

**MODEL DX-TDAH” MODELO PARA APOYAR EL DIAGNÓSTICO
TEMPRANO DE TDAH CON TÉCNICAS DE MACHINE LEARNING**

Autores:

CASTILLO ROSERO LIZETH BEATRIZ

CALZADA RIVAS KEVIN GIANCARLO

PROGRAMA DE INGENIERÍA DE SISTEMAS

DEPARTAMENTO DE SISTEMAS

FACULTAD DE INGENIERÍA

UNIVERSIDAD DE NARIÑO

AGOSTO, 2023

MODELDX-TDAH” MODELO PARA APOYAR EL DIAGNÓSTICO TEMPRANO DE TDAH CON TÉCNICAS DE MACHINE LEARNING

Autores:

CASTILLO ROSERO LIZETH BEATRIZ, lizcas97@hotmail.com

CALZADA RIVAS KEVIN GIANCARLO, giancarloscalzada@gmail.com

Informe final de trabajo de grado presentado como requisito para optar al título de Ingeniero de
Sistemas, **en modalidad investigación**

Director/Asesor

MORA PAZ HÉCTOR ANDRÉS

Codirector/Co-asesor

RIVERA ROSERO JORGE ALBEIRO

PROGRAMA DE INGENIERÍA DE SISTEMAS

DEPARTAMENTO DE SISTEMAS

FACULTAD DE INGENIERÍA

UNIVERSIDAD DE NARIÑO

AGOSTO, 2023

Nota exclusión de responsabilidad intelectual

“Las ideas y conclusiones aportadas en este Trabajo de Grado son responsabilidad de los autores”.

Artículo 1° del Acuerdo No. 324 de octubre 11 de 1966, emanado del Honorable Consejo Directivo de la Universidad de Nariño.

Nota de Aceptación

Firma del director

Firma del jurado evaluador

Firma del jurado evaluador

Ciudad y fecha de firma

AGRADECIMIENTOS

Primeramente, a Dios por darnos la sabiduría y paciencia que hemos necesitado, porque, aunque muchas veces intentamos tirar todo por la borda el nunca permitió que nuestra fe se terminara y hoy por hoy nos permitió darnos la vida para culminar con esta una de nuestras grandes metas.

A nuestros padres por siempre confiar en nosotros, ayudarnos en todo lo que hemos necesitado, pero sobre todo apoyarnos en todas y cada una de las cosas que hacíamos.

A nuestros abuelos porque siempre confiaron en nosotros, dándonos sus palabras de apoyo y aliento.

Al resto de nuestra familia porque de una u otra manera fueron participe de este trabajo, por tenernos la paciencia necesaria y darnos el aliento necesario para seguir adelante.

A nuestras amigas Lina Flores y Laura Montaña porque siempre nos ayudaron, apoyaron, y vivieron esta experiencia de cerca con nosotros de buena manera.

A esta gran institución como lo es la Universidad de Nariño por darnos la herramienta y la oportunidad de convertirnos en ingenieros en sistemas.

Y, por último, pero no menos importante a nuestro asesor de trabajo de grado Héctor Andrés Mora. sin él, sus virtudes, su paciencia y constancia no hubiéramos podido lograrlo. Sus consejos fueron siempre de gran ayuda, pues muchas veces no sabíamos ni cómo hacer las cosas y él con sus palabras y su guía nos daba ideas y podíamos seguir adelante. Ha formado parte muy importante de este proyecto con sus aportes y conocimientos profesionales que siempre lo han caracterizado. Gracias por sus orientaciones.

KEVIN GIANCARLO CALZADA RIVAS y LIZETH BEATRIZ CASTILLO ROSERO

DEDICATORIA

A Dios por darme todas las herramientas para culminar con esta etapa de formación.

A esas personas que hoy no están conmigo, pero fue un pilar fundamental apoyándome y sobre todo ayudando a mi madre en mi formación, a mi abuela Beatriz Márquez, sé que desde donde este está orgulloso de saber que hoy me convierto en una profesional.

A mi abuelo Virgilio Rosero porque fue él quien asumió el rol de padre en mi formación y ayudo a mi abuela y mi madre a hacer de mi la persona que hoy en día soy.

A mi madre Betty Rosero porque fue ella quien se me ha esforzado para lograr que hoy en día sea una profesional y por brindarme su comprensión y tolerancia para lograr la culminación de esta etapa.

LIZETH BEATRIZ CASTILLO ROSERO

A Dios por darme la fortaleza y la capacidad mental de afrontar mis retos y culminar mis estudios.

A mis padres Carmen Rivas y William Calzada porque han sido ellos quienes se han esforzado al máximo para lograr de mí una persona de bien y sobre todo que yo pueda ser hoy profesionales. Porque cuando no se quería seguir fueron ellos con sus palabras me alentaron a que siguiera y luchara por lo que quería, ser hoy profesional.

KEVIN GIANCARLO CALZADA RIVAS

RESUMEN

El trastorno por déficit de atención e hiperactividad (TDAH) es un trastorno neurobiológico que se presenta en personas de todas las edades. El principal síntoma de este trastorno es que las personas tienen problemas para mantener su atención y controlar sus impulsos, pero esto no significa que todas las personas que padezcan este trastorno lo manifiesten de la misma manera. En los últimos años, se han llevado a cabo investigaciones importantes para el diagnóstico cuantitativo de este trastorno donde se utilizan imágenes médicas y técnicas de procesamiento de señales.

Actualmente, se puede observar un crecimiento en los estudios relacionados con la inteligencia artificial, muchos de los cuales están relacionados con herramientas para el diagnóstico de diferentes enfermedades. Con el objetivo de contribuir a un diagnóstico más preciso de este trastorno, se planteó la creación de una herramienta capaz de diagnosticar el trastorno de déficit de atención e hiperactividad mediante técnicas de aprendizaje automático. Para ello, se construyó una herramienta para el apoyo al diagnóstico temprano de TDAH, utilizando algoritmos de Machine Learning. Para el entrenamiento de cada algoritmo, se utilizaron neuroimágenes de resonancias magnéticas funcionales (fMRI) que se preprocesaron utilizando el filtro Smith. Posteriormente, se seleccionó un modelo que pasó por una serie de configuraciones y pruebas, con las cuales se determinó que tenía mejores métricas de calidad para la predicción de esta enfermedad.

Los aspectos anteriores se llevaron a cabo mediante un conjunto de neuroimágenes llamado ADHD-200, el cual es de uso libre. También se usó la metodología para el descubrimiento de conocimiento en bases de datos (KDD), fundamentada en tres fases principales: Pre procesar los datos de ADHD-200 para conseguir un conjunto de conectomas cerebrales, estructurar una ruta de configuración justa para un conjunto de algoritmos de Aprendizaje Automático para seleccionar el mejor modelo, y finalmente, construir una interfaz para el apoyo al diagnóstico de TDAH utilizando imágenes de resonancia magnética funcional.

Palabras Clave: Déficit de atención e hiperactividad (TDAH), imágenes de resonancia magnética funcional (fMRI), ADHD-200, Machine Learning, Deep Learning.

ABSTRACT

The Attention Deficit Hyperactivity Disorder (ADHD) is a neurobiological disorder that presents itself in individuals of all ages. The primary symptom of this disorder is that individuals have difficulties sustaining their attention and controlling their impulses, but this does not mean that all individuals experiencing this disorder manifest it in the same way. In recent years, significant research has been conducted for the quantitative diagnosis of this disorder using medical images and signal processing techniques.

Currently, there is a growth in studies related to artificial intelligence, many of which are connected to tools for diagnosing different diseases. With the aim of contributing to a more precise diagnosis of this disorder, the creation of a tool capable of diagnosing Attention Deficit Hyperactivity Disorder through machine learning techniques was proposed. To achieve this, a tool was developed to support the early diagnosis of ADHD using machine learning algorithms. For the training of each algorithm, functional magnetic resonance imaging (fMRI) neuroimages were used, which were preprocessed using the Smith filter. Subsequently, a model was selected that underwent a series of configurations and tests, through which it was determined to have better quality metrics for predicting this disease.

The aforementioned aspects were carried out using a set of neuroimages called ADHD-200, which is freely available. The methodology for knowledge discovery in databases (KDD) was also employed, based on three main phases: preprocessing the ADHD-200 data to obtain a set of brain connectomes, structuring a fair configuration path for a set of machine learning algorithms to select the best model, and finally, constructing an interface for supporting the diagnosis of ADHD using functional magnetic resonance images.

Keywords: Attention Deficit Hyperactivity Disorder (ADHD), functional magnetic resonance imaging (fMRI), ADHD-200, Machine Learning, Deep Learning.

CONTENIDO

	Pág.
INTRODUCCION	15
I. CONTEXTUALIZACIÓN.....	17
A. Línea de investigación	17
B. Planteamiento del problema	17
C. Justificación	19
D. Objetivos	20
II. MARCO TEÓRICO	22
E. Marco de antecedentes	22
F. Fundamentos teóricos	26
III. MATERIALES Y MÉTODOS	34
A. Materiales	34
B. Métodos	35
C. Selección de datos	36
D. Preprocesamiento de datos adhd-200	37
E. Transformacion de datos adhd-200	37
F. Minería de datos – selección del algoritmo	37
G. Elección y aplicación del algoritmo	38
H. Evaluación	38
I. Interpretación	38
IV. RESULTADOS DE LA INVESTIGACIÓN	41
A. Preprocesar los datos del conjunto ADHD 200	41
B. Estructurar a partir de un conjunto de técnicas de aprendizaje automático y métricas de calidad una ruta base de configuración objetiva.	49
C. Seleccionar la técnica de aprendizaje supervisado que tenga el mejor compromiso en su exactitud y coste computacional e interpretabilidad de resultados.	53
D. Implementar de la interfaz web.	56
ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS.....	62

CONCLUSIONES63
RECOMENDACIONES64
BIBLIOGRAFÍA.....65
ANEXOS.....69

LISTA DE FIGURAS

	Pág.
Fig. 1. Resonancias magnéticas extraídas desde repositorio proporcionado por NITRC.	26
Fig. 2. Capa de red neuronal completamente conectada.	29
Fig. 3. Redes neuronales recurrentes.	29
Fig. 4. Funcionamiento redes LSTM	30
Fig. 5. Proceso metodología KDD	36
Fig. 6. Gráfico de preprocesamiento con la herramienta FmriPrep	42
Fig. 7. Mascara cerebral y segmentación del tejido cerebral.	43
Fig. 8. Normalización espacial de la referencia anatómica	43
Fig. 9. Alineación de datos de resonancia magnética funcionales y anatómicos (impulsados por superficie)	44
Fig. 10. Mascara cerebral y ROI de CompCor (Anatómico/temporal)	45
Fig. 11. Archivos resultantes del preprocesamiento en el directorio anat.	45
Fig. 12. Archivos resultantes del preprocesamiento en el directorio func	46
Fig. 13. Contraste cálculo de semilla	47
Fig. 14. Mascara Smith atlas	47
Fig. 15. Matriz resultante para evaluación de los modelos	48
Fig. 16. Diagrama de marco experimental	49
Fig. 17. Diagrama de ejecución de algoritmos de machine learning.	50
Fig. 18. Diagrama de ejecución de algoritmos Deep learning.	51
Fig. 19. Diagrama de pasos para predicción	52
Fig. 20. Configuración de algoritmos utilizando validación cruzada	53
Fig. 21. Grafica de evolución del algoritmo GRU con el conjunto de datos ADHD-200	56
Fig. 22. Arquitectura cliente servidor	58
Fig. 23. Función para hacer la predicción de la neuroimagen	59
Fig. 24. Creación de ruta para mostrar la predicción a la interfaz web	59
Fig. 25. Interfaz web para la predicción de pacientes con TDAH	60
Fig. 26. Proceso de validación según criterio medica del paciente	61

LISTA DE TABLAS

	Pág.
TABLA I ANTECEDENTE (Arah Itani et al,2019)	22
TABLA II ANTECEDENTE (Zhenyu Mao et al,2019).....	22
TABLA III ANTECEDENTE (Amandeep kaur et al,2022)	23
TABLA IV ANTECEDENTE (Zhenyu Mao et al,2019)	24
TABLA V. ANTECEDENTE (Chaitra, Shreya, Rajendra, Hojjat, Muralidhar et al,2017)	24
TABLA VI ANTECEDENTE (Javier Costa Rosa et al,2017)	25
TABLA VII ANTECEDENTE (PATRICIA AMADO CABALLERO et al,2019)	25
TABLA VIII DICCIONARIO DEL CONJUNTO ADHD-200.....	34
Tabla IX. BIBLIOTECAS UTILIZADAS EN LA INVESTIGACIÓN.....	35
TABLA XI PLAN DE ACCION	39
TABLA XII DESCRIPCIÓN DE RESULTADOS DE CADA UNO DE LOS MODELOS EXPERIMENTOS.....	54

LISTA DE ANEXOS

	Pág.
Anexo 1 Vista principal de prototipo de página web de la investigación	78
Anexo 2 Vista para agregar datos personales de paciente	78
Anexo 3 Vista de cómo se realiza el preprocesamiento con la herramienta FmriPrep	79
Anexo 4 Vista de preprocesamiento botón LEER MAS	79
Anexo 5 Visitar repositorio de MODEL-DX y los experimentos	80
Anexo 6 Vista del botón Github	80
Anexo 7 Vista del apartado VALIDAR	81
Anexo 8 Base de datos ADHD-200 descargado de manera local	81
Anexo 9 Base de datos de pacientes validadas por criterio medico	82
Anexo 10 Experimento sin pipeline	82
Anexo 11 continuación de experimento, mejores hiperparametros	83
Anexo 12 Continuación del experimento matriz de confusión	83
Anexo 13 Continuación del experimento, curva de roc	84
Anexo 14 Continuación del experimento grafica de recall	84
Anexo 15 Vista de conjunto de datos descargado y almacenado de forma local	85
Anexo 16 Conjunto de dato con todas particiones	85
Anexo 17 Comando para instalar la imagen de FmriPrep en Docker	86
Anexo 18 instalación de FmriPrep en Ubuntu	86
Anexo 19 Registro para adquirir licencia para el uso de FreeSurfer.....	87
Anexo 20 Ubicación de licencia proporcionada por FreeSurfer	88
Anexo 21 Comando para ejecutar FmriPrep	88
Anexo 22 descripción de comando para el preprocesamiento de las neuroimágenes	89

INTRODUCCION

“El Trastorno por Déficit de Atención e Hiperactividad TDAH es un trastorno neurobiológico que aqueja tanto a adultos como a niños”[1]. Es descrito como un patrón persistente o continuo de inatención y-o hiperactividad e impulsividad que impide las actividades diarias o el desarrollo típico de la persona. cuya complejidad para diagnosticarla con una buena exactitud mediante bitácoras psicológicas, se da debido a las diferentes maneras que este trastorno se manifiesta en los pacientes. Por esa razón, en los últimos años, se han llevado a cabo investigaciones importantes para el diagnóstico cuantitativo de este trastorno utilizando imágenes médicas y técnicas de procesamiento de señales.

El TDAH, como lo indica su nombre es un trastorno el cual puede catalogarse como crónico, este trastorno se puede dar en la infancia o edad adulta. Su principal síntoma en niños incluye la falta de atención, dificultades para organizar su tiempo y tareas, bajo rendimiento escolar y desobediencia. En el caso de los adultos, “el TDAH puede dificultar la realización de tareas cotidianas y la organización de sus obligaciones”[2], lo que puede afectar su vida social y laboral, estos síntomas se manifiestan de manera intensa, grave o moderada. La forma correcta de identificar este trastorno es utilizando pruebas clínicas; pero identificarlo es un proceso largo y tedioso, porque se requiere por mucho tiempo analizar el entorno del paciente, sea familiar o académico. Aparte se necesita de más pruebas realizadas por especialistas como psiquiatras, neurólogos, etc. “Este trastorno en ocasiones puede permitir que el paciente tenga problemas de baja autoestima, problemas en su relación, trabajo o escuela por su bajo rendimiento”[3].

Actualmente, se puede observar un crecimiento en los estudios relacionados con el apoyo a diagnóstico médico de enfermedades utilizando técnicas de aprendizaje automático, entre ellos el diagnóstico de *TDAH*. Esta investigación se alinea al anterior precedente, utilizando neuroimágenes de resonancias magnéticas funcionales (*FMRI*) del repositorio ADH-2000 las cuales se preprocesaron, para obtener conectomas cerebrales como entrada a diferentes algoritmos de aprendizaje automático, entre ellos, árboles de decisión, bosques aleatorios, máquinas vectoriales, regresión logística, LSTM y GRU, combinados con PCA y Standard Scaler. Esto se hizo con el objetivo de buscar el mejor compromiso en cuanto a su exactitud, el área bajo la curva ROC, la sensibilidad y el coste computacional.

De la anterior experimentación se encontró, que las transformaciones de escalado y reducción de dimensión no mejoran el compromiso y que el algoritmo GRU obtuvo los mejores resultados.

I. CONTEXTUALIZACIÓN

A. Línea de investigación

En este trabajo de grado se relaciona el tratamiento de datos, programación de algoritmos, inteligencia artificial y específicamente aprendizaje automático.

La línea de investigación en la que está suscrito es en la línea de software y manejo de información, en la construcción de aplicaciones mediante programación; descubrimiento de conocimiento en bases de datos KDD como herramienta metodológica para garantizar la adquisición, limpieza y transformación de datos y minería usando inteligencia artificial específicamente machine learning mediante obtención de resultados con algoritmos de clasificación.

B. Planteamiento del problema

Más del 4% de la población mundial tiene TDAH. El Trastorno por déficit de atención con hiperactividad (TDAH) se caracteriza por la dificultad para mantener la atención, la presencia de impulsividad y un aumento en la actividad motora que puede presentarse como una sensación constante de inquietud e imposibilidad de relajarse. En Estados Unidos la prevalencia del trastorno es del 4 al 6% (entre 8 y 9 millones de adultos) y según la Organización Mundial de la Salud más del 4% de la población mundial tiene TDAH; El trastorno por déficit de atención e hiperactividad constituye un modelo persistente o continuo de inatención y/o hiperactividad e impulsividad, que impide las actividades diarias o el desarrollo típico, pues provoca dificultades para mantener la atención, la función ejecutiva y la memoria de trabajo.

El TDAH es definido como un trastorno de carácter neurobiológico el cual se origina en la infancia presentando un déficit de atención, hiperactividad e impulsividad. Para su diagnóstico se utilizan, test de Atención D2, Escala de Magallanes de atención visual, EDAH (Evaluación del trastorno para el déficit de Atención e Hiperactividad), Test de desórdenes de atención e hiperactividad, Escala de Magallanes de Impulsividad Computarizadas (EMIC).

Es diagnóstico puede ser asistidos por artefactos tecnológicos producidos mediante técnicas de inteligencia artificial como el machine learning, el cual permite que las máquinas aprendan sin ser expresamente programadas para ello. Una habilidad indispensable para hacer sistemas, no solo inteligentes, sino autónomos, y capaces de identificar patrones entre los datos para hacer predicciones. “machine learning son técnicas de reconocimiento de patrones, y consiste en tener una base datos que trabaja con un algoritmo capaz de extraer conjuntos de datos para los que no ha sido entrenado previamente”[4].

El TDAH a nivel mundial es alto. Las estadísticas epidemiológicas muestran que 3 al 5% de los niños en edades de 9 a 14 años pueden recibir este diagnóstico. En el contexto clínico, alrededor de 30% de los pacientes que son valorados por primera vez en los servicios de psiquiatría infantil presentan problemas de inatención, impulsividad o hiperactividad. En la actualidad se admite que el TDAH es la patología neuro comportamental infantil y juvenil más frecuente y presenta una alta prevalencia a nivel mundial, En diversos estudios también se han descrito altas tasas de prevalencia. En algunos países se utilizan los criterios de diagnósticos del DSM-5; en Brasil, analizaron un total de 1013 estudiantes entre 12 y 14 años y observaron que el 87% de los portadores de TDAH eran repitentes de al menos un grado escolar, comparado con un 30% de los no portadores. Así mismo, encontraron que el 48% del primer grupo habían sido suspendidos al menos una vez y 17% habían sido incluso expulsados de escuelas anteriores.

En Colombia, las cifras de prevalencia se encuentran alrededor del 17 por ciento. “El Ministerio de Educación Colombiano reglamentó la organización del servicio de apoyo pedagógico para la oferta de educación inclusiva a los estudiantes que encuentran barreras para el aprendizaje”[5], la participación por su condición de discapacidad y a los estudiantes con capacidades o con talentos excepcionales, matriculados en los establecimientos educativos estatales. Dentro de los aspectos que trató dicha norma se encuentran unas definiciones, principios y responsabilidades de las entidades territoriales para se les debe brindar una educación inclusiva, la cual persigue que no existan ambientes segregados, sino que todos los niños y niñas, independientemente de sus necesidades educativas, puedan estudiar y aprender juntos, esto es, que la enseñanza se adapte a los alumnos y no éstos a la enseñanza, sin olvidar que en caso de ser necesario se deberán proporcionar las ayudas especiales que se demanden para optimizar su proceso de aprendizaje y desarrollar plenamente sus potencialidades, siempre con la concurrencia conjunta de la familia, la sociedad, el Estado y las instituciones educativas.

Actualmente el 30% de los pacientes que acuden en el contexto clínico por primera vez, presentan problemas de TDAH además, se podría decir que a este trastorno no se le presta mucha atención debido a que las personas lo asocian al comportamiento normal especialmente en niños entre 9 y 14 años, lo que hace que en un futuro allá más deserción escolar, puesto que los niños que sufren de este trastorno al no ser diagnosticados temprano se les dificultara alcanzar un nivel de estudios de competencias acorde a los estándares normales. Si este trastorno no es diagnosticado a tiempo puede causar problemas serios problemas en el desarrollo del niño, entre ellos, dificultades en la adaptación social y como resultado, serios desajustes emocionales, comportamientos conflictivos.

Teniendo en cuenta los anteriores factores esta herramienta facilitara el diagnóstico temprano del trastorno de déficit de atención e hiperactividad. Utilizando una aplicación orientada a la web la cual permitiría acceder desde cualquier lugar del mundo acortando distancia, mediante técnicas de inteligencia artificial específicamente de machine learning.

C. Justificación

Este proyecto se realizó para facilitar el proceso de diagnósticos temprano de trastornos por déficit de atención, hiperactividad o impulsividad (TDAH). “TDAH debe ser entendido como un factor de riesgo evolutivo, es decir, que ensombrece el futuro de quien lo sufre.”[6], Los niños con este déficit parecen no prestar atención cuando les hablan, se le dificulta tener el control de su tiempo, realizar sus deberes a tiempo dando a entender que son desordenados y desobedientes, su rendimiento escolar es bajo comparado con los otros niños y pueden perder hasta el año escolar; los adultos con TDAH que de niño no fueron tratados correctamente las tareas cotidianas pueden resultarles difícil y tener dificultades para establecer prioridades, olvidándose de reuniones o eventos sociales. “El adulto con TDAH, se distrae con facilidad, olvida con frecuencias sus obligaciones y es incapaz de realizar actividades que exigen concentración.” [2].

Los anteriores aspectos fueron posibles gracias a qué se obtuvo un conjunto de datos de TDAH de uso libre para poder hacer una serie de procesos como limpieza del ruido del conjunto de datos mediante estadísticos de corrección además, se realizó estudios para el diagnóstico temprano de TDAH como “Un modelo de predicción general para la detección de TDAH y autismo mediante resonancia magnética estructural y funcional”[7], y mediante técnicas de inteligencia artificial

demostrando que es posible extrapolar estos datos mediante modelo de machine learning. Gracias a que los métodos para la obtención de modelos están extendidos a nivel mundial hace posible mejorar los estudios realizados anteriormente utilizando técnicas más robustas. Existen diversas formas de acoplamiento de los modelos de machine learning a interfaces web dando un abanico plausible para desarrollar la herramienta propuesta.

Esta investigación además ayudo para el diagnóstico temprano de trastorno de déficit de atención e hiperactividad (TDAH) en los niños, se utilizaron algoritmos de aprendizaje automático para trabajar en conjuntos con modelos de clasificación con redes neuronales artificiales, scripts de depuración para el conjunto de datos seleccionado, metodología de comparación de algoritmos de machine learning y acoplamiento de modelos en una interfaz web, también sea hizo un aporte significativo al departamento de ingeniería de sistemas, a sus grupos de investigación Galeras.net y Grias kdd en cuanto al desarrollo de aplicaciones inteligentes.

En comparación con otros aplicativos orientado a la web esto se llevó a cabo con ayuda de la metodología KDD la cual nos permitió garantizar la adquisición, limpieza, transformación y minería de datos utilizando machine learning permitiendo un entorno adaptativo a las nuevas situaciones conforme reciba datos, hay autores como Fayyad, Piatetsky-Shapiro y Smith que definen KDD como “El proceso no trivial de identificación de patrones válidos, novedosos, potencialmente útiles y fundamentalmente entendibles al usuario a partir de los datos”[8], además no se encontraron herramientas web de diagnóstico temprano de TDAH en Nariño.

D. Objetivos

1) Objetivo general

Apoyar el diagnóstico temprano del TDAH a través de la construcción de un modelo de aprendizaje supervisado utilizando técnicas de machine learning.

2) Objetivos específicos

Preprocesar los datos del conjunto de datos ADHD-200 para reducir el ruido y obtener las variables más relevantes mediante técnicas de extracción, transformación, carga y limpieza (ETLC).

Estructurar a partir de un conjunto de técnicas de aprendizaje automático y métricas de calidad una ruta base de configuración objetiva.

Seleccionar la técnica de aprendizaje supervisado que tenga el mejor compromiso en su exactitud y coste computacional e interpretabilidad de resultados.

Implementar un prototipo de aplicación web para el consumo del mejor modelo obtenido como interfaz de validación médica y apoyo al diagnóstico de TDAH.

II. MARCO TEÓRICO

A continuación, se abordan los fundamentos epistemológicos teóricos e investigaciones que existen los cuales darán soporte científico a esta presente investigación.

E. Marco de antecedentes

1) Antecedentes internacionales

TABLA I ANTECEDENTE (Arah Itani et al,2019)

Modelos de aprendizaje automático interpretables para la ayuda al diagnóstico: un estudio de caso sobre el trastorno por déficit de atención / hiperactividad

Autor (as, es)	Arah Itani, Mandy Rossignol, Fabián Lecron, Philippe Fortemps
Fecha	25 de abril del 2019
País	Romania
Resumen	Este trastorno por déficit de atención es un trastorno que tiene unas consecuencias graves en el bienestar del niño, actualmente la evaluación de este trastorno se respalda en evaluaciones clínicas y pruebas escritas. Se dice que actualmente hay mucha investigación acerca del TDAH en curso con el fin de determinar la base neurofisiológica de este trastorno y tener algo más objetivo. este trabajo propone una metodología ML el cual tiene como objetivo lograr un equilibrio justo entre las necesidades de rendimiento y la interpretabilidad que se esperan de los sistemas de ayuda al diagnóstico médico. Esta metodología fue aplicada sobre una muestra de datos extraídos de la colección ADHD-200, mediante el desarrollo de árboles de decisión que son valorados por su legibilidad este análisis indica la relevancia del sistema límbico para el diagnóstico de este trastorno[9].
Aspectos útiles al presente estudio	Este proyecto aporta la forma que fue aplicada la metodología sobre los datos extraídos de la colección ADHD-200 mediante el desarrollo de árboles de decisión que son valorados por la legibilidad

Fuente: Plos One

TABLA II ANTECEDENTE (Zhenyu Mao et al,2019)

Método de aprendizaje profundo espacio temporal para la clasificación de TDAH fMRI

Autor (as, es)	Zhenyu Mao, Yi Su, Guangquan Xu, Xueping Wang, Yu Huang, Weihua Yue
Fecha	octubre del 2019
País	Estados unidos

Resumen	El Trastorno por Déficit de Atención e Hiperactividad (TDAH) es un tipo de trastorno del neurodesarrollo común en niños. Debido a la complejidad del mecanismo patológico, hasta ahora existe una falta de métodos de diagnóstico objetivos. Este artículo tiene como objetivo proponer un método automático de diagnóstico del TDAH utilizando datos de resonancia magnética funcional en estado de reposo (rs-fMRI) con modelos de aprendizaje profundo espacio temporales. A diferencia de los métodos tradicionales, este artículo construyó un método de aprendizaje profundo llamado CNN 4-D basado en cómputo granular, que fue entrenado en función de cambios derivados en la entropía, y puede calcular la granularidad a un nivel grueso mediante la superposición de capas. Se considera la estructura de rs-fMRI como secuencias temporales de cuadros en 3D, se propusieron varios modelos de cómputo granular y fusión espacial y temporal, incluyendo el agrupamiento de características, memoria a corto y largo plazo (LSTM) y convolución espacio temporal. Este artículo presentó un enfoque para aumentar el conjunto de datos que puede muestrear los cuadros rs-fMRI de un sujeto en varias partes relativamente cortas con un paso fijo. Se utilizó el conjunto de datos público del Consorcio ADHD-200 para entrenar y validar nuestro método. Y los resultados de las evaluaciones mostraron que nuestro método superó a los métodos tradicionales en el conjunto de datos (precisión: 71.3%, AUC: 0.80). Por lo tanto, nuestro método CNN 4-D puede utilizarse para construir una herramienta de diagnóstico asistido automático más preciso para el TDAH[10].
Aspectos útiles al presente estudio	Este proyecto aporta la forma que fue aplicada la metodología sobre los datos extraídos de la colección ADHD-200 mediante LSTM que es una red neuronal profunda.

Fuente: sciencedirect

TABLA III ANTECEDENTE (Amandeep kaurt al,2022)

Identificación precisa de TDAH entre adultos utilizando datos de actividad en tiempo real

Autor (as, es)	Amandeep kaur1, Karanjeet Singh Kahlon
Fecha	junio 2022
País	Estados unidos
Resumen	El Trastorno por Déficit de Atención e Hiperactividad (TDAH) es un tipo de trastorno del neurodesarrollo común en niños. Debido a la complejidad del mecanismo patológico, hasta ahora existe una falta de métodos de diagnóstico objetivos. Este artículo tiene como objetivo proponer un método automático de diagnóstico del TDAH utilizando datos de resonancia magnética funcional en estado de reposo (rs-fMRI) con modelos de aprendizaje profundo espacio temporales. A diferencia de los métodos tradicionales, este artículo construyó un método de aprendizaje profundo llamado CNN 4-D basado en cómputo granular, que fue entrenado en función de cambios derivados en la entropía, y puede calcular la granularidad a un nivel grueso mediante la superposición de capas. Se considera la estructura de rs-fMRI como secuencias temporales de cuadros en 3D, se propusieron varios modelos de cómputo granular y fusión espacial y temporal, incluyendo el agrupamiento de características, memoria a corto y largo plazo (LSTM) y convolución espacio temporal. Este artículo presentó un enfoque para aumentar el conjunto de datos que puede muestrear los cuadros rs-fMRI de un sujeto en varias partes relativamente cortas con un paso fijo. Se utilizó el conjunto de datos público del Consorcio ADHD-200 para entrenar y validar nuestro método. Y los resultados de las evaluaciones mostraron que nuestro método superó a los métodos tradicionales en el conjunto de datos (precisión: 71.3%, AUC: 0.80). Por lo tanto,

nuestro método CNN 4-D puede utilizarse para construir una herramienta de diagnóstico asistido automático más preciso para el TDAH[10].

Aspectos útiles al presente estudio	Este proyecto aporta la forma que fue aplicada la metodología sobre los datos extraídos de la colección ADHD-200 mediante LSTM que es una red neuronal profunda.
--	--

Fuente: sciencedirect.

TABLA IV ANTECEDENTE (Zhenyu Mao et al,2019)

Método de aprendizaje profundo espacio temporal para la clasificación de TDAH fMRI

Autor (as, es)	Zhenyu Mao, Yi Su, Guangquan Xu, Xueping Wang, Yu Huang, Weihua Yue
Fecha	octubre del 2019
País	Estados unidos
Resumen	El trastorno de déficit de atención con hiperactividad (TDAH) es un trastorno del desarrollo neurológico que impacta a numerosos niños y que comúnmente persiste hasta la edad adulta. Su diagnóstico preciso y constante es crucial para asegurar la aplicación de tratamientos apropiados, ya que, en caso contrario, podría tener repercusiones negativas en el rendimiento laboral, la salud mental y las relaciones personales del individuo. En este estudio, se realizaron ajustes previos a los datos de actividad motora de adultos con TDAH y a un grupo de control clínico, dando como resultado 788 características estadísticas relacionadas con la actividad. Posteriormente, se llevó a cabo un análisis de componentes principales para identificar características de relevancia con miras a una clasificación precisa. Estas características se integraron en seis algoritmos de aprendizaje automático diversos, como C4.5, kNN, Random Forest, LogitBoost, SVM y Naive Bayes, para llevar a cabo la clasificación. La evaluación exhaustiva de los resultados a través de una validación cruzada de diez repeticiones reveló que SVM sobrepasa a los demás clasificadores, alcanzando una precisión del 98,43 %, una medida F del 98,42 %, una sensibilidad del 98,33 %, una especificidad del 98,56 % y un AUC de 0,983. En consecuencia, parece que un enfoque basado en SVM y PCA resulta efectivo para la identificación precisa de pacientes con TDAH en comparación con otros individuos bajo control clínico, empleando análisis en tiempo real de datos de actividad[11].
Aspectos útiles al presente estudio	El estudio emplea una variedad de algoritmos de aprendizaje automático, incluidos C4.5, kNN, Random Forest, LogitBoost, SVM y Naive Bayes, para llevar a cabo la clasificación. Esta diversidad permite evaluar qué algoritmo es más efectivo en el contexto del estudio.

Fuente: brain sciences.

TABLA V. ANTECEDENTE (Chaitra, Shreya, Rajendra, Hojjat, Muralidhar et al,2017)

Diagnóstico del trastorno por déficit de atención con hiperactividad mediante técnicas de procesamiento de imágenes y señales

Autor (as, es)	Chaitra Sridhar, Shreya Bhat, U. Rajendra Acharya, Hojjat Adeli, G. Muralidhar Bairy
Fecha	1 de septiembre 2017
País	India

Resumen	El trastorno por déficit de atención es el trastorno infantil más común, este puede continuar hasta la adolescencia o edad adulta. La hiperactividad, la falta de atención y la impulsividad son las características claves que se pueden observar en los niños que tengan este trastorno. En los últimos grados se han realizado investigaciones importantes para el diagnóstico de este trastorno de forma cuantitativa utilizando imágenes médicas y técnicas de procesamiento de señales. este articulo presenta técnicas de procesamiento de señales e imágenes médicas[12].
Aspectos útiles al presente estudio	Este artículo presenta una revisión de la investigación reciente sobre el diagnóstico de TDAH utilizando técnicas de procesamiento de señales e imágenes médicas

Fuente: ScienceDirect revistas y libros

TABLA VI ANTECEDENTE (Javier Costa Rosa et al,2017)

Sistema de ayuda a la diagnosis y prognosis de diferentes enfermedades aplicando técnicas de Machine Learning

Autor (as, es)	Javier Costa Rosa
Fecha	2017-2018
País	España
Resumen	Este trabajo se basa en la creación de una herramienta web de ayuda a personal médico proporcionándoles la posibilidad de realizar distintos tipos de predicciones sobre determinadas enfermedades, ayudando a la incrementación de la cantidad de información a la hora de tomar una decisión sobre un tratamiento, pronostico o diagnóstico. Para esto esta herramienta hace uso de técnicas de Machine Learning. Esta herramienta se plantea desde cero los cuales se deben desarrollar tanto la parte del Frontend como el Backend, para esto se hace uso del modelo MEAN Stack[13].
Aspectos útiles al presente estudio	<ul style="list-style-type: none"> • Desarrollo de la herramienta web • Diseño del sistema • Utilización de la metodología ML

Fuente: Universidad Politécnica de València

TABLA VII ANTECEDENTE (PATRICIA AMADO CABALLERO et al,2019)

Ayuda al diagnóstico del TDAH en la infancia mediante técnicas de procesado de señal y aprendizaje

Autor (as, es)	PATRICIA AMADO CABALLERO
Fecha	3 de abril de 2019
País	España

Resumen	El trastorno por déficit de atención e hiperactividad es un trastorno crónico de carácter neuroconductual la cual se enmarca principalmente en la infancia, siendo uno de los trastornos psiquiátricos infantiles más frecuentes. Su diagnóstico actual la cual se ve marcada por las pruebas nos conduce a la necesidad de establecer un método de diagnóstico objetivo. En este trabajo se plantea un método de diagnóstico rápido basado en un sistema experto donde se aúnan la utilización de registros de astigrafía con técnicas de aprendizaje profundo (Deep Learning), Para la creación de este sistema experto, se ha realizado primero un procesado sobre los registros astigráficos, dividiéndolos en ventanas de actividad de diferentes duraciones y transformando estas nuevas señales en imágenes gracias al espectrograma, sirviendo así como datos de entrada para una red neuronal convolucional, núcleo central del sistema de diagnóstico. Una vez realizada esta primera clasificación de ventanas, se ha llevado a cabo una decisión de diagnóstico para cada paciente en función de la distribución obtenida para dichas ventanas[14].
Aspectos útiles al presente estudio	<ul style="list-style-type: none"> • Sistema experto • Reconocimiento de patrones • Deep Learning

Fuente: Escuela técnica superior de ingenieros de Telecomunicaciones, Universidad Valldolid

F. Fundamentos teóricos

A continuación, se muestran los fundamentos epistemológicos teóricos e investigaciones que existen, los cuales dan soporte científico a esta presente investigación.

1) Imágenes funcionales FMRI

La resonancia magnética funcional (FMRI) “es un procedimiento clínico y de investigación que permite obtener imágenes de áreas del cerebro que están activas”[15], por ejemplo, mientras se realizó una determinada tarea como mover una extremidad, tomar un objeto etc.

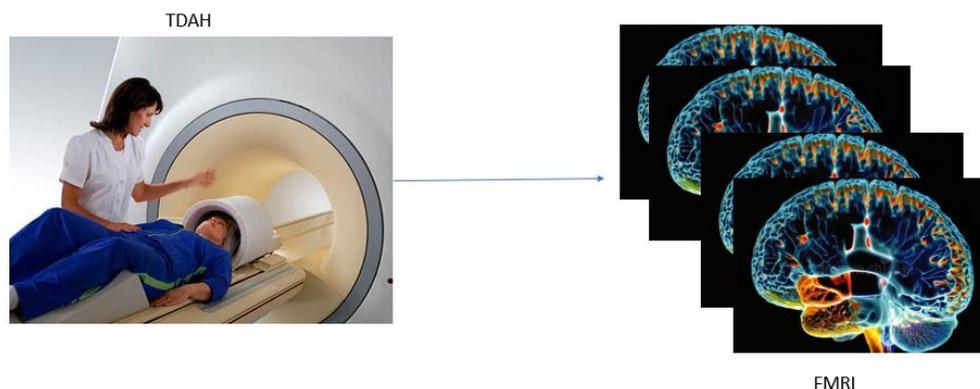


Fig. 1. Resonancias magnéticas extraídas desde repositorio proporcionado por NITRC.

Fuente: NITRC enlace

1) Machine learning

El aprendizaje automático, o machine learning en inglés, “es una rama de la inteligencia artificial que se centra en el desarrollo de algoritmos y modelos”[16], esta permite a las computadoras aprender patrones y realizar tareas sin ser programadas explícitamente. En lugar de seguir instrucciones específicas, las máquinas utilizan datos para mejorar su rendimiento en una tarea particular a medida que ganan experiencia. Esencialmente, el machine learning busca capacitar a las máquinas para que puedan hacer predicciones, tomar decisiones y automatizar procesos a partir de la información que reciben.

Los algoritmos más utilizados para la práctica aprendizaje profundo, en su mayoría son:

a) Aprendizaje supervisado

El algoritmo de aprendizaje supervisado se practica en un conjunto de datos de entrenamiento previamente etiquetado. El etiquetado significa que cada vez que aparece un conjunto de datos de entrenamiento, se conoce el valor de su atributo de destino. Esto permite que el algoritmo “aprenda” una función que puede predecir el atributo de destino de un nuevo registro. Las dos familias principales de algoritmos supervisados son:

- Algoritmo de regresión cuando el resultado esperado es un atributo numérico.
- Algoritmo de clasificación cuando el resultado esperado es un atributo categórico.

Ejemplos de este tipo de algoritmo son modelos de regresión lineal y logística, árboles de decisión, redes neuronales, bosque aleatorio.

b) Aprendizaje no supervisado

El método no supervisado es un algoritmo que ejecuta el proceso de entrenamiento basado en un conjunto de datos que no tiene etiquetas ni clases predefinidas. Es decir, los valores objetivos o de clase, ya sean categóricos o numéricos, no se conocen de antemano. “El aprendizaje no supervisado corresponde a tareas de agrupación”[17], también conocidas como clustering o segmentación. El propósito de esta tarea es encontrar grupos similares en el conjunto de datos. Hay dos grupos principales de métodos o algoritmos de agrupación.

- Método jerárquico. Cree una estructura jerárquica de las instancias que componen el conjunto de datos, lo que permite diferentes niveles de agrupación.
- Un método de partición o no jerárquico que crea un grupo de entidades que no se ajusta a ningún tipo de organización jerárquica. Ejemplos de este tipo de técnica son el agrupamiento jerárquico, PCA, los k-means.

2) *Deep Learning*

El aprendizaje profundo (Deep Learning) es un tipo de aprendizaje automático (machine learning) que utiliza algoritmos inspirados en las redes neuronales del cerebro humano para resolver problemas complejos. Las redes neuronales artificiales son sistemas que se componen de muchas capas de neuronas interconectadas, y cada neurona recibe información de las neuronas de las capas anteriores y la procesa para producir una salida.

El Deep Learning es un tipo de aprendizaje automático supervisado, lo que significa que se le proporciona a la red neuronal un conjunto de datos etiquetados, es decir, datos que ya han sido clasificados. La red neuronal utiliza estos datos para aprender a identificar patrones y clasificar nuevos datos de forma similar.

a) *Ann fully connect*

También conocido como red neuronal completamente conectada. “Estas son la capacidad que se tiene para conectar las redes de aprendizaje profundo las cuales serán utilizadas por una cantidad indescriptibles de aplicaciones”[18], una red neuronal totalmente conectada consta de una serie de capas totalmente conectadas. Una capa completamente conectada es una función de \mathbb{R} metro a \mathbb{R} norte. Cada dimensión de salida depende de cada dimensión de entrada. Al ser redes neurales completamente conectadas tienes una ventaja y es que pueden ser conectadas independientemente de una estructura. Obsérvese en Fig. 2

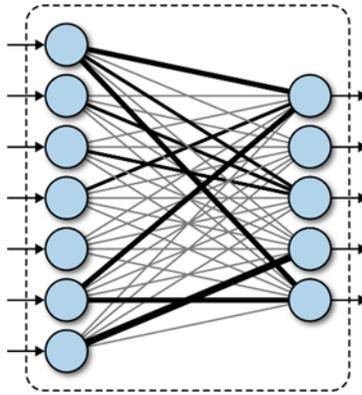


Fig. 2. Capa de red neuronal completamente conectada.

Fuente O'Reilly enlace <https://www.oreilly.com/library/view/tensorflow-for-deep/9781491980446/ch04.html>

b) Redes neuronales recurrentes (RNN)

Se consideran como redes neuronales artificiales “las cuales presentan bucles de información permitiendo tratar datos que no son independientes y que no tienen tamaño fijo, pero son limitadas debido a que tienen problema de memorias cortas”[19]. Estas ayudan hacer predicciones de lo que puede suceder en un futuro a partir de datos; permitiendo que la inteligencia artificial pueda darse el lujo de recordar u olvidar información (datos). Estas clases de redes son eficientes ya que permiten que se recuerde la salida anterior como una entrada. Obsérvese en la Fig. 3

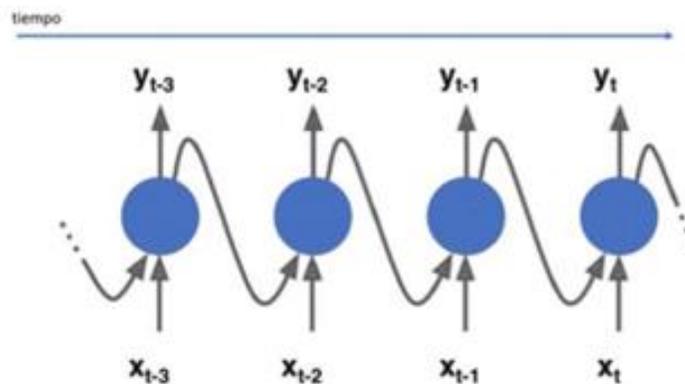


Fig. 3. Redes neuronales recurrentes.

Fuente: Trabajo de grado redes recurrentes profundas para el diagnóstico del tdah en la infancia. enlace
file:///C:/Users/57310/Downloads/tesis%20de%20redes%20profundas%20para%20el%20dianostico%20de%20tdah.pdf

Para solucionar el problema de memoria a corto plazo se crearon operaciones matemáticas con el fin de que funcionaran como puertas actuando como mecanismo para almacenar información importante permitiendo eliminar aquella información que no lo era. Los más utilizados comúnmente son:

c) Long-Short Term Memory (LSTM)

Traducido al español a Memoria a corto o largo plazo. “Estas son un tipo especial de redes las cuales su operación les permite controlar el flujo de información, dándoles a elegir si quieren que ciertos datos (información) sean recordados u olvidados”[20]. Dentro de estas memorias se añade nuevas informaciones las cuales vienen de secuencias anteriores. A continuación, se explica cómo actúa este tipo de memoria figura 4:

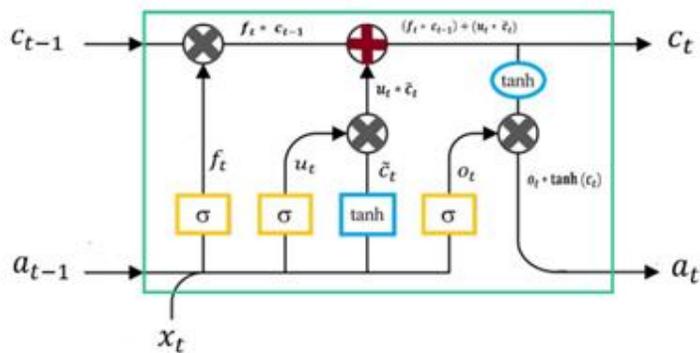


Fig. 4. Funcionamiento redes LSTM

Fuente: Abdatum enlace <https://abdatum.com/tecnologia/redes-neuronales-recurrentes>

Estas memorias funcionan ingresando celdas de estados las cuáles serán las encargadas de incluir o quitar información; para esto se usan las siguientes puertas:

- Forget gate: se encarga de decidir qué información permanece o se olvida; con ayuda de la función sigmoide la cual trabaja de 0 a 1; cero (0) significa que es menos importante y uno (1) más importante[8].
- Input gate: se encarga de actualizar el estado oculto. Para esto se agrega el estado oculto de un tiempo anterior. Permitiendo conservar la cantidad de información anterior; y se controla con la misma función sigmoide transformando los valores entre 0 y 1 lo que significa que entre más cerca al 0 este es menos importante y entre más cerca al 1 este es más importante y que esa información debe permanecer [8].
- Output gate: se encarga de decidir cuál será el estado oculto de la siguiente célula, haciendo uso de la función sigmoide y tanh. Tanh es una función la cual se encarga de comprimir los valores entre 1 y -1 para evitar que los valores incrementen o disminuyan mucho para así poder evitar problemas durante el entrenamiento[8].

d) Gated Recurrent Units (GRU)

traducido al español Unidades recurrentes cerradas. Estas son neuronas un poco diferentes a las LSTM, “estas solo utilizan el estado oculto para pasar información, y solo tiene dos gate; la de actualización y reinicio”[21]. A continuación, su explicación:

- Update gate: es similar a la función forget gate y de entrada de las LSTM, con el objetivo de que permanezca la información importante y eliminar los datos que no se consideran importantes[21].
- Reset gate: se encarga de controlar la cantidad de información que se olvida durante el aprendizaje de la red[21].

Se puede decir que estas GRU tienen menos información lo que hace que sean más rápidas que las LSTM a la hora de entrenar, pero ambas funcionan bien y dependiendo del tipo de problema que se tenga se utilizan para saber con cuál de las dos se obtiene un mejor resultado.

3) Temporal Convolutional Network

Traducido al español como Red Convolutiva Temporal; al igual que las redes neuronales están permiten detectar patrones en los datos de entrada, pero en este caso los datos de entrada son

imágenes. se ingresa una imagen la cual es procesada por una red convolucional y dentro de dicha red se encuentran algunos filtros (kernels), permitiendo extraer las características de dicha imagen y así poder realizar una clasificación en las categorías que se tengan. Esto lo hacen posible debido a que lo que hacen es imitar el cerebro humano y la forma de procesar las imágenes.

4) *El filtro (kernels)*

Básicamente consiste una matriz la cual está constituida por una serie de coeficientes numéricos y la idea de esta es que este realice una especie de barrido a través de la imagen usando una operación llamada convolution, dependiendo de los coeficientes que posea este filtro con la operación se es posible detectar algunas características en dicha imagen de entrada. La convolution se puede detectar en imágenes desde 1D hasta 3D o más dimensiones.

5) *Conjunto de datos*

Se le llama a una colección de datos organizados, que suelen abarcarse en dicho tema. Usualmente se presentan en tablas formadas por filas y columnas los cuales son representados por una variable en común donde cada fila o columna corresponden a un miembro del conjunto en cuestión.

6) *Metodología Knowledge Discovery in Databases (KDD)*

KDD representa un enfoque metodológico y secuencial empleado para descubrir información dentro de un conjunto de datos sin procesar. Este proceso se desglosa en nueve etapas:

- Selección de datos: Una vez se tenga el conjunto de datos reunido y los objetivos a lograr claramente establecidos, se selecciona los datos relevantes para llevar a cabo el estudio. Estos datos deben ser reunidos en un único conjunto, con el propósito de avanzar hacia los objetivos de nuestro análisis. En ocasiones, esta información puede estar disponible en una única fuente centralizada, o bien podría estar dispersa en diferentes fuentes.
- Limpieza y pre procesamiento: En esta etapa se encarga de limpiar los datos, realizándose antes de manipular y transformar la información; no necesariamente se debe hacer al iniciar

el proyecto, se puede hacer en el momento que se considere necesario con el fin de extraer patrones o reglas que se consideren poco útiles; considerándolos poco útiles ya sea porque están incompletos, tiene ruidos o son datos inconsistentes. con esto se obtiene un conjunto de datos pequeño y de mayor calidad permitiendo mejorar la eficiencia de la minería de datos.

- Transformación de los datos: Durante esta fase, se busca elevar la calidad de los datos mediante cambios que pueden incluir la reducción de la cantidad de características en el conjunto de datos (disminución de dimensiones) o llevar a cabo ajustes, como transformar valores numéricos en categorías definidas (discretización).
- Minería de Datos- selección de algoritmo: En este paso, se trata de seleccionar el enfoque adecuado de Minería de Datos, como la clasificación, regresión o agrupación, según los objetivos específicos de la investigación, ya sea para predecir resultados futuros y desconocidos (predicción) o para entender cómo se comportan los datos (descripción). La clasificación se utiliza para construir modelos que se puedan aplicar a situaciones venideras, mientras que la regresión y agrupación se emplean para analizar y observar el comportamiento de los datos.
- Elección y aplicación de algoritmo: Luego, se avanza a elegir la técnica o algoritmo, y en algunos casos, incluso más de uno, para descubrir patrones y adquirir conocimiento. El meta-aprendizaje se concentra en explicar por qué ciertos algoritmos funcionan mejor en ciertos problemas, y para cada técnica existen diversas formas de seleccionarlas. Cada algoritmo tiene su propia esencia y método de operación para obtener resultados, por lo que es aconsejable comprender las propiedades de las opciones potenciales y determinar cuál se adapta mejor a los datos.
- Evaluación: Una vez que se han aplicado los algoritmos al conjunto de datos, pasamos a analizar los patrones generados y el desempeño logrado para asegurarnos de que estén alineados con los objetivos establecidos en las etapas iniciales.
- Interpretación de los resultados: En esta última etapa se pasa a interpretar los resultados obtenidos en la evaluación logrando dar como resultado la selección del mejor algoritmo.

III. MATERIALES Y MÉTODOS

A. Materiales

Para la exploración de datos se utilizó los siguientes materiales; un conjunto de datos de acceso público el ADHD 200 consta de 40 imágenes de FMRI, cada imagen es un video en 3 dimensiones, por lo tanto, cada voxel está compuesto por 3 dimensiones espaciales y el tiempo, donde las dimensiones son (61, 73, 61, 176), donde cada voxel representan el volumen del tejido del paciente y se define como una unidad tridimensional de la imagen con tres ejes, siendo dos el área o píxeles y el tercero el grosor del corte y este contiene 176 marcas de tiempo, el cual fue obtenido mediante la plataforma NITRC (Colaboratorio de herramientas y recursos de neuroimagen), la composición de este conjunto de datos es de una agrupación de resonancias magnéticas funcionales (FMRI) en estado de reposo y datos anatómicos recopilados de múltiples centros de investigación.

El conjunto de datos contiene una estructura compuesta dos tipos de datos que son **Func** que contiene las neuroimágenes y **Confounds** que tiene las variables molestas que representan el ruido y movimiento del paciente como se muestra en la siguiente tabla 5.

TABLA VIII DICCIONARIO DEL CONJUNTO ADHD-200

Tipo	cantidad	datos
Func	40	rest_tshift_RPI_voreg_mni.nii.gz
confounds	40	regressors.csv

Fuente: Esta investigación.

Además, se utilizó una herramienta que se llama FmriPrep, que es una canalización de preprocesamiento de datos de resonancia magnética funcional (FMRI) que fue diseñado para ayudar y proporcionar una interfaz de última generación de muy fácil acceso para realizar el procesamiento básico como (corregistro, normalización, deformación, extracción de componentes de ruido, segmentación, desmontaje del cráneo, etc.), el funcionamiento de esta herramienta se puede observar fig 7.

Luego se procedió a importar las bibliotecas, para cada script con los que se obtuvieron los resultados, en la tabla 6 se muestra detalladamente la función de cada una.

Tabla IX. BIBLIOTECAS UTILIZADAS EN LA INVESTIGACIÓN

Bibliotecas	Descripción
Keras	Esta biblioteca fue utilizada para la creación de prototipos de redes neuronales, permitiéndose ser usado como un frameworks independiente
MatplotLib	Biblioteca Utilizada para la generación de gráficos a partir de datos contenidos en listas
Nilearn	Permite hacer el análisis de volúmenes cerebrales, es decir facilita utilizar técnicas avanzadas de aprendizaje profundo y facilitad la analizar datos adquiridos con resonancias magnéticas.
Numpy	Es una librería de Python especializada en el análisis de datos, especialmente para un gran volumen de datos.
Scikit-Learn o sklearn	Permite hacer el modelo de aprendizaje automatizo como agrupación, regresión, clasificación entre otros, además sirve para el análisis de datos.
Warnings	Biblioteca para controlar la salida de manera individual, es decir caso por caso y reducir tener que hacerlo varias veces

Fuente: Esta investigación

B. Métodos

Para el desarrollo de la presente investigación se utilizó la metodología Knowledge Discovery in Databases (KDD), traducida al español Descubrimiento de conocimiento en bases de datos.

El objetivo principal es proveer un conjunto de herramientas para automatizar el proceso de análisis de datos y el artesanal proceso estadístico de selección de hipótesis. El objetivo subyacente al concepto de KDD, es el de diferenciar a la minería de datos, entendiendo a esta como la actividad de aplicar distintos algoritmos en los datos para obtener patrones, del proceso necesario para generar conocimiento a partir de los datos. Es decir, entender a la minería de datos como un subproceso que integra a un proceso general destinado a obtener patrones de conocimiento. Mediante la incorporación de KDD, se agregan una serie de pasos destinados a favorecer y garantizar el éxito de los resultados obtenidos en la aplicación de minería de datos. No basta con obtener patrones a partir de la ejecución de algoritmos de Minería de Datos, sino que se debe preparar los datos, y es definida por un conjunto de pasos interactivos e iterativos en los cuales el usuario debe tomar una serie de decisiones, estos pasos son los siguientes:

- Selección de datos
- Preprocesamiento/limpieza
- Reducción/transformación
- Minería de datos- selección de algoritmo
- Elección y aplicación de algoritmo
- Evaluación
- Interpretación

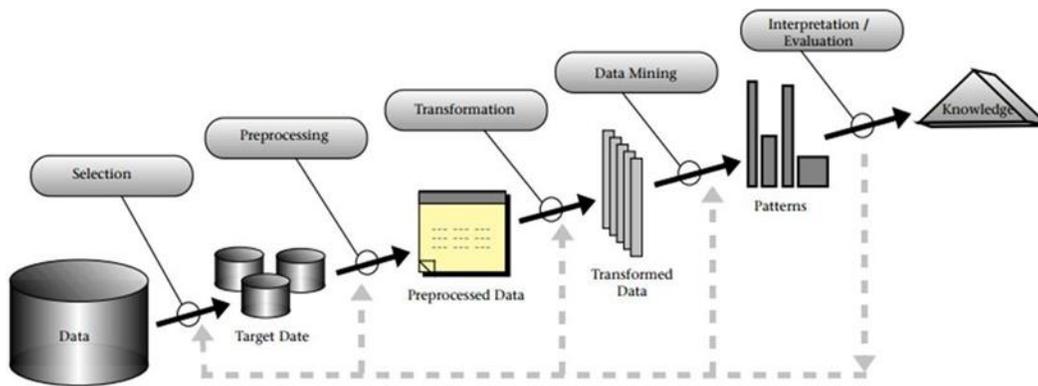


Fig. 5. Proceso metodología KDD

Fuente: <https://saludelectronica.com/descubrimiento-de-conocimiento-en-bases-de-datos-kdd/>

A continuación, se muestra de forma detallada las actividades que se realizaron para desarrollar cada una de las fases de esta metodología.

C. Selección de datos

En esta etapa se seleccionó un conjunto de datos basados en conectividad funcional en el trastorno de déficit de atención con hiperactividad (ADHD-200), el cual se extrajo de la plataforma NITRC, esta es de acceso público. Este conjunto utilizó datos como: fenotípica la cual es el estado de diagnóstico, edad, sexo, destreza, coeficiente intelectual de rendimiento y verbal, coeficiente intelectual de rendimiento, entre otras.

Para la obtención de esta base de datos se utilizaron resonancias magnéticas funcionales donde se pueda demostrar que las regiones del cerebro se conectan entre sí formando redes que permiten

saber cómo se comunica este. Además de los datos anteriormente dichos hay algunas variables molestas con el fin de que estas estén en un formato el cual ayuda a organizar la información de neuroimagen y su comportamiento. Una vez se obtuvo la información se pasó hacer revisión de los datos para pasar hacer el siguiente paso.

D. Preprocesamiento de datos adhd-200

Para esto se utilizó una herramienta llamada FmriPrep, la cual sirvió para hacer preprocesamiento de datos de resonancia magnética funcional, diseñada con el fin de ayudar a realizar un procesamiento básico y dar resultados que se puedan fácilmente interpretar. Esta herramienta permitió hacer una normalización, extracción de componentes de ruido, desmonte de cráneo, es decir, generar preprocesamiento de calidad los cuales puedan ser identificados fácilmente por los usuarios, lograr la calidad óptima de procesamientos de los datos utilizando las mejores herramientas, automatizar y generar una rapidez significativa a comparación de hacerlo manual.

E. Transformación de datos adhd-200

En esta parte esta misma herramienta (FmriPrep) se encargó de normalizar y transformar las neuroimágenes(datos), una vez hizo la corrección de movimiento, la corrección de distorsión y se encargó de limpiar todo lo que se considerase como variables confusas, luego se aplicó el filtro Smith y con ello se obtuvo como resultado una señal, representada como una serie temporal matricial que se pudo utilizar para hacer el análisis. Cada matriz de la serie obtenida representa un conectoma obteniendo un mapa detallado y completo de las conexiones neuronales en el cerebro a estudiar donde cada pixel (componente de la matriz) representa una unidad básica de información, que generalmente se refiere a una conexión o una fibra nerviosa entre dos puntos en el cerebro.

F. Minería de datos – selección del algoritmo

En esta fase de minería de datos se eligió un algoritmo de clasificación.

G. Elección y aplicación del algoritmo

Una vez ya seleccionado los algoritmos de clasificación, se aplicaron los siguientes: técnicas de aprendizaje supervisadas como máquina de soporte vectorial, arboles de decisión, bosques aleatorios, regresión logística, y técnicas de aprendizajes no supervisadas como estándar scaler y PCA.

H. Evaluación

Para evaluar estos algoritmos se realizó una tabla de comparación donde muestre las fases o algoritmo con los datos de hiperparametros y luego se eligió el modelo con mejor exactitud (accuracy).

I. Interpretación

En esta última etapa se dio paso a interpretar los resultados obtenidos en la evaluación logrando dar como resultado la selección del mejor algoritmo.

El desarrollo de la presente investigación se realizó en base a la siguiente tabla.

TABLA X PLAN DE ACCION

Fase	Actividad	Producto	Recursos
Preprocesar los datos del conjunto de datos ADH-200	<ul style="list-style-type: none"> • Seleccionar las herramientas para el almacenamiento y exploración de datos. • Adquirir los datos sobre diagnóstico de niños con TDAH. • Aplicar técnicas de extracción, transformación, limpieza, correlación y reducción de dimensión en los datos. 	<ul style="list-style-type: none"> • Datos depurados y obtención de las variables más relevantes, recopilados en un sistema gestor de base de datos. • Conjuntos de entrenamiento. 	<ul style="list-style-type: none"> • Internet • Computador • Instalar datacleaning. • Visitar página web de manual diagnóstico y estadístico de trastornos mentales. • Lenguaje de programación Python. • Motor de base de datos (PostgreSQL). • Software para la gestión de hojas cálculo.
Estructurar a partir de un conjunto de técnicas de aprendizaje automático y métricas de calidad una ruta base de configuración objetiva.	<ul style="list-style-type: none"> • Estudiar materiales y métodos más relevantes relacionados con el objeto de estudio. • Obtener un modelo de comparación de algoritmos de aprendizaje automático. 	<ul style="list-style-type: none"> • Diagramas de modelos de algoritmo de aprendizaje automático. 	<ul style="list-style-type: none"> • Artículos, pdf. • Internet
Fase	Actividad	Producto	Recursos
Seleccionar la técnica de aprendizaje supervisado que tenga el mejor compromiso en su exactitud y coste computacional e interpretabilidad de resultados.	<ul style="list-style-type: none"> • Probar algoritmo de aprendizaje con el conjunto en bases de datos conocidas. • Codificación de algoritmo de aprendizaje automático. 	<ul style="list-style-type: none"> • Scripts para la obtención de un modelo. • Configuración de modelos de algoritmos de aprendizaje automático. • Resultados del compromiso entre la eficiencia computacional y gráficos de interpretabilidad de bases de datos conocidas. 	<ul style="list-style-type: none"> • Lenguaje de programación (Python). • Motor de base de datos (PostgreSQL). • Computador

Implementar una aplicación web	<ul style="list-style-type: none"> • Codificar una interfaz web previamente validada. • Obtener matrices de métricas de calidad, gráficos de interpretabilidad y mapas de interpolación. • Codificar un marco experimental para la • Publicar el aplicativo con la evaluación en sus métricas de calidad e interpretabilidad. 	<ul style="list-style-type: none"> • Resultados de la eficiencia computacional e interpretabilidad de sobre el diagnóstico del TDAH. • Modelo para predicción para el diagnóstico del trastorno de déficit de atención e hiperactividad • Artículo científico. 	<ul style="list-style-type: none"> • Internet • Computador • Artículos, pdf. • Lenguaje de programación Python. • Servidor web. • Editor de datos.
--------------------------------	---	---	--

Fuente: Esta investigación

IV. RESULTADOS DE LA INVESTIGACIÓN

A. Preprocesar los datos del conjunto ADHD 200

A continuación, se presenta el desarrollo de cada uno de los objetivos de la presente investigación de acuerdo con las fases planteadas en el plan de acción propuesto en la tabla 4. En ese orden de ideas se muestra cómo se realizó la preparación de los datos del conjunto ADHD -200, no obstante, para la realización de este proceso se debe preprocesar las imágenes de forma individual.

Para el preprocesamiento de datos basados en conectividad funcional en el trastorno por déficit de atención con hiperactividad (TDAH), se utilizaron resonancias magnéticas funcionales que muestran cómo las regiones del cerebro se conectan entre sí y forman redes funcionales; permitiendo obtener información sobre cómo se comunica el cerebro. En primera instancia se realizó una selección de herramientas para la exploración de neuroimágenes.

Como se mencionó anteriormente, es posible que la imagen no contenga una longitud escaneo homogénea, esta presenta una variación muy grande. Sin embargo, la mayoría de los algoritmos de aprendizaje automático requieren una forma uniforme, para mejorar los datos y hacer preprocesamiento y proporcionar resultados que se pueden fácilmente interpretar y utilizar una combinación de herramientas de paquetes de software conocidos que incluyen FSL, ANT, FreeSurfer y AFNI. Esta tubería se diseñó para proporcionar la mejor implementación de software para cada estado de preprocesamiento y se actualizó a medida que haya disponible software de neuroimagen más nuevo y mejor como se observa en fig. 7

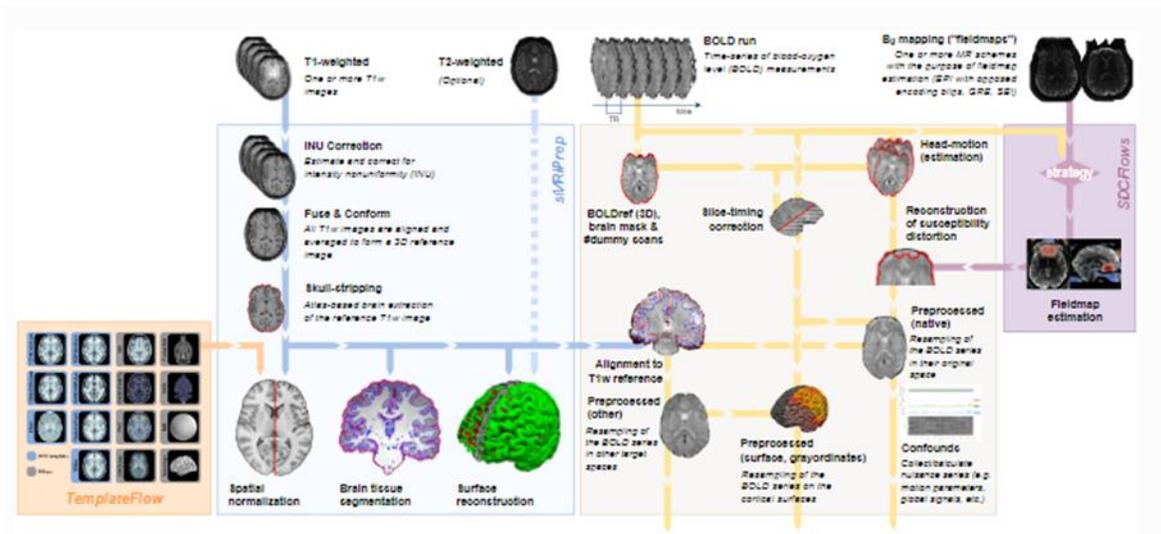


Fig. 6. Gráfico de preprocesamiento con la herramienta FmriPrep

Fuente: FmriPrep enlace <https://fmriprep.org/en/stable/index.html>

Esta herramienta permitió hacer lo siguiente:

- Tomar los datos de fMRI de la forma sin procesar a la forma completamente preprocesada.
- Implementar herramientas de diferentes paquetes de software.
- Lograr una calidad óptima de procesamiento de datos utilizando las mejores herramientas disponibles.
- Generar informes de calidad de preprocesamiento, con los que el usuario pueda identificar fácilmente los valores atípicos.
- Recibir resultados detallados sobre la etapa de preprocesamiento de cada tema, incluidos los errores significativos.
- Automatizar los pasos de procesamiento, lo que proporciona una aceleración significativa con respecto al procesamiento manual lineal típico.

FmriPrep recopila todos los resultados de preprocesamiento en un archivo HTML, a continuación, se dará detalles de lo que se utilizó para la normalización de estas imágenes.

En la primera sección se hizo un control de calidad anatómica, en la figura 8 se puede observar la imagen anatómica en vista sagital, axial y coronal. Mostrando la máscara cerebral delineada en

rojo y en fucsia se puede observar el límite de materia gris, por último, en azul el límite de materia blanca.

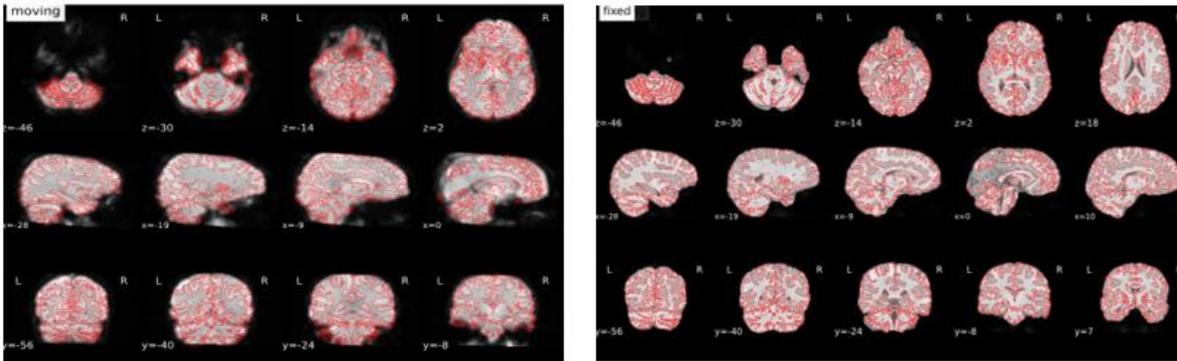


Fig. 7. Mascara cerebral y segmentación del tejido cerebral.

Fuente: Esta investigación, plantilla generada del preprocesamiento de la neuroimagen con la herramienta FmriPrep

Para la realización de normalización de la imagen anatómica a la plantilla MNI152NLin2009cAsym se verifico que en la alineación no se encuentren solo los contornos del cerebro, sino también las estructuras internas como los ventrículos. Véase en la fig. 9.

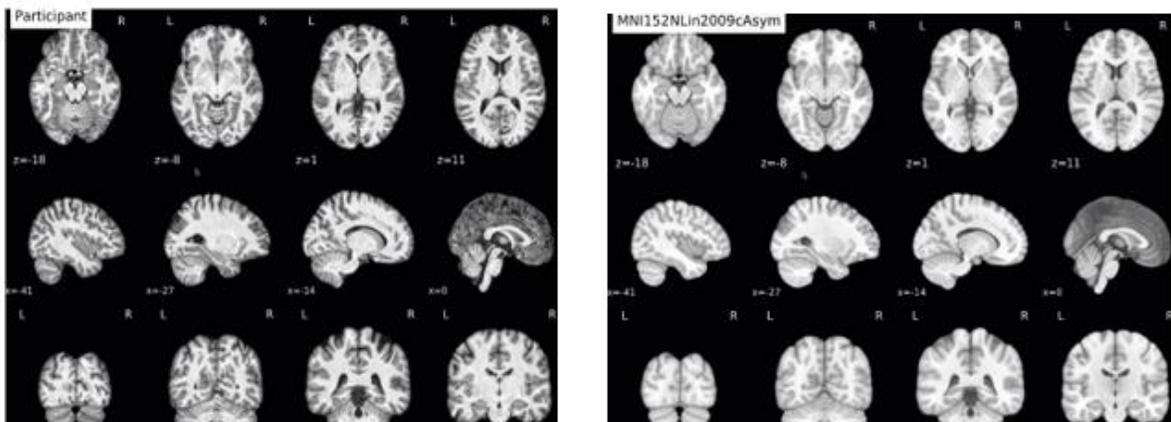


Fig. 8. Normalización espacial de la referencia anatómica

Fuente: Esta investigación, plantilla generada del preprocesamiento de la neuroimagen con la herramienta FmriPrep

En la segunda sección se hizo un control de calidad funcional, en la figura 10 se muestra las diferencias entre las imágenes anatómicas y funcionales. Al igual que en la sección anterior se verifico a través de un control de calidad de normalización, asegurando que se encuentren bien alineadas las estructuras internas y teniendo en cuenta que los voxeles más claros en las imágenes funcionales representan fluidos.

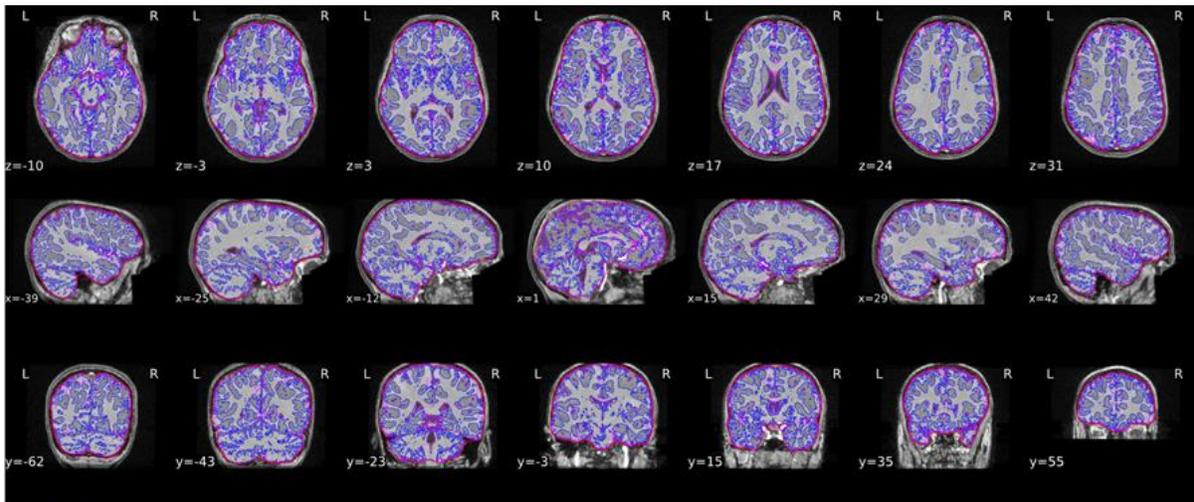


Fig. 9. Alineación de datos de resonancia magnética funcionales y anatómicos (impulsados por superficie)

Fuente: Esta investigación, plantilla generada del preprocesamiento de la neuroimagen con la herramienta FmriPrep

A continuación, la figura 11 muestra como la Máscara cerebral calculada en la señal BOLD (contorno rojo), junto con las regiones de interés (ROI) utilizadas en a/tCompCor para extraer componentes de confusión fisiológicos y de movimiento, además el ROI anatómico de CompCor (contorno magenta) es una máscara que combina LCR y WM (materia blanca), donde se han eliminado los vóxeles que contienen un volumen parcial mínimo y el ROI temporal de CompCor (contorno azul) contiene el 2% superior de los vóxeles más variables dentro de la máscara cerebral.

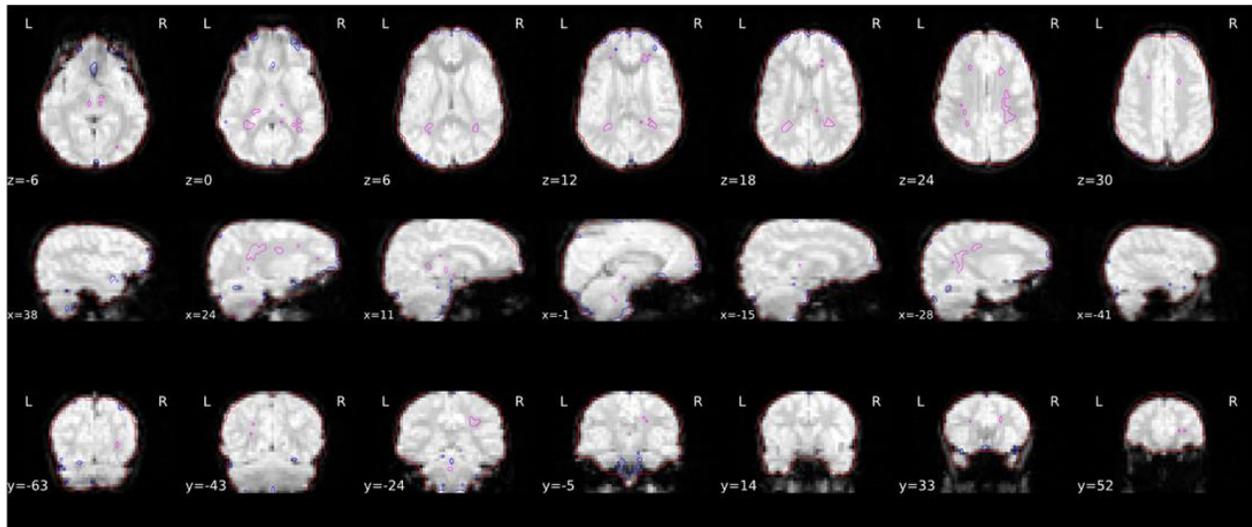


Fig. 10. Mascara cerebral y ROI de CompCor (Anatómico/temporal)

Fuente: Esta investigación. Plantilla generada del preprocesamiento de la neuroimagen con la herramienta FmriPrep

Como resultado del preprocesamiento de esta imagen se obtuvo que es una imagen anatómica normalizada para este modelo, mientras que los archivos sin esta secuencia están en el espacio original (es decir, no normalizado). en el archivo se obtuvo una imagen anatómica, limpia, procesada, normalizada y comprimida, que tiene como nombre sub-218_space-MNI152NLin2009cAsym_desc-preproc.ni.gz, la cual tiene como característica que es una imagen en 3 dimensiones a la cual se le realizó una extracción de cráneo. Muchos de estos archivos son imágenes intermedias que se utilizan para mejorar el proceso de normalización, así como para extraer el curso temporal de los tipos de tejido como regresores de confusión como se observa figura 12.

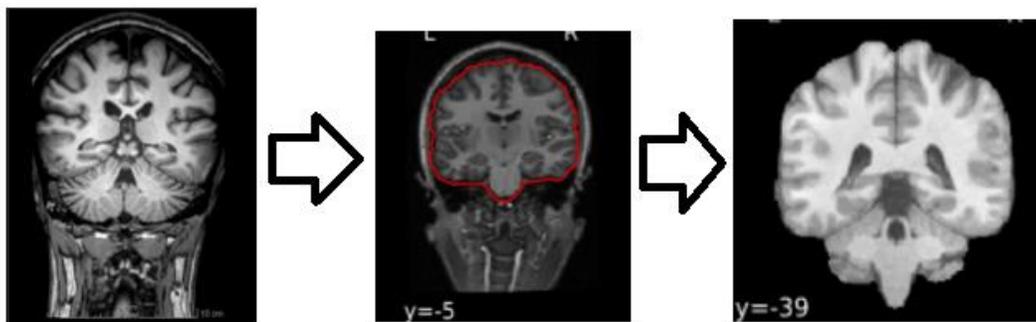


Fig. 11. Archivos resultantes del preprocesamiento en el directorio anat.

Fuente: Esta investigación

También se generó una hoja de cálculo llamada **confounds_timeseries.tsv** que contiene una lista de regresores confusos, como los cursos de tiempo de la sustancia blanca y el líquido cefalorraquídeo, y los parámetros de movimiento y sus derivados en las direcciones X, Y y Z, como se observa figura 13.

```

csfconstantlinearTrendwmglobalmotion-pitchmotion-rollmotion-yawmotion-xmotion-ymotion-zgmcompcor1compcor2compcor3compcor4compcor5
12481.7769203186041.00.09910.0183770440139935.6346597492450.08780.1387-0.117-0.03290.06790.0189916.73120867998-0.032152643133871120.03346199807
12484.975349426271.01.09909.9250292047169901.9300232086170.08910.1217-0.1587-0.02810.071-0.03239872.041485146341-0.072990540047069950.068871278
12505.8145523071291.02.09904.2199998601749879.9637011767570.08780.1084-0.1357-0.05960.0802-0.00359853.947979153843-0.103069025794583160.0442706
12470.2712097167971.03.09893.5553198981059874.4637781632050.09050.1285-0.1823-0.02140.06490.00619862.650327164058-0.0582182364627920060.160676C
12487.423870086671.04.09900.0107320182859889.7950203400450.11910.1182-0.1548-0.0160.0599-0.01019889.696330127812-0.079831327225074970.089410734
12527.5858993530271.05.09915.9357369122199911.468356615330.08670.1207-0.1227-0.04050.0470.01279900.508744330404-0.082373104696952470.0103210481
12576.3151245117191.06.09915.7200011313259922.3289052986350.10840.1151-0.1392-0.0290.05970.00649905.422571495088-0.12635114191318375-0.03792853
12550.7194709777831.07.09907.5576225898979913.0255645872310.12090.0918-0.1342-0.02710.0605-0.01759897.16953049209-0.12167285518694340.041633906
12487.1050109863281.08.09903.2242310483969916.1870273945660.08930.0857-0.142-0.03780.05150.02049894.690636411846-0.061011248265736010.09572132C
12502.8268661499021.09.09923.577391386429945.2985838877470.04420.1413-0.071-0.02590.06850.0359941.096544872933-0.011076145292907211-0.046196999
12426.0820884704591.010.09937.6334039034549954.3637531213970.02370.1518-0.0479-0.02190.08680.01689952.2013423294010.18716112708720214-0.0759357
12533.5356292724611.011.09937.6228479397189967.996046712380.05920.1266-0.0836-0.0160.0925-0.01659956.5765936187-0.0022204106560044217-0.1469550
12553.2105674743651.012.09945.3170059159429964.5028300569830.04090.0909-0.1218-0.03830.0964-0.02329946.716933766276-0.06538907190412567-0.18110
12527.6066360473631.013.09940.121484629239926.2774557387640.04830.0702-0.1554-0.04120.0868-0.00489900.298734699676-0.08457903402459944-0.103647
12455.4392318725591.014.09904.0896738923189913.0872324187150.03470.0675-0.1354-0.03870.08130.00229896.996934847113-0.0256374304681605580.123755
12494.9982910156251.015.09905.0910721594859918.7030758882630.03370.0559-0.1221-0.04340.08480.00869893.868354983735-0.079367075496925750.1146328

```

Fig. 12. Archivos resultantes del preprocesamiento en el directorio func

Fuente: Esta investigación.

Una vez terminado el preprocesamiento de las imágenes se crea una semilla para una región específica del cerebro que se seleccionó como punto de inicio[22] o referencia para realizar un análisis de conectividad funcional. La idea detrás de esta técnica es estudiar cómo se correlacionan las señales de esta región con otras regiones del cerebro, lo que puede proporcionar información sobre redes funcionales y conexiones entre áreas cerebrales.

La semilla se crea como una máscara esférica centrada en coordenadas específicas (como **pcc_coords = (0, -53, 26)** en el código). Esta máscara es esencialmente una región de interés (ROI) que se define por una esfera con un radio determinado alrededor de las coordenadas especificadas, véase figura 13.



Fig. 13. Contraste cálculo de semilla

Fuente: Esta investigación

Después de crear la máscara de la semilla, se aplicó una máscara llamada Smith atlas la cual se encargó de parcelar las imágenes en un conjunto de 176 regiones más pequeñas y discretas con el objetivo de calcular la serie de tiempo promedio de cada región específica del cerebro que se seleccionó como punto de inicio o referencia para realizar un análisis de conectividad funcional promediando la serie de tiempo de todos los vóxeles para crear una matriz de correlación simétrica de (176,70) para cada sujeto como se observa figura 14.

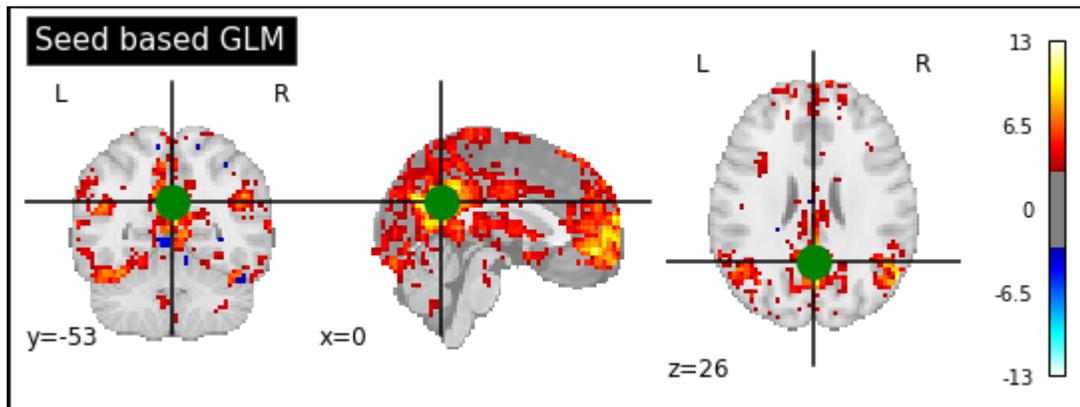


Fig. 14. Mascara Smith atlas

Fuente: Esta investigación.

Las series temporales pueden tener longitudes diferentes debido a la duración variable de las imágenes de fMRI, entonces se debió realizar un proceso para asegurarse de que todas las series temporales tuvieran la misma longitud en este caso 261. Esto se hizo rellenando con ceros las series más cortas para que todas coincidieran con la longitud de la serie más larga.

Una vez se obtuvo la limpieza se hizo la conversión a una estructura tridimensional que es adecuada para la entrada de los modelos de machine learning, donde la primera dimensión es la cantidad de sujetos, la segunda es la longitud de las series temporales y la tercera corresponde al número de regiones de interés. Como se ve en la siguiente figura 15.

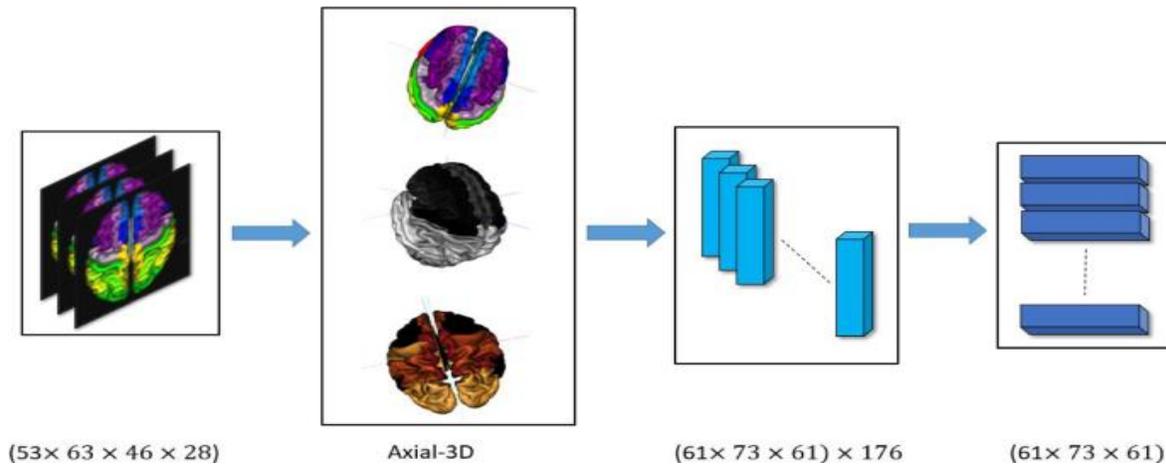


Fig. 15. Matriz resultante para evaluación de los modelos

Fuente: Esta investigación.

Por último, se utilizaron matrices binarias para representar las etiquetas de clase donde (0 y 1, correspondientes a control y ADHD), se utilizaron para categorizar los sujetos en grupos en el contexto de estudios de neuroimagen y análisis de datos relacionados con el trastorno por déficit de atención e hiperactividad. A continuación, su explicación:

1. **Control:** Los sujetos etiquetados como "control" pertenecen al grupo de individuos que no tienen el trastorno que se está estudiando. En este caso, se refiere a personas que no tienen el trastorno por déficit de atención e hiperactividad (ADHD). Estos individuos se consideran el grupo de referencia y se utilizan para comparar con el grupo de sujetos que tienen ADHD.
2. **ADHD:** Los sujetos etiquetados como "ADHD" son aquellos que tienen el trastorno por déficit de atención e hiperactividad. El ADHD es un trastorno neuropsiquiátrico caracterizado por síntomas como dificultad para mantener la atención, hiperactividad y comportamiento impulsivo. Estos individuos se estudian con el objetivo de entender las diferencias en la actividad cerebral, la conectividad funcional y otros marcadores de neuroimagen en comparación con el grupo de control.

B. Estructurar a partir de un conjunto de técnicas de aprendizaje automático y métricas de calidad una ruta base de configuración objetiva.

A continuación, en la figura 16 se observa el proceso a desarrollar según el marco experimental de esta investigación. Con la herramienta fmriprep se aplicó el filtro Smith para obtener conectoma de las imágenes electromagnéticas funcionales, después de ello se hicieron pruebas con algoritmos de *Machine learning* y *Deep Learning*, se seleccionó el algoritmo con base a la relación de exactitud y f1 score.

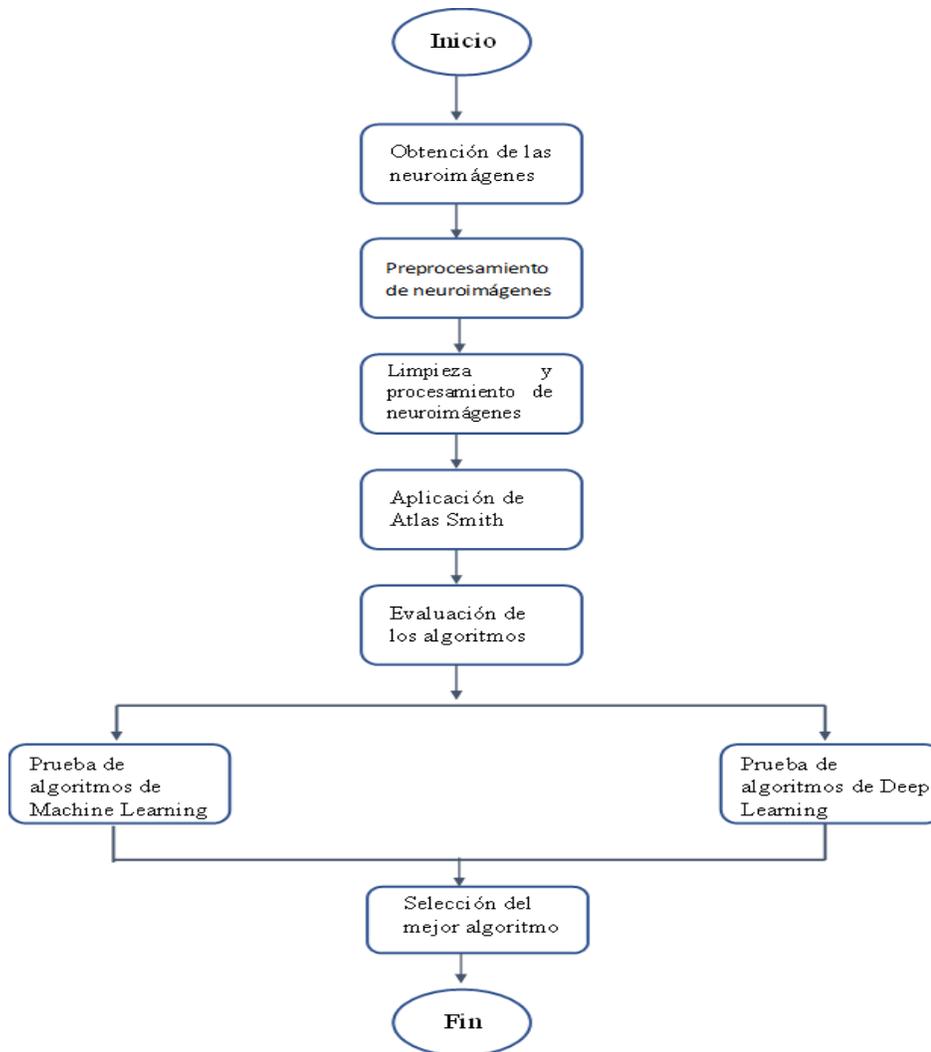


Fig. 16. Diagrama de marco experimental

Fuente: Esta investigación

Los algoritmos seleccionados para aplicar en la presente investigación fueron: arboles de decisión, bosques aleatorios, maquinas vectoriales, regresión logística, GRU y LSTM ya que frente a otros algoritmos, según [23] y [24], son adecuados para realizar esta evaluación, cabe resaltar que de cada uno de estos algoritmo se aprovechara su grado de predictibilidad mas no su grado de interpretabilidad de los parámetros.

La ejecución de los experimentos con los algoritmos de *Machine Learning* se llevó acabo de la siguiente manera. Obsérvese en la figura 17. En ella se hizo inicialmente previa selección de los algoritmos anteriormente enunciados, se realizó un aplanamiento de los datos, también se separaron los datos entrenamiento y prueba, también mediante búsqueda aleatoria se evaluaron los algoritmos utilizando un escalador estándar y el algoritmo de reducción de dimensión PCA, otra evaluación sin escalador estándar y reducción de dimensión, por último, se hizo una evaluación solo con escalador estándar.

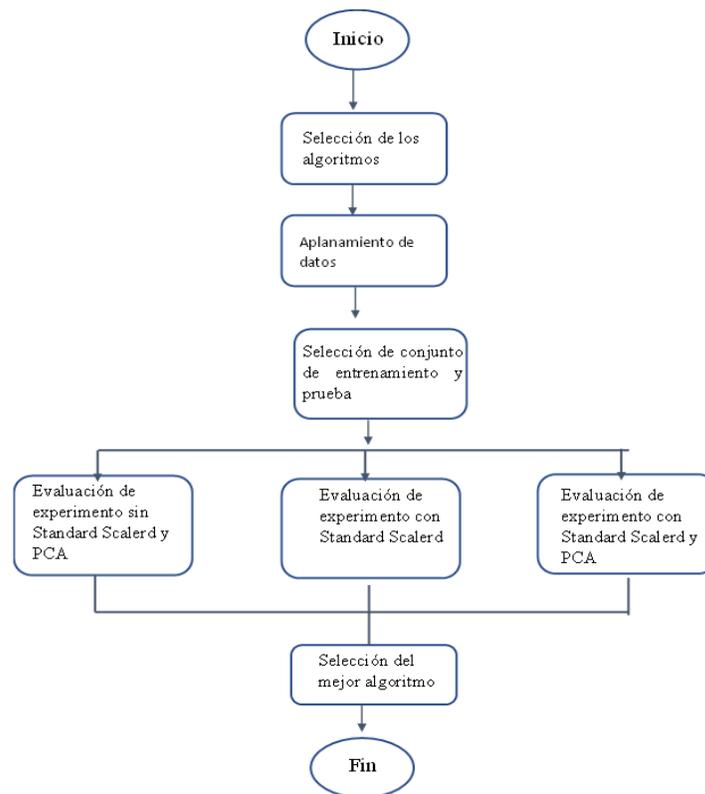


Fig. 17. Diagrama de ejecución de algoritmos de machine learning.

Fuente: Esta investigación

Para los algoritmos de *Deep Learning* (Lstm y Gru), se llevó a cabo de la siguiente configuración. Obsérvese en la figura 18. En ella se observa que al igual que la anterior configuración se seleccionaron y se aplanaron se partitionaron en entrenamiento y prueba, se evaluó el experimento sin utilizar el escalador estándar ni reducción de dimensión, se realizó otro experimento haciendo una transformación del conectomas a vector para realizar un escalador estándar para obtener la exactitud y el f1 score.

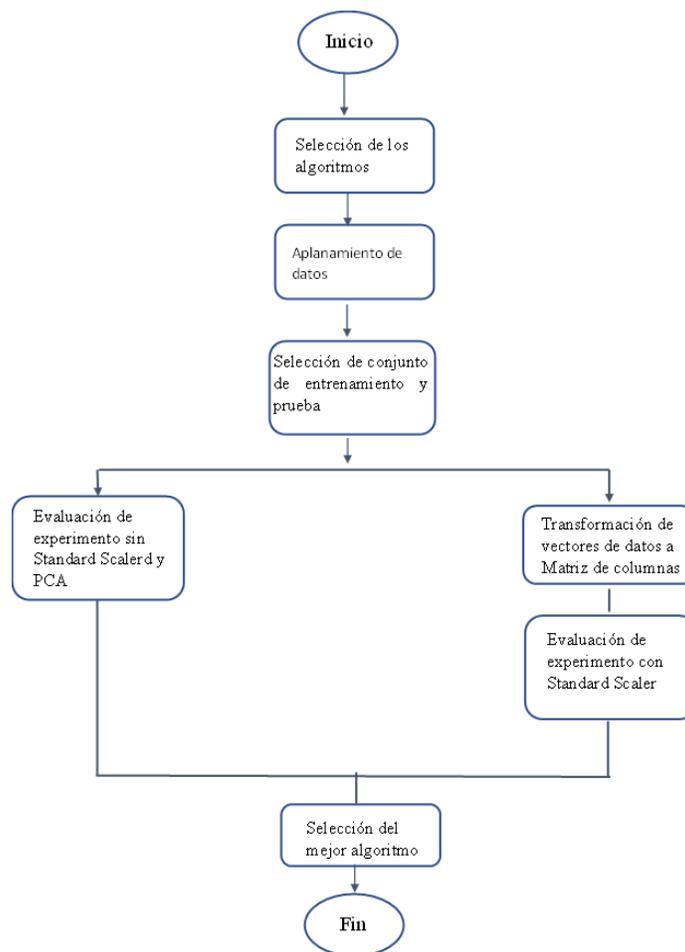


Fig. 18. Diagrama de ejecución de algoritmos Deep learning.

Fuente: Esta investigación

Una vez procesada la neuro imagen el siguiente paso fue cargar en una variable que se llama “Xp” para que esté contenga el resultado de la ejecución de la función previamente instanciada como lo muestra la figura 19.

Luego de la ejecución del anterior proceso se cargó el modelo previamente evaluado en una variable la cual se llamó “model_load”, por último, se ejecutó la variable “model_load” con el método predict el cual se encargó de hacer una predicción con los datos previamente cargados. Una vez terminado la ejecución de la predicción este arrojo un arreglo con dos valores, el primer valor es la cantidad de TDAH tiene un paciente.

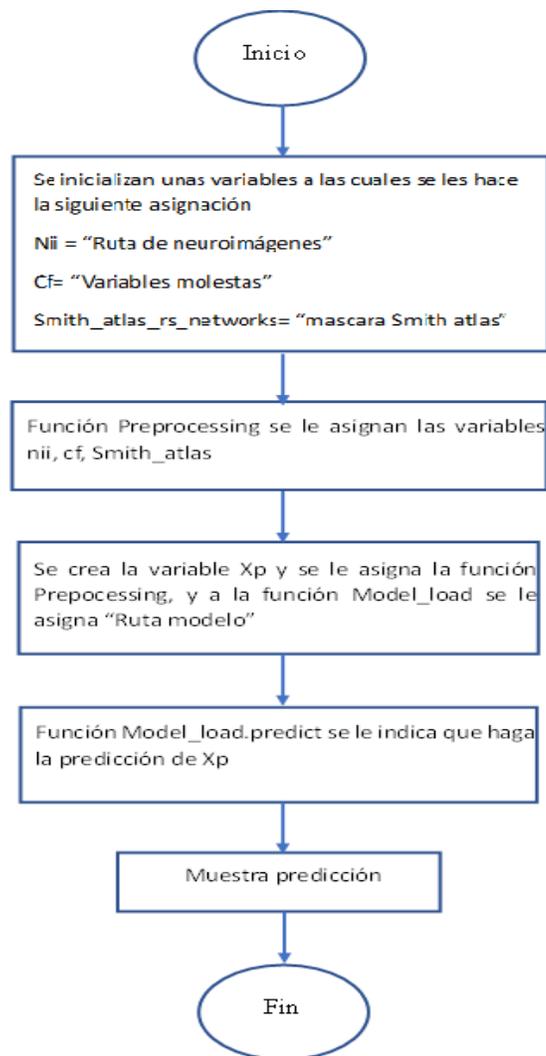


Fig. 19. Diagrama de pasos para predicción

Fuente: Esta investigación

C. Seleccionar la técnica de aprendizaje supervisado que tenga el mejor compromiso en su exactitud y coste computacional e interpretabilidad de resultados.

A partir de la configuración propuesta en la sección anterior (Sección B) se obtuvo el siguiente desempeño por cada combinación de algoritmos, escaladores y reductores de dimensión. Para ello se utilizó un pipeline por cada combinación y se aplicó un buscador aleatorio a los hiperparámetros más relevantes de cada algoritmo de aprendizaje automático y se escogió por cada uno de ellos la configuración con el mejor ajuste, para cada búsqueda se utilizaron los datos de entrenamiento los cuales corresponden al 80% del total de imágenes de FMRI, tal como se expresa en la figura 20, donde se observa que por cada búsqueda se aplica validación cruzada.

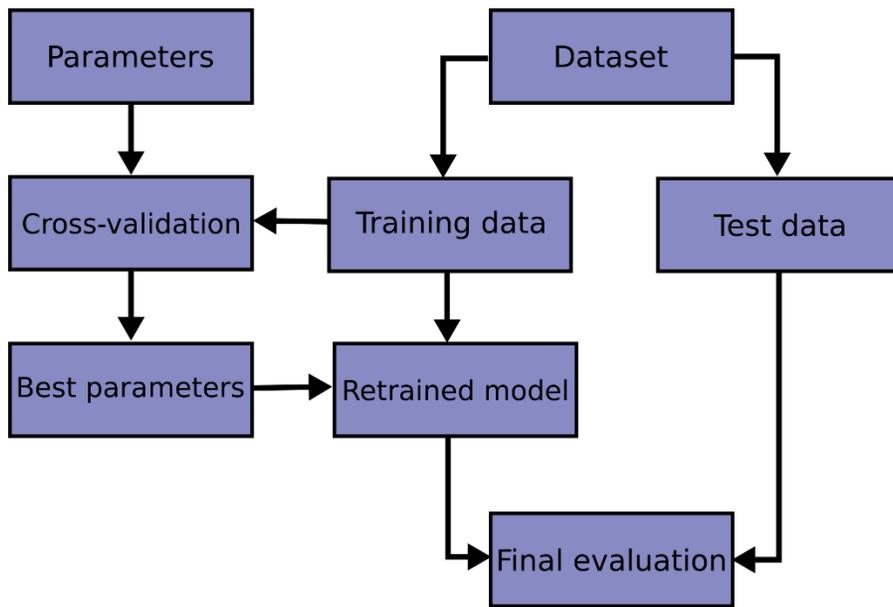


Fig. 20. Configuración de algoritmos utilizando validación cruzada

Fuente: (Pedregosa, F. and Varoquaux, 2011)

En la tabla 11 se observa el Pipeline (Combinación de algoritmo, escalador y reductor de dimensión), la mejor configuración de hiperparámetros por cada pipeline y su respectiva exactitud (accuracy), área bajo la curva de roc (Auc), combinación entre precisión recall (F1 score) y tiempo de predicción (Tpred). Se resalta que las métricas anteriormente enunciadas se

obtuvieron de imágenes no vistas en el entrenamiento las cuales corresponden al 20% del total de imágenes.

TABLA XI DESCRIPCIÓN DE RESULTADOS DE CADA UNO DE LOS MODELOS EXPERIMENTOS

Pipeline	Hiperparametros	Accuracy %	Auc %	F1 %	Tpred
SVC	Kernel=linear, C=1	0.375	0.58	0.28	0,0012 6 402
RandomForest	estimators=10, max depth=10, criterion=Gini	0.5	0.667	0.50	0,0004 0 408
DecisionTree	Max Depth=9, criterion= Gini	0.5	0.833	0.28	0,01162 6 59
LogisticRegres sion	Penalty=l2, C=9	0.375	0.500	0.50	0,0020 0 007
LSTM	units=70, dropout=0.4, recurrent_dropout=0.15,return_sequences=True,input_shape=(t_shape,RSN_shape), units=2, activation=sigmoid,loss='binary_crossentropy', optimizer=optimizers.Adam(lr=0.001),metrics=['binary_accuracy'])	0.625	0.62	0.28	0,0004 6 05
GRU	units=70, dropout=0.4, recurrent_dropout=0.15,return_sequences=True,input_shape=(t_shape,RSN_shape), units=2, activation=sigmoid,loss='binary_crossentropy', optimizer=optimizers.Adam(lr=0.001),metrics=['binary_accuracy'])	0.875	0.75	0.97	0,0043 9 98
Experimento con StandarScaler					
Pipeline	Hiperparametros	Accuracy %	Auc %	F1 %	Tpred
SVC	kernel=linear,C=1	0.375	0.58	0.28	0,0075 875
RandomForest	estimators=10,max_depth=13,criterion=Gini	0.375	0.417	0.00	0,0041 0 868
DecisionTree	max_depth=13, criterion=Gini	0.625	0.583	0.28	0,0038 6 268
LogisticRegres sion	penalty=l2, C= 9	0.375	0.750	0.28	0,0017 6 182
LSTM	units=70, dropout=0.4, recurrent_dropout=0.15,return_sequences=True,input_shape=(t_shape,RSN_shape), units=2, activation=sigmoid,loss='binary_crossentropy', optimizer=optimizers.Adam(lr=0.001),metrics=['binary_accuracy'])	0.375	0.775	0.44	0,00115 4 92

GRU	units=70, dropout=0.4, recurrent_dropout=0.15,return_sequences=True,input_shape=(t_shape,RSN_shape), units=2, activation=sigmoid,loss='binary_crossentropy', optimizer=optimizers.Adam(lr=0.001),metrics=['binary_accuracy'])	0.25	0.62	0.40	0,0064 896
Experimento con StandarScaler y PCA					
Pipeline	Hiperparametros	Accuracy %	Auc %	F1 %	Tpred
SVC	kernel=linear,C=100,clusters=13	0.25	0.67	0.50	0,0030 596
RandomForest	estimators=10,max_depth=10,criterion=entropy,clusters=7	0.25	0.773	0.80	0,0023 226
DecisionTree	clusters=5,decsT_max_depth=6,decsT_criterion=Gini	0.625	0.583	0.28	0,0001 399
LogisticRegression	penalty=l2,logistic_C=13,clusters=17	0.375	0.333	0.28	0,0004 007

Fuente: Esta investigación

Una vez se obtuvo las métricas de los experimentos se observa que el algoritmo que más se ajustó al modelo de datos de entrenamiento fue la del algoritmo Deep learning GRU, con una exactitud de 87%, un área bajo la curva de 75%, precisión de 97% y un tiempo de predicción del 0,0043 para observar el experimento completo visite GitHub. (2023, 07 de febrero). <https://github.com/gian097/ExperimentoTesis-.git>

Con los datos anteriormente obtenido se realizó la implementación del algoritmo Gru (Gated Recurrent Unit), el cual es una variación avanzada de las SRRN (Redes Neuronales Recurrentes Estándar). Este algoritmo memoriza todas las entradas que tiene gracias a una memoria interna que posee para todo tipo de problemas de aprendizaje autónomo con datos en series de tiempo, además tiende hacer un algoritmo robusto para gestionar y controlar el flujo automatización entre las células de la red neuronal.

Como se puede observar para probar el algoritmo Gru con el conjunto de datos se determinó usar el 20% y una un total de 50 épocas como parámetros para la evaluación del modelo previamente seleccionado, una vez terminada la evaluación del modelo se observó un aumento en la precisión del modelo y una disminución en la pérdida del modelo a medida que aumenta el número de épocas. Además, tanto los conjuntos de entrenamiento como los de validación muestran tendencias similares. Por lo tanto, se tuvo un modelo que funciona y probablemente no está sobre

ajustado, ya que la brecha en los datos de entrenamiento y prueba es corta (0.07) en su exactitud véase figura 21

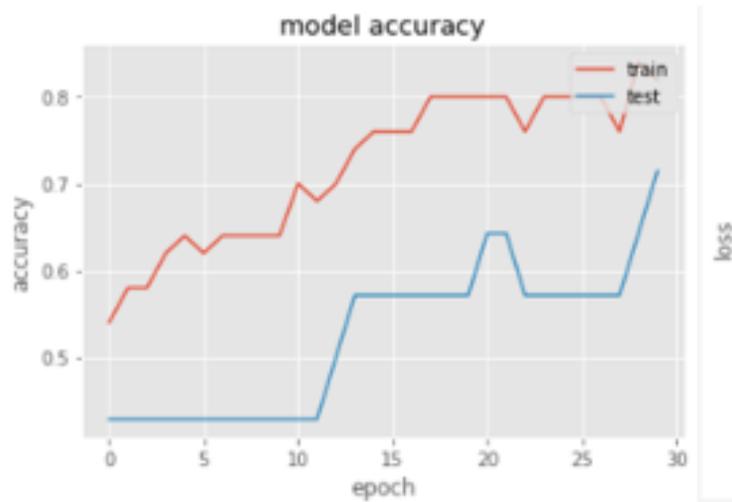


Fig. 21. Grafica de evolución del algoritmo GRU con el conjunto de datos ADHD-200

Fuente: Esta investigación

Para terminar, se guardó el modelo previamente entrenado con un método llamado “modelGRU.save” el cual se encargó de guardar el modelo con el nombre gru.h5 y una ruta para alojarlo en una carpeta con el fin de ser utilizado para la predicción de las neuro imágenes.

D. Implementar de la interfaz web.

En esta sección se detalla el diseño de la estructura y arquitectura de la interfaz web para la predicción del diagnóstico del TDAH, y las tecnologías usadas para el funcionamiento y comunicación entre ellas. Los detalles técnicos en cuanto a la implementación del prototipo de página web se concretan en el siguiente apartado.

El funcionamiento básico del sistema es el siguiente:

- El frontend, para satisfacer la petición del usuario, solicita datos a través de flask.

- flask es el encargado de recibir una petición de un cliente y se comunica con el servidor de predicción o con la base de datos.
- En caso de que la petición sea de predicción, el servidor de predicción se comunica con la base de datos para obtener los datos necesarios para realizar la misma (El funcionamiento del servidor de predicción se especifica en la sección 3)
- El flask obtiene la respuesta del servidor de predicción o de la base de datos y responde con los datos obtenidos al cliente que realizó la petición.
- El frontend recibe la imagen y la muestra el resultado en forma de porcentajes.

La arquitectura que se utilizó está basada en el sistema API REST “estilo arquitectónico basado en el protocolo cliente-servidor”[25]. Donde cada petición realizada por el cliente es independiente de cualquier otra y donde el servidor establece una API de comunicación en este caso se desarrolló dicha comunicación con el micro framework Flask como servidor web. Esta Arquitectura se compone de tres capas, en su primera capa se tiene la de presentación que corresponde con el frontend (cliente), la segunda capa es la lógica que corresponde con el servidor de predicción, por último, la tercera capa de persistencia que corresponde con la base de datos. Flask como backend hizo la función de servidor web que establece una conexión API para poder ser usado, y el enviar peticiones al servidor de predicción o a la base de datos según se requiera, haciendo estas arquitecturas invisibles para el cliente. Ya que se trata de un sistema REST, tanto el cliente como el servidor son componentes independientes. Gracias a esto, el cliente o el servidor se puede cambiar por otras tecnologías que se comporte de la misma forma y la arquitectura sigue funcionando. Por tanto, no dependen el uno del otro para poder funcionar correctamente de manera independiente. Así, el frontend puede desarrollarse en la tecnología que se requiera o podrán hacerse cambios en el mismo, pero podrá seguir comunicándose con el backend mientras sepa cómo usar su API y viceversa, sin afectar a otros componentes. El backend también se puede comunicar con un servidor de predicción y con una base de datos cualquiera de la misma forma. En la figura 22 se representa de forma gráfica la arquitectura del sistema, así como las comunicaciones entre las distintas tecnologías que lo conforman.

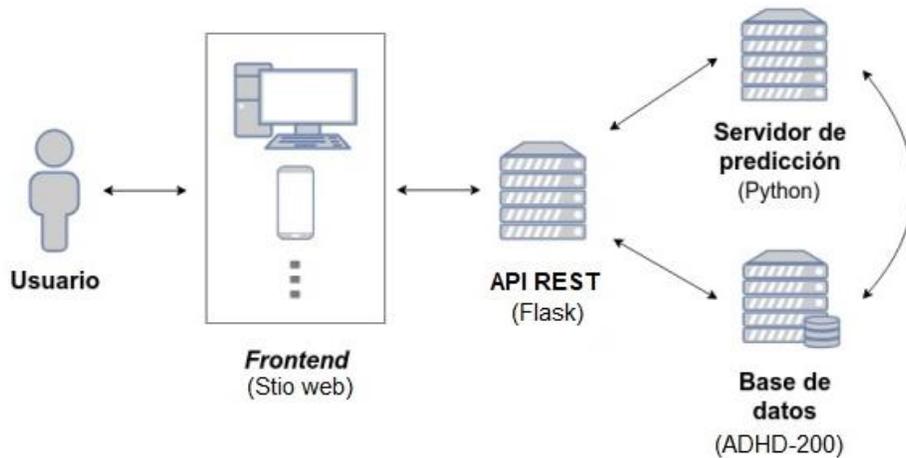


Fig. 22. Arquitectura cliente servidor

Fuente: Arquitectura de tecnologías requeridas para sistema de predicción para el diagnóstico de TDAH con imágenes FMRI.

El sistema se basó en la arquitectura cliente servidor, el cual uso las siguientes tecnologías:

- ADHD -200, el cual fue una base de datos diseñada por el grupo Nilearn como módulo de Python.
- Flask, es un micro Framework para Python que proporciona características para hacer Api de manera rápida y ligera para aplicaciones o páginas web que no requieran muchas extensiones o servicios.
- Sitio web, desarrollado en jinja2.
- Servidor de predicción desarrollado en el lenguaje Python.
- Una base de datos en my sql que recopila datos de la predicción.

A continuación, se observa cómo fue implementada la interfaz web que se desarrolló en base a la arquitectura anteriormente planteada.

Para comenzar el desarrollo del API REST, primero se definió una función que se llama model predict para que se encargue de recibir la neuro imagen y las variables molestas para poder procesarlas y enviar una predicción como se observa en la figura 23.

```

# Realizamos la predicción usando la imagen cargada y el modelo
def model_predict(file_path1,file_path2, path):
    smith_atlas = datasets.fetch_atlas_smith_2009()
    smith_atlas_rs_networks = smith_atlas.rsn70
    smith_atlas_rs_networks

    Xp=preprocessing(file_path1,file_path2,smith_atlas)
    model_load=keras.models.load_model(path+"model.h5")

    pred=model_load.predict(Xp)

    return pred

```

Fig. 23. Función para hacer la predicción de la neuroimagen

Fuente: Esta investigación

Una vez obtenida la predicción, se creó una ruta para enviar está a la interfaz, además recopiló una serie de datos que son nombre, apellido, genero, fecha, cedula, dirección, ciudad, rh, imagen, predicción y diagnóstico para enviarlo a la base de datos y realizar por medio de un médico la validación de la predicción de este paciente, observe la figura 24.

```

@app.route('/predict', methods=['GET', 'POST'])
def upload():
    if request.method == 'POST':
        nombre = request.form['nombre']
        apellido = request.form['apellido']
        genero = request.form['genero']
        fecha = request.form['fecha']
        cedula = request.form['cedula']
        direccion = request.form['direccion']
        ciudad = request.form['ciudad']
        rh = request.form['rh']

        f = request.files['file']
        nombreArchivo=secure_filename(f.filename)
        print(nombreArchivo)
        file_path1 = os.path.join(app.config['UPLOAD_FOLDER'],nombreArchivo)
        f.save(file_path1)

        f1 = request.files['file1']
        nombreArchivo1=secure_filename(f1.filename)
        print(nombreArchivo1)
        file_path2 = os.path.join(app.config['UPLOAD_FOLDER'],nombreArchivo1)
        f1.save(file_path2)
        preds = model_predict(file_path1,file_path2, path)
        preds = np.round(pred, decimals = 2)
        result = json.dumps(str(preds))
        diagnostico=" "
        cur = mysql.connection.cursor()
        cur.execute('INSERT INTO pacientes (nombre, apellido, genero, fecha, cedula, direccion, ciudad, rh, imagen, predicion, diagnosticd)
VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)',(nombre, apellido, genero, fecha, cedula, direccion, ciudad, rh, file_path1,result,diagnostico))
        mysql.connection.commit()
        return result
    return redirect(url_for('validar'))

```

Fig. 24. Creación de ruta para mostrar la predicción a la interfaz web

Fuente: Esta investigación

La interfaz web permite hacer un diagnóstico del TDAH gracias al modelo de machine learning obtenido. Para hacer esta predicción se debe cargar una serie de datos del paciente y posterior a esto una neuroimagen y sus variables molestas como se observa en la figura 25.

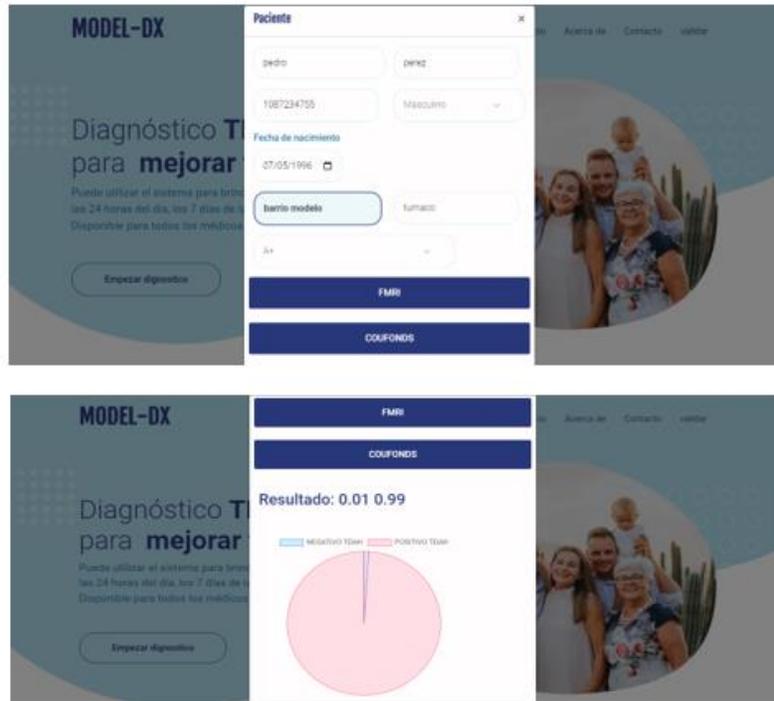


Fig. 25. Interfaz web para la predicción de pacientes con TDAH

Fuente: Esta investigación.

Una vez cargado los datos, la interfaz muestra la cantidad de TDAH del paciente para después poder hacer una validación médica y así dictaminar si la predicción es correcta del diagnóstico de TDAH, para hacer la validación se creó una vista la cual muestra la cantidad de pacientes valorados con sus respectivos datos y su predicción, además el personal médico podrá validar la predicción si esta es positiva para TDAH o negativa para TDAH como se observa en la figura 26 y 27.

#	Nombre	Apellido	Genero	Fecha de nacimiento	Cedula	Direccion	Ciudad	RH	Imagen	Prediccion	
1	carlos	perez	Masculino	1996-05-07	1087234755	barrio modelo	tumaco	A+	./uploads/0023008_rest_tshift_RPI_voreg_mni.nii.gz	"[0.35 0.65]"	validar DX
2	pedro	perez	Masculino	1996-05-07	1087234755	barrio modelo	tumaco	A+	./uploads/0027037_rest_tshift_RPI_voreg_mni.nii.gz	"[0.01 0.99]"	validar DX

Fig. 26. Proceso de validación según criterio medica del paciente

Fuente: Esta investigación.

Fig. 27. Validación paciente

Fuente: Esta investigación

ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS

En la presente investigación fue necesario aplicar técnicas de preprocesamiento para llevar las características de una imagen de resonancia electromagnética funcional (FMRI) a un conectoma con un filtro Atlas Smith de BrainMap, como lo especifica[26], no obstante con los actuales avances tecnológicos se podría utilizar una arquitectura de red que logre extraer automáticamente las características y de esa manera aprovechar la señal de 4 dimensiones de una FMRI como lo especifica[19]. Sin embargo, los resultados de estos artículos aún no son contundentes.

A diferencia de la investigación[27] en ese estudio se realizaron experimentos con varios algoritmos de aprendizaje automático y Deep learning, y acorde a[28] los algoritmos más apropiados para realizar el entrenamiento son los Deep Learning, ya que como se observa en la figura 26, LSTM y GRU tienen las mejores métricas de calidad, y es regular en cada una de ellas, esto quiere decir que los resultados son buenos tanto en *F1 score*, *AUC* y *Accuracy*.

De igual forma a diferencia de[29] en esta investigación se realizó 3 experimentaciones que consistieron, primero, una experimentación sin preprocesamiento, segundo, una experimentación con escalado estándar, y tercero, una experimentación con escalado estándar y reducción de dimensión con PCA, y se obtuvo que estos preprocesamiento empeoran el rendimiento general de todos los algoritmos, esto puede haber sucedido porque el filtro Smith ya hace un escalado y reducción de dimensión como lo expresa[30]. Aunque los resultados fueron contrarios a lo esperado (mejora de métricas de calidad), se aporta con los indicadores y configuraciones correspondientes para que puedan ser replicados o referenciados en otras investigaciones similares.

A diferencia de otros frameworks, Flask se considera una herramienta de fácil comprensión para crear API REST de manera más sencilla. Es un macro web ligero, escrito en el lenguaje de programación Python, que permite crear sitios web escalables. Aunque no depende de bibliotecas externas, cuenta con extensiones como sistemas abiertos de autenticación, validaciones de formularios, mapeadores de relaciones de objetos, mecanismos de carga y otras herramientas. Además, Flask se basa en el kit de herramientas WSGI y el motor de plantillas Jinja2.

CONCLUSIONES

El objetivo de la presente investigación era aportar desde la inteligencia artificial una herramienta que facilite al diagnóstico del Trastorno por Déficit de Atención e Hiperactividad (TDAH), evaluando algoritmos de machine learning y Deep learning. Esta evaluación consistió en realizar una clasificación de distintas técnicas supervisadas y no supervisadas, donde se pudo extraer distintas métricas que ayudaron a la selección del mejor algoritmo para lograr así una predicción del TDAH.

Para ello, fue necesario tomar prestados una serie de estudios que permitieron darle un enfoque apropiado a la investigación. En este sentido, los resultados obtenidos muestran que el algoritmo GRU tuvo un rendimiento con una exactitud del 85.7%, un AUC del 75%, y un F1 score del 97.9%, significativamente mejorando así los resultados del estudio[27] donde el algoritmo LSTM tuvo una exactitud 70%, con estos resultados demuestran la eficacia de GRU en la clasificación de pacientes con TDAH.

Esta investigación demuestra que Machine Learning, es una herramienta valiosa en el campo de la salud mental y en el diagnóstico médico. En particular, el uso de algoritmos de aprendizaje automático en el análisis de datos clínicos ha permitido identificar patrones y correlaciones que de otra manera serían difíciles de detectar. Esto es especialmente importante en el campo de las enfermedades mentales, donde los síntomas pueden ser sutiles y los diagnósticos a menudo requieren la evaluación de múltiples factores.

Además, los algoritmos de aprendizaje automático pueden ayudar a mejorar la precisión del diagnóstico y la eficacia de los tratamientos. Esto se debe a que los algoritmos pueden procesar grandes cantidades de datos de forma rápida y precisa, lo que permite a los médicos y profesionales de la salud obtener una comprensión más completa de la situación de cada paciente.

La herramienta propuesta en esta investigación es valiosa en el diagnóstico de TDAH y brinda un enfoque innovador para abordar este problema de salud mental. Los resultados obtenidos son prometedores y sugieren que la inteligencia artificial puede tener un papel importante en el futuro de la atención médica y en la mejora de la calidad de vida de las personas con TDAH.

RECOMENDACIONES

Después de analizar los resultados, se proponen ciertos aspectos que podrían mejorarse para aumentar la precisión del diagnóstico. En primer lugar, sería conveniente explorar una red convolucional que pudiera clasificar las imágenes de Casos y Controles con alta precisión y varias capas convolucionales. De esta manera, el número de características obtenidas después de la última capa convolucional sería suficiente para entrenar la GRU. Utilizando este enfoque, se podrían obtener las características más importantes de las imágenes mediante una red que se ha entrenado con el conjunto de datos estudiado. Además, otro aspecto a tener en cuenta sería la inclusión de varias capas GRU para ver si esto puede mejorar la precisión en los datos de prueba y además las siguientes recomendaciones:

Se podría trabajar en mejorar el coste computacional del preprocesamiento de neuroimágenes, ya que este proceso actualmente requiere de muchos recursos a nivel de hardware. Esto se podría lograr mediante la optimización de algoritmos y la implementación de técnicas de paralelización para acelerar el procesamiento.

Para futuras investigaciones, se podría crear un conjunto de datos propio con un preprocesamiento mejorado y con un análisis de calidad óptimo. Esto podría ayudar a asegurar que los datos sean más precisos y confiables, lo que a su vez podría mejorar los resultados de las investigaciones.

Sería útil desarrollar un estándar para el procesamiento de neuroimágenes. Esto ayudaría a garantizar que el preprocesamiento de datos sea consistente y comparable entre estudios, lo que podría mejorar la reproducibilidad y la comparabilidad de los resultados de las investigaciones.

Se podría desarrollar una herramienta de preprocesamiento de neuroimágenes más intuitiva, que permita a los investigadores preprocesar datos sin necesidad de tener un conocimiento técnico profundo. Esto podría hacer que el preprocesamiento de neuroimágenes sea más accesible para un público más amplio de investigadores, lo que podría conducir a un mayor número de estudios en el área.

BIBLIOGRAFÍA

- [1] «1. Introducción - GuíaSalud». <https://portal.guiasalud.es/egpc/tdah-introduccion/> (accedido 11 de agosto de 2023).
- [2] M. J. F. Asorey y P. F. Fernández, *Dificultades de aprendizaje y trastornos del desarrollo: manual didáctico*. Pirámide, 2017.
- [3] «Causas de una baja autoestima en niños con TDAH». <https://www.fundacioncadah.org/web/articulo/causas-de-una-baja-autoestima-en-ninos-con-tdah.html> (accedido 11 de agosto de 2023).
- [4] B. Sen, N. C. Borle, R. Greiner, y M. R. G. Brown, «A general prediction model for the detection of ADHD and Autism using structural and functional MRI», *PLoS ONE*, vol. 13, n.º 4, 2018, doi: 10.1371/journal.pone.0194856.
- [5] «Decreto 366 de 2009 - Gestor Normativo - Función Pública». <https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=35084> (accedido 26 de abril de 2023).
- [6] M. M. Boix *et al.*, *Hiperactividades y déficit de atención: Comprendiendo el TDAH*. Ediciones Octaedro, 2014.
- [7] E. Proal, J. G. Olvera, A. S. Blancas, P. J. Chalita, y F. X. Castellanos, «Neurobiología del autismo y TDAH mediante técnicas de neuroimagen: divergencias y convergencias», *Rev Neurol*, vol. 57, n.º 01, pp. S163-S175, sep. 2013.
- [8] S. R. Timarán Pereira, I. Hernández Arteaga, S. J. Caicedo Zambrano, A. Hidalgo Troya, y J. C. Alvarado Pérez, *Descubrimiento de patrones de desempeño académico con árboles de decisión en las competencias genéricas de la formación profesional*. Universidad Cooperativa de Colombia, 2016. doi: 10.16925/9789587600490.
- [9] «Towards interpretable machine learning models for diagnosis aid: A case study on attention deficit/hyperactivity disorder | PLOS ONE». <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0215720> (accedido 22 de agosto de 2023).

- [10] H. Ke, F. Wang, H. Ma, y Z. He, «ADHD identification and its interpretation of functional connectivity using deep self-attention factorization», *Knowledge-Based Systems*, vol. 250, p. 109082, ago. 2022, doi: 10.1016/j.knosys.2022.109082.
- [11] A. Kaur y K. S. Kahlon, «Accurate Identification of ADHD among Adults Using Real-Time Activity Data», *Brain Sciences*, vol. 12, n.º 7, Art. n.º 7, jul. 2022, doi: 10.3390/brainsci12070831.
- [12] S. C, B. S, A. Ur, A. H, y B. Gm, «Diagnosis of attention deficit hyperactivity disorder using imaging and signal processing techniques», *Computers in biology and medicine*, vol. 88, ene. 2017, doi: 10.1016/j.combiomed.2017.07.009.
- [13] J. Costa Rosa, «Sistema de ayuda a la diagnosis y prognosis de diferentes enfermedades aplicando técnicas de Machine Learning», Proyecto/Trabajo fin de carrera/grado, Universitat Politècnica de València, 2018. Accedido: 22 de agosto de 2023. [En línea]. Disponible en: <https://riunet.upv.es/handle/10251/106953>
- [14] P. Amado Caballero, «Ayuda al diagnóstico de TDAH en la infancia mediante técnicas de procesado de señal y aprendizaje», 2019, Accedido: 22 de agosto de 2023. [En línea]. Disponible en: <https://uvadoc.uva.es/handle/10324/38774>
- [15] «Imagen por resonancia magnética funcional (fMRI) | Memorial Sloan Kettering Cancer Center». <https://www.mskcc.org/es/cancer-care/patient-education/functional-magnetic-resonance-imaging-fmri> (accedido 12 de agosto de 2023).
- [16] «Algoritmos de rastreo de movimiento utilizando técnicas de inteligencia artificial y machine learning». https://www.scielo.cl/scielo.php?pid=S0718-07642020000300023&script=sci_arttext&tlng=pt (accedido 12 de agosto de 2023).
- [17] «Aprendizaje no supervisado en el perfilamiento de clientes para profit scoring: caso de estudio de una Fintech Latinoamericana». <https://repositorio.uniandes.edu.co/handle/1992/51481> (accedido 12 de agosto de 2023).
- [18] G. Deshpande, P. Wang, D. Rangaprakash, y B. Wilamowski, «Fully Connected Cascade Artificial Neural Network Architecture for Attention Deficit Hyperactivity Disorder Classification From Functional Magnetic Resonance Imaging Data», *IEEE Transactions on Cybernetics*, vol. 45, n.º 12, pp. 2668-2679, dic. 2015, doi: 10.1109/TCYB.2014.2379621.

- [19] «Redes neuronales convolucionales para la clasificación de componentes independientes de rs-fMRI». <https://www.redalyc.org/journal/3442/344264813015/html/> (accedido 6 de mayo de 2023).
- [20] S. Dobilas, «LSTM Recurrent Neural Networks — How to Teach a Network to Remember the Past», *Medium*, 5 de marzo de 2022. <https://towardsdatascience.com/lstm-recurrent-neural-networks-how-to-teach-a-network-to-remember-the-past-55e54c2ff22e> (accedido 16 de agosto de 2023).
- [21] «Gated Recurrent Unit Networks - GeeksforGeeks». <https://www.geeksforgeeks.org/gated-recurrent-unit-networks/> (accedido 22 de agosto de 2023).
- [22] G. Karni, «Identifying Resting-State Networks from fMRI Data Using ICAs», *Medium*, 24 de abril de 2020. <https://towardsdatascience.com/exploring-cognitive-differences-via-resting-state-networks-2112bf5291e2> (accedido 24 de agosto de 2023).
- [23] P. Lara Benítez, «Predicción de series temporales en streaming mediante Deep Learning», jun. 2022, Accedido: 5 de mayo de 2023. [En línea]. Disponible en: <https://idus.us.es/handle/11441/136449>
- [24] J. Z. Ruiz, «COMPARATIVA Y ANÁLISIS DE ALGORITMOS DE APRENDIZAJE AUTOMÁTICO PARA LA PREDICCIÓN DEL TIPO PREDOMINANTE DE CUBIERTA ARBÓREA».
- [25] «Arquitectura Cliente-Servidor». <https://reactiveprogramming.io/blog/es/estilos-arquitectonicos/cliente-servidor> (accedido 5 de mayo de 2023).
- [26] «BrainMap + RSNs». <https://www.fmrib.ox.ac.uk/datasets/brainmap+rsns/> (accedido 6 de mayo de 2023).
- [27] «Classifying ADHD from Healthy Controls using LSTMs with rs-fMRI Data | by Gili Karni | Towards Data Science». <https://towardsdatascience.com/classifying-adhd-from-healthy-controls-using-lstms-with-rs-fmri-data-300c1f3e9697> (accedido 6 de mayo de 2023).
- [28] S. Saurabh y P. K. Gupta, «Deep Learning-Based Modified Bidirectional LSTM Network for Classification of ADHD Disorder», *Arab J Sci Eng*, abr. 2023, doi: 10.1007/s13369-023-07786-w.

[29] J. M. M. McWilliams, «Aplicación de redes neuronales convolucionales y recurrentes al diagnóstico de autismo a partir de resonancias magnéticas funcionales».

[30] A. Abraham, «Learning functional brain atlases modeling inter-subject variability».

ANEXOS

Selección de algoritmo

Esta sección muestra los experimentos que se realizaron para obtener la mejor técnica de aprendizaje automático y sus métricas de calidad para el manejo de datos y así seleccionar el mejor algoritmo para el desarrollo de esta investigación.

Se seleccionaron una serie de algoritmos como arboles de decisión, arboles aleatorios, maquinas vectoriales, regresión logística, GRU y LSTM. Donde se les aplico una serie de pruebas como Standard Scaler y PCA de los cuales se obtuvieron matrices de confusión, curva de roc y recall para medir el compromiso en sus métricas de calidad.

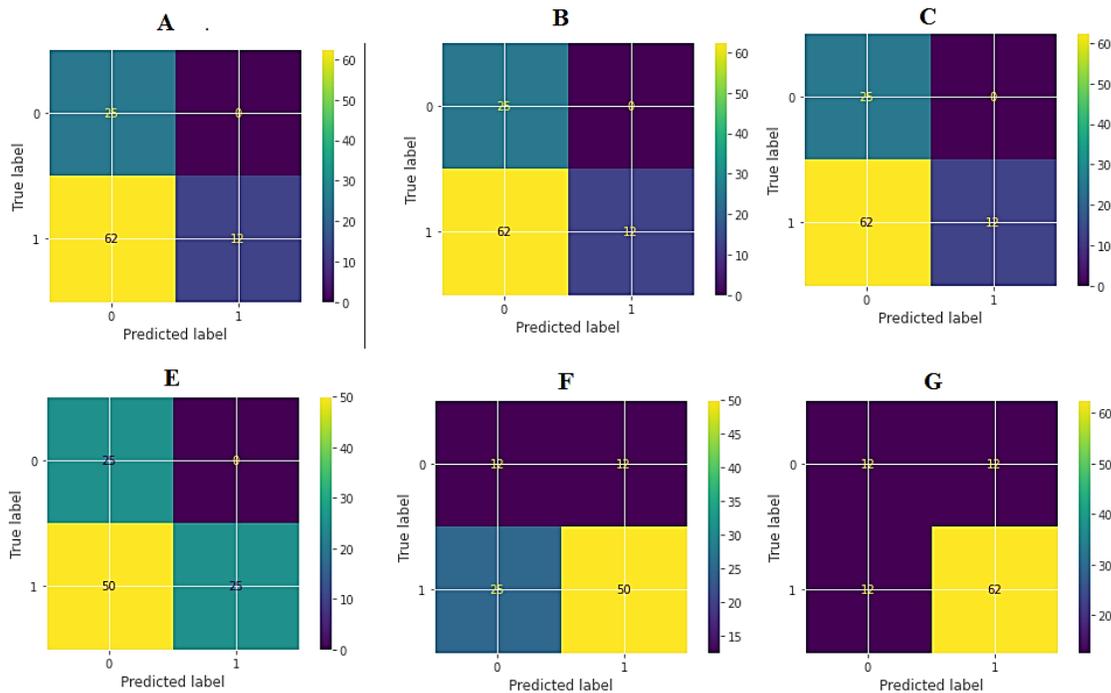


Fig. 1 Graficas de matrices de confusión de los modelos de machine learning.

Fuente: Esta investigación

En el siguiente experimento se aplicaron los seis (6) algoritmos sin Standard Scaler y PCA, los cuales se pueden observar en la figura 17, donde máquinas de soporte vectorial está indicada por la gráfica “A” , bosques aleatorios con la gráfica “B”, regresión logística con la gráfica “C”, arboles de decisión con la gráfica “D”, LSTM con la gráfica “E” y GRU con la gráfica “F”, de los

cuales se puede deducir que el mejor resultado en cuanto a la exactitud fue el algoritmo GRU, donde se tuvo como resultado de 87%, interpretando que este algoritmo es el mejor en este experimento frente a los demás, de las siguientes matrices de confusión se obtuvieron las métricas de Accuracy como se muestra en la Figura 26.

En la figura 18 se muestran las curvas roc referentes a los 6 algoritmos escogidos en esta investigación, de los cuales se extrajo el AUC (Area bajo la curva ROC), donde se sintetizan en la figura 26.

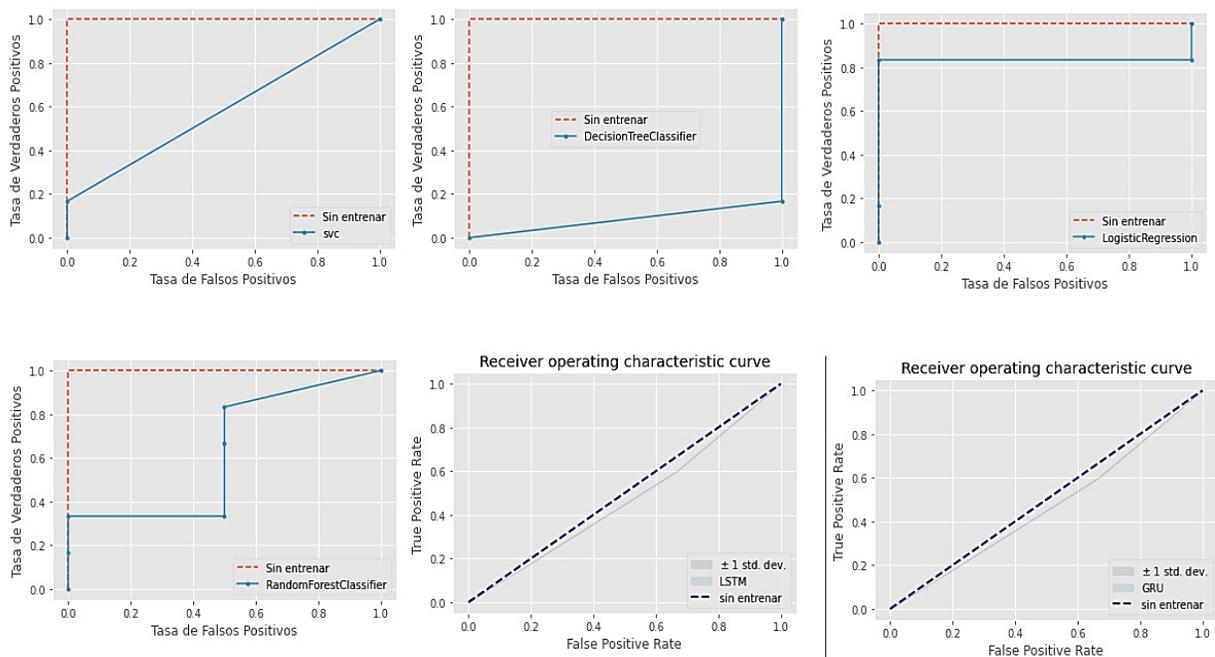


Fig. 2. Graficas de matrices de confusión de los modelos de machine learning

Fuente: Esta investigación.

En la figura 18 se observó los seis algoritmos, de los cuales se puede deducir que el mejor resultado en cuanto a Área bajo la curva (AUC) fue el algoritmo GRU, donde se tuvo como resultado un 75%, interpretando que este algoritmo es el mejor en este experimento frente a los demás. su visualización se puede apreciarse en la figura 26.

En la figura 19 se muestran las curvas de recall referentes a los 6 algoritmos escogidos en esta investigación, de los cuales se extrajo el f1 score, donde se sintetizan en la Figura 26.

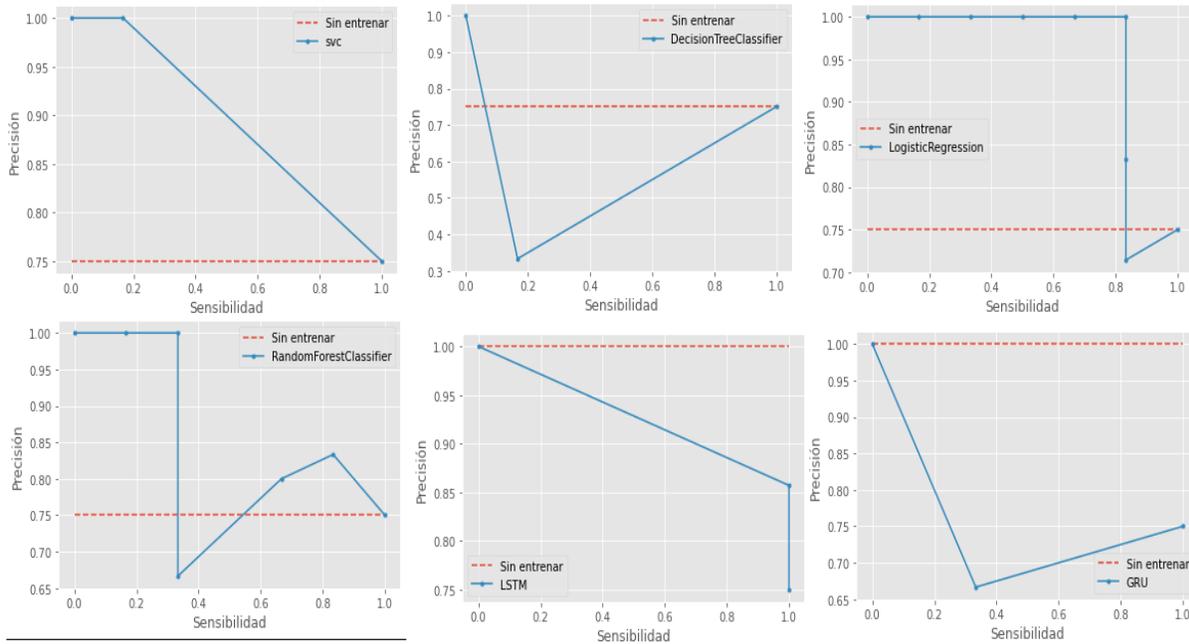


Fig. 3 Graficas de recall los modelos de machine learning

Fuente: Esta investigación.

En la figura 19 se observó los seis algoritmos, de los cuales se puede deducir que el mejor resultado en cuanto a F1 score (Precisión) fue el algoritmo GRU, donde se tuvo un resultado de 97.9%, interpretando que este algoritmo es el mejor en este experimento frente a los demás. su visualización se puede apreciarse en la figura 26.

En el siguiente experimento se aplicaron los seis (6) algoritmos con Standard Scaler, donde máquinas de soporte vectorial está indicada por la gráfica “A”, bosques aleatorios con la gráfica “B”, regresión logística con la gráfica “C”, arboles de decisión con la gráfica “D”, LSTM con la gráfica “E” y GRU con la gráfica “F”, de los cuales se puede deducir que el mejor resultado en cuanto a la exactitud fue el algoritmo arboles de decisión, donde se tuvo como resultado una exactitud de 62%, interpretando que este algoritmo es mejor frente a los demás. su visualización se puede apreciarse de la figura 20, donde se muestran las matrices de confusión referentes a los 6 algoritmos escogidos en esta investigación, de los cuales se extrajo el Accuracy, cuyos se sintetizan en la Figura 26.

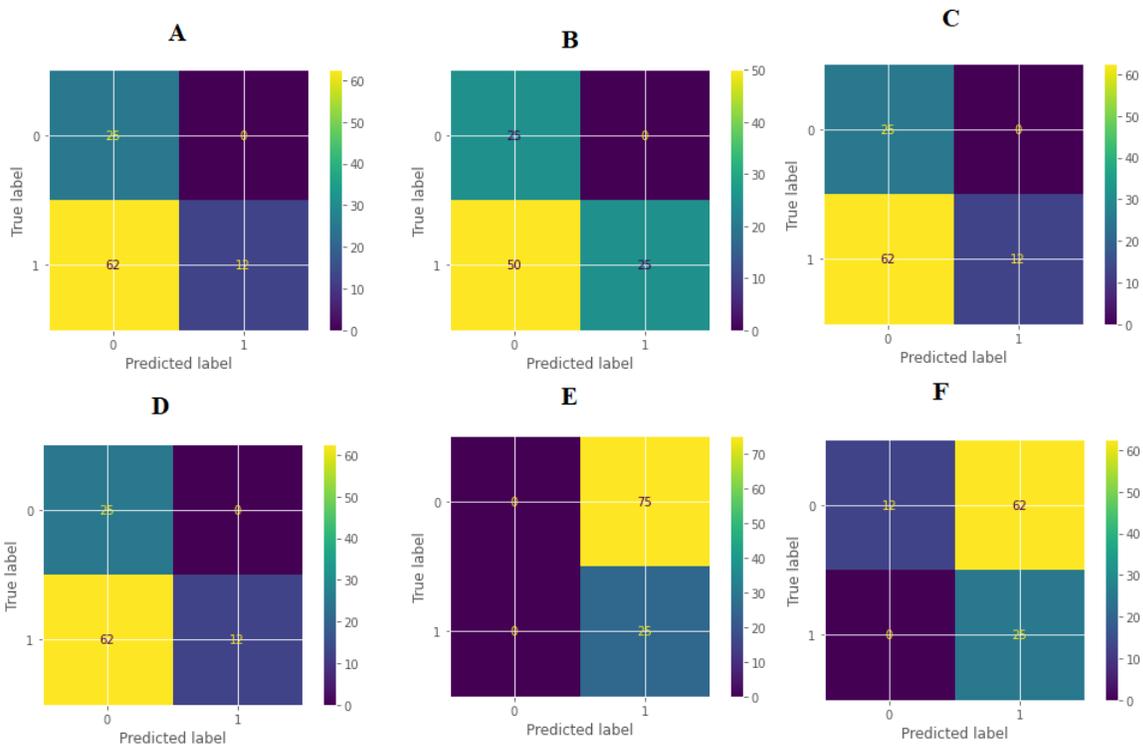


Fig. 4. Graficas de matrices de confusión mediante la tubería pipeline y el método standard scaler

Fuente: Esta investigación.

En la figura 21 se muestran las curvas roc referentes a los 6 algoritmos escogidos en esta investigación, de los cuales se extrajo el AUC (Area bajo la curva ROC), estos se sintetizan en la figura 26.

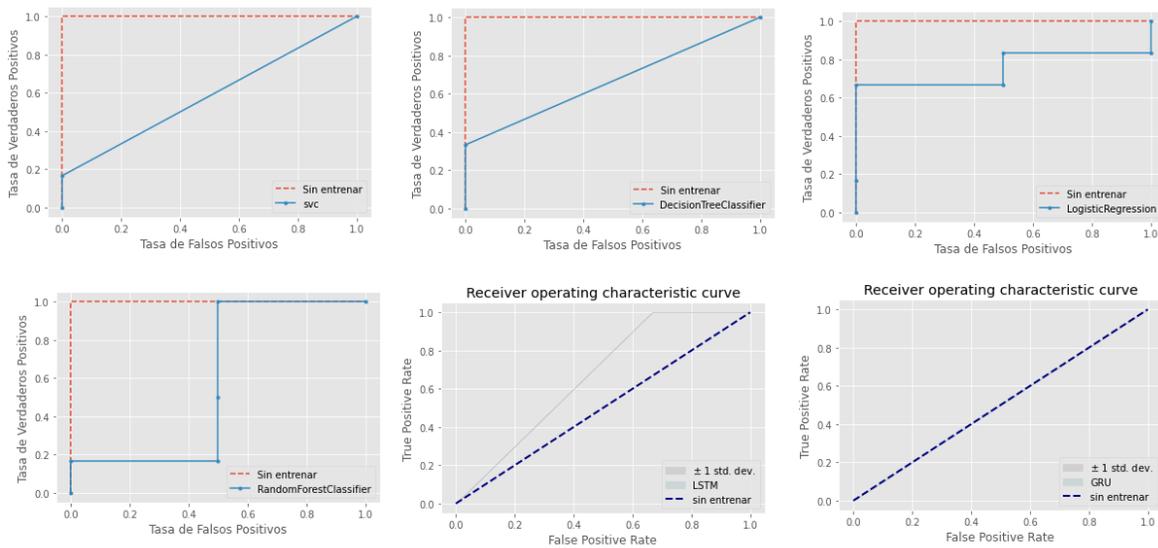


Fig. 5 Graficas de curvas de roc mediante la tubería pipeline y el método standard scaler

Fuente: Esta investigación

En la figura 21 se observó los seis algoritmos, de los cuales se puede deducir que el mejor resultado en cuanto a AUC (Area bajo la curva ROC), fue el algoritmo LSTM, donde se tuvo como resultado de 77%, interpretando que este algoritmo es el mejor en este experimento frente a los demás. su visualización se puede apreciarse en la figura 26.

En la figura 22 se muestran las curvas de recall referentes a los 6 algoritmos escogidos en esta investigación, de los cuales se extrajo el fl score, cuyos se sintetizan en la figura 26.

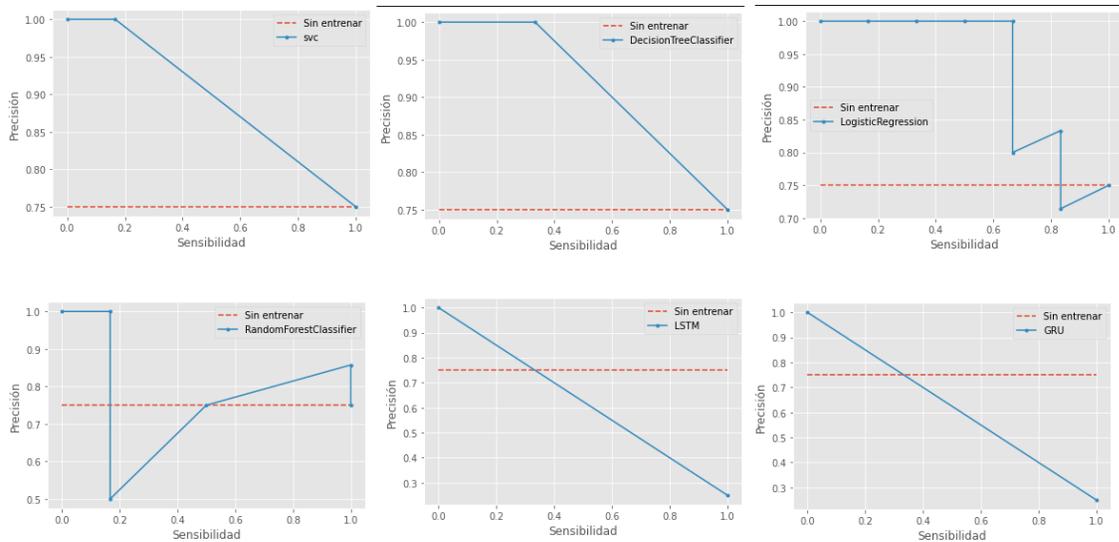


Fig.

6 Graficas de recall mediante la tubería pipeline y el método standard Scaler

Fuente: Esta investigación

En la figura 22 se observó los seis algoritmos, de los cuales se puede deducir que el mejor resultado en cuanto a F1 score (Precisión) fue el algoritmo LSTM, donde se tuvo como resultado de 44%, interpretando que este algoritmo es el mejor en este experimento frente a los demás. su visualización se puede apreciarse en la figura 26.

El experimento siguiente se aplicaron los 4 algoritmos con Standard Scaler y PCA, donde máquinas de soporte vectorial está indicada por la gráfica “A”, bosques aleatorios con la gráfica “B”, regresión logística con la gráfica “C”, arboles de decisión con la gráfica “D”, de los cuales se puede deducir que el mejor resultado en cuanto a la exactitud fue el algoritmo Árboles de decisión, donde se tuvo como resultado de 62%, interpretando que este algoritmo es el mejor en este experimento frente a los demás. su visualización se puede apreciarse en la figura 23, donde se pueden observar las matrices de confusión referentes a los 6 algoritmos escogidos en esta investigación, de los cuales se extrajo el Accuracy, donde se sintetizan en la figura 26.

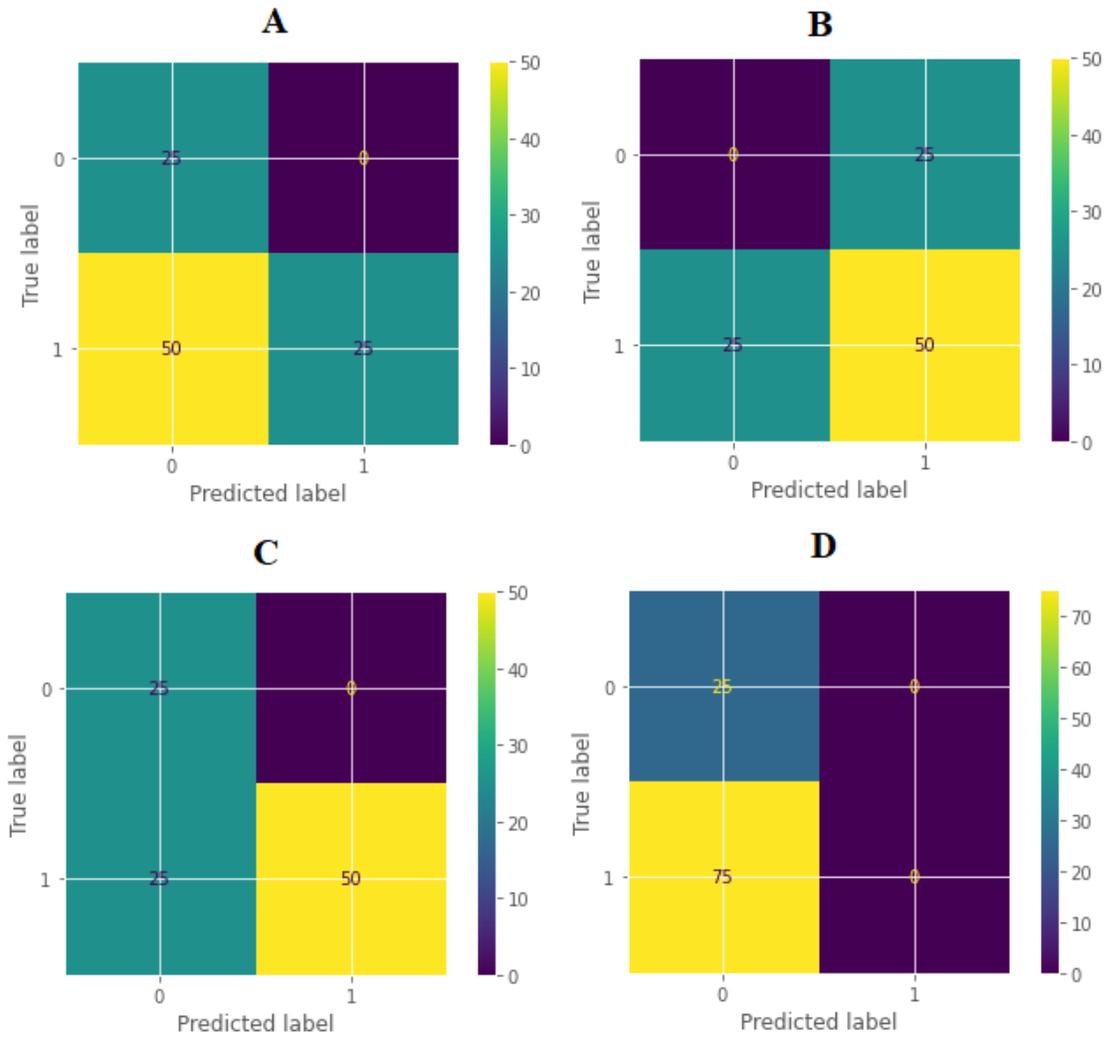


Fig. 7 Gráfica de matriz de confusión mediante método estándar scaler y PCA

Fuente: esta investigación

En la figura 24 se muestran las curvas roc referentes a los 6 algoritmos escogidos en esta investigación, de los cuales se extrajo el AUC (Area bajo la curva ROC), donde se sintetizan en la Figura 26.

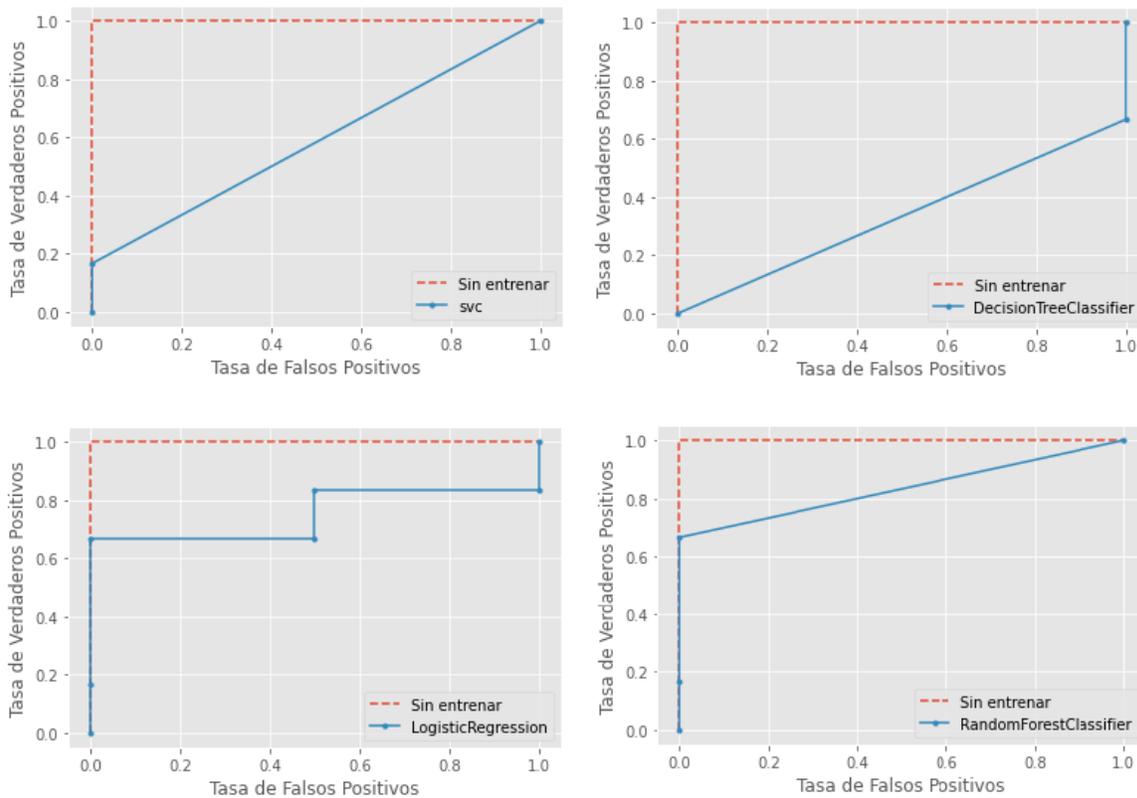


Fig. 8 Graficas de roc método estándar scaler y método PCA

Fuente: Esta investigación

En la figura 24 se observó los seis algoritmos, de los cuales se puede deducir que el mejor resultado en cuanto a AUC (área bajo la curva ROC), fue el algoritmo arboles aleatorios, donde se tuvo como resultado un 77%, interpretando que este algoritmo es el mejor en este experimento frente a los demás. su visualización se puede apreciarse en la figura 26.

En la figura 25 se muestran las curvas de recall referentes a los 4 algoritmos escogidos en esta investigación, de los cuales se extrajo el f1 score, cuyos se sintetizan en la figura 26.

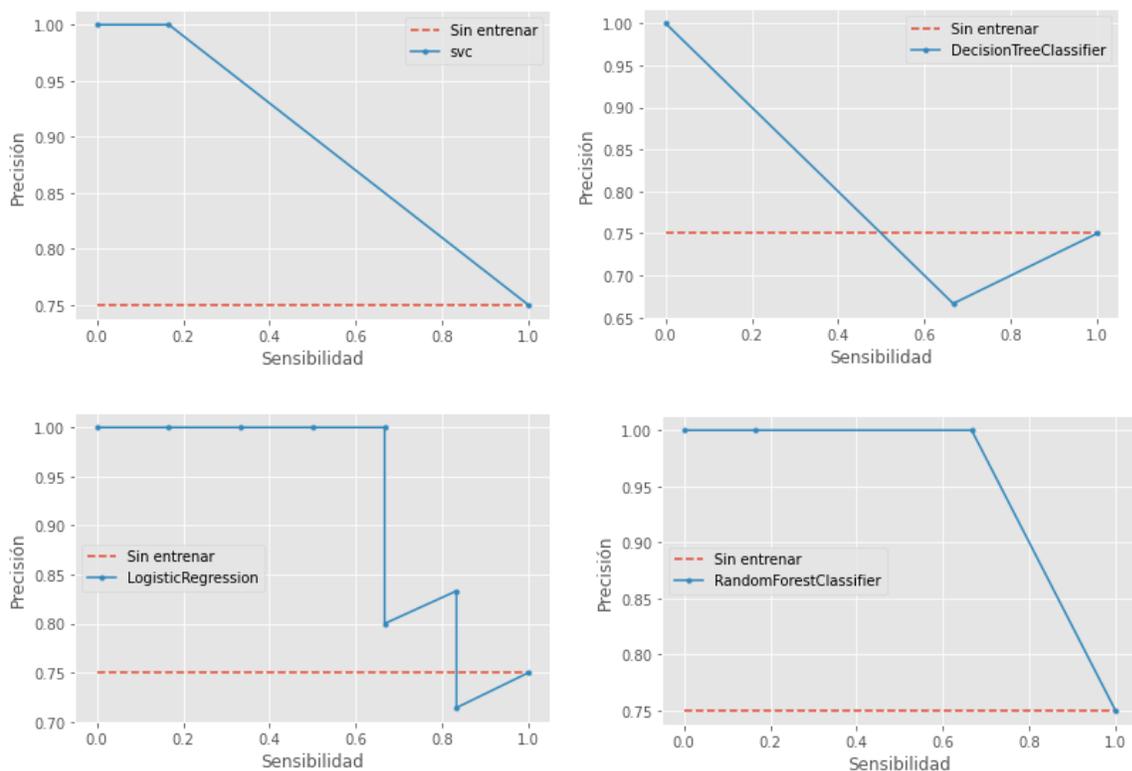


Fig. 9 Graficas de recall método estándar scaler y método PCA

Fuente: Esta investigación

A. **PROTOTIPO DE PAGINA WEB DE LA INVESTIGACION Y REPOSITORIO DEL MODELO**

Las siguientes imágenes se puede observar las diferentes vistas del prototipo de interfaz web de MODEL-DX de esta investigación



Anexo 1 Vista principal de prototipo de página web de la investigación

Fuente: Esta investigación, enlace <https://github.com/gian097/ModelDX>

Paciente ×

Nombre Apellido

Cedula Genero

Fecha de nacimiento

dd/mm/aaaa

Direccion Ciudad

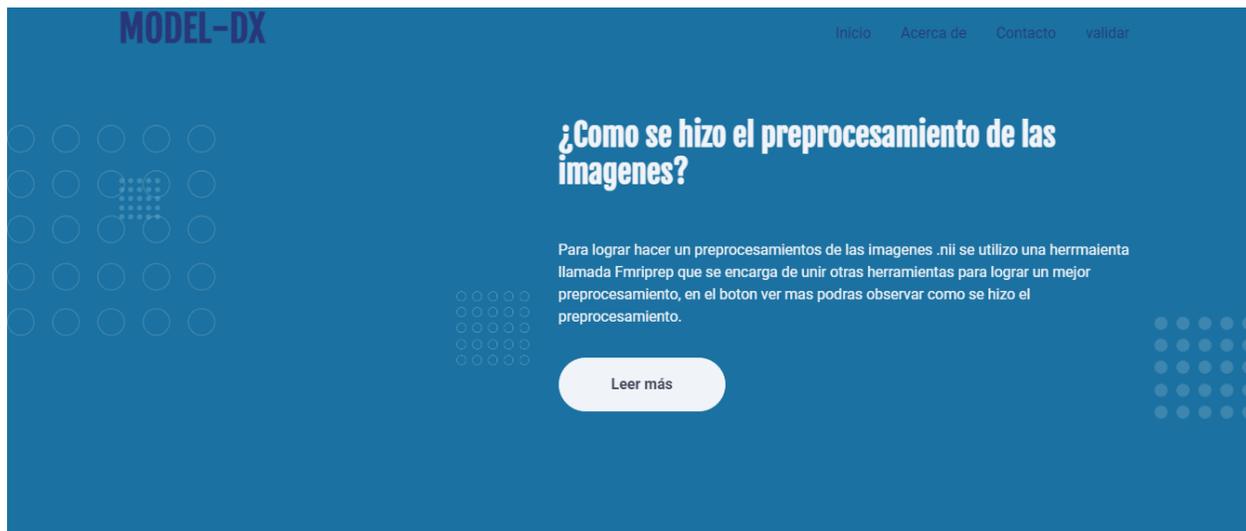
RH

FMRI

COUFONDS

Anexo 2 Vista para agregar datos personales de paciente

Fuente: Esta investigación



Anexo 3 Vista de cómo se realiza el preprocesamiento con la herramienta FmriPrep

Fuente: Esta investigación



Anexo 4 Vista de preprocesamiento botón LEER MAS

Fuente: Esta investigación

MODEL-DX Acerca de Inicio Acerca de Contacto validar

Estamos desarrollando una asistencia en línea, un sistema que ayudará...

Creemos que todo el mundo debería tener fácil acceso a excelentes cuidados de la salud. Nuestro objetivo es simplificar el procedimiento posible para los pacientes y ofrecer el servicio sin importar donde se encuentren, visita nuestro repositorio en github.

[Github](#)



Anexo 5 Visitar repositorio de MODEL-DX y los experimentos

Fuente: Esta investigación

gian097 / ModelDX Public

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags

Go to file Add file Code

gian097 Update app.py	fedd660 2 days ago	2 commits
models	Add files via upload	2 weeks ago
templates	Add files via upload	2 weeks ago
uploads	Add files via upload	2 weeks ago
app.py	Update app.py	2 days ago

Help people interested in this repository understand your project by adding a README. [Add a README](#)

About

Aplicativo web para diagnosticar TDAH con deep learning, desarrollado con flask.

0 stars
1 watching
0 forks

Releases

No releases published
[Create a new release](#)

Packages

No packages published

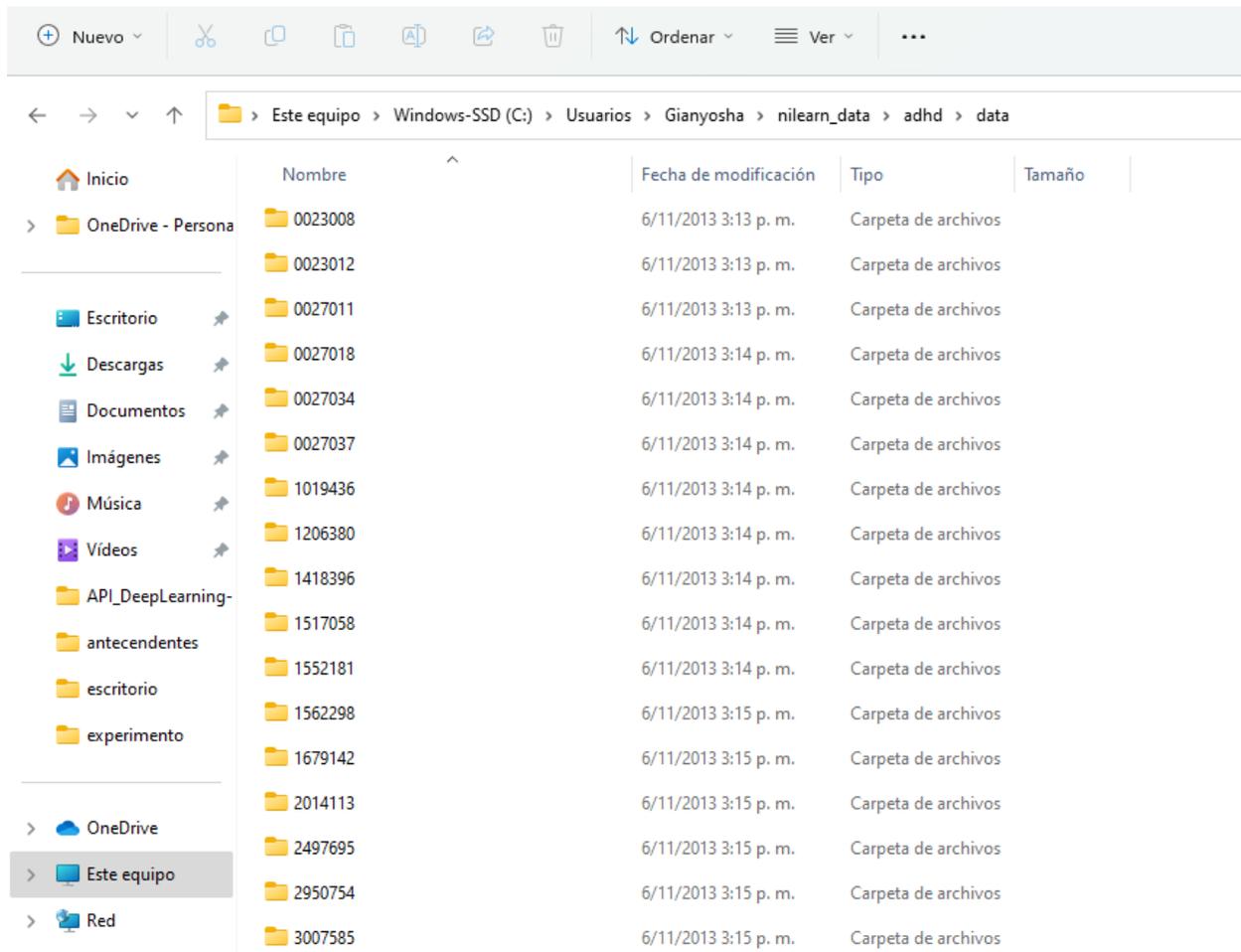
Anexo 6 Vista del botón Github

Fuente: Esta investigación

#	Nombre	Apellido	Genero	Fecha de nacimiento	Cedula	Direccion	Ciudad	RH	Imagen	Prediccion	
1	carlos	perez	Masculino	1996-05-07	1087234755	barrio modelo	tumaco	A+	./uploads\0023008_rest_tshift_RPI_voreg_mni.nii.gz	"[0.35 0.65]"	validar DX
2	pedro	perez	Masculino	1996-05-07	1087234755	barrio modelo	tumaco	A+	./uploads\0027037_rest_tshift_RPI_voreg_mni.nii.gz	"[0.01 0.99]"	validar DX

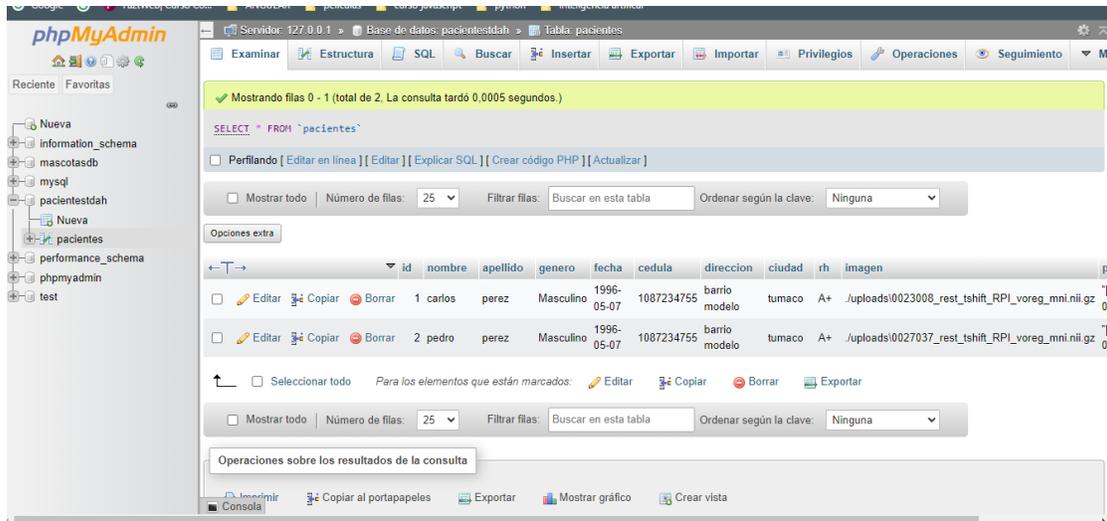
Anexo 7 Vista del apartado VALIDAR

Fuente: Esta investigación



Anexo 8 Base de datos ADHD-200 descargado de manera local

Fuente: Esta investigación



Anexo 9 Base de datos de pacientes validadas por criterio medico

Fuente: Esta investigación.

B. ANEXO II REPOSITORIO DE EXPERIMENTO PARA SELECCIÓN DEL MODELO DE MACHINE LEARNING PARA LA INVESTIGACION

A continuación, se presenta un ejemplo de experimento de un modelo de machine learning. Si desea observar todos los ejemplos de experimentos realizados en esta investigación dirigirse al siguiente enlace

```

modelo cargado standar scaler

In [123]:
import pandas as pd
from sklearn.pipeline import Pipeline
from sklearn.model_selection import GridSearchCV
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.preprocessing import StandardScaler
from keras.wrappers.scikit_learn import KerasClassifier
from keras.layers import Dense, Input, Dropout, GRU
from keras import Sequential

In [174]:
pipe = Pipeline(['svc', SVC()])
param_grid = [
    {'svc__C': [0.001, 0.01, 0.1, 1, 10, 100, 1000], 'svc__kernel': ['linear']},
    {'svc__C': [0.001, 0.01, 0.1, 1, 10, 100, 1000], 'svc__gamma': [0.1, 0.01, 0.001, 0.0001], 'svc__kernel': ['rbf']},
]
clf = RandomizedSearchCV(pipe, param_grid, random_state=2022)
clf.fit(X_train, y_train)

Out[174]:
RandomizedSearchCV(estimator=Pipeline(steps=[('svc', SVC())]),
  param_distributions=[{'svc__C': [0.001, 0.01, 0.1, 1, 10,
  100, 1000],
  'svc__kernel': ['linear']},
  {'svc__C': [0.001, 0.01, 0.1, 1, 10,
  100, 1000],
  'svc__gamma': [0.1, 0.01, 0.001,
  0.0001],
  'svc__kernel': ['rbf']}],
  random_state=2022)

```

Anexo 10 Experimento sin pipeline

Fuente: Esta investigación enlace <https://github.com/gian097/ExperimentoTesis->

```

In [175-] clf.best_params_

Out[175-] {'svc__kernel': 'linear', 'svc__C': 1}

In [178-] pipe_best1 = Pipeline([['svc', SVC(kernel='linear',C=1)]]
pipe_best1.fit(X_train,y_train)
pipe_best1.score(X_test,y_test)

```

Anexo 11 continuación de experimento, mejores hiperparametros

Fuente: Esta investigación

```

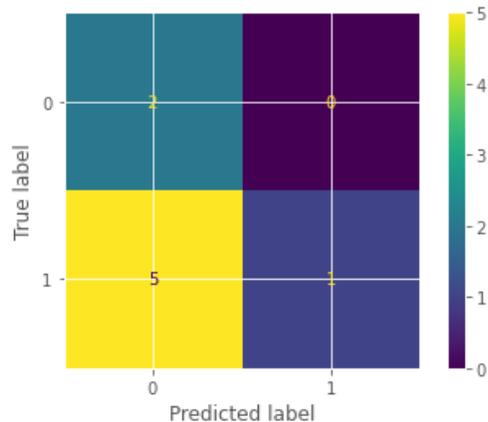
In [179-] from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score,precision_score
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import accuracy_score,precision_score
from sklearn.metrics import accuracy_score
from sklearn.metrics import confusion_matrix, ConfusionMatrixDisplay
from sklearn.datasets import make_classification

Y_pred=pipe_best1.predict(X_test)

cm = confusion_matrix(y_test, Y_pred, labels=clf.classes_)
disp = ConfusionMatrixDisplay(confusion_matrix=cm,
                             display_labels=clf.classes_)

disp.plot()
plt.show()
print("accuracy test:",accuracy_score(y_test, Y_pred))

```



accuracy test: 0.375

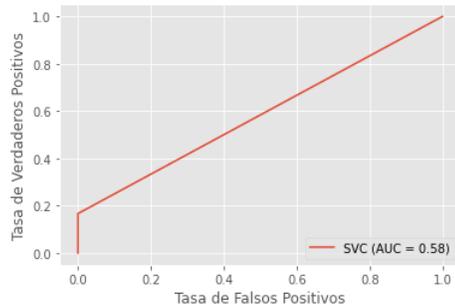
Anexo 12 Continuación del experimento matriz de confusión

Fuente: Esta investigación

In [180..

```
import matplotlib.pyplot as plt
from sklearn import datasets, metrics, model_selection, svm

Y_pred=pipe_best1.predict(X_test)
fpr, tpr, thresholds = metrics.roc_curve(y_test, Y_pred)
roc_auc = metrics.auc(fpr, tpr)
display = metrics.RocCurveDisplay(fpr=fpr, tpr=tpr, roc_auc=roc_auc, estimator_name='SVC')
display.plot()
plt.xlabel('Tasa de Falsos Positivos')
plt.ylabel('Tasa de Verdaderos Positivos')
plt.show()
```



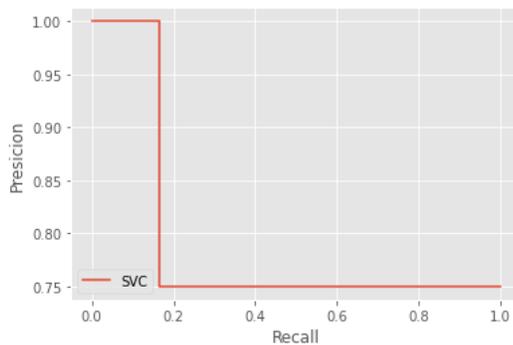
Anexo 13 Continuación del experimento, curva de roc

Fuente: Esta investigación

In [129..

```
import matplotlib.pyplot as plt
from sklearn.datasets import make_classification
from sklearn.metrics import (precision_recall_curve, PrecisionRecallDisplay)
from sklearn.model_selection import train_test_split
from sklearn.svm import SVC

predictions = pipe_best1.predict(X_test)
precision, recall, _ = precision_recall_curve(y_test, predictions)
disp = PrecisionRecallDisplay(precision=precision, recall=recall, estimator_name='SVC')
disp.plot()
plt.xlabel('Recall')
plt.ylabel('Presicion')
plt.show()
```

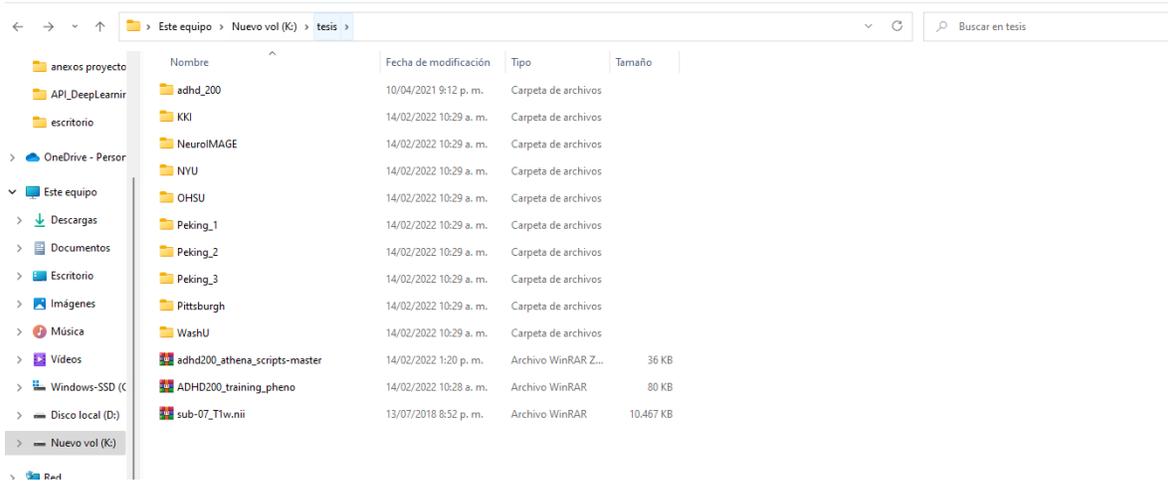


Anexo 14 Continuación del experimento grafica de recall

Fuente: esta investigación.

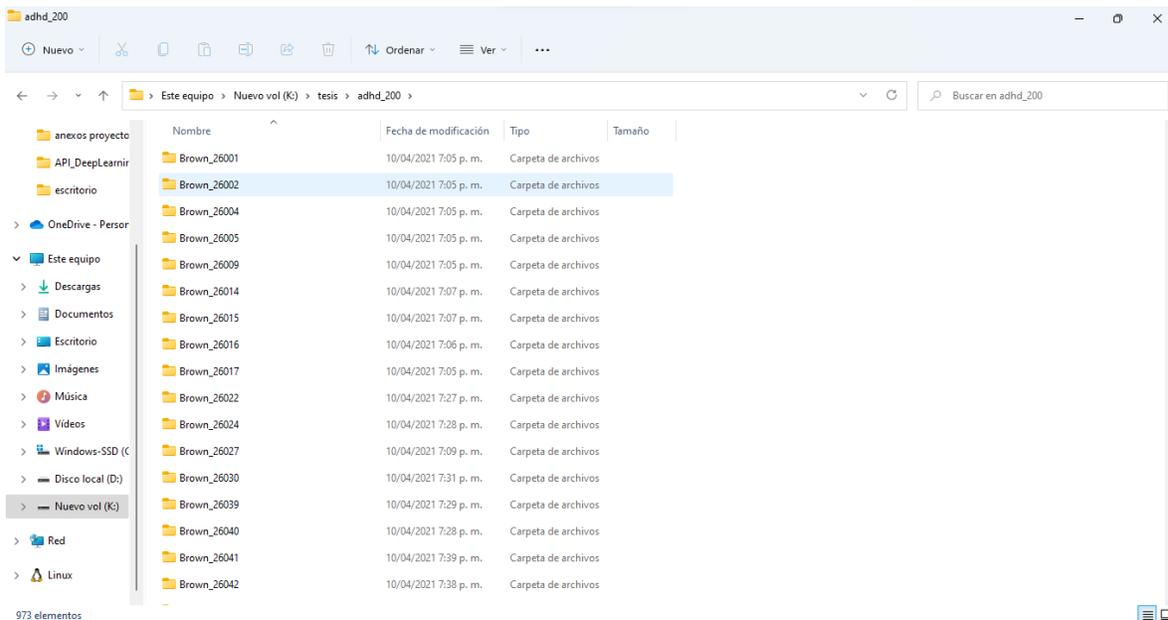
C. Conjunto de datos adhd-200 proporcionado por NITCR

A continuación, se adjuntan imágenes de conjunto de datos proporcionado por NITCR, el cual es el que se tomó para preprocesar y poder utilizarlo en esta investigación



Anexo 15 Vista de conjunto de datos descargado y almacenado de forma local

Fuente: Esta investigación.

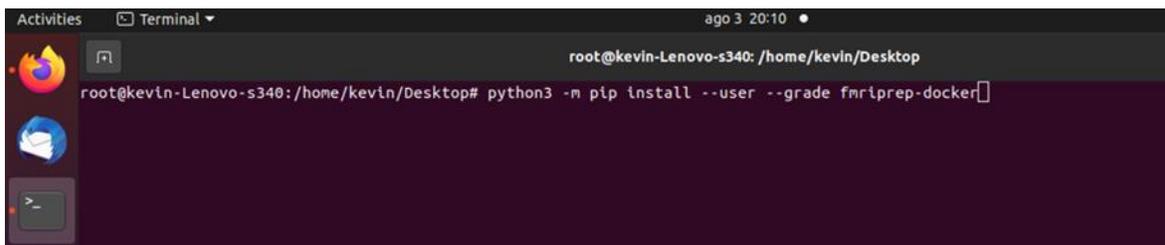


Anexo 16 Conjunto de dato con todas particiones

Fuente: Esta investigación

D. Pasos para instalar fmriprep para el preprocesamiento de neuroimágenes.

A continuación, se presentara los pasos que se requieren para la instalación de la herramienta para el preprocesamiento de las neuroimágenes.

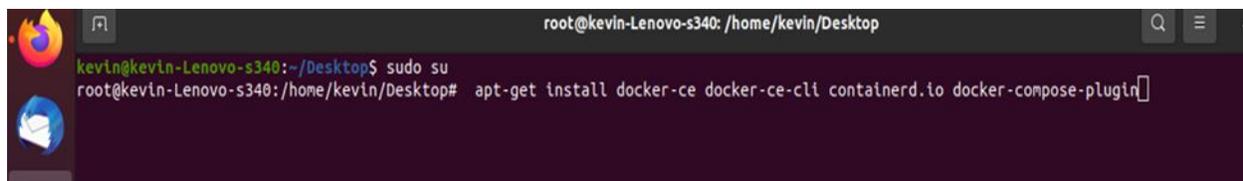


```
root@kevin-Lenovo-s340: /home/kevin/Desktop
root@kevin-Lenovo-s340:/home/kevin/Desktop# python3 -m pip install --user --grade fmriprep-docker
```

Anexo 17 Comando para instalar la imagen de FmriPrep en Docker

Fuente: Esta investigación

Una vez instalado Docker se instaló la herramienta FmriPrep en Linux en la distribución de Ubuntu en su versión 20.4 y también instalar Python (superior a la versión 3.7) para esta investigación véase en la figura 9.

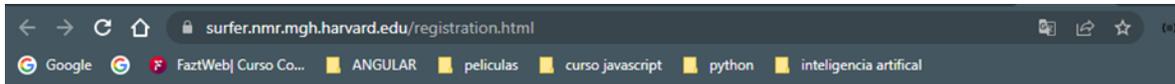


```
root@kevin-Lenovo-s340: /home/kevin/Desktop
kevin@kevin-Lenovo-s340:~/Desktop$ sudo su
root@kevin-Lenovo-s340:/home/kevin/Desktop# apt-get install docker-ce docker-ce-cli containerd.io docker-compose-plugin
```

Anexo 18 instalación de FmriPrep en Ubuntu

Fuente: Esta investigación

El siguiente paso después de instalar FmriPrep se solicitó una licencia a FreeSurfer el cual es una de las herramientas para imagen de código abierto que sirve para procesar, analizar y visualizar imágenes del cerebro humano como se puede ver en la siguiente imagen.



FreeSurfer Download and Registration

To download your copy of *FreeSurfer*, please go to:
<https://surfer.nmr.mgh.harvard.edu/fswiki/DownloadAndInstall>

While a registration key is not required to download and install *FreeSurfer*, a license.txt key file is necessary to make the software fully operational.

Please provide the following information to register your copy of *FreeSurfer* and obtain your license.txt key file:

Your name:

Title:
First:
Last:

Name of your academic/research institution:

Institution:

Type of institution which will use this software:

Non-profit education/research
 Medical
 Commercial

Your e-mail address:

e-mail:

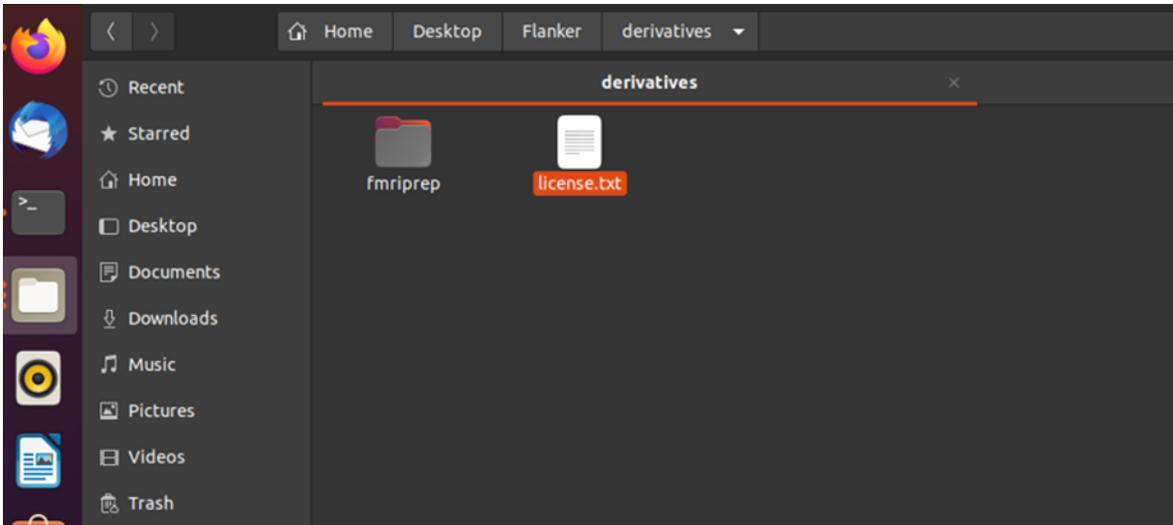
Your mailing address:

Address1:
Address2:
City:

Anexo 19 Registro para adquirir licencia para el uso de FreeSurfer

Fuente: FreeSurfer enlace <https://surfer.nmr.mgh.harvard.edu/registration.html>

Una vez obtenida la licencia de FreeSurfer se ubicó en el directorio previamente creado para el preprocesamiento que está en la siguiente ruta “Home/Desktop/Flaker/Derivatives” para proceder la ejecución del software FmriPrep y poder empezar el preprocesamiento de las imagen.



Anexo 20 Ubicación de licencia proporcionada por FreeSurfer

Fuente: Esta investigación

Para la ejecución de software Fmriprep se efectuó el siguiente comando el cual contiene la siguiente estructura



Anexo 21 Comando para ejecutar FmriPrep

Fuente: Esta investigación

Consecuentemente con la ejecución del comando en el enunciado en la Figura 8, se presenta la siguiente tabla con la descripción de cada parte del comando para el preprocesamiento de una imagen del conjunto de datos ADHD 200.

Comando	Descripción
---------	-------------

/home/Desktop/Flaker/Derivative/license.txt	lo siguiente es la ruta a una carpeta de base de datos PyBIDS existente, para una indexación más rápida (especialmente útil para grandes conjuntos de datos),
/home/kevin/Desktop/Flaker/fmriprep/out	es la ruta de salida para los resultados del preprocesamiento y los informes visuales
participant –participant-label 218	una lista delimitada por espacios de identificadores de participantes o un único identificador (el subprefijo se puede eliminar),
--skip-bids-validation	suponga que el conjunto de datos de entrada es compatible con BIDS y omita la validación
--boilerplate_only	omitir la generación de citas formateadas en HTML y LaTeX
-v, --verbose	aumenta la verbosidad del registro para cada ocurrencia, el nivel de depuración es -vvv
--fs-no-reconall	deshabilite el preprocesamiento de superficie de FreeSurfer.
-nthreads	número máximo de subprocesos por proceso
--stop-on-first-crash	Forzar la detención en el primer bloqueo, incluso si se especificó un directorio de trabajo.
--mem_mb	límite superior de memoria para procesos fMRIPrep
--output-spaces	Espacios estándar y no estándar para remuestrear imágenes anatómicas y funcionales. Los espacios estándar se pueden especificar mediante el formulario donde es una palabra clave (palabras clave válidas: “MNI152Lin”, “MNI152NLin2009cAsym”, “MNI152NLin6Asym”, “MNI152NLin6Sym”, “MNIInfant”, “MNIPediatricAsym”, “NKI”, “OASIS30ANTS”, “PNC”, “UNCInfant”, “fsLR”, “fsaverage”, “fsaverage5”, “fsaverage6”) o la ruta que apunta a una plantilla proporcionada por el usuario, y puede ir seguida de parámetros opcionales separados por dos puntos. Los espacios no estándar (palabras clave válidas: anat, T1w, run, func, sbref, fsnative) implican orientaciones y cuadrículas de muestreo específicas. Es importante tener en cuenta que el res-*modificador no define la resolución utilizada para la normalización espacial.

Anexo 22 descripción de comando para el preprocesamiento de las neuroimágenes

Fuente: Fmriprep enlace <https://normas-apa.org/estructura/tablas/>