaría si". El calculo mínimo se debe a que se busca que el caso real y el contrafáctico sea similar.

Por ejemplo, si a un cliente de un banco se le negara un préstamo, una CFE podría indicar que debería hacer para que la solicitud fuera aprobada obteniendo así una recomendación precisa y viable para conseguir un objetivo. Similar al ejemplo expuesto, este argumento es valido para una gran variedad de escenarios como la admisión a una universidad, seleccionar solicitantes de empleo o identificar personas con alto riesgo de una enfermedad futura.

Dada una entrada x, un clasificador f y una métrica de distancia d, encontramos una explicación contrafáctica c resolviendo el problema [11]:

$$\begin{aligned} &\min_{c} d(x, c) \\ &s.t. f(x) \neq f(c) \end{aligned} \tag{2.1}$$

En base a lo anterior, un explicador contrafactual [14] como método es una función que toma como entrada un modelo, un conjunto de casos conocidos y la instancia de interés con su clase. Su salida es una explicación contrafactual que puede estar formada únicamente por un ejemplo o por un conjunto de estos. Las explicaciones contrafactuales como se ha clasificado al presentar los enfoques de la IAX, pertenecen a la familia de explicaciones basadas en ejemplos.

Comúnmente, se da confusión entre lo que son las explicaciones contrastivas y contrafactuales que comparten el objetivo de mostrar el comportamiento subyacente de un modelo [16]. Una explicación contrastiva responde a la pregunta de por qué ocurrió un evento teniendo en cuenta alternativas hipotéticas no ocurridas ("¿Por qué ocurrió A en lugar de B?"). Por otro lado, una explicación contrafactual describe un estado que no ocurrió y que contradice el conocimiento factual exponiendo como serían las cosas si las circunstancias fueran distintas. Se puede expresar como una afirmación condicional de la forma "Si A, entonces B" que representa una relación causal entre los eventos posibles y que podría llegar a dar lugar a una explicación probabilística de cada opción. Las explicaciones contrafactuales se consideran de naturaleza contrastivas, pero presentando una fuente de información complementaria valiosa. Asimismo, también cabe destacar el aprendizaje adversarial como otra área muy relacionada con la búsqueda contrafactual [14]. Estas dos técnicas, aunque de forma similar pretenden generar el menor número de cambios en una entrada para clasificarla de forma distinta, difieren en objetivo ya que los ejemplos adversarios se utilizan para engañar al modelo y descubrir ejemplos de clasificación errónea altamente confiables.

El interés en las CFEs [11] se fundamenta, en parte, en la flexibilidad del método subyacente y en la facilidad con la que los usuarios pueden comprender las explicaciones resultantes. Las explicaciones que resaltan información contextualmente relevante son similares a las explicaciones humanas.

2.3.1. Objetivos y Principales Temas de Investigación

En esta sección se van a exponer una serie de propiedades deseables y generales a abordar en el campo de la explicabilidad contrafactual [14][15]:

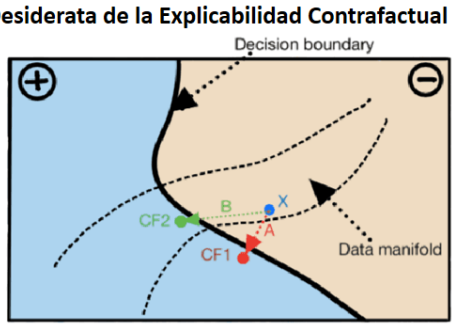


Figura 2.3: Representación de dos posibles caminos para un mismo punto, el límite de decisión y el manifold de los datos [13].

- Validez: debe clasificarse correctamente en la clase objetivo. Se pretende minimizar la distancia del caso original y el contrafactual, asegurando que la salida sea la etiqueta deseada.

- Similitud: se debe buscar que un contrafactual sea lo más similar posible al caso original.
- Accionalidad: se tiene que considerar la mutabilidad y la accionalidad al sugerir cambios con el fin de que sean factibles y prácticos de implementar. Por ejemplo, no modificar características inmutables.
- Esparsidad: se deben sugerir cambios efectivos de pocas características. Aunque sean de mayor magnitud pueden ser más fáciles de entender e implementar. Es importante encontrar un equilibrio entre el número de características cambiadas y la magnitud de los cambios realizados.
- Adhesión al manífold de datos (plausibilidad): las CFEs son más útiles cuando están cerca de la distribución de los datos que se utilizo para entrenar el modelo. Un punto lejos del conjunto posiblemente sea imposible o muy anómalo en la realidad. Por ejemplo, una medida para garantizar la plausibilidad podría ser limitar los valores de los atributos modificados para que no sean mayores o menores a los observables en el conjunto de datos y así, que no sea considerado como un valor atípico.
- Respeto por las relaciones causales: las explicaciones deben estar alineadas con las relaciones de causa y efecto que existen en el mundo real. Por ejemplo, conseguir un titulo educativo requiere un aumento de la edad.
- Discriminativo: debe ser capaz de destacar las razones detrás del cambio de una decisión, es decir, mostrar de forma clara las características que conducen a los diferentes resultados.
- Diversidad: si se forma un conjunto de contrafácticos para argumentar una instancia, la explicación contrafactual debe estar formada por casos diversos. Todos los contrafácticos generados deben pretender ser similares con el original y aumentar la diferencia entre ellos con el objetivo de presentar distintas acciones posibles que permitan cambiar el resultado.

Asimismo, también se destacan como deseables otras características con respecto al método encargado de generar las CFEs como que sea agnóstico del modelo y por lo tanto, que pueda funcionar con muchos tipos de sistemas de inteligencia artificial, que funcione correctamente aunque no se tenga acceso al modelo (solo accediendo a su función de predecir), que apueste por la eficiencia para garantizar su utilidad en aplicaciones de la vida real, que sea estable en las explicaciones que construye de manera que instancias similares tengan explicaciones similares y, que sea justo frente a cambios demográficos. A parte de estos desafíos más técnicos, también se debe pretender que sea seguro y que no revele detalles internos del modelo que pueden llevar a su robo [16].

2.3.2. Explicaciones Contrafactuales y el Razonamiento Humano

En esta última sección, se va a tratar la forma en que las personas piensan de manera contrafactual en relación con la mejora del desarrollo de sistemas de inteligencia artificial explicable [17].

Al realizar una persona un análisis de este tipo, se pueden obtener conclusiones sobre porque se tomo una decisión o cómo afectaría un cambio, sin embargo, es indiscu-

tible que cualquier evento real puede dar lugar a una cantidad de contrafactuales ilimitados y es un problema complicado identificar cuales son los casos que facilitan la construcción del modelo explicativo. Por este motivo, campos como la psicología y la ciencia cognitiva han estudiado la capacidad de razonamiento humano en este tipo de explicaciones e incorporar las ideas elaboradas por esta clase de estudios puede ser significativamente beneficioso.

En primer lugar, es común que las personas recurran a crear contrafactuales aditivos, los cuales agregan información nueva en una acción y ayudan a la resolución creativa de problemas en contraposición a los contrafactuales sustractivos que implican eliminar elementos establecidos y promueven un razonamiento lógico restringido a modificar una característica ya vista como negativa.

Las personas también tienden a pensar contrafactuales sobre las situaciones negativas y como podrían haber resultado diferentes en lugar de plantear casos desfavorables. Estas explicaciones que muestran un resultado beneficioso afectan a las intenciones futuras, pero conllevan un costo afectivo provocando emociones como la culpa o el arrepentimiento. En contraste, cuando se muestran como las cosas podrían ir a peor, las personas obtienen alivio y satisfacción, aunque no hay un aprendizaje sobre los errores. En definitiva, aunque el objetivo de esta clase de métodos es proporcionar explicabilidad y confianza sobre un sistema, pueden darse situaciones donde los contrafactuales deban adaptarse para proveer un mejor rendimiento futuro o justificar una decisión ya tomada.

Los contrafactuales también pueden servir para amplificar los juicios causales, destacando así las consecuencias de una acción particular. Además, si bien los eventos suelen tener múltiples causas, algunas de ellas son más destacadas al co-ocurrir con el resultado y son consideradas las principales del problema. Por otra parte, mostrar un conjunto de contrafactuales muy disperso tiende a destacar condiciones menos directas que permiten identificar relaciones significativas pero, al integrar acciones alternativas que conducen a un mismo resultado, también se puede generar la percepción de que este es inevitable.

Cuando se reflexiona sobre una acción pasada, las personas a menudo recurren a contrafactuales para justificar un rendimiento deficiente o tienden a enfocarse en controlar mejor el resultado. Estos contrafactuales asociados a algo negativo ya pasado también determinan culpabilidad sobre las acciones. Con respecto a estos, al imaginar cómo se podría cambiar una realidad, se tiende a enfocarse en los eventos inusuales o en considerar aquellas situaciones que para el sean de mayor probabilidad según sus creencias o imaginación. Todas estas tendencias pueden llevar a manipular una explicación por lo que las herramientas de apoyo deben centrarse en los eventos que realmente sean más probables.

Finalmente, cabe destacar que las personas entienden las conjeturas al visualizar el caso mentalmente. Una consecuencia de construir varios modelos mentales durante la comprensión de un evento cualquiera es que las personas hacen muchas más inferencias a partir de las condicionales contrafactuales lo cual, confirma la potencial utilidad de estas explicaciones frente a otras en el campo de la explicabilidad de sistemas.

Capítulo 3

Estado del Arte

En este capítulo se va a presentar una revisión de trabajos relevantes sobre metodología ya establecida diseñada para explicar modelos de caja negra que utilizan la importancia de las características en la construcción de contrafactuales, con un enfoque específico en resolver problemas tabulares.

Dado que el campo de investigación de las explicaciones contrafactuales se trata de un área que ha empezado a popularizarse y desarrollarse en los últimos años, como método de búsqueda bibliográfica, este estado del arte se ha centrado en trabajos principales publicados con un número no despreciable de citas y, a partir de estos, se han explorado sus relaciones con otros estudios para recopilar otras metodologías que aborden esta misma temática. Cabe destacar la ausencia de un estado del arte enfocado a cubrir el asunto específico expuesto.

Para formar una taxonomía sobre la que distinguir los distintos tipos de metodología, se ha tomado como base la presentada por Guidotti [14] que clasifica algoritmos generales respecto a la estrategia que emplean para recuperar las explicaciones contrafácticas. Se pueden distinguir dos tipos principales: los basados en optimización y los que siguen estrategias de búsqueda heurística. En resumen, los explicadores se ordenan primero con respecto a la estrategia y luego cronológicamente.

Otra forma de categorización podría haber sido realizando una distinción entre enfoques agnósticos y específicos al modelo o, como exponen Keane y Smyth [29], entre explicadores endógenos que intentan encontrar los casos dentro del conjunto de datos o generarlos a partir de combinaciones y exógenos que devuelven ejemplos generados sintéticamente. Los primeros están enfocados en garantizar la plausibilidad mientras que los segundos en formar los mejores resultados posibles.

3.1. Metodología Fundamentada en la Optimización

Este tipo de técnicas suelen ser desarrollas mediante el diseño de una función de perdida especifica que pretende garantizar un conjunto de propiedades deseadas en los contrafácticos obtenidos. En este contexto, el objetivo es hallar una instancia que minimice la pérdida empleando un algoritmo de optimización. Aunque, el primer explicador contrafactual basado en optimización fue el método OAE (Optimal Action Extraction) [18] propuesto por Cui et al. en 2015 y enfocado específicamente sobre modelos aditivos, el más famoso y considerado por muchos como el precursor de

estas explicaciones es el anteriormente presentado WATCH [1] de Wachter et al. en 2017. En el resto de la sección se van a exponer una serie de metodologías posteriores que introducen de alguna forma la importancia de las características dentro del problema de optimización.

En un primer lugar, el trabajo realizado por **Grath et al.** [19] en 2018 establece como punto de partida desarrollar explicaciones contrafactuales optimizando la función de perdida (3.1) propuesta por Wachter et al. [1]. En esta función, se emplea la distancia de Manhattan ponderada junto con la desviación absoluta mediana inversa (MAD) (3.2). La elección de MAD se justifica por su robustez a valores atípicos y su capacidad para introducir soluciones dispersas cuando la mayoría de las entradas son cero. Además, es importante destacar la necesidad de ajustar el parámetro λ dentro de la función para lograr la salida deseada y hacer los mínimos cambios posibles.

$$L(x, x', y', \lambda) = \lambda(\hat{f}(x') - y')^2 + d(x, x') \tag{3.1}$$

$$d(x, x') = \sum_{j=1}^{p} \frac{|x_j - x'_j|}{\text{MAD}_j} \tag{3.2}$$

Esta implementación general descrita asume que todas las características tienen la misma importancia en la decisión y que todas son modificables pero, con el fin de obtener contrafactuales más concisos y así, explicaciones más claras y comprensibles, los autores plantean la posibilidad de ser discriminativos introduciendo un vector de pesos a la métrica definida en la ecuación (3.2) siguiendo dos estrategias distintas:

- Basándose en la importancia global de las características utilizando un análisis de varianza (ANOVA) y creando un vector de pesos que promueve las características más influyentes.
- Basándose en un enfoque de vecinos más cercanos que pertenezcan a la clase deseada y construyendo un vector de pesos que capture las características locales más importantes, es decir, aquellas que muestren cambios. En este caso, los resultados obtenidos determinaron que esta no era la mejor alternativa.

Mas adelante en 2020, **Downs et al.** [20] presentan CRUDS (Counterfactual Recourse Using Disentangled Subspaces) como un método que utiliza el Codificador Variacional de Subespacio Condicional (CSVAE) [21] para extraer las características latentes más relevantes y generar múltiples contrafactuales que obedezcan las dependencias subyacentes, satisfagan las relaciones causales y respeten restricciones específicas marcadas por el usuario.

CRUDS en primer lugar distingue las características relevantes a partir de entrenar un CSVAE, una variante de un Autoencoder Variacional (VAE) en la que el espacio latente se divide en aprender las representaciones que son predictivas de las etiquetas y las representaciones latentes restantes necesarias para generar los datos. Después, emplea las características seleccionadas en el proceso anterior para formular contrafactuales siguiendo una estrategia de optimización y se filtran los casos resultantes para satisfacer las restricciones causales y las preferencias individuales. Como paso final para favorecer la interpretabilidad, se aproxima la distribución sobre contrafactuales utilizando un árbol de decisión y se resumen los caminos como

reglas condensadas y ordenadas según su probabilidad para resaltar que aspectos han tenido más influencia en la generación. En definitiva, CRUDS consigue producir recursos validos consistentes que respetan las dependencias, la causalidad y las restricciones.

Con una idea parecida a la anterior, **Pawelczyk et al.** [22] proponen el método C-CHVAE para encontrar conjuntos de características y formular contrafactuales con una alta probabilidad de ocurrencia. La idea que exponen es utilizar una extensión de VAE como es el Autoencoder Variacional Jerárquico Condicional (CHVAE) para aprender una representación latente de los datos que capture las características más importantes y modificar adecuadamente los valores de esta representación, lo que a su vez modifica los atributos accionables mientras se mantienen los inmutables.

El codificador toma los datos y los transforma en una representación de menor dimensión que captura las características esenciales determinando así, un espacio donde buscar los posibles contrafactuales. Seguidamente se perturba este vecindario de baja dimensión y se utiliza para alimentar un decodificador que reconstruye la entrada considerando esta alteración por lo que se produce un contrafactual potencial que se pasa al clasificador preentrenado para evaluar si se ha alterado la predicción. Como resultado se obtienen contrafactuales fieles, próximos y cercanos a regiones de alta densidad de datos y no se requiere de medidas de distancia predefinidas que actúen en el espacio de entrada real.

Posteriormente, en el trabajo desarrollado por **Chapman-Rounds et al.** [23] en 2021 se propone el algoritmo FIMAP (Feature Importance by Minimal Adversarial Perturbation) que proporciona explicaciones contrafactuales obteniendo la dirección en la que una instancia tiene que ser perturbada para cambiar su clasificación. Al aplicar la herramienta, se obtienen perturbaciones adversariales mínimas que pueden considerarse como la dirección contrafactual mínima, es decir, la dirección sobre las características en la que se podría perturbar una instancia para hacer que cambie su predicción de una forma óptima. En este enfoque, la plausibilidad no se considera en absoluto.

Este método describe cómo manejar el proceso de generación de perturbaciones adversariales mínimas considerando características de entrada continuas y discretas lo que amplía su aplicabilidad (un aspecto frecuentemente ignorado). Al trabajar con características continuas, el espacio de perturbaciones es significativamente grande por lo que asume una clase restringida de modelos que mapean el espacio de datos a perturbaciones. El objetivo es lograr los parámetros óptimos para esta función utilizando métodos basados en gradiente. Para ello, en primer lugar, se entrena un modelo sustito (ya que no se puede presuponer que el modelo sometido a estudio está disponible) con las entrada y etiquetas originales y otro modelo con las entradas originales y las etiquetas invertidas. Luego, las perturbaciones dadas por este segundo modelo se pasan a través del sustituto con los pesos congelados y se calculan los gradientes de perdida con respecto a las perturbaciones.

Por el otro lado, al manejar las características discretas, se presenta como un método que las perturba mediante el muestreo de una distribución categórica correspondiente. Para encontrar las muestras adversarias, se aplica el “truco” Gumbel-Softmax y estas se utilizan para calcular los gradientes de la perdida con respecto a las perturbaciones al pasarlas a través del modelo sustituto preentrenado como en el proceso anterior donde únicamente se ha añadido un término de regularización.

En otro estudio, **Galhotra et al.** [24] proponen LEWIS, un sistema capaz de desarrollar confractuales a partir de un modelo causal probabilístico subyacente especificado o aprendido de los datos, por lo que, puede resolverse simplemente utilizando datos históricos. En definitiva, crea contrafactuales contrastivos probabilísticos a partir de los datos (un enfoque completamente no paramétrico). Antes de generar cualquier tipo de explicación, en primer lugar, se definen las puntuaciones de explicación para cuantificar la influencia de cada atributo y así, la importancia de las características. Entonces, dado un algoritmo y un modelo causal probabilístico se obtiene un puntaje de necesidad que mide la atribución de la responsabilidad causal de las decisiones a un atributo, un puntaje de suficiencia que aborda la tendencia de un atributo para producir un resultado y una combinación de ambos que se puede utilizar para medir el poder explicativo general de un atributo. Seguidamente, para desarrollar las explicaciones contrafráctricas, LEWIS realiza intervenciones mínimas en un conjunto de variables accionables con un alto puntaje de suficiencia, es decir, las intervenciones que pueden modificar la decisión con una alta probabilidad. En definitiva, el recurso se calcula como un problema de optimización combinatoria sobre el dominio de variables accionables resolviendo una función de costo que determina la dificultad de modificar el valor actual de los atributos con una restricción que garantiza un puntaje de suficiencia definido.

Para terminar, en el trabajo realizado por **Jia et al.** [25] se presenta una metodología que emplea DeepLIFT (Deep Learning Important FeaTures) para descartar ejemplos contrafácticos en función de la importancia de las características en el cambio de categoría. DeepLIFT está desarrollado específicamente para trabajar de forma eficiente con redes neuronales profundas y funciona comparando las activaciones de cada neurona para las características de entrada con una activación de referencia (en este caso, los valores de entrada originales). Esta idea se ha desarrollado sobre un amplio conjunto de datos provenientes de la atención médica y empleando DICE (Diverse Counterfactual Explanation) ya que se trata del método más ampliamente popular y utilizado para generar los casos contrafactuales y presenta de una implentación específica para el aprendizaje profundo. Finalmente, se han observado los casos generados y, si muchas de las características presentaban una baja contribución, entonces el ejemplo era descartado por conllevar muchas transformaciones. En definitiva, este método pretende ayudar a elegir el ejemplo que implementar y filtrar resultados para únicamente generar contrafácticos dispersos y accionables.

3.2. Metodología Basada en las Estrategias de Búsqueda Heurística

La búsqueda heurística es una metodología que se basa en reglas o principios prácticos para encontrar soluciones a un problema, en lugar de seguir un algoritmo exhaustivo. Enfoques basados en esta idea tienden a ser mucho más eficientes a la hora de encontrar los contrafactuales que los algoritmos de optimización pero sus soluciones no suelen ser óptimas. Típicamente, se fundamentan en minimizar en iteraciones una función de coste que se basa en una elección local o heurística de un contrafactual valido. Dentro de esta tipología, se puede destacar como precursor el método SECD [26] de Martens y Provost en 2014, una de las primeras propuestas de explicación contrafactual.

En el resto de la sección se van a presentar métodos que combinan la importancia de las características con distintas estrategias de búsqueda.

En primer lugar, **Rathi et al.** [27] en 2019 proponen el método CFSHAP que consigue generar explicaciones contrafactuales sobre un conjunto de datos utilizando los valores Shapley para determinar qué factores funcionan a favor o en contra de una clasificación. Este planteamiento es la primera extensión del uso de SHAP a este espacio de problemas y, de entre los diferentes modelos probados, se destacan las máquinas de vectores de soporte (SVM) y las redes neuronales por ser los más adecuados al trabajar junto con esta metodología.

Primero, se estiman los valores Shapley del punto de datos de estudio y luego, se identifican aquellos que afecten negativamente a la clasificación deseada, es decir, los que presentan un valor negativo. A partir de estos, se forma el conjunto de características que van a ser modificadas con respecto al caso original. De seguido, se generan los vecinos más cercanos al caso en múltiplos de 50 hasta que se encuentran contrafactuales en estos puntos. Si se encuentran, se devuelven; de lo contrario, se continua la búsqueda. Esto puede ser una desventaja, ya que el contrafactual más cercano no siempre estará cerca del conjunto mutado.

Por otro lado, **White y Garcez** [28] proponen el método CLEAR (Counterfactual Local Explanations viA Regression), el cual ofrece explicaciones contrafactuales basadas en LIME [9] otro método líder en explicación local. En resumen, CLEAR pretende proporcionar explicaciones contrafactuales satisfactorias mostrando la importancia relativa de las características mediante coeficientes de regresión y cómo interactúan.

Para ello, CLEAR primeramente determina en el caso de estudio las b-perturbaciones (el cambio mínimo para lograr la clase objetivo) en cada característica mediante una búsqueda unidimensional. De seguido, se generan observaciones sintéticas etiquetadas y se crean conjuntos de datos equilibrados con observaciones cercanas a la original y distribuidas equitativamente entre las clases. A partir de estos datos, se ajusta un modelo de regresión local en el vecindario con la restricción de que el modelo debe pasar por la observación de interés y se calculan las b-perturbaciones estimadas. Finalmente, se evalúan los resultados con la realidad y se ajustan los parámetros del modelo de regresión de manera iterativa hasta encontrar un equilibrio óptimo entre interpretabilidad y fidelidad. CLEAR devuelve como explicaciones los contrafactuales reales y estimados, los regresores que muestran los coeficientes de las características y el error aproximado.

Seguidamente, **Keane y Smyth** [29] en 2020 presentan un trabajo muy destacable que será utilizado como base en varios estudios de esta sección. Además, antes de exponer el método desarrollado, los autores han examinado el potencial contrafactual de varios conjuntos de datos frecuentemente utilizados en la literatura, empleando un enfoque de coincidencia exacta. Sobre los resultados, estos revelan que encontrar "buenos" contrafácticos de forma natural con menos de dos diferencias con respecto al caso original no es lo usual. Estos representan menos del 1 % del total de contrafácticos.

Con el fin de abordar el desafío expuesto, se propone el explicador de contrafactuales basados en casos (CBCE) [29] donde se realiza un proceso para determinar qué características son las más importantes y permitir reutilizar contrafactuales existentes para orientar la búsqueda y generar análogos adecuados. En primer lugar, se identi-

fica el caso más similar con una predicción diferente al que se pretende explicar para tomarlo como punto de partida, considerando un cierto grado de tolerancia al calcular similitudes y asegurando que solo se difiera en un máximo de dos características. Luego, se toman estas características de diferencia como las únicas responsables del cambio y se formula un contrafáctico donde se sustituyen por las del caso original. Finalmente, se comprueba con el modelo subyacente si resulta tener la categoría objetivo y, en caso contrario, se realiza una adaptación para obtener nuevos valores en las características de diferencia iterando en orden sobre los vecinos más cercanos que pertenecen a la clase objetivo. La contribución principal de este proyecto radica en identificar las características más relevantes para la formación de contrafácticos plausibles y explicativos a partir de reutilizar valores de casos reales.

Tomando el trabajo expuesto como plantilla, los mismos autores han desarrollado otras metodologías [30] [31] orientadas a mejorar el enfoque y generalizar la técnica presentada. En uno de estos métodos, en vez de formar contrafácticos a partir del caso similar más cercano, se seleccionan los k más cercanos y se genera un candidato diferente para cada uno. En otro trabajo, se ha construido una variante que transforma los valores de diferencia de características en categóricos. Este último se centra en cumplir con los requisitos psicológicos en lugar de simplemente seguir las intuiciones de los diseñadores.

En otro estudio desarrollado por **Le et al.** [32] se propone GRACE (GeneRAting Constrastive samplEs), un algoritmo con el que se pretende dar muestras explicativas a redes neuronales modificando un pequeño subconjunto de características para cruzar el límite de decisión y formar un predicado explicativo. Para ello, se ha desarrollado una función que minimiza la distancia entre el caso de estudio y el contrastivo bajo ciertas restricciones que garantizan una explicación concisa e informativa. En resumen, se establece un límite de características a modificar, se añade una función de incertidumbre simétrica para verificar que cualquier información mutua entre las características sea inferior a un límite (se minimiza la entropía) y se asegura un resultado realista dentro del dominio. Finalmente, se genera una explicación en texto natural utilizando una plantilla rellenada con la diferencia entre los valores. Es importante señalar que, si bien GRACE aborda el problema de manera similar a los métodos que emplean optimización, utiliza un algoritmo heurístico para la generación de contrafácticos. Además, tanto en su función objetivo como en otros procesos, hace uso de los gradientes de la red, los cuales no siempre están disponibles.

Con el objetivo de trabajar con el mejor conjunto de características y hacer modificaciones óptimas sobre los casos, antes de resolver la función objetivo se hace una selección de los atributos más representativos en el cambio en la decisión. Para ello, se han expuesto dos estrategias, emplear los gradientes con respecto a la clase contrastiva más cercana (vista global) y utilizar los valores devueltos por un modelo interpretable (vista local) como se representa en la Figura 3.1. De seguido, se forma el conjunto como una lista ordenada donde se agrega cada característica de la más a la menos predictiva asegurándose de que la información mutua de cualquier par supere un umbral.

Al mismo tiempo, **Dandl et al.** [33] proponen la primera metodología desarrollada con el fin de cubrir varias limitaciones devolviendo así un conjunto de contrafactuales de Pareto (eficientes en varios aspectos). Este método, llamado MOC (Multi-Objective Counterfactual Explanations), trata la formulación de contrafactuales como un pro-

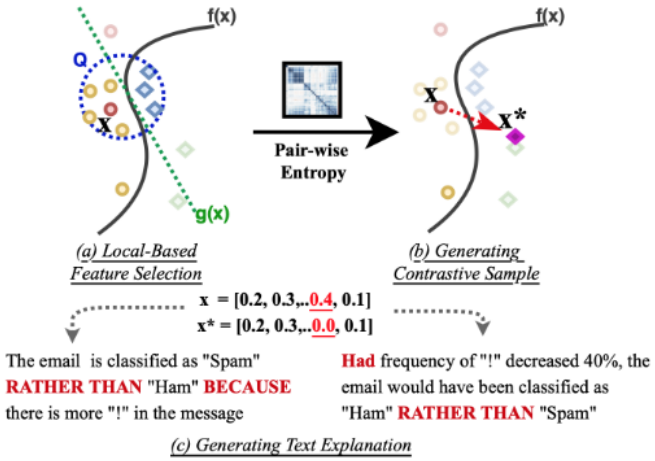


Figura 3.1: GRACE utilizando los valores devueltos por un modelo interpretable (vista local) para generar ejemplos contrastivos y una explicación textual

blema de optimización genética multiobjetivo.

MOC utiliza un algoritmo genético y se fundamenta en la minimización de la ecuación (3.3). En esta ecuación, el primer término cuantifica la distancia entre la predicción y el conjunto de datos deseado, el segundo termino cuantifica la distancia entre las características del contrafactual y las características originales utilizando la distancia de Gower (para tener en cuenta características mixtas), el tercer termino restringe el número de características modificadas siguiendo la norma L0 y el último término evalúa la distancia promedio ponderada de Gower entre el contrafactual y los puntos de datos más cercanos para determinar si es un caso plausible. Además de modificar la función objetivo, los autores proponen otros ajustes entre los que se destaca valorar la influencia de las características en la predicción. La importancia de cada característica para una sola predicción se mide con la desviación estándar de la curva de Expectativa Condicional Individual (ICE). Las curvas ICE muestran cómo cambian las predicciones cuando se varia una característica por lo que, cuanto mayor sea la desviación estándar, más alta será la probabilidad de que la característica se modifique con respecto a la del caso original.

$$\min_x o(x) := \min_x \left(o_1(\hat{f}(x), Y'),\ o_2(x, x^*),\ o_3(x, x^*),\ o_4(x, X^{\text{obs}}) \right) \tag{3.3}$$

Posteriormente, en un estudio realizado por **Ramon et al.** [34] se plantea como impracticable encontrar un contrafáctico en espacios de alta dimensionalidad al realizar una búsqueda completa. Para dar solución a esta problemática, los autores propusieron LIME-C y SHAP-C como dos métodos híbridos que combinan algoritmos capaces de generar una lista de características clasificadas por importancia y la búsqueda heurística a partir del trabajo realizado por Martens y Provost [26]. Cabe destacar que la propuesta de este articulo está enfocada para datos conductuales o textuales los cuales no son el objetivo del presente trabajo, pero, mediante una función para binarizar un conjunto de datos, se podría aplicar sobre datos tabulares.

En un primer paso, se utiliza el método LIME o SHAP para generar una explicación de atribución de importancia. A partir del resultado obtenido, se presenta un modelo

de explicación lineal con representación binaria de las características similar a una lista y, sobre esta, se aplica el algoritmo lin-SEDC [26] para formular un contrafactual. El algoritmo itera sobre las características activas ordenadas de mayor a menor coeficiente y las va eliminando (estableciendo en cero) hasta que se obtiene como resultado un cambio en la predicción. De entre los resultados, se destaca que LIME-C mostro ser más estable en todos los casos y relativamente más rápido y eficiente.

Mas adelante, **Wiratunga et al.** [35] en 2021 proponen el explicador DisCERN que se fundamenta en los vecinos más cercanos no similares (NUNs) y en la identificación de cambios mínimos a partir de las características más importantes. Esta es una ampliación de una metodología ya desarrollada por los mismos autores [36] donde únicamente se había considerado LIME como técnica para obtener las relevancias. En el estudio final, se emplean LIME y SHAP para indicar la magnitud de la influencia de un atributo en la salida. Estos pesos se han utilizado para definir una restricción de ordenamiento y así conseguir un listado con las características accionables sobre el que seleccionar un subconjunto mínimo de cambios que modifiquen la salida. Por otra parte, se ha calculado la distancia euclidiana entre el caso de estudio y los NUN candidatos para seleccionar aquel con el que se consiga un resultado de menor valor, el cual es considerado óptimo.

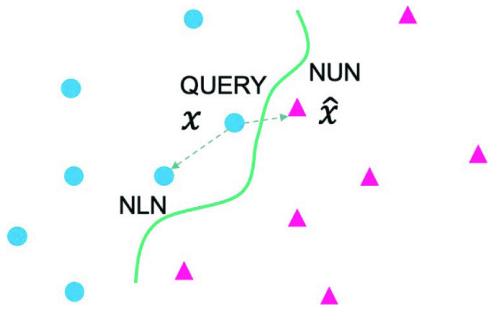


Figura 3.2: Ejemplo de vecindario donde se ilustra un vecino más cercano similar (NLN) y un vecino más cercanos no similar (NUNs).

Una vez presentados ambos procesos, se han propuesto dos métodos para determinar las características que deben ser modificadas al formar un contrafáctual:

- QRel donde se remplazan los valores de las características más importantes en el caso de estudio con los correspondientes en su NUN.
- NRel donde se identifican los valores de las características más importantes en el NUN y con estos se modifica los correspondientes en el caso de estudio.

Finalmente, las características se reemplazan iterativamente hasta que se cumple la condición de cambio accionable o la consulta se cambia completamente al NUN (lo que garantiza un cambio de clase). Cuantas menos iteraciones sean necesarias, mejor serán las características accionables descubiertas. Los resultados finales obtenidos en este estudio muestran que SHAP es el explicador más óptimo para ordenar las características según su importancia y señalan que los métodos QRel y NRel propuestos obtienen resultados comparables.

En 2022 **Zhong y Negre** [37] presentan un algoritmo para generar explicaciones contrafactuales mejoradas con SHAP en sistemas de recomendación. En resumen, pretenden explicar que características se deben cambiar sobre un elemento o una

lista de elementos para que no estén dentro del conjunto recomendado. Los autores destacan que utilizan SHAP frente a otros métodos de vanguardia como LIME ya que permite interpretar un modelo desde un punto de vista global y, al estar fundamentado en la teoría de juegos, facilita un cálculo justo de la contribución de cada característica. Además, se resuelven las preocupaciones sobre la estabilidad de LIME y se garantiza precisión local y la consistencia.

El método propuesto obtiene los valores SHAP sobre la información empleada para realizar la recomendación y establecer así un punto de partida inteligente. Como se pretende considerar únicamente las características con un mayor impacto, los autores deciden seleccionar solo aquellas que superen un umbral de 0 y ordenarlas de forma descendente. Seguidamente, se cambian los valores de forma secuencial en la lista hasta que no aparezca la instancia dentro las recomendaciones. En el ejemplo desarrollado se emplea como estrategia de sustitución una alteración aleatoria pero, se podría resolver utilizando valores sobre los casos más similares que no incluyan la misma recomendación. Finalmente, se destaca que al trabajar con listas (conjuntos) de recomendaciones se ha seguido un enfoque similar pero, al evaluar si se ha logrado formular un contrafactual valido, se ha comparado la similitud de las dos listas representada mediante el coeficiente de superposición con un umbral que determina si se ha alcanzado un porcentaje de recomendación explicado.

En otro estudio posterior, **Li et al.** [38] proponen FAST-CF, un método de explicación contrafactual para un trabajo de predicción de llamaradas solares. Este presenta mucho parecido con la idea presentada por Keane y Smyth [29].

FAST-CF se puede explicar en dos pasos: recuperar y partir del vecino más cercano no similar y adaptar la consulta original con valores de los atributos del NUN. En este segundo paso, se seleccionan cuáles son las características más importantes comparando la distancia por dimensión y, si la distancia es relativamente grande, entonces se considera como una dimensión relevante en la cual hay que enfocarse. Luego, se sustituyen las seleccionadas en la instancia original de manera que la etiqueta de clasificación cambie a la clase deseada. Finalmente, se destaca que en este proceso se satisfacen algunas propiedades como la interpretabilidad, la validez, la proximidad, la esparsidad y la contigüidad.

En un trabajo más reciente desarrollado por **Cho y Shin** [39] en 2023, se introduce un método de generación contrafactual ponderado por SHAP. Este proceso se lleva a cabo mediante un algoritmo genético (GA) de múltiples objetivos y se fundamenta en la ecuación (3.4) utilizada para el problema de optimización donde se presenta λ como parámetro de equilibrio entre los distintos términos. Cada uno de estos de refiere a diferentes cualidades requeridas por los contrafactuales.

$$\arg\min \ \ \text{dist}(x, \hat{x})\lambda_1 + \text{lof}(\hat{x})\lambda_2 + \text{spr}(\hat{x})\lambda_3, \quad \text{sujeto a} \quad f(\hat{x}) \neq f(x) \tag{3.4}$$

El primer término se corresponde con la distancia entre en caso original y un contrafactual donde se ha empleado como medida la distancia euclidiana ponderada por los valores obtenidos del método SHAP. En estos, no se ha considerado la dirección del impacto, es decir, si son positivos o negativos. El segundo término considera la viabilidad al minimizar el puntaje LOF (Factor de Valor Atípico Local) de la instancia dada, lo que proporciona una medida del grado en que la instancia es un valor atípico en su área local. LOF calcula su puntaje basándose en la densidad local, y una

localidad definida por k vecinos. El último término se refiere a la dispersión y refleja el número de características cambiadas.

Finalmente, se utiliza un algoritmo evolutivo con el objetivo de generar soluciones óptimas resolviendo este problema de optimización y encontrar así buenos contrafactuales. Para explicar brevemente el proceso, se inicia con un conjunto de cromosomas que representan los contrafactuales. Estos se evalúan utilizando la función definida y luego se aplican operadores genéticos como los procesos de selección, mutación y cruce para formar nuevas soluciones potenciales. Este proceso se repite de forma iterativa hasta cumplir con un criterio de parada y encontrar así un contrafactual óptimo.

3.3. Trabajos Complementarios

Este último apartado, se trata de una sección adicional al estado del arte del tema sometido a estudio donde se van a presentar varios trabajos que, aunque no traten el objetivo principal, están muy relacionados con el mismo y exponen ciertas ideas que han sido consideras interesantes para el desarrollo posterior.

En el artículo desarrollado por **Adhikari et al.** [40] se proponer el método explicativo LEAFAGE (Local Example and Feature importance-based model AGnostic Explanations) fundamentado en la importancia de las características y el razonamiento local sobre modelos complejos. Aunque, el objetivo del método no sea formar contrafactuales ya que simplemente pretende proporcionar explicaciones en forma de ejemplos extraídos, el proceso de selección puede ser interesante para formar ejemplos basados en los NUNs.

LEAFAGE utiliza un subconjunto de datos el cual contenga la frontera de decisión local y un número mínimo de instancias por clase para construir un modelo lineal y obtener así la importancia de cada característica. Aunque se podría usar la distancia euclidiana como en otros estudios ya presentados, los autores destacan que no refleja el razonamiento seguido por el clasificador y que solo se fija en las características. Por este motivo, proponen utilizar los pesos obtenidos para desarrollar la medida de disimilitud sobre dos instancias como en la ecuación (3.5) donde D es la distancia euclidiana y w los pesos de las características. Además, también se ha añadido un segundo factor para trabajar sobre las instancias más cercanas.

$$b(t) = D(w_z^T t, w_z^T z) \cdot D(t, z) \tag{3.5}$$

Un estudio realizado por **Poyiadzi et al.** [41] expone claras deficiencias al trabajar con contrafactuales al buscar siempre los cambios mínimos necesarios para encontrar el resultado deseado. Estos enfoques "normales" no tiene en cuenta consideraciones como metas inalcanzables o cuales son los caminos factibles entre la instancia de inicio y el contrafactual sugerido por lo que se pueden producir resultados inadecuados e imprácticos. Estas deficiencias se muestran claramente en la Figura 3.3 donde se aprecia como el punto A y B seguramente sean imposibles ya que se encuentran en regiones muy despobladas o, en el punto C, que podría llegar a ser inalcanzable ya que no se distingue ningún camino de puntos continuo hasta este, es decir, los casos intermedios son escasos. Este descubrimiento establece una nueva línea de investigación critica con varias de las propiedades deseables ya presentadas que consiste

en proporcionar un camino accionable y factible para transformar un punto de datos en otro que cumpla con ciertos objetivos.

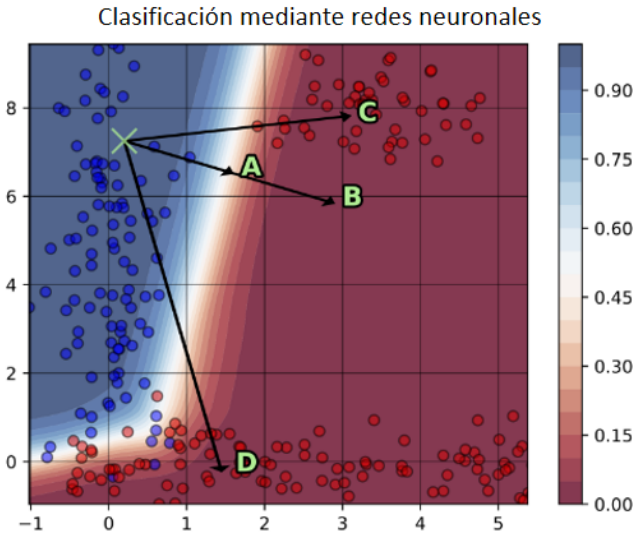


Figura 3.3: Representación de las deficiencias de los principios actuales sobre los contrafactuales.

Después de exponer esta necesidad sobre las explicaciones contrafactuales, los autores del estudio proponen el método FACE (Feasible and Actionable Counterfactual Explanations) [41] que respeta la distribución de los datos subyacente y realiza conexiones a través de caminos de alta densidad. Este algoritmo utiliza las distancias de camino más corto definidas mediante métricas ponderadas por densidad.

Primeramente, se construye un grafo sobre los puntos de datos siguiendo un enfoque KDE, k-NN o ε-grafo en la estimación de la densidad. Luego, el usuario decide ciertas propiedades sobre la instancia objetivo como la clase esperada u opcionalmente un umbral de confianza, un umbral de densidad, un umbral de peso personalizado donde se asignan pesos específicos sobre diferentes características o condiciones personalizadas (como restricciones en la magnitud de las modificaciones). De seguido, se actualiza el grafo suprimiendo las aristas descartadas por las restricciones del usuario y finalmente, se aplica el algoritmo de Primer Camino más Corto sobre todos los candidatos que cumplan los requisitos establecidos. Como salida se obtiene un grafo con el recorrido realizado por los puntos de datos.

Por otra parte, es importante destacar el método DICE (Diverse Counterfactual Explanations) propuesto por **Mothilal et al.** [42], que, extendiendo el trabajo de Wachter et al. [1], busca generar conjuntos de ejemplos contrafactuales diversos, accionables y viables. Aunque este enfoque no aborda directamente la problemática estudiada en el presente trabajo, se mencionará brevemente debido a su amplia aceptación y uso por parte de la comunidad científica en comparación con otros. Se puede resaltar sin lugar a duda como el método más popular en la actualidad.

En su desarrollo, se destaca la combinación de diversidad y factibilidad. La diversidad se logra al construir sobre procesos puntuales determinantes y al tener en cuenta los ejemplos más cercanos al original que se consideran los más útiles. Para esto, se cuantifica la proximidad entre las características de ambos casos, opcionalmente

ponderada por un peso. A partir de estas consideraciones, se formula la función (3.6) a optimizar mediante descenso de gradiente para la generación de contrafácticos. Esta función emplea una pérdida tipo hinge-loss, y en lo que respecta a la distancia, se utiliza la distancia L1 para características continúas dividida por la desviación media absoluta, mientras que para las categóricas se utiliza una métrica más simple que asigna una distancia de 1 si el valor difiere del de la entrada original.

$$\underset{x'_1,\ldots,x'_k}{\arg\min}\left(\frac{1}{k}\sum_{i=1}^{k}\max\left(0,1-y'\text{logit}\left(b(x'_i)\right)\right)+\frac{\lambda_1}{k}\sum_{i=1}^{k}d(x'_i,x)-\lambda_2\text{div}(x'_1,...,x'_k)\right) \tag{3.6}$$

Además de estos puntos, se valora la esparsidad mediante una operación posterior en la que se restauran características continuas a sus valores originales asegurando que la clase predicha no cambie. También se respetan restricciones establecidas por el usuario.

En otro artículo, **Albini et al.** [43] han desarrollado un método que muestra un proceso que puede ser aprovechable para generar ejemplos contrafactuales. En este trabajo, en lugar de emplear la importancia de las características en la formulación, se propone el método CF-SHAP que utiliza los casos contrafactuales para atribuir características contrafactuales en el cálculo de valores Shapley.

Para ello, en primer lugar, se trata la elección de los valores de fondo que se utilizan en contraste con la entrada para generar los valores Shapley. Algunas de las practicas más comunes propuestas son: utilizar todo o parte del conjunto de entrenamiento, utilizar solo las muestras con etiquetas diferentes en el conjunto de entrenamiento o emplear solo las muestras con predicciones distintas. Estas elecciones sobre el conjunto tienen en común que son elegidas a priori por lo que destaca que pueden llevar a veces a predicciones engañosas para un usuario, pero, aun así, el segundo y tercer enfoque pueden mejorar las explicaciones al informar sobre qué características fueron más importantes para el rechazo. Pues bien, se ha considerado que esta declaración puede ser aprovechable en un método que forme contrafactuales a partir de la importancia de las características.

Debido al objetivo del trabajo, finalmente se utiliza como datos de contraste un conjunto contrafáctico que presenta casos con una situación similar al original, pero de distinta predicción, en lugar de utilizar "ejemplos promedios". También se destaca que se enriquece la explicación proporcionada por SHAP con tendencias derivadas, es decir, información adicional para describir la dirección del cambio. Este enfoque culmina en resultados que superan los métodos de atribución previamente utilizados.

Para terminar, se destaca en un último trabajo la visualización y la representación de los resultados obtenidos como un aspecto fundamental para comprender y comunicar eficazmente el rendimiento. En el contexto del razonamiento contrafactual, la visualización de los límites de decisión es especialmente relevante ya que nos permite explorar cómo un modelo de aprendizaje automático toma decisiones en diferentes regiones del espacio de características. Esto no solo facilita la interpretación, sino que puede ser muy útil para determinar de una forma rápida si los casos obtenidos cumplen con propiedades deseables de los conjuntos contrafactuales como la plausibilidad, la esparsidad o la validez de un recurso. Por motivos como estos, **Sohns et al.** [44] proponen una metodología de exploración donde crean mapas en hiperplanos

de alta dimensionalidad en los que se separa visualmente el conjunto de datos por colores y se indican los límites de decisión.

En este trabajo, se destaca la elección de crear incrustaciones mediante una reducción de dimensionalidad lineal en lugar de optar por transformaciones de distancia no lineales. Esta decisión se justifica en base al éxito demostrado en la interacción con proyecciones lineales locales. Además, aunque las técnicas no lineales capturan correctamente las distintas variedades, introducen una distorsión significativa en el espacio de entrada modificando la distancia entre los puntos de incrustación y los límites de decisión. Finalmente, los autores decidieron emplear PCA restringiendo las muestras de entrenamiento al conjunto de vecinos más cercanos.

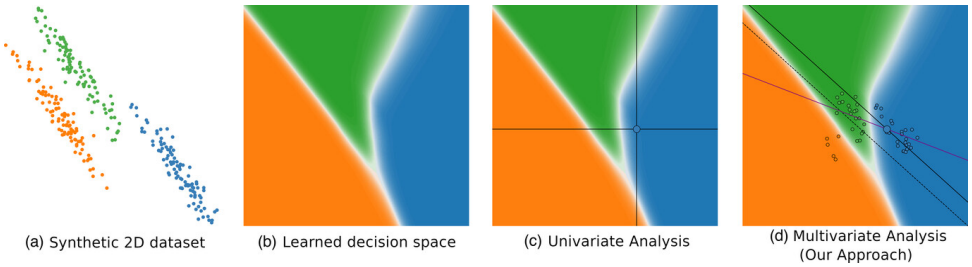


Figura 3.4: Ejemplo de representación según el trabajo realizado por Sohns et al. [44]

Aun así, es importante tener en cuenta la especial dificultad de formar una representación como la planteada de un conjunto de datos multidimensional completo y particularmente con características mixtas ya que no se terminan por representar de forma precisa la distancia entre muestras y puede terminar dando lugar a una incorrecta evaluación.

Capítulo 4

Diseño de Propuestas

En este capítulo se va a explicar el diseño y las decisiones tomadas al desarrollar la formulación de contrafactuales en el presente trabajo. Concretamente, en primer lugar se van a introducir las dos propuestas elaboradas, así como la motivación de su planteamiento que surge de la visión obtenida sobre las propiedades de las explicaciones contrafactuales y de varios de los trabajos expuestos en el estado del arte. Posteriormente, se va a entrar en detalle en el diseño de cada una de estas propuestas.

4.1. Propuestas de Desarrollo de Contrafactuales

Como se ha comentado, en base al conocimiento presentado en los capítulos anteriores, se han construido dos propuestas de algoritmos para generar contrafactuales guiados por la importancia de las características que, aunque trabajan en el mismo contexto, cubren aspectos bien diferenciados en el desarrollo de este tipo de explicaciones. En este punto se va a presentar la idea fundamental y la motivación detrás de cada uno de los enfoques:

- Formular contrafactuales sintéticos desde la optimización a partir de un punto de partida inteligente.
- Construir un camino incremental de prototipos para guiar el cambio hacia una situación contrafactual.

Atendiendo al primer enfoque, se pretende desarrollar un algoritmo que genere contrafactuales óptimos a partir de muestras reales de la clase objetivo combinando los dos enfoques metodológicos identificados en el estado del arte y guiando ambos procesos mediante la importancia de las características. Para ello, primeramente se realiza una búsqueda heurística para encontrar una solución base de manera eficiente y, a partir de esta, se resuelve un problema de optimización que mejora el resultado. Con este enfoque, no solo se busca mejorar el rendimiento de los métodos algorítmicos exhaustivos u obtener un resultado superior al que se consigue basándose únicamente en principios prácticos, sino que también se pretende aprovechar las ventajas individuales de cada metodología y el conocimiento sobre que características pueden ser más importantes al hacer modificaciones. Todo ello con el fin de garantizar las propiedades deseables de los contrafactuales y así las mejores soluciones posibles. Además, es importante destacar que se han combinado el descenso de

gradiente y heurísticas basadas en vecinos, los dos tipos de algoritmos más prevalentes utilizados.

Esta metodología, como es común al trabajar en este tipo de explicaciones, se fundamenta en conseguir una solución contrafactual valida con la mayor similaridad posible a partir de una métrica de distancia. En este caso, la distancia se va a utilizar junto con la importancia de las características para obtener resultados discriminativos que permitan identificar qué aspectos son los que realmente condicionan a la muestra y, no dispersos, cambiando en distinta magnitud el menor número posible de variables que son fundamentales al formar varios contrafactuales que conserven el sentido. Por otro lado, el partir de un contrafactual real favorece a que las soluciones sean accionables, plausibles y que se respeten las relaciones causales, aunque también se han desarrollado otra serie de métodos y comprobaciones que fomentan la plausibilidad y la causalidad.

Con la segunda propuesta, se pretende formar un camino de prototipos que se fundamente en las características que realmente importan para lograr el cambio de clase deseado, en resumen, al aplicar esta metodología sobre una instancia, se debe conseguir una sucesión de casos reales que fomenten modificaciones en las características más importantes hasta alcanzar un contrafactual y una explicación de cómo llegar al mismo. Con este planteamiento, se garantiza que las metas sean alcanzables y adecuadas, un problema que destacan Poyiadzi et al. [41] sobre los métodos de generación de contrafactuales actuales.

Para desarrollar una solución que tenga en cuenta la problemática planteada, se ha tomado como base el método FACE [41] elaborado por los mismos autores que destacan esta deficiencia en los explicadores actuales. Siguiendo su trabajo, se ha formado un grafo sobre el conjunto de datos según la similaridad y la plausibilidad de las muestras aunque de forma distinta y, en mi caso, ponderando la métrica de distancia con la importancia de las características. Seguidamente, se trabaja sobre el grafo para concluir cual es el camino más próximo hasta llegar a un buen contrafactual. Por último, cabe destacar que se han considerado otras propiedades de los contrafactuales como en la anterior metodología y que, de forma adicional, se ha añadido la posibilidad de sustituir las características más importantes de forma iterativa sobre la inicial permitiendo comprobar si es posible este cambio en el contexto de los datos.

Se presenta el código final desarrollado y comentado sobre ambas propuestas en `https://github.com/IsmaelBeviaB/CounterfactualesTFM_IBB` aunque se recomienda su visualización en forma de cuaderno de Jupyter Notebook en plataformas como Google Colab o Jupyter. En este se incluye una prueba ejecutada de cada propuesta que se corresponde a la primera descrita en sus respectivas secciones del apartado de experimentación utilizando el conjunto de datos Heart Disease. Cabe destacar que el objetivo es mostrar únicamente la implementación de las propuestas y un caso de uso, ofreciendo así un contenido claro y sencillo de revisar. Se han excluido los apartados de preprocesamiento, construcción de modelos, otras pruebas y comparativas los cuales van a ser desarrollados y explicados a lo largo de este documento.

En los siguientes apartados se van a explicar en detalle todos los aspectos relacionados con el diseño de ambas metodologías. Por otro lado, algunas partes específicas se van a complementar con pseudocódigo el cual no pretende sustituir la programa-

ción completa ni abarcar toda la información, sino que se ha incluido para mejorar la comprensión del desarrollo de los procesos, servir como esquema de las operaciones y facilitar un mejor entendimiento.

4.2. Fundamentos de Diseño

Para una mejor comprensión, antes de explicar la arquitectura y flujo de cada método, se van a tratar varios aspectos fundamentales que están incluidos en la forma de actuación de los algoritmos. Parte de las ideas que se van a desglosar en este aparatado son aplicadas en ambas propuestas por lo que se ha visto conveniente introducirlas previamente.

Uno de los aspectos fundamentales a considerar dentro de las explicaciones contrafactuales es la **similitud** entre muestras. Esta noción juega un papel crucial en determinar cuan parecidos son los ejemplos presentados para la comprensión de cómo afectan las modificaciones sobre la decisión final del modelo. En resumen, desde un punto de vista muy general, cuanto más parecidas son las muestras contracfactuales con la instancia original, mejor es la explicación que ofrecen. Pues bien, este concepto se puede cuantificar haciendo uso de métricas de distancia las cuales pueden verse como una medida de la diferencia entre dos objetos.

En el contexto del trabajo, se asume la presencia de variables continuas que pueden representar tanto valores numéricos como codificar categorías con un orden, así como de variables binarias que codifican datos categóricos. Para abordar este tipo de problemas, se ha explorado sobre métricas que permitan comparar adecuadamente conjuntos de datos mixtos. Finalmente, se ha utilizado como punto de partida la distancia de Gower [45], una técnica popular en esta tarea que hace comparable ambos tipos de datos ajustando las distancias entre valores de 0 a 1 en cada característica, donde 1 representa la máxima disimilitud. En esta métrica se define la diferencia entre las características continuas como $d(x_j, x'_j) = |x_j - x'_j|/\hat{R}_j$ donde $\hat{R}_j$ se corresponde al valor del rango de la característica j que se obtiene sobre el conjunto de datos y, en las diferencias dentro de las características categóricas, se analiza la coincidencia siendo un valor 0 cuando $x_j = x'_j$, y de 1 en el caso contrario.

La distancia de Gower, tal como se presenta, enfrenta varios problemas. Uno de ellos es la contribución desequilibrada de las variables cuantitativas en comparación con las categóricas ya que el cambio máximo en una variable numérica que posiblemente sea muy complicado de alcanzar da lugar a una distancia de 1 lo cual es equivalente a un cambio simple en una variable categórica. Por otra parte, el escalar las características continuas sobre el rango de la distribución de la variable, hace que la medida sea muy sensible a valores atípicos lo cual afecta al valor de la estimación pudiendo llegar a reducir mucho la contribución de una variable. Con el objetivo de solucionar estas problemática expuestas y conseguir una forma de implementación eficiente, se ha explorado sobre otras formas de normalización de las características numéricas sobre la ecuación inicial. Finalmente, se ha optado por emplear el rango intercuartílico (P80 % - P20 %) representado como IQR en la ecuación (4.1) que expone la implementación final. Este enfoque proporciona valores cercanos a 1 que los hace comparables con los datos categóricos, penaliza los valores atípicos aumentando su valor por encima de una diferencia estándar en función de su magnitud y hace que las diferencias normales entre los valores sean mejor consideradas. Esta

forma de normalización se ha considerado que presenta mejores resultados y es más eficiente que otras técnicas como la mediana absoluta de la desviación (MAD), el uso de la tangente hiperbólica y la normalización min-max.

Por último, es importante mencionar que se ha considerado la posibilidad de que, debido a la presencia de una gran cantidad de valores repetidos en una característica, el rango intercuartílico calculado entre los márgenes presentados pueda ser igual a 0. En tales casos excepcionales, se ha incrementado la diferencia entre estos márgenes en pasos cada vez más pequeños a medida que nos alejamos de los límites predeterminados con el fin de ajustarse a un valor cercano al ampliamente repetido. Esta práctica busca penalizar más la magnitud del cambio en estas características, asumiendo que, si hay muy pocas modificaciones dentro de una variable, este cambio es potencialmente más complicado que se dé y representa una diferencia destacable entre dos muestras.

Para referirse a la métrica que se ha descrito en este apartado durante el resto del documento, se le va a dar el nombre de distancia GIQR haciendo referencia a la distancia de Gower tomada como base y al rango intercuartílico empleado en la normalización de las variables continuas.

$$d(x_j, x'_j) = \begin{cases} \frac{|x_j - x'_j|}{IQR_j} & \text{si } x_j \text{ es numérico} \\ I_{x_j \neq x'_j} & \text{si } x_j \text{ es binario} \end{cases} \tag{4.1}$$

Otro aspecto relevante a considerar es la **plausibilidad**, es decir, la factibilidad como caso válido. Para evaluar esta cualidad en las muestras objetivo, se examina la relación de la totalidad de los datos y la distribución de los valores alrededor la muestra lo cual se puede interpretar como una medida de su densidad dentro del conjunto. Como se ha planteado esta problemática, se puede establecer una relación clara con la distancia ya que se fundamenta en que si un caso se encuentra alejado del conjunto original, seguramente se trate de una situación anómala o imposible. En base a esta premisa, se han desarrollado dos herramientas distintas que determinan si un punto se encuentra en una región lo suficientemente densa dentro del espacio de características.

Por un lado, se ha entrenado un algoritmo de vecinos más cercanos utilizando el conjunto de datos normalizado (en el caso de la primera propuesta, solo con los datos que presentan la clase objetivo) y, se han obtenido las distancias de cada punto con sus k vecinos. A partir de la matriz resultante y un porcentaje de restricción indicado por parámetro, se calcula una densidad umbral mínima como el inverso de las distancias siguiendo dos posibles estrategias: considerando únicamente el vecino más lejano o la media de las distancias de todos los vecinos excluyendo al más cercano. Posteriormente, este proceso se repite para cada muestra individual sometida a estudio y se compara con la densidad umbral para determinar si un caso es válido.

En una segunda técnica para evaluar la plausibilidad, se entrena el algoritmo Local Outlier Factor (LOF) sobre el conjunto de datos normalizado con la etiqueta objetivo. Esta herramienta está dedicada a la detección de anomalías y mide la desviación local de la densidad de una muestra con respecto a sus vecinos, es decir, compara la densidad local de una muestra con las densidades locales de sus vecinos lo que lo hace efectivo para detectar anomalías en conjuntos de datos con densidades variables

y agrupamientos. Este algoritmo entrenado se aplica posteriormente sobre los nuevos casos para identificar si las muestras en estudio son válidas. Cabe destacar que este último enfoque solo ha sido establecido dentro de la primera propuesta.

En el Algoritmo 1 se presenta la evaluación de un contrafactual como se ha explicado en ambas herramientas. Este proceso se lleva a cabo tras completar los preparativos específicos como son el entrenamiento de los algoritmos y la determinación de un valor mínimo de densidad para la prueba de vecinos más cercanos, finalmente se devuelve una indicación sobre si la muestra sometida a estudio se encuentra en una región lo suficientemente densa.

Algoritmo 1: Prueba de densidad sobre un counterfactual (*DensityTest*)

Input: Counterfactual (ct), prueba de densidad ($testDens \rightarrow \{\text{“}lof\text{”}, \text{“}nbrs\text{”}, \text{“}nbrs\_mean\text{”}\}$), modelo para estimar la densidad (D) y mínimo de densidad ($minDens$)
Output: Resultado de la prueba de densidad ($dense$)

```
dense ← False
if testDens = "lof" then
    if D(ct) = 1 then                          ▷ Algoritmo LocalOutlierFactor
        dense ← True
    end if
else
    distances ← D(ct)                          ▷ Algoritmo NearestNeighbors
    if testDens = "nbrs_mean" then
        density ← 1.0/media(distances)
    else
        density ← 1.0/last(distances)
    end if
    if density > minDens then
        dense ← True
    end if
end if
return dense
```

Por último, en lo que respecta a la obtención de la **importancia de las características**, ambas propuestas siguen un procedimiento similar por lo que se va a desarrollar en este apartado. Sin embargo, es importante destacar que aunque el proceso para obtener estos pesos es similar, cada metodología los trabaja de manera diferente. Por lo tanto, cómo se aplican en cada propuesta se explicará en sus respectivos apartados.

Dado que el objetivo principal de generar contrafactuales es alcanzar un escenario con condiciones diferentes sobre las que se habría dado una respuesta distinta, se puede deducir que obtener una estimación de cuáles son las variables críticas que hacen que se consiga este objetivo es una poderosa herramienta para guiar la conversión y además, ofrecer una mejor explicación sobre cuáles son los aspectos que realmente condicionan la muestra. Por este motivo, en ambas propuestas se sigue una perspectiva local donde se obtiene la importancia de las características de la instancia inicial utilizando SHAP (Shapley Additive Explanations) [10], una técnica que asigna pesos a las características de entrada en función de su contribución a la predicción basándose en la teoría de juegos y en los valores de regresión de Shapley. Esta metodología se fundamenta en que cada característica participa en un 'juego'

donde la predicción es el pago y el valor de Shapley determina cómo distribuir equitativamente este pago entre las características. En definitiva, los valores de Shapley reflejan la importancia de cada variable y se calculan formando modelos lineales que consideran todas las combinaciones posibles en una instancia y promediando las diferencias de predicción.

Es importante destacar que SHAP es la técnica más ampliamente utilizada en la literatura y, aunque puede ser menos eficiente en problemas complicados, garantiza explicaciones consistentes sobre interacciones complejas y es aplicable a una variedad muy amplia de modelos. Además, SHAP ofrece varios tipos de explicadores, entre los cuales se ha empleado KernelSHAP que es robusto, ofrece estimaciones estables sobre las contribuciones y es compatible con cualquier tipo de modelo.

Al utilizar el explicador sobre la instancia, se han considerado únicamente casos que presentan la etiqueta objetivo para establecer el contexto ya que esta práctica puede mejorar las explicaciones sobre qué características son más importantes para el rechazo [43]. Además, ha sido limitado a 1000 ejemplos (los cuales son considerados suficientes) con el fin de favorecer a la eficiencia. Como resultado, se obtiene un valor Shapley para cada característica y se calcula su valor absoluto ya que se pretende modificar tanto aquellas que contribuyen negativamente al cambio como aquellas que lo favorecen. A partir de estos valores, se forma un vector de pesos y una lista ordenada sobre cuáles son las características que más contribuyen, excluyendo la posiciones restringidas por el usuario como no modificables.

4.3. Propuesta 1 - Formulación de Contrafactuales

En este apartado, se va a desarrollar y explicar el diseño de la primera metodología presentada como un generador de contrafactuales. Como el método aprovecha diversos enfoques observados en el estado del arte como son la heurística basadas en vecinos y la optimización por descenso de gradiente, la explicación se va dividir en dos secciones. Además, antes de profundizar, se va a exponer un pseudocódigo general del algoritmo para dar una vista previa de cómo funciona el método. En las secciones posteriores se entrara en detalle sobre las decisiones y los procesos internos.

En resumen, primero se dan una serie de preparativos sobre los aspectos ya presentados como fundamentales y, de seguido, se inicia un bucle donde se van a formar tantos contrafactuales como se hayan indicado priorizando resultados donde se modifique el menor número de características posible. A medida que se generan más casos, se aumenta el número de características a considerar, buscando siempre primero obtener ejemplos con pocas diferencias respecto a la instancia de estudio. Para formar cada contrafactual, se crea un contraejemplo inicial sustituyendo las variables destacadas como más importantes por las de muestras similares que presenten la predicción objetivo y después, se optimiza el resultado buscando que sea lo mas similar posible a la instancia original respetando el resto de las propiedades deseables de los contrafactuales y buscando conseguir así las mejores soluciones posibles.

Algoritmo 2: Propuesta 1 - Formulación de Contrafactuales

Input: Conjunto de datos (x, y), instancia de estudio (x_i), categoría objetivo (t), modelo de clasificación (M), prueba de densidad $(testDens \rightarrow \{\text{“}lof\text{”}, \text{“}nbrs\text{”}, \text{“}nbrs\_mean\text{”}\})$, vecinos de la prueba de densidad (n), umbral mínimo de densidad $(denst \rightarrow \{0 \leq$

$dens t \leq 1\}$), número de contrafactuales objetivo (N_{CT}), número máximo de características modificables ($maxN_F$), número máximo de iteraciones ($maxIt$), tasa de aprendizaje (lr), factor de decaimiento ($lrDecay$) y valor mínimo de la tasa de aprendizaje ($minlr$)

Output: Conjunto de muestras contrafactuales (CTs)

$CTs \leftarrow [\,]$
$ShapleyValues \leftarrow SHAP(x|y = t, x_i, M)$ ▷ Algoritmo SHAP
$w \leftarrow abs(ShapleyValues)$
$\hat{w} \leftarrow OrderFeaturesImportance(x, w)$
$sortCT \leftarrow OrderCTDistances(x|y = t, x_i, w)$ ▷ Ver la Ecuación [4.1]
if $testDens =$ "lof" **then**
  $D \leftarrow lof(x|y = t, n)$ ▷ Algoritmo LocalOutlierFactor
  $minDens \leftarrow None$
else
  $D, allDistances \leftarrow nbrs(x|y = t, n)$ ▷ Algoritmo NearestNeighbors
  if $testDens ==$ "$nbrs\_mean$" **then**
    $density \leftarrow 1.0/media(allDistances)$
  else
    $density \leftarrow 1.0/last(allDistances)$
  end if
  $minDens \leftarrow percentile(density, denst * 100)$
end if
$N_F \leftarrow 0$
$nonN_F \leftarrow [\,]$
for $iteration = 1, 2, ..., N_{CT}$ **do**
  $InitialCT \leftarrow SustitionSearch(x_i, sortCT, \hat{w}, ...)$ ▷ Ver el Algoritmo 3
  $w[x_i = InitialCT] \leftarrow 0$
  $OptimicedCT \leftarrow DescendentGradient(InitialCT, x_i, w, ...)$ ▷ Ver el Algoritmo 4
  if $sameDiferences(OptimicedCT, CTs) = False$ **then**
    $CTs \leftarrow CTs \cup OptimicedCT$
  else
    $nonN_F \leftarrow nonN_F \cup N_F$
  end if
end for
return CTs

4.3.1. Heurística Basadas en Vecinos

En este primer proceso se busca conseguir un caso contrafactual siguiendo una estrategia de sustitución de valores entre la instancia original y casos que pertenezcan a la clase objetivo. Posteriormente, esta muestra se va utilizar como punto de partida inteligente en la optimización por descenso de gradiente.

Antes de iniciar la búsqueda, se cuantifica la similitud entre la instancia de estudio normalizada y los casos que pertenecen a la clase objetivo utilizando la distancia GIQR ponderada con la importancia de las características. Esto se realiza con el fin de obtener una lista ordenada de contrafactuales por similaridad prestándose especial atención a las variables relevantes, ya que son las que van a ser empleadas para formar el contraejemplo.

La formulación de cada contrafactual por sustitución se resume en el Algoritmo 3, el cual se presenta al final de la sección tras una explicación completa de todas las operaciones y del proceso desarrollado para conseguirlo.

En primer lugar, se intenta construir la muestra realizando modificaciones considerando solo una única característica, la más importante. En el caso de no lograr un resultado que cumpla con la predicción objetivo o con ciertas restricciones que se van a comentar más adelante, se van añadiendo más características a modificar por orden de contribución y sin incluir aquellas indicadas como no modificables por el usuario. De seguido, sobre cada una de las características seleccionadas se examinan sus relaciones causales directas con el resto para tenerlas en cuenta al realizar el cambio. Este análisis se lleva a cabo desde dos enfoques: identificando si se trata de una variable ficticia binaria dentro de un grupo que representa información categórica al identificar la estructura clásica del nombre de este tipo de características (por ejemplo: color_verde o color_rojo); y realizando un análisis de chi-cuadrado sobre un subconjunto de hasta 1000 muestras (considerado suficiente) para evaluar si existe una relación entre las variables categóricas y la característica seleccionada. Se calcula el coeficiente de contingencia de chi-cuadrado como una medida de la fuerza de asociación y se verifica si es superior a un umbral que, en este caso, se ha establecido en 0.8. Al realizar este último proceso, es importante asegurarse de que las variables categóricas no incluyan ficticias que se refieran a la misma característica global, que las identificadas como dummies de una misma variable sean evaluadas de forma conjunta y que las características de estudio no sean ampliamente dispersas, es decir, que estén limitadas a un número finito de valores.

Una vez se han escogido las características sobre las que realizar las sustituciones, tanto por importancia como por causalidad, se itera sobre los casos previamente ordenados formando contrafactuales modificando las variables en la instancia original. Luego, se comprueba si cumple con los requisitos establecidos de densidad y si el modelo lo clasifica correctamente con la etiqueta objetivo siendo posible limitar el número de pruebas en este paso asumiendo que tras muchos fallos en la predicción del modelo el cambio sobre las características seleccionadas puede no ser suficiente.

Si se cumplen las condiciones anteriores, se examina la diferencia de posiciones sustituidas con contrafactuales formados previamente para asegurar dispersión en los resultados y, si esta comprobación también es satisfactoria, se establece el contrafactual como una solución válida y se elimina el caso utilizado de la lista de muestras ordenadas como otra medida a favor de la dispersión.

Por otro lado, si no se cumple en ningún caso con las restricciones establecidas, se aumentan las características a considerar y se establece esa combinación como imposible.

Algoritmo 3: Contrafactual inicial por sustitución (*SustitutionSearch*)

Input: Instancia de estudio (x_i), muestras contrafactuales reales ordenadas por distancia ($sortCT$), características ordenadas por importancia ($\hat{w}$), categoría objetivo (t), modelo de clasificación (M), contrafactuales previos (CTs), número máximo de características modificables ($maxN_F$), características modificables (N_F), lista de cambios imposibles ($nonN_F$), prueba de densidad ($testDens \rightarrow \{$"lof", "$nbrs$", "$nbrs\_mean$"$\}$), modelo para estimar la densidad (D) y mínimo de densidad ($minDens$)

Output: Contrafactual inicial por sustitución de variables (x_i')

while $True$ **do**
  $N_F \leftarrow N_F + 1$
  while N_F **in** $nonN_F$ **or** $N_F > maxN_F$ **do**
    if $N_F > maxN_F$ **then**
      $N_F \leftarrow 1$
    else
      $N_F \leftarrow N_F + 1$
    end if
  end while
  $modF \leftarrow \hat{w}[: N_F]$
  $modF \leftarrow modF \cup \{RelatedDummies(modF) + Chi2Contingency(modF) > 0.8\}$
  for ct **in** $sortCT$ **do**
    $x'_i \leftarrow x_i$
    $x'_i[modF] \leftarrow ct[modF]$
    $density \leftarrow DensityTest(x'_i, testDens, D, minDens)$ ▷ Ver el Algoritmo 1
    $y'_i \leftarrow M(x'_i)$
    if $density = True$ **and** $y'_i = t$ **then**
      if $sameDiferences(x'_i, CTs) = False$ **then**
        return x'_i
      end if
    end if
  end for
  $nonN_F \leftarrow nonN_F \cup [N_F]$
end while

4.3.2. Optimización por Descenso de Gradiente

En una segunda etapa, se formula y aplica una estrategia de optimización por descenso de gradiente sobre el contrafactual construido por sustitución. Este proceso tiene como objetivo ajustar la solución minimizando la distancia con el caso de estudio y, en consecuencia, buscando una mayor similitud.

Como paso previo, se forma un vector de pesos en el que se mantienen únicamente los valores que corresponden a las características más importantes, excluyendo las similares con la instancia original y las variables binarias, con el fin de ponderar cambios coherentes. Además, también se establecen una serie de restricciones sobre las posibles modificaciones que se pueden dar en el proceso de optimización. En este punto, se definen los valores mínimos y máximos para cada caso, y, en lo que respecta a las categóricas, una lista de valores únicos disponibles. Por otra parte, como se ha considerado al formar los contrafactuales por sustitución, se van a tener en cuenta aquellas relacionadas directamente según los resultados del análisis de chi-cuadrado. En este último paso, se ha obtenido la moda sobre las característica asociada en cada alternativa con el fin de corregirla según varié la principal.

Una vez que todos los componentes necesarios están listos, se implementa el algoritmo de descenso de gradiente que se expone de forma resumida en el Algoritmo 4 al final de su explicación completa. En primer lugar, se calcula la distancia inicial entre ambas muestras, aumentada en 1 para establecer un punto de partida adecuado en las primeras comparaciones. Además, se verifica que haya elementos distintos de 0 dentro del vector de pesos ya que, en caso negativo, no se realizarán modificaciones y

se devolverá el contrafactual sin cambios. Seguidamente, se inician un número finito de iteraciones en las que se busca minimizar la función objetivo la cual se corresponde a la función de distancia normalizada y ponderada como se representa en la ecuación (4.2) donde x_i es la instancia original, x_c la muestra contrafactual, r el rango de normalización como se ha presentado al definir la métrica de distancia (IQR) y w es el vector de pesos.

$$L(x_i, x_c, r, w) = \sqrt{\left(\frac{(x_i - x_c)}{r} \cdot w\right)^2} \tag{4.2}$$

Con este propósito, se calcula el gradiente de esta función resolviendo las derivadas parciales respecto a las variables independientes. En esta problemática, dicho cálculo se realiza exclusivamente con respecto a las características del contrafactual, ya que son los elementos que se pretenden actualizar en las iteraciones del algoritmo. Esta operación implica resolver la ecuación (4.3).

$$\frac{dL}{dx_c} = \frac{w^2 \cdot (x_c - x_i)}{r^2 \cdot \sqrt{\left(\frac{(x_i - x_c)^2}{r^2} \cdot w^2\right)}} \tag{4.3}$$

El gradiente resultante se aplica sobre la muestra contrafactual condicionado por una tasa de aprendizaje que controla la magnitud del cambio para evitar la no convergencia en un mejor resultado o sobrepasar el mínimo de la función objetivo. Luego, se aplican las restricciones previamente construidas sobre las características del caso para asegurar la validez de sus valores y generar un contrafactual procesado.

Por último, antes de finalizar la iteración, se realizan ciertas comprobaciones sobre el contrafactual resultante para decidir cómo continuar con el proceso. En resumen, se calcula la distancia con el nuevo contrafactual sin procesar y se verifica que sea menor que la distancia previa, se comprueba que la estimación de la densidad de la muestra, tanto procesada como sin procesar, esté por encima del valor mínimo y, se confirma si la predicción del modelo sobre el nuevo contrafactual procesado coincide con la objetivo. Si todas estas condiciones se cumplen, se continúa con la siguiente iteración intentando mejorar el resultado. En caso contrario, se modifica el vector de pesos del cálculo de gradiente descartando de manera sucesiva las características más importantes con el fin de ajustar en las próximas iteraciones aquellas variables de menor fuerza en la métrica de distancia. Esto se hace con el fin de evitar, por ejemplo, que las modificaciones tempranas sobre la característica más importante dominen el cambio formando el contrafactual sin haber ajustado lo máximo posible las otras características.

Una vez realizadas todas las pruebas sobre la instrucción anterior, se reduce la tasa de aprendizaje según un factor de decaimiento, se almacena este contrafactual como posible resultado y se continúa iterando, repitiendo el mismo proceso hasta que se sobrepase una tasa de aprendizaje mínima o se den el número máximo de iteraciones.

Algoritmo 4: Optimización del contrafactual (*DescentGradient*)

Input: Instancia de estudio (x_i), contrafactual (ct), categoría objetivo (t), modelo de clasificación (M), pesos (w), número máximo de iteraciones $(maxIt)$, tasa de aprendizaje (lr), factor de decaimiento $(lrDecay)$, valor mínimo de la tasa de aprendizaje $(minlr)$, prueba de densidad $(testDens \rightarrow \{\text{“}lof\text{”}, \text{“}nbrs\text{”}, \text{“}nbrs\_mean\text{”}\})$, modelo para estimar la densidad (D) y mínimo de densidad $(minDens)$

Output: Contrafactual optimizado (ct)

$grad\_w \leftarrow w$
$newDist \leftarrow distance(x_i, ct, w) + 1$ ▷ Ver la Ecuación [4.1]
for $iteration = 1, 2, ..., maxIt$ **do**
  $lastCT \leftarrow ct$
  $lastDist \leftarrow newDist$
  $grad \leftarrow gradient\_ct(x_i, ct, grad\_w)$ ▷ Ver la Ecuación [4.3]
  $ct \leftarrow ct - (lr * grad)$
  $newDist \leftarrow distance(x_i, ct, w)$ ▷ Ver la Ecuación [4.1]
  $density \leftarrow DensityTest(ct, testDens, D, minDens)$ ▷ Ver el Algoritmo 1
  $pred \leftarrow M(ct)$
  if $density = False$ **or** $pred \neq t$ **or** $newDist > lastDist$ **or** $all(grad\_w) = 0$ **then**
    $ct \leftarrow lastCT$
    if not $all(grad\_w) = 0$ **then**
      $grad\_w[max] \leftarrow 0$
    else
      $grad\_w \leftarrow w$
      $lr \leftarrow lr * lrDecay$
      if $lr < minlr$ **then**
        break
      end if
    end if
  end if
end for
return ct

Como ultimo paso, se verifica la dispersión del contrafactual sintético resultante con los ya reservados. Esto se debe a que, en la optimización, las variables pueden ajustarse hasta alcanzar valores similares a los de un caso ya formado (exceptuando decimales redundantes). Por ejemplo, si tenemos un caso anterior trabajado sobre 3 variables y ajustamos una nueva característica en un caso actual, es posible que la optimización de esta nueva de como resultado un valor igual al de la instancia inicial por lo que el resto van a llegar hasta un mismo limite que el caso anterior. Es importante destacar que esta excepción solo se ha experimentado en un caso y se debe a la modificación de características continuas irrelevantes en la predicción y a una mala distribución de la característica. Asimismo, también se intuye que se podría dar por restricciones de densidad muy pobres que permitan cualquier alternativa las cuales no son posibles en los algoritmos formados en el trabajo.

Con este propósito, se verifica para cada variable optimizada si al sustituirla por su valor en la instancia inicial se mantiene la predicción y se cumplen las restricciones de densidad. En caso de darse esta condición, se sustituye el valor de la instancia inicial en un contrafactual conformado para esta prueba. Finalmente, se analizan las diferencias con el caso original y se compara con el resto de contrafactuales existentes. Si se cumple con la dispersión, se ajustan los decimales del contrafactual para garantizar la misma estructura que los casos del conjunto de datos y se almacena como un nuevo contrafactual final. En caso contrario, se marca esta combinación como no modificable y se continúa formulando un nuevo contrafactual.

4.4. Propuesta 2 - Generador Incremental de Contrafactuales

En este apartado, se va a exponer el diseño e implementación de la segunda propuesta que tiene como objetivo trazar un camino de casos reales desde la instancia de estudio hasta un contrafactual para así formar una guía de cambios necesarios y factibles sobre las características de mayor contribución y conseguir una transición de clase. Este proceso se basa en la construcción de un grafo atendiendo a una serie de restricciones de distancia y densidad ajustadas por el usuario a partir del conjunto de datos y la instancia indicada.

En un primer paso, se forma una matriz cuadrada donde se cuantifica la distancia entre pares de puntos de datos utilizando la misma métrica ponderada que la propuesta anterior pero siguiendo una estrategia diferente en lo que respecta a como se considera la importancia de las características. Dado que el objetivo es resaltar o buscar un gran cambio sobre las características de mayor contribución respecto al resto, se han modificado los pesos de manera que los más importantes contribuyan menos al cálculo. En resumen, se han trasformado según la formula $w = -w+1$. Además, las variables indicadas como no modificables por el usuario se han establecido con un peso nulo para no considerar sus modificaciones al formar el camino. Por otro lado, también se mide la densidad de cada punto teniendo en cuenta la distribución de todo el conjunto de datos, empleando cualquiera de los dos enfoques basados en el método de vecinos más cercanos.

Estas estimaciones van a ser esenciales para formular el grafo pero también se van a emplear para determinar el umbral de distancia máxima de conexión y la densidad mínima admitida según las indicaciones del usuario.

Una vez los preparativos están listos, se crea un grafo agregando todos los casos del conjunto de datos como nodos y se itera sobre cada uno para establecer conexiones con el resto, siempre y cuando ambos nodos no pertenezcan a la clase objetivo, cumplan con el mínimo de densidad y la distancia entre ellos no supere el máximo permitido. Además, para asegurar que sea posible obtener un resultado para la instancia inicial, su nodo correspondiente siempre establecerá aristas aunque no cumpla con el mínimo de densidad. Este proceso resulta en la formación de un grafo con aristas en las que se establece la distancia como su peso asociado.

Seguidamente, se ejecuta sobre el grafo un proceso basado en el algoritmo de Dijkstra [46], el cual tiene como objetivo calcular las distancias y caminos más cortos desde el nodo de inicio hasta cualquier otro nodo del grafo. Esto se logra mediante la formación de una cola de prioridad, donde en cada iteración se selecciona el nodo con la distancia más corta.

Los resultados obtenidos se filtran para incluir solo aquellos caminos que lleguen a un nodo con la etiqueta objetivo y se ordenan de forma ascendente según la distancia total. Después, se itera sobre la lista hasta que el modelo prediga el contrafactual en la categoría objetivo y, opcionalmente, se verifica si cumple con un umbral de densidad establecido considerando únicamente el conjunto de la etiqueta objetivo.

Como solución se obtiene un conjunto de muestras que incluye: la instancia inicial, una serie de muestras sucesivas de su misma clase en las que se puede observar una tendencia de cambio y el contrafactual más similar según las conexiones del grafo.

Una vez explicado en detalle el desarrollo de la propuesta, se va a exponer el pseudocódigo del algoritmo resultante para dar una visión más clara de su funcionamiento. Primero se realizan una serie de preparativos donde destacan el calculo de las distancias ponderadas entre casos y la estimación de su densidad ya que seguidamente, se van a emplear para limitar las conexiones del grafo según un umbral establecido por parámetro. De seguido, se obtiene el camino de nodos hasta la instancia contrafactual alcanzable más cercana, que se devuelve como resultado.

Algoritmo 5: Propuesta 2 - Generador incremental de contrafactuales

Input: Conjunto de casos (x, y), instancia de estudio (x_i, y_i), categoría objetivo (t), modelo de clasificación (M), prueba de densidad $(testDens \rightarrow \{\text{“}nbrs\text{”}, \text{“}nbrs\_mean\text{”}\})$, vecinos de la prueba de densidad (n), umbral mínimo de densidad $(denst \rightarrow \{0 \leq denst \leq 1\})$ y umbral máximo de distancia $(dist \rightarrow \{0 \leq dist \leq 10\})$

Output: Conjunto de muestras que guía un cambio desde la instancia de estudio hasta una situación contrafactual $(PathCT)$

$G \leftarrow Graph()$
$ShapleyValues \leftarrow SHAP(x|y = t, x_i, M)$ ▷ Algoritmo SHAP
$w \leftarrow abs(ShapleyValues)$
$D, allDistances \leftarrow nbrs(x, n)$ ▷ Algoritmo NearestNeighbors
if $testDens ==$ "$nbrs\_mean$" **then**
  $densities \leftarrow 1.0/media(allDistances)$
else
  $densities \leftarrow 1.0/last(allDistances)$
end if
$x \leftarrow [x_i] \cup x$
$y \leftarrow [y_i] \cup y$
$xWeighted \leftarrow x * (1 - w)$
$distances \leftarrow distance(xWeighted)$ ▷ Ver la Ecuación [4.1]
$maxDist \leftarrow percentile(distances, dist * 10)$
$minDens \leftarrow percentile(densities, denst * 100)$
$G \leftarrow node(0, 1, ..., len(x))$
for $i = 0, 1, ...len(x)$ **do**
  for $j = i + 1, i + 2, ..., len(x)$ **do**
    if not $(y[i] = t$ **and** $y[j] = t)$ **then**
      if $(i = 0$ **or** $densities[i] > minDens)$ **and** $distance[i, j] \neq 0$ **then**
        if $densities[j] > minDens$ **and** $distance[i, j] \leq maxDist$ **then**
          $G \leftarrow G.addEdge(i, j, w = distance[i, j])$
        end if
      end if
    end if
  end for
end for
$NodeDistances, Paths \leftarrow dijkstra(G, x_i)$ ▷ Algoritmo Dijkstra
$DistCT, PathCT \leftarrow NodeDistances|y = t, Paths|y = t$
$sortPathCT \leftarrow OrderPaths(PathCT, DistCT)$
for $PathCT$ **in** $sortPathCT$ **do**
  if $M(PathCT[-1]) = t$ **then**
    return $PathCT$
  end if
end for

Para concluir, como se mencionó al presentar la propuesta, de forma adicional se le da al usuario la posibilidad de ajustar el camino lo máximo posible a la situación de la instancia de estudio. Este enfoque guarda mucha similitud con la generación de contrafactuales por sustitución de la propuesta anterior, ya que consiste en reemplazar las variables de mayor contribución y sus relaciones causales directas hasta lograr un camino plausible en el contexto de los datos. Entonces, dado que ambas prácticas son muy similares, se van a comentar únicamente ciertos aspectos en los que difieren: se forma un caso para cada nodo del camino sobre la instancia inicial, se verifica que todos cumplen con el mínimo de densidad estipulado y, al terminar de formular el camino, se comprueba que el modelo sea capaz de predecir la categoría objetivo sobre la muestra final. En caso de que no se cumplan estas condiciones, se aumentan las características a considerar al realizar una siguiente iteración.

Capítulo 5

Experimentación y Resultados

En este capítulo se va a desarrollar el entorno experimental y las pruebas realizadas para explorar y validar las propuestas explicadas en la sección anterior. Esta parte desempeña un papel crucial al poner a prueba los algoritmos construidos y evaluar la efectividad de las soluciones.

Para lograr este objetivo, se han llevado a cabo una serie de ejecuciones cuidadosamente planificadas con las que abordar distintos tipos de situaciones comunes en el análisis de datos tabulares que pueden suponer dificultades a la hora aplicar los algoritmos de formulación de contrafactuales. Para ello, primeramente se han seleccionado una serie de bases de datos considerando aspectos como su frecuencia de uso en este tipo de estudios o por presentar características de interés. Seguidamente, se han construido modelos de clasificación para cada problemática y se han aplicado las propuestas variando los distintos parámetros que puede establecer un usuario. Finalmente, se ha evaluado si los resultados obtenidos han cumplido los objetivos y se han comparado con soluciones de otros algoritmos ya establecidos.

5.1. Bases de Datos

Siguiendo lo introducido, primeramente se va a tratar el proceso de búsqueda y selección de bases de datos destinadas a la experimentación donde se ha explorado sobre los conjuntos más frecuentemente utilizados en trabajos ya establecidos como son los vistos en el estado del arte.

Antes que nada, es relevante mencionar la revisión realizada por Verma et al. [13] en 2022 sobre las explicaciones contrafactuales en general. Aunque la mayor parte de los trabajos expuestos no tratan temas como la importancia o la selección de características, en un apartado del estudio se exponen más de 30 conjuntos de datos tabulares comúnmente empleados en la literatura. Teniendo en cuenta los más destacados, se van a presentar en la Tabla 5.1 aquellos que coincidan con los observados en algunos de los trabajos expuestos en el estado del arte. Además, también se van a incluir otros por verse repetidos o de interés.

Artículo	Conjuntos de Datos
Guidotti [14]	Adult income, COMPAS, German Credit, HELOC/FICO
Grath et al. [19]	HELOC/FICO
Jia et al. [25]	MIMIC
Rathi [27]	Iris, Mobile Features, Wine
White y Garcez [28]	Adult income, Breast Cancer, Default of Credit, Iris, Pima Diabetes
Keane et al. [29]	Abalone, Ecoli, Liver, Wine, Yeast
Le et al. [32]	biodeg, cancer92, cancer95, magic, mfeaf, musk, phoneme, Diabetes, segment, spam
Dandl et al. [33]	German Credit
Wiratunga et al. [35]	Adult income, Alcohol, Credit, LendingClub
Cho y Shin [39]	HMEQ, Taiwanese
Mothilal et al. [42]	Adult income, COMPAS, German Credit, LendingClub

Tabla 5.1: Conjuntos de datos utilizados en los artículos.

Al observar la tabla expuesta, se puede ver que aquellos que se repiten más de una vez son:

- Adult income (aparece en 3 artículos).
- German Credit (aparece en 3 artículos).
- HELOC/FICO (aparece en 2 artículos).
- COMPAS (aparece en 2 artículos).
- Iris (aparece en 2 artículos).
- Wine (aparece en 2 artículos).
- LendingClub (aparece en 2 artículos).

De entre estos conjuntos, se han escogido aquellos dos que tienen una mayor frecuencia de repetición: Adult Income y German Credit. Por otra parte, además de los seleccionados desde la literatura, se ha hecho una búsqueda propia con el fin de abordar ciertos aspectos sobre los anteriores y realizar pruebas novedosas con diferentes objetivos. Estos conjuntos son Heart Disease, Diabetes Health Indicators y Student Performance.

Se van a comentar todas las bases de datos seleccionadas, además de aspectos y decisiones generales en la preprocesado inicial de cada una:

Heart Disease [47]: El conjunto de datos Heart Disease y específicamente la base de datos de Cleveland, se trata de un conjunto ampliamente popular en el ámbito de datos tabulares y tiene su origen en el Instituto de Cardiología y Cirugía Torácica de Cleveland. Comprende 76 atributos de información médica sobre 303 pacientes recopilados por médicos durante la década de 1980 aunque generalmente, se utilizan solo 14 de ellos debido a su utilidad y relevancia clínica. De estos 14 atributos, 7 son categóricos y el resto son datos numéricos. En el contexto de uso de contrafactuales, un posible caso de uso podría ser destacar los factores que contribuyen a que un individuo específico padezca una enfermedad cardíaca.

Este conjunto de datos ha sido escogido por ser popular, compacto, variado y fácil de entender y manejar, lo que lo hace útil para realizar demostraciones y pruebas sobre los algoritmos.

Adult Income [48]: El conjunto de datos Adult Income (también visto como Census Income o Adult) está compuesto por datos personales recogidos por Barry Becker a partir de la base de datos del Censo de Estados Unidos de 1994. La tarea de predicción se trata de una clasificación binaria que pretende determinar si la persona tiene un ingreso superior o inferior a $50,000 al año basándose en 14 atributos que incluyen propiedades demográficas, educativas y personales. De estos atributos, 8 son variables categóricas, 6 son numéricas y en total se tienen 32561 casos. Como parte del preprocesamiento, se puede destacar que se han eliminado los datos de 2699 individuos por valores faltantes. Para terminar, un posible ejemplo del uso de contrafactuales para este problema podría ser indicar cambios en las circunstancias de un individuo para lograr un salario más alto.

German Credit [49]: El conjunto de datos de German Credit presentado por Hans Hofmann está compuesto por datos descriptivos de solicitantes de préstamos a partir de los cuales se pretende predecir si se le considera a una persona como un buen o mal riego crediticio. En el presente trabajo, se va a utilizar el conjunto original que presenta 1000 instancias y valores para 20 atributos de los cuales 7 son numéricos y 13 son categóricos. Como preporcesamiento, únicamente se ha convertido la salida objetivo de (1,2) a (0,1) con el fin de mejorar los resultados y la interpretación posterior. En este caso, un contraejemplo podría ayudar a un solicitante a entender que aspecto debe modificar para que se le conceda el préstamo.

Diabetes Health Indicators [50]: El conjunto de datos de Diabetes Health Indicators proviene de los resultados de una encuesta telefónica recopilada por los Centros para el Control y la Prevención de Enfermedades en Estados Unidos. Más concretamente, esta encuesta conocida como el Sistema de Vigilancia de Factores de Riesgo Conductuales, se repite anualmente desde el año 1984 agrupando información de 400000 estadounidenses sobre comportamientos de riesgo, condiciones crónicas y el uso de servicios preventivos. Uno de los datos recolectados es el diagnóstico de diabetes el cual es el enfoque principal para la clasificación en este problema. En el presente trabajo, se va a utilizar un conjunto de datos ya limpio y consolidado del disponible en el año 2015 que consta de 2 clases objetivo (no diabético o diabético) y 21 variables de características relevantes para este análisis, donde 18 corresponden con categorías. Cabe destacar que este conjunto ha sido escogido por ser significativamente más grande que el resto presentando un total de 253680 casos.

En este contexto, se podría hacer uso de contraejemplos para determinar que comportamientos debería cambiar una persona prediabética para mejorar su situación.

Student Performance [51]: El conjunto de datos de Student Performance (que resume el nombre original de Predict Students' Dropout and Academic Success) proporciona información sobre estudiantes matriculados en diversos programas universitarios ofrecidos por una institución de educación superior. Este conjunto incluye datos del momento de la inscripción, que abarcan factores socioeconómicos, demográficos y la trayectoria académica hasta la fecha actual, así como el desempeño del estudiante durante el primer y segundo periodo del curso. Pues bien, a partir de estos datos se busca predecir la probabilidad de deserción o éxito académico de cada estudiante. Para lograr este objetivo, se han establecido tres categorías que corresponden al es-

tado del alummno al final de la duración normal del curso: si abandonaron el curso sin completarlo, si están actualmente matriculados o si han completado y obtenido el título. Esto convierte el problema en una clasificación multiclase, motivo por el que ha sido escogido para trabajarse en la experimentación. Además, se destaca en la presentación de los datos que se ha llevado a cabo un riguroso preprocesamiento para manejar anomalías, valores atípicos y faltantes por lo que no se ha realizado ninguna modificación inicial. El conjunto final está compuesto por 4424 instancias con 36 atributos de entre los cuales 17 son categóricos, 16 son numéricos y 3 son la variable objetivo en formato one-hot encoding asignando un vector binario a cada categoría. Para terminar, como ejemplo del uso de contrafactuales sobre este problema, se podría destacar el apoyo a un alumno indicándole como mejorar su rendimiento académico lo que puede contribuir a reducir la deserción y el fracaso académico en la educación superior.

Por último, se han realizado una serie de transformaciones comunes a todos los conjuntos de datos anteriores con el fin de hacerlos compatibles y mejorar el rendimiento de los modelos que se van a formar en el siguiente apartado. Primeramente, todas las etiquetas de texto o categóricas se han convertido en valores numéricos siguiendo dos estrategias: las características categóricas nominales se han transformado en variables binarias ficticias y las ordinales se han representado como valores numéricos en una escala. De seguido, se han estandarizado los datos de todas aquellas variables no binarias, lo que implica restar la media y dividir por la desviación estándar en cada característica.

5.2. Modelos de Clasificación

En este apartado, se va a presentar el desarrollo de cada modelo construido para resolver los problemas de clasificación expuestos sobre las bases de datos y establecer así, un componente imprescindible para la experimentación. Se abordará el manejo de los datos, la arquitectura seleccionada tras un exhaustivo proceso de pruebas y los resultados obtenidos para su evaluación y selección frente a otras ejecuciones.

Cabe destacar que todos los modelos construidos se tratan de redes neuronales aunque, en algunos conjuntos (sobre todo en los más reducidos), se podrían obtener posiblemente resultados mejores o similares empleando otros algoritmos propios del aprendizaje automático. Esta decisión se debe a su popularidad actual frente al resto de tecnologías y su capacidad de aprender representaciones jerárquicas y características relevantes lo que las hace particularmente adecuadas para la tarea de modelado de datos tabulares y al capturar relaciones no lineales y patrones complejos.

Para una mejor organización, primeramente se va a explicar de forma general el proceso y las decisiones que se han tomado a la hora de construir y validar los modelos de red neuronal ya que se han desarrollado siguiendo los mismos pasos. Luego, se van a resaltar los detalles específicos de diseño para cada problemática y los resultados obtenidos.

5.2.1. Esquema General del Proceso

Antes de construir los algoritmos, en primer lugar se ha examinado el desbalance de clases. Como se puede ver en la Figura 5.1 donde se resume la distribución de

categorías de cada conjunto, la mayor parte de estos presenta un claro desequilibrio de muestras entre categorías lo cual puede influir en un sobreajuste haciendo que el algoritmo favorezca a la mayoritaria y se pierda información de la minoritaria. Para solucionar esta desigualdad y mejorar la eficiencia de los modelos, se ha explorado sobre distintas estrategias de balanceo utilizadas comúnmente para mejorar los problemas de proporción:

- Sub-muestreo: disminuir el número de casos de la clase mayoritaria descartando muestras de forma aleatoria hasta alcanzar un equilibrio entre clases.
- Sobre-muestreo: aumentar el número de casos de la clase minoritaria a partir de aplicar técnicas de interpolación y extrapolación para generar nuevas muestras sintéticas.
- Muestreo híbrido: se combinan las dos estrategias anteriores aplicando primeramente sub-muestreo sobre la clase mayoritaria y de seguido, sobre-muestreo de la minoritaria.

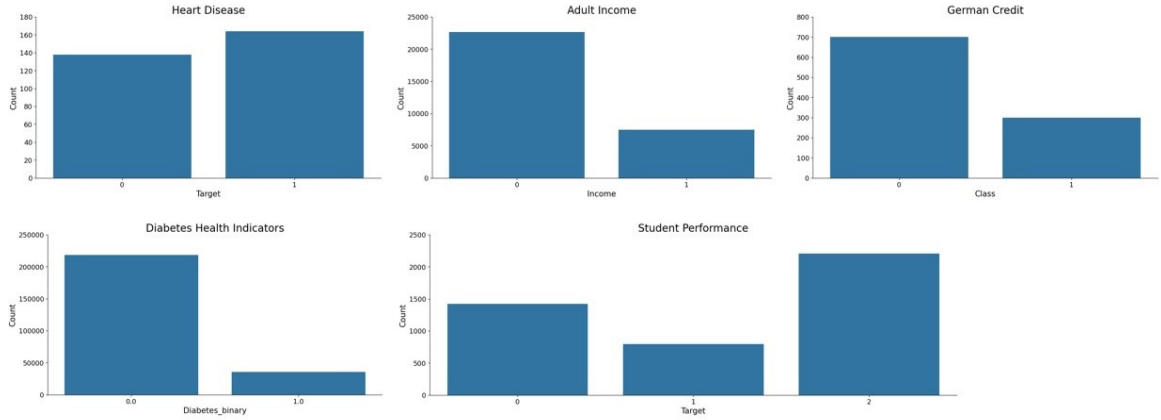


Figura 5.1: Distribución de categorías sobre cada conjunto.

Además, cabe destacar que antes de aplicar cualquier tipo de estas transformaciones, se ha separado un conjunto reducido sobre el que realizar una prueba final con datos que no hayan sido empleados en el proceso de aprendizaje lo más fiel posibles al conjunto original.

Por otra parte, en lo que respecta al algoritmo de clasificación, se han formado redes neuronales como una secuencial lineal de capas donde la salida de cada una se convierte en la entrada de la siguiente. La arquitectura de un modelo de este tipo, se compone de tres partes claramente distinguibles: una capa de entrada que recibe los datos, las capas ocultas que aprenden y extraen relaciones complejas de los datos de entrada y una capa de salida que produce las predicciones del modelo. Esta última, en el contexto de una tarea de clasificación, se configura como una capa densa donde se determina la función de activación y el número de neuronas dependiendo del problema que puede ser binario o multiclase.

Dentro de las capas ocultas, se han empleado capas densas con un número de neuronas elegido a partir de la experimentación y el inicializador de pesos establecido por defecto, *glorot_uniform*. Estas capas realizan operaciones de transformación lineal en los datos, lo que permite a la red aprender representaciones de los mismos.

De seguido y dependiendo de la problemática, se han añadido capas de normalización por lote para mejorar la estabilidad y acelerar el entrenamiento asegurando que las activaciones tengan una media cercana a cero y una desviación estándar próxima a uno. Luego, se ha aplicado una función de activación ReLU (Rectified Linear Unit) para introducir no linealidad en la red, siendo la elección más común en capas ocultas debido a sus ventajas en términos de eficiencia, velocidad de entrenamiento y rendimiento final. Esta función aborda el problema del gradiente evanescente y permite aprender representaciones jerárquicas y dispersas de los datos. Por último, con el fin de mejorar los resultados y la capacidad de generalización, en ciertos casos se ha añadido la técnica regularización Dropout que se fundamenta en la idea de apagar de aleatoriamente un porcentaje especificado de neuronas durante un paso de entrenamiento para introducir variabilidad.

Una vez definida la arquitectura del modelo, este se ha compilado especificando el algoritmo de optimización responsable de ajustar los pesos y sesgos, la función de pérdida a partir de la que calcular la discrepancia y las métricas que se mostrarán durante el entrenamiento. Sobre estas configuraciones, se ha elegido el optimizador Adam que combina las ventajas del impulso y de los ritmos de aprendizaje adaptativos, y suele demostrar una convergencia más rápida y mejor rendimiento en una amplia gama de tareas. Además, se ha especificado una tasa de aprendizaje inicial que se ha ido disminuyendo siguiendo como estrategia de monitorización el parámetro de pérdida o accuracy del conjunto de validación. Asimismo, también se ha implementado un método para almacenar el modelo en un archivo externo después de cada época en la que este parámetro mejore.

Por otro lado, también se han ajustado hiperparámetros como el número de épocas de entrenamiento o el tamaño de lote que se corresponde a la cantidad de muestras de datos utilizadas en cada paso de actualización de los pesos.

Estos modelos han sido entrenados siguiendo una estrategia de validación cruzada donde se divide de manera desordenada el conjunto de datos reservado para el entrenamiento en pliegues y, sobre estos, se realizan múltiples iteraciones de entrenamiento y evaluación considerando diferentes combinaciones. Este proceso resulta en tantos modelos como pliegues se hayan especificado los cuales se han evaluado por separado utilizando diversas métricas. Luego, se ha calculado la media de estos resultados para obtener una evaluación más robusta del rendimiento y de su capacidad para generalizar. Las métricas que se han empleado para ofrecer una visión detallada del desempeño incluyen:

- Accuracy: que representa el porcentaje real de valores correctamente clasificados.
- Precisión: que indica el porcentaje de casos clasificados en una categoría específica que realmente pertenecen a esa categoría.
- Exhaustividad (o recall): que se refiere al ratio de casos correctamente clasificados en una categoría con respecto a todos los casos reales.
- Puntuación F1: que combina la precisión y la exhaustividad para proporcionar una medida que indica la capacidad del modelo de minimizar los falsos positivos y negativos.
- Pérdida: que cuantifica la discrepancia entre las predicciones del modelo y los

valores reales del conjunto de datos.

Finalmente, se ha escogido el mejor modelo según los resultados individuales sobre las anteriores métricas y se ha evaluado de forma individual en el conjunto separado al principio del proceso. Para ello, además de los valores ya presentados, se han utilizado figuras como la matriz de confusión que muestra la cantidad de casos bien y mal clasificados por el modelo en cada una de las clases objetivo o las curvas de pérdida y accuracy formadas durante el entrenamiento y útiles para comprender la convergencia y ajustar el modelo de manera efectiva.

5.2.2. Evaluación del Rendimiento

Como se ha introducido, en esta segunda parte se van a exponer los detalles específicos, la evaluación de los algoritmos y el resultado obtenido del modelo finalmente seleccionado sobre un conjunto de datos reservado para realizar esta prueba. Antes de comentar cada caso en particular, en la Tabla 5.2 se ha expuesto un resumen de las métricas obtenidas en la validación cruzada para los distintos problemas.

Modelos	Accuracy	Pérdida	Precisión	Exhaustividad	Punt. F1
Heart Disease	0.84	0.42	0.84	0.88	0.86
Adult Income	0.83	0.36	0.75	0.74	0.75
German Credit	0.83	0.37	0.86	0.78	0.82
Diabetes Health	0.75	0.50	0.69	0.70	0.70
S Performance	0.77	0.59	0.77	0.77	0.77

Tabla 5.2: Resumen sobre las medias de las métricas de cada modelo en la validación cruzada.

Heart Disease es un conjunto de datos pequeño ampliamente utilizado en la comunidad científica para realizar demostraciones y pruebas de algoritmos. Esta base de datos aborda un problema de clasificación binaria que busca determinar el diagnóstico de enfermedad cardiaca en base a datos médicos y, como se puede ver en la Figura 5.1, se trata de un conjunto equilibrado, en el que concretamente se representan 138 individuos sanos y 164 enfermos. De los datos totales, se ha reservado un 15% para la evaluación final.

Una vez los diferentes conjuntos están preparados para ser utilizados, se ha construido y compilado un modelo de red neuronal para resolver este problema de clasificación binaria. La arquitectura final consta de dos capas dense, la primera con una función de activación ReLU seguida de dropout y, la segunda con una función de activación sigmoidea como capa de salida. El resto de las decisiones de diseño que se exponen en la tabla 5.3. Este modelo ha sido entrenado en una validación cruzada de 4 pliegues, por lo que se ha asignado un 25% de los datos para la validación de cada iteración durante el proceso.

Aspectos de diseño	**Modelo**
Tamaño de lote	32
Tasa de aprendizaje	0.001
Neuronas en capas Dense	4, 1
Dropout	0.1
Descenso de tasa de aprendizaje	Factor 0.5 cada 30 épocas sin mejora
Métrica de monitorización	*val_loss*
Función de pérdida	*binary_crossentropy*
Número de épocas	300

Tabla 5.3: Decisiones de diseño en el Modelo de Heart Disease.

Los resultados de la evaluación del modelo se presentan resumidos en la Figura 5.2. Cabe resaltar que las fluctuaciones o irregularidades que presentan las curvas de entrenamiento y validación se deben a la escasez de los datos en el proceso de aprendizaje.

Conjunto de prueba final

Métrica	Valor
Accuracy	0.89
Categoría 0:	
Precisión	0.88
Exactitud	0.83
Puntuación F1	0.86
Categoría 1:	
Precisión	0.90
Exactitud	0.93
Puntuación F1	0.91

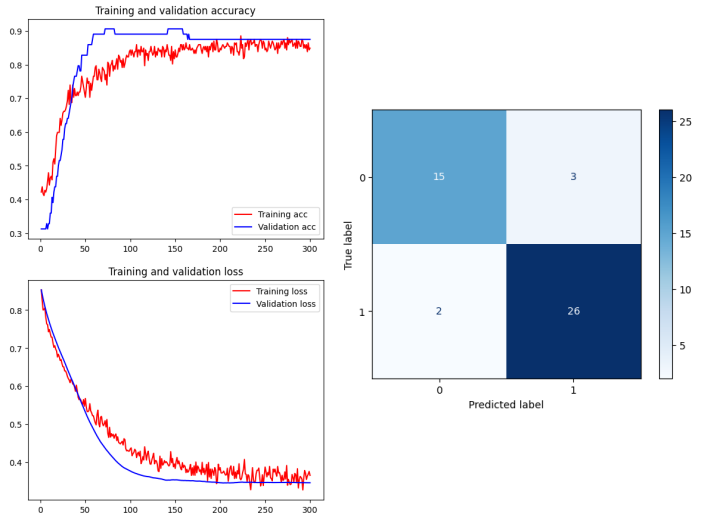


Figura 5.2: Resumen de la evaluación del modelo de Heart Disease.

Adult Income presenta una tarea de clasificación binaria donde se pretende distinguir a los individuos según tienen ingresos superiores o inferiores a $50,000 al año. Como se puede ver en la Figura 5.1, hay una clara diferencia entre los casos que representan a cada clase siendo 22645 los correspondientes a ingresos superiores y 7508 a ingresos inferiores. Esta diferencia supone el triple de instancias para una de las categorías por lo que se ha decidido disminuir el numero de la mayoritaria en un factor de 0.5 siguiendo la estrategia de sub-muestreo. En definitiva, previamente se ha separado un 10 % de las muestras del conjunto original para la evaluación final y, sobre el 90 % restante destinado al entrenamiento, se han descartado el 50 % de casos de la clase dominante.

Sobre este conjunto de datos y después de múltiples pruebas, se ha desarrollado un modelo de clasificación con las especificaciones de la Tabla 5.4 el cual presenta una estructura compuesta por tres capas densas. Las dos primeras se corresponden a las

capas ocultas y tienen una función de activación ReLU, seguida de normalización por lotes y dropout. La tercera capa se corresponde con la salida que, como el caso anterior, presenta una función de activación sigmoidea. Este modelo ha sido entrenado en 5 pliegues, destinando un 20 % de los datos para la validación en cada iteración.

Aspectos de diseño	**Modelo**
Tamaño de lote	1024
Tasa de aprendizaje	0.001
Neuronas en capas Dense	12, 6, 1
Dropout	0.1, 0.1
Descenso de tasa de aprendizaje	Factor 0.9 cada 20 épocas sin mejora
Métrica de monitorización	*val_loss*
Función de pérdida	*binary_crossentropy*
Número de épocas	125

Tabla 5.4: Decisiones de diseño en el Modelo de Adult Income.

Finalmente, se exponen los resultados del modelo al aplicarlo sobre el conjunto de evaluación:

Conjunto de prueba final

Métrica	Valor
Accuracy	0.85
Categoría 0:	
Precisión	0.91
Exactitud	0.88
Puntuación F1	0.90
Categoría 1:	
Precisión	0.69
Exactitud	0.75
Puntuación F1	0.72

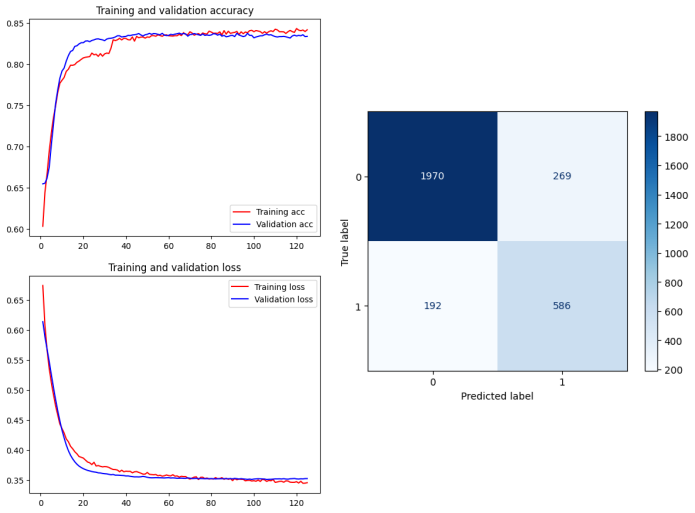


Figura 5.3: Resumen de la evaluación del modelo final de Adult Income

German Credit se trata de una pequeña base de datos donde se pretende clasificar a los solicitantes de un préstamo como un buen o mal riesgo crediticio. Dado la cantidad de instancias del problema y como están repartidas entre las categorías (Figura 5.1), después de haber separado un 10 % de los datos se ha aplicado una estrategia de sobre-muestreo sobre el conjunto destinado a resolver el entrenamiento. Esto se ha hecho con el fin de aumentar el número de instancias de la clase minoritaria hasta igualar la cantidad de casos correspondientes a un buen riesgo, que representan el 70 % de los datos. De seguido, se ha formado un modelo de red neuronal específico para el problema que se ha entrenado sobre 6 pliegues dejando un 17 % de los datos aproximadamente para la validación. La arquitectura final en este caso consta de una capa densa con función de activación ReLU, normalización por lotes y dropout, una

capa de salida con función de activación sigmoidea y las especificaciones expuestas en la Tabla 5.5.

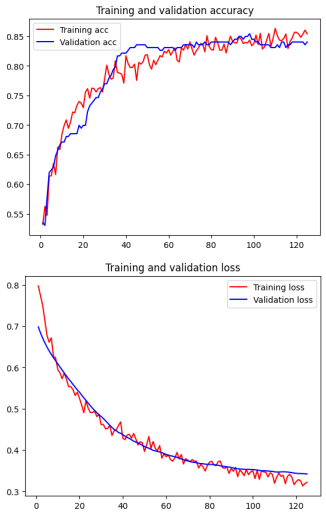
Aspectos de diseño	**Modelo**
Tamaño de lote	128
Tasa de aprendizaje	0.0007
Neuronas en capas Dense	15, 1
Dropout	0.4
Descenso de tasa de aprendizaje	Factor 0.7 cada 20 épocas sin mejora
Métrica de monitorización	*val_loss*
Función de pérdida	*binary_crossentropy*
Número de épocas	125

Tabla 5.5: Decisiones de diseño en el Modelo de German Credit.

Con respecto a la evaluación del modelo, igual que en el primer caso las curvas de entrenamiento y validación presentan una forma irregular debido a la falta de datos. Los resultados están resumidos en la Figura 5.4 como en los problemas anteriores.

Conjunto de prueba final

Métrica	Valor
Accuracy	0.79
Categoría 0:	
Precisión	0.79
Exactitud	0.90
Puntuación F1	0.84
Categoría 1:	
Precisión	0.79
Exactitud	0.59
Puntuación F1	0.68



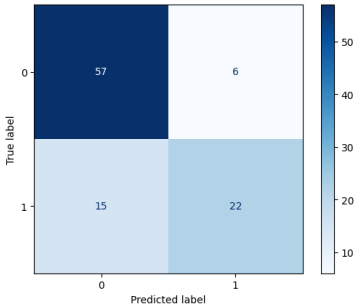


Figura 5.4: Resumen de la evaluación del modelo de German Credit.

Diabetes Health Indicators es un amplio conjunto de casos, pero la cantidad de no diabéticos es significativamente superior, con 218334 frente a 35346 casos diabéticos. Debido a la gran cantidad de instancias, a la enorme diferencia entre ambas categorías y a numerosas pruebas, se ha decidido aplicar la estrategia híbrida de balanceo. En resumen, después de separar un 10% de los datos, se han reducido los casos de la clase mayoritaria con un factor de 0.4 y, de seguido, se ha aumentado la clase minoritaria a un 65% de la nueva cantidad en la clase dominante. Posteriormente, se ha formado un modelo de seis capas dense, donde las capas intermedias comprenden una función de activación ReLU seguida de normalización por lotes. Además, antes de la última capa que utiliza una función de activación sigmoidea para producir la salida del modelo, se ha añadido un paso de dropout. En la Tabla 5.6 se detallan el resto de especificaciones, y el modelo se ha entrenado en 6 pliegues

respecto a la validación cruzada. Los resultados de la evaluación del modelo obtenido se exponen en la Figura 5.5.

Aspectos de diseño	**Modelo**
Tamaño de lote	512
Tasa de aprendizaje	0.001
Neuronas en capas Dense	32, 24, 8, 8, 8, 1
Dropout	0.2
Descenso de tasa de aprendizaje	Factor 0.75 cada 20 épocas sin mejora
Métrica de monitorización	*val_loss*
Función de pérdida	*binary_crossentropy*
Número de épocas	30

Tabla 5.6: Decisiones de diseño en el Modelo de Diabetes Health Indicator.

Conjunto de prueba final

Métrica	Valor
Accuracy	0.78
Categoría 0:	
Precisión	0.94
Exactitud	0.80
Puntuación F1	0.86
Categoría 1:	
Precisión	0.35
Exactitud	0.67
Puntuación F1	0.46

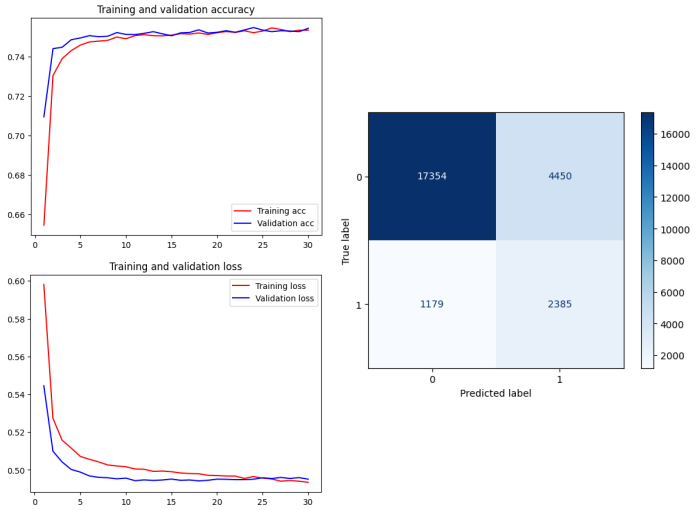


Figura 5.5: Resumen de la evaluación del modelo de Diabetes Health Indicators.

Students Performance incluye datos de 4424 estudiantes matriculados en diversos programas universitarios con el fin de hacer una distinción multiclase en relación con su éxito académico. De entre estos estudiantes, hay 1421 clasificados como abandono, 724 marcados como que continuaran matriculados pero no completaran sus estudios en el tiempo normal del curso y 2206 alumnos que completaran el curso académico con éxito. Aunque el desbalance entre los datos no sea tan significativo, se han obtenido resultados más equilibrados al aumentar a 1100 los casos de matriculados después de separar un 10 % para el conjunto de pruebas final. Seguidamente, se ha formado un modelo de cuatro capas densas con función de activación ReLU en sus intermedias y seguidas de normalización por lotes y dropout. A diferencia del resto, cabe destacar que se ha empleado la función de activación softmax en la capa de salida ya que este se trata de un problema multiclase. El modelo que compone esta arquitectura junto con las decisiones de diseño expuestas en la Tabla 5.7 ha sido entrenado en una validación cruzada de 5 pliegues.

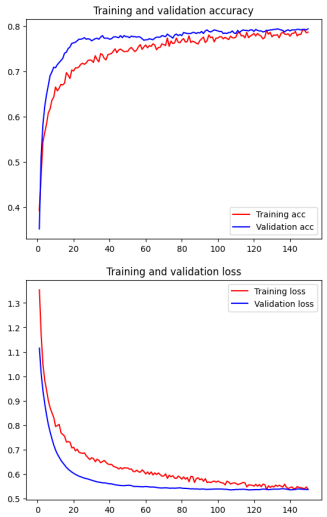
Aspectos de diseño	**Modelo**
Tamaño de lote	256
Tasa de aprendizaje	0.001
Neuronas en capas Dense	34, 34, 34, 1
Dropout	0.2, 0.2, 0.3
Descenso de tasa de aprendizaje	Factor 0.75 cada 25 épocas sin mejora
Métrica de monitorización	*val_accuracy*
Función de pérdida	*categorical_crossentropy*
Número de épocas	150

Tabla 5.7: Decisiones de diseño en el Modelo de Student Performance.

Para terminar, se muestran los resultados obtenidos sobre la evaluación final del modelo en la Figura 5.6.

Conjunto de prueba final

Métrica	Valor
Accuracy	0.79
Categoría 0:	
Precisión	0.90
Exactitud	0.71
Puntuación F1	0.79
Categoría 1:	
Precisión	0.49
Exactitud	0.54
Puntuación F1	0.51
Categoría 2:	
Precisión	0.85
Exactitud	0.88
Puntuación F1	0.87



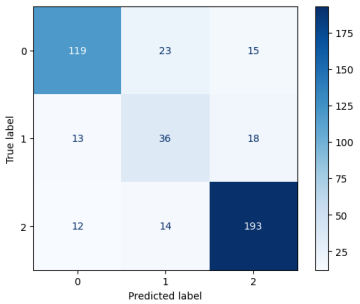


Figura 5.6: Resumen de la evaluación del modelo de Students Performance.

5.3. Algoritmo de Formulación de Contrafactuales

En esta sección, se van a desarrollar los aspectos relacionados con la implementación de la propuesta sobre la formulación de contrafactuales basada en las características más importantes. Se van a presentar los parámetros de entrada que pueden ser indicados por el usuario, tanto obligatorios como opcionales, la salida esperada y todas las pruebas hechas sobre los conjuntos y modelos expuestos.

En cuanto a los parámetros de entrada, es imprescindible que el usuario que utilice la herramienta proporcione un conjunto de datos en formato Dataframe. Este formato es ampliamente utilizado en inteligencia artificial y se asemeja mucho a una tabla de base de datos. Este conjunto debe incluir las características y la clase objetivo de

las instancias tal como se han empleado en el modelo que aborda el problema. Además, se recomienda que sea el mismo que el utilizado para el aprendizaje pero, en caso de no estar disponible, puede ser cualquier conjunto que tenga estas características o una extensión del mismo. Cabe destacar que este atributo influye en aspectos como ciertas restricciones o el método heurístico inicial por lo que su estructura y contenido es crucial para obtener el mejor resultado posible. Asimismo, también es necesario especificar: el caso sobre el que se va a formar el contrafactual como una instancia que no incluya la clase objetivo; la etiqueta que se pretende conseguir con las modificaciones; el nombre de la categoría objetivo en el conjunto de datos; indicaciones sobre si se trata de un problema que presente múltiples clases (True o False); y el modelo sobre el que se resuelve.

También se establecen algunos parámetros por defecto que se recomienda que sean ajustados según las necesidades del usuario:

- El método para establecer restricciones según la densidad de puntos, que acepta indicaciones como *lof*, *nbrs* y *nbrs_mean*. Por defecto *lof*.
- El número de vecinos que se van a emplear sobre la técnica anterior. Por defecto *50*.
- Un umbral de densidad mínima como un valor entre 0 y 1 que es imprescindible si se sigue una estrategia de vecinos más cercanos (*nbrs*). Por defecto *None*.
- El número de contrafactuales dispersos que se quiere obtener. Por defecto *3*.
- El número máximo de características que pueden ser modificadas. Por defecto *10*.
- La tasa de aprendizaje inicial para el algoritmo de descenso de gradiente. Por defecto *0.1*.
- El factor de decaimiento sobre el parámetro anterior. Por defecto *0.1*.
- Un valor mínimo de tasa de aprendizaje que hará de condición de parada. Por defecto *0.0001*.
- El número máximo de iteraciones que se puede dar sobre el método de optimización. Por defecto *1000*.
- Una lista de características que no pueden ser modificadas según las preferencias del usuario. Por defecto vacía.
- Una lista de características categóricas que comprenda tanto las ordinales como las codificadas en variables binarias ficticias (que se deben indicar con su nombre original, por ejemplo, *color* para *color_rojo* y *color_verde*). Por defecto vacía.
- Un número máximo de intentos para predecir la clase objetivo sobre el método heurístico a partir del cual se determina que es imposible formar un contrafactual con las modificaciones de la iteración. Por defecto *500*.

Por otro lado, en lo que respecta a la salida del algoritmo, se obtiene un conjunto en formato Dataframe que contiene todos los contrafactuales sobre la instancia especificada.

Finalmente, se van a exponer y comentar los resultado obtenidos a partir de aplicar el algoritmo sobre instancias aleatorias de los conjuntos de datos expuestos. En lo que respecta a los parámetros de cada ejecución, se van a destacar únicamente aquellos que han sido modificados sobre las restricciones o indicaciones. Estos han sido escogidos según la estructura de los datos, su significado y múltiples pruebas. Además, cabe destacar que para cada problemática se van a formar 4 contrafactuales.

Heart Disease

En esta primera prueba, se va a modificar la instancia expuesta en la Tabla 5.8 que no presenta enfermedad para distinguir qué cambios sobre las características podrían afectar a su estado y derivar en una enfermedad angiográfica. Este ejemplo se va a utilizar para explicar el proceso de evaluación que se va a repetir para cada prueba y por este motivo, se va desarrollar más en profundidad.

age	trestbps	chol	restecg	thalach	oldpeak	ca	sex_1
0.839089	0.364848	0.919336	0.901657	-1.905464	0.739054	2.269926	0.0

cp_1	cp_2	cp_3	fbs_1	exang_1	slope_1	slope_2	thal_1	thal_2	thal_3
0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.0	1.0	0.0

Tabla 5.8: Datos originales de la instancia inicial en Heart Disease.

Como se puede observar, aunque este se trata de un ejemplo sencillo, las transformaciones que hacen que los datos estén estandarizados o separados en diferentes columnas hacen muy difícil su representación y comprensión (sobre todo teniendo en cuenta que hay que hay ejemplos que llegan a presentar 100 columnas de datos). Por este motivo, se van a exponer las muestras ya procesadas como se enseña en la Tabla 5.9 y, en posteriores ejemplos, únicamente se van a mostrar aquellas características que varían entre casos.

age	trestbps	chol	restecg	thalach	oldpeak	ca	sex	cp	fbs	exang	slope	thal
62	138	294	1	106	1.9	3	0	0	1	0	1	2

Tabla 5.9: Datos de la instancia inicial procesada en Heart Disease.

Por otro lado, entre los parámetros de entrada escogidos para ejecutar esta prueba se puede destacar: *nbrs_mean* como método para estimar la densidad sobre *5* vecinos, un umbral de densidad mínima de *0.22* y una restricción de variables categóricas sobre las características *['sex', 'cp', 'fbs', 'restecg', 'exang', 'slope', 'ca', 'thal']*. Al tratarse del primer ejemplo, no se ha indicado ninguna categoría como no modificable. Los resultados obtenidos de ejecutar el algoritmo con los parámetros establecidos se presentan en la Tabla 5.10.

age	trestbps	chol	restecg	thalach	oldpeak	ca	sex	cp	fbs	exang	slope	thal
62	138	294	1	143	1.9	0	0	0	1	0	1	2
62	138	294	1	128	0.5	1	0	0	1	0	1	2
62	138	272	1	119	0.3	1	0	0	1	0	1	2
62	138	261	1	120	1.1	1	0	2	1	0	1	2

Tabla 5.10: Datos de los casos contrafactuales en Heart Disease

Al finalizar una prueba, se van a hacer una serie de comprobaciones y evaluaciones

sobre las características modificadas, las cuales se van a repetir en cada problemática. En cuanto a la similitud con la instancia original, esta se ha definido mediante dos métricas: la distancia GIQR sobre los casos procesados y la distancia GIQR ponderada en los casos sin procesar. En la primera se consideran los valores de las características midiendo la diferencia entre continuos normalizados y comparando las variables categóricas nominales sumando 1 en el caso de ser distintas. Por otro lado, en la ponderada se va a aplicar una métrica similar pero sobre los datos como los trabaja el algoritmo y aplicando e invirtiendo los pesos, dando así una menor contribución a aquellas características más importantes. Este último enfoque refleja la similitud de los casos haciendo ver más parecidos aquellos que varíen únicamente en características relevantes. También, se van a contabilizar las características que han sido modificadas sobre la instancia original para lograr el cambio y cuantas de estas pertenecen a las presentadas como más importantes analizando el top 3 y el top 5 de los atributos obtenidos mediante la técnica SHAP. Por último, se van a comentar puntos destacables sobre los resultados en relación con la comparativa entre casos, el sentido lógico de las explicaciones y con los valores de las características.

Distancia GIQR	2.42	2.30	2.42	3.20
Distancia GIQR ponderada	1.19	1.03	1.05	1.30
Total de características	13			
Características modificadas	2	3	4	5
SHAP top 3	2	3	3	3
SHAP top 5	2	3	4	4

Tabla 5.11: Resumen de la evaluación de contrafactuales en Heart Disease.

Sobre las soluciones obtenidas, en primer lugar, se puede destacar que se ha logrado un cambio de categoría modificando muy pocas características sobre la instancia original. Esto se refleja especialmente en el primer contrafactual, donde se modifican únicamente dos valores. Además, cabe resaltar que en ningún caso coinciden las características modificadas entre contrafactuales lo cual es un buen indicativo sobre la dispersión.

En cuanto a las distancias obtenidas, todas presentan valores cercanos pero no exactos lo cual es un aspecto deseable para asegurar que no hay contrafactuales mucho mejores que otros y validar las distintas soluciones. Considerando la distancia GIQR, se puede destacar el cuarto caso ya que presenta la mayor distancia con una diferencia considerable. Esto se debe al cambio de la variable binaria *'cp'* que, además, es la que manifiesta una menor importancia (la sexta) dentro de las modificadas por lo que tiene sentido que se dé su cambio ya habiéndose formado varios contraejemplos. Cabe resaltar que, aunque el valor de la métrica haya aumentado considerablemente, no se ha tomado como un error ya que por lo general cambiar datos binarios va a suponer una mayor diferencia. Por otro lado, si se disminuye el valor de las características más importantes en el cálculo, los resultados son más parejos. Analizándolos por separado, se puede distinguir que el primero se basa en un cambio grande en la característica más importante y el otros consiguen reducir esta diferencia variando otras.

Por lo general, los atributos alterados se encuentran entre las características más importantes. *'ca'*, según los resultados de aplicar la técnica SHAP, presenta una mayor contribución que el resto siendo más del doble que la del segundo, lo cual encaja con

la tendencia de cambio en los casos formados.

Como se ha introducido, a partir de esta explicación únicamente se van a comentar los aspectos más relevantes de cada apartado.

Adult Income

Sobre esta segunda problemática, se ha escogido una instancia aleatoria de ingresos inferiores a $50000 al año para determinar alternativas con las que a partir de cambios mínimos, conseguiría un aumento de salario. Sobre esta instancia inicial, se exponen en la Tabla 5.12 un subconjunto reducido de características escogidas según son de interés al compararlas con los contrafactuales obtenidos.

education-num	hours-per-week	education	occupation	workclass
9	40	11	5	2

Tabla 5.12: Datos de la instancia inicial en Adult Income.

De los parámetros escogidos para su ejecución, se destaca: *nbrs_mean* considerando *20* vecinos como el método de estimación de densidad, un umbral de *0.4* para establecer el mínimo, el conjunto *['workclass','education','education-num','hours-per-week','marital-status','occupation', 'relationship','race','sex','native-country']* de características como datos categóricos y el conjunto *['marital-status', 'relationship', 'race', 'sex', 'native-country']* como atributos no modificables de la instancia inicial.

education-num	hours-per-week	education	occupation	workclass
14	40	12	3	2
13	42	9	11	2
13	43	9	3	2
13	40	9	3	0

Tabla 5.13: Datos de los casos contrafactuales en Adult Income.

Distancia GIQR	3.25	3.16	3.21	4
Distancia GIQR ponderada	2.24	2.15	2.14	2.35
Total de características	14			
Características modificadas	3	4	4	4
SHAP top 3	1	1	1	1
SHAP top 5	2	2	2	2

Tabla 5.14: Resumen de la evaluación de contrafactuales en Adult Income.

Finalmente, en la Tabla 5.13 se presentan las características que han sufrido modificaciones con respecto a la instancia inicial, mientras que en la Tabla 5.14 se proporciona un resumen de su evaluación. En este caso, de las 5 características estimadas como más importantes, solo se han modificado 2. Esto se debe a que 2 de ellas han sido indicadas como no modificables y, la restante, que es la que tiene mayor importancia, ha sido penalizada debido a que se trata de una variable continua donde aproximadamente el 91 % de las muestras del conjunto tienen el mismo valor, lo que implica que cualquier cambio dentro de esta tendría un alto costo en términos de distancia con respecto a la distribución de los datos. Casos como este son considerados como “cambios atípicos” por el algoritmo.

Por otra parte, se puede destacar que *'occupation'* en la instancia inicial se refiere a limpiador, lo cual, según la predicción del modelo se ha establecido como muy negativo para alcanzar el objetivo. Por este motivo, esta característica binaria ha sido modificada en la mayoría de las ocasiones al trabajo que el explicador a determinado como más favorable para este caso (y en el contrafactual restante, a la tercera opción).

Por último, *'education'* es una característica interesante de comentar ya que está muy lejos de ser considerada como importante y que, si no hubiera sido modificada, el resultado de las predicciones sería similar. Su modificación se debe a presentar una relación directa con *'education-num'*. Mientras que la primera hace referencia a los últimos estudios, la segunda representa el nivel académico.

German Credit

En este ejemplo se va a trabajar con los datos de un solicitante de préstamos que ha sido considerado como un buen riesgo crediticio y sobre el cual se quieren distinguir que cambios se debería evitar para no obtener una resolución favorable. Como en la prueba anterior, se muestran en la Tabla 5.15 únicamente aquellos datos que se han considerado informativos sobre la instancia original procesada.

job	savings	property	existingcheckingstatus	employmentlength	otherinstallmentplans
2	2	1	3	4	2

Tabla 5.15: Datos de la instancia inicial en German Credit.

Al ejecutar esta prueba, se ha indicado el metodo *nbrs* con *5* vecinos y un umbral mínimo de *0.5* para valorar la densidad de los casos, el conjunto *['existingcheckingstatus','credithistory', 'purpose','savings', 'employmentlength', 'marriagesex', 'otherdebtors', 'property', 'otherinstallmentplans', 'housing', 'job', 'telephone', 'foreignworker']* como características categóricas y el conjunto *['age','credithistory', 'purpose', 'marriagesex', 'housing', 'foreignworker','peopleliableno']* como aspectos que no pueden ser modificados sobre el punto de partida. A partir de esta configuración, se han obtenido soluciones con las modificaciones expuestas en la Tabla 5.16 donde se puede resaltar la variable *'existingcheckingstatus'* por su contribución significativamente mayor respecto a las demás. En este ejemplo, la mayoría de las características son datos categóricos nominales y el explicador ha determinado que 11 de este tipo son las de mayor contribución (exceptuando 'age' que ha sido indicada como no modificable) por los que todos los cambios realizados son de esta índole.

job	savings	property	existingcheckingstatus	employmentlength	otherinstallmentplans
0	0	1	0	0	2
2	0	1	0	4	0
2	0	0	0	2	0
2	0	1	0	2	1

Tabla 5.16: Datos de los casos contrafactuales en German Credit.

Distancia GIQR	4.00	3.00	4.00	4.00
Distancia GIQR ponderada	1.80	1.55	2.06	2.29
Total de características	20			
Características modificadas	4	3	4	4
SHAP top 3	2	2	2	2
SHAP top 5	3	2	3	3

Tabla 5.17: Resumen de la evaluación de contrafactuales en German Credit.

Diabetes Health Indicators

En este ejemplo se pretende valorar el rendimiento y los resultados al aplicar el algoritmo sobre una instancia con un gran conjunto de datos de fondo. En lo que respecta al contexto del problema, se va a buscar que comportamientos o aspectos relacionados con la salud debería evitar un individuo para no llegar a ser diagnosticado como diabético. Las características destacables de la instancia original se muestran en la Tabla 5.18.

BMI	GenHlth	Income	HighChol
23	2	8	0

Tabla 5.18: Datos de la instancia inicial en Diabetes Health Indicators.

Al aplicar la propuesta sobre este caso, se ha especificado el conjunto *['HighBP' ,'HighChol' ,'CholCheck' ,'Smoker' ,'Stroke' ,'HeartDiseaseorAttack' ,'PhysActivity' ,'Fruits' ,'Veggies' ,'HvyAlcoholConsump' ,'AnyHealthcare' ,'NoDocbcCost', 'DiffWalk', 'Sex']* como datos categóricos, el conjunto *['HeartDiseaseorAttack' ,'NoDocbcCost' , 'DiffWalk', 'Sex' ,'Age']* como características no modificables y *lof* sobre *20* vecinos como el método con el que estimar la densidad.

Los atributos relevantes de los contrafactuales obtenidos y su evaluación se presentan en la Tabla 5.19 y 5.20. Sobre estas soluciones, se puede observar que son muy pocas las características que se habrían de modificar para que el modelo determine que el paciente es diabético. En el primer contrafactual, únicamente se varía el índice de masa corporal (*BMI*) alcanzando un valor que posiblemente coincida con una situación de sobrepeso. En el resto de las soluciones, aunque este atributo siempre aumenta con respecto a su valor inicial, se destacan otras variables que hacen que no tenga que aumentar hasta tal extremo. *BMI* es claramente la característica más contribuyente según el explicador pero, *GenHlth* también presenta una influencia significativamente mayor al resto lo cual se puede apreciar en los resultados obtenidos analizando su relación en el segundo contrafactual. Por último, cabe destacar que no se ha modificado la tercera característica más importante, *MentHealt*, debido a que aproximadamente el 70 % es un valor 0 y el resto está mayoritariamente concentrado en los primeros posibles (del 1 al 10) haciendo que un caso como el presentado de un valor 25 sea tratado como atípico. En definitiva, el algoritmo a penalizado levemente el costo de cambio sobre esta característica debido a la muy desigual distribución en sus datos y, además, pequeñas modificaciones en el valor de la muestra no tienen un impacto real en la predicción.

BMI	GenHlth	Income	HighChol
28	2	8	0
25	3	8	0
26	2	8	1
25	2	6	1

Tabla 5.19: Datos de los casos contrafactuales en Diabetes Health Indicators.

Distancia GIQR	0.59	1.16	1.30	1.67
Distancia GIQR ponderada	0.49	0.87	0.99	1.01
Total de características	21			
Características modificadas	1	2	2	3
SHAP top 3	1	2	1	1
SHAP top 5	1	2	2	2

Tabla 5.20: Resumen de la evaluación de contrafactuales en Diabetes Health Indicators.

Student Performance

Este último ejemplo se trata de un problema de múltiples categorías, donde se ha seleccionado aleatoriamente una instancia que representa a un estudiante matriculado en un programa universitario con riesgo de fracaso escolar. En definitiva, en este contexto se podría emplear para dar consejo al estudiante sobre cómo mejorar su rendimiento académico presentandole distintas alternativas. Los datos más relevantes de la instancia de estudio se muestran en la Tabla 5.21.

Schol. holder	1st enroll	1st eval	1st approv	2nd enroll	2nd approv	2nd grade
0	7	18	0	7	0	0

Tabla 5.21: Datos de la instancia inicial en Student Performance.

En este problema, además de los parámetros habituales que han sido modificados en los ejemplos anteriores, también se ha indicado este conjunto como una tarea multiclase. Con respecto al resto, se ha especificado el método *nbrs_mean* sobre *25* vecinos como estimador de densidad, 0.1 como umbral mínimo, *['Marital status', 'Application mode', 'Course', 'Daytime/evening attendance\t', 'Previous qualification', 'Nacionality', 'Mothers qualification', 'Father\'s qualification', 'Mother\'s occupation', 'Father\'s occupation', 'Displaced', 'Educational special needs', 'Debtor', 'Tuition fees up to date', 'Gender', 'Scholarship holder', 'International']* como el conjunto de características categóricas y *['Marital status', 'Application mode', 'Application order', 'Course', 'Previous qualification', 'Previous qualification (grade)', 'Nacionality', 'Mother\'s qualification', 'Father\'s qualification', 'Mother\'s occupation', 'Father\'s occupation', 'Gender', 'Age at enrollment']* como los atributos que no pueden ser modificados sobre el alumno. Los cambios sobre las soluciones obtenidas y la evaluación de los resultados se han expuesto en la Tabla 5.22 y en la Tabla 5.23.

Schol. holder	1st enroll	1st eval	1st approv	2nd enroll	2nd approv	2nd grade
0	7	9	4	7	5	11.68
0	5	8	4	7	4	10.5
0	6	9	4	6	4	10.75
1	5	7	4	5	3	11.72

Tabla 5.22: Datos de los casos contrafactuales en Student Performance.

Distancia GIQR	6.98	7.64	7.28	9.48
Distancia GIQR ponderada	3.52	3.49	3.35	3.97
Total de características	36			
Características modificadas	4	5	6	7
SHAP top 3	3	3	3	3
SHAP top 5	4	4	4	5

Tabla 5.23: Resumen de la evaluación de contrafactuales en Student Performance.

Los resultados muestran que hay más características que necesitan cambios en comparación con ejemplos anteriores. Esto podría ser debido a la complejidad del problema o a las restricciones establecidas. Además, en esta ejecución, hubo muchas características que se indicaron como no modificables, lo que dificulta cumplir con los mínimos de densidad especialmente en conjuntos relativamente pequeños con muchas características y varias categorías.

A pesar de estas dificultades, se considera que los resultados son satisfactorios en comparación con casos del conjunto de datos original.

5.4. Generador Incremental de Contrafactuales

En esta sección, se va a seguir una estructura similar a la del apartado anterior. Primeramente, se van a abordar aspectos relacionados con la implementación de la propuesta y luego, en un subapartado, se van a exponer los resultados obtenidos sobre las diferentes pruebas en los conjuntos de datos.

Con respecto a su ejecución, los parámetros de entrada que el usuario debe indicar de forma imprescindible son similares a los ya expuestos. Estos incluyen un conjunto de datos en formato Dataframe sobre el que se va a construir el grafo y se formaran los caminos, la instancia sobre la cual se desea realizar el estudio, indicaciones sobre si se trata de un problema multiclase, la salida esperada, el nombre de la categoría objetivo y el modelo sobre el que se resuelve el problema.

Por otro lado, se recomienda prestar especial atención a los parámetros resueltos por defecto, ya que varios de estos fundamentan la construcción del grafo y, por lo tanto, tienen un gran impacto sobre los resultados. Estos parámetros son:

- Un umbral sobre el límite de distancia máxima que pueden presentar dos nodos conectados en el grafo. Puede ajustarse en un rango de 0 a 10 y se recomienda realizar siempre una primera prueba en un valor 1, equivalente al percentil del 10 % sobre el conjunto total de distancias. Se trata de un parámetro difícil de ajustar ya que depende de la distribución del conjunto. Por defecto *1*.

- El método a emplear para estimar la densidad de un punto con respecto a la distribución de los datos. En este caso, solo se aceptan aquellos basados en vecinos como son *nbrs* y *nbrs_mean*. Por defecto *nbrs_mean*.
- El número de vecinos que se van a emplear sobre la técnica anterior. Por defecto *20*.
- Un umbral de densidad mínima, con un valor de 0 a 1. Por defecto *0*.
- Una indicación sobre si se quiere realizar una comprobación adicional sobre la densidad del contrafactual final con respecto a la distribución de datos con su misma clase. Por defecto *False*.
- Una indicación sobre si se desea intentar ajustar el camino a la situación de la instancia de estudio. Este proceso no siempre es posible. Por defecto *False*.
- Una lista de características no modificables que no van a ser consideradas al formar el camino de muestras. Además, tampoco se van a tener en cuenta al ajustar el conjunto sobre la instancia. Por defecto *vacio*.
- Una lista de las características categóricas tanto ordinales como nominales (binarias ficticias). Por defecto *vacio*.

Como resultado de una ejecución, el algoritmo ofrece un conjunto de casos en formato DataFrame que incluye la instancia de estudio como el primer caso, el contrafactual alcanzado como el último, y los casos intermedios que forman el camino recorrido. Además, también se presentan los índices de los nodos del grafo que componen el conjunto resultante y, si se ha indicado y es posible, los casos adaptados sobre la instancia original.

Finalmente, se van a exponer resultados sobre todas las problemáticas planteadas pero, en esta propuesta, al tener como objetivo crear un camino de casos según las restricciones indicadas, no son tan relevantes métricas como las empleadas en un algoritmo generador de contrafactuales normal ya que no están enfocadas a dar validez a la continuidad de los casos. Por ese motivo, se va a hacer una explicación en detalle sobre el conjunto de datos escogido para pruebas ya que permite hacer representaciones claras y, luego, se van a hacer breves demostraciones sobre el resto en las que se va a comentar la relación entre los contrafactuales teniendo en cuenta lo ya argumentado en el primer ejemplo.

Heart Disease

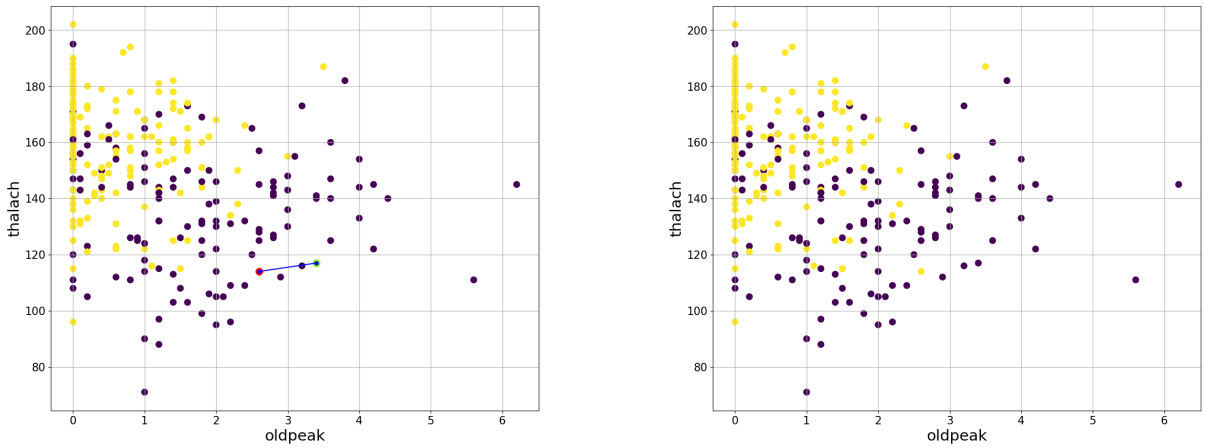
Con el fin de validar la forma en la que se construye el camino sobre los casos del conjunto de datos, en un primer lugar se van a representar gráficamente varios ejemplos de recorrido sobre dos características continúas indicadas como las únicas modificables por lo que solo estas se van a tener en cuenta en el cálculo de la distancia. En resumen, se van a ilustrar los cambios en los nodos recorridos al ir modificando la distancia máxima entre muestras, la densidad mínima del conjunto de casos y la específica del contrafactual. Seguidamente, se va a exponer una prueba sobre todas las características y se va a analizar igual que en las problemáticas siguientes.

Para la prueba de representación, como paso previo, se han seleccionado dos características del conjunto de datos que permitan una cómoda visualización con relación a sus valores en un espacio de dos dimensiones. Una vez identificadas, se ha explorado sobre una instancia que presente una importancia relativamente similar para ambas

variables con la finalidad de favorecer al entendimiento de la solución. Esto se debe a que la distancia se calcula considerando la importancia permitiendo más modificaciones en aquellos atributos con mayor contribución y, en el caso de destacar uno significativamente, el recorrido se formaría considerando en gran parte la similitud de los valores del eje contrario.

Siguiendo el proceso de selección explicado, se va a trabajar sobre las variables *'oldpeak'* y *'thalach'* siendo la primera la más importante para conseguir el cambio en la instancia de estudio. *'thalach'* es la tercera de mayor contribución, pero presentan poca diferencia entre ambas. En valores numéricos los pesos son 0.18 y 0.16 respectivamente. En definitiva, 'thalach' es más restrictiva al cambio por lo que el algoritmo priorizara reducir la distancia en su eje, pero, al tratarse un ejemplo con pesos muy similares, este aspecto no es tan determinante en la representación.

A continuación, se van a presentar varios ejemplos donde se van a ir variando las restricciones de densidad y distancia con el fin de realizar varias demostraciones. Cabe destacar que el color de los casos depende de la predicción del modelo y, sobre el camino, la instancia inicial se representa en verde, los nodos intermedios en azul y el contrafactual final en rojo. Por otra parte, todas estas pruebas se han realizado con el método *nbrs_mean* sobre *5* vecinos y el resto de parámetros de interés van a ser expuestos en el pie de cada subfigura como: UDist para el umbral de distancia, Udens para el umbral de densidad y Cd para indicar la restricción de densidad en el contrafactual en especifico.

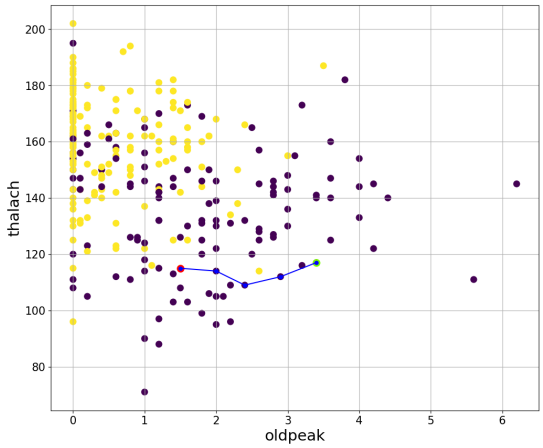


(a) UDist: 10, UDens: 0, Cd: False.

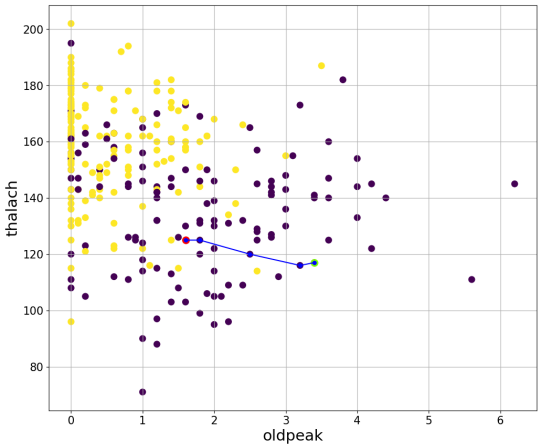
(b) UDist: 0, UDens: 1, Cd: True.

Figura 5.7: Representación de un camino para formar un contrafactual bajo mínimas y máximas restricciones.

En la Figura 5.7 se han representado las situaciones de mínimas y máximas restricciones donde se han obtenido los resultados esperados. En el primer ejemplo se ha ajustado directamente al caso contrafactual más cercano y, en el segundo al indicar una distancia máxima nula, no se ha encontrado ningún recorrido para llegar a un contrafactual.



(a) UDist: 1, UDens: 0.1, Cd: True.

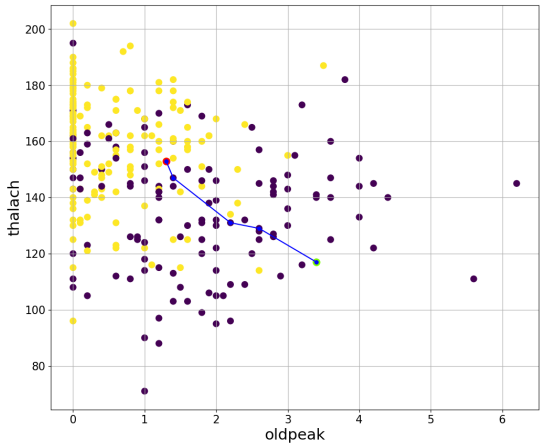


(b) UDist: 1.5, UDens: 0.1, Cd: True.

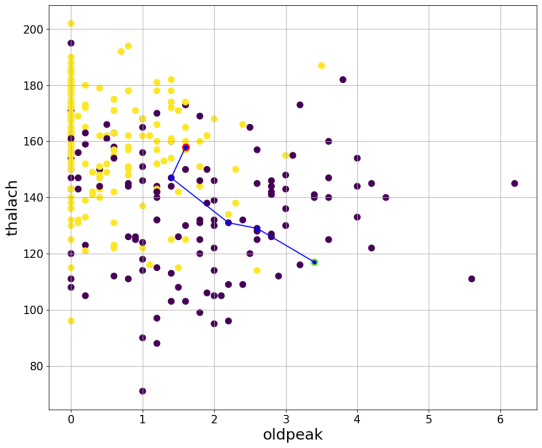
Figura 5.8: Representación de un camino para formar un contrafactual con diferente umbral de distancia mínima.

En la Figura 5.8 se muestra como afecta modificar la distancia máxima permitida entre casos sin variar los límites de densidad. Al comparar ambos ejemplos, se puede observar a simple vista que en el primero la distancia entre puntos es ligeramente más corta. Además, cabe destacar que modificar este parámetro a dado lugar a un cambio total en los nodos recorridos, lo que ha supuesto alcanzar un caso contrafactual diferente en cada ejecución. A partir del segundo ejemplo, se ha comprobado que seguir aumentando la distancia máxima entre puntos lleva siempre al mismo contrafactual final pero, modificando el recorrido y haciendo que haya cada vez menos nodos intermedios.

Por otro lado, en relación con la demostración anterior, se han añadido restricciones de densidad mínima a todos los puntos y al caso contrafactual final. Esto ha llevado a ignorar la solución anterior (la más próxima) por no presentar una densidad de muestras suficiente a su alrededor.



(a) UDist: 3, UDens: 0.8, Cd: False.



(b) UDist: 3, UDens: 0.8, Cd: True.

Figura 5.9: Representación de un camino para formar un contrafactual variando las indicaciones específicas de densidad del contrafactual.

Por último, en la Figura 5.9 se han expuesto dos ejemplos con limitaciones similares exceptuando que en el segundo caso se ha considerado la densidad del contrafactual de forma específica. Como se puede observar, aunque el recorrido de muestras realizado en ambos es similar, el nodo final difiere según las indicaciones dadas. Esto se debe a que en el primer ejemplo se está valorando con respecto a todo el conjunto de datos y, en el segundo caso, únicamente con los de su misma categoría.

Aunque este aspecto que estima la plausibilidad si tiene en cuenta todas las características y no solo las no modificables, se ha comprobado que en este ejemplo del conjunto Heart Disease se pueden aproximar las zonas de mayor cantidad de puntos como las de mayor densidad. Pues bien, como se puede contrastar en relación con las demostraciones anteriores donde el umbral de densidad mínima base es mucho menor, el recorrido se forma a través de nodos en zonas más pobladas hasta llegar a un caso final que cumple la condición específica.

Todas estas pruebas representadas validan la actuación correcta del algoritmo al formar el grafo en términos de densidad y distancia entre características.

Para terminar, se va a exponer una prueba del algoritmo sobre una instancia cualquiera considerando todas las características como modificables. Tanto en esta problemática como en las siguientes, se van a establecer las mismas restricciones que en el apartado anterior sobre cuáles van a ser las variables categóricas y no modificables por lo que no se van a volver a indicar. Con respecto a los otros parámetros, se ha especificado un umbral de distancia de *0.3*, un umbral de densidad mínima de *0.25*, el método *nbrs_mean* en *5* vecinos, que no sea validada la densidad específica del caso contrafactual y, para terminar, que se genere un recorrido plausible de sustitución sobre la instancia original si es posible. El conjunto de casos obtenido se expone en la Tabla 5.35.

age	trestbps	chol	restecg	thalach	oldpeak	ca	sex	cp	fbs	exang	slope	thal
68	144	193	1	141	3.4	2	1	0	1	0	1	3
58	146	218	1	105	2.0	1	1	0	0	0	1	3
63	130	254	0	147	1.4	1	1	0	0	0	1	3
54	120	258	0	147	0.4	0	1	2	0	0	1	3

Tabla 5.24: Recorrido de casos desde la instancia inicial hasta un caso contrafactual en Heart Disease.

Sobre el camino obtenido, se pueden destacar tendencias claras que podrían servir como guía del cambio para conseguir que la instancia inicial alcance la categoría objetivo. En este ejemplo en particular, el explicador de SHAP ha indicado que las variables *'oldpeak'* y *'ca'* son el doble de importantes que le resto, y ambas características presentan un descenso gradual y simultaneo de sus valores en el recorrido establecido. Además, otras variables lejos de ser consideradas como las más contribuyentes, presentan transformaciones que deberían tenerse en cuenta ya que se observa una clara relación. Por ejemplo, el caso más evidente es 'chol' la cual se refiere al nivel de colesterol de la persona y se puede observar cómo aumenta en cada uno de los casos. Asimismo, también se podrían valorar otras características como *'trestbps'*, *'restecg'*, *'fbs'* o *'cp'*.

Por último, el resto de las características simplemente se mantienen muy similares entre casos o presentan variaciones repentinas que se corrigen en la siguiente mues-

tra sin mostrar una tendencia clara. Estas últimas perturbaciones pueden deberse a una irregularidad no muy destacable en una muestra independiente o a un paso intermedio puntual a estudiar que se podría analizar sometiendo más instancias a estudio o ampliando el recorrido de muestras.

age	trestbps	chol	restecg	thalach	oldpeak	ca	sex	cp	fbs	exang	slope	thal
68	144	193	1	141	3.4	2	1	0	1	0	1	3
68	144	218	1	105	2.0	1	1	0	0	0	1	3
68	144	254	1	147	1.4	1	1	0	0	0	1	3
68	144	258	1	147	0.4	0	1	2	0	0	1	3

Tabla 5.25: Muestras por sustitución del recorrido sobre la instancia inicial en Heart Disease.

Para terminar, se han sustituido las características más importantes de forma iterativa sobre la instancia original hasta obtener un conjunto de muestras nuevas que cumplen con las restricciones normales de densidad al formar el grafo, aspectos relacionados con la causalidad y mantienen varias de las tendencias. Este resultado que se muestra en la Tabla 5.25 se puede ver cómo un resumen de los cambios pero, es importante destacar que, dependiendo de las características no modificables y el conjunto de datos, no siempre esta garantizado poder formar un camino por sustitución que cumpla las mismas limitaciones de densidad que las muestras escogidas para formar el recorrido. Esto es debido a la gran variabilidad y a las excepciones que se pueden dar dentro de los muestras escogidas. Además, algunas de las restricciones no modificables que no son tenidas en cuenta al formar el grafo o en la sustitución, pueden ser consideradas como atípicas lo cual penaliza la estimación de densidad.

Adult Income

Esta problemática, como se ha comentado al introducirla, presenta una gran cantidad de casos lo cual dificulta el trabajo con grafos. Para favorecer a la eficiencia de las pruebas sobre el algoritmo, se ha explorado sobre formas inteligentes con las que reducir el conjunto y perjudicar lo menos posible a los resultados. Finalmente, se ha optado como estrategia descartar muestras según presentan diferencias en las restricciones no modificables bajo cierta tolerancia escogida.

Esta práctica se podría incluir como parte del algoritmo como una indicación que garantice producir el camino únicamente considerando casos en una situación muy similar sobre aquellas características que no pueden ser modificadas pero, se ha decidido aplicar desde fuera de la implementación porque se ha considerado como un proceso externo más relacionado con la preparación del conjunto. Además, también se ha querido evitar aumentar más el número de parámetros de entrada, lo cual puede suponer un mayor trabajo de ajuste para obtener un resultado satisfactorio.

La implementación más eficiente de este algoritmo sobre conjuntos de datos muy grandes se va a presentar más adelante como un aspecto de mejora en un trabajo futuro.

En lo que respecta a la ejecución, se ha establecido por parámetro un umbral de distancia mínima de *0.6*, el método *nbrs_mean* en *20* vecinos y un umbral de *0.4* para establecer las restricciones de densidad. Además, se ha indicado que no se estudie la densidad específica del contrafactual y que, si es posible, se forme un recorrido

por sustitución. Los resultados de las características modificables más relevantes se muestran en la Tabla 5.26.

age	fnlwgt	education-num	hours-per-week	education	occupation	workclass
49	160187	5	16	6	7	2
49	177426	9	20	11	7	2
48	195491	9	30	11	11	2
47	191957	9	40	11	11	2

Tabla 5.26: Recorrido de casos desde la instacia inicial hasta un caso contrafactual en Adult Income.

Sobre la solución obtenida, se podría destacar como un primer paso mejorar el nivel de educación (la cual es la característica más importante) para posteriormente, cambiar a otra ocupación. También se destaca que el individuo trabaja muy pocas horas a la semana por lo que debe aumentarlas por ejemplo de forma gradual como indican los resultados del algoritmo.

Para terminar, cabe resaltar que no se ha logrado formar un recorrido por sustitución aun habiendo descartado al principio muestras no muy similares. Después de varias comprobaciones, se ha visto que en el conjunto original presenta únicamente 31 casos de individuos provenientes de Jamaica y 10 de personas en el mismo estado civil. Además, si se consideran las dos al mismo tiempo, únicamente existe esta instancia. Aunque todos estos casos han sido recogidos al formar el conjunto final, el hecho de que sean dos características tan anómalas hace que sea muy difícil cumplir con las restricciones de densidad al formular las nuevas muestras.

German Credit

Al aplicar el algoritmo en esta problemática se han indicado como parámetros un umbral de distancia de *0.1*, un umbral de densidad de *0.4* que actúa sobre el método *nbrs_mean* considerando *10* vecinos, un indicativo para que si se tenga en cuenta la densidad específica del contrafactual y la opción de que se forme un camino por sustitución.

duration	installmentrate	presentresidencelength	age	existingcheckingstatus
6	1	2	56	3
15	2	3	36	0
9	3	4	22	0

Tabla 5.27: Recorrido de casos desde la instacia inicial hasta un caso contrafactual en German Credit.

duration	installmentrate	presentresidencelength	age	existingcheckingstatus
6	1	2	56	3
15	2	2	56	0
9	3	2	56	0

Tabla 5.28: Muestras por sustitución del recorrido sobre la instancia inicial en German Credit.

Del conjunto resultante expuesto en la Tabla 5.27, se destaca el cambio inicial de *'existingchekingstatus'* que se trata de la característica más contribuyente. Por otro

lado, otros aspectos como la tasa de cuotas o la duración en la residencia actual aumentan de forma gradual por lo que se deben considerar también estas variables. Por último, aunque se haya logrado un recorrido por sustitución como se muestra en la Tabla 5.28, se puede observar una clara tendencia de cambio en la edad. Esta última característica es de carácter no modificable por lo que podría ser conveniente ajustar el conjunto de datos para tener presente únicamente individuos dentro de un rango de edad similar al de la instancia de estudio.

Diabetes Health Indicators

Al igual que en el ejemplo de Adult Income, en esta demostración se ha realizado un paso previo sobre el conjunto de datos para ajustar las muestras a las características no modificables de la instancia de estudio y reducir el número de casos total. Al ejecutar esta prueba, se ha indicado un umbral de distancia máxima de *0.5*, un umbral de densidad mínima de *0.6* que actúa sobre el método *nbrs_mean* en *25* vecinos, que se tenga en cuenta la densidad específica del caso final y que, a partir del recorrido conseguido, se forme un conjunto adaptado por sustitución.

BMI	GenHlth	Education	Income	HighChol
19	1	5	5	0
24	2	4	4	1
27	2	4	4	1

Tabla 5.29: Recorrido de casos desde la instancia inicial hasta un caso contrafactual en Diabetes Health Indicators.

Los datos que se muestran en la Tabla 5.29 sirven para representar el conjunto de muestras obtenidas y el recorrido conseguido en la sustitución ya que, para este caso, ambos resultados han sido exactos. Este ejemplo es fácil de comprender debido a que se muestra un camino a evitar para prevenir la diabetes a partir de atributos conocidos: se observa un aumento en el índice de masa corporal, el deterioro del estado de salud general, niveles elevados de colesterol y el empeoramiento de algunos aspectos sociales.

Student Performance

Esta última prueba se realiza sobre un problema de múltiples categorías por lo que se ha indicado esta característica entre los parámetros de entrada. Asimismo, también se establecen los parámetros habituales, tales como un umbral de distancia de *4.2*, un umbral de densidad de *0.6* en el método *nbrs_mean* considerando *80* vecinos e indicaciones sobre evaluar la densidad de la muestra contrafactual final. Es importante destacar que en este caso no se formará un conjunto por sustitución.

1st enroll	1st eval	1st aprov	1st grade	2nd enroll	2nd eval	2nd aprov	2nd grade
7	12	5	12.1	7	0	0	0
7	10	4	10.5	7	7	2	10.5
6	8	5	12.2	6	6	6	11.5

Tabla 5.30: Recorrido de casos desde la instancia inicial hasta un caso contrafactual en Student Performance.

En lo que respecta a la solución obtenida, se han representado en la Tabla 5.30 una serie de características seleccionadas según su importancia y relación entre las

mismas. En este contexto, todas se refieren a los aspectos académicos de un alumno separados en datos sobre los semestres cursados. En definitiva, se exponen dos ejemplos progresivos que mejoran la situación de la instancia inicial con cambios muy significativos en la parte relacionada con el segundo semestre. Estas indicaciones engloban mejorar la nota media de las asignaturas, reducir las asignaturas cursadas y aprobar el máximo de estas.

5.5. Comparativa con Otros Algoritmos

En esta última sección, se van a comparar el rendimiento y los contrafactuales resultantes al aplicar la primera propuesta sobre el conjunto de datos Heart Disease con otros algoritmos ya establecidos que han sido presentados en el estado del arte. Los métodos que se van a utilizar para realizar las comparaciones son WACH, DiCE y DisCERN.

En primer lugar, se van a exponer brevemente los aspectos sobre los que van a ser comparados ya que en su mayoría han sido desarrollados en otras secciones: se va a comprobar la validez de las soluciones obtenidas verificando que el modelo consiga predecir la categoría objetivo sobre los contrafactuales desarrollados, se va a determinar la similitud con la instancia de estudio a partir de las dos métricas presentas en la evaluación de los resultados de la primera propuesta (la distancia GIQR sobre las muestras procesadas y la distancia GIQR ponderada en las muestras sin procesar), se va a cuantificar la practicidad de los resultados a partir de la cantidad de características modificadas y, dentro de estas, cuantas se encuentra entre las 3 más contribuyentes, se va a medir el tiempo aproximado de ejecución del algoritmo en situaciones similares, se va a determinar si los resultados son dispersos entre ellos y, siguiendo el método *nbrs_mean* considerando *5* vecinos, se va a medir la densidad de cada muestra contrafactual.

A partir de este punto, se va a presentar cada metodología, su implementación y la evaluación de los resultados obtenidos junto con los del algoritmo desarrollado en el presente trabajo. Cabe destacar que, finalmente, se realizarán dos pruebas comparativas. La primera consistirá en una única ejecución por cada algoritmo sobre la instancia expuesta en la Tabla 5.31 donde se generarán varios contrafactuales mostrando los resultados de la evaluación de cada uno (los contrafactuales obtenidos en cada ejecución se han expuesto mientras se presentaban los algoritmos). La segunda prueba se desarrollara a partir de la media de los resultados obtenidos en la evaluación de un único contrafactual al ejecutar los algoritmos sobre un conjunto de 8 instancias aleatorias.

age	trestbps	chol	restecg	thalach	oldpeak	ca	sex	cp	fbs	exang	slope	thal
62	138	294	1	106	1.9	3	0	0	1	0	1	2

Tabla 5.31: Datos de la instancia original.

WATCH es un acrónimo comúnmente utilizado para referirse al método contrafactual desarrollado por Wachter et al. en 2017 [1]. Esta metodología introdujo el enfoque general de formar explicaciones contrafactuales en el campo de la inteligencia artificial de manera automatizada y ha sido ampliamente reconocida y aplicada por la comunidad. Para su implementación, se ha empleado la función 'create_counterfactual' de

Mlxtend [52], una biblioteca de Python que ofrece multitud de herramientas útiles para tareas en la ciencia de datos.

Con el fin de conseguir una correcta ejecución en la prueba, se ha tenido que modificar la función debido a que esta no era capaz de manejar las situaciones donde la desviación absoluta media (MAD) era igual a 0. En su corrección, se ha sustituido por 1 en estos casos tal como se hace en DiCE. Por otro lado, también se ha ido variando el hiperparámetro λ lo cual es una práctica habitual al ejecutar este método para buscar los mejores resultados ya que este término regula que el contrafactual generado sea lo más similar posible a la instancia original y logre alcanzar la predicción deseada. Finalmente, se exponen en la Tabla 5.32 los resultados de distintas ejecuciones sobre valores de λ en una escala multiplicativa de 10 desde 0.1 hasta 100. Cabe destacar que los valores han sido ajustados manualmente ya que las modificaciones no se restringen en un rango y no se manejan de forma especifica las variables categóricas por lo que se dan casos imposibles.

age	trestbps	chol	restecg	thalach	oldpeak	ca	sex	cp	fbs	exang	slope	thal
62	138	233	1	120	0.3	1	0	0	1	0	0	2
62	138	211	1	105	1.9	1	0	0	1	0	1	2
62	138	227	0	133	0	1	0	0	1	0	1	2
62	138	233	1	122	0.17	1	0	0	1	0	1	2

Tabla 5.32: Contrafactuales resultantes de aplicar WATCH.

DiCE desarrollado por Mothilal et al. en 2020 [42] y ampliado a redes neuronales con un método basado en gradientes en 2021 [53], es posiblemente el algoritmo más popular en la actualidad en lo que respecta a la formulación de contrafactuales. Este ofrece varias técnicas de entre las cuales se va a utilizar el método 'Random' y 'Gradient' ya que son las únicas compatibles con los modelos desarrollados en el presente trabajo.

Como problema en su implementación, cabe destacar que los desarrolladores no han considerado el uso de variables numéricas (ni ordinales ni binarias ficticias) para representar características categóricas lo cual es una practica habitual. Por un lado, para que una característica sea tratada como categórica el método debe recibir como dato una cadena y, en caso contrario, aunque se indique que no se trata de una variable continua por parámetro, el método modifica su valor como tal. Este problema hace que las variables categóricas puedan presentar valores intermedios imposibles. Por otro lado, no hay una forma establecida para indicar que se consideren variables conjuntamente por lo que aquellas codificadas en one-hot se modifican de manera independiente dando lugar a casos incongruentes. Estos problemas se han visto ampliamente comentados en foros y los desarrolladores han expresado su intención de abordarlos en una versión futura.

Con el fin de minimizar el error, se han establecido todas las categóricas como continuas ya que están representadas en valores numéricos y, las variables codificadas en varias alternativas ficticias, se han indicado como no modificables para evitar soluciones imposibles. El resto de valores se han aproximado al valor valido más cercano. Cabe destacar que, en esta instancia escogida para realizar la prueba, ni el algoritmo propuesto en este trabajo ni el explicador SHAP han dado una importancia significativa a ninguna de las restringidas en el cambio por lo que se suponen resultados comparables.

Método Random: se basa en la búsqueda aleatoria de casos con la etiqueta deseada cerca del punto de consulta.

age	trestbps	chol	restecg	thalach	oldpeak	ca	sex	cp	fbs	exang	slope	thal
62	138	294	1	185	1.9	3	0	3	1	0	1	2
62	138	294	1	192	1.9	0	0	0	1	0	1	2
62	138	294	1	106	1.9	0	0	1	1	0	1	2
62	138	294	1	179	1.9	1	0	0	1	0	1	2

Tabla 5.33: Contrafactuales resultantes de aplicar DiCE con el método aleatorio.

Método Gradient: se basa en la optimización por descenso de gradiente y es especifico para modelos de aprendizaje profundo.

age	trestbps	chol	restecg	thalach	oldpeak	ca	sex	cp	fbs	exang	slope	thal
62	138	277	1	151	1.5	1	0	0	1	0	1	2
62	138	294	1	150	1.2	1	0	0	1	0	1	2
62	138	293	1	164	1.9	1	0	0	1	0	1	2
63	135	294	1	156	1.7	1	0	0	1	0	1	2

Tabla 5.34: Contrafactuales resultantes de aplicar DiCE con el método de gradiente.

DisCERN es un algoritmo de enfoque heurístico propuesto por Wiratunga et al. en 2021 [35] [54] que se basa en iterar sobre los vecinos más cercanos que pertenecen a la categoría objetivo, identificando y aplicando cambios mínimos al sustituir las características más importantes obtenidas según el explicador LIME o SHAP. Para una mejor comparación, en este trabajo se emplea SHAP.

age	trestbps	chol	restecg	thalach	oldpeak	ca	sex	cp	fbs	exang	slope	thal
69	140	239	1	151	1.8	3	0	3	0	0	2	2

Tabla 5.35: Contrafactual resultante de aplicar DisCERN.

Sobre la primera prueba, que consistía en la evaluación de una única ejecución en una instancia inicial conocida, todos los resultados obtenidos a través de los algoritmos expuestos se han evaluado y se presentan en la Tabla 5.36 junto con los correspondientes a la propuesta del presente trabajo que han sido discutidos en su sección (Tabla 5.36). Los mejores valores de cada algoritmo se han resaltado en negro.

En primer lugar, en cuanto a la validez y la dispersión, todos han demostrado conseguir un cambio de clase en la predicción del modelo y, de entre los algoritmos probados, DiCE y el propuesto son los únicos capaces de formar varias soluciones desde una misma ejecución. Observando las diferencias entre las características modificadas de estos, aunque no hay resultados con valores similares en ningún caso, se puede destacar que el propuesto y DiCE basado en gradiente logran una mejor diversidad.

Antes de resaltar conclusiones sobre la Tabla 5.36, es importante señalar que el método WATCH aunque obtiene en uno de sus contrafactuales el mayor valor de densidad, entre sus casos se presentan situaciones imposibles que dan lugar a estimaciones erróneas de este aspecto de evaluación con respecto al conjunto (como por ejemplo valores que descienden por debajo del 0 en variables binarias) por lo que se

considera que no se debe tener en cuenta como el mejor obtenido. Los valores finales únicamente han sido corregidos para la representación de la Tabla 5.32. Además, en el resto de los resultados es muy inferior en comparación. Por otro lado, el algoritmo propuesto es capaz de forzar mejores valores en este aspecto ya que es el único que permite ajustar este atributo por parámetro pero podría haber derivado en peores conclusiones de similitud al formar casos más restrictivos. En definitiva, para favorecer una mejor comparación, se ha buscado obtener resultados similares en densidad o ligeramente superiores al resto para demostrar que es capaz de ser mejor y valorar los demás puntos de forma justa.

Algoritmo	Propuesta	WATCH	DiCE R	DiCE G	DisCERN
Distancia GIQR	2.42 **2.30** 2.42 3.20	5.83 **5.34** 7.03 6.07	2.97 3.66 **2.5** 2.83	2.54 2.47 **2.46** 2.54	**5.35**
Distancia GIQR ponderada	1.19 **1.03** 1.05 1.30	1.41 **1.31** 1.62 1.51	1.87 1.98 **1.36** 1.60	1.13 **1.13** 1.32 1.20	**2.28**
Características modificadas (Top 3 Shap)	**2(2)** 3(3) 4(3) 5(3)	5(3) **3(2)** 5(3) 4(3)	2(1) **2(2)** 2(1) 2(2)	4(3) **3(3)** 3(2) 5(3)	**8(2)**
Densidad del counterfactual	0.57 0.60 **0.60** 0.58	0.55 0.53 **0.65** 0.50	0.47 0.50 0.50 **0.52**	**0.60** 0.59 0.56 0.57	**0.56**
Tiempo de ejecución (seg)	90	490 (123)	1	623	3

Tabla 5.36: Resumen de la evaluación de los algoritmos sobre un caso aleatorio.

Esta evaluación sobre un caso específico exhibe que la propuesta del presente trabajo ha obtenido los mejores resultados o similares en todas las áreas evaluadas. Como punto negativo, se puede distinguir el cuarto contrafactual por presentar menor similitud en lo que respecta a la distancia GIQR pero, como se ha argumentado en su apartado, no se ha considerado como un error ya que se ha formado a favor de mejorar la dispersión variando características binarias menos relevantes lo cual hace que aumente significativamente esta métrica y está fuertemente restringido en DiCE ya que se ha indicado así por parámetro.

Para analizar como actúa cada metodología y las soluciones obtenidas en cada uno de los distintos aspectos, se van a exponer también los resultados obtenidos de la segunda prueba en la Tabla 5.37. Como se ha comentado, esta se trata de la presentación de los valores medios obtenidos en la evaluación de un conjunto de pruebas efectuadas sobre 8 muestras aleatorias distintas. En cada una, se ha formulado un único contrafactual sobre el cual se ha hecho el mismo análisis que en la prueba anterior.

Con el objetivo de que todos los resultados sean comparables, se han especificado una serie de parámetros comunes y se han modificado levemente las implementaciones de WATCH y DisCERN.

En primer lugar, se han aplicado las mismas restricciones a todos los métodos, limitando en cada caso el mismo conjunto de características seleccionadas como in-

mutables al ejecutar DiCE, con el fin de asegurar resultados válidos y posibles. Para ello, todos los algoritmos aceptan una indicación como parámetro de entrada, exceptuando WATCH. Sobre este último, se ha modificado levemente su función de pérdida para que el conjunto de características indicado como inmutable nunca varíe y únicamente se consideren los cambios en las características modificables al calcular la métrica de distancia y al hacer las comprobaciones sobre la predicción.

Por otra parte, el algoritmo de DisCERN base solo hace pruebas en el vecino más cercano no similar y en el caso de no poder lograr un contrafactual a partir de este, termina su ejecución declarando que no se ha encontrado un resultado. En una prueba como la anterior sin limitaciones, este aspecto no representaba ningún problema ya que siempre se iba a poder lograr un contrafactual (aunque sea sustituyendo todos los valores) pero, al excluir ciertas variables en el proceso de sustitución, puede que los cambios de los valores no lleguen a ser suficientes en algunos casos. Con el fin de lograr un resultado para todos los ejemplos sometidos a la prueba, se ha modificado el código para que cuando no sea posible formar un contrafactual sobre el primer vecino considerado, se continúe intentándolo con los siguientes sin detener la ejecución hasta lograr una solución.

Algoritmo	Propuesta	WATCH	DiCE R	DiCE G	DisCERN
Distancia GIQR	1.96	3.89	2.27	4.40	3.41
Distancia GIQR ponderada	1.07	1.77	1.33	1.70	1.36
Características modificadas	2.38	5.13	2.25	7.25	5.88
Densidad del counterfactual	0.71	0.57	0.58	0.57	0.76
Tiempo de ejecución (seg)	27	139	2	101	6

Tabla 5.37: Evaluación media de los algoritmos en un conjunto de 8 instancias.

De entre los resultados obtenidos, se puede destacar que los logrados por la propuesta del trabajo son en su mayoría superiores al resto.

La distancia GIQR representa la similitud del contrafactual generado en cada método con la instancia en estudio, combinando la distancia de Gower típicamente utilizada en conjuntos de datos mixtos con la normalización a partir del rango intercuartílico. Los resultados indican que los contrafactuales generados por la propuesta son los más próximos. Además, esta misma distancia ponderada muestra que también son los más similares cuando se da menos peso al cambio en las características más relevantes. Asimismo, se puede destacar que se ha logrado el cambio de clase modificando muy pocas características en comparación con los otros algoritmos; solo DiCE cuando reemplaza variables de forma aleatoria lo supera ligeramente, pero este método no garantiza la coherencia de los casos generados.

Sobre la densidad, también se presenta como el segundo mejor. Esto seguramente se debe a que al aplicar DisCERN para formar un contrafactual se suele conseguir la solución a partir de la sustitución directa de un alto número de valores entre las características. Aun así, como ya se ha explicado anteriormente, se podría forzar obtener mejores resultados en la propuesta modificando los parámetros de entrada.

En lo que respecta a los algoritmos que consiguen los contrafactuales por una estrategia únicamente basada en la sustitución, el método Random de DiCE es el único que al repetir la ejecución en varias ocasiones no da resultados semejantes y además, como el nombre de su forma de actuar indica, se basan en ajustes totalmente

aleatorios sobre un número pequeño de características que dan un cambio de predicción. Este comportamiento después de varias pruebas se ha considerado indeseable ya que, en muchas ocasiones, se obtienen resultados ilógicos muy difíciles de diferenciar del resto, no se consiguen los mejores casos posibles en base a ningún aspecto y además, suelen ser muy similares con el caso original lo que los hace parecer siempre adecuados. En definitiva, al utilizar este algoritmo es imprescindible disponer del conocimiento suficiente o de algún proceso adicional que distinga los buenos contrafactuales. Por otra parte, DisCERN si plantea una estrategia para seleccionar los atributos basada en la importancia pero, por lo general, se realizan muchos cambios al formar el contrafactual. Esto se debe a que se fuerza el cambio sobre el vecino más cercano independientemente de cuantos sean los necesarias para lograr la predicción objetivo y utilizando como métrica la distancia euclidiana básica la cual no considera correctamente valores mixtos (en la mayoría de los ejemplos, la muestra escogida sobre la que basar el cambio no va a ser la idónea). Además, en ninguna de las dos metodologías esta optimizado el valor de la característica por lo que pueden presentar una modificación mayor a la necesaria.

En lo que respecta a los algoritmos que siguen una estrategia de optimización, aparentemente muestran resultados bastante inferiores a los de la propuesta del trabajo. Además, cabe destacar que WATCH aunque parezca obtener mejores soluciones, la mayor parte de las muestras que ha generado contienen valores fuera de la distribución de los datos y requiere de mucho más esfuerzo ya que se ha tenido que explorar sobre un valor de lambda adecuado para conseguir la predicción en cada caso de estudio. Por otro lado, DiCE basado en Gradiente, aunque no siempre consigue buenos resultados lo cual ha penalizado mucho sus valores en la Tabla 5.37, estos siempre son válidos y, en algunos casos, los más cercanos a los obtenidos por la propuesta.

Sobre el tiempo de ejecución, se puede observar y destacar como los algoritmos puramente heurísticos son casi instantáneos pero, de aquellos que presentan un proceso de optimización más complejo, el algoritmo propuesto es el que ha logrado los resultados más rápidamente. Aunque este tiempo puede variar dependiendo de los parámetros, en ningún caso se ha acercado a los alcanzados sobre los otros dos algoritmos.

Capítulo 6

Conclusiones

En el presente trabajo se han explorado diversos enfoques y metodologías que aprovechan la información sobre la importancia de las características en el desarrollo de contrafactuales para problemas con datos tabulares. A partir de esta investigación preliminar, se han desarrollado dos propuestas de algoritmos que buscan satisfacer las propiedades deseables de este tipo de explicaciones, plantear un nuevo enfoque para su desarrollo, construir herramientas competitivas y abordar aspectos menos destacados que a menudo son ignorados en este tipo de trabajos.

La primera propuesta consiste en una herramienta completa de formulación de contrafactuales con un objetivo muy similar al de la mayor parte de los trabajos de la actualidad: conseguir los mejores ejemplos que expliquen el cambio de categoría sobre una instancia de estudio, considerando aspectos como la similaridad, plausibilidad, dispersión o practicidad de las muestras generadas. Para ello, se ha diseñado un método de enfoque híbrido que combina las estrategias clásicas de búsqueda heurística y optimización, ambas guiando sus modificaciones según la importancia de las características. Cabe destacar que, aunque ambos enfoques han sido ampliamente investigados por separado, no se ha observado de ningún trabajo que intente combinar sus ventajas.

En cuanto a los resultados obtenidos, se ha demostrado el buen funcionamiento tanto de forma lógica como a través de distintas métricas adaptadas al contexto del trabajo en varios conjuntos de datos incluyendo algunos muy destacados en el campo de la experimentación de contrafactuales y otros con ciertas características de interés. Además, en una sección final se han comparado las soluciones obtenidas con otras herramientas populares bajo las mismas condiciones, donde se ha demostrado que es un algoritmo claramente competitivo, que logra excelentes resultados y, en algunos casos, superiores con un margen significativo.

Por otro lado, la segunda propuesta consiste en una metodología que busca asegurar metas alcanzables y explicar cómo lograrlas, un objetivo menos común en la actualidad al trabajar con contrafactuales. Esta se basa en la construcción de un grafo considerando aspectos como la similaridad entre muestras, la plausibilidad de los casos individuales y el cambio sobre las características determinadas como más importantes a partir del cual formar un camino incremental entre muestras reales hacia un contrafactual de la clase objetivo. Los resultados de esta herramienta sobre los distintos conjuntos de datos han demostrado ser capaces de extraer una colección de

casos que muestran cambios sucesivos y lógicos hasta conseguir el cambio.

En definitiva, se han desarrollado dos metodologías competitivas que abordan aspectos diferentes de las explicaciones contrafactuales y donde se destaca el uso de la importancia de las características por tener un papel crucial para identificar que variables son las más relevantes a considerar cuando se está formulando un cambio. Estas dos propuestas son herramientas prometedoras que pueden ayudar a mejorar la comprensión y la confianza en los modelos de caja negra, así como a cumplir con la legislación actual incluida en el Reglamento General de Protección de Datos y La Ley de Inteligencia Artificial.

6.1. Trabajos Futuros

Existen multitud de líneas interesantes para trabajos futuros tomando el presente como punto de partida. Entre estas se pueden destacar algunas como las comentadas a continuación.

En relación con ciertos aspectos compartidos entre ambas propuestas, como la similaridad entre casos mixtos y la plausibilidad, aunque han sido ampliamente estudiados por los científicos, no existe un enfoque único que destaque como el mejor indiscutible por encima del resto. Por este motivo, se propone continuar con la investigación en estrategias similares, considerando otras alternativas y contribuyendo así al avance y a ofrecer cada vez soluciones más robustas y efectivas.

Además, también se propone experimentar con otras metodologías al obtener la importancia de las características. Durante el desarrollo del presente trabajo se ha visto el explicador SHAP como la mejor alternativa según el estado del arte y diversas pruebas realizadas pero, aun así, es importante explorar sobre otras opciones para verificar si existen métodos que mejoren la calidad de las explicaciones contrafactuales.

Con respecto a cada propuesta en particular, la primera se presenta como una herramienta más completa dentro de un campo donde ha habido una mayor investigación. En el presente trabajo se han intentado cubrir todos los objetivos deseables sobre el desarrollo de contrafactuales pero, en el futuro pueden surgir nuevos requerimientos por lo que se propone continuar explorando sobre las necesidades en la formulación de contrafactuales. Por ejemplo, se podría dar más relevancia a la personalización dando a los usuarios nuevas opciones como podría ser la capacidad de seleccionar que característica quiere que sea la principal al modificar la instancia de estudio.

La segunda propuesta se puede emplear como punto de partida para la investigación de nuevas metodologías en este campo. Se propone explorar distintas formas de construir grafos y encontrar el camino mínimo hacia el contrafactual con el objetivo de obtener el mejor rendimiento posible en conjuntos de datos masivos.

Para terminar, este trabajo se ha enfocado en el desarrollo de explicaciones para problemas de datos tabulares pero la investigación no tiene por qué detenerse aquí, se propone continuar y aplicar estas técnicas y enfoques en otras áreas como pueden ser en el procesamiento de imágenes o en la bioinformática por ejemplo.

Bibliografía

[1] Wachter, S., Mittelstadt, B. and Russell, C. (2017). Counterfactual explanations without opening the black box: Automated decisions and the GDPR. *Harv. JL & Tech*, 31, 841.

[2] Van Lent, M., Fisher, W. and Mancuso, M. (2004). An explainable artificial intelligence system for small-unit tactical behavior. In *Proceedings of the national conference on artificial intelligence*, pp. 900–907.

[3] Confalonieri, R., Coba, L., Wagner, B. and Besold, T. R. (2021). A historical perspective of explainable Artificial Intelligence. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 11(1), e1391.

[4] Adadi, A. and Berrada, M. (2018). Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). *IEEE access*, 6, pp. 52138–52160.

[5] Angelov, P. P., Soares, E. A., Jiang, R., Arnold, N. I. and Atkinson, P. M. (2021). Explainable artificial intelligence: an analytical review. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 11(5), e1424.

[6] Phillips, P. J., Hahn, C. A., Fontana, P. C., Broniatowski, D. A. and Przybocki, M. A. (2020). Four principles of explainable artificial intelligence. *Gaithersburg, Maryland*, 18.

[7] Hassija, V., Chamola, V., Mahapatra, A., Singal, A., Goel, D., Huang, K., ... and Hussain, A. (2024). Interpreting black-box models: a review on explainable artificial intelligence. *Cognitive Computation*, 16(1), pp. 45–74.

[8] Das, A. and Rad, P. (2020). Interpreting black-box models: a review on explainable artificial intelligence. *arXiv preprint arXiv:2006.11371*.

[9] Ribeiro, M. T., Singh, S. and Guestrin, C. (2016)."Why should i trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144.

[10] Lundberg, S. M. and Lee, S. I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

[11] Bhatt, U., Xiang, A., Sharma, S., Weller, A., Taly, A., Jia, Y., ... and Eckersley, P. (2020). Explainable machine learning in deployment. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pp. 648–657.

[12] Saarela, M. and Jauhiainen, S. (2021). Comparison of feature importance measures as explanations for classification models. *SN Applied Sciences*, 3, pp. 1–12.

[13] Verma, S., Dickerson, J. and Hines, K. (2021). Counterfactual explanations for machine learning: Challenges revisited. *arXiv preprint arXiv:2106.07756*.

[14] Guidotti, R. (2022). Counterfactual explanations and how to find them: literature review and benchmarking. *Data Mining and Knowledge Discovery*, pp. 1–55.

[15] Verma, S., Boonsanong, V., Hoang, M., Hines, K. E., Dickerson, J. P. and Shah, C. (2020). Counterfactual explanations and algorithmic recourses for machine learning: A review. *arXiv preprint arXiv:2010.10596*.

[16] Stepin, I., Alonso, J. M., Catala, A. and Pereira-Fariña, M. (2021). A survey of contrastive and counterfactual explanation generation methods for explainable artificial intelligence. *IEEE Access*, 9, pp. 11974–12001.

[17] Byrne, R. M. (2019). Counterfactuals in Explainable Artificial Intelligence (XAI): Evidence from Human Reasoning. In *IJCAI,* pp. 6276–6282.

[18] Cui, Z., Chen, W., He, Y. and Chen, Y. (2015). Optimal action extraction for random forests and boosted trees. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 179–188.

[19] Grath, R. M., Costabello, L., Van, C. L., Sweeney, P., Kamiab, F., Shen, Z. and Lecue, F. (2018). Interpretable credit application predictions with counterfactual explanations. *arXiv preprint arXiv:1811.05245.*

[20] Downs, M., Chu, J. L., Yacoby, Y., Doshi-Velez, F. and Pan, W. (2020). Cruds: Counterfactual recourse using disentangled subspaces. *ICML WHI, 2020*, pp. 1–23.

[21] Klys, J., Snell, J. and Zemel, R. (2018). Learning latent subspaces in variational autoencoders. *Advances in neural information processing systems*, 31.

[22] Pawelczyk, M., Broelemann, K. and Kasneci, G. (2020). Learning model-agnostic counterfactual explanations for tabular data. In *Proceedings of the web conference 2020*, pp. 3126–3132.

[23] Chapman-Rounds, M., Bhatt, U., Pazos, E., Schulz, M. A. and Georgatzis, K. (2021). FIMAP: Feature importance by minimal adversarial perturbation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(13), pp. 11433–11441.

[24] Galhotra, S., Pradhan, R. and Salimi, B. (2021). Explaining black-box algorithms using probabilistic contrastive counterfactuals. In *Proceedings of the 2021 International Conference on Management of Data*, pp. 577–590.

[25] Jia, Y., McDermid, J. and Habli, I. (2021). Enhancing the value of counterfactual explanations for deep learning. In *International Conference on Artificial Intelligence in Medicine*, pp. 389–394. Cham: Springer International Publishing.

[26] Martens, D. and Provost, F. (2014). Explaining data-driven document classifications. *MIS quarterly,* 38(1), pp. 73–100.

[27] Rathi, S. (2019). Generating counterfactual and contrastive explanations using SHAP. *arXiv preprint arXiv:1906.09293.*

[28] White, A. and Garcez, A. D. A. (2019). Measurable counterfactual local explanations for any classifier. *arXiv preprint arXiv:1908.03020.*

[29] Keane, M. T. and Smyth, B. (2020). Good Counterfactuals and Where to Find Them: A Case-Based Technique for Generating Counterfactuals for Explainable AI (XAI). In *Case-Based Reasoning Research and Development: 28th International Conference, ICCBR 2020, Salamanca, Spain, June 8–12, 2020, Proceedings 28*, pp. 163–178. Springer International Publishing.

[30] Warren, G., Smyth, B. and Keane, M. T. (2022). "Better" Counterfactuals, Ones People Can Understand: Psychologically-Plausible Case-Based Counterfactuals Using Categorical Features for Explainable AI (XAI). In *International conference on case-based reasoning*, pp. 63–78. Cham: Springer International Publishing.

[31] Keane, M. T. and Smyth, B. (2022). A few good counterfactuals: generating interpretable, plausible and diverse counterfactual explanations. In *International Conference on Case-Based Reasoning*, pp. 18–32. Cham: Springer International Publishing.

[32] Le, T., Wang, S. and Lee, D. (2020). Grace: generating concise and informative contrastive sample to explain neural network model's prediction. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 238–248.

[33] Dandl, S., Molnar, C., Binder, M. and Bischl, B. (2020). Multi-objective counterfactual explanations. In *International Conference on Parallel Problem Solving from Nature*, pp. 448–469. Cham: Springer International Publishing.

[34] Ramon, Y., Martens, D., Provost, F. and Evgeniou, T. (2020). A comparison of instance-level counterfactual explanation algorithms for behavioral and textual data: SEDC, LIME-C and SHAP-C. *Advances in Data Analysis and Classification*, 14, pp. 801–819.

[35] Wiratunga, N., Wijekoon, A., Nkisi-Orji, I., Martin, K., Palihawadana, C. and Corsar, D. (2021). Discern: Discovering counterfactual explanations using relevance features from neighbourhoods. In *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*, pp. 1466–1473. IEEE.

[36] Wijekoon, A., Wiratunga, N., Nkisi-Orji, I., Martin, K., Palihawadana, C. and Corsar, D. (2021). Discern: Counterfactual explanations for student outcome prediction with Moodle footprints. *CEUR Workshop Proceedings*.

[37] Zhong, J., and Negre, E. (2022). Shap-enhanced counterfactual explanations for recommendations. In *Proceedings of the 37th ACM/SIGAPP Symposium on Applied Computing*, pp. 1365–1372.

[38] Li, P., Bahri, O., Boubrahimi, S. F. and Hamdi, S. M. (2022). Fast counterfactual explanation for solar flare prediction. In *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 1238–1243. IEEE

[39] Cho, S. H. and Shin, K. S. (2023). Feature-Weighted Counterfactual-Based Explanation for Bankruptcy Prediction. *Expert Systems with Applications*, 216, 119390.

[40] Adhikari, A., Tax, D. M., Satta, R. and Faeth, M. (2019). LEAFAGE: Example-based and Feature importance-based Explanations for Black-box ML models. In *2019 IEEE international conference on fuzzy systems (FUZZ-IEEE)*, pp. 1–7. IEEE.

[41] Poyiadzi, R., Sokol, K., Santos-Rodriguez, R., De Bie, T. and Flach, P. (2020). FACE: feasible and actionable counterfactual explanations. In *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society,* pp. 344–350.

[42] Mothilal, R. K., Sharma, A. and Tan, C. (2020). Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 conference on fairness, accountability, and transparency,* pp. 607–617.

[43] Albini, E., Long, J., Dervovic, D. and Magazzeni, D. (2022). Counterfactual shapley additive explanations. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency,* pp. 1054–1070.

[44] Sohns, J. T., Garth, C. and Leitte, H. (2023). Decision Boundary Visualization for Counterfactual Reasoning. In *Computer Graphics Forum,* 42(1), pp. 7–20.

[45] Gower, J. C. (1971). A general coefficient of similarity and some of its properties. *Biometrics,* pp. 857–871.

[46] Dijkstra, E. W. (2022). A note on two problems in connexion with graphs. In *Edsger Wybe Dijkstra: His Life, Work, and Legacy,* pp. 287–290.

[47] Janosi, A., Steinbrunn, W., Pfisterer, M. and Detrano, R. (1988). Heart Disease. *UCI Machine Learning Repository.* https://doi.org/10.24432/C52P4X.

[48] Becker, B. and Kohavi, R. (1996). Adult. *UCI Machine Learning Repository.* https://doi.org/10.24432/C5XW20.

[49] Hofmann, H. (1994). Statlog (German Credit Data). *UCI Machine Learning Repository.* https://doi.org/10.24432/C5NC77.

[50] Teboul, R. (2021). Diabetes Health Indicators Dataset. *Kaggle.* https://www.kaggle.com/datasets/alexteboul/diabetes-health-indicators-dataset.

[51] Realinho, V., Martins, M. V., Machado, J. and Baptista, L. (2021). Predict Students' Dropout and Academic Success. *UCI Machine Learning Repository.* https://doi.org/10.24432/C5MC89.

[52] Raschka, S. (2018). MLxtend: Providing machine learning and data science utilities and extensions to Python's scientific computing stack. *The Journal of Open Source Software,* 3(24).

[53] Mothilal, R. K., Mahajan, D., Tan, C. and Sharma, A. (2021). Towards unifying feature attribution and counterfactual explanations: Different means to the same end. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society,* pp. 652–663.

[54] Wijekoon, A., Wiratunga, N., Nkisi-Orji, I., Palihawadana, C., Corsar, D. and Martin, K. (2022). THow close is too close? The role of feature attributions in discovering counterfactual explanations. In *International Conference on Case-Based Reasoning,* pp. 33–47. Cham: Springer International Publishing.