ESCUELA DE INGENIERÍAS AGRARIAS DE LA UEX

USO DE TÉCNICAS DE APRENDIZAJE AUTOMÁTICO PARA PREDECIR LA ADHERENCIA A LA DIETA MEDITERRÁNEA EN FUNCIÓN DE FACTORES DEMOGRÁFICOS, ECONÓMICOS, CULTURALES, PSICOLÓGICOS Y ANTROPOMÉTRICOS

TRABAJO FIN DE GRADO

GRADO EN CIENCIA Y TECNOLOGÍA DE LOS ALIMENTOS

Seal Kevin Boy Abad

Badajoz, enero de 2024.

TRABAJO FIN DE GRADO

USO DE TÉCNICAS DE APRENDIZAJE AUTOMÁTICO PARA PREDECIR LA ADHERENCIA A LA DIETA MEDITERRÁNEA EN FUNCIÓN DE FACTORES DEMOGRÁFICOS, ECONÓMICOS, CULTURALES, PSICOLÓGICOS Y ANTROPOMÉTRICOS

GRADO EN CIENCIA Y TECNOLOGÍA DE LOS ALIMENTOS

AUTOR: Seal Kevin Boy Abad

TUTOR: Valentín Masero Vargas

Tutor

Fdo:.....

ÍNDICE

1. RESUMEN

En este estudio se ha desarrollado un modelo de aprendizaje automático orientado a predecir la adherencia a la dieta mediterránea en función de diversos factores. Se contó con la participación de 286 estudiantes matriculados en el año académico 2017-2018, cada uno de los cuales completó un cuestionario comprehensivo que abordaba, por un lado, el cuestionario Mediterranean Diet Adherence Screener, una herramienta diseñada para cuantificar la adhesión a la dieta mediterránea. Los resultados de este cuestionario se emplearon como variable objetivo o "target". Por otro lado, se recogieron datos de los factores demográficos, socioeconómicos, culturales, psicológicos y físicos o antropométricos utilizando cuestionarios validados en la literatura científica para evaluar cada uno de estos aspectos. Estos factores se utilizaron como características o "features" en el diseño del modelo. El modelo de aprendizaje automático con mayor precisión fue XGBoost con 84.21% de *accuracy*, y las características con más poder predictivo fueron 8: a0 (Facultad), a1 (Grado), a3(Edad), a4 (Sexo), c1 (FCQ-SQP), e (IMC), d41m ((¿Qué nivel educativo tiene tu padre?),), d42m ((¿Qué actividad laboral realiza tu madre?).

2. INTRODUCCIÓN

2.1. Alimentación y su impacto en la salud

La historia de la especie humana se puede explicar con bastante precisión mediante la historia de la alimentación (Navarro-Prado, 2016). La alimentación ha sufrido importantes cambios desde el hombre prehistórico a nuestros días. Si para nuestros ancestros el alimento significaba la supervivencia, y por tanto la mayor preocupación se centraba en el gran esfuerzo que suponía conseguirlo y en la cantidad de éste; hoy, cuando el hombre dispone de una gran variedad de alimentos, su atención se centra en la elección del alimento adecuado, acorde a su apetencia (Bolaños Ríos, 2009; Flandrin, 2004).

Los alimentos son sustancias naturales o transformadas que contienen uno o, más a menudo, varios elementos nutritivos. Los seres humanos los ingieren para saciar el hambre o por otros motivos (Cervera et al., 2004). La alimentación, influye definitivamente en la salud, por lo que debe contener una cantidad suficiente de los diferentes macro y micronutrientes para cubrir la mayoría de las necesidades fisiológicas (De Luis et. al, 2010). La regulación de la alimentación es la resultante de una serie de procesos que incluyen señales hormonales, metabólicas y neuronales, integradas en el hipotálamo, órgano encargado de generar respuestas de saciedad o de búsqueda de alimentos (Troncoso y Amaya, 2009; Valenzuela, 2009). Realizar una dieta equilibrada y adaptada a las necesidades de las diferentes etapas de la vida, es importante para un adecuado crecimiento físico y psicológico de la persona, para prevenir enfermedades y para obtener un óptimo estado de salud, entendida como bienestar y capacidad funcional (OMS, 1948; Riba, 2002; Rodriguez, 1995; Salleras, 1985; Sancho et al., 2007).

2.2.Hábitos alimentarios

Los hábitos alimentarios se definen como una serie de conductas y comportamientos colectivos, que influyen en la manera de escoger, preparar y consumir un determinado alimento,

el cual debe cumplir con un aporte nutricional, que le permita al cuerpo obtener la energía suficiente para el desarrollo de las actividades diarias (Paillacho Chamorro y Solano Andrade, 2011). Por su parte, Cervera et al. (2004) añaden que los hábitos alimenticios son aquellos procesos por el cual un individuo selecciona sus alimentos. Teniendo en cuenta, para ello, la disponibilidad y al aprendizaje obtenido de su entorno. Este aprendizaje, afirman los autores, son influenciados a su vez por factores socioculturales, psicológicos, geográficos y socioeconómicos.

El aprendizaje de los hábitos alimentarios está condicionado por numerosas influencias procedentes, sobre todo, de la familia, del ámbito escolar y a través de la publicidad (Birch & Fisher, 1998; Stroebele & de Castro, 2004). Estos hábitos se suelen adquirir desde la infancia y pueden ser el origen de patologías crónicas relacionadas con la malnutrición por exceso de ingesta de alimentos en edades posteriores (Pelto et al., 1989; Duarte et al., 2016; Busdiecker et al., 2000; Royo, 2017). En un principio, la familia desempeña un papel fundamental en la configuración del patrón alimentario del niño (Story, Neumark, & French, 2002). Al alcanzar la adolescencia el papel de la familia pierde relevancia y el grupo de amigos y las referencias sociales se convierten en condicionantes claves de la dieta del joven adolescente (Cusatis & Shannon, 1996; Montero, Úbeda, & García, 2006).

Las instituciones sanitarias internacionales ven en los hábitos alimentarios una vía para mejorar la salud, y resaltan su importancia. Así, la FAO (Food and Agriculture Organization) incluye entre sus objetivos mejorar los hábitos alimentarios que promuevan una mejor calidad de vida y reconoce la capacidad de la educación nutricional para mejorar por sí sola el comportamiento dietético y el estado nutricional de las personas (FAO, 2011). Esto es importante debido a que, como hemos afirmado anteriormente, los hábitos alimentarios tienen un gran impacto en la masa corporal y evitan el desarrollo de enfermedades (Brown & Roberts, 2012; Cuenca, 2011). Algunas de las enfermedades que se desarrollan debido a los factores de riesgo producto de malos hábitos alimentarios son las enfermedades crónicas no transmisibles como el cáncer, las enfermedades cardiovasculares y los accidentes cerebrovasculares (Canova-Barrios, 2017). Es necesario que se produzca la modificación o abandono de hábitos alimentarios insanos y erróneos, para poder conseguir una dieta sana y equilibrada (Escuela de alimentación, 2012; Alonso et al., 2004; de Cantabria, 2010).

Por otro lado, los efectos de la globalización están produciendo un cambio en los estilos de vida mediterráneos, y especialmente notables en los hábitos alimentarios de la población joven, por ser los más expuestos a los diversos cambios transfronterizos (Durá & Castroviejo, 2011; Ortiz-Moncada et al., 2012). Las transformaciones emocionales y fisiológicas propias de la juventud (García, 2002), las presiones publicitarias y los patrones estéticos actuales, son algunos de los factores que influyen en su falta de adhesión a la dieta mediterránea.

En el caso de la población universitaria, a todo esto, se añade la asunción de nuevas responsabilidades en la compra de alimentos, la elaboración de sus menús y la organización de los horarios de comidas que resultan ser bastante irregulares (Martínez et al., 2005; Montero et al., 2006). Algunos autores como Christoph y An (2018) han afirmado que los estudiantes universitarios son una población con elevado riesgo de mala alimentación y hábitos alimentarios.

La etapa universitaria suele coincidir con el momento en el que el adolescente pasa a ser adulto, y el inicio de su independencia (Ruiz et al., 2010). El abandono del domicilio familiar los convierte por primera vez en responsables de su autocuidado y por tanto, de su alimentación. Así, la responsabilidad e independencia a la hora de elegir los alimentos y la preparación de los mismos serán aspectos, a menudo, deteriorados. El consumo frecuente de productos de contenido nutricional inadecuado o el alto consumo de alcohol derivan a trastornos en la conducta alimentaria (Ortiz-Moncada et al., 2012; Magaña, 2003; Greppi, 2012).

2.3. Malnutrición: un problema global de salud

El deterioro de los hábitos alimentarios y la tendencia al sedentarismo han producido un incremento alarmante en la prevalencia de la obesidad. Es más, la obesidad y sus determinantes son factores de riesgo de tres de las cuatro principales causas de enfermedades no transmisibles (ENT) en todo el mundo, incluidas las enfermedades cardiovasculares, la diabetes de tipo 2 y ciertos tipos de cáncer (Swinburn et al., 2019).

Según la Organización Mundial de la Salud (OMS), en la región europea se ha alcanzado una epidemia de tasas de sobrepeso y obesidad, afectando al 59% de los adultos y a casi 1 de cada 3 niños (OMS, 2022). En un informe sobre obesidad titulado “Obesity: preventing and managing the global epidemic. Report of a WHO consultation”, la OMS define la obesidad como una enfermedad crónica compleja y multifactorial, en gran parte prevenible, caracterizada

por la acumulación anormal o excesiva de grasa (OMS, 2000). El sobrepeso se identifica cuando el Índice de Masa Corporal (IMC) es mayor a 25 kg/m² y la obesidad cuando supera los 30 kg/m². El IMC, a pesar de sus limitaciones, es ampliamente utilizado para evaluar el grado de sobrepeso y obesidad en la población, y se calcula dividiendo el peso en kilogramos por el cuadrado de la estatura en metros. Este índice se relaciona estrechamente con el porcentaje de grasa corporal y la masa grasa corporal en la mayoría de los grupos poblacionales. (Nuttall, F. Q., 2015).

En un estudio realizado en 195 países por GBD Diet Collaborators (2019) se halló que alrededor de 17 millones de muertes en el año 2019 estuvieron asociadas a comportamientos alimentarios poco saludables, como la alta ingesta de azúcares, grasas y sodio, así como un bajo consumo de frutas, verduras y cereales integrales. Por consiguiente, implementar una dieta equilibrada de la manera más sana posible es importante para controlar aquellos parámetros metabólicos que ayuden a conservar la salud cardiovascular y mental de las personas (González Valero et al., 2017).

2.4.Dieta mediterránea y sus beneficios para la salud

La dieta mediterránea (DM), reconocida por la Organización Mundial de Salud como un patrón saludable de alimentación (Mayo Clinic, 2023), se considera un modelo de alimentación y estilo de vida saludable asociado a la reducción del riesgo de diferentes enfermedades crónicas no transmisibles, como la diabetes tipo 2 y los problemas cardiovasculares (Estruch et al., 2018), junto a una mayor esperanza de vida (Martínez-González et al., 2018).

Esta dieta se caracteriza por combinar alimentos como el aceite de oliva, cereales, legumbres, verduras, hortalizas, frutas, frutos secos, pescado fresco y bebidas fermentadas, como el vino durante las comidas, y la ingesta moderada de lácteos, carnes y un bajo consumo de carnes rojas, embutidos y alimentos procesados (Sánchez-Muniz y Goñi, 2006). Sus efectos sobre la salud se deben a su bajo contenido en ácidos grasos saturados, al aporte de grasas monoinsaturadas del aceite de oliva y poliinsaturadas de los frutos secos y del pescado azul, así como la abundancia de sustancias antioxidantes procedentes de frutas y hortalizas (Estruch et al., 2018).

Se ha observado que las personas que residen en países mediterráneos tienen un patrón diferente de salud y enfermedad. Las personas que viven en estas zonas (Grecia, España, Italia, Francia) tienen una mayor esperanza de vida. Estas diferencias, que no se deben solo a factores genéticos, podrían estar relacionadas con factores ambientales, entre los cuales los patrones de alimentación podrían tener un papel relevante (Carbajal y Ortega, 2001).

En numerosos estudios epidemiológicos se ha observado su trascendencia para la salud. La alta adherencia a la DM se ha asociado a una disminución significativa del riesgo de mortalidad (Trichopoulou et al., 2003; Sánchez-Villegas et al., 2006; Kontogianni et al., 2010; Sofi et al., 2005; Trichopoulou, 2005). Estudios revelan que la DM reduce el riesgo de enfermedades cardiovasculares (González et al., 2023), previene la diabetes tipo 2 (Mayo Clinic, 2023), protege contra ciertos tipos de cáncer, debido a la presencia de antioxidantes y fitoquímicos con propiedades antiinflamatorias y antitumorales (Hospital Clínic, 2023), y, también, reduce el riesgo de deterioro cognitivo y demencia al estimular la neurogénesis y la plasticidad neuronal, debido a la reducción del estrés oxidativo y la inflamación cerebral (Samieri et al., 2013).

La adherencia a la DM es un factor importante para la salud. Sin embargo, una serie de factores pueden dificultar la adherencia a la DM, incluyendo los factores demográficos, económicos, socioculturales, psicológicos y antropométricos.

2.5. Factores que influyen en los hábitos alimentarios

La influencia de diferentes factores en los hábitos alimentarios y la elección de alimentos ha sido explicada en diferentes estudios que han analizado los determinantes demográficos, socioeconómicos, culturales, psicológicos y biológicos que afectan las preferencias y el consumo de alimentos de las personas (Eufic, 2019).

En cuanto a los factores económicos, se observó, por ejemplo, en el trabajo de Freech (2003) que una reducción del 50% del precio de un snack saludable aumentó las ventas de este un 93%, y también que una reducción del 50% en el precio de frutas frescas y zanahorias baby en dos cafeterías de escuelas secundarias aumentó cuatro veces las ventas de frutas frescas y dos veces las ventas de zanahorias. En una revisión sistemática publicada por Zorbas et al. (2018), se observó que los alimentos saludables, como frutas y verduras, eran percibidos como caros en comparación con los alimentos poco saludables y que la comida rápida se consideraba

más barata que la comida preparada en casa. En el mismo estudio se observó que algunas personas invertían más dinero en alimentos saludables con tal de ahorrar dinero a largo plazo en atención médica. También se identificó que la asequibilidad de las dietas era más relevante para los grupos socioeconómicos más bajos y para los estudiantes. Estos son ejemplo de cómo el factor económico puede afectar el comportamiento alimentario.

El factor psicológico juega un papel crucial en el comportamiento alimentario. En los seres humanos, una serie de características psicológicas pueden influir en el comportamiento alimentario: estrés, alimentación restringida, neuroticismo, depresión y disforia premenstrual (Polivy, J., 1996). El estudio de Zorbas (2018) señaló los factores psicológicos que influyen en la alimentación son: autopercepción, estado emocional y bienestar mental. Según participantes de múltiples estudios, comer puede estar impulsado por múltiples estados emocionales, usualmente relacionados con el estrés, los antojos y el comer para sentirse cómodo. Por otro lado, también se encontró que la adherencia a patrones de alimentación saludable se veía facilitada por la creencia de que puede mejorar la apariencia y el atractivo físico.

El motivo por el que una persona se adhiere a una dieta tiene un impacto en el largo plazo en la adherencia a dicha dieta. Como afirma Gibson (2006) “la dieta autoimpuesta parece resultar en atracones de comida una vez que hay comida disponible y en manifestaciones psicológicas como la preocupación por la comida y la alimentación”.

Algunos autores sugieren que el exceso de consumo de alimentos es el resultado de la dependencia de los alimentos (Pelchat, 2009), de una estrategia para afrontar situaciones ansiógenas (Blum, Liu, Shriner y Gold, 2011), o que surge como respuesta a emociones negativas (Gibson y Green. 2002; Macht y Mueller. 2007).

Las personas estresadas pueden ser más propensas a comer alimentos poco saludables: la dulzura y las señales sensoriales de alta densidad de energía, como la textura grasa, pueden mejorar el estado de ánimo y mitigar los efectos del estrés a través de la neurotransmisión cerebral opioidérgica y dopaminérgica (Polivy, J., 1996). Además, continúa Polivy, los alimentos dulces y grasos bajos en proteínas también pueden proporcionar alivio del estrés en personas vulnerables a través de una mayor función del sistema serotoninérgico.

La DM, debido a sus componentes nutricionales (Kastorini et al., 2011; Yahfoufi et al., 2018; Su et al., 2014), tiene efectos antiinflamatorios, y en ensayos clínicos aleatorizados

de agentes antiinflamatorios no esteroideos (p.e. inhibidores de citoquinas) se ha demostrado que estos pueden reducir significativamente los síntomas de la depresión (Köhler-Forsberg, 2019). Varios ensayos recientes en personas con depresión han mostrado mejoras moderadamente significativas al seguir intervenciones basadas en la dieta mediterránea. Esto podría significar que no sólo los factores psicológicos tienen un efecto sobre el comportamiento alimentario, sino que el comportamiento alimentario tiene un efecto sobre los factores psicológicos, resaltando la importancia de este factor.

En cuanto al factor antropométrico o físico, se ha observado que hay una relación entre un alto IMC y saltarse comidas, lo cual desemboca en una ingesta compensatoria de alimentos, durante la cual se consume un número de calorías superior al de una comida norma (Kaye et al., 1992; Stewart et al., 2022; Herman et al., 1987). El IMC elevado también ha sido asociado al consumo de sustancias para evitar sentir hambre, el uso de diuréticos y el uso de laxantes se ha asociado con un IMC elevado (Rodríguez Santamaría, 2009).

En cuanto a la explicación demográfica, los rangos de edades parecen ser un factor influyente, ya que en el estudio de Zorbas et al. (2018) se observó que en las universidades carecían de opciones saludables en las cafeterías, máquinas expendedoras y eventos sociales. También ocurre que, en las universidades, la carga de estudio es alta y ese estilo de vida promueve el consumo de alimentos convenientes. Por otro lado, se observó en el estudio de Rodríguez Santamaría et al. (2009) una estrecha relación entre el nivel educativo de las personas y su IMC. Las personas con nivel educativo más bajo tienen un IMC más alto. Con respecto a las diferencias entre sexos, en el estudio de González-Arratia López Fuentes et al. (2016) se halló que los hombres experimentaron emociones agradables y de mayor intensidad durante el consumo de agua de sabor, barra de cereal, bistec, caldos, enchiladas, milanesa, pescado, pizza, quesadillas y tacos; mientras que las mujeres experimentaron emociones agradables y mayor intensidad durante el consumo de chocolate y licuado de fruta. Siguiendo con estas diferencias, algunos estudios encontraron que los hombres tienden a ser consumidores pasivos, a diferencia de las mujeres; es decir, consumen los alimentos que están directamente delante de ellos (Dumbrell, 2008). Esto demuestra que puede haber diferencias en cuanto a la preferencia alimentaria en base al sexo.

Con respecto a la explicación cultural, Zorbas et al. (2018) encontraron que la aceptabilidad social de la alimentación saludable fue baja, lo cual supone una barrera para

adoptar patrones saludables de alimentación. Cabe destacar que si eran los familiares, amigos o compañeros los que apoyaban la alimentación saludable, las personas adoptaban esas dietas. Por otro lado, se halló que el consumo de bebidas alcohólicas y snacks poco saludables como dulces estaba relacionado con la sociabilidad, contrario al consumo de frutas y verduras.

Tras observar cómo los diferentes factores influyen en el comportamiento alimentario, se puede sostener que la alimentación exhibe un carácter multifactorial, lo cual implica la necesidad de utilizar técnicas analíticas más sofisticadas que las tradicionales para abordar esa complejidad de forma efectiva.

2.6.Inteligencia artificial y su impacto en la investigación sobre nutrición

Los avances tecnológicos y computacionales han posibilitado a los investigadores el uso de datos de alta dimensión, lo que les permite descubrir patrones y relaciones complejas entre datos que no podían identificarse mediante métodos tradicionales de análisis de datos. El big data, la computación en la nube, la inteligencia artificial y el aprendizaje automático son algunos de estos avances. En concreto, el aprendizaje automático (ML, por sus siglas en inglés), una rama de la inteligencia artificial (IA), se ocupa del desarrollo de algoritmos y técnicas que capacitan a las computadoras para aprender y adquirir inteligencia basándose en experiencias pasadas; es un proceso computacional en el que el sistema es capaz de identificar y comprender datos de entrada y en consecuencia aplicar la información adquirida para tomar decisiones y hacer predicciones sobre diversos fenómenos (Lantz B., 2019). Wiens & Shenoy (2018) definen ML, de forma más concisa, como el estudio de herramientas y métodos para identificar patrones en los datos, con el objetivo de encontrar un modelo que explique mejor los datos.

En lo que respecta a la nutrición, el uso de técnicas de ML se puede utilizar para predecir la adherencia a la dieta mediterránea, ayudando a superar los desafíos que surgen en la investigación sobre nutrición. Como señala Daniel Kirk et al. (2022), los problemas nutricionales suelen ser complejos y multifactoriales, y su resolución a través de métodos tradicionales de análisis de datos a menudo es difícil.

La evidencia que se usa para dar pautas de alimentación saludable se suele obtener de estudios epidemiológicos o clínicos grandes, en los cuales se establecen promedios en un intento de proporcionar consejos nutricionales a la población, sin embargo, con esos estudios no se consigue capturar los efectos biológicos individuales que puede tener la nutrición (Kirk

et al., 2021). Por lo tanto, herramientas alternativas, como las técnicas de ML, pueden desempeñar un papel fundamental al descubrir patrones y relaciones complejas entre los diferentes factores, revelando qué factores son los que más influyen en la adherencia a la dieta mediterránea, considerando las características específicas de un determinado segmento de la población.

Estas técnicas han resultado útiles en desarrollar intervenciones personalizadas en el campo de la nutrición (Raphaeli y Singer, 2021). Ese tipo de intervenciones han demostrado ser más efectivas que métodos genéricos, como se observa en los estudios de Celis-Morales et al., 2017 y Celis-Morales & Lara, 2015, en el que un estudio aleatorizado realizado en Europa, los grupos que recibieron asesoramiento personalizado mantuvieron en su dieta cambios considerados como saludables. En la revisión sistemática realizada por Kirk, D. et al. (2021), en la que se analizaron 60 estudios que utilizaron aprendizaje automático en nutrición personalizada o áreas relacionadas, se observó que el uso de técnicas de ML en la nutrición personalizada tiene mucho potencial, permitiendo desarrollar intervenciones de mayor rendimiento y eficiencia, además de proporcionar una visión más completa y precisa de la respuesta individual a las intervenciones de nutrición.

2.7.Relevancia del estudio

La relevancia de este estudio se fundamenta en la necesidad de comprender la compleja interacción de los factores que contribuyen a la adopción de hábitos alimentarios poco saludables entre los estudiantes universitarios. Como afirma Maza (2022), más del 87% de los estudios sobre los hábitos alimentos en estudiantes universitarios señalan que los estudiantes tienen malos hábitos alimentarios, mientras que solo un 2% evidenciaba una adopción de patrones alimentarios saludables. En el mismo estudio, se identificó que los hábitos alimentarios más comunes entre los estudiantes universitarios fueron el bajo consumo de frutas y verduras (71,93%), el alto consumo de dulces (57,89%), saltarse las comidas (45,61%), el alto consumo de grasas (45,61%), el consumo recurrente de comidas rápidas y productos ultra procesados (45,61%) y el bajo consumo de lácteos y derivados (38,60%), cuestiones que podrían abordarse con una mayor adherencia a la dieta mediterránea.

La falta de adherencia a una dieta saludable, como la dieta mediterránea, representa un desafío significativo para la salud pública y la economía en la sociedad moderna. Esta falta de

adherencia conlleva una serie de costos que afectan tanto a nivel individual como a nivel colectivo, y es esencial abordar este problema de manera efectiva.

Este estudio pretende analizar el comportamiento alimentario de los estudiantes de la Universidad de Extremadura, campus de Badajoz utilizando técnicas de ML. Los resultados de esta investigación ofrecerán una visión integral de cómo los diferentes factores actúan sobre la adherencia a la dieta mediterránea. Este enfoque tiene el potencial de proporcionar una herramienta para comprender mejor y abordar el problema de la falta de adherencia a la dieta mediterránea.

3. OBJETIVOS

3.1.Objetivo principal

Crear un modelo para predecir la adherencia a la dieta mediterránea usando técnicas de aprendizaje automático y, al mismo tiempo, identificar los factores demográficos, socioeconómicos, culturales, psicológicos y físicos que están asociados con una mayor adherencia.

3.2.Objetivos específicos

1. Recopilar datos de una muestra representativa de la población española. Se recopilarán a través de un cuestionario que incluirá datos sobre la adherencia a la dieta mediterránea e información demográfica, socioeconómica, cultural, psicológica y física de los participantes.

2. Desarrollar un modelo de aprendizaje automático que pueda predecir la adherencia a la dieta mediterránea en base los datos recopilados. El modelo se entrenará después de limpiar y procesar los datos, realizar un análisis exploratorio para comprender las relaciones entre los diferentes factores y seleccionar las características relevantes.

3. Comparar el rendimiento del modelo desarrollado con el rendimiento de otros modelos de aprendizaje automático. Se evaluará la precisión y el desempeño de los modelos mediante métricas relevantes (exactitud, sensibilidad, precisión, etc.). Se empleará la validación cruzada para garantizar la robustez de los modelos.

4. Analizar los resultados obtenidos para determinar la efectividad del modelo en la predicción de la adherencia a la dieta mediterránea. Identificar las características más influyentes en la predicción de la adherencia.

4. MATERIAL Y MÉTODOS

4.1.Diseño del estudio y participantes

El presente trabajo técnico consiste en desarrollar un modelo de ML basado en factores culturales, demográficos, socioeconómicos y físicos de los estudiantes de grado de la Universidad de Extremadura (Campus de Badajoz), que simultáneamente no sean docentes o personal de servicios.

Según datos del Observatorio de Indicadores de la Universidad de Extremadura (2017), el total de la población estudiantil matriculada en facultades del Campus de Badajoz era de 8.769. Para el desarrollo de este trabajo no se ha aplicado la hipótesis de máxima varianza. En su lugar, se ha escogido el 26% como proporción de la variable objetivo, la adherencia a la dieta mediterránea, tras hacer una revisión de la literatura. En estudios previos, se encontró que la alta adherencia variaba entre un 53% y un 14,4% en poblaciones similares (Cervera Burriel, F. et al., 2013; Chacón-Cuberos, R. et al., 2016; Cervera Burriel, F. et al., 2021; Barrios-Vicedo, R. et al., 2015). Se escogió la alta adherencia como variable objetivo porque es más consistente con la definición de adherencia. El muestreo se realizó utilizando la proporción de variable objetivo de 26%, un nivel de confianza 95% para tener un alto grado de confianza en la representatividad de la muestra en relación con la población de estudio, y una precisión de 5% para limitar el margen de error. Se obtuvo un tamaño muestral de 286 estudiantes, distribuidos según sexo con 129 hombres y 157 mujeres, elegidos de forma representativa por Escuela o Facultad.

Los estudiantes fueron seleccionados de forma aleatoria y la recolección de sus datos se realizó de dos formas: mediante entrevista personal y rellenando un cuestionario online creado con la herramienta Google Forms.

4.2. Instrumentos y materiales

4.2.1. Software utilizado

Para el desarrollo del proyecto se usó la Google Colaboratory, un entorno en el que se puede escribir y ejecutar código en el navegador.

4.2.2. Herramientas de medida

Todass las herramientas de medida que a continuación se describen están adjuntadas en la sección de Anexos (1-7).

4.2.2.1. Adherencia a la dieta mediterránea

La adherencia a la dieta mediterránea se midió utilizando la Mediterranean Diet Adherence Screener (MEDAS) (Estruch et al., 2018; Schröder et al., 2011). Este cuestionario consta de 14 preguntas sobre la frecuencia de consumo de alimentos y los hábitos de ingesta de alimentos considerados característicos de la dieta mediterránea española. Cada pregunta se puntúa con 0 o 1. La puntuación total varía entre 0 y 14, lo que permite diferenciar tres niveles de adherencia a la dieta mediterránea: baja (0 - 6), media (7 - 8) y alta (≥ 9) (León-Muñoz et al., 2012).

4.2.2.2. Factores demográficos

En el cuestionario se incluyeron preguntas sobre el grado que cursaba el estudiante, el curso, su edad, el sexo y el país de nacimiento.

4.2.2.3. Factores psicológicos

Los factores psicológicos se midieron utilizando dos instrumentos. El primero de ellos es el cuestionario Food Choice Questionnaire Spanish Version (FCQ-SP). Una versión adaptada y validada para la población española, desarrollada por Jáuregui-Lobera y Bolaños-Ríos (2011).

Este instrumento evalúa la importancia que los individuos otorgan a siete factores relacionados con la elección de los alimentos: salud, estado de ánimo, conveniencia, atractivo sensorial, precio, peso y familiaridad. El FCQ-SP consta de 34 ítems con opciones de respuesta múltiple y una escala tipo Likert de siete posibilidades. La escala de evaluación va desde “nada

importante" a "muy importante" y mide siete factores subyacentes asociados con las razones para elegir los alimentos (Jáuregui-Lobera y Bolaños-Ríos, 2011).

El FCQ-SP posee propiedades psicométricas adecuadas para su aplicación en la población española. Es más, el FCQ-SP también se ha utilizado en otros países de habla hispana como México (García-González et al., 2018) para estudiar los hábitos nutricionales y las creencias sobre los alimentos en diferentes grupos de edad.

El segundo, el cuestionario Three-Factor Eating Questionnaire-R18 Spanish Version (TFEQ-SP). Una versión, también, adaptada y validada para la población española, desarrollada por Jáuregui-Lobera et al. (2014).

Este instrumento mide tres aspectos diferentes de la conducta alimentaria: (a) restricción cognitiva (RC), definida como la restricción consciente de la ingesta de alimentos con el objetivo de controlar el peso corporal y/o promover la pérdida de peso; (b) ingesta incontrolada (II), la tendencia a comer más de lo habitual debido a la pérdida de control sobre la ingesta con una sensación subjetiva de hambre; y (c) ingesta emocional (IE), incapacidad para resistirse a las señales emocionales, comer como respuesta a diferentes emociones negativas. El cuestionario consta de 18 ítems que se miden en una escala de respuesta de 4 puntos (definitivamente cierto: 1, mayoritariamente cierto: 2, mayoritariamente falso: 3, definitivamente falso: 4) y las puntuaciones de los ítems se suman en puntuaciones de subescala: RC, II y IE" (Jáuregui-Lobera et al., 2014).

El TFEQ-SP es una herramienta útil para medir diferentes estilos de comportamiento alimentario en la población española de edades entre 12 y 27 años (Jáuregui-Lobera et al., 2014), un rango de edad que comprende a la mayoría de los individuos de nuestra población de estudio.

4.2.2.4. Factores culturales

Se utilizó un cuestionario de 6 ítems sobre fuentes de conocimiento acerca de alimentos, adaptado a partir otros cuestionarios en los que se estudiaba el comportamiento alimentario (Morris, 2010; Aranceta, 2015). Las preguntas abordan la formación académica en nutrición o alimentación, el interés en conocer información nutricional de los alimentos, la prioridad que se le da a los diferentes nutrientes en las etiquetas, influencia de la publicidad en la elección de

los alimentos, identificación de los canales publicitarios sobre alimentación más influyentes y los recursos que se suelen utilizan para obtener información sobre alimentación.

4.2.2.5. Factores socioeconómicos

Se utilizó un cuestionario de 9 ítems a partir de otros cuestionarios que estudiaba también los hábitos alimentarios (Ramos et al., 2002; García, 2004). Las preguntas abordan temas como las fuentes de ingresos de los estudiantes, el tipo de residencia, si se vive sólo o acompañado, la situación vital y el nivel de educación de sus padres.

4.2.2.6.Factores antropométricos

Se utilizó un cuestionario de 2 ítems. El primero solicitaba el peso del individuo, en kilogramos. El segundo, la altura en centímetros. A partir de las medidas de peso y altura se calculó el Índice de Masa Corporal (IMC), cuya fórmula es IMC = peso (kg)/altura (m)², y se utilizó la clasificación de la Organización Mundial de la Salud (OMS) para clasificar los resultados del IMC (Durá & Castroviejo, 2011; Marrodán et al., 2013; Organización Mundial de la Salud [OMS], 2012).

Es importante destacar que, a pesar de que ambas medidas se basaron en autorreportes proporcionados por los participantes, en estudios epidemiológicos previos se ha hallado que las medidas auto reportadas y las medidas objetivas están usualmente correlacionadas de forma significativa (Rimm et al., 1990; Weaver et al., 1996). Además, teniendo en cuenta que la mayoría de la población universitaria, en estudios anteriores, ha sido clasificada como “normopeso” según la clasificación del IMC (Prieto-González et al., 2022; Linares-Manrique et al., 2016; Yahia et al., 2008; Vadeboncoeur et al., 2014), y que se ha hallado un menor porcentaje de error en la estimación del IMC cuando la clasificación es "normopeso" (Marrodán et al., 2013; Sánchez-Álvarez et al., 2012), se considera que este grupo es fiable a la hora de autoreportar estas medidas.

4.3. Variables

4.3.1. Variable dependiente o “target”

Según el libro "Machine Learning: A Probabilistic Perspective" de Kevin P. Murphy (2012) y "Elements of Statistical Learning" de Trevor Hastie, Robert Tibshirani y Jerome

Friedman (2009), se define la variable dependiente como "la variable que se desea predecir a partir de las variables independiente".

En este trabajo la variable “target” es el resultado obtenido de la encuesta MEDAS. Esta variable es la que se predice en base a los datos de entrenamiento.

4.3.2. Variables independientes o “features”

Murphy (2012) y Hastie, Tibshirani y Friedman (2009) definen las variables independientes como las variables que se utilizan para predecir la variable dependiente.

Las variables independientes o “features” serían los siguientes: IMC, FCQ-SP, TFEQ-R18, factores demográficos, factores socioeconómicos y factores culturales. Estas son las variables que se utilizan para predecir la variable dependiente. Estas son las variables que se utilizan para entrenar el modelo.

4.4. Análisis de datos

4.4.1. Preparación inicial de los datos y creación del DataFrame

En esta fase inicial se llevó a cabo el montaje de los datos a la plataforma de Google Colab, se creó el DataFrame (df) y se hizo una descripción básica de los datos.

Las librerías que inicialmente se usaron fueron importadas y aquellas que se usaron con frecuencia se les asignó un alias. El código para ello se observa en la tabla 1.

Código para la asignación de alias

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

FIGURA. 4.1. Código para la asignación de alias a las librerías

“import pandas as pd”: esta línea indica que se está importando una biblioteca de Pandas a la plataforma de Google Colab y se le está asignando el alias “pd”. Pandas es un paquete de Python que proporciona estructuras de datos rápidas, flexibles y expresivas diseñadas para facilitar el trabajo con datos ‘relacionales’ o ‘etiquetados’. Su objetivo es ser el bloque de construcción fundamental de alto nivel para hacer análisis de datos prácticos y reales en Python (McKinney, 2010). El alias “pd” es una convención común que facilita el uso de las

funciones y clases de Pandas. Por ejemplo, para crear un objeto DataFrame (una de las estructuras de datos principales de Pandas) se puede usar pd.DataFrame() en lugar de pandas.DataFrame (pandas, 2023).

"import numpy as np": esta línea importa la biblioteca de Numpy a la plataforma de Google Colab y se le asigna el alias "np". NumPy es una extensión al lenguaje de programación Python, que añade soporte para arreglos y matrices grandes y multidimensionales, junto con una gran biblioteca de funciones matemáticas de alto nivel para operar sobre estos arreglos. La principal razón para usar NumPy en el análisis de datos es que proporciona una interfaz conveniente para optimizar y simplificar las operaciones sobre los arreglos de datos (Oliphant, 2006).

El proceso de montaje de los datos consistió en cargar el conjunto de datos en la plataforma de Google Colab desde la ubicación de almacenamiento Google Drive.

Una vez cargados los datos del archivo XLSX en la plataforma, se procedió a crear un DataFrame de Pandas. Un DataFrame es una estructura de datos que organiza los datos en una tabla bidimensional de filas y columnas (Databricks, n.d.). Esta estructura es ampliamente utilizada para el análisis y manipulación de datos.

En Pandas, existen diferentes métodos para crear un DataFrame, pero en este caso, se utilizó un archivo en formato XLSX como fuente de datos. Cabe resaltar que la razón por la que los datos se cargan en un DataFrame es porque al hacerlo se obtiene la capacidad de filtrar, realizar cálculos, agregar o eliminar columnas, así como llevar a cabo otras operaciones de manipulación y análisis de datos.

Para crear el DataFrame se han utilizado las siguientes líneas de código:

Código para creación de DataFrame

```
!pip install openpyxl
path = "/content/drive/MyDrive/Colab_Notebooks/tfe.xlsx"
df = pd.read_excel(path, engine="openpyxl")
df.head()
```

FIGURA 4.2. Código que permite crear el DataFrame.

"!pip install openpyxl": esta línea se utiliza para instalar el paquete openpyxl utilizando el comando "pip". Openpyxl es un módulo de Python para trabajar con archivos Excel sin involucrar el software de aplicación MS Excel. Se utiliza ampliamente en diferentes

operaciones desde la copia de datos hasta la minería y el análisis de datos por parte de operadores de computadoras, analistas de datos y científicos de datos (Clark, 2017).

“**path = "/content/drive/MyDrive/Colab_Notebooks/tfe.xlsx"”:** en esta línea se define una variable llamada “path” que contiene la ubicación del archivo XLSX.

“df = pd.read_excel(path, engine="openpyxl")”: esta línea utiliza la función “read_excel” de la biblioteca Pandas para cargar el archivo XLSX en un DataFrame “df”. El argumento “path” indica la ubicación del archivo, y el argumento “engine=”openpyxl”” se utiliza para especificar el motor de lectura de archivos XLSX, que en este caso es el motor “openpyxl”.

“df”: esta línea hace referencia al DataFrame que se ha creado y que almacena los datos cargados del archivo XLSX.

La siguiente línea de código, “`df.head()`”, sirve para mostrar las primeras cinco filas del DataFrame (“df”). En este caso, se muestra el “df” en formato CSV. Cada columna tiene asignado un título codificado para facilitar su referencia en análisis posteriores:

	a0	a1	a2	a3	a4	b1	b2	c1	c21	c22	...	d3	d4p	d4m	d41p	d41m	d42p	d42m	e	b1_class	e_class
0	FACULTAD DE CIENCIAS	Grado en Física	1	17	Mujer	1	ComerSinControl	Precio	No	Si	...	Con otros estudiantes	Vive	Vive	Secundaria	Secundaria	Trabajando	Trabajando	20.312500	Adherencia Débil	Peso normal
1	FACULTAD DE CIENCIAS ECONÓMICAS Y EMPRESARIALES	Doble grado en administración y dirección de e...	4	21	Mujer	3	ResCog	Conv	Si	Si	...	Con otros estudiantes	Vive	Vive	Secundaria	Formación Profesional	Trabajando	Trabajando	22.038567	Adherencia Débil	Peso normal
2	FACULTAD DE EDUCACIÓN	Grado en Psicología	1	18	Mujer	4	ComerSinControl	ApaSens	No	Si	...	Padres y hermanos	Vive	Vive	Secundaria	Secundaria	En paro	Trabajando	18.645344	Adherencia Débil	Peso normal
3	FACULTAD DE CIENCIAS	Grado en Física	1	18	Mujer	4	ComerSinControl	ApaSens	No	No	...	Con otros estudiantes	Vive	Vive	Primaria	Secundaria	Trabajando	Trabajando	20.371209	Adherencia Débil	Peso normal
4	FACULTAD DE EDUCACIÓN	Grado en Psicología	1	18	Mujer	4	ComerSinControl	ApaSens	Si	Si	...	Con padres	Vive	Vive	Formación Profesional	Formación Profesional	Jubilado	Trabajando	17.569651	Adherencia Débil	Bajo peso
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
281	FACULTAD DE CIENCIAS ECONÓMICAS Y EMPRESARIALES	Grado en Relaciones Laborales y Recursos Humanos	4	23	Mujer	14	ResCog	ApaSens	No	Si	...	Con otros estudiantes	Vive	Vive	Universidad	Universidad	Trabajando	En casa	19.312952	Adherencia Buena o Muy Buena	Peso normal
282	FACULTAD DE CIENCIAS ECONÓMICAS Y EMPRESARIALES	Grado en Relaciones Laborales y Recursos Humanos	4	24	Hombre	14	ResCog	Precio	Si	Si	...	En pareja	Vive	Vive	Secundaria	Secundaria	Trabajando	Trabajando	21.357796	Adherencia Buena o Muy Buena	Peso normal
283	FACULTAD DE CIENCIAS ECONÓMICAS Y EMPRESARIALES	Grado en Relaciones Laborales y Recursos Humanos	1	42	Mujer	14	ResCog	SalyCnatural	No	No	...	Con familiares	Vive	No vive	Secundaria	Primaria	En paro	En paro	24.515595	Adherencia Buena o Muy Buena	Peso normal
284	FACULTAD DE CIENCIAS ECONÓMICAS Y EMPRESARIALES	Grado en Relaciones Laborales y Recursos Humanos	4	24	Hombre	14	ResCog	Precio	Si	Si	...	En pareja	Vive	Vive	Secundaria	Secundaria	Trabajando	Trabajando	21.357796	Adherencia Buena o Muy Buena	Peso normal
285	FACULTAD DE CIENCIAS ECONÓMICAS Y EMPRESARIALES	Grado en Relaciones Laborales y Recursos Humanos	4	24	Hombre	14	ResCog	Precio	Si	Si	...	En pareja	Vive	Vive	Secundaria	Secundaria	Trabajando	Trabajando	21.357796	Adherencia Buena o Muy Buena	Peso normal

286 rows × 26 columns

FIGURA 4.3. El DataFrame que se obtiene en Google Colaboratory para visualizar los datos de forma tabulada.

A continuación, se listan abreviaturas que se utilizaron para designar cada característica durante todo el proceso.

- a0: Esc_Facul
- a1: Grado
- a2: Curso
- a3: Edad
- a4: Sexo
- b1: MDS_Puntaje
- b2: TFEQ-R18
- c1: FCQ-SQP
- e: IMC

- c21: ¿Has cursado alguna asignatura de Nutrición y/o temas de Alimentación?
- c22: ¿Lees las etiquetas de los alimentos para saber su composición?
- c23: ¿Qué elemento de la información nutricional es más importante para ti?
- c24: ¿Qué medio publicitario es más relevante para la elección de tus alimentos?
- c25: ¿En qué medio viste la última vez publicidad sobre alimentos?
- c26: ¿A qué fuente habitualmente acudes para consultar sobre alimentación o dietas?
- d1: ¿Cómo te financias económicamente?
- d2: ¿Dónde vives en el curso académico actual?
- d3: ¿Con quién vives actualmente?
- d4p: Padre_vive
- d4m: Madre_vive
- d41p: Padre_Educación
- d41m: Madre_Educación
- d42p: Padre_Actividad
- d42m: Madre_Actividad

Luego se realizó una descripción inicial del DataFrame. En esa descripción se buscaba comprender la dimensión del "df" (número de filas y columnas) y los tipos de datos que contenía el "df", así como sus variables categóricas y numéricas.

Código para obtener la dimensión del DataFrame

```
df.shape
```

FIGURA 4.4. Código para obtener la dimensión del DataFrame

En esta línea de código, **"df.shape"** devuelve una tupla que representa la dimensionalidad del DataFrame. Esta propiedad es útil para conocer el tamaño del conjunto de datos. Esto se hace para evitar la "maldición de la dimensionalidad", que se refiere a la proporción entre el número de variables y el número de observaciones. Si el número de variables es mayor o cercano al número de observaciones, se puede decir que el conjunto de datos tiene una alta dimensionalidad, lo que puede dificultar la estimación de los parámetros y la generalización de los resultados (Cross Validated, 2015). Como indica Zhang y Wang (2021), si el número de variables de entrada fuera muy alto, se requerirían técnicas de reducción o selección de variables. En este caso, el conjunto de observaciones es de 286, mientras que las variables son 26, por lo que la "maldición de la dimensionalidad" no es un problema.

4.4.2. Limpieza y procesamiento de datos

Se realizó una exhaustiva revisión de los datos en busca de valores atípicos, duplicados y anómalos, se detectaron algunos valores anómalos que fueron corregidos porque se determinó que eran errores de entrada en la columna "a1" (Titulación) del DataFrame.

En estas líneas código se reemplazan errores de entrada por su correcto nombre, de lo contrario se contarían variables iguales como dos variables diferentes.

Código para corregir errores de entrada

```
df["a1"].value_counts()
mapping = {
    'Grado en Física ': 'Grado en Física',
    'Doble grado en administración y dirección de empresas / economía': 'Doble
grado en administración y dirección de empresas / economía',
    'Grado en Relaciones Laborales y Recursos Humanos': 'Grado en Relaciones
Laborales y Recursos Humanos',
    'Grado en comunicación audiovisuall': 'Grado en Comunicación Audiovisual',
    'Grado en ciencia y tecnología de los alimentos': 'Grado en Ciencia y
Tecnología de los Alimentos',
    'Grado en Enfermería': 'Grado en Enfermería',
    'Doble grado en administración y dirección empresas / Derecho': 'Doble grado
en administración y dirección empresas / Derecho',
    'Grado en Ingeniería Electrónica y Automática': 'Grado en Ingeniería
Electrónica y Automática',
    'Grado en biología': 'Grado en Biología',
    'Grado en Ciencia y Tecnología de los Alimentos': 'Grado en Ciencia y
Tecnología de los Alimentos',
    'Grado en Ciencia y Tecnología de los alimentos': 'Grado en Ciencia y
Tecnología de los Alimentos',
    'Grado en Ingeniería de hortofruticultura y jardinería ': 'Grado en
Ingeniería de Hortofruticultura y Jardinería',
    'Grado en Ingenieria de las Industrias Agrarias y Alimentarias': 'Grado en
Ingeniería de las Industrias Agrarias y Alimentarias',
    'Grado en ingeniería electrónica y automática': 'Grado en Ingeniería
Electrónica y Automática',
    'Grado en Ingeniería electrónica y automática': 'Grado en Ingeniería
Electrónica y Automática',
    'Grado Comunicación audiovisual/ información y documentación': 'Grado
Comunicación Audiovisual / Información y Documentación',
    'Grado en Ciencias ambientales ': 'Grado en Ciencias Ambientales',
    'Grado en información y documentación': 'Grado en Información y
Documentación',
    "Grado Comunicación Audiovisual / Información y Documentación": "Doble grado
en comunicación audiovisual / información y documentación",
    "Doble grado en administración y dirección empresas / Derecho": "Doble grado
en administración y dirección empresas / derecho",
    'Grado en Psicología ': 'Grado en Psicología',
}
df['a1'] = df['a1'].replace(mapping)
df["a1"].nunique()
```

FIGURA 4.5. Código que permite corregir errores de entrada

Para detectar los valores únicos que tiene una columna del df se utilizó el método **“.value_counts()”**. Este método nos muestra la frecuencia de valores únicos en una serie de datos, o columna del df. Nos devuelve una serie de pandas que muestra los valores únicos y sus recuentos. A partir de ese recuento podemos determinas si hay valores que a pesar de referirse a lo mismo, se cuentan como dos valores diferentes debido a errores de entrada.

Por ejemplo: “Grado en biología” se cuenta como un valor diferente a “Grado en Biología”, o “Grado en Física ” y “Grado en Física”. Estos valores tienen que ser reemplazados por un valor único: “Grado en Biología” y “Grado en Física”.

La variable "**mapping"** se define para crear un diccionario. Un diccionario es una colección de elementos que almacenan elementos en pares (clave-valor: C-V) separados por comas y agrupados en llaves {}. Cada clave tiene asociado un valor. Además del diccionario (**"mapping"**), se utiliza la función "**replace()"**. Esta función toma dos argumentos: el primero presenta un valor que quiero reemplazar, el segundo un valor con el que deseo reemplazar el primero. El primer valor sería la C del diccionario; el segundo, la V. En el caso del ejemplo, "Grado en biología" (C), sería reemplazado por "Grado en Biología" (V).

4.4.2.1.Reducción de cardinalidad

Se define como el proceso de reducción de los valores posibles de un feature. La reducción de cardinalidad puede ser útil para: reducir el tamaño del dataset, mejorar el rendimiento y la interpretación de los modelos de aprendizaje automático (Han et al., 2011).

- c23: ¿Qué elemento de la información nutricional es más importante para ti?

Código para reducir la cardinalidad

```
df["c23"] = df["c23"].str.split(",").str[0]
```

FIGURA 4.6. Código para reducir la cardinalidad

Lo que se hizo fue seleccionar únicamente el primer elemento de cada entrada en la columna "c23". Por ejemplo, se pasó de "Proteína, Vitaminas y minerales" a "Proteína". Esto se hizo debido a que el primer elemento de cada entrada fue considerado como el más importante para el alumno. Respondiendo así a la pregunta inicial con información suficiente. Además, se redujo la cardinalidad del feature de 25 a 5, facilitando la visualización y el análisis posterior.

- c24: ¿Qué medio publicitario es más relevante para la elección de tus alimentos?

Código para reducir la cardinalidad

```
mapping = {
            'Internet': 'Audiovisual',
            'Ninguno': 'Ninguno',
            "Degustaciones": "Experiencial",
            "TV": "Audiovisual",
            "Folletos Publicitarios": "Impreso",
            "Revistas": "Impreso",
            "Vallas": "Impreso",
        }
```

```
df["c24"] = df["c24"].replace(mapping)
```

FIGURA 4.7. Código para reducir la cardinalidad

Se redujo la cardinalidad de 7 a 4.

- c25: ¿En qué medio viste la última vez publicidad sobre alimentos?

Código para reducir la cardinalidad

```
mapping = {
            'Internet': 'Audiovisual',
            'Ninguno': 'Ninguno',
            "Degustaciones": "Experiencial",
            "TV": "Audiovisual",
            "Folletos Publicitarios": "Impreso",
            "Revistas": "Impreso",
            "Vallas": "Impreso",
            "Radio": "Audiovisual"
        }
df["c25"] = df["c25"].replace(mapping)
```

FIGURA 4.8. Código para reducir la cardinalidad

Se redujo la cardinalidad de 8 a 4.

- c26: ¿A qué fuente habitualmente acudes para consultar sobre alimentación o dietas?

Código para reducir la cardinalidad

```
mapping =  {
            "Médico": "Profesional de la salud",
            "Familiar": "Personal",
            "Amigos": "Personal",
            "TV": "Medios",
            "Internet": "Medios",
            "Otro": "Otro",
```

```
            "Libros": "Medios"
        }
df["c26"] = df["c26"].replace(mapping)
```

FIGURA 4.9. Código para reducir la cardinalidad

Se redujo la cardinalidad de 7 a 4.

- d1: ¿Cómo te financias económicamente?

Código para reducir la cardinalidad

```
mapping = {
    "Padres": "Padres",
    "Padre, Madre": "Padres",
    "Padre": "Padres",
    "Padres, Madre": "Padres",
    "Madre": "Padres",
    "Padres, Trabajo, Beca": "Trabajo",
    "Trabajo": "Trabajo",
    "Padres, Trabajo": "Trabajo",
    "Beca": "Beca",
    "Madre, Beca": "Beca",
    "Padres, Beca": "Beca",
    "Padre, Beca": "Beca",}
df["d1"] = df["d1"].replace(mapping)
```

FIGURA 4.10. Código para reducir la cardinalidad

Se redujo la cardinalidad de 12 a 3.

- d2: ¿Dónde vives en el curso académico actual?

Código para reducir la cardinalidad

```
mapping = {
          "Mi casa": "Vivienda propia",
          "Mi domicilio": "Vivienda propia",
          "Mi domicilio ": "Vivienda propia",
          "Domicilio paterno": "Domicilio familiar",
          "Domicilio de un familiar": "Domicilio familiar",
      }
df["d2"] = df["d2"].replace(mapping)
```

FIGURA 4.11. Código para reducir la cardinalidad

Se redujo la cardinalidad de 6 a 4.

- d3: ¿Con quién vives actualmente?

Código para reducir la cardinalidad

```
mapping = {
          "Padres y hermanos": "Familiares",
          "Con padres": "Familiares",
          "Con familiares": "Familiares",
          "Madre": "Familiares",
```

```
            "Solo": "Solo",
            "Con otros estuiantes": "Estudiantes",
            "En pareja": "Pareja"
        }
df["d3"] = df["d3"].replace(mapping)
```

FIGURA 4.12. Código para reducir cardinalidad

Se redujo la cardinalidad de 7 a 4.

- d42p: Padre_Actividad

Código para reducir la cardinalidad

```
mapping = {
          "En paro": "Desempleado",
          "En casa": "Desempleado",
}
df["d42p"] = df["d42p"].replace(mapping)
```

FIGURA 4.13. Código para reducir la cardinalidad

Se redujo la cardinalidad de 5 a 4.

- d42m: Madre_Actividad

Código para reducir la cardinalidad

```
mapping = {
    "En paro": "Desempleado",
    "En casa": "Desempleado",
}
df["d42m"] = df["d42m"].replace(mapping)
```

FIGURA 4.14. Codigo para reducir la cardinalidad

Se redujo la cardinalidad de a 5 a 4.

- a1: Grado

Código para reducir la cardinalidad

```
mapping = { 'Grado en Física ': 'Grado en Física',
    'Doble grado en administración y dirección de empresas / economía': 'Doble grado
en administración y dirección de empresas / economía',
    'Grado en Relaciones Laborales y Recursos Humanos': 'Grado en Relaciones
Laborales y Recursos Humanos','Grado en comunicación audiovisuall': 'Grado en
Comunicación Audiovisual','Grado en ciencia y tecnología de los alimentos': 'Grado
en Ciencia y Tecnología de los Alimentos','Grado en Enfermería': 'Grado en
Enfermería','Doble grado en administración y dirección empresas / Derecho': 'Doble
grado en administración y dirección empresas / Derecho',
    'Grado en Ingeniería Electrónica y Automática': 'Grado en Ingeniería Electrónica
y Automática','Grado en biología': 'Grado en Biología','Grado en Ciencia y
Tecnología de los Alimentos': 'Grado en Ciencia y Tecnología de los Alimentos',
    'Grado en Ciencia y Tecnología de los alimentos': 'Grado en Ciencia y Tecnología
de los Alimentos','Grado en Ingeniería de hortofruticultura y jardinería ': 'Grado
en Ingeniería de Hortofruticultura y Jardinería',
```

```
    'Grado en Ingenieria de las Industrias Agrarias y Alimentarias': 'Grado en
Ingeniería de las Industrias Agrarias y Alimentarias',
    'Grado en ingeniería electrónica y automática': 'Grado en Ingeniería Electrónica
y Automática','Grado en Ingeniería electrónica y automática': 'Grado en Ingeniería
Electrónica y Automática','Grado Comunicación audiovisual/ información y
documentación': 'Grado Comunicación Audiovisual / Información y Documentación',
    'Grado en Ciencias ambientales ': 'Grado en Ciencias Ambientales',
    'Grado en información y documentación': 'Grado en Información y Documentación',
    "Grado Comunicación Audiovisual / Información y Documentación": "Doble grado en
comunicación audiovisual / información y documentación",
    "Doble grado en administración y dirección empresas / Derecho": "Doble grado en
administración y dirección empresas / derecho",
    'Grado en Psicología ': 'Grado en Psicología',
} df['a1'] = df['a1'].replace(mapping)
```

FIGURA 4.15. Código para reducir la cardinalidad

Se redujo la cardinalidad de 31 a 23.

4.4.2.2. Codificación de variables categóricas

Se decidió no codificar las variables categóricas debido a que se usarán modelos con soporte nativo para ellas. Como señala Murphy (2012), "los modelos con soporte nativo para ellas suelen ser más eficientes y precisos que los modelos que requieren la codificación manual de características categóricas". La codificación de variables categóricas por métodos tradicionales (codificación de etiquetas, codificación one-hot, codificación de frecuencia, codificación promedio) pueden ocasionar pérdida de información. Además, usar los modelos con soporte nativo simplifica el flujo de trabajo, ya que el preprocesamiento es menos complejo (Zheng, A., & Casari, A., 2018).

4.4.3. Análisis exploratorio de datos

El análisis exploratorio de datos (EDA, por sus iniciales en inglés) es un paso importante en el análisis de datos. Implica revisar las características y los rasgos de los datos con una “mente abierta”; en otras palabras, sin intentar aplicar ningún modelo particular a los datos (Dodge, 2008). Tukey (1977) define el análisis exploratorio como un proceso que consiste en examinar un conjunto de datos con el fin de obtener una comprensión general de los datos. Al hacerlo, se busca identificar patrones y tendencias que puedan resultar valiosas para análisis posteriores. En resumen, el objetivo principal del EDA es comprender y extraer información útil de los datos sin imponer modelos preconcebidos.

4.4.3.1. Análisis univariado o análisis descriptivo

El análisis univariado involucra un conjunto de técnicas para estudiar una sola variable o característica. En el contexto del aprendizaje automático estas técnicas se utilizan para comprender la distribución, características y propiedades de las diferentes características, así como para detectar valores atípicos (Hastie et al., 2009; Murphy, K. P., 2012; James et al., 2013).

4.4.3.1.1. Tipos de datos y datos faltantes

4.4.3.1.1.1.Tipos de datos

Con esta línea de código, **"df.dtypes",** se obtuvieron los tipos de datos de las diferentes columnas del DataFrame.

Código para obtener los tipos de datos del DataFrame

```
df.dtypes
```

FIGURA 4.16. Código para obtener los tipos de datos en el DataFrame

Conocer el tipo de datos de cada columna del DataFrame ayuda a detectar y corregir posibles errores o inconsistencias en los datos. Como indica VanderPlas (2016), "Cuando se trabaja con un conjunto de datos grande, puede ser útil comprobar los tipos de datos de las columnas antes de hacer ningún análisis. Esto puede ayudar a evitar errores o resultados inesperados al aplicar operaciones o funciones matemáticas a los datos. Por ejemplo, si se intenta calcular la media de una columna que contiene cadenas de texto, se obtendrá un error. De forma similar, si se intenta graficar una columna que contiene valores booleanos, no se obtendrá una visualización significativa.". Los resultados de **"df.dtypes"** fueron los siguientes:

- a0: object
- a1: object
- a2: int64
- a3: int64
- a4: object
- b1: int64
- b2: object
- c1: object
- c21: object
- c22: object
- 1c23: object
- c24: object
- c25: object
- c26: object
- d1: object
- d2: object
- d3: object
- d4p: object
- d4m: object
- d41p: object
- d41m: object
- d42p: object
- d42m: object
- e: float64

Con estas líneas de código se extrajeron las variables categóricas y numéricas del DataFrame:

Código para extraer las características categóricas y numéricas

```
#COLUMNAS CATEGÓRICAS
categorical = df.dtypes == "object"
categorical_cols = df.select_dtypes(include=["object"]).columns.tolist()
print ("VARIABLES CATEGÓRICAS")
print ("Total:", categorical.sum())
print ("Columnas:", categorical_cols)
print()

#COLUMNAS NUMÉRICAS
numerical = sum(df[col].dtype in ["int64", "float64"] for col in df.columns)
print ("VARIABLES NUMÉRICAS")
numerical_cols = df.select_dtypes(include=["int64", "float64"]).columns.tolist()
print ("Total:", numerical)
print ("Columnas:", numerical_cols)
```

FIGURA 4.17. Código para extraer las características según tipo.

El análisis de las variables del estudio reveló un total de 20 variables categóricas y 4 variables numéricas.

En cuanto a las variables categóricas, estas se definen como aquellas que pueden clasificarse en categorías mutuamente exclusivas. En el estudio se identificaron las siguientes variables categóricas: 'a0', 'a1', 'a4', 'b2', 'c1', 'c21', 'c22', 'c23', 'c24', 'c25', 'c26', 'd1', 'd2', 'd3', 'd4p', 'd4m', 'd41p', 'd41m', 'd42p', 'd42m'.

En cuanto a las variables numéricas, estas se definen como aquellas que pueden expresarse en números. En este estudio se identificaron las siguientes variables numéricas: 'a2', 'a3', 'b1', 'e'.

Es importante hacer esto como paso previo para la creación de un modelo de aprendizaje automático (ML) porque las variables categóricas y numéricas tienen diferentes propiedades y requieren diferentes formas de codificación y tratamiento. Algunos modelos de ML solo pueden trabajar con variables numéricas, por lo que es necesario transformar las variables categóricas en valores numéricos mediante diferentes métodos de codificación (Kraus, 2021). Para facilitar el trabajo es recomendable tener identificados qué features son categóricos para codificarlos adecuadamente de tal forma que se puedan utilizan al entrenar los diferentes algoritmos de ML que se irán probando.

4.4.3.1.1.2. Datos faltantes

En esta etapa del estudio se llevó a cabo el proceso de limpieza y procesamiento de datos. El objetivo de esta etapa es garantizar la coherencia de los datos y prepararlos para su análisis posterior.

Se buscaron datos faltantes en cada columna del DataFrame. Para ello se utilizó la siguiente línea de código:

Código para extraer datos faltantes

```
total=df.isnull().sum().sort_values(ascending=False)
porcentaje=(df.isnull().sum() / df.isnull().count()).sort_values(ascending=False)
missing_data = pd.concat([total, porcentaje], axis=1, keys=["Total", "Porcentaje"])
```

FIGURA 4.18. Código para extraer datos faltantes

En estas líneas de código **“total = df.isnull().sum().sort_values(ascending=False)”** se explica de la siguiente manera: se asigna a la variable “total” el cálculo del número total de valores nulos en cada columna y los ordena de mayor a menor. La función “isnull()” de pandas verifica cada celda del DataFrame df para ver si contiene un valor nulo “(NaN)” y devuelve un DataFrame booleano con “True” en las ubicaciones donde hay valores nulos y “False” donde no los hay. Luego, sum() se utiliza para contar la cantidad de True en cada columna, lo que cuenta la cantidad de valores nulos en cada columna y “sort_values(ascending=False)” ordena los resultados en orden descendente en función de la cantidad de valores nulos en cada columna. “Porcentaje = (df.isnull().sum() / df.isnull().count()).sort_values(ascending=False)” calcula el porcentaje de valores nulos en cada columna y los ordena de mayor a menor, df.isnull().count() cuenta el número total de filas en el df. Finalmente, “missing_data = pd.concat([total, porcentaje], axis=1, keys=["Total", "Porcentaje"])” combina los resultados de los valores nulos y los porcentajes en un nuevo DataFrame llamado “missing_data”, con etiquetas "Total" y "Porcentaje" para las columnas. El resultado fue que no se encontraron datos faltantes en ninguna columna por lo que no es necesario aplicar técnicas de imputación de datos faltantes.

4.4.3.1.2. Análisis de características numéricas: discretas o continuas

Se analizaron las variables de tipo numérico: a2 (Curso), a3 (Edad), b1 (MEDAS) y e (IMC) y estableció una diferencia entre nominales y ordinales debido a que dependiendo de su tipo recibirán uno u otro tratamiento.

Tanto para obtener los resultados del análisis estadístico como los del análisis gráfico se utilizaron las siguientes líneas de código:

Código para obtener resultados de análisis estadístico y análisis gráfico

```
# Seleccionar las columnas numéricas del DataFrame
columnas_numericas = df[["Curso", "Edad", "MEDAS", "IMC"]]
# Crear subtramas para mostrar estadísticas e histogramas
num_columnas = len(columnas_numericas)
# Crear una única figura para los histogramas
fig, ejes = plt.subplots(2, 2, figsize=(10, 8))
# Definir la transparencia para el color de fondo y las líneas de la cuadrícula
transparencia_fondo = 0.7  # Ajustar el valor alfa para el color de fondo
transparencia_cuadricula = 0.3  # Ajustar el valor alfa para las líneas de la
cuadrícula
# Iterar a través de las características numéricas
       for i, columna in enumerate(columnas_numericas):
           # Estadísticas de cuantiles
           cuantiles = df[columna].quantile([0.25, 0.5, 0.75])
           valor_min = df[columna].min()
           valor_max = df[columna].max()
           rango_intercuartil = cuantiles[0.75] - cuantiles[0.25]
           # Estadísticas descriptivas
           conteo_unico = df[columna].nunique()
           conteo = df[columna].count()
           media = df[columna].mean()
           moda = df[columna].mode().values[0]
           desviacion_estandar = df[columna].std()
           mediana = df[columna].median()
           mad = df[columna].mad()
           curtosis = df[columna].kurtosis()
           asimetria = df[columna].skew()
           # Calcular el límite inferior y superior para valores atípicos
           limite_inferior = cuantiles[0.25] - 1.5 * rango_intercuartil
           limite_superior = cuantiles[0.75] + 1.5 * rango_intercuartil
           # Calcular la cantidad de valores por debajo y por encima de los
límites
           cantidad_por_debajo = df[columna][df[columna] <
limite_inferior].count()
           cantidad_por_encima = df[columna][df[columna] >
limite_superior].count()
           # Histograma de la distribución
           fila = i // 2
           columna = i % 2
           ejes[fila, columna].set_xlabel(columna)
           ejes[fila, columna].set_ylabel(' ')
           # Establecer el color de fondo para la subtrama con mayor transparencia
           ejes[fila, columna].set_facecolor('#F0F0F0AA')
           sns.histplot(df[columna], ax=ejes[fila, columna], bins='auto',
kde=True, edgecolor='none', linewidth=0)
           sns.set(color_codes=True)
           # Ocultar el borde cuadrado alrededor de toda la subtrama
           for borde in ['top', 'right', 'bottom', 'left']:
```

```
                ejes[fila, columna].spines[borde].set_visible(False)
            # Mostrar las estadísticas en formato de texto
            print(f"Columna: {columna}")
            print(f"Cuantiles (25%, 50%, 75%): {cuantiles[0.25]}, {cuantiles[0.5]},
{cuantiles[0.75]}")
            print(f"Mínimo: {valor_min}, Máximo: {valor_max}")
            print(f"Rango Intercuartil (IQR): {rango_intercuartil}")
            print(f"Conteo de Valores Únicos: {conteo_unico}")
            print(f"Conteo: {conteo}")
            print(f"Media: {media}")
            print(f"Moda: {moda}")
            print(f"Desviación Estándar: {desviacion_estandar}")
            print(f"Mediana: {mediana}")
            print(f"MAD (Desviación Mediana Absoluta): {mad}")
            print(f"Curtosis: {curtosis}")
            print(f"Asimetría: {asimetria}")
            print(f"Límite Inferior para Valores Atípicos: {limite_inferior}")
            print(f"Límite Superior para Valores Atípicos: {limite_superior}")
            print(f"Cantidad de Valores por Debajo del Límite Inferior:
{cantidad_por_debajo}")
            print(f"Cantidad de Valores por Encima del Límite Superior:
{cantidad_por_encima}")
            print("\n")
# Ajustar el diseño de la figura
plt.tight_layout()
# Guardar la figura como una imagen PNG
plt.savefig('numerical.png', dpi=1000, bbox_inches='tight')
# Mostrar el gráfico
plt.show()
```

FIGURA 4.19. Código para obtener resultado de análisis estadístico y gráfico

4.4.3.1.2.1. Análisis estadístico

	25%	50%	75%	min	max	IQR	conteo único	media	moda	mediana
a2	1.0	3.0	4.0	1	6	3.0	6	2.84	4	3.0
a3	18.25	21.0	23.0	17	42	4.75	16	21.32	18	21.0
b1	6.0	8.0	9.0	1	14	3.0	13	7.80	8	8.0
e	19.70	21.97	24.30	14.69	35.99	4.59	134	22.26	22.03	21.97

	σ	DMA	curtosis	asimetría	**LIVA[a]**	**LSVA[b]**	**RVDLI[c]**	**RVELS[d]**
a2	1.45	1.33	-1.41	-0.08	-3.5	8.5	0	0
a3	3.62	2.50	8.64	2.28	11.125	30.125	0	8
b1	1.97	1.50	1.19	0.51	1.5	13.5	1	6
e	3.39	2.58	2.32	1.13	12.80	31.20	0	5

[a] Límite inferior para valores atípicos. [b] Límite superior para valores atípicos. [c] Recuento de valores por debajo del límite inferior. [d] Recuento de valores por encima de límite superior.

TABLA 4.1. Tabla en donde se muestran los análisis estadísticos más frecuentes para características numéricas.

4.4.3.1.2.2. Análisis gráfico

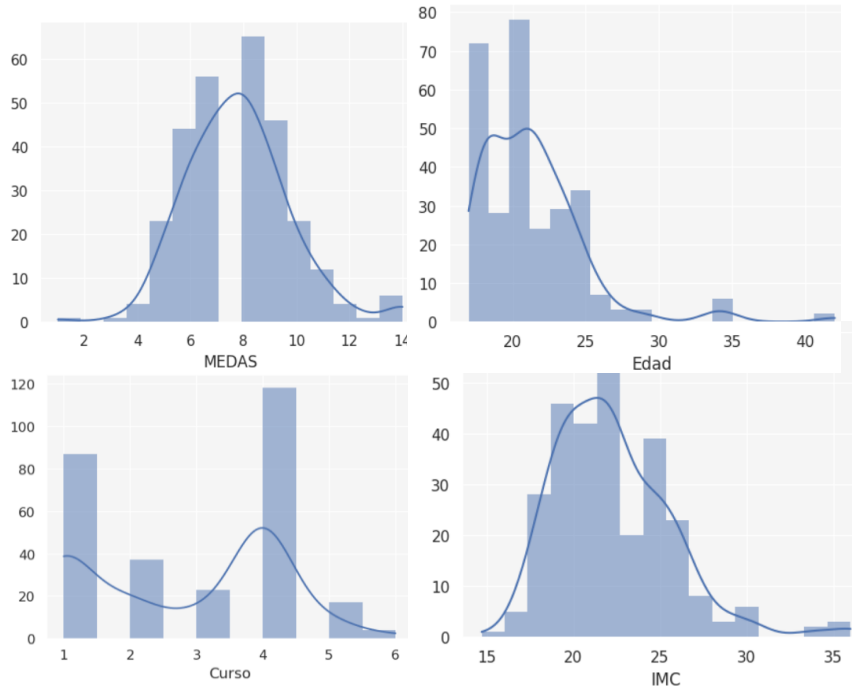

FIGURA 4.20. Gráfico de barras para observar la frecuencia de cada instancia.

4.4.3.1.2.3. Detección y tratamiento de "outliers"

La detección de outliers es una parte fundamental en la creación de un modelo de aprendizaje automático. Los outliers son datos que se desvían significativamente del resto y su inclusión en el modelo disminuye la precisión ya que el modelo se ajustará a esos valores inusuales por lo que tratar los datos es una tarea fundamental.

El objetivo principal de la detección de outliers es la mejora de la precisión y la robustez del modelo, dos métricas que evalúan el rendimiento del modelo (Jain, Murty & Flynn, 1999).

Existen diferentes métodos para la detección y el tratamiento de outliers. En primer lugar, para la detección, se utilizaron los siguientes métodos estadísticos: el rango intercuartílico, límites para valores atípicos (inferior y superior) y el conteo de los valores atípicos por debajo y por encima de los límites. En segundo lugar, para el tratamiento, se utilizó la técnica de truncamiento de outliers.

4.4.3.1.2.3.1. Outliers numéricos

El análisis de la variable a2 reveló que no hay valores atípicos fuera de los límites para valores atípicos.

En el caso de a3, el conteo de valores atípicos señaló 8 valores por encima del límite para valores atípicos. Para tratar estos outliers se utilizó una técnica de truncamiento llamada "límites de Tukey" (Figura 4.21). Se utilizó esta técnica debido a que se buscó conservar la mayor cantidad de información.

Código para tratamiento de outliers

```
# Calcula los límites de Tukey
lower_bound = 11.125
upper_bound = 30.125
# Aplica los límites de Tukey para eliminar los valores atípicos
df = df[(df["a3"] >= lower_bound) & (df["a3"] <= upper_bound)]
```

FIGURA 4.21. Código para tratamiento de outliers usando los límites de Tukey

El resultado del uso de esa técnica fue que los valores superiores a 30.125 fueron convertidos a 30.125, y los valores inferiores a 11.125 fueron convertidos en 11.125

En la característica b1 se encontraron 7 valores fuera de los límites para valores atípicos. Para tratar estos outliers se utilizó la técnica de truncamiento (Figura 4.22).

Código para tratamiento de outliers

```
# Calcula los límites de Tukey
lower_bound = 1.5
upper_bound = 13.5
# Aplica los límites de Tukey para eliminar los valores atípicos
df = df[(df["b1"] >= lower_bound) & (df["b1"] <= upper_bound)]
```

FIGURA 4.22. Códiho para tramiento de outliers usando los límites de Tukey.

El resultado del uso de esa técnica fue que los valores superiores a 13.5 fueron convertidos a 13.5, y los valores inferiores a 1.5 fueron convertidos en 1.5.

En la característica e se encontraron 5 valores por encima de los límites para valores atípicos. Para tratar estos outliers se utilizó la técnica de truncamiento (Figura 4.23).

Código para tratamiento de outliers

```
# Calcula los límites de Tukey
lower_bound = 12.806904236849626
upper_bound = 31.203246061724947
# Aplica los límites de Tukey para eliminar los valores atípicos
df = df[(df["e"] >= lower_bound) & (df["e"] <= upper_bound)]
```

FIGURA 4.23. Código para tratamiento de outliers usando los límites de Tukey.

4.4.3.1.3. Análisis de características categóricas

Se analizaron las características de tipo categórico: a0 (Facultad), a1 (Grado), a4 (Sexo), b2 (TFEQ-R18), c1 (FCQ-SQP), c21 (¿Has cursado alguna asignatura de Nutrición y/o temas de Alimentación?), c22 (¿Lees las etiquetas de los alimentos para saber su composición?), c23 (¿Qué elemento de la información nutricional es más importante para ti?), c24 (¿Qué medio publicitario es más relevante para la elección de tus alimentos?), c25 (¿En qué medio viste la última vez publicidad sobre alimentos?), c26 (¿A qué fuente habitualmente acudes para consultar sobre alimentación o dietas?), d1 (¿Cómo te financias económicamente?), d2 (¿Dónde vives en el curso académico actual?), d3 (¿Con quién vives actualmente?), d4p (¿Vive tu padre?), d4m (¿Vive tu madre?), d41p (¿Qué nivel educativo tiene tu padre?), d41m (¿Qué nivel educativo tiene tu padre?), d42p (¿Qué actividad laboral realiza tu padre?), d42m (¿Qué actividad laboral realiza tu madre?).

Las características categóricas nominales fueron a0, a1, a4, b2, c1, c21, c22, c23, c24, c25, c26, d1, d2, d3, d4p, d4m, d42p, d42m. Las características categóricas ordinales fueron d41p, d41m.

Se estableció una diferencia entre nominales y ordinales debido a que dependiendo de su tipo recibieron uno u otro tratamiento.

Para realizar el análisis estadístico de las variables categóricas se utilizó el análisis de cardinalidad, la frecuencia de cada categoría y el análisis de baja entropía. La cardinalidad se refiere al número de valores únicos y el conteo de valores únicos se refiere a el número de veces que cada valor único se repite.

4.4.3.1.3.1. Análisis estadístico

a0	Cardinalidad	Categorías	Frecuencia
	7	Facultad de ciencias económicas y empresariales	67
		Facultad de educación	62
		Facultad de ciencias	47
		Facultad de medicina	42
		Escuela de ingenierías industriales	28
		Escuela de Ingenierías agrarias	19
		Facultad de ciencias de la documentación y comunicación	16

a1	23		
		Grado en psicología	48
		Grado en medicina	33
		Grado en ingeniería electrónica y automática	26
		Grado en administración y dirección de empresas	22
		Grado en relaciones laborales y recursos humanos	20
		Grado en magisterio	14
		Grado en biotecnología	14
		Grado en ciencia y tecnología de los alimentos	13
		Grado en matemáticas	12
		Doble grado en administración y dirección de empresas / economía	12
		Grado en información y documentación	12
		Grado en física	11
		Doble grado en administración y dirección de empresas / derecho	10
		Grado en enfermería	9
		Grado en biología	7
		Grado en ciencias ambientales	3
		Grado en ingeniería hortofrutícola y jardinería	3
		Grado en ingeniería de las industrias agrarias y alimentarias	3
		Grado en economía	3
		Grado en ingeniería mecánica	2
		Grado en comunicación audiovisual	2
		Grado en comunicación audiovisual / información y documentación	1
		Grado en enología	1
a4	2		
		Hombre	128
		Mujer	153
c21	2		
		No	178
		Sí	103
b2	3		
		ResCog	129
		ComerSinControl	83
		ComerEmocional	69
c1	6		
		ApaSens	117
		SalyCnatural	70
		Precio	55
		Conv	22
		Peso	12
		Famil	5

c21	2		
		No	178
		Sí	103
c22	2		
		No	92
		Sí	189
c23	5		
		Calorías	124
		Grasa	83
		Proteína	62
		Carbohidratos	8
		Vitaminas y minerales	4
c24	7		
		Ninguno	101
		TV	76
		Internet	69
		Degustaciones	22
		Folletos publicitarios	7
		Revistas	4
		Vallas	2
c25	4		
		Audiovisual	236
		Impreso	35
		Ninguno	8
		Experiencial	2
c26	4		
		Medios	198
		Personal	47
		Profesional de la salud	22
		Otros	14
d1	3		
		Padres	198
		Beca	47
		Trabajo	22
d2	4		
		Piso de alquiler	147
		Domicilio familiar	115
		Residencia universitaria	16
		Vivienda propia	3
d3	4		
		Con otros estudiantes	143

		Familiares	115
		Solo	12
		Pareja	11
d41p	5		
		Secundaria	278
		Formación profesional	56
		Universidad	56
		Primaria	50
		Sin estudios	16
d41m	5		
		Secundaria	116
		Formación profesional	66
		Universidad	54
		Primaria	30
		Sin estudios	15
d42p	4		
		Trabajando	230
		Desempleado	24
		Jubilado	18
		Invalidez laboral	9
d42m	4		
		Trabajando	177
		Desempleado	89
		Jubilado	8
		Invalidez laboral	7
d4m	2		
		Vive	278
		No vive	3
d4p	2		
		Vive	268
		No vive	13

TABLA 4.2. Tabla en la que se muestra la cardinalidad y la frecuencia de cada variable categórica.

4.4.3.1.3.2.Análisis gráfico

Los histogramas son la representación gráfica de la distribución de un conjnnto de datos. En este caso, los histogramas muestran la frecuencia para cada categoría por cada variabla categórica del dataset original. La interpretación de esta disposición gráfica de información se hace de la siguiente manera: el eje horizontal representa las categorías de cada variable; el eje vertical o la altura de las barras representa la frecuencia por cada categoría.

Este tipo de gráficas nos ayuda a detectar diferentes patrones de distribución en los datos. Los patrones de distribución más frecuentes son: distribución uniforme, distribución sesgada y distribicuón bimodal. Una distribución uniforme significa que todas las categorías tienen la misma frecuencia, una distribibución sesgada significa que una categoría tiene una frecuencia mucho mayor que las demás. Una distribución bimodal significa que dos categorías tienen una frecuencia mayor que las demás.

En el contexto de este trabajo, lo que se busca analizar con estas gráficas es encontrar características que estén sesgadas ya que esto puede afectar el rendimiento del modelo. Con sesgo nos referimos a que una de las categorías de una de las variables es muy superior en frecuencia en comparación a las otras categorías de esa variable.

Una observación importante es que las características d42p, d4p y d4m muestran una predominancia muy superior de una sola categórica. Esto puede indicar que existe un sesgo en el conjunto de datos, ya que la mayoría de las instancias caen en una categoría. Este sesgo puede afectar el rendimiento del modelo, ya que los modelos pueden aprender a sesgarse hacia esa categoría.

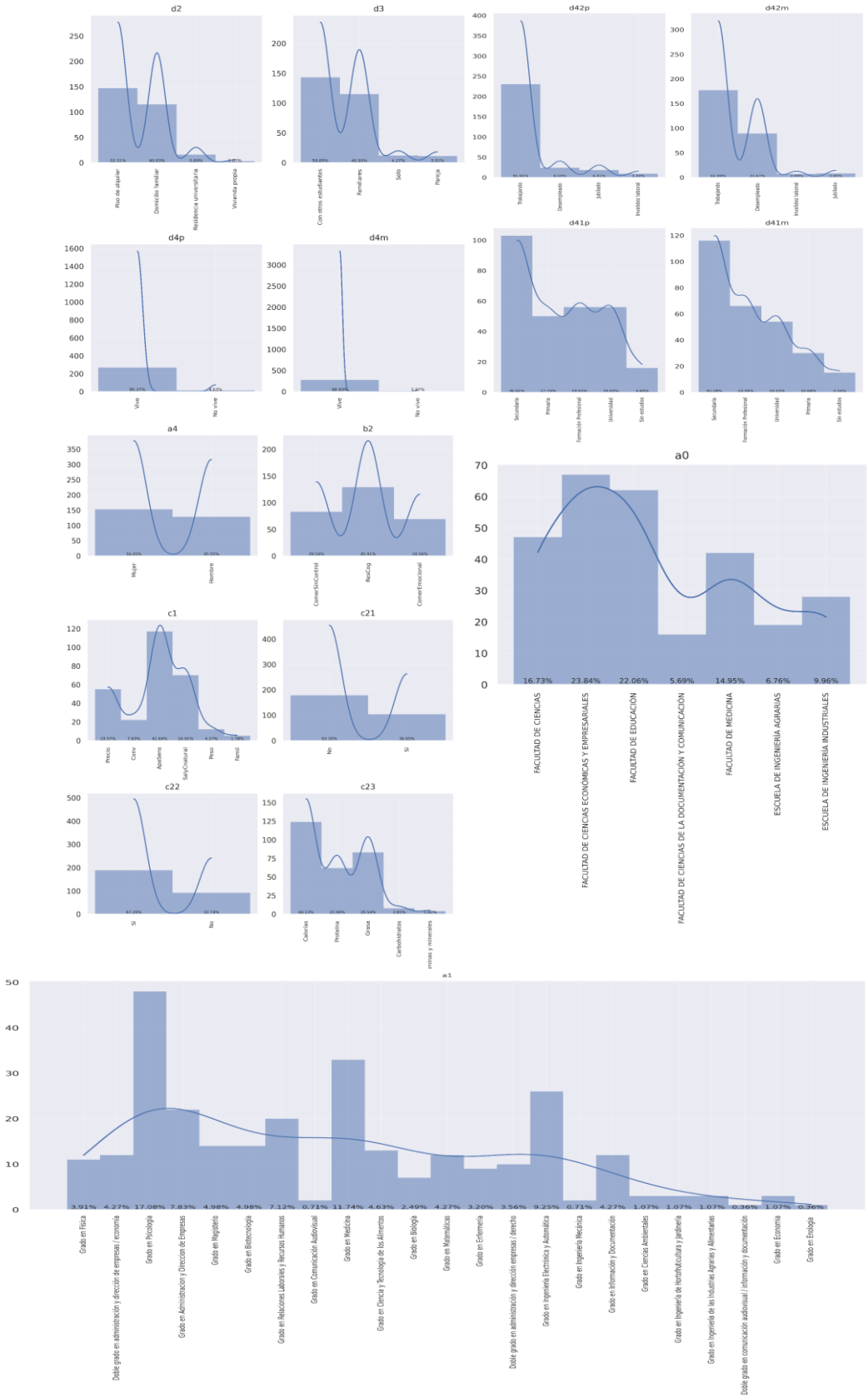

FIGURA 4.24. Histogramas que muestran la frecuencia por cada categoría de cada característica categórica.

4.4.3.1.3.3. Detección y tratamiento de características categóricas poco informativas

Se definió como característica categórica poco informativa aquella que cumplía con algunos de los siguientes requisitos:

(i) La característica tenía una baja entropía. La entropía es una medida de incertidumbre de una distribución de probabilidad. Una distribución con baja entropía es una distribución muy concentrada en un pequeño número de valores. Una baja entropía indica que la mayoría de los datos están en un pequeño número de categorías (Cover & Thomas, 2006; McKay & Conover, 1992).

(ii) La característica tiene una frecuencia superior al 70%. Este requisito indica que la característica tiene una frecuencia de aparición muy alta en comparación con otras categorías de la misma característica (Blei, Ng, & Jordan, 2003; Cui & Grossman, 2006).

En el contexto del aprendizaje automático, las características son variables que se utilizan para entrenar el modelo. Las características deben aportar suficiente variabilidad y deben ser informativas para el modelo. La variabilidad se refiere a la dispersión de los datos alrededor de la media; en el caso de las variables categóricas, la variabilidad de calcula utilizando, entre otros métodos, la entropía. Una distribución con baja entropía es una distribución muy concentrada en un pequeño número de valores. Por otro lado, la información se refiere a la cantidad de conocimiento que proporciona una característica sobre los datos. McKay y Conover (1992, p. 395) definen la información como "la cantidad de incertidumbre que se elimina al conocer el valor de una variable". En general, las características categóricas poco informativas son aquellas que tienen una baja variabilidad. Esto quiere decir que la característica no proporciona mucha información sobre los datos, ya que la mayoría de los valores son los mismos.

Teniendo en cuenta alguno de los dos requisitos mencionados anteriormente, y habiendo revisado el análisis estadístico y gráfico, las características categóricas poco informativas detectadas fueron las siguientes: d4p y d4m. Estas características no contribuyen significativamente al rendimiento del modelo, por lo que fueron excluidas del entrenamiento del modelo sin pérdida significativa de información. En la Figura 4.25 se muestra el código utilizado ara realizar el análisis de bajo entropía.

Código para hacer el análisis de baja entropía

```
from scipy.stats import entropy
columnas_categoricas = ["a0", "a1", "a4", "b2", "c1", "c21", "c22", "c23", "c24",
"c25", "c26", "d1", "d2", "d3", "d4p", "d4m", "d41p", "d41m", "d42p", "d42m"]
# Establece tu umbral de entropía
umbral_entropia = 0.7
caracteristicas_entropia_baja = []
for caracteristica in columnas_categoricas:
# Calcula la distribución de probabilidad para cada categoría
conteo_valores = df[caracteristica].value_counts(normalize=True)
# Calcula la entropía
entropia = entropy(conteo_valores, base=2)
# Verifica si la entropía está por debajo del umbral
if entropia < umbral_entropia:
caracteristicas_entropia_baja.append(caracteristica)
print("Características con baja entropía:", caracteristicas_entropia_baja)
```

FIGURA 4.25. Código que sirva para hallar las características con baja entropía

4.4.3.2. Análisis bivariado o análisis de correlación

El análisis bivariado es un método estadístico que se utiliza para encontrar relación entre dos variables. En el contexto de este trabajo, el análisis bivariado se utiliza para comprender la relación entre sí de las características, y entre la variable objetivo y las características (Hastie et al., 2009; Murphy, K. P., 2012; James et al., 2013).

Se hizo un análisis de correlación estadística y un análisis de correlación gráfica. Estos análisis ayudaron a identificar relaciones entre las diferentes características y la variable objetivo, así como las relaciones entre sí de las características, para así descubrir las variables que están altamente correlacionadas con la variable objetivo, y aquellos que están correlacionadas entre sí, para evaluar su consideración en la creación del modelo.

4.4.3.2.1. Análisis de correlación entre características categóricas y target

4.4.3.2.1.1. Análisis estadístico

Las tablas de contingencia son una herramienta estadística que se utiliza para representar la relación entre dos o más variables categóricas. La interpretación de las tablas, el análisis cualitativo, se centra en detectar patrones de freuencia en las tablas. Por otro lado, el análisis estadísticio cuantitiativo se centra en la fuerza de la relación entre las variables. Las herramientas estadísticas que se utilizaron este análisis fueron el chi-cuadrado y el valor de p. El chi cuadrado es una herramienta que sirve para medir la distancia entre la distribución observada de los datos y la dsitribución esperada si no hubiera relación entre las variables. El valor p es una medida de probabilidad para obtener los resultados observados si no hubiera relación entre las variables. Un valor de p inferior a un nivel de significación predeterminado, en este caso 0.05, inidca que es probable que exista una relación entre las variables (Hernández et al., 2014; Sánches & Reyes-Lagunes, 2013; Creswell, 2014; Tabachnick & Fidell, 2013).

El valor chi-cuadrado y el valor de p son importantes en el proceso de creación de un modelo de aprendizaje automático (AA). Debido a que pueen ayudarnos a encontrar relaciones significativas entre las características y la variable objetivo. Si existe una relación significativa, esta información se puede utilizar para mejorar el rendimiento del modelo.

	Categorías	Adherencia Buena o Muy Buena	Adherencia Débil	Adherencia Moderada a Justa
a0				
	Facultad de ciencias económicas y empresariales	12	13	36
	Facultad de educación	7	10	45
	Facultad de ciencias	8	4	33
	Facultad de medicina	0	0	37
	Escuela de ingenierías industriales	9	0	19
	Escuela de Ingenierías agrarias	3	0	16
	Facultad de ciencias de la documentación y comunicación	1	1	14
a1				
	Grado en psicología	2	8	38
	Grado en medicina	0	0	28
	Grado en ingeniería electrónica y automática	9	0	17
	Grado en administración y dirección de empresas	3	7	12
	Grado en relaciones laborales y recursos humanos	4	2	8
	Grado en magisterio	5	2	7
	Grado en biotecnología	1	2	11
	Grado en ciencia y tecnología de los alimentos	3	0	10
	Grado en matemáticas	3	0	9
	Doble grado en administración y dirección de empresas / economía	3	4	5
	Grado en información y documentación	0	0	12
	Grado en física	2	2	6
	Doble grado en administración y dirección de empresas / derecho	2	0	8
	Grado en enfermería	0	0	9
	Grado en biología	2	0	5
	Grado en ciencias ambientales	0	0	2
	Grado en ingeniería hortofrutícola y jardinería	0	0	3
	Grado en ingeniería de las industrias agrarias y alimentarias	0	0	3
	Grado en economía	0	0	3
	Grado en ingeniería mecánica	0	0	2
	Grado en comunicación audiovisual	0	1	1
	Grado en comunicación audiovisual / información y documentación	0	0	1
	Grado en enología	1	0	0
a4				
	Hombre	33	0	86
	Mujer	7	28	114
c21				
	No			
	Sí			
b2				
	ResCog	13	14	90
	ComerSinControl	16	8	58
	ComerEmocional	11	6	52

c1				
	ApaSens	17	10	84
	SalyCnatural	9	5	53
	Precio	5	3	43
	Conv	6	3	13
	Peso	0	7	5
	Famil	3	0	2
c21				
	No	33	19	116
	Sí	7	9	84
c22				
	No	12	8	65
	Sí	28	20	135
c23				
	Calorías	22	13	80
	Grasa	8	10	65
	Proteína	7	5	46
	Carbohidratos	3	0	5
	Vitaminas y minerales	0	0	4
c24				
	Ninguno	10	8	74
	TV	16	5	52
	Internet	7	13	48
	Degustaciones	6	2	14
	Folletos publicitarios	1	0	6
	Revistas	0	0	4
	Vallas	0	0	2
d2				
	Piso de alquiler	21	10	110
	Domicilio familiar	19	17	72
	Residencia universitaria	0	1	15
	Vivienda propia	0	0	3
d3				
	Con otros estudiantes	21	10	109
	Familiares	17	11	80
	Solo	2	5	5
	Pareja	0	2	6
d42m				
	Trabajando	20	24	128
	Desempleado	9	4	68

	Jubilado	5	0	3
	Invalidez laboral	6	0	1
d41p				
	Secundaria	18	9	69
	Formación profesional	9	5	37
	Universidad	10	7	38
	Primaria	2	5	43
	Sin estudios	1	2	13
d41m				
	Secundaria	19	14	78
	Formación profesional	7	2	52
	Universidad	13	11	29
	Primaria	0	1	27
	Sin estudios	1	0	14

TABLA 4.3. Tablas de contigencia de todas las características categóricas en relación con la variable objetivo y sus diferentes categorías.

El siguiente análisis cuantitativo, Tabla 4.4, nos ayuda a determinar qué variables están más correlacionadas con la variable objetivo (b1). Se puede observar que las características con un valor de p inferior a 0.05 son a0, a1, a4, c1, c21, d3, d42m y d41m, estas, marcadas en negrita, serán consideradas en el modelo debido a su significancia estadística, mientra que las demás no se considerarán.

	Chi-Cuadrado	p-value
a0	37.5616	**0.000181063**
a1	82.3041	**0.000410333**
a4	46.0387	**1.00653e-10**
b2	3.09686	0.541748
c1	44.8338	**2.33008e-06**
c21	8.91141	**0.0116121**
c22	0.238993	0.887367
c23	9.50482	0.301513
c24	16.6577	0.162932
d2	10.5262	0.104169
d3	17.5007	**0.0076089**
d42m	49.4574	**6.03851e-09**
d41p	8.17733	0.416343
d41m	27.211	**0.000650036**

TABLA 4.4. Se muestran los valores de chi-cuadrado y los valores p, que son el resultado de la relación entre las diferentes características categóricas y la variable objetivo.

Otra herramienta estadística que se utilizó para detectar características relevantes fue la ganancia de información. La ganancia de información (GI) se define como la reducción de la entropía de la variable objetivo después de conocer el valor de la característica. La entropía es una medida de incertidumbre de una variable. Una entropía baja indica que la variable es predecible, mientras que una entropía alta indica que la variable es impredecible (Alpaydin, 2014).

En general se considera que una GI de 0.05 o superior indica que la característica es importante para predecir la variable objetivo. Una GI de 0.05 indica que la característica reduca la incertidumbre de la variable objetivo en un 5% (James et al., 2013; Hastie et al., 2009). En la Tabla X se señala en negrita aquellas que superan ese umbral.

	Ganancia de Información
a0	**0.130903**
a1	**0.253124**
a4	**0.155406**
b2	0.00836653
c1	**0.0831304**
c21	0.0261177
c22	0.000650622
c23	0.0286762
c24	0.0471474
d2	0.036179
d3	0.037405
d42m	**0.0962295**
d41p	0.0266477
d41m	**0.0887632**

TABLA 4.5. Se muestran los valores de Ganancia de Información, que son el resultado de la relación entre las

En base a los resultados de chi-cuadrado, valor de p y ganancia de información se cofirma que las característicasb2, c21, c22, c23, c24, d2, d41p no son significativas, por lo que no se consideraron en el entrenamiento del modelo.

El código que se utilizó para obtener los resultados de la tabla 1 y la tabla 2 se indican en las figuras 1 y 2, respectivamente.

Código para obtener chi-cuadrado y valor p

```
!pip install scipy tabulate

import pandas as pd
from scipy.stats import chi2_contingency
from tabulate import tabulate

# Lista de características categóricas
categorical_features = ["a0", "a1", "a4", "b2", "c1", "c21", "c22", "c23", "c24",
"d2", "d3", "d42m", "d41p", "d41m"]

# Variable objetivo categórica
target_variable = "b1_class"

# Crear una lista para almacenar los resultados de la prueba de chi-cuadrado
chi2_results = []

# Recorrer cada característica categórica y realizar la prueba de chi-cuadrado
for feature in categorical_features:
    contingency_table = pd.crosstab(df[feature], df[target_variable])
    chi2, p, _, _ = chi2_contingency(contingency_table)
    chi2_results.append([feature, chi2, p])

# Crear un DataFrame a partir de los resultados
results_df = pd.DataFrame(chi2_results, columns=["Característica", "Estadística
Chi2", "Valor p"])

# Imprimir los resultados tabulados
print(tabulate(results_df, headers='keys', tablefmt='fancy_grid'))
```

FIGURA 4.26. Código necesario para obtener los valores de chi-cuadrado y valor p.

Código para obtener ganancia de información

```
import pandas as pd
import numpy as np
from tabulate import tabulate

# Definir una función para calcular la entropía
def entropy(series):
    _, counts = np.unique(series, return_counts=True)
    probabilities = counts / len(series)
    return -np.sum(probabilities * np.log2(probabilities + 1e-10))  # Añadiendo
un pequeño término para evitar log(0)

# Crear una copia del DataFrame original para el análisis de Information Gain
df_coded_informationgain = df.copy()

# Lista de características categóricas
columnas_categoricas = ["a0", "a1", "a4", "b2", "c1", "c21", "c22", "c23", "c24",
"d2", "d3", "d42m", "d41p", "d41m"]

# Variable objetivo categórica
variable_objetivo = "b1_class"

# Crear una lista para almacenar los resultados de Information Gain
info_gain_results = []

# Iterar a través de cada característica categórica y calcular Information Gain
for caracteristica in columnas_categoricas:
    # Calcular la entropía de la variable objetivo
    entropy_target = entropy(df_coded_informationgain[variable_objetivo])

    # Calcular la entropía condicional de la variable objetivo dada la
característica
    entropy_conditional = 0
    for valor in df_coded_informationgain[caracteristica].unique():
        subset =
df_coded_informationgain[df_coded_informationgain[caracteristica] == valor]
```

```
        entropy_conditional += (len(subset) / len(df_coded_informationgain)) *
entropy(subset[variable_objetivo])

    # Calcular la ganancia de información
    info_gain = entropy_target - entropy_conditional
    info_gain_results.append([caracteristica, info_gain])

# Crear un DataFrame a partir de los resultados
results_df = pd.DataFrame(info_gain_results, columns=["Característica", "Ganancia
de Información"])

# Imprimir los resultados tabulados
print(tabulate(results_df, headers='keys', tablefmt='fancy_grid'))
```

FIGURA 4.27. Código necesario para obtener los valores de Ganancia de Información.

4.4.3.2.1.2.Análisis gráfico

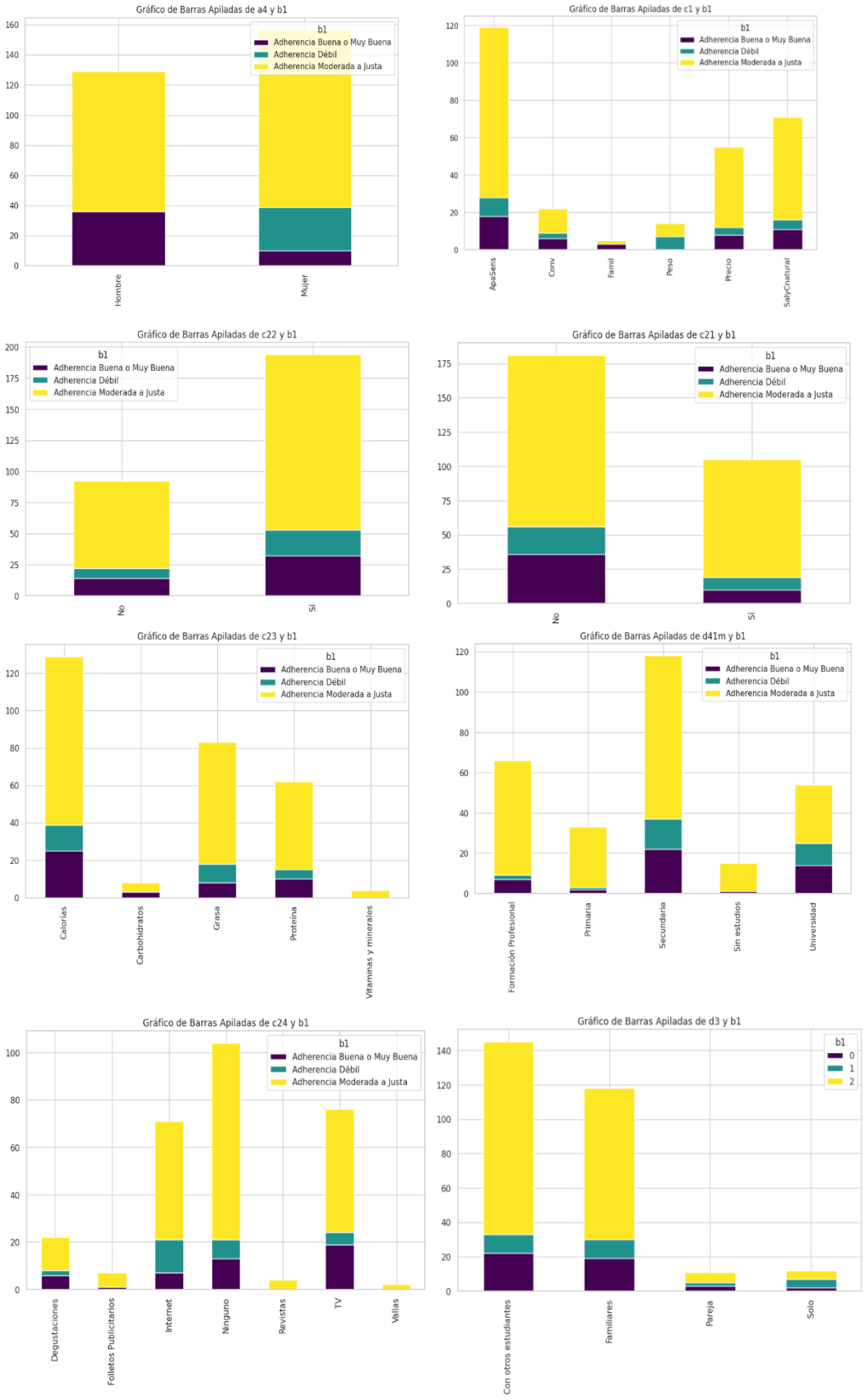

FIGURA 4.28. Se muestran gráficos de barras apiladas para analizar la relación entre las diferentes categorías de las variables categóricas y las de la variable objetivo.

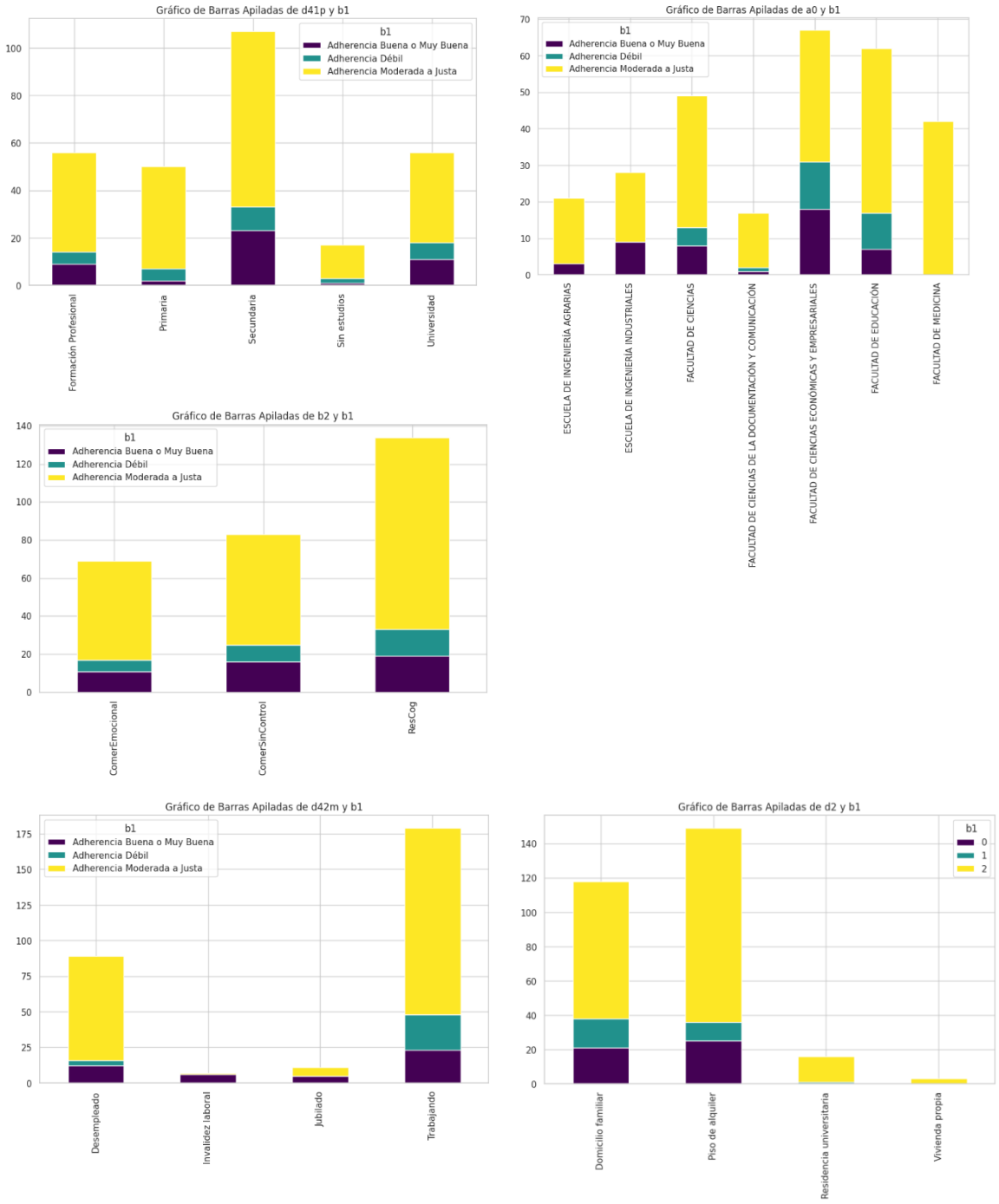

FIGURA 4.29. Se muestran gráficos de barras apiladas para analizar la relación entre las diferentes categorías de las variables categóricas y las de la variable objetivo.

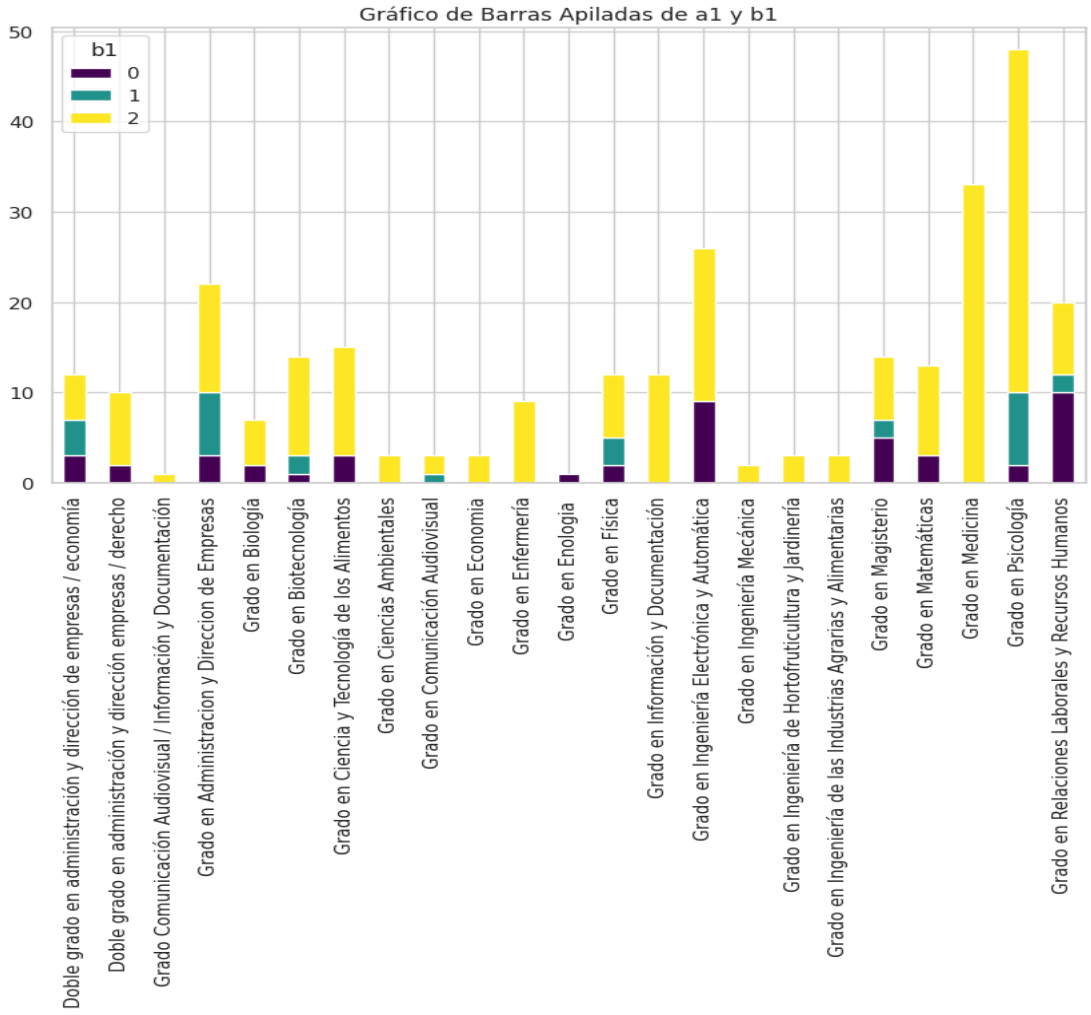

FIGURA 4.30. Se muestran gráficos de barras apiladas para analizar la relación entre las diferentes categorías de las variables categóricas y las de la variable objetivo.

El código utilizado para obtener las gráficas de barras apiladas se muestar en Figura 4.31.

Código para obtener gráfico de barras apiladas

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
# Lista de características categóricas
columnas_categoricas = ["a0", "a1", "a4", "b2", "c1", "c21", "c22", "c23", "c24",
"d2", "d3", "d42m", "d41p", "d41m"]
# Variable objetivo categórica
variable_objetivo = "b1_class"
# Configura el estilo para seaborn
sns.set(style="whitegrid")
# Recorre cada característica categórica y crea gráficos de barras apiladas
for caracteristica in columnas_categoricas:
    tabla_contingencia = pd.crosstab(df[caracteristica], df[variable_objetivo])
    tabla_contingencia.plot(kind="bar", stacked=True, figsize=(10, 6),
colormap="viridis")
    plt.title(f"Gráfico de Barras Apiladas de {caracteristica} y
{variable_objetivo}")
    plt.xlabel("")
    plt.ylabel("")  # Elimina la etiqueta "Count" en el eje y
    plt.xticks(rotation=0)  # Alinea los nombres de los valores en el eje x
    plt.legend(title=variable_objetivo)
    plt.show()
```

FIGURA 4.31. Código para obtener los gráficos de las barras apiladas

4.4.3.2.2. Análisis de correlación entre características numéricas y target

4.4.3.2.2.1.Análisis estadístico

a3	Test	Target_Class	T-statistic	P-value
	Student's t-test	Adherencia Débil	-1.61682	0.107102
	Student's t-test	Adherencia Moderada a Justa	-1.66846	0.0964006
	Student's t-test	Adherencia Buena o Muy Buena	3.48491	0.000575589
	ANOVA	All Classes	6.75221	0.00137983
	Logistic Regression	Accuracy	0.740741	

a3	Test	P-value	Correlation
	Pearson Correlation	0.007985958713209243	-0.1617250614305765
	Kendal Correlation	0.17766010960073575	-0.06968590819932229
	Spearman Correlation	0.1609864697596912	-0.08586935365894373

e	Test	Target_Class	T-statistic	P-value
	Student's t-test	Adherencia Débil	-0.709579	0.478587
	Student's t-test	Adherencia Moderada a Justa	-1.61859	0.10672
	Student's t-test	Adherencia Buena o Muy Buena	2.60814	0.00961806
	ANOVA	All Classes	3.45044	0.0331643
	Logistic Regression	Accuracy	0.740741	

e	Test	P-value	Correlation
	Pearson Correlation	0.026926738970149338	-0.13516569190789363
	Kendal Correlation	0.0352001250191897	-0.10272925881959664
	Spearman Correlation	0.03638694726216804	-0.12789688613901057

a2	Test	Target_Class	T-statistic	P-value
	Student's t-test	Adherencia Débil	-1.11451	0.266068
	Student's t-test	Adherencia Moderada a Justa	0.222115	0.824395
	Student's t-test	Adherencia Buena o Muy Buena	0.683848	0.494667
	ANOVA	All Classes	0.759904	0.468728
	Logistic Regression	Accuracy	0.740741	

a2	Test	P-value	Correlation
	Pearson Correlation	0.8416455890331312	-0.012260908196607849
	Kendal Correlation	0.9780972509913293	0.001501494167744641
	Spearman Correlation	0.9897805557070413	0.0007860793344093003

TABLA 4.6. En estas tablas se observan los resultados de distintas pruebas estadísticas que se utilizaron para calcular la correlación entre las características numéricas y la variable objetivo.

4.4.3.2.2.2.Análisis gráfico

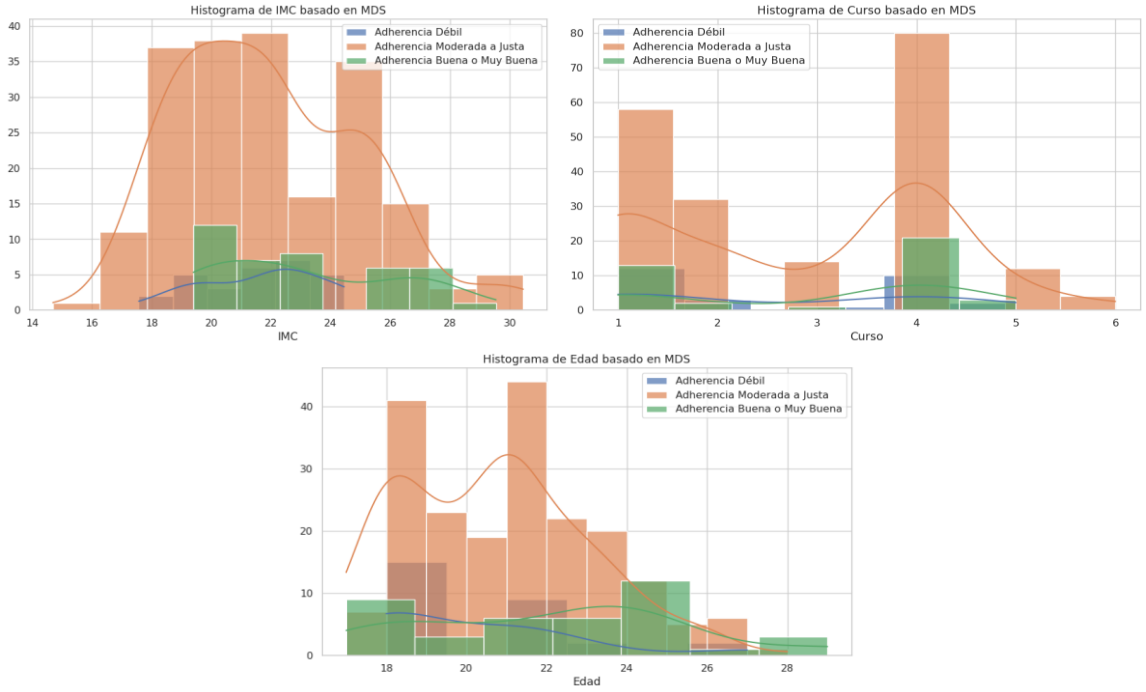

FIGURA 4.32. En estas figuras se muestran los histogramas de correlación entre las características numéricas y variable objetivo.

Basándonos en el análisis estadístico y gráfico, se quedaría fuera del entrenamiento del modelo la característica a2.

4.4.3.2.3. Análisis de correlación entre características numéricas y características numéricas

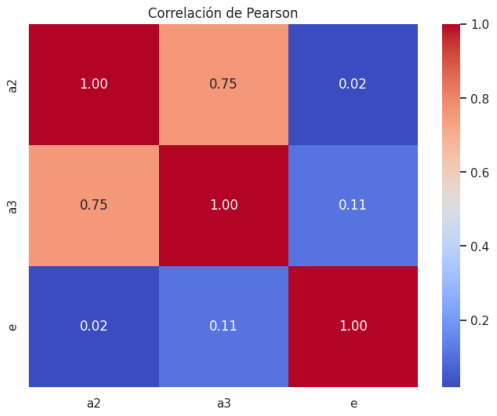

TABLA 4.7. Mapa de calor que muestra la correlación entre variables numéricas utilizando el coeficiente de correlación de Pearson.

En la tabla 4.7 se muestra una matriz de correlación que utiliza un mapa de calor para representar los coeficientes de correlación de Pearson. La correlación es una medida de fuerza

de relación entre dos o más variables. En este caso, se muestran los coeficientes de Pearson, que son una medida de la correlación lineal entre dos variables, y su valor puede variar entre -1 y +1. Un valor de 0 indica que no hay correlación entre las variables, un valor de +1 indica una correlación positiva perfecta entre las dos variables, un valor de -1 indica correlación negativa perfecta.

En el mapa de calor se observa que las características a2 y a3 están altamente correlacionadas, esto podría sugerir que la eliminación de una de ellas sería favorable para el modelo debido a que la información adicional que puede proveer una de las dos variables al modelo no supone un aporte significativo. En este caso, considerando los resultados de Test t-student, ANOVA y regresión logística, se descartó la característica a2 del entrenamiento del modelo.

4.4.3.2.4. Análisis de correlación entre características categóricas y características categóricas

Característica 1	Característica 2	Chi2	P-value
a0	a1	1588.538095	4.698169e-248
a0	a4	30.071239	3.810137e-05
a0	c1	72.714168	2.083113e-05
a0	d41m	70.874971	1.610235e-06
a0	d42m	49.261706	9.752963e-05
a1	a4	64.905689	4.083383e-06
a1	c1	200.050728	3.382299e-07
a1	d41m	245.795597	7.579009e-17
a1	d42m	103.372873	2.246457e-03
a4	c1	14.134184	1.477916e-02
a4	d41m	0.009820	4.026259e-02
a4	d42m	7.063865	6.988981e-02
c1	d41m	58.954614	1.032354e-05
c1	d42m	25.098158	4.864062e-02
d41m	d42m	58.946385	3.510283e-08

TABLA 4.8. Muestra la correlación entre variables categóricas utilizand chi-cuadrado y valor p.

4.5. Ingeniería de características

La ingeniería de características es el proceso de selección, construcción y transformación de características en un conjunto de datos para mejorar la precisión y la eficiencia de un modelo de aprendizaje automático (Hastie et al., 2009).

El proceso de selección de características puede dividirse en tres fases: selección de características, construcción de características y transformación de características.

En este trabajo sólo se ha considerado relevante el uso de la selección de características y transformación de características.

4.5.1. Selección de características

La selección de características es un proceso de preprocesamiento que consiste en identificar y seleccionar las características más relevantes para un modelo de aprendizaje automático. Como señala Guyon y Elisseeff (2003), “la selección de características es una técnica que puede mejorar el rendimiento de los modelos de aprendizaje automático”.

El objetivo principal de la selección de características es reducir la dimensión del dataset y descartar características no significativas, minimizando así la complejidad del modelo y mejorando su capacidad de generalización (Hall et al., 2009). Como señala Witten et al. (2016), "las características irrelevantes son aquellas que no aportan información significativa para la predicción de la variable objetivo". Las características redundantes son aquellas que "aportan la misma información que otras características".

Existen muchos métodos de selección de características para crear modelos predictivos. Los métodos de filtrado, como el Information Gain, se basan en la evaluación de las características de forma independiente. Los métodos envolventes, utilizan el rendimiento del modelo como criterio para la selección de características, entrenando y evaluando el modelo de forma iterativa con subconjuntos diferentes de características. Los métodos incorporados realizan la selección de características como parte del proceso de entrenamiento del modelo, durante este proceso algunas características reciben pesos nulos o cercanos a nulos, lo que indica su poca importancia. Otro método ampliamente utilizado es la selección secuencial, que consiste en la selección hacia adelante (añadiendo características) o hacia atrás (quitando características), basándose en su contribución al modelo.

4.5.1.1. Selección de características basada en análisis descriptivo

El análisis descriptivo ofrece estadística básica acerca de cada característica. En base a esa descripción se pueden identificar algunos problemas en las características.

Algunos de los problemas que se pueden encontrar son características con un alto porcentaje de valores faltantes, baja varianza de características numéricas, baja entropía de características categóricas, desbalance de la variable categórica objetivo, sesgo de distribución y alta cardinalidad. Los cambios o transformaciones asociadas a estos problemas fueron resueltos anteriomente a lo largo del punto IV.

4.5.1.2.Selección de características basada en análisis de correlación

El análisis de correlación examina la relación entre dos características. Las principales correcciones que se llevan a cabo tras un análisis de correlación son: baja correlación entre característica y variable objetivo, alta correlación entre características.

1. Baja correlación entre característica y variable objetivo: si la correlación es demasiado baja puede ser porque (a) la característica no es buena para predecir la variable objetivo, y por lo tanto puede ser eliminada del estudio, o (b) la característica no es útil en su formato actual y requiere ser transformada para revelar una relación más fuerte con la variable objetivo (Liu & Motoda, 2012; Guyon & Elisseeff, 2003).
2. Alta correlación entre características o multicolinealidad: este puede ser un problema debido a que si dos características están altamente correlacionadas entonces añadir las dos no proveen información adicional ni mejora el rendimiento del modelo. Además, algunos modelos lineales pueden volverse inestables. En el caso de que tratásemos con un dataset más grande esto sería de especial importancia debido al mayor uso de recursos tecnológicos durante el entrenamiento a cambio de ningún beneficio adicional. La solución en estos casos es eliminar características altamente correlacionadas, aplicar alguna técnica de reducción de dimensionalidad o usar modelos no lineales como redes neuronales, que suele más robusta en presencia de características correlacionadas (Google, 2018; Hastie et al., 2009; James et al., 2013).

4.5.2. Transformación de características

Las características se transforman para mejorar su utilidad en el aprendizaje automático. Esto se puede hacer normalizando, o codificando características.

La normalización de características implica transformar las características para que tengan una distribución similar. Si las características tienen distribuciones muy diferentes, el modelo puede tener dificultades para aprender patrones y generalizar nuevos datos.

Hay varios métodos diferentes para realizar la normalización de características. Un método común es la mediación, que consiste en restar la media de cada característica a cada valor de la característica. Otro método común es la estandarización, que consiste en restar la media de cada característica a cada valor de la característica y luego dividir por la desviación estándar de la característica (Hastie et al., 2009).

En este trabajo esos pasos se han llevado a cabo durante el entrenamietno del modelo.

4.6.Desarrollo del modelo de aprendizaje automático

Habiendo reducido el número de características, se procedió a usar esas características con diferentes modelos de aprendizaje automático para evaluar su capacidad predictiva con respecto a la variable objetivo.

Para elegir de forma correcta el modelo a usar se tiene que tener en cuenta el tipo de problema y el tipo de variables que se van a usar. En el caso de este trabajo, y tras haber hecho la selección de características, el dataset comprendería las siguientes características: a0, a1, a3, a4, c1, e, d41m, d42m. De las cuales, a2 es una característica numérica discreta, a3 es numérica discreta, e es discreta continua, a0 es categórica nominal, a1 es categórico nominal, a4 es categórico nominal, c1 es categórico nominal, d41m es categórico ordinal, d42m es categórico nominal. Por otro lado, la variable objetivo, b1, es ordinal categórica, con 3 clases (adherencia buena o muy buena, adherencia moderada a justa y adherencia débil) y desbalance de clase. El problema se podría tipificar como un problema de aprendizaje supervisado, de clasificación multiclase con una sola etiqueta.

En este tipo de problemas, los datos están “etiquetados” de acuerdo a un resultado específico de interés. El algoritmo aprende un mapeo a partir de un conjunto de características para obtener el resultado de interés (clasificar a la persona según el tipo de adherencia a la dieta). Esta parte se lleva a cabo durante la fase de entrenamiento del modelo. Cuando el modelo ha aprendido a mapear las características en base al resultado de interés, entonces este mapeo se puede aplicar a un nuevo conjunto de datos para hacer predicciones.

En base a las características del problema, se consideraron los siguientes modelos random forest (RF), eXtreme Gradient boosting (XGBoost), Category Boosting (CatBoost) y Redes Neuronales Artificiales (ANN) para resolver el problema.

La razón por la que se utilizaron estos modelos y no otros es debido a que a diferencia de otros modelos estos modelos se suelen utilizar para resolver problemas de clasificación muticlase, además de ser robustos al desbalance de clases. CatBoost es similar a XGBoost, modelos basados en árboles de decisiones, pero utiliza una técnica diferente para manejar variables categóricas, lo que podría mejorar la precisión del modelo. Las ANNs aunque pueden aprender relaciones complejas entre las variables cabe destacar que suelen tener un menor nivel de interpretabilidad.

Una vez seleccionados los algoritmos se procedió a dar lugar al entrenamiento de estos.

Código para modelo CatBoost

```
import pandas as pd
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.metrics import accuracy_score, balanced_accuracy_score,
precision_recall_fscore_support
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
from catboost import CatBoostClassifier
import optuna

# Iniciar el tiempo de ejecución
start_time = time.time()

# Cargar tu conjunto de datos (sustituye 'your_dataset.csv' con tu archivo real)
# Suponiendo que df es tu DataFrame
df = df

# Definir características y variable objetivo
features = ['a0', 'a1', 'a3', 'a4', 'c1', 'e', 'd41m', 'd42m']
target_variable = 'b1_class'

# Separar el conjunto de datos en conjuntos de entrenamiento, validación y prueba
X_train, X_test, y_train, y_test = train_test_split(df[features],
df[target_variable], test_size=0.2, random_state=42)
X_train, X_val, y_train, y_val = train_test_split(X_train, y_train,
test_size=0.2, random_state=42)

# Convertir columnas categóricas al tipo 'category'
categorical_features = ['a0', 'a1', 'a4', 'c1', 'd41m', 'd42m']
X_train[categorical_features] = X_train[categorical_features].astype('category')
X_val[categorical_features] = X_val[categorical_features].astype('category')
X_test[categorical_features] = X_test[categorical_features].astype('category')

# Convertir la variable objetivo a numérica usando LabelEncoder
label_encoder = LabelEncoder()
y_train_encoded = label_encoder.fit_transform(y_train)
y_val_encoded = label_encoder.transform(y_val)
y_test_encoded = label_encoder.transform(y_test)

# Definir la función objetivo para Optuna
def objective(trial):
    # Construir el modelo CatBoostClassifier
    model = CatBoostClassifier(
```

```
        iterations=10,  # Ajustar el número de iteraciones según sea necesario
        cat_features=categorical_features,
        learning_rate=trial.suggest_float('learning_rate', 0.001, 1.0, log=True),
        depth=trial.suggest_int('depth', 1, 10),
        random_seed=42
    )

    # Obtener puntuaciones de validación cruzada
    cv_scores = cross_val_score(model, X_train, y_train_encoded, cv=10,
scoring='accuracy', verbose=0)

    # Optimizar según la precisión media de validación cruzada
    accuracy = cv_scores.mean()

    return accuracy

# Crear un estudio de Optuna
study = optuna.create_study(direction='maximize')
study.optimize(objective, n_trials=10)

# Obtener los mejores hiperparámetros
best_params = study.best_params

# Construir el mejor modelo CatBoost utilizando los mejores hiperparámetros
best_catboost_model = CatBoostClassifier(
    iterations=10,  # Ajustar el número de iteraciones según sea necesario
    cat_features=categorical_features,
    learning_rate=best_params['learning_rate'],
    depth=best_params['depth'],
    random_seed=42
)

# Ajustar el mejor modelo en todo el conjunto de entrenamiento
best_catboost_model.fit(X_train, y_train_encoded, eval_set=(X_val,
y_val_encoded), verbose=0)

# Evaluar el mejor modelo en el conjunto de prueba
y_test_pred_proba = best_catboost_model.predict_proba(X_test)
y_test_pred = y_test_pred_proba.argmax(axis=1)

# Métricas de evaluación
accuracy = accuracy_score(y_test_encoded, y_test_pred)
balanced_accuracy = balanced_accuracy_score(y_test_encoded, y_test_pred)
precision, recall, f1, _ = precision_recall_fscore_support(y_test_encoded,
y_test_pred, average='weighted', zero_division=1)

# Mostrar métricas
metrics_table = pd.DataFrame({
    'Accuracy': [accuracy],
    'Balanced Accuracy': [balanced_accuracy],
    'Precision': [precision],
    'Recall': [recall],
    'F1 Score': [f1]
})

# Finalizar el tiempo de ejecución
end_time = time.time()
runtime = end_time - start_time
print("Tiempo de ejecución:", runtime)

print("Tabla de Métricas:")
print(metrics_table)
```

FIGURA 4.33. Código necesario para entrenar, validar y evaluar el modelo CatBoost.

Código para modelo Redes Neuronales Artificiales

```
import pandas as pd
from sklearn.model_selection import train_test_split, StratifiedKFold
from sklearn.metrics import accuracy_score, balanced_accuracy_score,
precision_recall_fscore_support
from sklearn.preprocessing import LabelEncoder, OneHotEncoder, MinMaxScaler
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, InputLayer
from tensorflow.keras.losses import SparseCategoricalCrossentropy
from tensorflow.keras.optimizers import Adam
import tensorflow as tf
import optuna
import matplotlib.pyplot as plt
import time

# Iniciar el tiempo de ejecución
start_time = time.time()

# Cargar el conjunto de datos (sustituir 'tu_dataset.csv' con tu archivo real)
df = df

# Definir características y variable objetivo
features = ['a0', 'a1', 'a3', 'a4', 'c1', 'e', 'd41m', 'd42m']
target_variable = 'b1_class'

# Separar el conjunto de datos en conjuntos de entrenamiento, validación y prueba
X_train, X_test, y_train, y_test = train_test_split(df[features],
df[target_variable], test_size=0.2, random_state=42)
X_train, X_val, y_train, y_val = train_test_split(X_train, y_train,
test_size=0.2, random_state=42)

# Convertir columnas categóricas a tipo 'category'
categorical_features = ['a0', 'a1', 'a4', 'c1', 'd41m', 'd42m']
X_train[categorical_features] = X_train[categorical_features].astype('category')
X_val[categorical_features] = X_val[categorical_features].astype('category')
X_test[categorical_features] = X_test[categorical_features].astype('category')

# Convertir la variable objetivo a numérica usando LabelEncoder
label_encoder = LabelEncoder()
y_train_encoded = label_encoder.fit_transform(y_train)
y_val_encoded = label_encoder.transform(y_val)
y_test_encoded = label_encoder.transform(y_test)

# Características numéricas - Normalización
numerical_features = ['a2', 'a3', 'e']
scaler = MinMaxScaler()

X_train[numerical_features] = scaler.fit_transform(X_train[numerical_features])
X_val[numerical_features] = scaler.transform(X_val[numerical_features])
X_test[numerical_features] = scaler.transform(X_test[numerical_features])

# Codificación one-hot de características categóricas
encoder = OneHotEncoder(sparse=False, handle_unknown='ignore')

# Ajustar y transformar en el conjunto de entrenamiento
X_train_encoded = encoder.fit_transform(X_train[categorical_features])

# Transformar en el conjunto de validación
X_val_encoded = encoder.transform(X_val[categorical_features])

# Transformar en el conjunto de prueba
X_test_encoded = encoder.transform(X_test[categorical_features])

# Definir la función objetivo para Optuna
def objective(trial):
    # Construir el modelo de Red Neuronal
    model = Sequential([
        InputLayer(input_shape=(X_train_encoded.shape[1],)),
```

```
        Dense(units=trial.suggest_int('units', 32, 512, step=32),
activation='relu'),
        Dense(64, activation='relu'),
        Dense(3, activation='softmax')  # Suponiendo 3 clases para tu tarea de
clasificación
    ])

    # Compilar el modelo

model.compile(optimizer=Adam(learning_rate=trial.suggest_float('learning_rate',
1e-5, 1e-1, log=True)),
                  loss=SparseCategoricalCrossentropy(),
                  metrics=['accuracy'])

    # Ajustar el modelo
    model.fit(X_train_encoded, y_train_encoded, epochs=10,
validation_data=(X_val_encoded, y_val_encoded), verbose=0)

    # Evaluar el modelo en el conjunto de validación
    _, accuracy = model.evaluate(X_val_encoded, y_val_encoded, verbose=0)

    return accuracy

# Crear un estudio de Optuna
study = optuna.create_study(direction='maximize')
study.optimize(objective, n_trials=10)

# Obtener los mejores hiperparámetros
best_params = study.best_params

# Construir el mejor modelo utilizando los mejores hiperparámetros
best_model = Sequential([
    InputLayer(input_shape=(X_train_encoded.shape[1],)),
    Dense(units=best_params['units'], activation='relu'),
    Dense(64, activation='relu'),
    Dense(3, activation='softmax')  # Suponiendo 3 clases para tu tarea de
clasificación
])

# Compilar el mejor modelo
best_model.compile(optimizer=Adam(learning_rate=best_params['learning_rate']),
                   loss=SparseCategoricalCrossentropy(),
                   metrics=['accuracy'])

# Ajustar el mejor modelo en todo el conjunto de entrenamiento
best_model.fit(X_train_encoded, y_train_encoded, epochs=10, verbose=1)

# Evaluar el mejor modelo en el conjunto de prueba
y_pred_proba = best_model.predict(X_test_encoded)
y_pred = tf.argmax(y_pred_proba, axis=1).numpy()

# Métricas de evaluación
accuracy = accuracy_score(y_test_encoded, y_pred)
balanced_accuracy = balanced_accuracy_score(y_test_encoded, y_pred)
precision, recall, f1, _ = precision_recall_fscore_support(y_test_encoded,
y_pred, average='weighted')

# Mostrar métricas
metrics_table = pd.DataFrame({
    'Accuracy': [accuracy],
    'Balanced Accuracy': [balanced_accuracy],
    'Precision': [precision],
    'Recall': [recall],
    'F1 Score': [f1]
})

print("Tabla de Métricas:")
print(metrics_table)
```

```
# Usar StratifiedKFold para validación cruzada
kf = StratifiedKFold(n_splits=10, shuffle=True, random_state=42)

# Listas para almacenar resultados de la validación cruzada
cv_accuracy = []
cv_balanced_accuracy = []
cv_precision = []
cv_recall = []
cv_f1 = []

for train_index, val_index in kf.split(X_train_encoded, y_train_encoded):
    X_train_fold, X_val_fold = X_train_encoded[train_index],
X_train_encoded[val_index]
    y_train_fold, y_val_fold = y_train_encoded[train_index],
y_train_encoded[val_index]

    # Construir el modelo de Red Neuronal
    model = Sequential([
        InputLayer(input_shape=(X_train_encoded.shape[1],)),
        Dense(units=study.best_params['units'], activation='relu'),
        Dense(64, activation='relu'),
        Dense(3, activation='softmax')
    ])

    # Compilar el modelo

model.compile(optimizer=Adam(learning_rate=study.best_params['learning_rate']),
                  loss=SparseCategoricalCrossentropy(),
                  metrics=['accuracy'])

    # Ajustar el modelo
    model.fit(X_train_fold, y_train_fold, epochs=10, verbose=0)

    # Evaluar el modelo en el conjunto de validación
    y_pred_fold_proba = model.predict(X_val_fold)
    y_pred_fold = tf.argmax(y_pred_fold_proba, axis=1).numpy()

    # Calcular métricas para cada fold
    accuracy_fold = accuracy_score(y_val_fold, y_pred_fold)
    balanced_accuracy_fold = balanced_accuracy_score(y_val_fold, y_pred_fold)
    precision_fold, recall_fold, f1_fold, _ =
precision_recall_fscore_support(y_val_fold, y_pred_fold, average='weighted')

    cv_accuracy.append(accuracy_fold)
    cv_balanced_accuracy.append(balanced_accuracy_fold)
    cv_precision.append(precision_fold)
    cv_recall.append(recall_fold)
    cv_f1.append(f1_fold)

# Calcular métricas promedio entre los folds
avg_accuracy = sum(cv_accuracy) / len(cv_accuracy)
avg_balanced_accuracy = sum(cv_balanced_accuracy) / len(cv_balanced_accuracy)
avg_precision = sum(cv_precision) / len(cv_precision)
avg_recall = sum(cv_recall) / len(cv_recall)
avg_f1 = sum(cv_f1) / len(cv_f1)

# Mostrar métricas de validación cruzada
print("Resultados de Validación Cruzada:")
print(f"Promedio de Accuracy: {avg_accuracy}")
print(f"Promedio de Balanced Accuracy: {avg_balanced_accuracy}")
print(f"Promedio de Precision: {avg_precision}")
print(f"Promedio de Recall: {avg_recall}")
print(f"Promedio de F1 Score: {avg_f1}")

# Construir y ajustar el modelo final en todo el conjunto de entrenamiento
final_model = Sequential([
    InputLayer(input_shape=(X_train_encoded.shape[1],)),
```

```
    Dense(units=study.best_params['units'], activation='relu'),
    Dense(64, activation='relu'),
    Dense(3, activation='softmax')
])

final_model.compile(optimizer=Adam(learning_rate=study.best_params['learning_rate
']),
                    loss=SparseCategoricalCrossentropy(),
                    metrics=['accuracy'])

final_model.fit(X_train_encoded, y_train_encoded, epochs=10, verbose=1)

# Evaluar el modelo final en el conjunto de prueba
y_pred_proba = final_model.predict(X_test_encoded)
y_pred = tf.argmax(y_pred_proba, axis=1).numpy()

# Métricas de evaluación
accuracy = accuracy_score(y_test_encoded, y_pred)
balanced_accuracy = balanced_accuracy_score(y_test_encoded, y_pred)
precision, recall, f1, _ = precision_recall_fscore_support(y_test_encoded,
y_pred, average='weighted')

# Finalizar el tiempo de ejecución
end_time = time.time()
tiempo_ejecucion = end_time - start_time
print("Tiempo de Ejecución:", tiempo_ejecucion)

# Mostrar métricas del conjunto de prueba
print("Resultados del Conjunto de Prueba:")
print(f"Accuracy: {accuracy}")
print(f"Balanced Accuracy: {balanced_accuracy}")
print(f"Precisión: {precision}")
print(f"Recall: {recall}")
print(f"F1 Score: {f1}")
```

FIGURA 4.34. Código necesario para entrenar, validar y evaluar el modelo de Redes Neuronales Artificiales.

Código para modelo XGBoost

```
from xgboost import XGBClassifier
from sklearn.model_selection import train_test_split, GridSearchCV,
cross_val_score
from sklearn.metrics import accuracy_score, balanced_accuracy_score,
precision_score, recall_score, f1_score
from sklearn.preprocessing import LabelEncoder
import pandas as pd

# Iniciar tiempo de ejecución
tiempo_inicio = time.time()

# Suponiendo que df es tu DataFrame con características categóricas
caracteristicas = ['a0', 'a1', 'a2', 'a3', 'a4', 'c1', 'e', 'd41m', 'd42m']
variable_objetivo = 'b1_class'

# Separar el conjunto de datos en conjuntos de entrenamiento, validación y prueba
train_df, test_df = train_test_split(df, test_size=0.2, random_state=42)
train_df, val_df = train_test_split(train_df, test_size=0.2, random_state=42)

X_train, y_train = train_df[caracteristicas].copy(),
train_df[variable_objetivo].copy()
X_val, y_val = val_df[caracteristicas].copy(), val_df[variable_objetivo].copy()
X_test, y_test = test_df[caracteristicas].copy(),
test_df[variable_objetivo].copy()

# Convertir columnas categóricas al tipo 'category'
categorias = ['a0', 'a1', 'a4', 'c1', 'd41m', 'd42m']
X_train[categorias] = X_train[categorias].astype('category')
```

```
X_val[categorias] = X_val[categorias].astype('category')
X_test[categorias] = X_test[categorias].astype('category')

# Convertir la variable objetivo a numérica usando LabelEncoder
label_encoder = LabelEncoder()
y_train_encoded = label_encoder.fit_transform(y_train)
y_val_encoded = label_encoder.transform(y_val)
y_test_encoded = label_encoder.transform(y_test)

# Crear clasificador XGBoost
modelo = XGBClassifier(enable_categorical=True)

# Definir una cuadrícula de parámetros para ajuste de hiperparámetros
param_grid = {
    'learning_rate': [0.01, 0.1, 0.8],
    'max_depth': [3, 14, 24],
    'n_estimators': [50, 100, 200]
}

# Inicializar GridSearchCV
busqueda_grid = GridSearchCV(estimator=modelo, param_grid=param_grid,
scoring='accuracy', cv=3)

# Ajustar el modelo con ajuste de hiperparámetros
busqueda_grid.fit(X_train, y_train_encoded)

# Obtener los mejores parámetros
mejores_parametros = busqueda_grid.best_params_

# Imprimir los mejores parámetros
print("Mejores Parámetros:", mejores_parametros)

# Utilizar los mejores parámetros para crear el modelo final
mejor_modelo = XGBClassifier(enable_categorical=True, **mejores_parametros)

# Validar cruzadamente el modelo final
puntuaciones_cv = cross_val_score(mejor_modelo, X_train, y_train_encoded, cv=10,
scoring='accuracy')

# Imprimir puntuaciones de validación cruzada
print("Puntuaciones de Validación Cruzada:", puntuaciones_cv)
print("Precisión Media:", puntuaciones_cv.mean())

# Ajustar el mejor modelo en todo el conjunto de entrenamiento
mejor_modelo.fit(X_train, y_train_encoded)

# Predicciones usando el mejor modelo
y_pred = mejor_modelo.predict(X_test)

# Invertir la transformación para obtener etiquetas originales
y_pred_originales = label_encoder.inverse_transform(y_pred)

# Evaluación
precision = accuracy_score(y_test, y_pred_originales)
precision_balanceada = balanced_accuracy_score(y_test, y_pred_originales)
sensibilidad = recall_score(y_test, y_pred_originales, average='weighted')
puntuacion_f1 = f1_score(y_test, y_pred_originales, average='weighted')

# Finalizar tiempo de ejecución
tiempo_fin = time.time()
tiempo_ejecucion = tiempo_fin - tiempo_inicio
print("Tiempo de Ejecución:", tiempo_ejecucion)

# Imprimir resultados en el conjunto de prueba
print("Resultados en el Conjunto de Prueba:")
print("Precisión:", precision)
print("Precisión Balanceada:", precision_balanceada)
print("Sensibilidad:", sensibilidad)
```

```
print("Puntuación F1:", puntuacion_f1)
```

FIGURA 4.35. Código necesario para entrenar, validar y evaluar el modelo XGBoost.

Código para modelo Random Forest

```
import time
import pandas as pd
from sklearn.model_selection import cross_val_score, GridSearchCV,
train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, balanced_accuracy_score,
precision_score, recall_score, f1_score
from sklearn.preprocessing import LabelEncoder, OneHotEncoder

# Iniciar el tiempo de ejecución
start_time = time.time()

# Suponiendo que df es tu DataFrame con características categóricas
caracteristicas = ['a0', 'a1', 'a2', 'a3', 'a4', 'c1', 'e', 'd41m', 'd42m']
variable_objetivo = 'b1_class'

# Separar el conjunto de datos en conjuntos de entrenamiento, validación y prueba
conjunto_entrenamiento, conjunto_prueba = train_test_split(df, test_size=0.2,
random_state=42)
conjunto_entrenamiento, conjunto_validacion =
train_test_split(conjunto_entrenamiento, test_size=0.2, random_state=42)

X_entrenamiento, y_entrenamiento = conjunto_entrenamiento[caracteristicas],
conjunto_entrenamiento[variable_objetivo]
X_validacion, y_validacion = conjunto_validacion[caracteristicas],
conjunto_validacion[variable_objetivo]
X_prueba, y_prueba = conjunto_prueba[caracteristicas],
conjunto_prueba[variable_objetivo]

# Convertir columnas categóricas a tipo 'category'
caracteristicas_categoricas = ['a0', 'a1', 'a4', 'c1', 'd41m', 'd42m']
X_entrenamiento[caracteristicas_categoricas] =
X_entrenamiento[caracteristicas_categoricas].astype('category')
X_validacion[caracteristicas_categoricas] =
X_validacion[caracteristicas_categoricas].astype('category')
X_prueba[caracteristicas_categoricas] =
X_prueba[caracteristicas_categoricas].astype('category')

# Convertir la variable objetivo a numérica usando LabelEncoder
label_encoder = LabelEncoder()
y_entrenamiento_codificada = label_encoder.fit_transform(y_entrenamiento)
y_validacion_codificada = label_encoder.transform(y_validacion)
y_prueba_codificada = label_encoder.transform(y_prueba)

# Codificación one-hot de características categóricas
encoder = OneHotEncoder(sparse=False, handle_unknown='ignore')

# Ajustar y transformar en el conjunto de entrenamiento
X_entrenamiento_codificada =
encoder.fit_transform(X_entrenamiento[caracteristicas_categoricas])

# Transformar en el conjunto de validación
X_validacion_codificada =
encoder.transform(X_validacion[caracteristicas_categoricas])

# Transformar en el conjunto de prueba
X_prueba_codificada = encoder.transform(X_prueba[caracteristicas_categoricas])

# Crear clasificador RandomForest
modelo = RandomForestClassifier()
```

```
# Definir una cuadrícula de parámetros para ajuste de hiperparámetros
parametros_grid = {
    'n_estimators': [50, 100, 200],
    'max_depth': [None, 10, 20],
    'min_samples_split': [2, 5, 10],
    'min_samples_leaf': [1, 2, 4],
    'bootstrap': [True, False]
}

# Inicializar GridSearchCV
busqueda_grid = GridSearchCV(estimator=modelo, param_grid=parametros_grid,
scoring='accuracy', cv=3)

# Ajustar el modelo con ajuste de hiperparámetros usando cross_val_score
resultados_validacion_cruzada = cross_val_score(busqueda_grid,
X_entrenamiento_codificada, y_entrenamiento_codificada, cv=3, scoring='accuracy')
print("Resultados de Validación Cruzada:", resultados_validacion_cruzada)

# Ajustar el modelo en todo el conjunto de entrenamiento
busqueda_grid.fit(X_entrenamiento_codificada, y_entrenamiento_codificada)

# Obtener los mejores parámetros
mejores_parametros = busqueda_grid.best_params_

# Imprimir los mejores parámetros
print("Mejores Parámetros:", mejores_parametros)

# Predicciones usando el mejor modelo
y_prediccion = busqueda_grid.predict(X_prueba_codificada)

# Inversión de la transformación para obtener las etiquetas originales
y_prediccion_original = label_encoder.inverse_transform(y_prediccion)

# Evaluación
precision = accuracy_score(y_prueba, y_prediccion_original)
precision_equilibrada = balanced_accuracy_score(y_prueba, y_prediccion_original)
precision = precision_score(y_prueba, y_prediccion_original, average='weighted')
recall = recall_score(y_prueba, y_prediccion_original, average='weighted')
f1 = f1_score(y_prueba, y_prediccion_original, average='weighted')

# Finalizar el tiempo de ejecución
end_time = time.time()
tiempo_ejecucion = end_time - start_time
print("Tiempo de Ejecución:", tiempo_ejecucion)

print("Accuracy:", accuracy)
print("Precisión:", precision)
print("Precisión Equilibrada:", precision_equilibrada)
print("Recall:", recall)
print("F1 Score:", f1)
```

FIGURA 4.36. Código necesario para entrenar, validar y evaluar el modelo Random Forest.

La separación del conjunto de datos en datos de entrenamiento, validación y prueba (marcado en el código de la siguiente manera: **`# Separar el conjunto de datos en conjuntos de entrenamiento, validación y prueba`**), se usó para evaluar los modelos de forma adecuada. El conjunto de entrenamiento se utiliza para ajustar los modelos o crear el mapeo, los conjuntos de validación para ajustar los hiperparámetros y los conjuntos de prueba para evaluar el rendimiento final del modelo con datos no vistos. Esta estrategia se suele utilizar para evitar el

sobreajuste del modelo a los datos de prueba y asegurar que la evaluación del modelo sea representativa de su capacidad para generalizar a nuevos datos.

Se utilizó también la técnica de validación cruzada en todos los modelos. En este caso, un sistema de proceso de validación cruzada de diez veces se utilizó para evaluar el rendimiento y el error general de todos los modelos de clasificación. Esta es una técnica de evaluación de modelos de aprendizaje automático que consiste en dividir el conjutno de datos en varios subconjuntos. Cada subconjunto se utiliza como conjunto de prueba una vez y los demás subconjuntos se utilizan como conjuntos de entrenamiento. Este proceso se repite muchas veces, y el resultado final es la media de las evaluaciones de los subconjuntos (Hastie et al., 2009; Kohavi, R., 1995).

Para la optimización de hiperparámetros se utilizó GridSearchCV y optuna para encontrar la configuración de hiperparámetros que maximiza las métricas de evaluación. La optimización puede mejorar significativamente el rendimiento del modelo (Bergstra & Bengio, 2013).

Para la evaluación del modelo se realizaron las siguientes acciones: predicciones y cálculo de métricas. Se realizaron predicciones en el conjunto de prueba utilizando el mejor modelo encontrado durante el ajuste de hiperparámetros. Las predicciones son importantes para evaluar el rendimiento de un modelo debido a que proporcionan una estimación de cómo el modelo se desempeñará con datos nuevos (Hastie et al., 2009). Además, se calcularon las métricas de evaluación como runtime, balanced accuracy, precisión, recall, F1-score y accuracy. Estas métricas son indicadores que se utilizan para cuantificar el rendimiento del modelo. La precisión nos indica la proporción de predicciones correctas, el recall nos indica la proporción de instancias positivas que se identificaron correctamente, el F1-score es una combinación de precisión y recall, accuracy indica la proporción de instancias correctamente identificadas, el run time es el tiempo que tarda el modelo en realizar la predicción, y balanced accuracy se calcula como la media de la precisión y el recall, el balanced accuracy es una métrica más robusta que la precisión o el recall cuando las clases están desbalanceadas. (James et al., 2013; Chakraborty & Chawla, 2014; Otero & Pérez, 2018)

5. RESULTADOS

Algoritmo	Tiempo de ejecución (segundos)	Accuracy (Validación cruzada)	Balanced Accuracy (Test set)	Precision (Test set)	Recall (Test set)	F1 Score (Test set)	Parámetros (Hyperparameter Tuning)
XGBoost	13.433656692504883	**0.8421568627450980**	**0.8011904761904761**	0.9094889463106518	0.9074074074074074	0.9038575956246638	{'learning_rate': 0.8, 'max_depth': 14, 'n_estimators': 200}
RF	385.53399634361267	0.783625726	0.7535714285714286	0.8834175084175084	0.8888888888888888	0.8781958781958781	{'bootstrap': True, 'max_depth': 20, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 50}
CatBoost	7.23622727394104	0.8009803921568628	0.380952	0.818309	0.759259	0.669604	{'learning_rate': 0.17599677536365474, 'depth': 5}
ANN	**46.830013036727905**	**0.8251633986928105**	**0.7678571428571429**	**0.8332858499525166**	**0.8333333333333334**	**0.8279211825377127**	**{'units': 192, 'learning_rate': 0.08292744965431652}**

TABLA 5.1. En esta table se muestran los resultados de cada métrica por cada modelo de aprendizaje automático.

Los resultados de cada modelo se muestran en la Figura X. En base a esta tabla se puede observar que Random Forest obtuvo los peores resultados con un accuracy (validación cruzada) de 78.36%, precision de 88.34%, tiempo de ejecución de 385.53 segundos, 75.35% de balanced accuracy, 88.88% de recall y 87.81% de F1 Score. Cabe destacar que a excepción de Random Forest, todos los algoritmos consiguieron accuracies superiores a 80%, siendo XGBoost el algoritmo con el más alto porcentaje, 84.21% en accuracy (validación cruzada), indicando que este algoritmo podría tener valor significativo de predicción para la variable objetivo.

6. DISCUSIÓN

Los resultados del studio muestran que los algoritmos de aprendizaje automático pueden ser una herramienta eficaz para predecir la adherencia a la dieta mediterránea. El modelo XGBoost alcanzó un accuracy de 84.21% con validación cruzada, lo que indica que es capaz de identificar con precision la proporción de instancias correctamente clasificadas. Las características que se usaron para alcanzar ese procentaje de predicción fueron 8: a0 (Facultad), a1 (Grado), a3(Edad), a4 (Sexo), c1 (FCQ-SQP), e (IMC), d41m ((¿Qué nivel educativo tiene tu padre?),), d42m ((¿Qué actividad laboral realiza tu madre?).

El trabajo tiene ciertas limitaciones que deben ser consideradas. La primera es que la recolección inicial de datos se hizo no pensando en la creación de un modelo predictivo, si no únicamente para hacer un análisis descriptivo y exploratorio. Este problema se ha manifestado en la reducción de características relevantes. De haberse planteado la creación de un modelo predictivo en primera instancia, el enfoque habría variado para intentar encontrar un conjunto de características y variable objetivo con relaciones más significativas, esto podría haber

llevado a recopilar más datos u otros datos que podrían haberse obtenido tras consultar a expertos en el área de nutrición.

Sin duda, la inteligencia artificial en general, y los algoritmos predictivos en específico, pueden revolucionar el área de la salud al ayudarnos a comprender mejor qué variables influyen en la adherencia a una buena dieta, o en general a hábitos saludables, existen retos técnicos a los que se deberá hacer frente. Estos modelos, debido a que dependen de la disponibilidad de grandes cantidades de datos de alta calidad y significativos, se deberá hacer un especial énfasis en recopilar datos representativos de la población objetivo y del problema que se intenta resolver. De las 29 características iniciales, solo 8 se consideraron como relevantes, esto quiere decir que los datos disponibles no eran lo suficientemente informativos para aprender las relaciones entre las características y la variable objetivo. Esto sugiere que, en futuras investigaciones, podría ser beneficioso considerar las 8 características seleccionadas en este trabajo como punto de partida, sin necesariamente incluir las otras 21 características en el estudio de la adherencia a la dieta mediterránea.

Los resultados del estudio tienen implicaciones para investigaciones futuras. En primer lugar, el estudio sugiere que los algoritmos de AA pueden ser una herramienta eficaz para predecir la adherencia. Esto abre la posibilidad de utilizar estos algoritmos para desarrollar programas de intervención más efectivos, cuyo objetivo sea promover la adherencia a la dieta mediterránea.

En segundo lugar, los resultados proporcionan información sobre los factores que influyen en la adherencia a la dieta mediterránea. Esta información se podría utilizar para desarrollar programas más personalizados de intervención que se adapten a las necesidades específicas de los individuos.

7. CONCLUSIONES

En este estudio se ha demostrado que los algoritmos de AA son una herramienta eficaz para predecir la adherencia a la dieta mediterránea. El modelo de AA con más precision, 84.21%, fue el XGBoost, lo que indica que es capaz de identificar con ese porcentaje de probabilidad el tipo de adherencia que tiene una persona.

Algunas consideraciones para futuras investigaciones podrían ser la ampliación del tamaño del conjunto de datos para evaluar la capacidad del modelo para generalizar a otros grupos de personas, examinar la causalidad de las relaciones entre caraterísticas seleccionadas y adherencia a la dieta mediterránea e investigar el impacto de los programas de intervención dirigidos a personas con las características seleccionadas en este estudio.

8. BIBLIOGRAFÍA

Alonso, J., Caballero, G., Fernández, L., García, V., Lama, C., Muñoz, J., ... & Rubio, J. (2004). Programa de promoción saludable en la escuela. Junta de Andalucía.

Alpaydin, E. (2014). Introduction to machine learning. Cambridge, MA: MIT Press.

Aranceta, J. (2015). Influencia de los medios de comunicación en la elección de alimentos y en los hábitos de consumo alimentario [Tesis doctoral, Universidad del País Vasco]. Recuperado de https://addi.ehu.es/bitstream/handle/10810/18487/TESIS_ARANCETA_BARTRINA_JAVIER.pdf?sequence=1

Barrios-Vicedo, Ricardo, Navarrete-Muñoz, Eva María, García de la Hera, Manuela, González-Palacios, Sandra, Valera-Gran, Desirée, Checa-Sevilla, José Francisco, Gimenez-Monzo, Daniel, & Vioque, Jesús. (2015). Una menor adherencia a la dieta mediterránea se asocia a una peor salud auto-percibida en población universitaria. Nutrición Hospitalaria, 31(2), 785-792. https://dx.doi.org/10.3305/nh.2015.31.2.7874

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. Journal of Machine Learning Research, 13(1), 281-305.

Birch, L.L., & Fisher, J.O. (1998). Development of Eating Behaviors Among Children and Adolescents. Pediatrics, 101(Supplement 2).

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent Dirichlet allocation. Journal of Machine Learning Research, 3(4), 993-1022.

Blum, K., Liu, Y., Shriner, R., & Gold, M. S. (2011). Reward circuitry dopaminergic activation regulates food and drug craving behavior. Current

Bolaños Ríos, P. (2009). Evolución de los hábitos alimentarios. De la salud a la enfermedad por medio de la alimentación. Retrieved from http://www.tcasevilla.com/archivos/evolucion_de_los_habitos_alimentarios._de_la_salud_a_la_enfermedad_por_medio_de_la_alimentacion.pdf

Busdiecker, B.S., Castillo, D.C., & Salas, A.I. (2000). Cambios en los hábitos de alimentación durante la infancia: una visión antropológica. Revista Chilena de Pediatría, 71(1), 5–11. Retrieved from http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0370-41062000000100003&lng=en&nrm=iso&tlng=en

Carbajal, A. y Ortega, R. M. (2001). Dieta mediterránea: ¿Qué se entiende por dieta mediterránea? Revista Chilena de Nutrición, 28(3), 238-2431

Celis-Morales, C., Lara, J., & Mathers, J. C. (2015). Personalización de la orientación nutricional para un cambio de comportamiento más efectivo. Proceedings of the Nutrition Society, 74(2), 130-138. https://doi.org/10.1017/S0029665114001633.

Celis-Morales, C., Livingstone, K. M., Marsaux, C. F. M., Macready, A. L., Fallaize, R., O'Donovan, C. B., ... Mathers, J. C. (2017). Efecto de la nutrición personalizada en el cambio de comportamiento relacionado con la salud: evidencia del ensayo controlado aleatorio europeo Food4Me. International Journal of Epidemiology, 46(2), 578-588. https://doi.org/10.1093/ije/dyw186.

Cervera Burriel, F., Serrano Urrea, R., Vico García, C., Milla Tobarra, M., & García Meseguer, M. J. (2013). Hábitos alimentarios y evaluación nutricional en una población universitaria. Nutrición Hospitalaria, 28(2), 438-446.

Cervera Burriel, F., Serrano Urrea, R., Vico García, C., Milla Tobarra, M., & García Meseguer, M. J. (2021). Hábitos de alimentación y calidad de dieta en estudiantes universitarias de Magisterio en relación a su adherencia a la dieta mediterránea. Revista Española de Salud Pública, 95, e1-e12.

Cervera, F. (2014). Hábitos alimentarios en estudiantes universitarios: Universidad de Castilla-La Mancha. Estudio piloto en la Universidad Virtual de Túnez.

Cervera, P., Clapés, J., & Rigolfas, R. (2004). Alimentación y Dietoterapia. Recuperado de https://vizcayanutricion.files.wordpress.com/2013/10/alimentacion-y-dietoterapia-4ed-cervera-p.pdf

Chacón-Cuberos, R., Castro-Sánchez, M., Muros-Molina, J. J., Espejo-Garcés, T., Zurita-Ortega, F., & Linares-Manrique, M. (2016). Adhesión a la dieta mediterránea en

estudiantes universitarios y su relación con los hábitos de ocio digital. Nutrición Hospitalaria, 33(2), 437-444.

Chakraborty, S., & Chawla, N. V. (2014). A comparison of evaluation measures for imbalanced classification. In Proceedings of the 2014 ACM SIGKDD international conference on knowledge discovery and data mining (pp. 1920-1928). ACM.

Christoph, M. J., & An, R. (2018). Effect of nutrition labels on dietary quality among college students: A systematic review and meta-analysis. Nutrition Reviews, 76(3), 187-203. https://doi.org/10.1093/nutrit/nux069

Clark, M. (2017). Automate the boring stuff with Python: Practical programming for total beginners. No Starch Press.

Cover, T. M., & Thomas, J. A. (2006). Elements of information theory (2nd ed.). New York: Wiley-Interscience.

Creswell, J. W. (2014). Research design: Qualitative, quantitative, and mixed methods approaches (4th ed.). Thousand Oaks, CA: Sage Publications.

Cross Validated. (2015). Practical ways to deal with a large number of variables. https://stats.stackexchange.com/questions/174638/practical-ways-to-deal-with-a-large-number-of-variables

Cui, Y., & Grossman, R. L. (2006). A Bayesian approach to feature selection for classification. IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(11), 1929-1940.

Databricks. (n.d.). What are DataFrames? Recuperado el 3 de julio de 2023, de https://www.databricks.com/glossary/what-are-DataFrames

De Luis, D., Bellido, D., & García, P. (2010). Dietoterapia, Nutrición Clínica y Metabolismo. Editorial Díaz de Santos.

Dodge, Y. (2008). The Concise Encyclopedia of Statistics. Springer New York, NY

Duarte, C., Ramos, D., Latorre, A., & González, P. (2016). Factores relacionados con las prácticas alimentarias de estudiantes de tres universidades de Bogotá. Revista Salud Pública,

17(6), 925–937. Retrieved from http://revistas.unal.edu.co/index.php/revsaludpublica/article/view/38368

Dumbrell S, Mathai D. Getting young men to eat more fruit and vegetables: a qualitative investigation. Health Promot J Austr. 2008 Dec;19(3):216-21. doi: 10.1071/he08216. PMID: 19053939.

Durá, T., & Castroviejo, A. (2011). Adherencia a la dieta mediterránea en la población universitaria. Nutrición Hospitalaria, 26(3).

Escuela de alimentación. (2012). Energía para crecer.

Estruch, R., Ros, E., Salas-Salvadó, J., Covas, M.-I., Corella, D., Arós, F., Gómez-Gracia, E., Ruiz-Gutiérrez, V., Fiol, M., Lapetra, J., Lamuela-Raventos, R.M., Serra-Majem, L., Pintó, X., Basora, J., Muñoz, M.A., Sorlí, J.V., Martínez, J.A., Fitó, M., Gea, A., Hernán, M.A., ... Martínez-González, M.A. (2018). Primary prevention of cardiovascular disease with a Mediterranean diet supplemented with extra-virgin olive oil or nuts. The New England Journal of Medicine, 378(25), e34. [https://doi.org/10.1056/NEJMoa1800389]

Eufic. (2019). Los factores determinantes de la elección de alimentos. Recuperado el 23 de agosto de 2023, de https://www.eufic.org/es/vida-sana/articulo/los-factores-determinantes-de-la-eleccion-de-alimentos.

FAO. (2011). La importancia de la educación nutricional.

Firth J, Gangwisch J E, Borsini A, Wootton R E, Mayer E A. Food and mood: how do diet and nutrition affect mental wellbeing? BMJ 2020; 369 :m2382 doi:10.1136/bmj.m2382

Flandrin, J-L. (2004). Historia de la alimentación: Por una ampliación de las perspectivas. Recuperado de https://ddd.uab.cat/pub/manuscrits/02132397n6/02132397n6p7.pdf

Flandrin, J-L. (2004). Historia de la alimentación: Por una ampliación de las perspectivas. Recuperado de https://ddd.uab.cat/pub/manuscrits/02132397n6/02132397n6p7.pdf

French, S. A. (2003). Pricing Effects on Food Choices. *The Journal of Nutrition*, 133(3), 841S-843S. https://doi.org/10.1093/jn/133.3.841S.

García, A. (2004). Estudio de los hábitos alimentarios en población universitaria y sus condicionantes (Tesis doctoral). Universitat Autònoma de Barcelona, España.

García, P. (2002). Evaluación del estado nutricional de la población en la Universidad Politécnica de Valencia.

García-González, Á., Achón, M., Alonso-Aperte, E., & Varela-Moreiras, G. (2018). Análisis de los factores que influyen en la elección de alimentos en estudiantes universitarios mexicanos. Revista Española de Nutrición Humana y Dietética, 22(3), 217-224.

GBD Diet Collaborators. (2019). Health effects of dietary risks in 195 countries, 1990-2017: a systematic analysis for the Global Burden of Disease Study 2017. The Lancet, 393(10184), 1-15. https://doi.org/10.1016/S0140-6736(19)30041-8

Gibson, E. L. (2006). Emotional influences on food choice: Sensory, physiological and psychological pathways. Physiology & Behavior, 89(1), 53-61. https://doi.org/10.1016/j.physbeh.2006.01.024

González Valero, G., Padial Ruz, R., Espejo Garcés, T., Chacón Cuberos, R., Puertas Molero, P., y Pérez Cortés, A. J. (2017). Relación entre clima motivacional hacia el deporte y adherencia a la dieta mediterránea en estudiantes universitarios de educación física. International Journal of Developmental and Educational Psychology. Revista INFAD de Psicología., 4(1), 285. https://doi.org/10.17060/ijodaep.2017.n1.v4.1058

González, S., Moreno-Villares, J. M., & Maldonado, J. (2023). Evidencia actual sobre los beneficios de la dieta mediterránea en salud. Revista médica de Chile, 144(8), 1044-1053.

González-Arratia López Fuentes, N. I., Valdez Medina, J. L., & Oudhof van Barneveld, H. (2016). Las emociones y la conducta alimentaria. Acta de Investigación Psicológica / Psychological Research Records, 6(1), 2348-2359.

Google. (2018). Exploratory data analysis for feature selection in machine learning [Archivo PDF]. Recuperado de https://services.google.com/fh/files/misc/exploratory_data_analysis_for_feature_selection_in_machine_learning.pdf

Greppi, D. (2012). Hábitos alimentarios en escolares adolescentes. Universidad Abierta Interamericana.

Guyon, I., & Elisseeff, A. (2003). An introduction to variable and feature selection. The Journal of Machine Learning Research, 3(Mar), 1157-1182.

Hall, M. A., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. H. (2009). The WEKA data mining software: an update. ACM SIGKDD Explorations Newsletter, 11(1), 10-18.

Han, Jiawei, Micheline Kamber, and Jian Pei. Data mining: concepts and techniques. Elsevier, 2011.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning. New York, NY: Springer.

Herman CP, Polivy J, Esses VM. The illusion of counter-regulation. Appetite. 1987 Dec;9(3):161-9. doi: 10.1016/s0195-6663(87)80010-7. PMID: 3435133.

Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, M. P. (2014). Metodología de la investigación (6a ed.). México: McGraw-Hill.

Hospital Clínic. (2023). “Beneficios para la salud de la dieta mediterránea”. Recuperado de: https://www.mayoclinic.org/es/healthy-lifestyle/nutrition-and-healthy-eating/in-depth/mediterranean-diet/art-20047801

Hospital Clínic. (2023). “Beneficios para la salud de la dieta mediterránea”. Recuperado de: https://www.mayoclinic.org/es/healthy-lifestyle/nutrition-and-healthy-eating/in-depth/mediterranean-diet/art-20047801

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. ACM Computing Surveys (CSUR), 31(3), 264-323.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (2nd ed.). New York, NY: Springer.

Jáuregui-Lobera, I., & Bolaños Ríos, P.. (2011). What motivates the consumer's food choice?. Nutrición Hospitalaria, 26(6), 1313-1321. Recuperado en 22 de agosto de 2023, de

http://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S0212-16112011000600018&lng=es&tlng=en.

Jáuregui-Lobera, I., García-Cruz, P., Carbonero-Carreño, R., Magallares, A., & Ruiz-Prieto, I. (2014). Psychometric Properties of Spanish Version of the Three-Factor Eating Questionnaire-R18 (Tfeq-Sp) and Its Relationship with Some Eating- and Body Image-Related Variables. Nutrients, 6(12), 5619–5635. MDPI AG. Retrieved from http://dx.doi.org/10.3390/nu6125619

Kastorini CM, Milionis HJ, Esposito K, Giugliano D, Goudevenos JA, Panagiotakos DB. The effect of Mediterranean diet on metabolic syndrome and its components: a meta-analysis of 50 studies and 534,906 individuals. J Am Coll Cardiol. 2011 Mar 15;57(11):1299-313. doi: 10.1016/j.jacc.2010.09.073. PMID: 21392646.

Kaye WH, Welztin TE, McKee M, McConaha C, Hansen D, Hsu LK. Laboratory assessment of feeding behavior in bulimia nervosa and healthy women: methods for developing a human-feeding laboratory. Am J Clin Nutr 1992; 55: 372-380.

Kirk D, Catal C, Tekinerdogan B. Precision nutrition: A systematic literature review. Comput Biol Med. 2021 Jun;133:104365. doi: 10.1016/j.compbiomed.2021.104365. Epub 2021 Apr 7. PMID: 33866251.

Kirk D, Kok E, Tufano M, Tekinerdogan B, Feskens EJM, Camps G. Machine Learning in Nutrition Research. Adv Nutr. 2022 Dec 22;13(6):2573-2589. doi: 10.1093/advances/nmac103. Erratum in: Adv Nutr. 2023 May;14(3):584. Erratum in: Adv Nutr. 2023 Apr 1;: PMID: 36166846; PMCID: PMC9776646.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation. In Proceedings of the 14th International Joint Conference on Artificial Intelligence (pp. 1191-1196). Montreal, Canada.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation. In Proceedings of the 14th International Joint Conference on Artificial Intelligence (pp. 1191-1196). Montreal, Canada.

Köhler-Forsberg O, N Lydholm C, Hjorthøj C, Nordentoft M, Mors O, Benros ME. Efficacy of anti-inflammatory treatment on major depressive disorder or depressive symptoms:

meta-analysis of clinical trials. Acta Psychiatr Scand. 2019 May;139(5):404-419. doi: 10.1111/acps.13016. Epub 2019 Mar 28. PMID: 30834514.

Kontogianni, M. D., Farmaki, A. E., Vidra, N., Sofrona, S., Magkanari, F., & Yannakoulia, M. (2010). Associations between lifestyle patterns and body mass index in a sample of Greek children and adolescents. Journal of the American Dietetic Association, 110, 215-221.

Kraus Barragán, R. (2021). Tratamiento de variables categóricas en modelos de machine learning (Trabajo de Fin de Máster). Universidad Complutense de Madrid y Universidad Politécnica de Madrid. Recuperado de https://eprints.ucm.es/id/eprint/75311/1/TFM_Rodrigo_Kraus_Barragan.pdf

Lantz, B. Machine learning with R. 3rd edition. Birmingham, UK; Packt Publishing Ltd, 2019, pp. 1–26.

Leigh Gibson, E., & Green, M. W. (2002). Nutritional influences on cognitive function: mechanisms of susceptibility. Nutrition Research Reviews, 15(1), 169-206. https://doi.org/10.1079/NRR200131

León-Muñoz, L. M., Guallar-Castillón, P., Graciani, A., López-García, E., Mesas, A. E., Aguilera, M. T., et al. (2012). Adherence to the Mediterranean Diet Pattern Has Declined in Spanish Adults. Journal of Nutrition, 142(10), 1843-1850.

Linares-Manrique, M., Linares-Girela, D., Schmidt-Rio-Valle, J., Mato-Medina, O., Fernández-García, R., & Cruz-Quintana, F. (2016). Relación entre autoconcepto físico, ansiedad e IMC en estudiantes universitarios mexicanos / The Relation of Physical Self-Concept, Anxiety, and BMI Among Mexicam University Students. Revista Internacional De Medicina Y Ciencias De La Actividad Física Y Del Deporte, (63), 7-22. https://doi.org/10.15366/rimcafd2016.63.007

Liu, H., & Motoda, H. (2012). Feature selection for knowledge discovery and data mining. Springer Science & Business Media.

Liu, H., & Motoda, H. (2012). Feature selection for knowledge discovery and data mining. Springer Science & Business Media.

Marrodán, M. D., Martínez-Álvarez, J. R., Villarino, A., Alférez-García, I., de Espinosa, M. G., López-Ejeda, N., ... & Grupo de Estudio de Nutrición y Obesidad (GENU). (2013). Utilidad de los datos antropométricos autodeclarados para la evaluación de la obesidad en la población española; estudio EPINUT-ARKOPHARMA. Nutrición Hospitalaria, 28(3), 657-663.

Martínez, C., Veiga, P., López, A., Cobo, J., & Carbajal, A. (2005). Evaluación del estado nutricional de un grupo de estudiantes universitarios mediante parámetros dietéticos y de composición corporal. Nutrición Hospitalaria, 20(3), 197-203.

Martínez-González, M. A., Gea, A., & Ruiz-Canela, M. (2018). La dieta mediterránea como ejemplo de una alimentación y nutrición saludables. Nutrición Hospitalaria, 35(1), 18-24. https://doi.org/10.20960/nh.2133.

Mayo Clinic. (2023). “La dieta mediterránea, saludable para el corazón”. Recuperado de: https://www.mayoclinic.org/es/healthy-lifestyle/nutrition-and-healthy-eating/in-depth/mediterranean-diet/art-20047801

Maza Avila, F. J., Caneda-Bermejo, M. C., & Vivas-Castillo, A. C. (2022). Hábitos alimenticios y sus efectos en la salud de los estudiantes universitarios. Una revisión sistemática de la literatura: Dietary habits and health effects among university students. A systematic review. Psicogente, 25(47), 1–31. https://doi.org/10.17081/psico.25.47.4861

McKay, D. J., & Conover, W. J. (1992). A comparison of three methods for selecting values of input variables in the analysis of categorical data. Journal of the American Statistical Association, 87(419), 394-404.

McKay, D. J., & Conover, W. J. (1992). A comparison of three methods for selecting values of input variables in the analysis of categorical data. Journal of the American Statistical Association, 87(419), 394-404.

McKinney, W. (2010). Data structures for statistical computing in Python. In S. van der Walt & J. Millman (Eds.), Proceedings of the 9th Python in Science Conference (pp. 51-56). Recuperado el 26 de agosto de 2023, de https://conference.scipy.org/proceedings/scipy2010/mckinney.html

Montero, A., Úbeda, N., & García, A. (2006). Evaluación de los hábitos alimentarios de una población de estudiantes universitarios en relación con sus conocimientos nutricionales. Nutrición Hospitalaria, 21(4), 466-473. Disponible en http://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S0212-16112006000700004

Moreno, M. (2012). Definición y clasificación de la obesidad. Revista Médica Clínica Los Condes, 23(2), 124-128.

Morris, M. (2010). Identificación de los determinantes sociales de la alimentación en un grupo de familias pertenecientes a los estratos 1, 2 y 3 de la localidad de Fontibon. Tesis de maestría, Pontificia Universidad Javeriana.

Murphy, K. P. (2012). Machine learning: A probabilistic perspective. Cambridge, MA: MIT Press.

Navarro-Prado, S. (2016). Hábitos, estilo de vida y nivel nutricional de la población universitaria del campus de Melilla: Factores condicionantes y riesgos en salud.

Nuttall F Q (2015). Body Mass Index: Obesity, BMI, and Health: A Critical Review. Nutrition today 50(3):117–128.

Obesity: preventing and managing the global epidemic. Report of a WHO consultation. World Health Organ Tech Rep Ser. 2000;894:i-xii, 1-253. PMID: 11234459.

Oliphant, T. E. (2006). A guide to NumPy (Vol. 1). Trelgol Publishing USA.

OMS. (1948). Constitución de la Organización Mundial de la Salud [Internet]. DOCUMENTOS BÁSICOS. Retrieved from http://apps.who.int/gb/bd/PDF/bd48/basic-documents-48th-edition-sp.pdf?ua=1#page=7

Organización Mundial de la Salud (OMS). (2012). OMS | 10 datos sobre la obesidad. OMS.

Organización Mundial de la Salud. (2022). WHO European Regional Obesity Report 2022. https://www.who.int/europe/publications/i/item/9789289057738. Accessed 11 June 2022.

Ortiz-Moncada, R., Norte Navarro, A. I., Zaragoza Martí, A., Fernández Sáez, J., Davó Blanes, M. C., & Miguel Hernández España, U. (2012). ¿Siguen patrones de dieta mediterránea los universitarios españoles?

Otero, J., & Pérez, C. (2018). A survey of machine learning in real-time applications. Pattern Recognition Letters, 108, 200-216.

Paillacho Chamorro, J. E., y Solano Andrade, C. E. (2011). Hábitos alimentarios y su relación con los factores sociales y estilo de vida de los profesionales del volante de la Coop. 28 de septiembre de la Ciudad de Ibarra [Universidad Técnica del norte]. http://repositorio.utn.edu.ec/handle/123456789/663

pandas. (2023). Python Data Analysis Library. Recuperado el 26 de agosto de 2023, de https://pandas.pydata.org/

Pelchat, M. L. (2009). Food Addiction in Humans. The Journal of Nutrition,

Pelto, G.H., Pelto, P.J., Messer, E., & United Nations University. (1989). Research methods in nutritional anthropology. Retrieved from http://archive.unu.edu/unupress/unupbooks/80632e/80632E00.htm

Pharmaceutical Design, 17(12), 1158-1167

Prieto-González P, Sánchez-Infante J, Fernández-Galván LM. Association between Adherence to the Mediterranean Diet and Anthropometric and Health Variables in College-Aged Males. Nutrients. 2022 Aug 24;14(17):3471. doi: 10.3390/nu14173471. PMID: 36079727; PMCID: PMC9458199.

Ramos, L., Solís, S., Cancio, D., Robles, B., & Suarez, A. (2002). Estudio sobre dietas y hábitos alimentarios en la población española. Consejo de Seguridad Nuclear. http://www.csn.es/documents/10182/1007505/DOC-05.01+Estudios+sobre+dietas+y+h%C3%A1bitos+alimentarios+en+la+poblaci%C3%B3n+espa%C3%B1ola

Raphaeli, O., & Singer, P. (2021). Towards personalized nutritional treatment for malnutrition using machine learning-based screening tools. Clinical Nutrition, 40(10), 5249-5251. https://doi.org/10.1016/j.clnu.2021.08.013.

Riba, M. (2002). Estudio de los hábitos alimentarios en población universitaria y sus condicionantes. Universidad autónoma de Barcelona.

Rimm, E. B., Stampfer, M. J., Colditz, G. A., Chute, C. G., Litin, L. B., & Willett, W. C. (1990). Validity of self-reported waist and hip circumferences in men and women. Epidemiology, 1(5), 466-473.

Rodríguez Santamaría, Ana, Amigo Vázquez, Isaac, Paz Caballero, Dolores, & Fernández Rodríguez, Concepción. (2009). Eating habits and attitudes and their relationship with Body Mass Index (BMI). The European Journal of Psychiatry, 23(4), 214-224. Recuperado en 23 de agosto de 2023, de http://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S0213-61632009000400002&lng=es&tlng=en.

Rodriguez, J. (1995). Psicología social de la salud [Internet]. Retrieved from https://acunar-2a459.firebaseapp.com/5/Psicologia-Social-De-La-Salud.pdf

Royo, M. (2017). Nutrición en salud pública. Retrieved from http://gesdoc.isciii.es/gesdoccontroller?action=download&id=11/01/2018-5fc6605fd4

Ruiz, E., Del pozo, S., Valero, T., Ávila, J., & Varela, G. (2010). Estudio de hábitos alimentarios y estilos de vida de los universitarios españoles.

Salleras, L. (1985). Educación sanitaria : principios, métodos y aplicaciones [Internet]. Ediciones Díaz de Santos. Retrieved from https://dialnet.unirioja.es/servlet/libro?codigo=169240

Samieri, C., Okereke, O. I., Devore, E. E., & Grodstein, F. (2013). Mediterranean diet and cognitive function in older age. Neurology, 81(13), 1129-1137.

Sánchez-Álvarez, M., de Espinosa, M. G., & Dolores, M. (2012). Comparación entre el Índice de Masa Corporal auto-referido, auto-percibido y antropométrico en adolescentes madrileños. Antropo, 26, 91-97.

Sánchez-Meca, J., & Reyes-Lagunes, I. (2013). Análisis de datos en investigación psicológica. México: Pearson Educación.

Sánchez-Muniz, F. J., & Goñi, I. (2006). Efectos beneficiosos de la dieta mediterránea. Offarm, 25(2), 116-125. Recuperado de [https://www.elsevier.es/es-revista-offarm-4-articulo-

efectos-beneficiosos-dieta-mediterranea-15467](https://www.elsevier.es/es-revista-offarm-4-articulo-efectos-beneficiosos-dieta-mediterranea-15467).

Sánchez-Villegas, A., Bes-Rastrollo, M., Martínez-González, M. A., & Serra-Majem, L. (2006). Adherence to a Mediterranean dietary pattern and weight gain in a follow-up study: the SUN cohort. International Journal of Obesity, 30, 350-358.

Sancho, L., Pérez, G., Torres, M., & Campillo, E. (2007). Estilo de vida y hábitos alimentarios de los adolescentes extremeños. Semergen, 33(3), 113–118. http://dx.doi.org/10.1016/S1138-3593(02)74052-5

Schröder, H., Fitó, M., Estruch, R., Martínez-González, M. A., Corella, D., Salas-Salvadó, J., et al. (2011). A Short Screener Is Valid for Assessing Mediterranean Diet Adherence among Older Spanish Men and Women. Journal of Nutrition, 141(6), 1140-1145.

Sofi, F., Vecchio, S., Giuliani, G., Martinelli, F., Marcucci, R., & Gori, A. M., et al. (2005). Dietary habits, lifestyle, and cardiovascular risk factors in a clinically healthy Italian population: the "Florence" diet is not Mediterranean. European Journal of Clinical Nutrition, 59, 584-591.

Stewart TM, Martin CK, Williamson DA. The Complicated Relationship between Dieting, Dietary Restraint, Caloric Restriction, and Eating Disorders: Is a Shift in Public Health Messaging Warranted? Int J Environ Res Public Health. 2022 Jan 3;19(1):491. doi: 10.3390/ijerph19010491. PMID: 35010751; PMCID: PMC8745028.

Stroebele, N., & de Castro, J.M. (2004). Television viewing is associated with an increase in meal frequency in humans. Appetite, 42(1), 111–3. http://www.ncbi.nlm.nih.gov/pubmed/15036790

Su KP, Lai HC, Yang HT, Su WP, Peng CY, Chang JP, Chang HC, Pariante CM. Omega-3 fatty acids in the prevention of interferon-alpha-induced depression: results from a randomized, controlled trial. Biol Psychiatry. 2014 Oct 1;76(7):559-66. doi: 10.1016/j.biopsych.2014.01.008. Epub 2014 Jan 24. PMID: 24602409.

Swinburn BA, Kraak VI, Allender S, Atkins VJ, Baker PI, Bogard JR, Brinsden H, Calvillo A, De Schutter O, Devarajan R, Ezzati M, Friel S, Goenka S, Hammond RA, Hastings G, Hawkes C, Herrero M, Hovmand PS, Howden M, Jaacks LM, Kapetanaki AB, Kasman M,

Kuhnlein HV, Kumanyika SK, Larijani B, Lobstein T, Long MW, Matsudo VKR, Mills SDH, Morgan G, Morshed A, Nece PM, Pan A, Patterson DW, Sacks G, Shekar M, Simmons GL, Smit W, Tootee A, Vandevijvere S, Waterlander WE, Wolfenden L, Dietz WH. The Global Syndemic of Obesity, Undernutrition, and Climate Change: The Lancet Commission report. Lancet. 2019 Feb 23;393(10173):791-846. doi: 10.1016/S0140-6736(18)32822-8. Epub 2019 Jan 27. Erratum in: Lancet. 2019 Feb 23;393(10173):746. PMID: 30700377.

Tabachnick, B. G., & Fidell, L. S. (2013). Using multivariate statistics (6th ed.). Boston, MA: Pearson Education.

Trichopoulou, A. (2005). Modified Mediterranean diet and survival: EPIC-elderly prospective cohort study. BMJ, 330, 991-995.

Trichopoulou, A., Costacou, T., Bamia, C., & Trichopoulou, D. (2003). Adherence to a Mediterranean diet and survival in a Greek population. New England Journal of Medicine, 348, 2599-2608.

Troncoso P C, Amaya P JP. Factores sociales en las conductas alimentarias de estudiantes universitarios. Rev Chil Nutr [Internet]. 2009 Dec [cited 2018 Feb 9];36(4):1090–7. Available from: http://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0717-75182009000400005&lng=en&nrm=iso&tlng=en

Tukey, J.W. (1977). Exploratory Data Analysis. Addison-Wesley.

Universidad de Extremadura. (2017). Estadísticas e indicadores universitarios. Recuperado de https://www.unex.es/organizacion/servicios-universitarios/unidades/utec/funciones/estadisticas-e-indicadores-universitarios

Vadeboncoeur C., Townsend N., & Foster C.(2014). A meta-analysis of weight gain in first year university students: is freshman 15 a myth?. BMC Obesity 1(22). https://doi.org/10.1186/s40608-014-0022-x

Valenzuela A. Obesidad: Una mirada más allá de la sobrealimentación y el sedentarismo. | SOCHOB [Internet]. 2009 [cited 2018 Feb 9]. Available from: http://www.sochob.cl/web1/obesidad-una-mirada-mas-alla-de-la-sobrealimentacion-y-el-sedentarismo/

VanderPlas, J. (2016). Python data science handbook: Essential tools for working with data. O'Reilly Media. ISBN: 978-1-4919-1209-2

Weaver, T. W., Kushi, L. H., McGovern, P. G., Potter, J. D., Rich, S. S., King, R. A., … & Olson, J. E. (1996). Validation study of self-reported measures of fat distribution. International Journal of Obesity and Related Metabolic Disorders, 20(7), 644-650.

Wiens, J., & Shenoy, E. S. (2018). Machine Learning for Healthcare: On the Verge of a Major Shift in Healthcare Epidemiology. Clinical Infectious Diseases, 66(1), 149–153. https://doi.org/10.1093/cid/cix731

Witten, I. H., Frank, E., & Hall, M. A. (2016). Data mining: practical machine learning tools and techniques (4th ed.). Burlington, MA: Morgan Kaufmann.

Yahfoufi N, Alsadi N, Jambi M, Matar C. The Immunomodulatory and Anti-Inflammatory Role of Polyphenols. Nutrients. 2018 Nov 2;10(11):1618. doi: 10.3390/nu10111618. PMID: 30400131; PMCID: PMC6266803.

Yahia, N., Achkar, A., Abdallah, A., Rizk S (2008). Eating habits and obesity among Lebanese university students. Nutr J 7(32). https://doi.org/10.1186/1475-2891-7-32

Zhang, Y., & Wang, J. (2021). Feature dimensionality reduction: a review. Artificial Intelligence Review, 54(5), 3875–3900. https://doi.org/10.1007/s10462-021-09967-0

Zheng, A., & Casari, A. (2018). Feature engineering for machine learning: principles and techniques. O'Reilly Media.

Zorbas, C., Palermo, C., Chung, A., Iguacel, I., Peeters, A., Bennett, R., & Backholer, K. (2018). Factors perceived to influence healthy eating: a systematic review and meta-ethnographic synthesis of the literature. Nutrition Reviews, 76(12), 861-874. [https://doi.org/10.1093/nutrit/nuy043]

9. ANEXOS

9.1. Cuestionario de adherencia a la dieta mediterránea

CUESTIONARIO DE ADHERENCIA A LA DIETA MEDITERRANEA

Nº	Pregunta	Modo de valoración	Puntos
1	¿Usa usted el aceite de oliva como principal grasa para cocinar?	Si= 1 punto No= 0 puntos	
2	¿Cuánto aceite de oliva consume en total al día (incluyendo el usado para freir, el de las comidas fuera de casa, las ensaladas, etc.)?	4 o más cucharadas = 1 punto Menos de 4cucharadas = 0 puntos	
3	¿Cuántas raciones de verdura u hortalizas consume al día (las guarniciones o acompañamientos contabilizan como ½ ración?	Dos o más a día (al menos una de ellas en ensaladas o crudas)=1 punto Menos de dos raciones= 0 puntos	
4	¿Cuántas piezas de fruta (incluyendo zumo natural) consume al día?	Tres o más al día=1 punto Menos de tres= 0 puntos	
5	¿Cuántas raciones de carnes rojas, hamburguesas, salchichas o embutidos consume al día (una ración equivale a 100-150 gr)?	Menos de una al día=1punto Más de una ración= 0puntos	
6	¿Cuántas raciones de mantequilla, margarina o nata consume al día (una porción individual equivale a 12 gr)?	Menos de una al día=1punto Más de una ración= 0puntos	
7	¿Cuántas bebidas carbonatadas y/o azucaradas (refrescos, colas, tónicas, bitter) consume al día?	Menos de una al día=1 punto Más de una = 0 puntos	
8	¿Bebe vino? ¿Cuánto consume a la semana?	7 o más vasos/semana=1punto Menos de 7/semana= 0puntos	
9	¿Cuántas raciones de legumbres consume a la semana (una ración o plato equivale a 150 gr)?	3 o más por semana=1 punto Menos de 3/semana = 0puntos	
10	¿Cuántas raciones de pescado o mariscos consume a la semana (un plato, pieza o ración equivale a 100-150 gr de pescado ó 4-5 piezas de marisco)?	Tres o más por semana=1punto Menos de tres a la semana = 0puntos	
11	¿Cuántas veces consume repostería comercial (no casera) como galletas, flanes, dulces o pasteles a la semana?	Menos de dos por semana=1 punto	
12	¿Cuántas veces consume frutos secos a la semana (una ración equivale a 30 gr)?	Tres o más por semana=1 punto Menos de 3 a la semana=0 puntos	
13	¿Consume preferentemente carne de pollo, pavo o conejo en vez de ternera, cerdo, hamburguesas o salchichas(carne de pollo: una pieza o ración equivale a 100-150 gr)?	Si= 1 punto No= 0puntos	
14	¿Cuántas veces a la semana consume los vegetales cocinados, la pasta, el arroz u otros platos aderezados con una salsa de tomate, ajo, cebolla o puerro elaborada a fuego lento con aceite de oliva (sofrito)?	Dos o más por semana=1 punto Menos de dos a la semana= 0 puntos	
		RESULTADO FINAL (Total)	

Fuente: Modificado de: Trichopoulou A, Costacou T, Bamia C, Trichopoulos D. Adherente to a mediterranean diet and survival in a greek population. N Engl J Med 2003; 348: 2.599-2.608.

9.2.Factores psicológicos: cuestionario TFEQ-18-SP (Spanish Version)

Spanish Version of the Three Factor Eating Questionnaire-R18 (TFEQ-SP)
Adaptation and validation by Jáuregui-Lobera, I.; García-Cruz, P.; Carbonero-Carreño, R.; Magallares, A. and Ruiz-Prieto, I.; 2014.

(1) Cuando huelo una comida deliciosa me resulta muy difícil no probarla, incluso si acabo de terminar mi comida.
(2) Deliberadamente tomo pequeñas cantidades de comida como medio para controlar mi peso.
(3) Cuando me siento ansioso/a (nervioso/a) sin darme cuenta me encuentro comiendo.
(4) A veces cuando empiezo a comer parece que no puedo parar.
(5) Estar con alguien mientras come me hace sentir hambre como para ponerme a comer también.
(6) Cuando me siento mal (depresivo, infeliz) suelo comer demasiado.
(7) Cuando veo algo muy exquisito me entra tanta hambre que tengo que comerlo en ese mismo momento.
(8) Me siento tan hambriento/a que mi estómago a menudo parece un pozo sin fondo.
(9) Siempre tengo hambre, de modo que para mí es difícil parar de comer hasta que acabo la comida del plato.
(10) Cuando me siento solo/a me consuelo comiendo.
(11) Me controlo conscientemente en las comidas para no ganar peso.
(12) No suelo comer algunos alimentos porque me hacen engordar.
(13) Siempre siento tanta hambre como para poder comer en cualquier momento.
(14) ¿Con qué frecuencia te sientes hambriento/a?
(15) ¿Con qué frecuencia evitas almacenar alimentos muy tentadores/apetecibles?
(16) ¿Con qué probabilidad comes conscientemente menos de lo que quieres?
(17) ¿Continúas comiendo excesivamente, aunque no tengas hambre?
(18) En una escala de 1 a 8, donde 1 significa no restringir la ingesta y 8 significa restricción total, ¿con qué número te valorarías a ti mismo/a?

El cuestionario consta de 18 ítems que se miden en una escala de respuesta de 4 puntos (definitivamente cierto: 1, en su mayoría cierto: 2, en su mayoría falso: 3, definitivamente falso: 4) y las puntuaciones de los ítems se suman en puntuaciones de subescalas: comer con moderación, comer sin control y comer emocional.

9.3. Factores psicológicos: cuestionario FCQ-SP (Spanish Version)

FOOD CHOICE QUESTIONNAIRE-SPANISH VERSION (FCQ-SP)

Steptoe, Pollard and Wardle, 1995
Adaptation and validation by Jáuregui-Lobera and Bolaños-Ríos, 2011

Teniendo en cuenta la siguiente escala…

1. "Nada importante"
2. "No importante"
3. "Ligeramente no importante"
4. "Ni no importante ni importante"
5. "Ligeramente importante"
6. "Importante"
7. "Muy importante"

Es importante para mi que la comida que tomo un día normal…

1. Sea fácil de preparar	1	2	3	4	5	6	7
2. No contenga aditivos	1	2	3	4	5	6	7
3. Sea baja en calorías	1	2	3	4	5	6	7
4. Sepa bien	1	2	3	4	5	6	7
5. Contenga ingredientes naturales	1	2	3	4	5	6	7
6. No sea cara	1	2	3	4	5	6	7
7. Sea baja en grasa	1	2	3	4	5	6	7
8. Sea familiar	1	2	3	4	5	6	7
9. Sea rica en fibra	1	2	3	4	5	6	7
10. Sea nutritiva	1	2	3	4	5	6	7
11. Esté fácilmente disponible en tiendas y supermercados	1	2	3	4	5	6	7
12. Tenga buena relación calidad-precio	1	2	3	4	5	6	7
13. Me anime	1	2	3	4	5	6	7
14. Huela bien	1	2	3	4	5	6	7
15. Pueda cocinarse de forma sencilla	1	2	3	4	5	6	7
16. Me ayude a combatir el estrés	1	2	3	4	5	6	7
17. Me ayude a controlar el peso	1	2	3	4	5	6	7
18. Tenga una textura agradable	1	2	3	4	5	6	7
19. Sea similar a la comida que tomaba cuando era niño	1	2	3	4	5	6	7
20. Contenga muchas vitaminas y minerales	1	2	3	4	5	6	7
21. No tenga ingredientes artificiales	1	2	3	4	5	6	7
22. Me mantenga despierto, alerta	1	2	3	4	5	6	7
23. Parezca agradable	1	2	3	4	5	6	7
24. Me ayude a relajarme	1	2	3	4	5	6	7
25. Sea alta en proteínas	1	2	3	4	5	6	7
26. No me lleve tiempo prepararla	1	2	3	4	5	6	7
27. Me mantenga sano	1	2	3	4	5	6	7
28. Sea buena para mi piel, dientes, pelo, uñas, etc.	1	2	3	4	5	6	7
29. Me haga sentir bien	1	2	3	4	5	6	7
30. Tenga el país de origen claramente señalado	1	2	3	4	5	6	7
31. Sea lo que como habitualmente	1	2	3	4	5	6	7
32. Me ayude a enfrentarme con la vida	1	2	3	4	5	6	7
33. Pueda comprarse en tiendas cerca de la casa o el trabajo	1	2	3	4	5	6	7
34. Sea barata	1	2	3	4	5	6	7

9.4. Factores socioeconómicos: cuestionario socioeconómico

- ¿Cómo te financias económicamente?
- ¿Dónde vives en el curso académico actual?
- ¿Con quién vives actualmente?
- ¿Vive tu padre?
- ¿Vive tu madre
- ¿Cuál es el nivel educativo de tu padre?
- ¿Cuál es el nivel educativo de tu madre?
- ¿Qué actividad laboral desempeña tu padre?
- ¿Qué actividad laboral desempeña tu madre?

9.5. Factores culturales: cuestionario de fuentes y conocimientos sobre alimentación

- ¿Has cursado alguna asignatura de Nutrición y/o temas de Alimentación?
- ¿Lees las etiquetas de los alimentos para saber su composición?
- ¿Qué elemento de la información nutricional es más importante para ti?
- ¿Qué medio publicitario es más relevante para la elección de tus alimentos?
- ¿En qué medio viste la última vez publicidad sobre alimentos?
- ¿A qué fuente habitualmente acudes para consultar sobre alimentación o dietas?

9.6. Factores demográficos: cuestionario demográfico

- ¿En qué facultad estás?
- ¿Qué grado estudias?
- ¿En qué curso estás?
- ¿Qué edad tienes?
- ¿Cuál es tu sexo?

9.7. Factores biológicos: IMC (índica de masa corporal)

$$IMC = \frac{peso(Kg)}{altura^2(m)}$$