Santiago de Cali, 07 de junio de 2024

Doctor,

DIEGO LUIS LINARES
Director de Maestría en Ciencia de Datos
Facultad de Ingeniería y Ciencias
Pontificia Universidad Javeriana de Cali

Asunto: Presentación para evaluación del proyecto aplicado

Cordial Saludo,

Con el fin de cumplir con los requisitos exigidos por la Universidad para optar por el título de Magíster en Ciencia de Datos, nos permitimos presentar a su consideración el proyecto denominado ***"Modelo de clasificación para predecir la salud mental en estudiantes universitarios en Cali: Un enfoque basado en algoritmos desde el Modelo de determinantes sociales de la salud"***, el cual fue realizado por los estudiantes ***Alix Meryam Chaparro Jiménez, Gustavo Ruiz Chacón & Jorge Charry Jiménez,*** con códigos ***8975651, 8979786 y 8979562,*** pertenecientes a la Maestría en Ciencia de Datos, bajo la dirección del Dr. Hernán Camilo Rocha, , en codirección con la Dra. Natalia Cadavid Ruiz.

El suscrito director del Proyecto Aplicado autoriza para que se proceda a hacer la evaluación de este proyecto, toda vez que ha revisado cuidadosamente el documento y avala que ya se encuentra listo para ser presentado y sustentado oficialmente.

Atentamente,

DocuSigned by: 76E4A2B918EC409...
Alix Chaparro

DocuSigned by: 5BB3EFD78D1B498...
Jorge Ivan Charry Jimenez

DocuSigned by: 38B9BDE50F394AD...
Gustavo Ruiz

C.C. 1065831061 de Valledupar

C.C. 1110522505 de Ibagué

C.C. 1090429724 de Cúcuta

Camilo Rocha

Camilo Rocha

Natalia Cadavid Ruiz Firmado digitalmente por Natalia Cadavid Ruiz Fecha: 2024.06.07 15:38:45 -05'00'

Natalia Cadavid Ruiz

C.C. 79948061 de Bogotá

C.C. 37557869 de Bucaramanga

FICHA RESUMEN

PROYECTO APLICADO – MAESTRÍA EN CIENCIA DE DATOS

Modelo de clasificación para predecir la salud mental en estudiantes universitarios en Cali: Un enfoque basado en algoritmos desde el Modelo de determinantes sociales de la salud.

1. **ÁREA DE TRABAJO:** Salud.
2. **TIPO DE PROYECTO (Aplicado, Innovación, Investigación):** Aplicado.
3. **ESTUDIANTE(S):** Alix Meryam Chaparro Jiménez, Gustavo Ruiz Chacón, Jorge Charry Jiménez.
4. **CORREO ELECTRÓNICO:** achaparro18@javerianacali.edu.co, gruiz@javerianacali.edu.co, jorgecharry12@javerianacali.edu.co.
5. **DIRECCIÓN Y TELEFONO:** Calle 28 Número 19-45 Valledupar, Cesar. 304 375 0542; Av. 19 #12A-90 Barrio Cundinamarca, Cúcuta, Norte de Santander. 321 237 9968; Jordan 4 etapa manzana 2 casa 2 Ibagué, Tolima. 320 858 2264.
6. **DIRECTOR:** Dr. Hernán Camilo Rocha.
7. **VINCULACIÓN DEL DIRECTOR:** Pontificia Universidad Javeriana, Cali.
8. **CORREO ELECTRÓNICO DEL DIRECTOR:** camilo.rocha@javerianacali.edu.co.
9. **CO-DIRECTOR (Si aplica):** Dra. Natalia Cadavid Ruiz.
10. **GRUPO O EMPRESA QUE LO AVALA (Si aplica):** Universidad Pontificia Javeriana de Cali
11. **PALABRAS CLAVE (al menos 5):** Salud mental, desempeño académico, factores de riesgo, tecnología, técnicas de predicción, machine learning, ciencia de datos.
12. **FECHA DE INICIO:** 28 de abril de 2023.
13. **FECHA DE FINALIZACIÓN:** 07 de junio de 2024.
14. **RESUMEN:**

Comprender los factores asociados con la salud mental de los estudiantes universitarios es de suma importancia en la sociedad actual. Con la intención de abordar este panorama, se realizó un estudio predictivo para identificar los determinantes sociales de la salud que inciden en la percepción de tres aspectos de salud mental negativa y tres de salud mental positiva de estudiantes de pregrado de una institución educativa en Cali.

Para ello, se utilizó un enfoque basado en algoritmos en el cual se emplearon datos de 2.786 estudiantes universitarios, documentándose el desarrollo de cuatro modelos de clasificación y su ejecución para cada una de las variables a predecir.

Modelo de clasificación para predecir la salud mental en estudiantes universitarios en Cali: Un enfoque basado en algoritmos desde el Modelo de determinantes sociales de la salud.

Alix Meryam Chaparro Jiménez, 8975651
Gustavo Ruiz Chacón, 8979786
Jorge Charry Jiménez, 8979562

Proyecto Aplicado para optar al título de
Magister en Ciencia de Datos

Director(a)
Dr. Camilo Rocha.

Codirector(a)
Dra. Natalia Cadavid Ruiz.

FACULTAD DE INGENIERÍA Y CIENCIAS
MAESTRÍA EN CIENCIA DE DATOS
SANTIAGO DE CALI, JUNIO 07 DE 2024

TABLA DE CONTENIDO

1. LISTADO DE FIGURAS

2. LISTADO DE TABLAS

3. INTRODUCCIÓN

La salud mental es un tema de creciente importancia en la sociedad actual, especialmente entre los estudiantes universitarios. En ese sentido, “hay evidencias de mayor prevalencia de cuadros depresivos entre estudiantes universitarios en comparación con la población general” [1].

En la actualidad “los problemas de salud mental influyen significativamente en el desempeño académico de los estudiantes y su calidad de vida” [1]. En ese orden de ideas, existen una gran cantidad de variables que pueden incidir en la salud mental de los estudiantes. Algunas de las más comunes son: la carga académica, el estrés, la presión social, la falta de apoyo emocional, la falta de sueño y descanso, la dieta, el consumo de sustancias psicoactivas, entre otras.

Los determinantes sociales, de índole económico y académico, contribuyen a comprender la salud de una persona, ya sea para definir el riesgo de padecer una afección a su salud mental o para precisar cómo favorecen a su salud mental positiva. Por ejemplo, “la situación socioeconómica carenciada de muchos estudiantes no sólo es una variable influyente en su desempeño académico sino también se constituye como un factor de riesgo para el desarrollo de trastornos en la salud mental, dado el contexto vulnerable en que se ven insertos” [2]. A la vez, la percepción de redes de apoyo de calidad favorece la salud mental positiva.

Conscientes de la naturaleza compleja de la salud mental, se ha llevado a cabo una encuesta institucional sobre la salud mental, con el fin de identificar qué determinantes sociales de la salud pueden explicar la percepción de salud mental de estudiantes de pregrado de una institución de educación superior en la ciudad de Cali.

En ese orden de ideas, para responder a la pregunta problema se construyeron cuatro modelos de clasificación basados en algoritmos adecuados para predecir la salud mental en estudiantes universitarios en Cali a partir de datos recogidos por la encuesta institucional. Así mismo, se identificaron patrones y relaciones existentes entre las variables recopiladas en la encuesta, en particular los determinantes que podrían estar relacionados con los aspectos positivos y negativos de la salud mental de los encuestados.

4. DEFINICIÓN DEL PROBLEMA

4.1. PLANTEAMIENTO DEL PROBLEMA

La OMS define la salud mental como “un estado de bienestar en el cual cada individuo desarrolla su potencial, puede afrontar las tensiones de la vida, puede trabajar de forma productiva y fructífera, y puede aportar algo a su comunidad” [5]. Así mismo, en Colombia el artículo 3 de la Ley 1616 de 2013 (Ley de Salud Mental) señala que “la salud mental se define como un estado dinámico que se expresa en la vida cotidiana a través del comportamiento y la interacción de manera tal que permite a los sujetos individuales y colectivos desplegar sus recursos emocionales, cognitivos y mentales para transitar por la vida cotidiana, para trabajar, para establecer relaciones significativas y para contribuir a la comunidad” [6].

En Colombia en un estudio descriptivo de corte transversal efectuado con representatividad de las regiones Atlántica, Oriental, Central y pacífica se evidenció que en adultos de 18 a 44 años, el 9.6% presenta síntomas relacionados a algún tipo de trastorno mental, el 52,9% tiene uno o más síntomas de ansiedad y el 80,2% indica que presenta de 1 a 3 síntomas depresivos [7].

Frente a este panorama debe tenerse presente que existen ciertas variables o factores de vulnerabilidad que hacen que algunos individuos presenten un mayor porcentaje de riesgos en padecer alguna afección a su salud mental. En ese orden de ideas, los estudiantes de educación superior son un grupo que presenta mayor riesgo de sufrir afectaciones a su salud mental. Se ha estimado que aproximadamente el 20% de los estudiantes de educación superior presentan síntomas de trastorno mental, ansiedad, depresión, entre otros relacionados además con el consumo de sustancias psicoactivas [8].

Ante esta problemática, en el 2023 la Universidad Javeriana ha manifestado su preocupación y conciencia frente a su población estudiantil conformada por cerca de 23.500 estudiantes, cuyo rango de edad se encuentra dentro de un grupo poblacional que presenta mayor riesgo de padecer afecciones relacionadas a su salud mental [9].

Sin embargo, poco se conoce de las relaciones entre los determinantes sociales, económicos y académicos, así como su influencia en la salud mental de los estudiantes. A tal efecto, aunque estos factores o variables puedan resultar en afecciones relacionadas con la salud mental, el diagnóstico es difícil ya que los síntomas varían entre los individuos [10]. Esta falta de comprensión dificulta la identificación temprana del riesgo de deterioro en la salud mental de los estudiantes.

Actualmente, el diagnóstico de salud mental es responsabilidad de médicos, quienes se centran en identificar la presencia de síntomas asociados a los trastornos de salud mental, como a su frecuencia e intensidad. Pocas veces exploran los determinantes contextuales que favorecen la aparición de

estos síntomas o la presencia de determinantes que actúan como factores protectores. Por ende, es importante identificar a tiempo las poblaciones que presentan mayor riesgo de padecer afecciones a su salud mental y, de esta forma, mejorar las estrategias de diagnóstico [10].

Con todo esto surge la necesidad de generar modelos de clasificación computarizado basados en algoritmos que permitan evaluar los determinantes sociales y académicos, con el objetivo de predecir la salud mental de estudiantes de pregrado, tanto su expresión negativa como positiva, a partir de los resultados obtenidos de la encuesta de salud y bienestar universitario.

4.2. FORMULACIÓN DEL PROBLEMA

Hoy en día existe una preocupación latente sobre el riesgo que tienen los estudiantes universitarios de padecer una afección en su salud mental. Identificar los factores sociales, económicos y académicos que pueden influir en la salud mental de los estudiantes es indispensable para la identificación temprana de esta población en riesgo, así como precisar los determinantes que pueden actuar como factores protectores para promover su salud mental positiva.

La construcción de un modelo de clasificación basado en algoritmos adecuados, a partir de datos tomados de una encuesta institucional sobre los determinantes sociales, económicos y académicos que impactan la salud mental de los estudiantes, busca predecir qué estudiantes dentro de la población de pregrado de una institución de educación superior están en riesgo de presentar síntomas asociados con una salud mental negativa, así como identificar los factores protectores que favorecen una expresión de salud mental positiva.

Bajo este contexto, en esta investigación se responde a la siguiente pregunta problema:

¿Qué determinantes sociales predicen la salud mental, tanto su expresión positiva como negativa, de estudiantes de pregrado de una institución de educación superior en la ciudad de Cali?

5. OBJETIVOS DEL PROYECTO

5.1. OBJETIVO GENERAL

Predecir qué determinantes explican la salud mental, tanto su expresión positiva como negativa, de estudiantes de pregrado de una institución de educación superior en la ciudad de Cali.

5.2. OBJETIVOS ESPECÍFICOS

A. Identificar las variables determinantes sociales, económicas y académicas relacionadas con la salud mental positiva y negativa de la población estudiantil de pregrado encuestada.
B. Desarrollar cuatro modelos de clasificación basados en algoritmos adecuados para predecir qué estudiantes de pregrado de una institución de educación superior en la ciudad de Cali están en riesgo de padecer alguna afección que afecte su salud mental utilizando los datos recopilados.
C. Validar el rendimiento del modelo de clasificación utilizando técnicas apropiadas de validación cruzada y particionamiento de datos.

6. MARCO TEÓRICO Y ANTECEDENTES

6.1. MARCO TEÓRICO

En este apartado se presentan las bases teóricas que sustentan el proyecto de investigación y que orientan su desarrollo. Para ello, se aclara la importancia de comprender la salud mental de la población estudiantil universitaria, así como la aplicabilidad de modelo de clasificación y de algoritmos adecuados para analizar datos relacionados con la percepción de salud mental negativa y positiva, como sus determinantes sociales.

6.1.1. Población estudiantil.

En el escenario educativo contemporáneo, la población estudiantil constituye un elemento central y dinámico que demanda una comprensión profunda y precisa. Este grupo heterogéneo de individuos, inmerso en diversos niveles y modalidades educativas, representa el núcleo vital de la sociedad en constante desarrollo.

Desde las aulas de preescolar hasta las instituciones de educación superior, la población estudiantil abarca una amplia gama de edades, contextos socioeconómicos, culturales y capacidades. El Instituto Vaso de Estadística – Eustat define a la población estudiantil como "todos aquellos individuos que por su edad son susceptibles de ser incluidos en cualquiera de los niveles del sistema educativo vigente" [11].

La concepción sobre la población estudiantil ha sido influenciada significativamente por el trabajo seminal de P. Bourdieu y J. C. Passeron en *"Les héritiers: les étudiants et la culture"* (1964), donde desafían la noción previa de que la educación escolar era liberadora para todos, revelando así una nueva comprensión sobre los efectos sociales de la educación [12]. Esta obra fue seguida por *"La Reproduction"* (1970), del mismo equipo de autores, consolidando aún más la importancia de considerar las desigualdades sociales en los caminos educativos [12].

A partir de la revisión teórica realizada, se observa que cada análisis sobre la población estudiantil ha venido acompañado por una concepción implícita en cuanto a la homogeneidad o diversidad de los estudiantes. En este sentido, las obras de Bourdieu y Passeron destacan la complejidad de esta población, desafiando la idea tradicional de uniformidad en los procesos educativos. La población estudiantil ha evolucionado hacia una comprensión más contextualizada de la diversidad estudiantil, reconociendo la intersección de factores sociales, económicos y culturales que influyen en las trayectorias educativas de los estudiantes.

En este contexto, explorar a fondo las características, tendencias y desafíos que enfrenta la población estudiantil se vuelve esencial para informar políticas, programas y prácticas educativas que impulsen el bienestar de todos los estudiantes.

Respecto a la población estudiantil que hace parte de las instituciones de educación superior, UNESCO (1997) la define como la conformada por estudios "posteriores a la enseñanza secundaria, impartidos por universidades u otros establecimientos que estén habilitados como instituciones de enseñanza superior por las autoridades competentes del país y/o sistemas reconocidos de homologación" [13].

En ese orden de ideas, la educación superior abarca dos niveles fundamentales: el pregrado y el posgrado. En el contexto específico de esta investigación, el nivel de pregrado se divide en tres categorías distintas de formación. Primero, el Nivel Técnico Profesional, que engloba programas diseñados para proporcionar habilidades específicas en áreas técnicas y profesionales. Segundo, el Nivel Tecnológico, dirigido a la capacitación en programas tecnológicos que integran conocimientos teóricos y prácticos en campos especializados. Y tercero, el Nivel Profesional, que comprende programas universitarios diseñados para la formación integral de profesionales en diversas disciplinas académicas y profesionales [14].

En el contexto de la educación superior, la experiencia del estudiante universitario se ve moldeada por la complejidad de los programas de formación ofrecidos. Estos programas, que abarcan desde niveles técnicos y tecnológicos hasta profesionales universitarios, constituyen un escenario en el que los estudiantes se enfrentan a la incertidumbre inherente a su transición hacia el mundo académico y profesional.

G. Felouzis (2001) ofrece un marco de análisis interpretativo que destaca la condición del estudiante universitario como marcada principalmente por la incertidumbre, lo que los impulsa a desarrollar acciones tácticas para adaptarse al entorno universitario emergente [15].

La salud mental de los estudiantes se ve influenciada por una serie de variables determinantes sociales, económicas y académicas, que pueden aumentar la presión y el estrés asociados con el rendimiento académico y la adaptación al entorno universitario. Por lo tanto, la atención a las variables sociales que afectan la salud mental de los estudiantes, es esencial para entender la dinámica de adaptación y desarrollo en la educación superior, así como la implementación de intervenciones preventivas y de apoyo de manera oportuna.

6.1.2. Salud mental.

La salud mental es un componente crucial del bienestar humano que abarca aspectos emocionales, cognitivos y sociales. La Organización Mundial de la Salud define la salud mental como "un estado de bienestar en el cual el individuo es consciente de sus propias capacidades, puede afrontar las tensiones normales de la vida, puede trabajar de forma productiva y fructífera y es capaz de hacer una contribución a su comunidad" [16].

En Colombia la Ley 1616 de 2003 define la salud mental como "un estado dinámico que se expresa en la vida cotidiana a través del comportamiento y la interacción de manera tal que permite a los sujetos individuales y colectivos desplegar sus recursos emocionales, cognitivos y mentales para

transitar por la vida cotidiana, para trabajar, para establecer relaciones significativas y para contribuir a la comunidad" [17].

Bajo esta misma línea, el Ministerio de Salud y Protección Social se ha pronunciado al respecto indicando que "cualquier persona puede presentar un trastorno, problema o evento de salud mental en algún momento de su vida; esto dependerá de la forma como interactúen sus particularidades genéticas, congénitas, biológicas, psicológicas, familiares, sociales y los acontecimientos de su historia de vida" [18].

En el contexto global de la salud mental, la Organización Mundial de la Salud (OMS) informa que cerca de 450 millones de personas padecen trastornos mentales o neurológicos [19], lo que subraya la magnitud del desafío que representa esta problemática en la sociedad contemporánea.

La afectación de la salud mental representa un desafío significativo en la sociedad contemporánea, afectando a individuos de todas las edades, géneros, y contextos socioeconómicos. Para comprender plenamente esta compleja realidad, es fundamental examinarla desde una perspectiva multidimensional que considere no solo los factores biológicos y psicológicos, sino también los determinantes sociales, económicos y culturales que influyen en su origen, manifestación y tratamiento.

Los determinantes sociales, económicos y culturales juegan un papel crucial en la dinámica de la salud mental a lo largo del ciclo de vida y en su expresión negativa como positiva. Como afirma Patel y Kleinman, los determinantes sociales y económicos, como la pobreza, el desempleo, la falta de educación y el estigma social, pueden contribuir significativamente a la carga mental en las comunidades [20].

En este contexto, el presente marco teórico busca explorar diversas teorías, modelos y enfoques que permitan comprender la salud mental en su totalidad. Al respecto, la interacción entre factores biológicos y psicológicos es fundamental en la comprensión de la salud mental. Según el modelo bio-psico-social propuesto por Engel las afecciones mentales son el resultado de la interacción compleja entre factores biológicos, psicológicos y sociales [21].

El modelo de la vulnerabilidad-estrés de Zubin y Spring sugiere que los trastornos mentales surgen de la interacción entre factores genéticos de vulnerabilidad y factores ambientales estresantes [22]. En contraste, el modelo de diátesis-estrés postula que las personas tienen una predisposición biológica (diátesis) que, junto con factores ambientales estresantes, puede desencadenar trastornos mentales [23].

Al explorar una variedad de teorías y modelos para comprender la salud mental, emerge una imagen más completa y multifacética de su naturaleza compleja. Desde el énfasis en la interacción entre factores biológicos, psicológicos y sociales hasta la consideración de la vulnerabilidad genética y los estresores ambientales, estas perspectivas ofrecen un panorama enriquecedor que contribuye a una comprensión más profunda de los trastornos mentales.

6.1.3. Modelo de clasificación.

El Modelo de Clasificación es una herramienta utilizada en diversos campos para organizar, categorizar y estructurar información de manera sistemática. Este modelo se basa en la identificación de características comunes o atributos compartidos entre diferentes elementos, lo que permite agruparlos en categorías o clases específicas.

En palabras de B. Baradwaj & S. Pal, la clasificación es una técnica de análisis de datos muy utilizada, donde se emplea un conjunto de ejemplos preclasificados para construir un modelo capaz de asignar categorías y clasificar a la población de registros en general [24].

Zárate, Bedregal & Cornejo, en un trabajo sobre modelos de clasificación para reconocer patrones de deserción en estudiantes universitarios, precisaron lo siguiente:

"El proceso de clasificación de datos implica dos fases, aprendizaje y clasificación. En la fase de aprendizaje, el algoritmo de clasificación analiza los datos de entrenamiento. En la fase de clasificación, se utilizan los datos de prueba para estimar la precisión de las reglas de clasificación. Si la precisión es aceptable, las reglas se pueden aplicar a las nuevas tuplas de datos.

El algoritmo de entrenamiento del clasificador utiliza los ejemplos preclasificados para determinar el conjunto de parámetros necesarios para una discriminación adecuada, luego codifica esos parámetros en un modelo llamado clasificador" [25].

Existen varios tipos de modelos de clasificación, cada uno adaptado a las necesidades y características del contexto en el que se aplican. Algunos de los modelos más comunes incluyen:

6.1.3.1.Modelo Taxonómico.

Este modelo organiza los elementos en categorías jerárquicas basadas en similitudes y diferencias observadas entre ellos. Un ejemplo de modelo taxonómico es la clasificación biológica de especies [26].

6.1.3.2.Modelo de Aprendizaje Automático.

Este modelo utiliza algoritmos y técnicas computacionales para clasificar datos en función de patrones identificados en un conjunto de entrenamiento. Es ampliamente utilizado en campos como la inteligencia artificial y la minería de datos [27].

6.1.3.3.Modelo de Clasificación por Criterios.

Este modelo establece criterios específicos para asignar elementos a categorías particulares. Por ejemplo, en el campo de la psicología, el DSM (Manual Diagnóstico y Estadístico de los Trastornos Mentales) clasifica los trastornos mentales en función de criterios diagnósticos establecidos [28].

6.1.4. Algoritmos adecuados.

La Real Academia Española define algoritmo como un “conjunto ordenado y finito de operaciones que permite hallar la solución de un problema” [29].

No obstante, los algoritmos juegan un papel fundamental en el proceso de clasificación de datos, permitiendo la identificación y categorización de patrones en conjuntos de datos complejos. La elección de un algoritmo adecuado es crucial para obtener resultados precisos y significativos en tareas de clasificación.

La selección de algoritmos adecuados depende en gran medida de las características específicas del conjunto de datos y de los objetivos de investigación. Es fundamental considerar factores como el tipo de datos (numéricos, categóricos, etc.), el tamaño del conjunto de datos, la naturaleza de los patrones a detectar y la interpretabilidad de los resultados.

Entre los algoritmos más utilizados en tareas de clasificación se encuentran:

6.1.4.1.Regresión Logística.

Aunque comúnmente se utiliza para problemas de regresión, la regresión logística también se puede emplear para clasificación binaria, donde estima la probabilidad de pertenencia a una clase [30].

6.1.4.2.Árboles de Decisión.

Estos modelos utilizan una estructura de árbol para representar decisiones basadas en características de los datos. Son especialmente útiles para problemas de clasificación con datos categóricos y numéricos [31].

6.1.4.3.Máquinas de Soporte Vectorial (SVM).

Las SVM son algoritmos de aprendizaje supervisado que pueden utilizarse tanto para clasificación como para regresión. Buscan encontrar el hiperplano óptimo que mejor separa las clases en el espacio de características [30].

6.1.4.4.K Vecinos más Cercanos (K-NN).

Este algoritmo clasifica un punto de datos basado en la mayoría de las clases de sus vecinos más cercanos en el espacio de características [31].

6.1.5. Datos.

En la era digital, los datos han adquirido una importancia fundamental en todos los ámbitos de la sociedad. Se refieren a la información o registros que pueden ser cuantificados, almacenados y analizados para extraer conocimientos significativos. Los datos pueden ser de naturaleza diversa, incluyendo texto, imágenes, audio, video, números, entre otros formatos.

Bajo una perspectiva constructivista, los datos se interpretan como una representación simbólica, ya sea numérica, alfabética, u otro tipo, que refleja un atributo o característica de una entidad, la cual se evidencia mediante un acontecimiento o un procedimiento [32].

En ese orden de ideas, los datos "permiten representar un estado de la realidad asociado a un momento (tiempo) a través de la codificación (símbolos pertenecientes a un lenguaje) de esta realidad en un medio (soporte y formato) que puede ser entendido, utilizado, compartido y transformado tanto por un humano como por una máquina (Hardware y Software)" [33].

Los datos son un recurso invaluable para la toma de decisiones, la innovación y el progreso en diversos campos, incluyendo la ciencia, la industria, la medicina y la investigación. Permiten identificar patrones, tendencias y correlaciones que de otro modo podrían pasar desapercibidos.

También se emplean para certificar los resultados de la investigación realizada y son reconocidos por la comunidad científica [34].

Los datos pueden incluir "cuadernos de laboratorio, cuadernos de campo, datos de investigación primaria (incluidos los datos en papel o en soporte informático), cuestionarios, cintas de audio, videos, desarrollo de modelos, fotografías, películas, y las comprobaciones y las respuestas de la prueba" [35].

La National Science Foundation [36] ofrece una categorización de datos de investigación según su origen, que facilita la comprensión de su variedad y necesidades de gestión:

6.1.5.1.Datos Observacionales.

Estos registros históricos son únicos en tiempo y lugar, lo que los hace crucialmente importantes para preservar, ya que no pueden ser recreados si se pierden. Ejemplos incluyen encuestas de opinión como las del Centro de Investigaciones Sociológicas (CIS) y datos climatológicos del Banco Nacional de Datos Climatológicos.

6.1.5.2.Datos Experimentales.

Acompañan a los experimentos desde su planificación hasta la obtención de resultados. Aunque en muchos casos los experimentos pueden repetirse para obtener los mismos datos, su repetición puede resultar costosa. Ejemplos incluyen datos generados por el acelerador de partículas del CERN y laboratorios de investigación en diversas disciplinas.

6.1.5.3.Datos Computacionales.

Acompañan a las simulaciones y suelen incluir datos de entrada, programas y resultados. En muchos casos, los resultados pueden ser reproducidos utilizando solo los datos de entrada y los programas. Ejemplos incluyen datos producidos por centros de computación avanzada para simular el funcionamiento de órganos humanos, el movimiento de los astros o la predicción del tiempo.

6.1.6. Variance Inflation Factor (VIF).

El Variance Inflation Factor (VIF) es una medida estadística utilizada en análisis de regresión para identificar la multicolinealidad entre variables predictoras. Introducido por primera vez en 1970 por Brumberg y Johnson [44], el VIF ayuda a cuantificar cuánto aumenta la varianza de un coeficiente de regresión debido a la multicolinealidad.

La multicolinealidad, una preocupación central en el modelado de regresión, se refiere a la alta correlación entre variables predictoras, lo que puede comprometer la precisión de los coeficientes de regresión y dificultar su interpretación [45]. El VIF se calcula ajustando un modelo de regresión lineal para cada variable predictora, tratándola como variable dependiente mientras se utilizan todas las demás variables predictoras como independientes [46]. Un VIF cercano a 1 indica que una variable no está correlacionada con otras, mientras que valores más altos sugieren una multicolinealidad significativa.

El uso del VIF en la selección de variables es crucial, ya que permite identificar y mitigar los efectos perjudiciales de la multicolinealidad en el modelo de regresión. Además, proporciona una base objetiva para la exclusión o inclusión de variables predictoras, lo que mejora la calidad y la interpretabilidad del modelo [47].

6.2. ANTECEDENTES

En los últimos años, ha habido una creciente preocupación por la salud mental de los estudiantes universitarios. Numerosos estudios han demostrado que los jóvenes que ingresan a la educación superior enfrentan desafíos significativos que pueden afectar su bienestar psicológico y emocional. Factores como el estrés académico, la presión social, la carga de trabajo, la falta de apoyo emocional y la transición a la vida universitaria pueden contribuir al desarrollo de trastornos mentales.

Stallman (2010), en un estudio sobre estudiantes matriculados en dos grandes universidades australianas, concluyó que la prevalencia extremadamente alta de problemas de salud mental en los estudiantes universitarios proporciona evidencia de que se trata de una población en riesgo [37].

Dada la complejidad y la gravedad de los problemas de salud mental entre los estudiantes universitarios, es crucial desarrollar estrategias efectivas de predicción y prevención. Identificar tempranamente a los estudiantes en riesgo de padecer afecciones a su salud mental y proporcionar intervenciones adecuadas puede ayudar a mitigar el impacto negativo en su vida académica y personal.

Los trastornos mentales, comunes entre los estudiantes universitarios, tienen inicios que ocurren principalmente antes de ingresar a la universidad, en el caso de trastornos pre-matriculación están

asociados con la deserción universitaria, y típicamente no son tratados. La detección y el tratamiento efectivo de estos trastornos al inicio de la carrera universitaria podrían reducir la deserción y mejorar el funcionamiento educativo y psicosocial [38].

Ante esta situación, el análisis de datos y la inteligencia artificial han emergido como herramientas poderosas para abordar problemas de salud mental. La construcción de modelos de clasificación basados en algoritmos adecuados puede permitir la identificación de patrones y la predicción de riesgos con una precisión sin precedentes. La combinación de datos provenientes de encuestas institucionales sobre determinantes sociales con algoritmos de aprendizaje automático ofrece una oportunidad única para comprender mejor los factores que impactan en la salud mental de los estudiantes universitarios y tomar medidas preventivas efectivas.

La inteligencia artificial y el análisis de datos están transformando rápidamente la atención médica, incluida la salud mental, al permitir una comprensión más profunda de los patrones de afecciones y la personalización de los tratamientos [39].

En general, la aplicación de machine learning (ML) y big data a estudios sobre la salud mental, ha demostrado una variedad de beneficios en áreas como diagnóstico, tratamiento, apoyo, investigación y administración clínica. Con la mayoría de los estudios identificados centrados en la detección y diagnóstico de condiciones de salud mental, es evidente que hay un espacio significativo para la aplicación de ML en áreas como la psicología y, en especial, la salud mental [40].

La preocupación por la salud mental de los estudiantes universitarios ha ido en aumento en los últimos años, evidenciando los desafíos significativos que enfrentan al ingresar a la educación superior. Factores como el estrés académico, la presión social y la transición a la vida universitaria contribuyen al desarrollo de trastornos mentales entre esta población. La alta prevalencia de estos problemas subraya la importancia de desarrollar estrategias efectivas de predicción y prevención.

La aplicación de herramientas de análisis de datos y de inteligencia artificial ofrece una oportunidad única para comprender mejor estos desafíos y tomar medidas preventivas. Además, la rápida transformación en la atención médica, incluida la salud mental, a través de la inteligencia artificial y el análisis de datos, destaca el potencial de estas tecnologías para mejorar el diagnóstico, tratamiento y apoyo en este campo.

7. CONSTRUCCIÓN DEL CONJUNTO DE DATOS

Este capítulo, detalla los procesos empleados para la recopilación y preparación de datos en el marco del proyecto de grado. Se abordan aspectos esenciales, desde la recopilación inicial de datos, hasta el preprocesamiento y la selección de variables pertinentes para la construcción del conjunto de datos a emplear. Estas etapas son cruciales para desarrollar los modelos predictivos y así abordar el tema de la salud mental en la institución educativa y proporcionar apoyo y recursos adecuados a los estudiantes.

7.1. RECOPILACIÓN DE DATOS

El proceso de recopilación de datos se llevó a cabo mediante el acceso a la Encuesta Javeriana de Bienestar y Salud del año 2022, asegurando la conformidad con los protocolos éticos y la protección de la privacidad de los participantes involucrados. Se extendió una invitación al 100% de los estudiantes de pregrado matriculados en la Pontificia Universidad Javeriana de Cali durante el segundo semestre de 2022. Esto resultó en un total de 6.850 estudiantes, de los cuales 2.786 estudiantes aceptaron participar y proporcionaron su consentimiento informado.

Las variables que fueron analizadas en la Encuesta Javeriana de Bienestar y Salud abarcan una amplia gama de aspectos que influyen en la salud mental y el bienestar de los estudiantes universitarios. Desde factores individuales, como la edad y el nivel educativo, hasta variables relacionadas con el entorno social y económico, como el acceso a programas de salud y seguridad social. Este análisis permite comprender no sólo los desafíos que enfrentan los estudiantes en su vida académica, sino también las oportunidades y recursos disponibles para promover su bienestar integral.

Los datos recopilados por la universidad en la Encuesta Javeriana de Bienestar y Salud del año 2022, evaluó la salud mental de los estudiantes a través de cuestionarios que abarcan tanto aspectos positivos, como la resiliencia, los recursos psicológicos y la satisfacción con la vida, como aspectos negativos, que incluyen la depresión, la ansiedad, el estrés y la soledad. Estos aspectos fueron complementados con la exploración de determinantes sociales que influyen en la salud mental de los estudiantes.

Las variables que incluye la base de datos son:

- **Determinantes intermedios individuales sociodemográficos:** Incluye aspectos como el sexo, la edad, el nivel educativo, el estrato socioeconómico, la procedencia y la residencia rural/urbana.

- **Determinantes intermedios psicosociales:** Se consideraron antecedentes de violencia y abuso sexual, apoyo social, funcionamiento familiar y estrategias de afrontamiento.
- **Determinantes intermedios del contexto educativo:** Incluyeron el programa académico, la ubicación semestral, el promedio académico, la carga académica, la participación en trabajo y estudio, la satisfacción académica, los estresores académicos, el conocimiento y acceso a programas de salud y bienestar en la universidad y el ambiente alimentario universitario.
- **Condiciones de vida:** Evaluadas a través de las condiciones de la vivienda y del barrio, la convivencia, la dependencia del núcleo familiar, la seguridad alimentaria, el transporte hogar-universidad y la seguridad social en salud.
- **Determinantes estructurales:** Se incluye el género y la etnia.
- **Conocimiento y acceso a programas de salud y bienestar en la universidad:** Incluyeron diversos programas de apoyo disponibles para los estudiantes.

Esta extensa batería de cuestionarios fue diseñada y administrada a través de la plataforma RedCap, sometiéndola previamente a una prueba piloto para asegurar su eficacia y comprensión antes de la recolección de datos. A partir de los datos recopilados por la universidad, los cuales incluían todas las variables necesarias para el desarrollo de los cuatro modelos construidos, se realizaron selecciones focalizadas en variables consideradas relevantes para el estudio, centrándose especialmente en aquellos aspectos que se identificaron como determinantes clave para la salud mental de los estudiantes de pregrado.

En aras de facilitar la comprensión de las variables utilizadas en este proyecto, así como de promover la transparencia y replicabilidad de los análisis efectuados, se ha incluido en el capítulo de anexos el diccionario de variables correspondiente a la base de datos de la Encuesta Javeriana de Bienestar y Salud del año 2022. Este valioso recurso, proporcionado por la Pontificia Universidad Javeriana de Cali, detalla las definiciones y categorías de cada una de las variables abordadas durante el proceso de encuesta.

7.2. PREPROCESAMIENTO DE DATOS

En este capítulo, se llevó a cabo un análisis exploratorio de los datos con el objetivo de identificar y abordar eficientemente los valores atípicos, así como para gestionar cualquier dato faltante y posibles errores presentes en los registros. Este proceso se erigió como una fase crucial en la preparación de los datos, permitiendo asegurar la calidad y confiabilidad de la información que se utilizó en las etapas subsiguientes del estudio.

7.2.1. Selección y Evaluación de Variables.

En el transcurso de la investigación, se examinaron las variables incluidas en el estudio con el objetivo de identificar aquellas que contribuyen significativamente al análisis. Se emplean técnicas de análisis descriptivo para evaluar la relevancia y comportamiento de las variables, con especial atención a aquellas con baja variabilidad, distribución sesgada o valores constantes. Además, se aborda la gestión de datos faltantes, destacando la importancia de identificar y manejar adecuadamente los datos faltantes para garantizar la integridad del análisis.

7.2.1.1.Datos faltantes.

En cuanto a los datos faltantes, se evaluaron e implementaron estrategias para abordar este desafío. Esta fase, además de centrarse en la recuperación de los datos faltantes, si los hubiera, también se centró en mantener la integridad e importancia a la distribución de los datos, garantizando así la integridad de los resultados obtenidos.

Para llevar a cabo este proceso, se utilizó Python y la biblioteca Pandas para manipular y analizar los datos. Se procedió a cargar el conjunto de datos desde un archivo CSV, identificando y calculando el conteo de datos faltantes en cada columna. Posteriormente, se dividió el conjunto de datos en subconjuntos más pequeños para calcular el conteo de datos faltantes en cada uno de ellos. Esta división en subconjuntos permitió una evaluación más detallada de los datos faltantes en diferentes partes del conjunto de datos.

A continuación, se presenta un fragmento del código utilizado para calcular y guardar la información consolidada del conteo de datos faltantes en subconjuntos:

```
from google.colab import drive
import pandas as pd

# Mount Google Drive to access and save files
df = pd.read_csv('/content/drive/MyDrive/+ Proyect/info_db_proyecto_aplicado.csv')


# Check for missing values in each column and calculate the count of missing values
missing_count = df.isnull().sum()

# Exclude variables with no missing values
missing_count = missing_count[missing_count > 0]

# Order the missing plot by the number of missing values from max to min
missing_count = missing_count.sort_values(ascending=False)

# Create subsets of the data in 20 ranges and calculate missing count in each subset
num_subsets = 20
subset_size = len(df) // num_subsets
subset_missing_count = []

# Create a DataFrame to store missing count information for each subset
subset_missing_df = pd.DataFrame()

for i in range(num_subsets):
    subset_start = i * subset_size
    subset_end = (i + 1) * subset_size
    subset = df.iloc[subset_start:subset_end]
    subset_missing_count.append(subset.isnull().sum())

    # Add missing count information for each variable in the subset to the DataFrame
    subset_missing_df[f'Subset_{i + 1}'] = subset_missing_count[-1]

# Transpose the DataFrame to have the subset identifier as the first column
subset_missing_df = subset_missing_df.transpose()

# Save the consolidated missing count information to a CSV file
subset_missing_df.to_csv('/content/drive/MyDrive/+ Proyect/Consolidated_Missing_Count_1.csv', index=True)

# Display success message
print("Consolidated missing count information saved as 'Consolidated_Missing_Count_1.csv'.")
```

El análisis del código previo reveló información sobre la cantidad de datos faltantes por columna, expresada en el siguiente formato numérico:

Variable	*enfermedadcual*	*condpsiqucual*	*dolorcual*	*spailegales*	*sexpreserv*	*sexsatisfsex*	*imc*
SUBSET 1	114	116	134	138	41	39	25
SUBSET 2	117	120	134	137	44	44	20
SUBSET 3	106	116	128	135	44	44	15
SUBSET 4	118	123	128	137	39	39	28
SUBSET 5	125	126	132	138	60	60	19
SUBSET 6	122	125	132	137	47	47	27

SUBSET 7	119	119	128	138	35	35	18
SUBSET 8	120	126	131	137	48	48	21
SUBSET 9	126	128	129	138	48	48	7
SUBSET10	123	127	132	136	38	39	18
SUBSET11	124	131	133	137	39	40	22
SUBSET12	111	115	130	135	41	40	19
SUBSET13	122	120	133	134	27	27	24
SUBSET14	123	127	133	134	32	33	22
SUBSET15	124	124	131	136	37	37	22
SUBSET16	116	123	127	139	41	42	20
SUBSET17	121	118	130	137	32	32	20
SUBSET18	121	121	123	136	40	40	14
SUBSET19	121	118	127	136	47	47	9
SUBSET20	117	119	120	135	43	43	11
TOTAL MISSINGS	2390	2442	2595	2730	823	824	381

Tabla 1. Datos faltantes por columna [50]

Tras la conversión de todas las variables categóricas de la base de datos a numéricas, se llevó a cabo un estudio descriptivo de cada columna empleando la librería pandas de Python. A continuación, se presenta un fragmento del código utilizado:

```python
import pandas as pd
from sklearn.preprocessing import LabelEncoder

# Leer el archivo CSV
file_path = 'db_proyecto_aplicado.csv'
df = pd.read_csv(file_path)

# Inicializar LabelEncoder
le = LabelEncoder()

# Convertir todas las columnas categóricas a numéricas
for column in df.columns:
    if df[column].dtype == 'object':
        df[column] = le.fit_transform(df[column].astype(str))

# Obtener la descripción estadística para cada columna
for column in df.columns:
    print(f"\nDescripción estadística de la columna: {column}")
    print(df[column].describe())
```

A partir de la descripción estadística previa, se identificaron varios puntos clave respecto a la distribución de las variables. En el siguiente apartado, se presentan algunas conclusiones sobre qué variables podrían ser excluidas del análisis y las razones detrás de esta decisión.

7.2.1.2.Variables con Baja Variabilidad.

Las variables que presentan una desviación estándar (`std`) muy baja, y que tienen la mayor parte de sus valores concentrados en un solo valor, pueden no aportar información significativa al análisis. Un ejemplo de estas variables incluye:

Variable	*Mean*	*Std*	*Observación*	*Conclusión*
cuantoshijos	0.001131	0.020178	La mayoría de los valores están en 0	Baja variabilidad, podría ser excluida

Tabla 2. Análisis descriptivo de variable cuantoshijos [50]

7.2.1.3.Variables con Distribución Sesgada.

Las variables en las que la mayoría de los valores se encuentran en un extremo (0 o 1) pueden indicar un sesgo significativo que podría afectar la validez del análisis. Ejemplos de tales variables incluyen:

Variable	*Mean*	*Std*	*Observación*	*Conclusión*
discapauditiva	0.005384	0.073192	La mayoría de los valores están en 0	Alta concentración en un solo valor, podría ser excluida debido a que existe en la base de datos la variable “Discapacidad” que reúne todos los tipos de discapacidad.
discapintelect	0.009332	0.096170	La mayoría de los valores están en 0	Alta concentración en un solo valor, podría ser excluida debido a que existe en la base de datos la variable “Discapacidad” que reúne todos los tipos de discapacidad.
vivehijos	0.010409	0.101511	La mayoría de los valores están en 0	Alta concentración en un solo valor, podría ser excluida

Tabla 3. Análisis descriptivo de variables con distribución sesgada [50]

7.2.1.4.Resumen de Variables a Excluir.

El resumen de variables a excluir destaca aquellas que, por baja variabilidad o distribución sesgada, no contribuyen significativamente al análisis y, por ende, se recomienda su exclusión del conjunto de datos, así:

Tipo de Variable	*Variables*
Baja Variabilidad	cuantoshijos, discapauditiva, discapintelect, vivehijos

Tabla 4. Variables a excluir [50]

7.2.2. Análisis de Variables Desechadas por Alta Multicolinealidad.

En el contexto de la ciencia de datos, la identificación y eliminación de variables con alta multicolinealidad es un paso crítico para asegurar la integridad y eficacia de los modelos predictivos. La multicolinealidad se refiere a la situación en la que una variable explicativa en un modelo de regresión está altamente correlacionada con una o más de las otras variables explicativas. Para evaluar la multicolinealidad, se utiliza el *Variance Inflation Factor* (VIF), que mide cuánto aumenta la varianza de un estimador debido a la multicolinealidad.

El VIF emerge como una herramienta clave para identificar la presencia y severidad de multicolinealidad entre las variables predictoras. Según la literatura especializada [48], un VIF superior a 5 indica la presencia de una moderada multicolinealidad, mientras que un VIF superior a 10 sugiere una alta multicolinealidad, lo que justifica la consideración de eliminar o transformar las variables afectadas.

A continuación, se presenta un script detallado para la aplicación del VIF y la detección efectiva de multicolinealidad, proporcionando una guía práctica para la identificación y mitigación de este fenómeno en el análisis de los datos.

```
# Identificar las columnas categóricas
categorical_cols = df.select_dtypes(include=['object']).columns

# Convertir columnas categóricas a numéricas si es necesario
label_encoders = {}
for col in categorical_cols:
    le = LabelEncoder()
    df[col] = le.fit_transform(df[col].astype(str))
    label_encoders[col] = le

# Calcular VIF para detectar multicolinealidad
def calculate_vif(df):
    vif_data = pd.DataFrame()
    vif_data["feature"] = df.columns
    vif_data["VIF"] = [variance_inflation_factor(df.values, i) for i in range(len(df.columns))]
    return vif_data

# Seleccionar solo las columnas numéricas
df_numeric = df.select_dtypes(include=[np.number])

# Calcular VIF
vif_data = calculate_vif(df_numeric)

# Mostrar el resultado del VIF
print("VIF para cada variable:")
print(vif_data)

# Determinar si se deben desechar variables basándose en el VIF
threshold = 10  # Un VIF mayor a 10 indica alta multicolinealidad
variables_to_drop = vif_data[vif_data['VIF'] > threshold]['feature']
print("\nVariables a considerar para desechar debido a alta multicolinealidad (VIF > 10):")
print(variables_to_drop)
```

Este código lleva a cabo una serie de operaciones destinadas a la detección de multicolinealidad dentro de un conjunto de datos. A continuación, se detalla el propósito de cada sección del código, delineando las funciones específicas que cada una cumple en el proceso de identificación de la multicolinealidad.

Paso	***Código***	***Propósito***
1. Identificar las columnas categóricas	python categorical_cols = df.select_dtypes(include=['object']).columns	Seleccionar las columnas del DataFrame que contienen datos categóricos.
2. Convertir columnas categóricas a numéricas si es necesario	python label_encoders = {} for col in categorical_cols: le = LabelEncoder() df[col] = le.fit_transform(df[col].astype(str)) label_encoders[col] = le	Convertir las columnas categóricas a datos numéricos utilizando LabelEncoder, necesario para calcular el VIF.
3. Calcular el VIF para detectar multicolinealidad	python def calculate_vif(df): vif_data = pd.DataFrame() vif_data["feature"] = df.columns vif_data["VIF"] = [variance_inflation_factor(df.values, i) for i in range(len(df.columns))] return vif_data	Calcular el Factor de Inflación de la Varianza (VIF) para cada columna en el DataFrame y devolver un DataFrame con las características y sus respectivos VIF.
4. Seleccionar solo las columnas numéricas	python df_numeric = df.select_dtypes(include=[np.number])	Filtrar el DataFrame original para que solo contenga columnas con datos numéricos, ya que el VIF solo se puede calcular para variables numéricas.
5. Calcular el VIF	python vif_data = calculate_vif(df_numeric)	Calcular el VIF para las columnas numéricas seleccionadas del DataFrame.
6. Mostrar el resultado del VIF	python print("VIF para cada variable:") print(vif_data)	Imprimir los resultados del cálculo del VIF, mostrando el DataFrame con las características y sus respectivos VIF.
7. Determinar si se deben desechar variables basándose en el VIF	python threshold = 10 variables_to_drop = vif_data[vif_data['VIF'] > threshold]['feature'] print("\nVariables a considerar para desechar debido a alta multicolinealidad (VIF > 10):") print(variables_to_drop)	Identificar y listar las variables que tienen un VIF superior a un umbral definido (en este caso, 10), lo cual indica alta multicolinealidad. Estas variables se pueden considerar para ser eliminadas del modelo para reducir la multicolinealidad.

Tabla 5. Identificación de Multicolinealidad [49]

El procedimiento inicial del código consiste en la conversión de las variables categóricas a numéricas, seguido por el cálculo del Factor de Inflación de la Varianza (VIF) para todas las variables numéricas presentes en el DataFrame. Posteriormente, se procede a identificar las variables que exhiben una alta multicolinealidad, con el propósito de considerar su exclusión.

En el curso del análisis realizado, se ha observado que las variables enlistadas a continuación presentan un VIF superior a 10, lo cual sugiere una alta multicolinealidad y fundamenta su exclusión de los modelos:

Variable	*Justificación*
genero	La alta multicolinealidad sugiere que esta variable tiene una fuerte relación con la variable género en el conjunto de datos, haciendo que su inclusión no aporte valor adicional y complique la interpretación del modelo.
cuantoshijos	Similar a las anteriores, esta variable presenta una alta multicolinealidad con la variable 'tienehijos', la cual se agregó al análisis..

Tabla 6. Variables con alta multicolinealidad [49]

7.2.3. Análisis de Variables Desechadas.

La eliminación de variables con alta multicolinealidad es esencial por varias razones. En primer lugar, mejora la estabilidad del modelo, ya que al reducir la colinealidad entre variables, se obtienen coeficientes más estables y significativos, lo que fortalece la robustez del análisis. En segundo lugar, clarifica la interpretación del modelo. Sin la interferencia de variables redundantes, es más sencillo interpretar el efecto de cada variable explicativa sobre la variable dependiente, lo que facilita la comprensión del modelo.

Además, un modelo libre de multicolinealidad severa tiende a ser más preciso en sus predicciones, mejorando su desempeño y fiabilidad en nuevos datos. La eliminación de variables con alta multicolinealidad, como las mencionadas, es una práctica recomendada en el análisis de datos y modelado predictivo para asegurar la fiabilidad y efectividad del modelo.

Así mismo, como parte del análisis exploratorio de datos, se empleó el coeficiente de correlación de Pearson para evaluar la relación lineal entre las variables predictoras y la variable objetivo. El coeficiente de correlación de Pearson proporciona información sobre la fuerza y la dirección de la relación entre dos variables, lo que resulta crucial para entender cómo influyen las características en las variables que se buscan predecir.

Roy, Rivas, Pérez & Palacios precisan que “el coeficiente de correlación de Pearson fue introducido por Galton en 1877 y desarrollado más adelante por Pearson. Es un indicador usado para describir cuantitativamente la fuerza y dirección de la relación entre dos variables cuantitativas de distribución normal y ayuda a determinar la tendencia de dos variables a ir juntas, a lo que también se denomina covarianza” [41].

Mediante esta medida, se identificaron aquellas variables que están más estrechamente relacionadas con los objetivos y que, por lo tanto, pueden tener un mayor impacto en la precisión de los modelos predictivos. Este análisis permitió seleccionar características relevantes y construir modelos más robustos y precisos para las predicciones.

Durante la investigación, también se empleó la correlación de distancia como una medida alternativa para comprender las relaciones entre variables. Este enfoque permite evaluar las relaciones entre variables de una manera diferente, proporcionando información adicional sobre la estructura de los datos.

Como señala Bengfort, la correlación de distancia ofrece una perspectiva única al reemplazar los conceptos tradicionales de covarianza y desviación estándar con medidas de distancia, lo que puede ser especialmente útil en contextos donde los datos no se distribuyen normalmente y pueden contener valores atípicos [42]. Esta perspectiva alternativa permite explorar las relaciones entre variables desde una nueva óptica, complementando así el análisis basado en el coeficiente de correlación de Pearson.

Así mismo, se introdujo el Coeficiente de Información Máxima (MIC) como una medida de dependencia entre dos variables que detecta relaciones no lineales y no monótonas entre ellas. Según Reshef, el término MIC es una medida de dependencia entre dos variables aleatorias que generaliza las correlaciones de Pearson [43]. El coeficiente proporciona una medida de la cantidad de información mutua entre dos variables, capturando incluso relaciones complejas que no pueden ser capturadas por otras medidas de correlación más tradicionales. Su inclusión en el trabajo mejorará la capacidad de los modelos para capturar relaciones subyacentes y aumentar la precisión de las predicciones.

A continuación, se presentará un script de código destinado a detectar la multicolinealidad en el conjunto de datos. Este script emplea tanto la correlación de Pearson como el MIC (Máxima Información Coeficiente) para identificar las variables más correlacionadas con la variable objetivo. La implementación de este código permite una evaluación precisa de las relaciones entre las variables.

```python
# Set the threshold for Pearson correlation
pearson_threshold = 0.45

for target_variable in target_variables:
    print(f"\nRunning for target variable: {target_variable}")

    # Calculate correlations
    correlations = []

    for column_name in df.columns:
        if column_name not in target_variables:  # Ensure target variable is excluded
            mine = MINE()
            mine.compute_score(df[column_name], df[target_variable])
            mic = mine.mic()
            pearson_corr, _ = pearsonr(df[column_name], df[target_variable])

            if pearson_corr >= pearson_threshold:
                correlations.append((column_name, mic, pearson_corr))

    # Sort correlations by Pearson correlation
    correlations.sort(key=lambda x: x[2], reverse=True)

    # Add top correlated features based on MIC and pearson
    for column_name in df.columns:
        if column_name not in target_variables and column_name not in [corr[0] for corr in correlations[:10]]:  # Exclude already added features
            mine = MINE()
            mine.compute_score(df[column_name], df[target_variable])
            mic = mine.mic()
            pearson_corr, _ = pearsonr(df[column_name], df[target_variable])

            if mic >= 0.1 and pearson_corr < pearson_threshold:
                correlations.append((column_name, mic, pearson_corr))

    top_related_features = [correlation[0] for correlation in correlations[:10]]

    print("Top 10 related features:")
```

Este código calcula las correlaciones entre las variables de un DataFrame y una o más variables objetivo (`target_variables`), utilizando tanto el coeficiente de correlación de Pearson como la Máxima Información Colectiva (MIC). Posteriormente, selecciona y ordena las características más correlacionadas basándose en estos criterios. A continuación, se detalla el propósito de cada sección del código:

Paso	***Código***	***Propósito***
1. Establecer el umbral para la correlación de Pearson	python pearson_threshold = 0.45	Definir un umbral mínimo de correlación de Pearson para considerar una característica como relevante.
2. Iterar sobre cada variable objetivo	python for target_variable in target_variables: print(f"\nRunning for target variable: {target_variable}")	Iterar sobre cada variable en target_variables y realizar el cálculo de correlaciones para cada una de ellas.
3. Calcular las correlaciones	python correlations = [] for column_name in df.columns: if column_name not in target_variables: mine = MINE() mine.compute_score(df[column_name], df[target_variable]) mic = mine.mic() pearson_corr, _ = pearsonr(df[column_name], df[target_variable]) if pearson_corr >= pearson_threshold: correlations.append((column_name, mic, pearson_corr))	Calcular el coeficiente de correlación de Pearson y la MIC entre cada característica y la variable objetivo, y almacenar los resultados en la lista correlations si el coeficiente de Pearson supera el umbral definido.
4. Ordenar las correlaciones por el coeficiente de Pearson	python correlations.sort(key=lambda x: x[2], reverse=True)	Ordenar las correlaciones en orden descendente basado en el coeficiente de correlación de Pearson.
5. Añadir las características más correlacionadas basadas en MIC y Pearson	python for column_name in df.columns: if column_name not in target_variables and column_name not in [corr[0] for corr in correlations[:10]]: mine = MINE() mine.compute_score(df[column_name], df[target_variable]) mic = mine.mic() pearson_corr, _ = pearsonr(df[column_name], df[target_variable]) if mic >= 0.1 and pearson_corr < pearson_threshold: correlations.append((column_name, mic, pearson_corr)) top_related_features = [correlation[0] for correlation in correlations[:10]]	Añadir las características más correlacionadas basadas en la MIC y el coeficiente de Pearson. Excluir las características ya añadidas en la lista de correlaciones top 10. Se considera la MIC mayor o igual a 0.1 y el coeficiente de Pearson mayor al umbral definido.
6. Imprimir las 10 características más relacionadas	print("Top 10 related features:")	Se imprimen las características que tienen las correlaciones más altas con la variable objetivo.

Tabla 7. Identificación y selección de correlaciones [49]

Con base en lo anterior, la configuración del umbral establece un límite mínimo para considerar la relevancia de las correlaciones de Pearson. Luego, se calcula la correlación iterando a través de todas las características, excluyendo las variables objetivo, y calculando tanto la Máxima Información Colectiva (MIC) como el coeficiente de correlación de Pearson. Se guardan las características que cumplen con el umbral de Pearson. Posteriormente, se ordenan las características por su valor de correlación de Pearson en orden descendente.

En una selección adicional, se añaden características basadas en la MIC que no se hayan incluido ya en el top 10 de correlaciones de Pearson. Finalmente, se imprime el resultado mostrando las características más relevantes. Este enfoque permite identificar y priorizar las características más importantes para cada variable objetivo, basándose en múltiples métricas de correlación.

Guiados por los métodos descritos anteriormente, se ha identificado un conjunto de columnas que pueden ser eliminadas del estudio. A continuación, se presenta una tabla detallada con cada una de estas columnas:

Columna	***Observaciones***
'fpsapoyosoc1','fpsapoyosoc2','fpsapoyosoc3','punapoyosoc'	Hacen parte de la variable global APOYOSOC
'fpsfuncfliar2','fpsfuncfliar3','fpsfuncfliar4'	Hacen parte de la variable global FUNCFLIAR
'rpsoptimis', 'rpsinterotros','rpsresolprob','rpssentbien','rpssentcercan','rpssentseguro','rpstomadecis','rpssentquerido','rpsaparfisica'	Hacen parte de la variable global NIVRECPSIC
'punrecpsic'	Hace parte de la variable global NIVRECPSIC
'punresil'	Conforma la variable global NIVRESIL
'punsatvida'	Conforma la variable global NIVSATVIDA
'punbienpsico'	la base de datos posee una columna que explica la misma variable y además tiene mayor correlación con variables objetivo, la cual posee el nombre 'biepsico'
'punbienfis'	la base de datos posee una columna que explica la misma variable, la cual se incluyó al análisis 'biefisico'
'punbiensoc'	la base de datos posee una columna que explica la misma variable, la cual se incluyó al análisis 'biesoc'
'punbienespir'	la base de datos posee una columna que explica la misma variable, la cual se incluyó al análisis 'bienespir'
'punans'	Variable global nivans1 la cual se incluyó al análisis.
'punsueno'	Existe otra columna que explica la misma variable, la cual se incluyó al análisis nivsueno

'pundep'	Existe otra columna que explica la misma variable, llamada nivdep1. Esta, representa la variable Depresión, la cual es un determinante que este proyecto busca predecir.
'punest'	Existe otra columna que explica la misma variable, llamada nivest1. Esta, representa la variable Estrés, la cual es un determinante que este proyecto busca predecir.
'nivans1'	Existe otra columna que explica la misma variable, llamada nivans, el cual es un determinante que este proyecto busca predecir.
'nivdep1'	Existe otra columna que explica la misma variable, llamada nivdep, el cual es un determinante que este proyecto busca predecir.
'nivest1'	Existe otra columna que explica la misma variable, llamada nivest, el cual es un determinante que este proyecto busca predecir.
'enfermedadcual'	Más del 85% de la columna está vacía
'condpsiqucual'	Más del 87% de la columna está vacía
'dolorcual'	Más del 93% de la columna está vacía
'spailegales'	El 98% de la columna está vacía
'genero'	La alta multicolinealidad con la variable generoeco, la cual fue incluida en el análisis, sugiere que esta variable tiene una fuerte relación con otras variables explicativas en el conjunto de datos, haciendo que su inclusión no aporte valor adicional y complique la interpretación del modelo.
'cuantoshijos'	La alta multicolinealidad sugiere que esta variable tiene una fuerte relación con la variable 'tienehijos' en el conjunto de datos, haciendo que su inclusión no aporte valor adicional y complique la interpretación del modelo.
'discapauditiva'	Más del 98% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado. A su vez, en el análisis se incluye la variable Discapacidad, la cual engloba esta columna.
'discapintelect'	Más del 99% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado. A su vez, en el análisis se incluye la variable Discapacidad, la cual engloba esta columna.
discapotra	Más del 98% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado. A su vez, en el análisis se incluye la variable Discapacidad, la cual engloba esta columna.
discapfisica	Más del 97% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado. A su vez, en el análisis se incluye la variable Discapacidad, la cual engloba esta columna.

'vivehijos'	Más del 98% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
percsaludreco	Cerca del 89% de los datos pertenecen a un único valor lo que puede influenciar en una análisis sesgado.
dolor	Más del 98% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
tocioentretsolit	Más del 94% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
vivesolo	El 92% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
vivepareja	Más del 96% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
viveprimos	Más del 95% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
viveabuelo	Más del 86% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
vivesobrinos	Más del 98% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
vivefamiliares	El 93% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
viveconocidos	Más del 97% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
vivecompaneros	Cerca del 91% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
viveotro	El 94% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
cvsssplancompl	Más del 91% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
cvsssotro	Más del 97% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
cvsssningun	Más del 99% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
cvsssnosabe	Más del 98% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado
tipoestudiante	El 100% de la variable contiene un único valor lo cual puede influenciar un análisis sesgado
dobletitul	Más del 91% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado

horasclasesmes	Más del 99% de la columna posee un único valor. Lo cual puede influenciar un análisis sesgado

Tabla 8. Columnas identificadas para eliminación [49]

El conjunto de datos inicial consistía en 222 columnas y 2.768 filas. Según la tabla anterior, se eliminaron un total de 60 columnas de este análisis. Por lo tanto, quedaron 162 columnas y 2.786 filas después del proceso de eliminación.

La eliminación de estas columnas se llevó a cabo utilizando la función drop() de la biblioteca Pandas. A continuación se muestra el resultado, donde la función head() proporciona información sobre las primeras 5 filas del conjunto de datos:

```
data = data.drop(columns=columns_to_drop)
data.head()
```

	nivsoled	nivresil	nivsatvida	nivrecpsic	percsalud	peso	estatura
0	normal	media	Media	Alto	Buena	90	170
1	severo	media	Media	Bajo	Muy buena	74	171
2	severo	media	Alta	Medio	Muy buena	68	160
3	normal	media	Alta	Alto	Muy buena	76	187
4	severo	baja	Media	Bajo	Regular	65	0

7.2.4. Identificación de Valores Faltantes y Atípicos.

Para llevar a cabo la identificación de datos faltantes y atípicos por columna en nuestro conjunto de datos, se implementó una transformación de los datos categóricos o de tipo texto a formatos numéricos. Este proceso se realizó mediante el siguiente código:

```
import pandas as pd
from sklearn.preprocessing import LabelEncoder

# Leer el archivo CSV
file_path = 'processed_data.csv'
df = pd.read_csv(file_path)

# Inicializar LabelEncoder
le = LabelEncoder()

# Convertir todas las columnas categóricas a numéricas
for column in df.columns:
    if df[column].dtype == 'object':
        df[column] = le.fit_transform(df[column].astype(str))
```

El script previamente mencionado realiza varias operaciones secuenciales para preparar y analizar el conjunto de datos. Comienza importando las bibliotecas necesarias, incluyendo pandas con el alias pd para la manipulación de datos, y LabelEncoder de sklearn.preprocessing para la conversión de datos categóricos a numéricos.

Luego, lee un archivo CSV llamado processed_data.csv y carga sus contenidos en un DataFrame de pandas denominado df. Posteriormente, inicializa una instancia de LabelEncoder llamada le, que será utilizada para convertir etiquetas categóricas en números.

El siguiente paso consiste en iterar sobre todas las columnas del DataFrame df. Si una columna contiene datos categóricos o de texto (identificados por el tipo de datos object), se convierte en una representación numérica utilizando el método fit_transform de LabelEncoder. Este método ajusta el codificador a la columna, identificando todas las categorías únicas y asignándoles números enteros correspondientes.

Es importante resaltar que se emplea el método astype(str) para garantizar que los datos en la columna sean tratados como cadenas de texto, lo cual es esencial si la columna contiene datos mixtos.

Después de este proceso, todas las columnas categóricas del DataFrame df se convierten en columnas numéricas, lo cual es fundamental para muchos algoritmos de aprendizaje automático que requieren datos en este formato. Finalmente, se procede a calcular la suma de valores faltantes

```
import numpy as np
import seaborn as sns

# Identificar valores faltantes
missing_values = data.isnull().sum()
missing_values = missing_values[missing_values > 0]
print("Valores faltantes por columna:")
print(missing_values)

# Identificar valores atípicos utilizando el método del rango intercuartílico (IQR)
def find_outliers(df):
    outliers = pd.DataFrame()
    for column in df.select_dtypes(include=[np.number]).columns:
        Q1 = df[column].quantile(0.25)
        Q3 = df[column].quantile(0.75)
        IQR = Q3 - Q1
        lower_bound = Q1 - 1.5 * IQR
        upper_bound = Q3 + 1.5 * IQR
        outliers[column] = ((df[column] < lower_bound) | (df[column] > upper_bound))
    return outliers

outliers = find_outliers(data)
outliers_summary = outliers.sum()
print("Valores atípicos por columna:")
print(outliers_summary)

# Visualización de valores atípicos utilizando boxplots
numeric_columns = data.select_dtypes(include=[np.number]).columns
for column in numeric_columns:
    plt.figure(figsize=(10, 6))
    sns.boxplot(x=data[column])
    plt.title(f'Boxplot de {column}')
    plt.show()
```

en cada columna e imprimirlos, así como a visualizar los datos atípicos de cada columna utilizando las librerías numpy y seaborn.

Una vez identificado los valores faltantes en cada columna del conjunto de datos utilizando el método isnull(), se calcula su suma con el método sum(). Luego, se imprime esta información para ofrecer una visión clara de la cantidad de datos faltantes en cada columna.

Se utiliza el método del rango intercuartílico (IQR) para detectar valores atípicos en las columnas numéricas del conjunto de datos. Los valores que se sitúan fuera de los límites establecidos por [Q1 - 1.5 * IQR, Q3 + 1.5 * IQR] se consideran atípicos. La salida de este proceso es una tabla que muestra la cantidad de valores atípicos en cada columna.

De este modo, se emplean diagramas de caja (boxplots) para visualizar la distribución de los datos en cada columna numérica. Esta representación gráfica facilita la identificación visual de los valores atípicos en el conjunto de datos.

Ahora, se presenta un ejemplo del resultado obtenido:

```
Valores faltantes por columna:
nivsoled       6
nivresil      17
nivsatvida    17
nivrecpsic    45
percsalud     27
```

```
Valores atípicos por columna:
peso             165
estatura         334
horassuesem      315
horassuefinde    311
punabusotics      56
punbienpsico       0
punbienambie      35
edad             232
estratosoc         0
punfuncfliar      34
semestre           0
ica              479
horasclasesem      0
creditos         465
asignaturas      147
horastrabaja     456
```

7.2.5. Visualización de Valores Atípicos Usando Boxplots.

A continuación, se presentan algunos diagramas de caja (boxplots) correspondientes a varias columnas del conjunto de datos, el cual cuenta con un total de 222 columnas. Estos boxplots permiten identificar visualmente los valores atípicos presentes en dichas columnas, proporcionando una representación clara de la distribución de los datos y facilitando la detección de posibles anomalías.

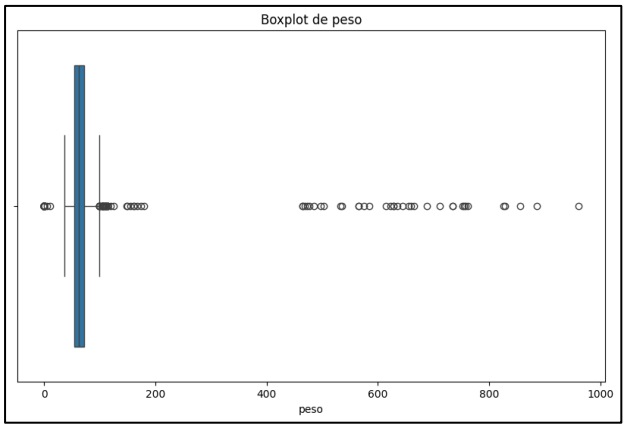

Figura 4. Boxplot de peso [50]

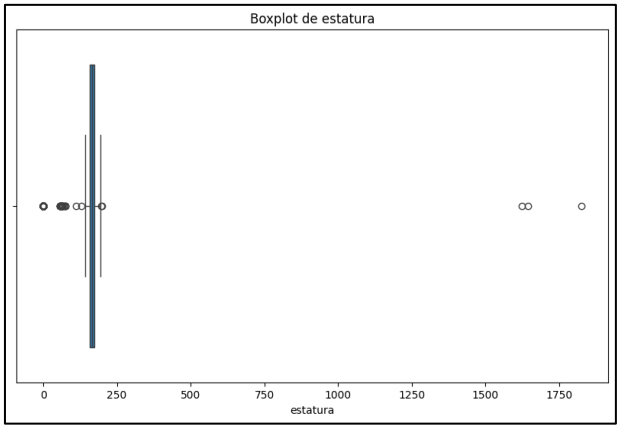

Figura 3. Boxplot de estatura [50]

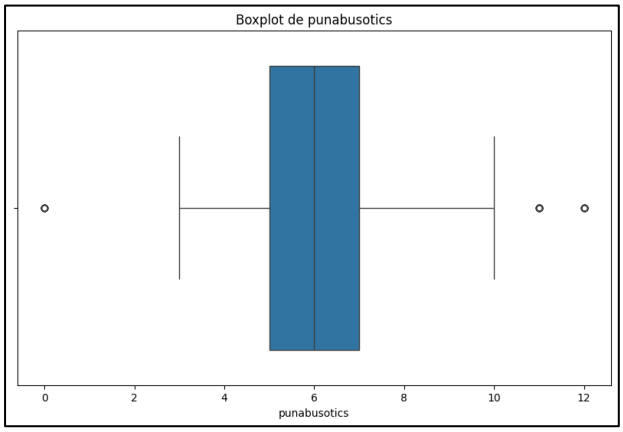

Figura 1. Bloxpot de punabusotics [50]

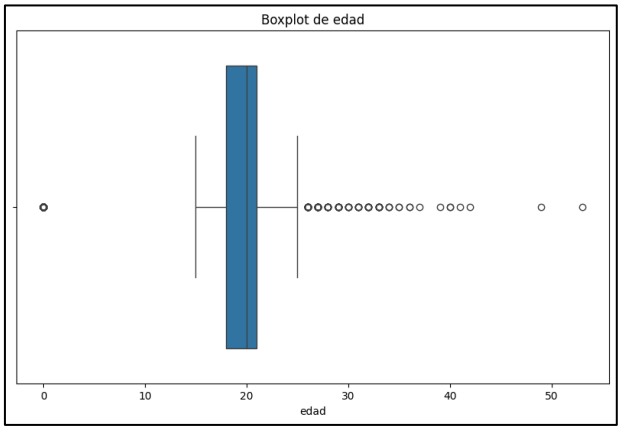

Figura 2. Bloxpot de edad [50]

En las figuras adjuntas, podemos observar cómo estos valores atípicos se dispersan fuera de los límites esperados de los datos, destacando puntos individuales que se encuentran significativamente alejados del rango intercuartílico. Esta visualización es esencial para comprender mejor la naturaleza de los datos y decidir sobre las técnicas de tratamiento adecuadas para los valores atípicos.

7.2.6. Manejo de Datos Atípicos y Faltantes.

Para este apartado, se llevó a cabo la imputación de valores faltantes utilizando la media para las columnas numéricas, aplicando la función SimpleImputer. Para el manejo de valores atípicos, se empleó el método del rango intercuartílico (IQR), identificando y recortando estos valores en las columnas numéricas.

Los valores que se encontraban fuera de los límites establecidos por [Q1 - 1.5 * IQR, Q3 + 1.5 * IQR] fueron ajustados a estos límites. Posteriormente, se imprimieron los valores faltantes y atípicos tras la imputación y el recorte, garantizando así que se habían manejado adecuadamente.

Todo esto se realizó mediante una serie de pasos estructurados para asegurar la integridad y la calidad del conjunto de datos, así:

7.2.6.1. Imputación de Valores Faltantes.

Para llevar a cabo este primer paso, se empleó el siguiente código:

```
# Manejar valores faltantes
# Usar la media para imputar valores faltantes en columnas numéricas
imputer = SimpleImputer(strategy='mean')
data[data.select_dtypes(include=[np.number]).columns] = imputer.fit_transform(data.select_dtypes(include=[np.number]))
```

En este segmento de código, se emplea el método de imputación para llenar los valores faltantes en las columnas numéricas del DataFrame `data`. Se utiliza `SimpleImputer` con la estrategia de imputar los valores faltantes mediante la media de cada columna. Específicamente, se crea una instancia de `SimpleImputer` configurada para usar la media como estrategia de imputación. A continuación, se seleccionan exclusivamente las columnas numéricas del DataFrame utilizando `data.select_dtypes(include=[np.number]).columns`. Finalmente, mediante `imputer.fit_transform(...)`, se ajusta el imputador a los datos, calculando la media de cada columna, y se transforman los datos reemplazando los valores faltantes con las medias correspondientes.

7.2.6.2. Identificación y Manejo de Valores Atípicos.

En el segundo paso, se procede a la identificación y manejo de valores atípicos en las columnas numéricas del DataFrame mediante el método del rango intercuartílico (IQR). Para ello, se utiliza el siguiente código:

```python
# Identificar y manejar valores atípicos utilizando el método del rango intercuartílico (IQR)
def cap_outliers(df):
    for column in df.select_dtypes(include=[np.number]).columns:
        Q1 = df[column].quantile(0.25)
        Q3 = df[column].quantile(0.75)
        IQR = Q3 - Q1
        lower_bound = Q1 - 1.5 * IQR
        upper_bound = Q3 + 1.5 * IQR
        df[column] = np.where(df[column] < lower_bound, lower_bound, df[column])
        df[column] = np.where(df[column] > upper_bound, upper_bound, df[column])
    return df

data = cap_outliers(data)
```

Primero, se define una función denominada 'cap_outliers', la cual ajusta los valores atípicos en el DataFrame 'df'. Dentro de esta función, se calculan el primer cuartil (Q1) y el tercer cuartil (Q3) de cada columna numérica utilizando 'df[column].quantile(0.25)' y 'df[column].quantile(0.75)' respectivamente. Luego, se determina el rango intercuartílico (IQR) con la fórmula 'IQR = Q3 - Q1'.

Los límites inferior y superior para identificar valores atípicos se definen como 'lower_bound = Q1 - 1.5 * IQR' y 'upper_bound = Q3 + 1.5 * IQR'. Los valores que se encuentran por debajo del límite inferior o por encima del límite superior se ajustan utilizando 'np.where(df[column] < lower_bound, lower_bound, df[column])' y 'np.where(df[column] > upper_bound, upper_bound, df[column])', reemplazándolos con los límites correspondientes.

Finalmente, la función 'cap_outliers' se aplica al DataFrame 'data' mediante la instrucción 'data = cap_outliers(data)'.

7.2.6.3. Verificación de Valores Faltantes.

En el tercer paso, se lleva a cabo la verificación de valores faltantes en el DataFrame 'data' tras la imputación. Para ello, se utiliza el siguiente código:

```python
# Verificar que no haya más valores faltantes ni atípicos
missing_values = data.isnull().sum()
print("Valores faltantes por columna después de la imputación:")
print(missing_values)
```

Este bloque de código comienza contando los valores faltantes en cada columna del DataFrame mediante 'data.isnull().sum()'. Los resultados de esta operación, que indican la cantidad de valores faltantes por columna, se almacenan en la variable 'missing_values'. Posteriormente, se imprime el contenido de 'missing_values' utilizando 'print(missing_values)' para confirmar que no quedan valores faltantes en el DataFrame tras la imputación realizada previamente.

7.2.6.4.Verificación de Valores Atípicos.

En el cuarto paso, se realiza una verificación para asegurar que los valores atípicos han sido manejados adecuadamente en el DataFrame 'data'. Este bloque de código se ejecuta de la siguiente manera:

```
# Verificar que los valores atípicos hayan sido manejados
outliers = find_outliers(data)
outliers_summary = outliers.sum()
print("Valores atípicos por columna después del recorte:")
print(outliers_summary)
```

Se llama a una función, presumiblemente definida en otra parte del código, 'find_outliers(data)', para identificar los valores atípicos presentes en el DataFrame. A continuación, se calcula la suma de los valores atípicos en cada columna utilizando 'outliers.sum()'.

Los resultados, que indican la cantidad de valores atípicos por columna tras el tratamiento, se almacenan en la variable 'outliers_summary'. Finalmente, se imprime el contenido de 'outliers_summary' mediante 'print(outliers_summary)', confirmando así que los valores atípicos han sido adecuadamente manejados.

Este conjunto de operaciones garantiza que los datos están limpios y listos para análisis o modelado posterior, libres de la influencia de valores faltantes o atípicos extremos.

7.3. ESCOGENCIA DE VARIABLES DEPENDIENTES

La selección de variables dependientes estuvo a cargo de los investigadores del proyecto, seleccionándose las siguientes: Ansiedad, Estrés, Depresión, Resiliencia, Recursos Psicológicos y Satisfacción con la vida.

En esta selección, se tuvo especial consideración en equilibrar las variables que reflejan aspectos positivos y negativos de la salud mental de los individuos. Así, se identificaron tres variables que pueden ser interpretadas como positivas (Resiliencia, Recursos Psicológicos, y Satisfacción con la Vida) y tres variables que podrían tener connotaciones negativas (Ansiedad, Estrés y Depresión). Este balance es crucial para obtener una visión equilibrada del bienestar psicológico de los participantes en el estudio.

La clasificación de estas variables no solo responde a un criterio académico, sino que también es el resultado de un consenso alcanzado con un cuerpo especializado de nuestra institución. La colaboración interdisciplinaria ha sido esencial para asegurar que las variables seleccionadas sean

representativas y pertinentes, garantizando así la relevancia y la aplicabilidad de los resultados obtenidos a partir del análisis de datos.

7.4. ESCOGENCIA DE VARIABLES INDEPENDIENTES

Las variables independientes seleccionadas del conjunto de datos con el apoyo de los investigadores del proyecto, correspondieron a las siguientes: nivsoled, percsalud, peso, estatura, imc, enfermedad, condpsiqu, discapacidad, discapvisual, nivactfis, tiemsed, horassuesem, horassuefinde, nivsueno, tociorelaj, tocioartist, tociomusic, tociomanual, tocioespirsolit, tocioespirgrup, tocioentretgrup, spaalcohol, spacigarrillo, spavapeo, spamarihuana, alimfrutas, alimverdur, alimembutid, alimpaquetes, alimcomidrapid, alimgaseos, alimdulces, alimcomidprepar, alimcafeteriau, alimcomertv, alimmaquinas, alimtiempo, alimhoras, alimcomerotros, alimdesayuno, alimrefrigmanana, alimalmuerzo, alimrefrigtarde, alimcena, alimdespucena, sexrelacsex, sexpreserv, sexsatisfsex, sexorientacsex, nivabusotics, biefisico, biepsico, biesoc, bienespir, bieambi, sexo, genero, edad, rangoedad, estratosoc, nivelsocioec, niveducativo, estadocivil, razaetnia, nacioen, nivedumadre, nivedupadre, residencia, residencia_valle, residencia_cauca, residencia_otro, zonaresidencia, nopersonasvivecon, tienehijos, vivepadre, vivemadre, vivehermanos, fpsafrontam1, fpsafrontam2, fpsafrontam3, fpsafrontam4, fpsafrontam5, apoyoso, funcfliar, fpsantecviol1, fpsantecviol2, fpsantecviol3, fpsantecviol4, fpsantecviol5, fpsantecviol6, cvservpub, cvinternet, cvzonasocial, cvcentrodepor, cvtransp, cvparques, cvcentrossalud, cvespaccomunit, cvsegurbarrio, cvviolenbarrio, cvinundac, cvruido, cvbasura, cvinvasespac, cvvias, cvtransvehicprop, cvtransvehicompar, cvtranspublicomas, cvtranspublicotax, cvtransbici, cvtranscamina, cvdecis, cvsust, cvdepecon, cvingresufic, cvingreshogar, cvssseps, cvsssmedprep, faculpre, programapre, semestre, beca, ica, nivica, horasclasesem, creditos, asignaturas, estudiatrabaja, horastrabaja, satisfacprograma, desempeno, conaccprog1, conaccprog2, conaccprog3, conaccprog4, conaccprog5, conaccprog6, conaccprog7, conaccprog8, conaccprog9, conaccprog10, conaccprog11, conaccprog12, conaccprog13, conaccprog14, conaccprog15, conaccprog16, conaccprog17, conaccprog18, conaccprog19, ambalim1, ambalim2, ambalim3, ambalim4, ambalim5.

7.5. DISTRIBUCIÓN DE VARIABLES OBJETIVO

A continuación, se presentan las conclusiones derivadas del análisis de los gráficos de barras para las variables objetivo `nivrecpsic`, `nivans`, `nivdep`, `nivest`, `nivsatvida`, y `nivresil`:

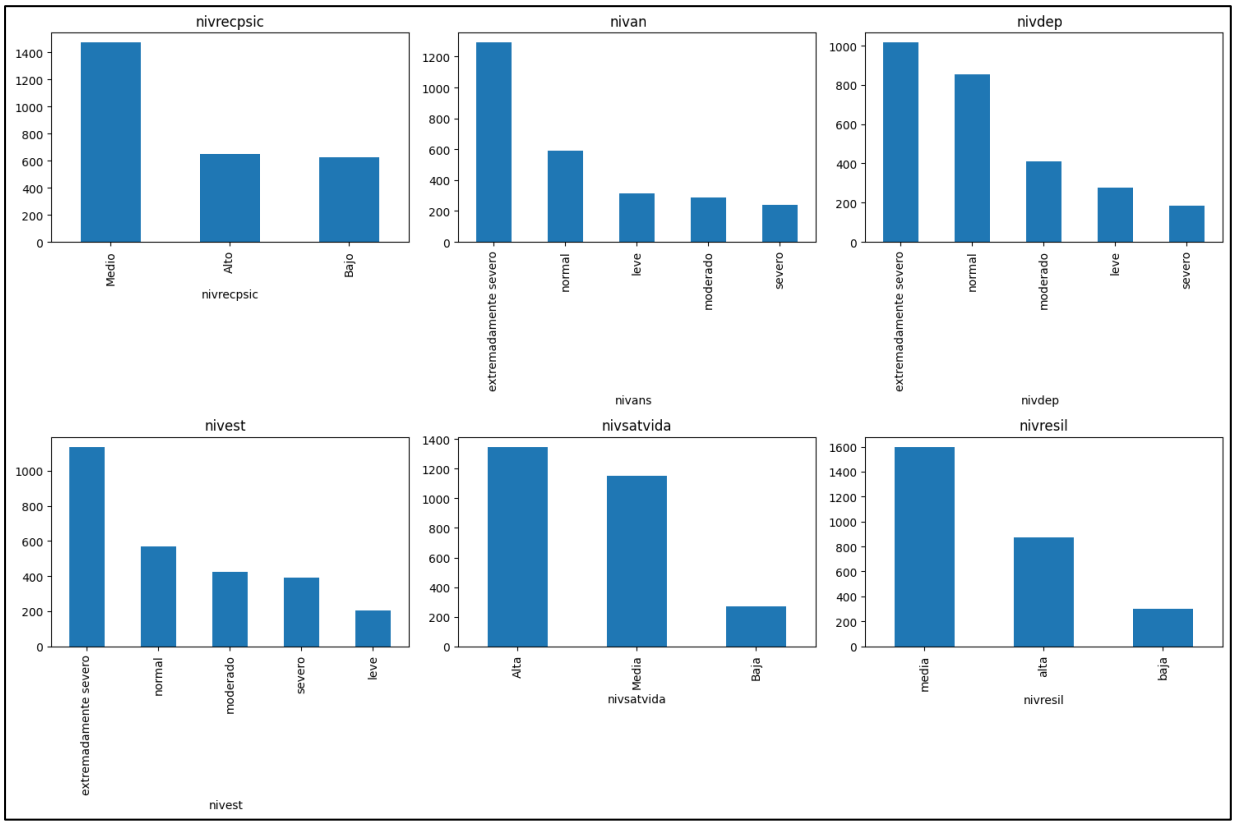

Figura 5. Gráficos de barras para las variables `nivrecpsic`, `nivans`, `nivdep`, `nivest`, `nivsatvida`, y `nivresil` [50]

Para la variable `nivrecpsic`, la mayoría de las personas muestran un nivel medio de recuperación psicológica, seguido por niveles altos y bajos. En cuanto a la variable `nivans`, se observa que la mayoría de los individuos se encuentran en un estado de ansiedad extremadamente severo, seguido de estados normales, leves, moderados y severos.

La distribución de la variable `nivdep` es similar a la de `nivans`, con la mayoría de las personas en un estado de depresión extremadamente severo, seguido de estados normales, moderados, leves y severos. Respecto a la variable `nivest`, nuevamente se identifica que la mayoría de los individuos están en un estado de estrés extremadamente severo, seguido por estados normales, moderados, severos y leves.

Con relación a la variable `nivsatvida`, la mayoría de las personas reportan una alta satisfacción con la vida, seguida por niveles medios y bajos. Finalmente, para la variable `nivresil`, la mayoría de los individuos presentan un nivel medio de resiliencia, seguido por niveles altos y bajos.

De manera general, se observa una tendencia significativa de altos niveles de ansiedad, depresión y estrés entre los individuos evaluados. Adicionalmente, la mayoría de las personas también reportan niveles medios tanto en recursos psicológicos como en resiliencia. En términos de satisfacción con la vida, se destaca que una mayor proporción de personas se sienten altamente satisfechas, aunque también existe un número considerable con niveles medios de satisfacción. Estas conclusiones proporcionan una visión integral del estado psicológico de los individuos en el estudio, resaltando áreas críticas para intervenciones futuras.

8. GENERACIÓN DE MODELOS DE PREDICCIÓN

La etapa de Generación de Modelos de Predicción constituye un momento crucial en el proceso de desarrollo de un proyecto de análisis predictivo, adentrándose en la generación de modelos de predicción que permite obtener *insights* valiosos y tomar decisiones fundamentadas. Se abordará el proceso de construcción, evaluación y selección de modelos predictivos, utilizando técnicas avanzadas de aprendizaje automático, análisis estadístico y algoritmos que puedan prever comportamientos o resultados futuros.

El objetivo es desarrollar modelos robustos y precisos, basados en los datos recopilados en las fases anteriores, que puedan predecir con fiabilidad los resultados deseados, contribuyendo así al avance de la investigación y a la consecución de los objetivos.

En este contexto, se destaca la implementación de técnicas de balanceo de datos y evaluación de modelos de clasificación utilizando Python y bibliotecas como pandas para la manipulación de datos y scikit-learn para el desarrollo y evaluación de modelos. Este enfoque integral permite no solo abordar el desequilibrio en los conjuntos de datos, sino también realizar una evaluación de la efectividad de los modelos construidos, lo que resulta fundamental para obtener resultados precisos y confiables en tareas de clasificación.

8.1. PREPARACIÓN Y EVALUACIÓN DE MODELOS

En este capítulo, se aborda la fase crucial de preparación y evaluación de modelos. Se detalla la importancia de la correcta importación de bibliotecas de Python y se describen las funciones clave utilizadas para llevar a cabo el balanceo de datos, la codificación de variables categóricas y la evaluación de los modelos de clasificación. Estos pasos son fundamentales para garantizar la fiabilidad, precisión y robustez de los modelos desarrollados, estableciendo así una base sólida para el éxito en aplicaciones de machine learning.

8.1.1. Importe de Bibliotecas.

En el proceso de preparación para la ejecución de los modelos, se llevó a cabo la importación de diversas bibliotecas de Python con el propósito de facilitar operaciones fundamentales. Entre estas operaciones se incluyeron el balanceo y división de datos, la codificación de variables categóricas, así como la realización de la validación cruzada para la evaluación de los modelos.

```
import pandas as pd
from sklearn.utils import resample
from sklearn.model_selection import train_test_split, cross_val_score
from sklearn.metrics import roc_curve, auc, confusion_matrix, accuracy_score, classification_report
from sklearn.svm import SVC
from sklearn.neural_network import MLPClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.preprocessing import LabelEncoder
```

Específicamente, se utilizó la biblioteca 'pandas' para la gestión y manipulación eficiente de los datos, mientras que el balanceo de datos se efectuó mediante la técnica de remuestreo, empleando la función 'resample'. Para la división de los datos en conjuntos de entrenamiento y prueba, así como para la validación cruzada con el fin de evaluar la eficacia de los modelos, se recurrió a las funciones 'train_test_split' y 'cross_val_score', respectivamente.

Además, se destacó el empleo de diversas métricas para la evaluación de los modelos generados, tales como 'roc_curve', 'auc', 'confusion_matrix', 'accuracy_score' y 'classification_report'. Estas métricas proporcionan una visión detallada del rendimiento de los modelos, permitiendo una evaluación adecuada de su capacidad predictiva y discriminativa. Asimismo, se hicieron uso de distintos algoritmos de clasificación, tales como 'SVC', 'MLPClassifier', 'KNeighborsClassifier' y 'DecisionTreeClassifier', para explorar y comparar diferentes enfoques en la resolución del problema.

Finalmente, para la codificación de variables categóricas, se empleó la función 'LabelEncoder', la cual facilitó la transformación de estas variables en formatos numéricos, requisito indispensable para el correcto funcionamiento de muchos algoritmos de machine learning.

8.1.2. Función para Balancear Datos.

```
def balance_data(df, target_var):
    classes = df[target_var].value_counts().index
    balanced_df = pd.DataFrame()

    for cls in classes:
        class_df = df[df[target_var] == cls]
        if len(class_df) < 1000:
            class_df = resample(class_df, replace=True, n_samples=1000, random_state=42)
        balanced_df = pd.concat([balanced_df, class_df])

    return balanced_df

balanced_data = data.copy()
```

Se implementó una función destinada a abordar el desbalanceo de clases dentro del conjunto de datos, con el objetivo primordial de contrarrestar posibles sesgos que podrían surgir en los modelos. La importancia de esta función radica en el hecho de que los modelos entrenados en conjuntos de datos desbalanceados tienden a favorecer las clases mayoritarias, lo que puede resultar en una disminución significativa en la precisión de la predicción para las clases minoritarias. En consecuencia, esta función se diseñó con el propósito de equilibrar la distribución de clases,

asegurando así que los modelos resultantes sean más justos y precisos en su capacidad predictiva.

8.1.3. Codificación de Variables Categóricas.

```
# Codificar columnas categóricas
le = LabelEncoder()
for column in balanced_data.columns:
    if balanced_data[column].dtype == 'object':
        balanced_data[column] = le.fit_transform(balanced_data[column].astype(str))
```

Se introdujo un proceso de codificación diseñado para transformar variables categóricas en valores numéricos. El propósito fundamental de este procedimiento es facilitar el manejo de las variables categóricas dentro del análisis de datos, convirtiéndolas en formatos numéricos que pueden ser interpretados por los algoritmos de análisis. La importancia de esta codificación radica en el hecho de que muchos algoritmos de análisis de datos y modelado no pueden manejar directamente variables categóricas en su forma original. Por lo tanto, este proceso de codificación es esencial para preparar adecuadamente los datos y permitir que los algoritmos de machine learning puedan realizar análisis efectivos y precisos.

8.1.4. Evaluación de modelos.

A continuación, se presenta una descripción detallada de las funciones evaluate_svm_model, evaluate_mlp_model, evaluate_knn_model y evaluate_decision_tree_model, las cuales siguen una estructura similar en su implementación. Estas funciones tienen como propósito principal entrenar y evaluar modelos de clasificación dentro del contexto del análisis de datos. Su importancia radica en la aplicación de una serie de procedimientos fundamentales para la evaluación del rendimiento de los modelos.

8.1.4.1. Función evaluate_svm_model:

```
# Evaluar el modelo de SVM
def evaluate_svm_model(X, y):
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

    model = SVC(probability=True)
    model.fit(X_train, y_train)

    cv_scores = cross_val_score(model, X_train, y_train, cv=5)

    y_pred = model.predict(X_test)
    y_proba = model.predict_proba(X_test)[:, 1]

    fpr, tpr, _ = roc_curve(y_test, y_proba, pos_label=model.classes_[1])
    roc_auc = auc(fpr, tpr)

    cm = confusion_matrix(y_test, y_pred)
    accuracy = accuracy_score(y_test, y_pred)
    report = classification_report(y_test, y_pred)

    return {
        'model': model,
        'cv_scores': cv_scores,
        'roc_curve': (fpr, tpr, roc_auc),
        'confusion_matrix': cm,
        'accuracy': accuracy,
        'classification_report': report
    }
```

Para este código, se utiliza la función train_test_split para dividir los datos en conjuntos de entrenamiento y prueba, permitiendo así una evaluación independiente del rendimiento del modelo. Luego, la función model.fit se emplea para entrenar el modelo con los datos de entrenamiento, ajustándolo a los patrones presentes en los datos. Posteriormente, la función cross_val_score se utiliza para realizar una validación cruzada, evaluando la estabilidad y generalización del modelo mediante la división iterativa de los datos en subconjuntos de entrenamiento y prueba.

Además, las funciones predict y predict_proba son utilizadas para generar predicciones sobre las etiquetas y las probabilidades asociadas a las mismas en los datos de prueba, respectivamente. Esto proporciona una evaluación más detallada del rendimiento del modelo en la clasificación de los datos de prueba. Por otra parte, las funciones roc_curve y auc son empleadas para calcular la curva ROC y el área bajo la misma, métricas cruciales para evaluar la capacidad de discriminación del modelo en la clasificación binaria.

Finalmente, las funciones confusion_matrix, accuracy_score y classification_report son utilizadas para proporcionar métricas detalladas sobre el rendimiento del modelo, incluyendo la matriz de confusión, la precisión global y un reporte detallado de las métricas de clasificación. Estas métricas son esenciales para comprender el rendimiento del modelo en diferentes aspectos de la clasificación, facilitando así la toma de decisiones informadas en el análisis de datos.

8.1.4.2. Función evaluate_mlp_model:

```python
# Evaluar el modelo de MLP
def evaluate_mlp_model(X, y):
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

    model = MLPClassifier(max_iter=1000)
    model.fit(X_train, y_train)

    cv_scores = cross_val_score(model, X_train, y_train, cv=5)

    y_pred = model.predict(X_test)
    y_proba = model.predict_proba(X_test)[:, 1]

    fpr, tpr, _ = roc_curve(y_test, y_proba, pos_label=model.classes_[1])
    roc_auc = auc(fpr, tpr)

    cm = confusion_matrix(y_test, y_pred)
    accuracy = accuracy_score(y_test, y_pred)
    report = classification_report(y_test, y_pred)

    return {
        'model': model,
        'cv_scores': cv_scores,
        'roc_curve': (fpr, tpr, roc_auc),
        'confusion_matrix': cm,
        'accuracy': accuracy,
        'classification_report': report
    }
```

El código proporcionado describe una función destinada a evaluar un modelo de aprendizaje automático mediante un clasificador de perceptrón multicapa (MLP). La función `evaluate_mlp_model` toma como parámetros los datos de entrada `x` y las etiquetas objetivo `y`. Primero, los datos y las etiquetas se dividen en conjuntos de entrenamiento y prueba, asignando el 20% de los datos para pruebas y el 80% restante para el entrenamiento, utilizando un estado

aleatorio de 42 para asegurar la reproducibilidad.

Luego, se instancia un clasificador MLP con un máximo de 1000 iteraciones, el cual se entrena utilizando los datos de entrenamiento. Para evaluar la estabilidad y eficacia del modelo, se realiza una validación cruzada con 5 particiones sobre los datos de entrenamiento. Posteriormente, el modelo entrenado se utiliza para predecir las etiquetas en el conjunto de prueba y para obtener las probabilidades de la clase positiva.

El rendimiento del clasificador se evalúa calculando la curva ROC y el área bajo dicha curva (AUC), lo que proporciona una medida de la calidad del modelo. Además, se genera una matriz de confusión para mostrar el número de predicciones correctas e incorrectas, y se calcula la precisión general del modelo.

Finalmente, se crea un informe de clasificación que incluye métricas clave como la precisión, la recuperación y el puntaje F1 para cada clase. La función retorna un diccionario con el modelo, los puntajes de validación cruzada, la curva ROC, la matriz de confusión, la precisión y el informe de clasificación, ofreciendo así una evaluación adecuada del desempeño del modelo.

8.1.4.3. Función evaluate_knn_model:

```python
# Evaluar el modelo de KNN
def evaluate_knn_model(X, y):
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

    model = KNeighborsClassifier()
    model.fit(X_train, y_train)

    cv_scores = cross_val_score(model, X_train, y_train, cv=5)

    y_pred = model.predict(X_test)
    y_proba = model.predict_proba(X_test)[:, 1]

    fpr, tpr, _ = roc_curve(y_test, y_proba, pos_label=model.classes_[1])
    roc_auc = auc(fpr, tpr)

    cm = confusion_matrix(y_test, y_pred)
    accuracy = accuracy_score(y_test, y_pred)
    report = classification_report(y_test, y_pred)

    return {
        'model': model,
        'cv_scores': cv_scores,
        'roc_curve': (fpr, tpr, roc_auc),
        'confusion_matrix': cm,
        'accuracy': accuracy,
        'classification_report': report
    }
```

El código anterior describe una función para evaluar un modelo de clasificación utilizando el algoritmo de los K Vecinos más Cercanos (KNN). A continuación, se detallan cada uno de los pasos y su importancia de manera comprensible.

En primer lugar, la función `evaluate_knn_model(X, y)` se define para evaluar el rendimiento de un modelo basado en el algoritmo KNN con un conjunto de datos específico, donde `X` representa las características (datos de entrada) y `y` las etiquetas (resultados esperados). Posteriormente, los datos se dividen en dos partes: una para entrenar el modelo (80%) y otra para probar su eficacia

(20%), garantizando mediante un "estado aleatorio" que esta división sea consistente en cada ejecución del código.

El siguiente paso implica la creación y entrenamiento del modelo KNN. Se instancia el modelo con `KNeighborsClassifier()` y se entrena utilizando `model.fit(X_train, y_train)`, permitiendo que el modelo aprenda a identificar patrones en los datos de entrenamiento para poder realizar predicciones. Para verificar la confiabilidad del modelo, se utiliza la validación cruzada con `cross_val_score(model, X_train, y_train, cv=5)`, técnica que evalúa el modelo en múltiples subconjuntos del conjunto de entrenamiento, asegurando su estabilidad y robustez.

El modelo, ya entrenado, se utiliza para predecir los resultados en el conjunto de prueba con `model.predict(X_test)`. Además de las predicciones, el modelo calcula las probabilidades de que cada resultado pertenezca a una clase particular mediante `model.predict_proba(X_test)[:, 1]`, lo cual es crucial en aplicaciones donde las probabilidades son más informativas que las decisiones binarias.

La evaluación del rendimiento del modelo se realiza calculando la curva ROC y el área bajo la curva (AUC) con `roc_curve(y_test, y_proba, pos_label=model.classes_[1])` y `auc(fpr, tpr)`, respectivamente, proporcionando una medida de la capacidad del modelo para distinguir entre clases. Se genera una matriz de confusión con `confusion_matrix(y_test, y_pred)`, que visualiza el rendimiento del modelo mostrando el número de predicciones correctas e incorrectas. La precisión general del modelo se calcula con `accuracy_score(y_test, y_pred)`, indicando la frecuencia de las predicciones correctas.

Finalmente, se crea un informe detallado con `classification_report(y_test, y_pred)`, que incluye métricas como precisión, recall (sensibilidad) y puntuación F1 para cada clase, proporcionando una visión profunda del rendimiento del modelo. La función retorna un diccionario con todos los resultados y estadísticas relevantes, facilitando la revisión y comparación del comportamiento del modelo.

Cada uno de estos pasos es esencial para garantizar que el modelo KNN no solo funcione adecuadamente con los datos de entrenamiento, sino que también sea capaz de realizar predicciones precisas y confiables con nuevos datos.

8.1.4.4. Función evaluate_decision_tree_model:

```python
# Evaluar el modelo de Decision Tree
def evaluate_decision_tree_model(X, y):
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

    model = DecisionTreeClassifier()
    model.fit(X_train, y_train)

    cv_scores = cross_val_score(model, X_train, y_train, cv=5)

    y_pred = model.predict(X_test)
    y_proba = model.predict_proba(X_test)[:, 1]

    fpr, tpr, _ = roc_curve(y_test, y_proba, pos_label=model.classes_[1])
    roc_auc = auc(fpr, tpr)

    cm = confusion_matrix(y_test, y_pred)
    accuracy = accuracy_score(y_test, y_pred)
    report = classification_report(y_test, y_pred)

    return {
        'model': model,
        'cv_scores': cv_scores,
        'roc_curve': (fpr, tpr, roc_auc),
        'confusion_matrix': cm,
        'accuracy': accuracy,
        'classification_report': report
    }
```

El código mostrado describe una función para evaluar un modelo de clasificación mediante un árbol de decisión. A continuación, se detallan cada uno de los pasos y su relevancia de una manera comprensible para personas sin conocimientos técnicos.

La función `evaluate_decision_tree_model(X, y)` se define para evaluar el rendimiento de un modelo de árbol de decisión, donde `X` representa las características (datos de entrada) y `y` las etiquetas o resultados esperados. En primer lugar, los datos se dividen en conjuntos de entrenamiento (80%) y prueba (20%), garantizando que esta división sea reproducible mediante un estado aleatorio fijo (42).

A continuación, se crea una instancia del modelo de árbol de decisión con `DecisionTreeClassifier()`, y se entrena el modelo utilizando los datos de entrenamiento a través de `model.fit(X_train, y_train)`. Durante este proceso, el modelo aprende a tomar decisiones basándose en los valores de las características para predecir las etiquetas correspondientes. Para evaluar la estabilidad del modelo, se realiza una validación cruzada con cinco divisiones mediante `cross_val_score(model, X_train, y_train, cv=5)`, entrenando y evaluando el modelo en diferentes subconjuntos de los datos de entrenamiento para asegurar su robustez.

El modelo entrenado se utiliza para predecir las etiquetas del conjunto de prueba con `model.predict(X_test)`, evaluando su capacidad para generalizar a nuevos datos. Además de las predicciones, el modelo calcula las probabilidades de que las predicciones pertenezcan a la clase positiva utilizando `model.predict_proba(X_test)[:, 1]`, lo cual es útil en escenarios donde las decisiones no son simplemente binarias.

Para evaluar la capacidad del modelo de discriminar entre clases, se calcula la curva ROC mediante `roc_curve(y_test, y_proba, pos_label=model.classes_[1])`, y el área bajo la curva (AUC) en la

línea siguiente, donde un AUC grande indica un buen rendimiento del modelo. Se genera una matriz de confusión con `confusion_matrix(y_test, y_pred)`, que muestra el número de predicciones correctas e incorrectas divididas por clase, proporcionando una visión clara de los errores del modelo.

La precisión general del modelo se calcula con `accuracy_score(y_test, y_pred)`, indicando la proporción de predicciones correctas sobre el total. Además, se proporciona un informe de clasificación con `classification_report(y_test, y_pred)`, que incluye métricas clave como precisión, recall y el puntaje F1 para cada clase, ofreciendo una comprensión detallada del rendimiento del modelo.

Finalmente, la función retorna un diccionario con todos los resultados y métricas relevantes, facilitando el análisis y la comparación del rendimiento del modelo. Cada uno de estos pasos es crucial para asegurar que el modelo no solo aprende correctamente, sino que también puede hacer predicciones precisas en situaciones nuevas, un aspecto vital para la aplicación práctica de modelos de aprendizaje automático.

Las funciones implementadas demuestran un flujo de trabajo completo para la preparación y evaluación de datos, incluyendo técnicas esenciales como el balanceo de datos y la codificación de variables categóricas, seguidas de la implementación y evaluación de varios algoritmos de clasificación. Estos pasos son fundamentales para garantizar que los modelos sean justos, precisos y robustos.

8.2. DEFINICIÓN DE VARIABLES Y EVALUACIÓN DE RESULTADOS

En este apartado, se amplía la implementación previa, centrándose en la evaluación de varios modelos de clasificación para múltiples variables objetivo y la presentación de los resultados de manera estructurada.

Se definen claramente las variables de interés y se describen los procesos de preprocesamiento y transformación de datos necesarios para adaptarlos a los distintos algoritmos de clasificación. Además, se detallan los parámetros y configuraciones específicas utilizados para cada modelo, asegurando una evaluación precisa y consistente.

8.2.1. Descripción de variables.

```
X = balanced_data.drop(columns=target_variables)
```

El conjunto de características, representado como `X`, incluye todas las variables del conjunto de datos, excluyendo las variables objetivo. Las variables objetivo que serán evaluadas se encuentran listadas bajo `target_variables` = ['nivrecpsic', 'nivans', 'nivdep', 'nivest', 'nivsatvida', 'nivresil'].

8.2.2. Impresión de resultados.

```python
# Función para imprimir los resultados
def print_results(model_name, results, target_variable):
    print(f"Resultados para el modelo {model_name} - {target_variable}:")
    print("Precisión (Accuracy):", results['accuracy'])
    print("\nMatriz de Confusión:")
    print(results['confusion_matrix'])
    print("\nInforme de Clasificación:")
    print(results['classification_report'])
    print("\nCurva ROC AUC:", results['roc_curve'][2])
    print("\nPuntuaciones de validación cruzada:", results['cv_scores'])
    print("\n---\n")
```

El propósito de este código fue presentar los resultados de la evaluación del modelo de manera clara y organizada. Esta estructuración no solo mejora la comprensión de los resultados, sino que también permite una interpretación precisa de los datos obtenidos.

La importancia de esta presentación radica en su capacidad para facilitar la comparación entre diferentes modelos y variables objetivo. De esta forma, se simplifica el análisis comparativo, permitiendo identificar rápidamente las fortalezas y debilidades de cada modelo en relación con las distintas variables objetivo. Esta claridad en la presentación es fundamental para tomar decisiones informadas y optimizar el desempeño de los modelos evaluados.

8.2.3. Evaluación y presentación de resultados.

```python
# Evaluar y mostrar resultados para todas las target_variables
for target_variable in target_variables:
    results_svm = evaluate_svm_model(X, balanced_data[target_variable])
    print_results("SVM", results_svm, target_variable)

    results_mlp = evaluate_mlp_model(X, balanced_data[target_variable])
    print_results("MLP", results_mlp, target_variable)

    results_knn = evaluate_knn_model(X, balanced_data[target_variable])
    print_results("KNN", results_knn, target_variable)

    results_dt = evaluate_decision_tree_model(X, balanced_data[target_variable])
    print_results("Decision Tree", results_dt, target_variable)
```

Con la implementación de este código se buscó evaluar los modelos de clasificación para cada variable objetivo y presentar los resultados obtenidos. Este enfoque integral permite realizar una evaluación de los diversos modelos, asegurando que se seleccione el más adecuado para cada variable objetivo.

La importancia de esta metodología radica en su capacidad para proporcionar una visión clara y comparativa del rendimiento de cada modelo. La estructura del código está diseñada para garantizar que cada modelo se evalúe de manera sistemática y rigurosa para cada variable objetivo, permitiendo así una comparación justa y precisa. Los resultados se presentan de manera clara y organizada, facilitando su interpretación y utilización para la toma de decisiones informadas en el proceso de selección y optimización de modelos.

8.3. RESULTADOS DE LOS MODELOS

Los resultados de los modelos no se limitaron únicamente a métricas numéricas. Se llevó a cabo un análisis detallado de las características más influyentes para el modelo y se desarrolló una interfaz que facilita la predicción de las variables utilizando el mejor modelo de este proyecto.

8.3.1. Resumen de Resultados de Modelos para la Variable Recursos Psicológicos.

En este capítulo, se presenta un análisis integral que resume los resultados obtenidos de los diversos modelos empleados en función de la variable de Recursos Psicológicos y los algoritmos utilizados.

8.3.1.1. Resultados del Modelo SVM para la Variable Recursos Psicológicos.

El modelo SVM para Recursos Psicológicos fue evaluado para comprender su desempeño y capacidad predictiva. En esta sección, se presentan los resultados detallados de dicha evaluación. Se exploran métricas clave como la precisión, la matriz de confusión, el informe de clasificación y las puntuaciones de validación cruzada. Estos análisis proporcionan una visión del rendimiento del modelo SVM en la tarea específica de clasificación de la variable de recursos psicológicos, lo que permite una evaluación informada de su idoneidad y eficacia en este contexto.

8.3.1.1.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.3319027181688126
Curva ROC AUC	0.5693534954585082

Tabla 9. Métricas generales del modelo SVM para la Variable Recursos Psicológicos [50]

8.3.1.1.2. Matriz de Confusión.

[[0 0 241 147]

[0 0 263 168]

[0 0 372 274]

[0 0 308 324]]

8.3.1.1.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.00	0.00	0.00	388
1	0.00	0.00	0.00	431
2	0.31	0.58	0.41	646

3	0.35	0.51	0.42	632
accuracy	**0.33**			2097
macro avg	**0.17**	**0.27**	**0.21**	2097
weighted avg	**0.20**	**0.33**	**0.25**	2097

Tabla 10. Informe de clasificación del modelo SVM para la Variable Recursos Psicológicos [50]

8.3.1.1.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.32240763
Fold 2	0.32359952
Fold 3	0.31782946
Fold 4	0.30769231
Fold 5	0.30351819

Tabla 11. Puntuaciones de validación cruzada del modelo SVM para la Variable Recursos Psicológicos [50]

8.3.1.2. Resultados del Modelo MLP para la Variable Recursos Psicológicos.

En esta sección se presentan los resultados obtenidos del modelo MLP aplicado al análisis de recursos psicológicos.

8.3.1.2.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.30138292799237004
Curva ROC AUC	0.5

Tabla 12. Métricas generales del modelo MLP para la Variable Recursos Psicológicos [50]

8.3.1.2.2. Matriz de Confusión.

[[0 0 0 388]

[0 0 0 431]

[0 0 0 646]

[0 0 0 632]]

8.3.1.2.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.00	0.00	0.00	388

1	0.00	0.00	0.00	431
2	0.00	0.00	0.00	646
3	0.30	1.00	0.46	632
accuracy	**0.30**			2097
macro avg	**0.08**	**0.25**	**0.12**	2097
weighted avg	**0.09**	**0.30**	**0.14**	2097

Tabla 13. Informe de clasificación del modelo MLP para la Variable Recursos Psicológicos [50]

8.3.1.2.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	***Puntuación***
Fold 1	0.3045292
Fold 2	0.29558999
Fold 3	0.29576625
Fold 4	0.19499106
Fold 5	0.30411449

Tabla 14. Puntuaciones de validación cruzada del modelo MLP para la Variable Recursos Psicológicos [50]

8.3.1.3. Resultados del Modelo KNN para la Variable Recursos Psicológicos.

En esta sección se presentan los resultados obtenidos del modelo KNN aplicado al análisis de recursos psicológicos. Se examinan diversas métricas de rendimiento, como la precisión, la matriz de confusión, el informe de clasificación y las puntuaciones de validación cruzada.

8.3.1.3.1. Métricas Generales.

Métrica	***Valor***
Precisión (Accuracy)	0.5622317596566524
Curva ROC AUC	0.7646132699019283

Tabla 15. Métricas generales del modelo KNN para la Variable Recursos Psicológicos [50]

8.3.1.3.2. Matriz de Confusión.

[[254 12 88 34]

[133 133 126 39]

[239 35 310 62]

[122 0 28 482]]

8.3.1.3.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.34	0.65	0.45	388
1	0.74	0.31	0.44	431
2	0.56	0.48	0.52	646
3	0.78	0.76	0.77	632
accuracy	**0.56**			2097
macro avg	**0.61**	**0.55**	**0.54**	2097
weighted avg	**0.62**	**0.56**	**0.56**	2097

Tabla 16. Informe de clasificación del modelo KNN para la Variable Recursos Psicológicos [50]

8.3.1.3.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.55244338
Fold 2	0.51728248
Fold 3	0.56171735
Fold 4	0.5515802
Fold 5	0.5515802

Tabla 17. Puntuaciones de validación cruzada del modelo KNN para la Variable Recursos Psicológicos [50]

8.3.1.4. Resultados del Modelo Decision Tree para la Variable Recursos Psicológicos.

En esta sección se presentan los resultados obtenidos del modelo Decision Tree aplicado al análisis de recursos psicológicos. Estos resultados comprenden diversas métricas de rendimiento, incluyendo precisión, matriz de confusión, informe de clasificación y puntuaciones de validación cruzada.

8.3.1.4.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.9594659036719122
Curva ROC AUC	0.9646143840366772

Tabla 18. Métricas generales del modelo DT para la Variable Recursos Psicológicos [50]

8.3.1.4.2. Matriz de Confusión.

[[378 5 4 1]

[5 408 18 0]

[25 24 594   3]

[0   0   0 632]]

8.3.1.4.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.93	0.97	0.95	388
1	0.93	0.95	0.94	431
2	0.96	0.92	0.94	646
3	0.99	1.00	1.00	632
accuracy	**0.96**			2097
macro avg	**0.95**	**0.96**	**0.96**	2097
weighted avg	**0.96**	**0.96**	**0.96**	2097

Tabla 19. Informe de clasificación del modelo DT para la Variable Recursos Psicológicos [50]

8.3.1.4.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.93563766
Fold 2	0.94874851
Fold 3	0.95110316
Fold 4	0.94752534
Fold 5	0.95050686

Tabla 20. Puntuaciones de validación cruzada del modelo DT para la Variable Recursos Psicológicos [50]

8.3.2. Resumen de Resultados de Modelos para la Variable Ansiedad.

En este capítulo, se presenta un análisis integral que resume los resultados obtenidos de los diversos modelos empleados en función de la variable de Ansiedad y los algoritmos utilizados.

8.3.2.1. Resultados del Modelo SVM para la Variable Ansiedad.

En esta sección se presentan los resultados obtenidos del modelo SVM aplicado al análisis de la variable Ansiedad.

8.3.2.1.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.2756318550309967

Curva ROC AUC	0.5474055424692948

Tabla 21. Métricas generales del modelo SVM para la Variable Ansiedad [50]

8.3.2.1.2. Matriz de Confusión.

[[578 0 0 0 0 0]

[239 0 0 0 0 0]

[258 0 0 0 0 0]

[471 0 0 0 0 0]

[286 0 0 0 0 0]

[265 0 0 0 0 0]]

8.3.2.1.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.28	1.00	0.43	578
1	0.00	0.00	0.00	239
2	0.00	0.00	0.00	258
3	0.00	0.00	0.00	471
4	0.00	0.00	0.00	286
5	0.00	0.00	0.00	265
accuracy	**0.28**			2097
macro avg	**0.05**	**0.17**	**0.07**	2097
weighted avg	**0.08**	**0.28**	**0.12**	2097

Tabla 22. Informe de clasificación del modelo SVM para la Variable Ansiedad [50]

8.3.2.1.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.27890346
Fold 2	0.27890346
Fold 3	0.27906977
Fold 4	0.27847346
Fold 5	0.27847346

Tabla 23. Puntuaciones de validación cruzada del modelo SVM para la Variable Ansiedad [50]

8.3.2.2. Resultados del Modelo MLP para la Variable Ansiedad.

En esta sección se presentan los resultados obtenidos del modelo MLP aplicado al análisis de la variable Ansiedad.

8.3.2.2.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.12303290414878398
Curva ROC AUC	0.5

Tabla 24. Métricas generales del modelo MLP para la Variable Ansiedad [50]

8.3.2.2.2. Matriz de Confusión.

[[0 0 578 0 0 0]

[0 0 239 0 0 0]

[0 0 258 0 0 0]

[0 0 471 0 0 0]

[0 0 286 0 0 0]

[0 0 265 0 0 0]]

8.3.2.2.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.00	0.00	0.00	578
1	0.00	0.00	0.00	239
2	0.12	1.00	0.22	258
3	0.00	0.00	0.00	471
4	0.00	0.00	0.00	286
5	0.00	0.00	0.00	265
accuracy	**0.12**			2097
macro avg	**0.02**	**0.17**	**0.04**	2097
weighted avg	**0.02**	**0.12**	**0.03**	2097

Tabla 25. Informe de clasificación del modelo MLP para la Variable Ansiedad [50]

8.3.2.2.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.13170441

Fold 2	0.21573302
Fold 3	0.11508646
Fold 4	0.21645796
Fold 5	0.21645796

Tabla 26. Puntuaciones de validación cruzada del modelo MLP para la Variable Ansiedad [50]

8.3.2.3. Resultados del Modelo KNN para la Variable Ansiedad.

En esta sección se presentan los resultados obtenidos del modelo KNN aplicado al análisis de la variable Ansiedad. Se incluyen métricas clave de rendimiento como la precisión, la matriz de confusión, el informe de clasificación y las puntuaciones de validación cruzada.

8.3.2.3.1. Métricas Generales.

Métrica	***Valor***
Precisión (Accuracy)	0.5183595612780162
Curva ROC AUC	0.7414921790200468

Tabla 27. Métricas generales del modelo KNN para la Variable Ansiedad [50]

8.3.2.3.2. Matriz de Confusión.

[[521 7 15 2 15 18]

[7 227 2 0 2 1]

[3 0 253 0 2 0]

[0 0 0 471 0 0]

[5 5 9 2 263 2]

[4 3 0 0 0 258]]

8.3.2.3.3. Informe de Clasificación.

Clase	***Precision***	***Recall***	***F1-Score***	***Support***
0	0.53	0.52	0.53	578
1	0.38	0.30	0.34	239
2	0.54	0.34	0.42	258
3	0.61	0.87	0.72	471
4	0.41	0.26	0.32	286
5	0.43	0.54	0.47	265
accuracy	**0.52**			2097

macro avg	**0.48**	**0.47**	**0.47**	2097
weighted avg	**0.50**	**0.52**	**0.50**	2097

Tabla 28. Informe de clasificación del modelo KNN para la Variable Ansiedad [50]

8.3.2.3.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.50178784
Fold 2	0.47318236
Fold 3	0.48300537
Fold 4	0.50089445
Fold 5	0.49135361

Tabla 29. Puntuaciones de validación cruzada del modelo KNN para la Variable Ansiedad [50]

8.3.2.4. Resultados del Modelo Decision Tree para la Variable Ansiedad.

En esta sección se presentan los resultados obtenidos del modelo Decision Tree aplicado al análisis de la variable Ansiedad.

8.3.2.4.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.9504053409632809
Curva ROC AUC	0.9708587989965366

Tabla 30. Métricas generales del modelo DT para la Variable Ansiedad [50]

8.3.2.4.2. Matriz de Confusión.

[[521 7 15 2 15 18]

[7 227 2 0 2 1]

[3 0 253 0 2 0]

[0 0 0 471 0 0]

[5 5 9 2 263 2]

[4 3 0 0 0 258]]

8.3.2.4.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.96	0.90	0.93	578
1	0.94	0.95	0.94	239

2	0.91	0.98	0.94	258
3	0.99	1.00	1.00	471
4	0.93	0.92	0.93	286
5	0.92	0.97	0.95	265
accuracy	**0.95**			2097
macro avg	**0.94**	**0.95**	**0.95**	2097
weighted avg	**0.95**	**0.95**	**0.95**	2097

Tabla 31. Informe de clasificación del modelo DT para la Variable Ansiedad [50]

8.3.2.4.4. Puntuaciones de Validación Cruzada

Validación Cruzada	*Puntuación*
Fold 1	0.92550656
Fold 2	0.92550656
Fold 3	0.93082886
Fold 4	0.93559928
Fold 5	0.92486583

Tabla 32. Puntuaciones de validación cruzada del modelo DT para la Variable Ansiedad [50]

8.3.3. Resumen de Resultados de Modelos para la Variable Depresión.

En este capítulo, se presenta un análisis integral que resume los resultados obtenidos de los diversos modelos empleados en función de la variable de Depresión y los algoritmos utilizados.

8.3.3.1. Resultados del Modelo SVM para la Variable Depresión.

En esta sección se presentan los resultados obtenidos del modelo SVM aplicado al análisis de la variable Depresión. Se incluyen métricas clave de rendimiento como la precisión, la matriz de confusión, el informe de clasificación y las puntuaciones de validación cruzada.

8.3.3.1.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.31998092513113974
Curva ROC AUC	0.5475575482702866

Tabla 33. Métricas generales del modelo SVM para la Variable Depresión [50]

8.3.3.1.2. Matriz de Confusión.

[[225 0 0 46 275 0]

[69 0 0 18 152 0]

[89 0 0 2 129 0]

[110 0 0 172 76 0]

[220 0 0 42 274 0]

[75 0 0 10 113 0]]

8.3.3.1.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.29	0.41	0.34	546
1	0.00	0.00	0.00	239
2	0.00	0.00	0.00	220
3	0.59	0.48	0.53	358
4	0.27	0.51	0.35	536
5	0.00	0.00	0.00	198
accuracy	**0.32**			2097
macro avg	**0.19**	**0.23**	**0.20**	2097
weighted avg	**0.24**	**0.32**	**0.27**	2097

Tabla 34. Informe de clasificación del modelo SVM para la Variable Depresión [50]

8.3.3.1.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.26460072
Fold 2	0.2443385
Fold 3	0.25819917
Fold 4	0.25641026
Fold 5	0.25044723

Tabla 35. Puntuaciones de validación cruzada del modelo SVM para la Variable Depresión [50]

8.3.3.2. Resultados del Modelo MLP para la Variable Depresión.

En esta sección se presentan los resultados obtenidos del modelo MLP (Multilayer Perceptron) aplicado al análisis de la variable Depresión. El análisis incluye métricas clave de rendimiento como la precisión, la matriz de confusión, el informe de clasificación y las puntuaciones de validación cruzada.

8.3.3.2.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.25560324272770624
Curva ROC AUC	0.5

Tabla 36. Métricas generales del modelo MLP para la Variable Depresión [50]

8.3.3.2.2. Matriz de Confusión.

[[0 0 0 0 546 0]

[0 0 0 0 239 0]

[0 0 0 0 220 0]

[0 0 0 0 358 0]

[0 0 0 0 536 0]

[0 0 0 0 198 0]]

8.3.3.2.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.00	0.00	0.00	546
1	0.00	0.00	0.00	239
2	0.00	0.00	0.00	220
3	0.00	0.00	0.00	358
4	0.26	1.00	0.41	536
5	0.00	0.00	0.00	198
accuracy	**0.26**			2097
macro avg	**0.04**	**0.17**	**0.07**	2097
weighted avg	**0.07**	**0.26**	**0.10**	2097

Tabla 37. Informe de clasificación del modelo MLP para la Variable Depresión [50]

8.3.3.2.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.25268176
Fold 2	0.17818832
Fold 3	0.17769827
Fold 4	0.17769827

Fold 5	0.25223614

Tabla 38. Puntuaciones de validación cruzada del modelo MLP para la Variable Depresión [50]

8.3.3.3. Resultados del Modelo KNN para la Variable Depresión.

En esta sección, se presentan los resultados obtenidos del modelo KNN (K-Nearest Neighbors) aplicado al análisis de la variable Depresión.

8.3.3.3.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.5512637100619934
Curva ROC AUC	0.7880183397813818

Tabla 39. Métricas generales del modelo KNN para la Variable Depresión [50]

8.3.3.3.2. Matriz de Confusión.

[[340 59 23 6 102 16]

[61 131 12 1 34 0]

[66 40 50 0 52 12]

[50 12 0 285 9 2]

[149 46 32 6 294 9]

[77 24 10 1 30 56]]

8.3.3.3.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.46	0.62	0.53	546
1	0.42	0.55	0.48	239
2	0.39	0.23	0.29	220
3	0.95	0.80	0.87	358
4	0.56	0.55	0.56	536
5	0.59	0.28	0.38	198
accuracy	**0.55**			2097
macro avg	**0.56**	**0.50**	**0.52**	2097
weighted avg	**0.57**	**0.55**	**0.55**	2097

Tabla 40. Informe de clasificación del modelo KNN para la Variable Depresión [50]

8.3.3.3.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.48569726
Fold 2	0.51430274
Fold 3	0.52415027
Fold 4	0.51699463
Fold 5	0.51222421

Tabla 41. Puntuaciones de validación cruzada del modelo KNN para la Variable Depresión [50]

8.3.3.4. Resultados del Modelo Decision Tree para la Variable Depresión.

En esta sección se presentan los resultados obtenidos del modelo de árbol de decisión aplicado al análisis de la variable Depresión. Estos resultados representan el rendimiento del modelo en términos de precisión, matriz de confusión, informe de clasificación y puntuaciones de validación cruzada.

8.3.3.4.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.9523128278493085
Curva ROC AUC	0.9792269998333566

Tabla 42. Métricas generales del modelo DT para la Variable Depresión [50]

8.3.3.4.2. Matriz de Confusión.

[[501 10 14 3 8 10]

[3 231 1 0 3 1]

[7 0 206 0 6 1]

[0 0 0 358 0 0]

[10 4 9 0 508 5]

[1 1 1 2 0 193]]

8.3.3.4.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.96	0.92	0.94	546
1	0.94	0.97	0.95	239
2	0.89	0.94	0.91	220

3	0.99	1.00	0.99	358
4	0.97	0.95	0.96	536
5	0.92	0.97	0.95	198
accuracy	**0.95**			2097
macro avg	**0.94**	**0.96**	**0.95**	2097
weighted avg	**0.95**	**0.95**	**0.95**	2097

Tabla 43. Informe de clasificación del modelo DT para la Variable Depresión [50]

8.3.3.4.4. Puntuaciones de Validación Cruzada

Validación Cruzada	*Puntuación*
Fold 1	0.92312277
Fold 2	0.92491061
Fold 3	0.92546213
Fold 4	0.93321407
Fold 5	0.93619559

Tabla 44. Puntuaciones de validación cruzada del modelo DT para la Variable Depresión [50]

8.3.4. Resumen de Resultados de Modelos para la Variable Estrés.

En este capítulo, se presenta un análisis integral que resume los resultados obtenidos de los diversos modelos empleados en función de la variable de Estrés y los algoritmos utilizados.

8.3.4.1. Resultados del Modelo SVM para la Variable Estrés.

A continuación, se presentan los resultados obtenidos del modelo SVM aplicado a la variable de estrés. Estos resultados comprenden métricas de desempeño como la precisión (accuracy), la matriz de confusión, el informe de clasificación, la curva ROC AUC y las puntuaciones de validación cruzada. Estas métricas proporcionan una visión integral del rendimiento del modelo SVM en la predicción del nivel de estrés en el conjunto de datos analizado.

8.3.4.1.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.36576061039580354
Curva ROC AUC	0.5660413747308494

Tabla 45. Métricas generales del modelo SVM para la Variable Estrés [50]

8.3.4.1.2. Matriz de Confusión.

[[767 0 0 0 0 0]

[218 0 0 0 0 0]

[282 0 0 0 0 0]

[309 0 0 0 0 0]

[302 0 0 0 0 0]

[219 0 0 0 0 0]]

8.3.4.1.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.37	1.00	0.54	767
1	0.00	0.00	0.00	218
2	0.00	0.00	0.00	282
3	0.00	0.00	0.00	309
4	0.00	0.00	0.00	302
5	0.00	0.00	0.00	219
Accuracy	0.37			2097
Macro avg	0.06	0.17	0.09	2097
Weighted avg	0.13	0.37	0.20	2097

Tabla 46. Informe de clasificación del modelo SVM para la Variable Estrés [50]

8.3.4.1.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.3545888
Fold 2	0.3545888
Fold 3	0.35480024
Fold 4	0.35480024
Fold 5	0.35420394

Tabla 47. Puntuaciones de validación cruzada del modelo SVM para la Variable Estrés [50]

8.3.4.2. Resultados del Modelo MLP para la Variable Estrés.

En esta sección se presentan los resultados obtenidos del modelo de Perceptrón Multicapa (MLP) aplicado a la variable de estrés.

8.3.4.2.1. Métricas Generales.

Métrica	*Valor*

Precisión (Accuracy)	0.36576061039580354
Curva ROC AUC	0.5

Tabla 48. Métricas generales del modelo MLP para la Variable Estrés [50]

8.3.4.2.2. Matriz de Confusión.

[[767 0 0 0 0 0]
[218 0 0 0 0 0]
[282 0 0 0 0 0]
[309 0 0 0 0 0]
[302 0 0 0 0 0]
[219 0 0 0 0 0]]

8.3.4.2.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.37	1.00	0.54	767
1	0.00	0.00	0.00	218
2	0.00	0.00	0.00	282
3	0.00	0.00	0.00	309
4	0.00	0.00	0.00	302
5	0.00	0.00	0.00	219
accuracy	-	-	0.37	2097
macro avg	0.06	0.17	0.09	2097
weighted avg	0.13	0.37	0.20	2097

Tabla 49. Informe de clasificación del modelo MLP para la Variable Estrés [50]

8.3.4.2.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.14898689
Fold 2	0.11620977
Fold 3	0.15444246
Fold 4	0.12820513
Fold 5	0.35420394

Tabla 50. Puntuaciones de validación cruzada del modelo MLP para la Variable Estrés [50]

8.3.4.3. Resultados del Modelo KNN para la Variable Estrés.

En esta sección, se presentan los resultados obtenidos mediante la aplicación del modelo KNN a la variable de estrés. Estos resultados se analizan detalladamente en términos de precisión, matriz de confusión, informe de clasificación y puntuaciones de validación cruzada, con el objetivo de evaluar el rendimiento del modelo en la predicción de la variable de interés.

8.3.4.3.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.5445875059608966
Curva ROC AUC	0.7809773400842728

Tabla 51. Métricas generales del modelo KNN para la Variable Estrés [50]

8.3.4.3.2. Matriz de Confusión.

[[501 28 83 4 119 32]

[49 89 34 0 40 6]

[81 11 119 1 55 15]

[5 4 10 228 61 1]

[80 6 60 4 137 15]

[58 10 42 0 41 68]]

8.3.4.3.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.65	0.65	0.65	767
1	0.60	0.41	0.49	218
2	0.34	0.42	0.38	282
3	0.96	0.74	0.84	309
4	0.30	0.45	0.36	302
5	0.50	0.31	0.38	219
Accuracy	-	-	0.54	2097
macro avg	0.56	0.50	0.52	2097
weighted avg	0.58	0.54	0.55	2097

Tabla 52. Informe de clasificación del modelo KNN para la Variable Estrés [50]

8.3.4.3.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.49404052
Fold 2	0.49761621
Fold 3	0.51162791
Fold 4	0.51222421
Fold 5	0.50387597

Tabla 53. Puntuaciones de validación cruzada del modelo KNN para la Variable Estrés [50]

8.3.4.4. Resultados del Modelo Decision Tree para la Variable Estrés.

Los resultados obtenidos del modelo Decision Tree aplicado a la variable de estrés se presentan en esta sección. Estas tablas ofrecen una visión detallada del desempeño del modelo en términos de precisión, matriz de confusión, informe de clasificación y puntuaciones de validación cruzada.

8.3.4.4.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.9403910348116357
Curva ROC AUC	0.9858711202035048

Tabla 54. Métricas generales del modelo DT para la Variable Estrés [50]

8.3.4.4.2. Matriz de Confusión.

[[717 7 17 4 11 11]

[0 213 0 0 5 0]

[14 1 254 3 8 2]

[0 0 0 309 0 0]

[12 1 4 2 282 1]

[10 1 4 0 7 197]]

8.3.4.4.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.95	0.93	0.94	767
1	0.96	0.98	0.97	218
2	0.91	0.90	0.91	282
3	0.97	1.00	0.99	309

4	0.90	0.93	0.92	302
5	0.93	0.90	0.92	219
Accuracy	-	-	0.94	2097
macro avg	0.94	0.94	0.94	2097
weighted avg	0.94	0.94	0.94	2097

Tabla 55. Informe de clasificación del modelo DT para la Variable Estrés [50]

8.3.4.4.4. Puntuaciones de Validación Cruzada

Validación Cruzada	***Puntuación***
Fold 1	0.91895113
Fold 2	0.9272944
Fold 3	0.93142516
Fold 4	0.93261777
Fold 5	0.9177102

Tabla 56. Puntuaciones de validación cruzada del modelo DT para la Variable Estrés [50]

8.3.5. Resumen de Resultados de Modelos para la Variable Satisfacción de Vida.

En este capítulo, se presenta un análisis integral que resume los resultados obtenidos de los diversos modelos empleados en función de la variable de Satisfacción de Vida y los algoritmos utilizados.

8.3.5.1. Resultados del Modelo SVM para la Variable Satisfacción de Vida.

A continuación se presentan los resultados del modelo SVM aplicado a la variable de atisfacción con la vida. Estas tablas ofrecen una visión detallada del desempeño del modelo, incluyendo la precisión, la matriz de confusión, el informe de clasificación y las puntuaciones de validación cruzada.

8.3.5.1.1. Métricas Generales.

Métrica	***Valor***
Precisión (Accuracy)	0.46542680019074867
Curva ROC AUC	0.5528850867732131

Tabla 57. Métricas generales del modelo SVM para la Variable Satisfacción de Vida [50]

8.3.5.1.2. Matriz de Confusión.

[[976 0 0 0]

[217 0 0 0]

[640 0 0 0]

[264 0 0 0]]

8.3.5.1.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.47	1.00	0.64	976
1	0.00	0.00	0.00	217
2	0.00	0.00	0.00	640
3	0.00	0.00	0.00	264
Accuracy	-	-	0.47	2097
Macro avg	0.12	0.25	0.16	2097
Weighted avg	0.22	0.47	0.30	2097

Tabla 58. Informe de clasificación del modelo SVM para la Variable Satisfacción de Vida [50]

8.3.5.1.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.47854589
Fold 2	0.47854589
Fold 3	0.47883125
Fold 4	0.47823494
Fold 5	0.47883125

Tabla 59. Puntuaciones de validación cruzada del modelo SVM para la Variable Satisfacción de Vida [50]

8.3.5.2. Resultados del Modelo MLP para la Variable Satisfacción de Vida.

En esta sección se presentan los resultados obtenidos del modelo de Perceptrón Multicapa (MLP) aplicado a la variable de Satisfacción de Vida.

8.3.5.2.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.10348116356700048
Curva ROC AUC	0.5

Tabla 60. Métricas generales del modelo MLP para la Variable Satisfacción de Vida [50]

8.3.5.2.2. Matriz de Confusión.

[[0 976 0 0]

[0 217 0 0]

[0 640 0 0]

[0 264 0 0]]

8.3.5.2.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.00	0.00	0.00	976
1	0.10	1.00	0.19	217
2	0.00	0.00	0.00	640
3	0.00	0.00	0.00	264
Accuracy	-	-	0.10	2097
Macro avg	0.03	0.25	0.05	2097
Weighted avg	0.01	0.10	0.02	2097

Tabla 61. Informe de clasificación del modelo MLP para la Variable Satisfacción de Vida [50]

8.3.5.2.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.47854589
Fold 2	0.30810489
Fold 3	0.11985689
Fold 4	0.30828861
Fold 5	0.30769231

Tabla 62. Puntuaciones de validación cruzada del modelo MLP para la Variable Satisfacción de Vida [50]

8.3.5.3. Resultados del Modelo KNN para la Variable Satisfacción de Vida.

En esta sección, se presentan los resultados obtenidos mediante la aplicación del modelo KNN a la variable de satisfacción de vida. Estos resultados se analizan detalladamente en términos de precisión, matriz de confusión, informe de clasificación y puntuaciones de validación cruzada, con el objetivo de evaluar el rendimiento del modelo en la predicción de la variable de interés.

8.3.5.3.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.670004768717215
Curva ROC AUC	0.7469347485047553

Tabla 63. Métricas generales del modelo KNN para la Variable Satisfacción de Vida [50]

8.3.5.3.2. Matriz de Confusión.

[[818 16 140 2]

[135 44 38 0]

[306 24 306 4]

[24 0 3 237]]

8.3.5.3.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.64	0.84	0.72	976
1	0.52	0.20	0.29	217
2	0.63	0.48	0.54	640
3	0.98	0.90	0.93	264
Accuracy	-	-	0.67	2097
Macro avg	0.69	0.60	0.62	2097
Weighted avg	0.67	0.67	0.65	2097

Tabla 64. Informe de clasificación del modelo KNN para la Variable Satisfacción de Vida [50]

8.3.5.3.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.66448153
Fold 2	0.66865316
Fold 3	0.68276685
Fold 4	0.6726297
Fold 5	0.67680382

Tabla 65. Puntuaciones de validación cruzada del modelo KNN para la Variable Satisfacción de Vida [50]

8.3.5.4. Resultados del Modelo Decision Tree para la Variable Satisfacción de Vida.

Los resultados obtenidos del modelo Decision Tree aplicado a la variable de satisfacción de vida se presentan en esta sección.

8.3.5.4.1. Métricas Generales.

Métrica	*Valor*

Precisión (Accuracy)	0.9589890319504053
Curva ROC AUC	0.9809454358270417

Tabla 66. Métricas generales del modelo DT para la Variable Satisfacción de Vida [50]

8.3.5.4.2. Matriz de Confusión.

[[945 4 27 0]

[0 210 7 0]

[41 7 592 0]

[0 0 0 264]]

8.3.5.4.3. Informe de Clasificación.

Clase	***Precision***	***Recall***	***F1-Score***	***Support***
0	0.96	0.97	0.96	976
1	0.95	0.97	0.96	217
2	0.95	0.93	0.94	640
3	1.00	1.00	1.00	264
Accuracy	-	-	0.96	2097
Macro avg	0.96	0.97	0.96	2097
Weighted avg	0.96	0.96	0.96	2097

Tabla 67. Informe de clasificación del modelo DT para la Variable Satisfacción de Vida [50]

8.3.5.4.4. Puntuaciones de Validación Cruzada

Validación Cruzada	***Puntuación***
Fold 1	0.93802145
Fold 2	0.93563766
Fold 3	0.94871795
Fold 4	0.94335122
Fold 5	0.94156231

Tabla 68. Puntuaciones de validación cruzada del modelo DT para la Variable Satisfacción de Vida [50]

8.3.6. Resumen de Resultados de Modelos para la Variable Resiliencia.

En este capítulo, se presenta un análisis integral que resume los resultados obtenidos de los diversos modelos empleados en función de la variable de Resiliencia y los algoritmos utilizados.

8.3.6.1. Resultados del Modelo SVM para la Variable Resiliencia.

Los siguientes resultados detallan el desempeño del modelo SVM aplicado a la variable de resiliencia en el estudio analizado.

8.3.6.1.1. Métricas Generales.

Métrica	*Valor*
Precisión (Accuracy)	0.5436337625178826
Curva ROC AUC	0.5491039426523298

Tabla 69. Métricas generales del modelo SVM para la Variable Resiliencia [50]

8.3.6.1.2. Matriz de Confusión.

[[223 0 449 27]

[82 0 135 20]

[191 0 730 42]

[1 0 10 187]]

8.3.6.1.3. Informe de Clasificación.

Clase	*Precision*	*Recall*	*F1-Score*	*Support*
0	0.45	0.32	0.37	699
1	0.00	0.00	0.00	237
2	0.55	0.76	0.64	963
3	0.68	0.94	0.79	198
Accuracy	-	-	0.54	2097
Macro avg	0.42	0.51	0.45	2097
Weighted avg	0.47	0.54	0.49	2097

Tabla 70. Informe de clasificación del modelo SVM para la Variable Resiliencia [50]

8.3.6.1.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	*Puntuación*
Fold 1	0.47794994
Fold 2	0.43027414
Fold 3	0.46213476

Fold 4	0.44543828
Fold 5	0.47465713

Tabla 71. Puntuaciones de validación cruzada del modelo SVM para la Variable Resiliencia [50]

8.3.6.2. Resultados del Modelo MLP para la Variable Resiliencia.

Los resultados presentados a continuación ofrecen un análisis detallado del desempeño del modelo de perceptrón multicapa (MLP) en relación con la variable de resiliencia en el contexto del estudio.

8.3.6.2.1. Métricas Generales.

Métrica	***Valor***
Precisión (Accuracy)	0.11301859799713877
Curva ROC AUC	0.5

Tabla 72. Métricas generales del modelo MLP para la Variable Resiliencia [50]

8.3.6.2.2. Matriz de Confusión.

[[0 699 0 0]

[0 237 0 0]

[0 963 0 0]

[0 198 0 0]]

8.3.6.2.3. Informe de Clasificación.

Clase	***Precision***	***Recall***	***F1-Score***	***Support***
0	0.00	0.00	0.00	699
1	0.11	1.00	0.20	237
2	0.00	0.00	0.00	963
3	0.00	0.00	0.00	198
Accuracy	-	-	0.11	2097
Macro avg	0.03	0.25	0.05	2097
Weighted avg	0.01	0.11	0.02	2097

Tabla 73. Informe de clasificación del modelo MLP para la Variable Resiliencia [50]

8.3.6.2.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	***Puntuación***
Fold 1	0.35756853
Fold 2	0.35756853

Fold 3	0.43768634
Fold 4	0.43709004
Fold 5	0.43709004

Tabla 74. Puntuaciones de validación cruzada del modelo MLP para la Variable Resiliencia [50]

8.3.6.3. Resultados del Modelo KNN para la Variable Resiliencia.

En este apartado se presentan los resultados obtenidos mediante el modelo de vecinos más cercanos (KNN) aplicado a la variable de resiliencia.

8.3.6.3.1. Métricas Generales.

Métrica	***Valor***
Precisión (Accuracy)	0.648068669527897
Curva ROC AUC	0.8108876638990972

Tabla 75. Métricas generales del modelo KNN para la Variable Resiliencia [50]

8.3.6.3.2. Matriz de Confusión.

[[534 8 157 0]

[106 82 48 1]

[380 25 556 2]

[5 1 5 187]]

8.3.6.3.3. Informe de Clasificación.

Clase	***Precision***	***Recall***	***F1-Score***	***Support***
0	0.52	0.76	0.62	699
1	0.71	0.35	0.46	237
2	0.73	0.58	0.64	963
3	0.98	0.94	0.96	198
Accuracy	-	-	0.65	2097
Macro avg	0.73	0.66	0.67	2097
Weighted avg	0.68	0.65	0.65	2097

Tabla 76. Informe de clasificación del modelo KNN para la Variable Resiliencia [50]

8.3.6.3.4. Puntuaciones de Validación Cruzada.

Validación Cruzada	***Puntuación***
Fold 1	0.62038141

Fold 2	0.62038141
Fold 3	0.63565891
Fold 4	0.63565891
Fold 5	0.63088849

Tabla 77. Puntuaciones de validación cruzada del modelo KNN para la Variable Resiliencia [50]

8.3.6.4. Resultados del Modelo Decision Tree para la Variable Resiliencia.

En esta sección se presentan los resultados derivados del uso del modelo de árbol de decisión para analizar la variable de resiliencia. Estas métricas ofrecen una visión detallada del rendimiento del modelo en la predicción de la resiliencia, proporcionando información valiosa sobre su capacidad para clasificar eficazmente las distintas categorías de esta variable en el contexto del estudio.

8.3.6.4.1. Métricas Generales.

Métrica	***Valor***
Precisión (Accuracy)	0.9518359561278016
Curva ROC AUC	0.9363583775690758

Tabla 78. Métricas generales del modelo DT para la Variable Resiliencia [50]

8.3.6.4.2. Matriz de Confusión.

[[673 3 22 1]

[8 209 19 1]

[30 14 917 2]

[0 0 1 197]]

8.3.6.4.3. Informe de Clasificación.

Clase	***Precision***	***Recall***	***F1-Score***	***Support***
0	0.95	0.96	0.95	699
1	0.92	0.88	0.90	237
2	0.96	0.95	0.95	963
3	0.98	0.99	0.99	198
Accuracy	-	-	0.95	2097
Macro avg	0.95	0.95	0.95	2097
Weighted avg	0.95	0.95	0.95	2097

Tabla 79. Informe de clasificación del modelo DT para la Variable Resiliencia [50]

8.3.6.4.4. Puntuaciones de Validación Cruzada

Validación Cruzada	*Puntuación*
Fold 1	0.94278903
Fold 2	0.94219309
Fold 3	0.94812165
Fold 4	0.94931425
Fold 5	0.93679189

Tabla 80. Puntuaciones de validación cruzada del modelo DT para la Variable Resiliencia [50]

8.4. INTERPRETACIÓN DE RESULTADOS

Variable	Modelo	Precisión	ROC AUC	Interpretación General
Recursos Psicológicos	SVM	33%	0.57	Bajo rendimiento en precisión y curva ROC AUC.
Recursos Psicológicos	Decision Tree	96%	0.96	Mejor modelo con precisión y ROC AUC.
Recursos Psicológicos	KNN	56%	0.76	Mejor rendimiento con una precisión de 56%.
Recursos Psicológicos	MLP	30%	0.50	Bajo rendimiento en precisión y curva ROC AUC.
Satisfacción de Vida	SVM	47%	0.55	Bajo rendimiento en precisión y curva ROC AUC.
Satisfacción de Vida	Decision Tree	96%	0.98	Mejor modelo con alta precisión y ROC AUC (96%).
Satisfacción de Vida	KNN	67%	0.75	Rendimiento moderado con precisión del 67%.
Satisfacción de Vida	MLP	10%	0.50	Bajo rendimiento en precisión y curva ROC AUC.
Resiliencia	SVM	54%	0.55	Bajo rendimiento en precisión y curva ROC AUC.
Resiliencia	Decision Tree	95%	0.94	Mejor modelo con alta precisión y ROC AUC (95%).
Resiliencia	KNN	65%	0.81	Rendimiento moderado con precisión del 65%.
Resiliencia	MLP	11%	0.50	Bajo rendimiento en precisión y curva ROC AUC.
Ansiedad	SVM	28%	0.55	Bajo rendimiento en precisión y curva ROC AUC.
Ansiedad	Decision Tree	95%	0.97	Mejor modelo con precisión y ROC AUC muy altas.
Ansiedad	KNN	52%	0.74	Mejor rendimiento con una precisión de 52%.
Ansiedad	MLP	12%	0.50	Bajo rendimiento en precisión y curva ROC AUC.
Depresión	SVM	32%	0.55	Bajo rendimiento en precisión y curva ROC AUC.
Depresión	Decision Tree	95%	0.98	Mejor modelo con alta precisión y ROC AUC (95%).

Depresión	KNN	55%	0.79	Rendimiento moderado con precisión del 55%.
Depresión	MLP	26%	0.50	Bajo rendimiento en precisión y curva ROC AUC.
Estrés	SVM	36%	0.57	Bajo rendimiento en precisión y curva ROC AUC.
Estrés	Decision Tree	94%	0.99	Mejor modelo con alta precisión y ROC AUC (94%).
Estrés	KNN	54%	0.78	Rendimiento moderado con precisión del 54%.
Estrés	MLP	36%	0.50	Bajo rendimiento en precisión y curva ROC AUC.

Tabla 81. Resumen de resultados [50]

El análisis de los modelos aplicados revela tendencias claras en cuanto a su eficacia en la predicción de variables. Destaca el rendimiento consistente y superior de los modelos de árbol de decisión, evidenciado por su alta precisión, que supera el 94%, y una curva ROC AUC cercana a 0.98. Estos modelos demuestran una capacidad excepcional para la clasificación precisa dentro del conjunto de datos estudiado.

Además, se observa que el método KNN muestra un desempeño aceptable, con una precisión que oscila entre el 52% y el 56%, y una curva ROC AUC que varía de 0.74 a 0.79. Aunque no alcanza el nivel de precisión de los árboles de decisión, el KNN podría ser una alternativa viable en ciertos contextos de análisis, dependiendo de las particularidades del estudio.

Por otro lado, tanto los modelos SVM como MLP exhiben un rendimiento mediocre en general, con precisiones entre el 12% y el 33% y curvas ROC AUC que no superan el 0.57. Estos resultados sugieren una capacidad predictiva limitada en comparación con los otros modelos evaluados.

En cuanto a la métrica de la curva ROC AUC, se destaca su importancia como indicador del rendimiento del modelo, especialmente en la diferenciación entre clases. Aquí, los modelos de árbol de decisión sobresalen nuevamente, mostrando las mejores curvas ROC AUC y, por ende, una excelente capacidad para distinguir entre diferentes clases.

Los modelos de KNN también demuestran un rendimiento considerable en términos de curva ROC AUC, aunque no alcanzan el nivel de excelencia exhibido por los árboles de decisión.

La variabilidad en el desempeño de los modelos arroja luces sobre su capacidad de generalización. Los resultados de la validación cruzada revelan una consistencia notable en los modelos de Decision Tree, con puntuaciones cercanas entre sí en diferentes subconjuntos del conjunto de datos. Por otro lado, los modelos de SVM y MLP muestran una mayor variabilidad y menor precisión, sugiriendo que podrían no ser la elección más adecuada para este conjunto de datos específico.

Analizando las matrices de confusión, se observa que los modelos de Decision Tree presentan un alto número de verdaderos positivos y un bajo número de falsos negativos y falsos positivos en todas las clases, indicando una alta precisión y consistencia en sus clasificaciones. Por el contrario, tanto los modelos SVM como MLP muestran una proporción significativa de falsos negativos y falsos positivos, especialmente en las clases minoritarias, lo que sugiere una dificultad para

diferenciar adecuadamente entre las clases. En este sentido, los modelos KNN muestran un equilibrio aceptable entre verdaderos positivos y falsos negativos en sus matrices de confusión, aunque con cierta variabilidad en las clases minoritarias.

En resumen, para las tareas de clasificación en este conjunto de datos, los modelos de Decision Tree emergen como la opción más robusta y efectiva, seguidos por los modelos de KNN. Por el contrario, los modelos de SVM y MLP no logran ofrecer un rendimiento satisfactorio y podrían no ser recomendables para su implementación en estos casos. Las matrices de confusión respaldan estas conclusiones al resaltar la capacidad superior de los modelos de Decision Tree para realizar clasificaciones precisas en todas las clases, mientras que los modelos de SVM y MLP enfrentan dificultades significativas en este aspecto.

8.5. PARÁMETROS DEL MEJOR MODELO

Los parámetros del mejor modelo corresponden a los hiperparámetros de un modelo de árbol de decisión implementado en `scikit-learn`. A continuación, se presenta una explicación detallada de cada uno de ellos:

El parámetro `ccp_alpha` está configurado en `0.0`. Este es el parámetro de coste-complejidad utilizado para la poda del árbol. Un valor mayor de `ccp_alpha` simplifica el árbol al reducir el número de nodos. El parámetro `class_weight` se encuentra en `None`, lo que implica que todas las clases tienen el mismo peso, aunque podría configurarse como un diccionario, una lista de diccionarios, "balanced", entre otros.

El `criterion` utilizado es 'gini', que es la función para medir la calidad de una división, aunque también se podría utilizar 'entropy' para medir la ganancia de información. La profundidad máxima del árbol (`max_depth`) está en `None`, permitiendo que los nodos se expandan hasta que todas las hojas sean puras o contengan menos muestras de las especificadas en `min_samples_split`.

El número de características a considerar al buscar la mejor división (`max_features`) también está en `None`, lo que significa que se consideran todas las características. El parámetro `max_leaf_nodes` está configurado en `None`, permitiendo un número ilimitado de nodos hoja. Por otro lado, `min_impurity_decrease` está en `0.0`, lo que define una fracción mínima de la disminución de la impureza necesaria para realizar una división; las divisiones no se realizan a menos que la disminución de la impureza sea al menos de este valor.

El número mínimo de muestras requeridas para estar en un nodo hoja (`min_samples_leaf`) es `1`, haciendo que el árbol sea lo suficientemente detallado. En cuanto a `min_samples_split`, está configurado en `2`, el número mínimo de muestras requeridas para dividir un nodo interno, ayudando a evitar que el modelo ajuste demasiado los datos de entrenamiento.

El parámetro `min_weight_fraction_leaf` está en `0.0`, lo que representa la fracción mínima ponderada de las sumas totales de los pesos necesarios para estar en un nodo hoja. La semilla utilizada por el generador de números aleatorios para obtener resultados reproducibles está definida por `random_state`, configurado en `42`.

Finalmente, el `splitter` está establecido en 'best', indicando que la estrategia utilizada para elegir la división en cada nodo es la mejor división posible.

Estos parámetros en conjunto definen cómo se construye y entrena el árbol de decisión, influyendo en su complejidad, precisión y capacidad de generalización.

9. INTERFAZ EN STREAMLIT

Para predecir variables objetivo basadas en entradas proporcionadas por el usuario. A continuación, se presenta una descripción ampliada de esta interfaz:

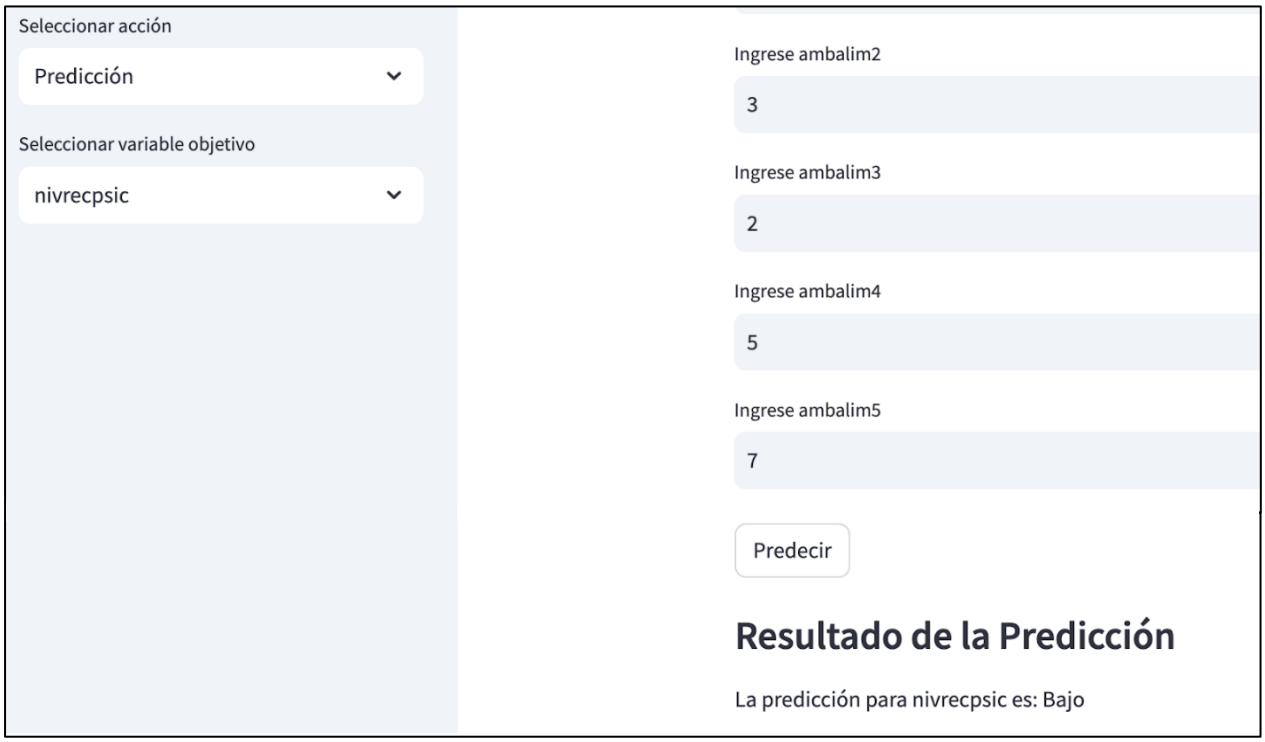

Figura 6. Interfaz en Streamlit [50]

9.1. DESCRIPCIÓN DE LA INTERFAZ

Esta interfaz de Streamlit está diseñada para realizar predicciones basadas en varios parámetros proporcionados por el usuario. A continuación, se explica detalladamente cómo utilizar esta interfaz.

Primero, selecciona la acción deseada en el menú desplegable "Seleccionar acción" y elige "Predicción", indicando así que deseas realizar una predicción basada en los datos proporcionados. Luego, en el menú desplegable "Seleccionar variable objetivo", selecciona "nivrecpsic", que es la variable que se va a predecir, en este caso nivel de recursos psicológicos.

A continuación, ingresa los datos necesarios en los campos correspondientes. Por ejemplo, en el campo ambalim2, que corresponde a la variable de ambiente alimentario universitario, ingresa un valor (en el ejemplo, se ha ingresado '3'); en el campo ambalim3, ingresa otro valor (en el ejemplo, se ha ingresado '2'); en el campo ambalim4, proporciona un valor adicional (en el ejemplo, se ha ingresado '5'); y en el campo ambalim5, ingresa el último valor requerido (en el ejemplo, se ha ingresado '7').

Una vez ingresados todos los valores necesarios, haz clic en el botón "Predecir". La interfaz utilizará estos datos para realizar la predicción. Finalmente, el resultado de la predicción se mostrará en la sección "Resultado de la Predicción". En el ejemplo proporcionado, la predicción para "nivrecpsic" es "Bajo".

Esta interfaz es muy útil para realizar predicciones rápidas basadas en datos específicos ingresados por el usuario.

9.2. FUNCIONAMIENTO GENERAL

El funcionamiento general de la interfaz comienza con el input de usuario, donde este introduce valores numéricos en los campos correspondientes. Luego, la interfaz utiliza estos valores para generar una predicción sobre la variable objetivo seleccionada. Finalmente, el resultado de la predicción se muestra en la interfaz de manera clara para que el usuario pueda interpretarlo fácilmente.

10. CONCLUSIONES Y TRABAJOS FUTUROS

En la presente sección, se presentan las respectivas conclusiones del proyecto y el trabajo futuro que se podría implementar más adelante teniendo en cuenta los comentarios y las pruebas de concepto.

10.1. CONCLUSIONES

Tras un análisis de los datos utilizando técnicas de aprendizaje automático, se han obtenido resultados significativos que arrojan luz sobre la relación entre determinantes sociales y aspectos positivos y negativos de la salud mental de una muestra representativa de la población estudiada. Los modelos desarrollados han demostrado una capacidad considerable para predecir los niveles de ansiedad, depresión, estrés, satisfacción de vida, resiliencia y recursos psicológicos.

Este trabajo es crucial en el contexto actual, donde el bienestar emocional ha cobrado una importancia sin precedentes. Proporciona una visión integral de cómo los factores psicológicos pueden influir en la salud mental de las personas, lo que puede tener implicaciones significativas en la prevención, detección temprana y tratamiento de trastornos mentales.

Se han identificado factores psicológicos específicos que están estrechamente relacionados con diferentes dimensiones del bienestar emocional. Por ejemplo, se observó que niveles más altos de resiliencia están asociados con menores niveles de ansiedad y depresión, mientras que una mayor satisfacción de vida se correlaciona con mayores niveles de optimismo y apoyo social percibido.

En cuanto a los determinantes sociales empleados en el trabajo, se eliminaron de los análisis aquellos que no mostraron una relevancia significativa, manteniéndose sólo los determinantes relevantes.

Los hallazgos de este estudio tienen importantes implicaciones para la salud pública, ya que proporcionan información valiosa para el diseño e implementación de intervenciones dirigidas a mejorar el bienestar emocional y prevenir los trastornos mentales. Al comprender mejor los factores que contribuyen al bienestar emocional, los profesionales de la salud pueden desarrollar estrategias más efectivas para promover la salud mental y el bienestar en la comunidad.

Al considerar todos los análisis realizados, se sugiere que la universidad podría emplear estos modelos para predecir niveles preocupantes de salud mental negativa y positiva entre los estudiantes.

A pesar de los logros obtenidos, este estudio abre la puerta a nuevas investigaciones que podrían profundizar en varios aspectos. Además, sería beneficioso realizar un seguimiento a largo plazo de

los participantes para evaluar la efectividad de las intervenciones diseñadas en base a estos hallazgos.

Finalmente, este estudio tiene algunas limitaciones, como la posible falta de generalización debido a la muestra específica de estudiantes y la exclusión de factores externos que podrían influir en la salud mental. Futuras investigaciones podrían centrarse en la inclusión de estos factores y en el seguimiento a largo plazo de los participantes para evaluar la efectividad de las intervenciones basadas en estos hallazgos. La universidad debería considerar el desarrollo continuo de estos modelos para un monitoreo efectivo de la salud mental de los estudiantes y la implementación de estrategias de apoyo más precisas y personalizadas.

En resumen, este proyecto representa un paso importante hacia la comprensión y promoción del bienestar emocional en la población, y conforma una base para futuras investigaciones y acciones destinadas a mejorar la salud mental y el bienestar en la sociedad.

10.2. TRABAJOS FUTUROS

- **Exploración de Factores Adicionales:** Se recomienda explorar otros factores que pueden influir en el bienestar emocional, como variables contextuales y características individuales. Integrar estas variables en futuras investigaciones podría proporcionar una comprensión más completa de los determinantes del bienestar emocional.
- **Validación Externa de Modelos Predictivos:** Para mejorar la generalización y aplicabilidad de los modelos predictivos desarrollados en este estudio, se sugiere realizar validaciones externas en diferentes poblaciones y entornos. Esto ayudaría a evaluar la robustez y la eficacia de los modelos en diferentes contextos y garantizaría su utilidad práctica en diversos escenarios.
- **Desarrollo de Intervenciones Basadas en Resultados:** Basándose en los hallazgos de este estudio, se pueden diseñar e implementar intervenciones específicas destinadas a promover el bienestar emocional y prevenir los trastornos mentales. Estas intervenciones podrían incluir programas de desarrollo de habilidades de afrontamiento, intervenciones de apoyo social y programas de promoción de la resiliencia.
- **Investigación Longitudinal:** Se sugiere realizar estudios longitudinales para examinar las relaciones entre los factores psicológicos y el bienestar emocional a lo largo del tiempo. Esto permitiría investigar la naturaleza dinámica de estas relaciones y proporcionar información sobre cómo los cambios en los factores psicológicos pueden influir en el bienestar emocional a lo largo de la vida.

11. REFERENCIAS BIBLIOGRÁFICAS

[1] T. Baader. “Diagnóstico de la prevalencia de trastornos de la salud mental en estudiantes universitarios y los factores de riesgo emocionales asociados”. Scielo. Available: https://www.scielo.cl/scielo.php?script=sci_arttext&pid=S0717-92272014000300004. (Accessed Jun. 24, 2023).

[2] M. López. “Características de consultantes y proceso terapéutico de universitarios en un servicio de psicoterapia”. Revista Iberoamericana de psicología: ciencia y tecnología. Available: https://reviberopsicologia.ibero.edu.co/article/view/rip.3110/164. (Accessed Jun. 24, 2023).

[3] C. Chau, “Determinantes de la salud mental en estudiantes universitarios de Lima y Huánuco”. Pontificia Universidad Católica del Perú. Available: https://revistas.pucp.edu.pe/index.php/psicologia/article/view/18789/19010. (Accessed Jun. 24, 2023).

[4] K. Trigueros. “Factores de riesgo que pueden afectar la salud mental de los y las estudiantes de primer a tercer semestre en la universidad del valle sede Zarsal”. Universidad del Valle. Available: https://bibliotecadigital.univalle.edu.co/bitstream/handle/10893/16568/0598411.pdf?sequence=1&isAllowed=y. (Accessed Jun. 24, 2023).

[5] OMS, “Por qué la salud mental debe ser una prioridad al adoptar medidas relacionadas con el cambio climático”. WHO. Available: https://www.who.int/es/news/item/03-06-2022-why-mental-health-is-a-priority-for-action-on-climate-change#:~:text=La%20OMS%20define%20la%20salud,aportar%20algo%20a%20su%20comunid ad%C2%BB (Accessed Jun. 24, 2023).

[6] Congreso de Colombia, LEY 1616 DE 2013, Diario Oficial No. 48.680 de 21 de enero de 2013. Secretaria Senado. Available: http://www.secretariasenado.gov.co/senado/basedoc/ley_1616_2013.html (Accessed Jun. 24, 2023).

[7] Observatorio nacional de salud mental, “Subdirección de Enfermedades No Transmisibles Grupo Funcional: Gestión Integrada para la Salud Mental”. Minsalud. Available: https://www.minsalud.gov.co/sites/rid/Lists/BibliotecaDigital/RIDE/VS/ED/GCFI/guia-ross-saludme ntal.pdf (Accessed Jun. 24, 2023).

[8] J. Zapata, “Intervenciones para la salud mental de estudiantes universitarios durante la pandemia por COVID-19: una síntesis crítica de la literatura”. Scielo. Available: http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0034-74502021000300048#B9. (Accessed Jun. 24, 2023).

[9] P. Acosta, “El bienestar integral y la salud mental: un desafío a nivel mundial”. Pontificia Universidad Javeriana. Available: https://www.javeriana.edu.co/hoy-en-la-javeriana/la-salud-mental-un-desafio-a-nivel-mundial. (Accessed Jun. 24, 2023).

[10] J. Sun, “A Machine-Learning Approach for Predicting Depression Through Demographic and Socioeconomic Features”. IEEE Xplore. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9994921 (Accessed Jun. 24, 2023).

[11] Eustat - Euskal Estatistika Erakundea - Instituto Vasco de Estadística. (s.f.). Población en edad escolar. [Online]. Available: https://www.eustat.eus/documentos/opt_1/tema_47/elem_1450/definicion.html#:~:text=Se%20refiere%20a%20to

[12] I. Soler, “UNA TIPOLOGÍA DE LA POBLACIÓN ESTUDIANTIL UNIVERSITARIA”, RASE, vol. 7, no. 1, pp. 104-122.

[13] Organización de las Naciones Unidas para la Educación, la Ciencia y la Cultura, “EDUCACIÓN SUPERIOR”, IIPE-UNESCO, 2019.

[14] Ministerio de Educación Nacional (2009, julio 20). ¿Qué es la educación superior? [Online] Available: https://www.mineducacion.gov.co/1621/article-196477.html

[15] I. Soler, “UNA TIPOLOGÍA DE LA POBLACIÓN ESTUDIANTIL UNIVERSITARIA”, RASE, vol. 7, no. 1, pp. 104-122.

[16] World Health Organization, “Salud mental: un estado de bienestar”, Ginebra: OMS, 2011. [Online]. Available: http://www.who.int/features/factfiles/mental_health/es/index.html.

[17] Congreso de Colombia, LEY 1616 DE 2013, Diario Oficial No. 48.680 de 21 de enero de 2013. Secretaria Senado. [Online]. Available: http://www.secretariasenado.gov.co/senado/basedoc/ley_1616_2013.html

[18] Ministerio de Salud y Protección Social, “ABECÉ sobre la salud mental, sus trastornos y estigma”, República de Colombia, 2014.

[19] Organización Mundial de la Salud, “Informe sobre la salud mental en el mundo”, OMS, Ginebra, 2001. [Online]. Available: https://www.fundacion-salto.org/wp-content/uploads/2018/10/INFORME-SOBRE-LA-SALUD-MENTAL-ENEL-MUNDO.pdf.

[20] V. Patel y A. Kleinman, “Poverty and common mental disorders in developing countries”, Bulletin of the World Health Organization, vol. 81, no. 8, pp. 609-615, 2003. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2572527/pdf/14576893.pdf.

[21] G. L. Engel, “The need for a new medical model: A challenge for biomedicine”, Science, vol. 196, no. 4286, pp. 129-136, 1977. [Online]. Available:

https://www.science.org/doi/10.1126/science.847460?url_ver=Z39.88-2003&rfr_id=ori:rid:crossref.org&rfr_dat=cr_pub%20%200pubmed.

[22] J. Zubin y B. Spring, “Vulnerability: A new view of schizophrenia”, Journal of Abnormal Psychology, vol. 86, no. 2, pp. 103-126, 1977.

[23] S. M. Monroe y A. D. Simons, “Diathesis-stress theories in the context of life stress research: Implications for the depressive disorders”, Psychological Bulletin, vol. 110, no. 3, pp. 406-425, 1991. [Online]. Available: https://psycnet.apa.org/record/1992-05606-001.

[24] B. Baradwaj y S. Pal, “Mining Educational Data to Analyze Students' Performance”, International Journal of Advanced Computer Science and Applications, vol. 2, no. 6, 2011.

[25] J. Zárate-Valderrama, N. Bedregal-Alpaca y V. Cornejo-Aparicio, “Modelos de clasificación para reconocer patrones de deserción en estudiantes universitarios”, Ingeniare. Revista chilena de ingeniería, vol. 29, no. 1, pp. 168-177, 2021. [Online]. Available: https://dx.doi.org/10.4067/S0718-33052021000100168.

[26] E. Mayr, “The Growth of Biological Thought: Diversity, Evolution, and Inheritance”, Harvard University Press, 1982.

[27] R. Caruana y A. Niculescu-Mizil, “An empirical comparison of supervised learning algorithms”, in Proceedings of the 23rd International Conference on Machine Learning, pp. 161-168, 2006.

[28] American Psychological Association, “Manual Diagnóstico y Estadístico de los Trastornos Mentales (DSM-5)”, Editorial Médica Panamericana, 2013. [Online]. Available: https://www.federaciocatalanatdah.org/wp-content/uploads/2018/12/dsm5-manualdiagnsticoyestadisticodelostrastornosmentales-161006005112.pdf.

[29] Real Academia Española, “algoritmo”, Diccionario de la lengua española, [Online]. Available: https://dle.rae.es/algoritmo.

[30] T. Hastie, R. Tibshirani y J. Friedman, “The Elements of Statistical Learning: Data Mining, Inference, and Prediction”, Springer Science & Business Media, 2009. [Online]. Available: https://www.sas.upenn.edu/~fdiebold/NoHesitations/BookAdvanced.pdf.

[31] C. M. Bishop, “Pattern Recognition and Machine Learning”, Springer, 2006. [Online]. Available: https://github.com/peteflorence/MachineLearning6.867/blob/master/Bishop/Bishop%20-%20Pattern%20Recognition%20and%20Machine%20Learning.pdf.

[32] T. H. Davenport y L. Prusak, “Working knowledge: How organizations manage what they know”, Harvard Business School Press, 1998. [Online]. Available: http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=7259.

[33] Ministerio de Ciencia Tecnología e Innovación. (2022, julio 9). Definiciones y Conceptos Básicos. [Online]. Available: https://red-documentacion.minciencias.gov.co/Gestion_Datos_Investigacion/gestion-datos

[34] D. Torres-Salinas, N. Robinson-García y Á. Cabezas-Clavijo, "Compartir los datos de investigación en ciencia: introducción al data sharing", Profesional De La Información Information Professional, vol. 21, no. 2, pp. 173–184, 2012. DOI: 10.3145/epi.2012.mar.08.

[35] Grupo de Trabajo de "Depósito y Gestión de datos en Acceso Abierto" del proyecto RECOLECTA, "La conservación y reutilización de los datos científicos en España. Informe del grupo de trabajo de buenas prácticas", Fundación Española para la Ciencia y la Tecnología, FECYT, Madrid, 2012. [Online]. Available: www.fecyt.es.

[36] National Science Foundation, "Cyberinfrastructure Vision for 21st Century Discovery", 2007. [Online]. Available: http://www.nsf.gov/pubs/2007/nsf0728/index.jsp.

[37] H. M. Stallman, "Psychological distress in university students: A comparison with general population data", Australian Psychologist, vol. 45, no. 4, pp. 249–257, 2010. [Online]. DOI: 10.1080/00050067.2010.482109.

[38] R. P. Auerbach et al., "Mental disorders among college students in the World Health Organization World Mental Health Surveys", Psychological Medicine, vol. 46, no. 14, pp. 2955-2970, 2016. [Online]. DOI: 10.1017/S0033291716001665.

[39] T. H. Davenport y R. Kalakota, "The AI advantage: How to put the artificial intelligence revolution to work", MIT Press, 2019. [Online]. DOI: 10.7551/mitpress/11781.001.0001.

[40] A. B. R. Shatte, D. M. Hutchinson y S. J. Teague, "Machine learning in mental health: A scoping review of methods and applications", Psychological Medicine, vol. 49, no. 9, pp. 1426-1448, 2019. [Online]. DOI: 10.1017/S0033291719000164.

[41] I. Roy, R. Rivas, M. Pérez y L. Palacios, "Correlación: no toda correlación implica causalidad", Alergia México, vol. 66, no. 3, pp. 354-360, 2019. DOI: https://doi.org/10.29262/ram.v66i3.651

[42] Bengfort, B., Bilbro, R., & Ojeda, T. (2018). "Applied Text Analysis with Python: Enabling Language-Aware Data Products with Machine Learning". O'Reilly Media, Inc.

[43] Reshef, D. N., et al. "Detecting novel associations in large datasets". Science, vol. 334, no. 6062, pp. 1518-1524, 2011.

[44] A. F. Brumberg and R. V. K. Johnson, "On the Problem of Multicollinearity: A New Approach Towards Canonical Parameter Orthogonalization", *American Journal of Sociology*, vol. 76, no. 4, pp. 734-738, 1970.

[45] Belsley, D. A. (1991). “Conditioning Diagnostics: Collinearity and Weak Data in Regression”. *John Wiley & Sons.*

[46] Kutner, M. H., Nachtsheim, C. J., & Neter, J. (2004). “Applied Linear Regression Models”. *McGraw-Hill.*

[47] Fox, J. (1992). “Regression Diagnostics: An Introduction”. *SAGE Publications.*

[48] Gujarati, D. N. (2003). “Basic Econometrics” (4th ed.). McGraw-Hill Education.

[49] A. Chaparro, G. Ruiz y J. Charry. (2024). “Distribución y detalle de correlación”.

[50] A. Chaparro, G. Ruiz y J. Charry. (2024). “Estudio de Estudio de proyecto aplicado”.

12. ANEXOS

12.1. ANEXO 1. BD ESTUDIANTES MCD

12.2. ANEXO 1. DICCIONARIO DE VARIABLES (BD MCD)