

Escuela Superior Politécnica del Litoral

Facultad de Ingeniería en Electricidad y Computación

Análisis de alteraciones en los movimientos oculares en enfermos de Parkinson,
utilizando técnicas de Inteligencia Artificial.

TECH-326

Proyecto Integrador

Previo a la obtención del Título de:

Ingeniero en Ciencias de la computación

Presentador por:

Milca Elizabeth Valdez Flores

Andrea Adelina Mero Plaza

Guayaquil - Ecuador

Año: 2023

Dedicatoria

A Dios y a todas las personas que enfrentan la lucha diaria contra el Parkinson, que cada día demuestran una fuerza y perseverancia inigualables. Este proyecto está dedicado a ustedes, con la esperanza de que este aporte sea una luz en la búsqueda de respuestas y soluciones. A sus familias y seres queridos, cuyo amor y apoyo son el verdadero motor que impulsa la investigación y el deseo de comprender mejor esta enfermedad. Que este esfuerzo sea un paso más hacia un futuro donde el Parkinson sea solo una palabra y no una batalla diaria.

A nuestras familias y amigos quienes siempre nos han apoyado.

Andrea Mero, Milca Valdez

Agradecimiento

Agradecemos a ESPOL, por darnos la oportunidad de ingresar al área de la investigación.

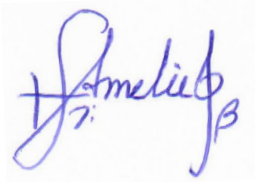
Andrea Mero, Milca Valdez

Declaración Expresa

Nosotras, Andrea Mero y Milca Valdez reconocemos que:

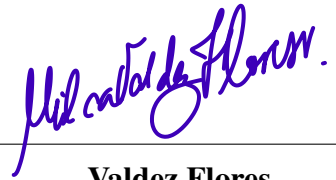
La titularidad de los derechos patrimoniales de autor (derechos de autor) del proyecto de graduación corresponderá al autor o autores, sin perjuicio de lo cual la ESPOL recibe en este acto una licencia gratuita de plazo indefinido para el uso no comercial y comercial de la obra con facultad de sublicenciar, incluyendo la autorización para su divulgación, así como para la creación y uso de obras derivadas. En el caso de usos comerciales se respetará el porcentaje de participación en beneficios que corresponda a favor del autor o autores. La titularidad total y exclusiva sobre los derechos patrimoniales de patente de invención, modelo de utilidad, diseño industrial, secreto industrial, software o información no divulgada que corresponda o pueda corresponder respecto de cualquier investigación, desarrollo tecnológico o invención realizada por nosotros durante el desarrollo del proyecto de graduación, pertenecerán de forma total, exclusiva e indivisible a la ESPOL, sin perjuicio del porcentaje que nos corresponda de los beneficios económicos que la ESPOL reciba por la explotación de nuestra innovación, de ser el caso. En los casos donde la Oficina de Transferencia de Resultados de Investigación (OTRI) de la ESPOL comunique los autor/es que existe una innovación potencialmente patentable sobre los resultados del proyecto de graduación, no se realizará publicación o divulgación alguna, sin la autorización expresa y previa de la ESPOL.

Guayaquil, 03 de febrero del 2024.

A handwritten signature in blue ink, appearing to read 'Adelina Mero Plaza'.

Mero Plaza Andrea

Adelina

A handwritten signature in blue ink, appearing to read 'Milca Valdez Flores'.

Valdez Flores

Milca Valdez

Evaluadores

Luis Eduardo Mendoza M.,

Ph.D

Profesor de la materia

Enrique Peláez Ph. D

Tutor del proyecto

Resumen

La Enfermedad de Parkinson es un trastorno neurodegenerativo progresivo no solo afecta a la postura del paciente, si no también al movimiento ocular. Estas anomalías son más evidentes en las sacadas voluntarias. Este estudio busca determinar si existen o no alteraciones en el movimiento ocular entre pacientes de Parkinson y controles sanos, mediante el uso de técnicas de Inteligencia Artificial, así como un prototipo funcional, Parkinson Ocular, que utiliza el mejor modelo para generar un diagnóstico. En el desarrollo del prototipo se contemplaron las siguientes fases: recolección de datos, preprocesamiento, modelos de Machine Learning y Deep Learning y la generación de la interfaz. El modelo final fue el modelo de Random Forest, el cual obtuvo una precisión del 87%, siendo el mejor modelo con relación a los demás modelos analizados. También, el prototipo se desarrolló para dos tipos de roles: Médico, y Administrador. El prototipo provee la probabilidad de que el paciente tenga o no la enfermedad, así como la visualización del movimiento ocular. El prototipo funcional Parkinson Ocular, aprovecha el mejor modelo para la detección temprana de la Enfermedad de Parkinson y es una herramienta útil para los médicos en el análisis, estudio y diagnósticos oculares de los pacientes.

Palabras claves: Machine Learning, Deep Learning, Redes Neuronales, Random Forest, Parkinson

Abstract

Parkinson's Disease is a progressive neurodegenerative disorder that affects not only the patient's posture, but also eye movement. These anomalies are more evident in voluntary saccades. This study seeks to determine whether there are alterations in eye movement between Parkinson's patients and healthy controls, through the use of Artificial Intelligence techniques, as well as a functional prototype, Parkinson's Ocular, which uses the best model to generate a diagnosis. In the development of the prototype, the following phases were considered: data collection, preprocessing, Machine and Deep Learning models, and interface generation. The final model was the Random Forest model, which obtained an accuracy percentage of 87%, being the best model in relation to the other models analyzed. Also, the prototype was developed for two types of roles: Doctor, and Administrator. The prototype provides the probability that the patient has or does not have the disease, as well as the visualization of eye movement. The functional Parkinson Ocular prototype takes advantage of the best model for the early detection of Parkinson's disease and is a useful tool for doctors in the analysis, study, and ocular diagnoses of patients

Key words: Machine Learning, Deep Learning, Neuronal Nets, Parkinson

Índice general

Resumen	i
Abstract	ii
Índice general	iii
Abreviaturas	v
Índice de figuras	vi
Índice de tablas	viii
Capítulo 1	1
1 Introducción	2
1.1 Descripción del problema	3
1.2 Justificación del problema	4
1.3 Objetivos	4
1.3.1 Objetivos generales	4
1.3.2 Objetivos específicos	4
1.4 Marco teórico	5
Capítulo 2	7
2 Metodología	8
2.1 Recolección de Datos	8
2.1.1 Dataset	9
2.2 Preprocesamiento	9
2.2.1 Herramienta de preprocesamiento: Pymovements	9
2.2.2 Selección de características	10
2.3 Modelos de aprendizaje	11
2.3.1 Modelos de Machine Learning	11
2.3.2 Modelos de Deep Learning	13

2.4	Experimento	16
2.5	Métricas de desempeño	17
2.6	Prototipo funcional	18
2.6.1	Requerimientos funcionales	18
2.6.2	Requerimientos no funcionales	18
2.7	Diagramas del sistema	19
2.8	Diseño inicial del prototipo	19
2.9	Elección de tecnologías	20
Capítulo 3		21
3	Resultados y Análisis	22
3.1	Recolección de datos	22
3.2	Preprocesamiento	22
3.2.1	Normalización	22
3.2.2	Extracción de nuevas características	23
3.2.3	Synthetic Data Augmentation	23
3.3	Modelos de Machine Learning	24
3.3.1	Resultados de los experimentos	25
3.4	Modelos de Deep Learning	29
3.4.1	Resultados de los experimentos	30
3.5	Prototipo funcional	33
3.5.1	Rol de Administrador	34
3.5.2	Rol del Médico	37
Capítulo 4		40
4	Conclusiones y Recomendaciones	41
4.1	Conclusiones	41
4.2	Recomendaciones	42

Abreviaturas

EP	Enfermedad del Parkinson
GB	Ganglios Basales
CNN	Redes Neuronales Convolucionales
ML	Machine Learning
DL	Deep Learning
SN	Sustancia Negra
SNpc	Pars compacta de la Sustancia Negra
LPD	Hemicuerpo izquierdo
RPD	Hemicuerpo derecho
HC	Control Sano
RF	Random Forest
TL	Transfer Learning
CEDIA	Corporación Ecuatoriana para el Desarrollo de la Investigación y la Academia.

Índice de figuras

Figura 1: Representación esquemática de la circuitería básica de las sacadas	3
Figura 2: Fases de Metodología Aplicada	8
Figura 3: Variabilidad de importancia de características oculares según RF	10
Figura 4: Ejemplo del modelo Decision Tree	11
Figura 5: Ejemplo del Random Forest	12
Figura 6: Modelo de Regresión Logística	12
Figura 7: Arquitectura VGG-16	13
Figura 8: Arquitectura de la VGG-19	14
Figura 9: Arquitectura ResNet-50	15
Figura 10: Arquitectura de InceptionV3	15
Figura 11: Arquitectura del MobileNet	16
Figura 12: Mapa de calor de la mirada de paciente	16
Figura 13: Trayectoria del movimiento ocular de paciente	17
Figura 14: Diagrama de casos de uso	19
Figura 15: Datos de Entrenamiento SIN SMOTE	23
Figura 16: Datos de Entrenamiento CON SMOTE	24
Figura 17: Matriz de Confusión del Sub-exp.3	26
Figura 18: Matriz de Confusión del Sub-exp.4	27
Figura 19: Estructura del Árbol de Decisión del Sub-exp. 4	28
Figura 20: VGG-16 – Mapa de calor precisión y error	31
Figura 21: Evaluación del modelo VGG-16 en mapa de calor	31
Figura 22: VGG-16 – Trayectoria precisión y error	32
Figura 23: Evaluación del modelo VGG-16 en trayectoria	32
Figura 24: Pantalla de inicio de sesión	33
Figura 25: Pantalla de creación del Médico	34
Figura 26: Médicos del sistema	35
Figura 27: Archivos del Médico	36

Figura 28: Descargar Archivos del Médico	36
Figura 29: Subir Archivos	37
Figura 30: Archivos Seleccionados	37
Figura 31: Diagnóstico de Parkinson	38
Figura 32: Lista de archivos del Médico	39

Índice de tablas

Tabla 1: Reporte Métricas de Clasificación Sub-exp 1, Sub-exp 2	25
Tabla 2: Reporte Métricas de Clasificación Sub-exp 3, Sub-exp 4	26
Tabla 3: Otras Métricas de Calidad	28
Tabla 4: Resultados de experimentos en imágenes de mapa de calor	30
Tabla 5: Resultados de experimentos en imágenes de trayectoria	30

Capítulo 1

1. Introducción

La Enfermedad de Parkinson (EP) es un trastorno neurodegenerativo progresivo. Las manifestaciones clínicas clásicas son: el temblor en reposo, rigidez muscular, acinesia (o bradicinesia) e inestabilidad postural [1] [2]. Entre las características típicas de la EP se incluyen la postura flexionada y la congelación de la marcha. También, son comunes los síntomas no motores como deterioro cognitivo, apatía, depresión, anosmia, disautonomía y trastornos del sueño [3]. Las manifestaciones motoras de la EP son causadas por la pérdida de células dopaminérgicas en la Pars compacta de la Sustancia Negra (SNpc), lo que resulta en disfunción de los ganglios basales (BG) [3]

Los ganglios basales son núcleos subcorticales de la base del cerebro anterior y están conectados directa e indirectamente con las estructuras corticales y subcorticales [4]. Los BG promueven la iniciación del movimiento dirigido a metas al eliminar la inhibición sostenida del movimiento deseado y suprimir los movimientos no deseados [5]. A pesar de los avances considerables en la comprensión de la anatomía y función de los BG, su papel complejo en la modulación del comportamiento motor aún está lejos de estar completamente dilucidado [5]. El sistema sacádico ofrece ventajas únicas para estudiar los BG porque los circuitos neuronales subyacentes son relativamente bien comprendidos y sus bucles corticobasales funcionales son probablemente similares a aquellos involucrados en la regulación de otros movimientos [5]. Además, las sacadas pueden medirse fácil y precisamente [3].

Las sacadas son movimientos oculares rápidos necesarios para llevar objetos de interés a la fovea [6]. Las anomalías en los movimientos oculares son comunes en la EP: las anomalías son más evidentes en las sacadas voluntarias que en las sacadas reflejadas en las etapas iniciales, pero las sacadas guiadas visualmente también pueden estar involucradas en etapas posteriores [3]. La hipometría sacádica (que incluye sacadas anormalmente fragmentadas), la reducción de la precisión y el aumento de la latencia se encuentran entre los déficits más prominentes [3]

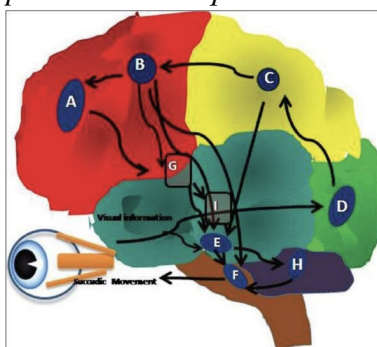
1.1. Descripción del problema

A diferencia de los movimientos lentos de las extremidades, las personas con EP generan movimientos oculares automáticos rápidos en respuesta a estímulos sensoriales y muestran una capacidad deteriorada para generar movimientos oculares voluntarios en tareas cognitivas [7] [8]. La medición de las sacadas se puede llevar a cabo de manera sencilla y precisa, y existe un considerable entendimiento de la circuitería neural que controla la planificación y ejecución de los movimientos oculares sacádicos [7] [8] [9] .

Aunque no se observa físicamente una relación evidente entre los síntomas de la enfermedad y las alteraciones visuales, hay pocos estudios que describen efectos visuales específicos producidos por la EP [6]. El estudio de los movimientos sacádicos oculares en condiciones clínicas, como la EP, se ha vuelto bastante popular en tiempos recientes debido a su potencial para proporcionar información sobre el comportamiento y la alteración de la circuitería básica (ver Figura 4) [6]. De hecho, en muchos estudios de neuroimagen, se observan alteraciones en el consumo de oxígeno en áreas visuales del cerebro (regiones 4 occipitales) en comparación con controles sanos [6]. Sin embargo, la explicación de estas alteraciones observadas se atribuye más a un mecanismo de compensación de la parte motora que a un efecto intrínseco de la enfermedad [6]

Figura 1

Representación esquemática de la circuitería básica de las sacadas



Nota: A: Corteza Prefrontal Dorsolateral (DLPFC); B: Campo Visual Frontal (FEF); C: Campo Visual Parietal (PEF); D: Corteza Visual; E: Colículo Superior (SC); F: Generadores de sacadas en el tronco cerebral; G: Ganglios Basales (BG); H: Cerebelo; I: Tálamo [10]

Con base a lo mencionado, se propone la implementación de un mecanismo para la

detección y diagnóstico temprano de la EP en pacientes; por medio del desarrollo de un algoritmo inteligente que analice el movimiento ocular del paciente, obtenido mediante el dispositivo EyeTracker. Este dispositivo recoge la trayectoria de los movimientos oculares en forma de coordenadas (x, y), así como el diámetro de la pupila en cada movimiento. Este enfoque busca contribuir significativamente al avance del conocimiento actual en este campo.

1.2. Justificación del problema

La EP es una afección degenerativa que consiste en la reducción progresiva de las neuronas dopaminérgicas que se manifiesta principalmente en alteraciones de los movimientos corporales, tales como: temblor en reposo, lentitud de movimientos, alteraciones de postura o rigidez [1]. Según datos del año 2019 de la Organización Mundial de la Salud (WHO, por sus siglas en inglés) [10], esta enfermedad ha afectado a 8.5 millones de personas a nivel global y aumentó en más del 100% a partir del año 2000. Además [11] menciona que debido a la incapacidad motora, se ha comprometido la calidad de vida del paciente y sus cuidadores o familiares, quienes también ocupan muchas horas del día para dedicarse a su cuidado [10].

Cabe destacar que en los últimos años se han producido diversos avances en la comprensión y el tratamiento de la EP. Sin embargo, persisten importantes áreas poco exploradas, por ejemplo, en identificación de métodos de diagnóstico precoz.

1.3. Objetivos

1.3.1. Objetivos generales

Implementar un prototipo que permita determinar si existen o no alteraciones en el movimiento ocular entre pacientes de Parkinson y controles sanos, mediante el uso de técnicas de Inteligencia Artificial.

1.3.2. Objetivos específicos

Se establecieron los siguientes objetivos específicos:

- Preprocesar los datos obtenidos de la reconstrucción de registros digitales generados mediante un EyeTracker en sujetos sanos y pacientes con la EP y extraer características

clínicas que se utilizarán para entrenar modelos de clasificación usando técnicas de IA.

- Reconstruir los datos en imágenes que representan el movimiento de la pupila, para analizarlos empleando técnicas de Deep Learning (DL).
- Diseñar la arquitectura de los modelos de aprendizaje autónomo poco profundo y redes neuronales de tipo profundo.
- Evaluar los resultados a través de métricas de medición del desempeño de los modelos.
- Implementar un prototipo basado en una aplicación web con el mejor modelo de aprendizaje y presentar los resultados de la clasificación de las características de los movimientos oculares.

1.4. Marco teórico

El movimiento ocular se puede medir mediante rastreadores oculares, y estos datos se pueden utilizar para analizar las características del movimiento ocular en la EP. El análisis de los movimientos oculares ofrece la posibilidad de realizar un diagnóstico y, por lo tanto, puede ayudar a comprender mejor la EP [12].

Los datos de salida (sacadas, diámetro de la pupila) proporcionan una señal proxy de alta fidelidad de la integridad subyacente de los circuitos cerebrales [13].

Los datos adquiridos en los rastreadores en el seguimiento ocular son las coordenadas “x” y “y”, así como el tamaño de la pupila. Las características básicas hechas a mano pueden extraer información valiosa sobre el comportamiento temporal de la funcionalidad motora ocular y han sido el enfoque en el pasado [14].

En [12], los autores propusieron métodos de extracción de características en los datos y posteriormente las analizaron, puesto que las características extraídas permitieron utilizar métodos de agrupación, tales como K-means difusos para crear grupos y luego exploraron subgrupos dentro de los primeros grupos. Con el fin de extraer las características, los autores propusieron usar redes profundas, como Stack Autoencoders y Redes Neuronales Convolucionales (CNN por sus siglas en inglés).

En [15], los autores definieron un Autoencoder como una variedad particular de Red Neuronal, cuya principal función fue transformar la entrada en una representación comprimida y significativa para luego revertirla, de manera que la entrada reconstruida se asemeje lo más posible a la original. Normalmente, los Autoencoders se han empleado con el objetivo de reducir la dimensionalidad y adquisición de características. Los Autoencoders de tipo subcompletos son valiosos para obtener representaciones más concisas de los datos de entrada, lo que resulta beneficioso para la compresión de datos y la obtención de representaciones más idóneas. Por otro lado, los Autoencoders sobredimensionados buscan representaciones más amplias de los datos de entrada, lo cual puede ser ventajoso cuando los datos son densos, ya que puede dispersarlos en dimensiones superiores, lo que es útil para las redes que no manejan adecuadamente conjuntos de datos densos. Sin embargo, un inconveniente de los Autoencoders sobredimensionados es la posibilidad de que simplemente reproduzcan la entrada y asignen 0 a todos los demás elementos, lo cual no es el resultado deseado [12].

Para evitar este comportamiento de replicación, se puede aplicar un método de regularización que guíe al Autoencoder a cumplir con nuestros objetivos y no simplemente duplicar la entrada. I. Goodfellow et al. [16] concluyen que las redes CNN han demostrado ser altamente efectivas en aplicaciones prácticas. Estas redes se distinguen por el uso de una operación matemática llamada Convolución en al menos una de sus capas, en lugar de la multiplicación de matrices generalizadas y la operación de Pooling. Estas redes destacan como ejemplos de la influencia de principios neurocientíficos en el DL, y se exploran estos principios en el contexto de las CNN.

Los métodos de agrupación que los autores en [12] mencionan son clústeres, los cuales se emplean para reunir datos similares en grupos, asegurando que los elementos cercanos pertenezcan al mismo grupo, y disimiles con los elementos en otros grupos. Esto puede servir tanto para validar la agrupación de datos conocidos como para descubrir nuevas categorías que representen subgrupos dentro de las clases existentes. Además, se analizan las propiedades de los clústeres utilizando diferentes enfoques, como el coeficiente de Silhouette, el índice de Dunn y el coeficiente de partición difusa [17].

Capítulo 2

2. Metodología

La metodología empleada fue:

- Recopilación de Datos
- Preprocesamiento de Datos
- Reconstrucción de Datos
- Diseño de la arquitectura del modelo de aprendizaje autónomo
- Evaluación de Resultados
- Implementación de Prototipo

El seguimiento de estas fases facilitó el tratamiento adecuado de los datos, permitiendo la selección de características relevantes para entrenar la red neuronal, validar resultados y cargar el modelo en una interfaz web para el usuario final (ver Figura 2).

Figura 2

Fases de Metodología Aplicada



Nota: El diagrama muestra la secuencia del tratamiento de los datos.

2.1. Recolección de Datos

La fase inicial se enfocó en reunir y organizar los datos necesarios para el desarrollo del proyecto, contando con la colaboración directa del cliente. Gracias a la experiencia del equipo y la guía del mentor, se justificó la selección de métodos apropiados para el proyecto.

2.1.1. Dataset

Los datos se obtuvieron de personas sanas (HC) y pacientes con EP en el hemisferio izquierdo (LPD) y el hemisferio derecho (RPD), durante un proceso de recolección de imágenes de Resonancia Magnética Funcional (RMF). Se recolectaron dos grupos de datos: el primero constó de 30 pacientes con EP (11 LPD, 11 RPD) y 8 sujetos de control (HC), y el segundo constó de 27 pacientes con EP (4 LPD, 10 RPD) y 13 sujetos de control (HC). El primer grupo realizó experimentos con la tarea Stop&Go, mientras que el segundo con la tarea Perceptive. La frecuencia de muestreo del Eyetracker fue de 60Hz, generando aproximadamente 2'600,000 registros en ambos experimentos con un total de alrededor de 150,000 muestras por sujeto.

2.2. Preprocesamiento

2.2.1. Herramienta de preprocesamiento: Pymovements

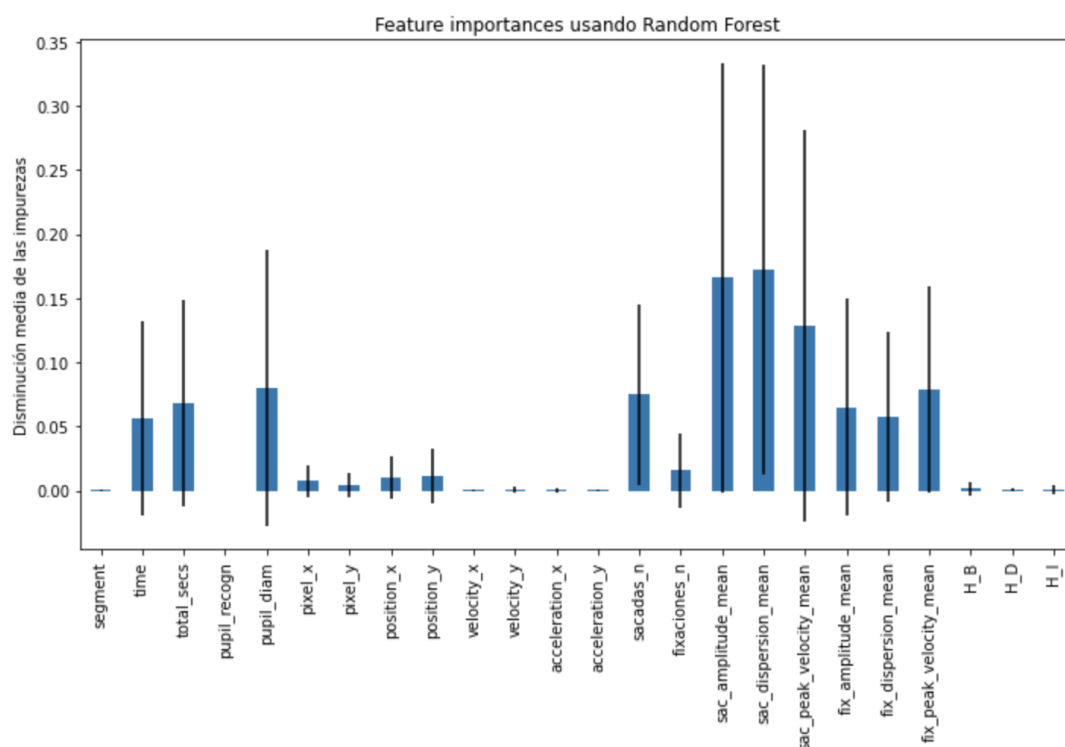
Para el procesamiento del dataset y con el fin de generar más características de las obtenidas en el experimento, se utilizó Pymovements [18], un paquete de Python diseñado para analizar datos de eye-tracking, con pruebas rigurosas y cumplimiento de estándares de codificación. Este paquete abarca procesos en toda la cadena de preprocesamiento de datos, desde el análisis de archivos de datos de seguimiento ocular hasta la visualización de resultados mediante diversos métodos gráficos. Además, Pymovements ofrece una interfaz fácilmente accesible para la descarga y procesamiento de conjuntos de datos públicos. Se destaca la importancia de las pruebas rigurosas en paquetes de software científico para garantizar la reproducibilidad y transparencia en la investigación, permitiendo que otros investigadores verifiquen y amplíen hallazgos previos [9]. Entre las características obtenidas por las librerías, tenemos la posición, velocidad y aceleración de la mirada del paciente. Otras características obtenidas están relacionadas a eventos tales como las sacadas, microsacadas, la duración de estos, así como la amplitud, dispersión y pico de velocidad.

2.2.2. Selección de características

Para el experimento Perceptive, en la selección de características se empleó la función Feature Importances de Sklearn, librería del lenguaje de programación Python, dado que nos permitió evaluar la importancia media y variabilidad de la desviación estándar de cada una de las características en todos los árboles del Random Forest (RF) al momento de realizar la clasificación artificial [19].

Figura 3

Variabilidad de importancia de caraterísticas oculares según RF



Inicialmente, se contó con 24 características oculares, sin embargo, se efectuó el entrenamiento con las 15 características más importantes que contribuían a la reducción de impurezas en los datos desde un 0,4% (ver Figura 3).

Por otro lado, para el experimento Stop&Go, las características seleccionadas en las arquitecturas son la posición, velocidad y aceleración de la mirada del paciente que forman parte clave para poder también obtener las características de las sacadas, microsacadas y la

duración de estas, así como la amplitud y pico de velocidad. Se tomaron estas métricas ya que, mediante ellas, Pymovements es capaz de generar las imágenes de los mapas de calor de la mirada del paciente y la trayectoria del movimiento ocular, con el fin de contrastar los resultados obtenidos entre los modelos de Machine Learning (ML) y DL.

A continuación, se presentan los modelos para la clasificación de la EP, empleando como lenguaje de programación Python, debido a la versatilidad de este en el área de Inteligencia Artificial.

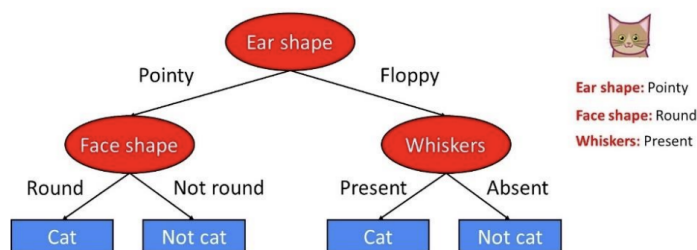
2.3. Modelos de aprendizaje

2.3.1. Modelos de Machine Learning

Decision Tree: El Decision Tree (DT) es un método de Aprendizaje Supervisado Jerárquico, se compone de nodos de decisión internos y hojas terminales [20]. Cada nodo utiliza una función de prueba que etiqueta las ramas con puntuaciones discretas [20]. El objetivo es crear un modelo que prediga el valor de una variable objetivo aprendiendo reglas de decisión simples inferidas de las características de los datos [21].

Figura 4

Ejemplo del modelo Decision Tree



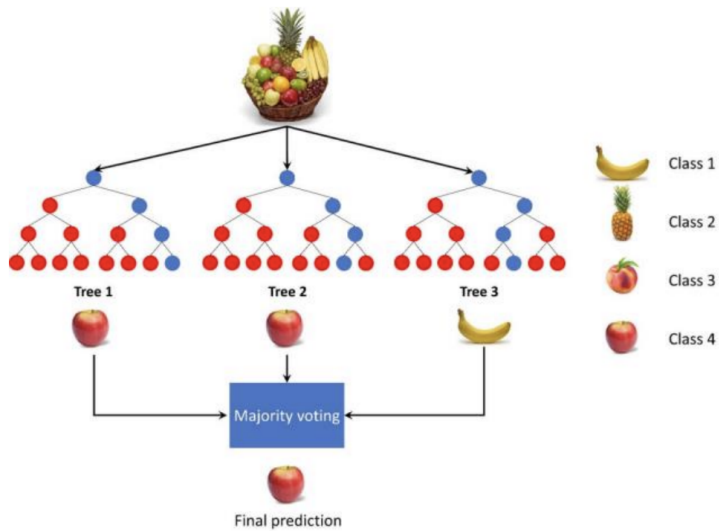
[22]

Random Forest: En [20], los autores se refieren a que el RF es una combinación de árboles de decisiones como clasificador principal y usa esta técnica para caracterizar la información. En estos clasificadores, los vectores aleatorios independientes se distribuyen de manera uniforme, haciendo que cada árbol de decisión vote individualmente para la clase más

popular. Un vector aleatorio es independiente de los vectores aleatorios anteriores con la misma distribución, y la muestra de entrenamiento se utiliza para crear un árbol.

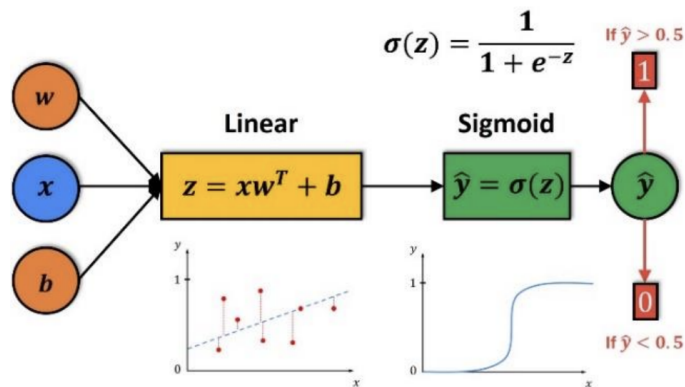
Figura 5

Ejemplo del Random Forest



[23]

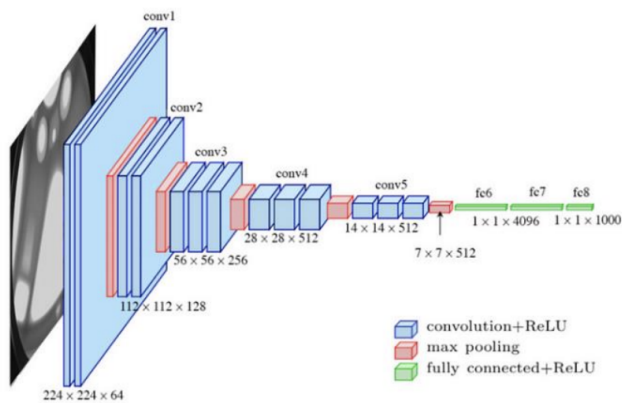
Regresión Logística: La Regresión Logística es un modelo de clasificación más que un modelo de Regresión; es un método simple y más eficiente para problemas de clasificación binaria y lineal siendo un método estadístico para la clasificación binaria que se puede generalizar a la clasificación multiclase [20]. En [24] menciona que la Regresión Logística es apropiada porque la variable dependiente es discreta y categórica. En teoría uno podría realizar regresión con valores de datos numéricos (como Regresión Lineal), pero la Regresión Logística puede manejar predictores mixtos, continuos y discretos.

Figura 6*Modelo de Regresión Logística*

[25]

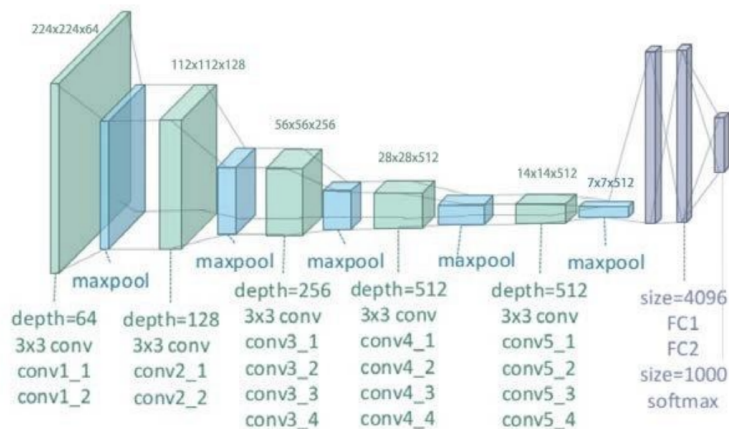
2.3.2. Modelos de Deep Learning

VGG-16: Es un modelo de red neuronal convolucional (CNN) desarrollado por N.K. Simonyan y A. Zisserman de la Universidad de Oxford, según se describe en su publicación “Very Deep Convolutional Networks for Large-Scale Image Recognition” [26]. En este estudio, mencionan que el modelo de VGG-16 alcanzó una precisión del 92,77% en el conjunto de datos de ImageNet, que contiene alrededor de 14 millones de imágenes divididas en 1000 clases. Está arquitectura comienza con imágenes de $224 \times 224 \times 3$ píxeles que pasan por dos capas convolucionales seguidas de una capa de agrupación máxima. Luego, incluye dos capas convolucionales más y otra capa de agrupación máxima. Posteriormente, se compone de tres capas convolucionales, una capa de MaxPooling, tres capas convolucionales más, otra capa de MaxPooling, seguidas de otras tres capas convolucionales y una última capa de MaxPooling. Después, se incorpora la capa ReLU y luego una capa densa. El número de filtros varía en cada capa. Las capas convolucionales utilizan un filtro de 3×3 con un paso de 1, mientras que las capas MaxPooling usan un filtro de 2×2 con un paso de 2.

Figura 7*Arquitectura VGG-16*

[27]

VGG-19: La arquitectura VGG-19 es similar a la arquitectura VGG-16 pero más extensa, pues incluye tres capas completamente conectadas, dieciséis capas convolucionales, una capa SoftMax y cinco capas MaxPool. Los filtros en las capas convolucionales varían entre 64, 128 y 256 [28].

Figura 8*Arquitectura de la VGG-19*

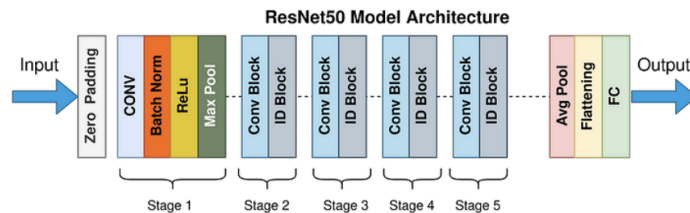
[29]

ResNet-50: ResNet-50 es otro modelo basado en CNN, compuesto por 48 capas convolucionales. Además, incluye una capa de Max Pooling y otra de Average Pooling. La

arquitectura de ResNet permite la adición de conexiones directas y el uso de funciones residuales. La conexión directa en esta arquitectura facilita la omisión de algunas de las capas del modelo [28].

Figura 9

Arquitectura ResNet-50

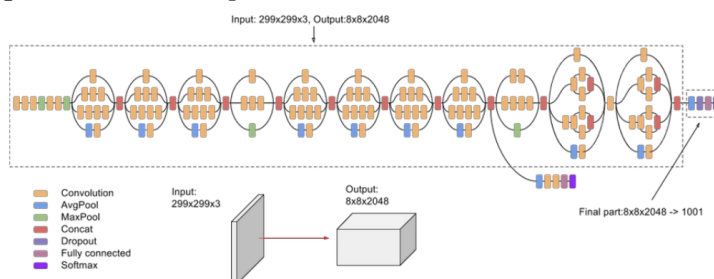


[30]

InceptionV3: La red InceptionV3 [31] es una arquitectura convolucional profunda diseñada para tareas de clasificación en ImageNet [32], un conjunto de datos que consta de 1,2 millones de imágenes RGB de 1000 clases. Dada una imagen X , la tarea de la red es generar una etiqueta de clase y , en forma de un vector de probabilidades $p(y|x) \in [0, 1]^{1000}$, que indica la probabilidad que la red asigna a cada una de las clases etiquetadas. La red InceptionV3 es una de las redes más utilizadas para el aprendizaje por transferencia y los modelos previamente entrenados están disponibles en la mayoría de las bibliotecas de software de DL [33].

Figura 10

Arquitectura de InceptionV3



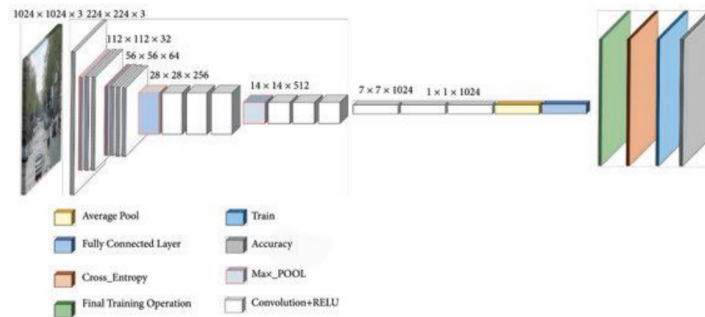
[34]

MobileNet: MobileNet es un modelo liviano con menos parámetros y menos tiempo

de cálculo. MobileNet no utiliza convoluciones estándar, sino convoluciones separables en profundidad que requieren solo una octava parte del costo de cálculo [35].

Figura 11

Arquitectura del MobileNet



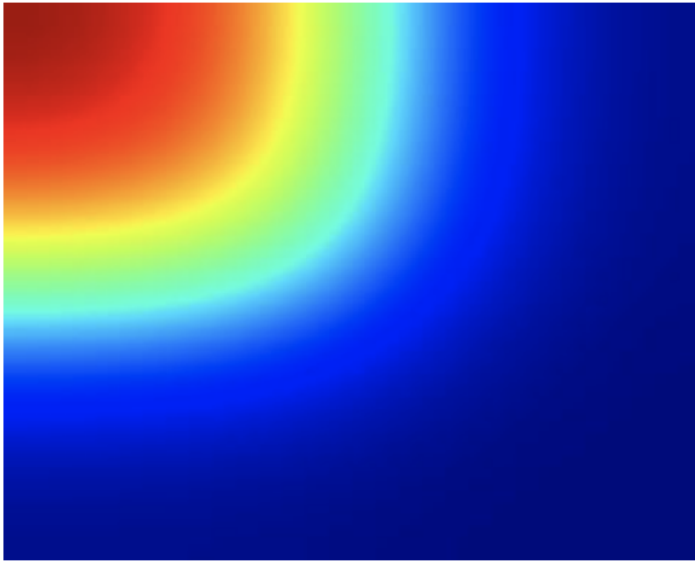
[36]

2.4. Experimento

En nuestro trabajo se experimentó con 5 modelos de ML para el experimento Perceptive y 5 modelos de DL para el experimento Stop&Go. Los modelos fueron entrenados con las características mencionadas en la sección 2.2.2, además para el experimento Stop&Go se obtuvieron imágenes del mapa de calor de la mirada del paciente, así como la trayectoria del movimiento ocular de los pacientes que fueron usadas para entrenar los modelos de DL.

Figura 12

Mapa de calor de la mirada de paciente

**Figura 13**

Trayectoria del movimiento ocular de paciente



2.5. Métricas de desempeño

En cuanto a la medición de los modelos se tomaron en cuenta las siguientes métricas

- Modelos de ML: exactitud (accuracy), precisión (precision) y el F1 score
- Modelos DL: exactitud (accuracy), precisión (precision)

2.6. Prototipo funcional

2.6.1. *Requerimientos funcionales*

Los requerimientos funcionales del sistema fueron:

- Una aplicación web debe permitir a los médicos cargar los archivos del paciente generados por el Eyetracker, visualizar el movimiento ocular y diagnosticar al paciente con la EP.
- Mostrar la exactitud en el diagnóstico del modelo en términos porcentuales, proporcionando a los médicos una medida cuantitativa de la confianza del diagnóstico.
- Integrar una sección para añadir observaciones clínicas o notas adicionales pertinentes al caso del paciente, permitiendo una interpretación más completa del diagnóstico.
- Distinguir entre los roles de Administrador y Médico. El rol del Administrador incluye la creación de usuarios médicos, la administración de los diagnósticos del médico, y la descarga de los archivos con las características médicas añadidas de uno o varios médicos. Por otra parte, el rol del Médico tiene atributos para ejecutar el modelo de diagnóstico, cargar los datos, visualizar los movimientos, diagnóstico y observaciones a los resultados del diagnóstico.
- Implementar acceso único de sesión de forma segura y sencilla por medio del correo y contraseña.

2.6.2. *Requerimientos no funcionales*

Los requerimientos no funcionales del sistema fueron:

- Usabilidad: La aplicación web debe priorizar la experiencia del usuario, por medio de una interfaz comprensible y fácil de usar para los médicos con cualquier nivel de habilidad técnica, mediante una navegación sencilla y de pocos botones e interacciones que guían al Médico para generar el diagnóstico del paciente.

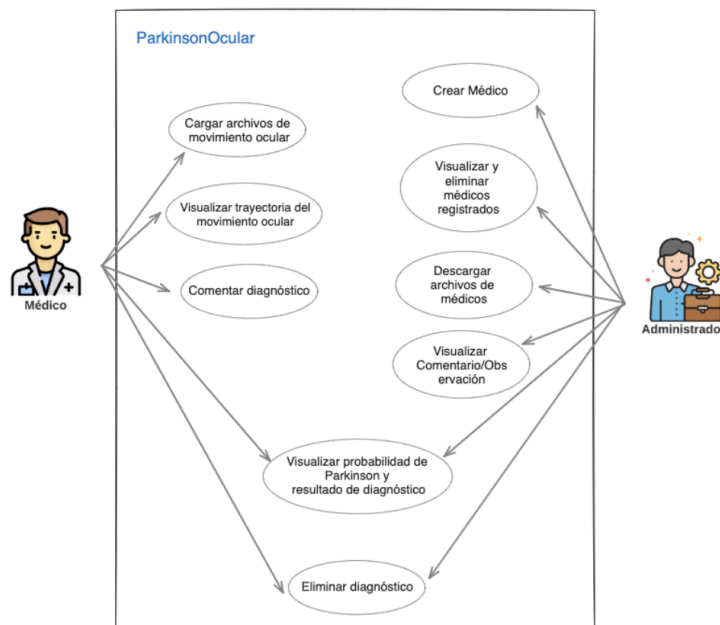
- **Precisión:** La aplicación web debe integrar un modelo de diagnóstico con un alto nivel de exactitud.
- **Accesibilidad:** La aplicación web debe garantizar el acceso a todos los usuarios registrados por el Administrador en el sistema, garantizando una inclusión completa y control de acceso dentro del sistema.
- **Seguridad:** La aplicación web debe contar con un robusto sistema de seguridad que utilice tokens de autenticación para proteger contra accesos no autorizados.

2.7. Diagramas del sistema

En el análisis se representó el siguiente diagrama de casos de uso (ver Figura 14).

Figura 14

Diagrama de casos de uso



2.8. Diseño inicial del prototipo

El prototipo consta de 6 pantallas. La primera pantalla principal permite ingresar con usuario y contraseña. Dependiendo del rol se puede acceder a la segunda pantalla en modo

Médico o Administrador. Si el usuario tiene el rol del Médico, entonces la segunda pantalla muestra la pantalla de diagnóstico, donde se cargan los archivos del movimiento ocular del paciente; al presionar el botón de analizar se muestra la trayectoria del movimiento ocular, así como la probabilidad de que el paciente tenga Parkinson. Se tiene además un área en donde el Médico pueda ingresar una observación de dicho diagnóstico. En caso de ser Administrador, la segunda pantalla permite realizar las tareas de crear más usuarios con rol de Médico. En la tercera pantalla, puede eliminar Médicos registrados, visualizar sus respectivos archivos, diagnósticos y observaciones, además de poder descargar la data cargada por los Médicos de forma masiva o unitaria con las características médicas obtenidas por Pymovements empleadas para el análisis.

2.9. Elección de tecnologías

Las tecnologías seleccionadas para el desarrollo del prototipo fue Django y PgAdmin, esto debido a la facilidad de uso, así como la compatibilidad con el desarrollo del modelo obtenido.

Capítulo 3

3. Resultados y Análisis

En esta sección, se presentan los resultados de la recolección de datos, el preprocesamiento, los modelos de aprendizaje pocos profundos, profundos y el prototipo funcional.

3.1. Recolección de datos

La recolección de los datos en ambos experimentos (Perceptivo y Stop&Go), se realizó mediante una sesión, donde el paciente tenía que responder unas preguntas en base a lo que se mostraba en el tomógrafo, mientras tenía unos controles en las manos para responder y todo esto usando el dispositivo del Eyetracker, el cual le permitía captar los movimientos. Los datos obtenidos del Eyetracker pasaban por un software que permitía transformar los archivos que se generaban con una extensión “eyd” al formato “xls”.

3.2. Preprocesamiento

3.2.1. Normalización

Para ambos experimentos, en el preprocesamiento de los datos se removieron primero los registros donde el paciente cierra los ojos puesto que no se obtienen coordenadas y tampoco diámetro de la pupila. Además, se eliminaron valores atípicos mediante rango intercuartil y se efectuó el balanceo de los segmentos, el cual es un conjunto de datos que son obtenidos al responder una pregunta por medio del dispositivo. En algunos pacientes se hicieron varias preguntas, por ello se buscó el mínimo número de segmentos en todos los pacientes, para tener un balance en los datos.

Además, para el experimento de Perceptive, se empleó el escalador MinMaxScaler de Sklearn donde se emplea la ecuación 1 en el conjunto de entrenamiento, ajustando los valores en un rango de entre cero y uno [37] [38].

$$X_{std} = \frac{X - X.\min(\text{axis} = 0)}{X.\max(\text{axis} = 0) - X.\min(\text{axis} = 0)} \quad (1)$$

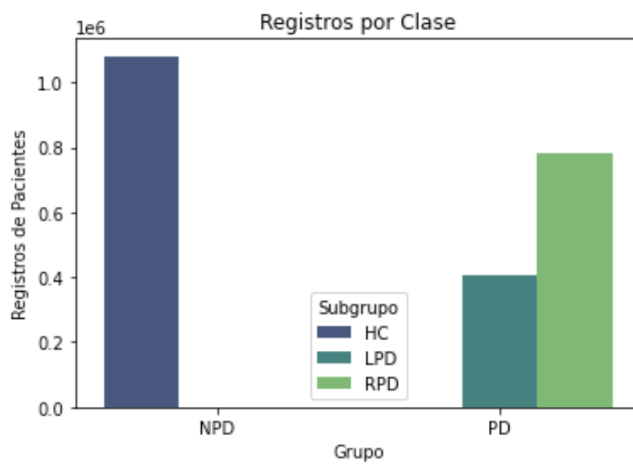
$$X_{scaled} = X_{std} \times (\max - \min) + \min$$

3.2.2. *Extracción de nuevas características*

Se obtuvieron nuevas características, tal como se mencionan en la sección 2.2.2, por medio de la librería Pymovements. En la extracción de estas nuevas características se usó la marca de tiempo de la captura del dato, el diámetro de la pupila del ojo, así como las coordenadas x , y . La librería trató a las coordenadas como posición y de éstas se obtuvo la velocidad y aceleración de la mirada del paciente. Además, se obtuvo otras características relacionadas a eventos como las sacadas, microsacadas, la duración de estos, así como la amplitud, dispersión y pico de velocidad. Para el experimento Stop&Go, gracias a estas nuevas características se pudieron generar las correlaciones a través de mapas de calor de la mirada del paciente y la trayectoria del movimiento ocular por segmento. Estas imágenes son usadas para los modelos de DL.

3.2.3. *Synthetic Data Augmentation*

Para el experimento Perceptive, se generaron nuevos datos mediante Data Augmentation con Synthetic Minority Over-sampling Technique (SMOTE, por sus siglas en inglés), librería de Python que permitió equilibrar la distribución de clases ampliando la representación de la clase minoritaria [40]. Inicialmente, el conjunto de entrenamiento consistía en 2,270,787 registros, de los cuales 1,080,931 pertenecían a la categoría ‘No Parkinson Diseases’ (NPD), que constaba de los sujetos del grupo de control HC; y 1,189,586 registros eran de la categoría ‘Parkinson Diseases’ (PD), subdividida en 408,402 registros de LPD y 781,454 registros de RPD. Mediante la aplicación de SMOTE, se logró un balance de 1,189,856 registros para cada una de las dos clases principales, facilitando así una comparación más equitativa entre ellas.

Figura 15*Datos de Entrenamiento SIN SMOTE***Figura 16***Datos de Entrenamiento CON SMOTE*

3.3. Modelos de Machine Learning

Para generar los modelos de aprendizaje correspondientes al experimento Perceptive, se instaló las librerías de ML necesarias, tales como: GridSearchCV para determinar los 'mejores' parámetros de entrenamiento para árboles de decisión aleatorios, por ejemplo:

`n_estimators`, `max_features` y `random_state`, dado que realiza una búsqueda exhaustiva entre los árboles [39], y además `RandomForestClassifier` para clasificar con estos parámetros y obtener los mejores resultados posibles.

Se llevaron a cabo cuatro sub-experimentos, cada uno con variaciones en los datos de entrenamiento, la distribución por clases y los parámetros del clasificador. En los dos sub-experimentos iniciales, se dividió el conjunto de datos donde el 80% de los sujetos se destinó al entrenamiento y el 20% a la prueba, de modo que los sujetos fueron clasificados de la siguiente manera, según las 3 clases: RPD con 8 pacientes (938,399 registros), HC con 10 (1,128,879 registros), y LPD con 3 (408,402 registros). Para los dos últimos sub-experimentos, la clasificación se realizó de tal manera que haya equilibrio entre la cantidad de pacientes y cantidad de registros entre las clases, es por ello que se decidió unificar las clases RPD y LPD en PD y HC en NPD, constanding cada una de 10 sujetos y aproximadamente 1,000,000 de registros.

3.3.1. Resultados de los experimentos

Según los sub-experimentos mencionados en la sección 3.3, el primer sub-experimento se configuró con 100 árboles de decisión, especificado en el parámetro `n_estimators`, y un valor en `random_state` de 42 para asegurar resultados consistentes y reproducibles entre ejecuciones. Este valor actúa como una semilla fija para el generador de números aleatorios, contribuyendo a la verificación de la estabilidad del modelo. En contraste, el segundo sub-experimento utilizó un `random_state` de 50. Con estas configuraciones, se logró un accuracy del 52% en el primer experimento y del 64% en el segundo, sin embargo, estos resultados no obtuvieron la precisión, el recall y f1-score requeridos en cada clase, sobretodo en la clase LPD (ver Tabla 1).

Tabla 1

Reporte Métricas de Clasificación Sub-exp 1, Sub-exp 2

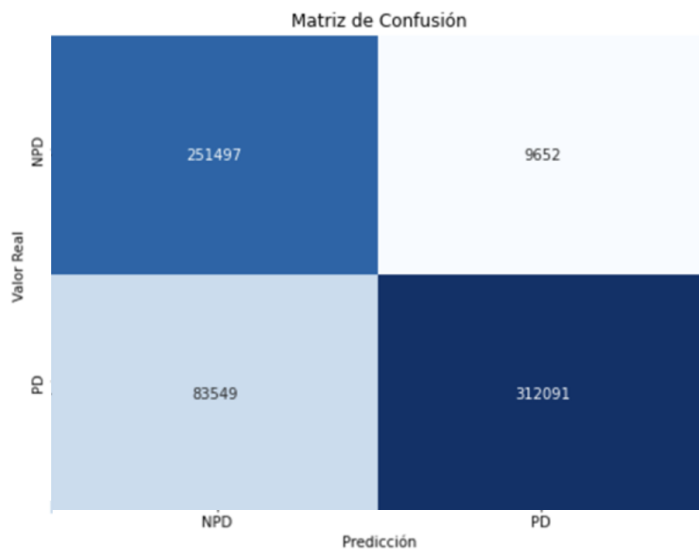
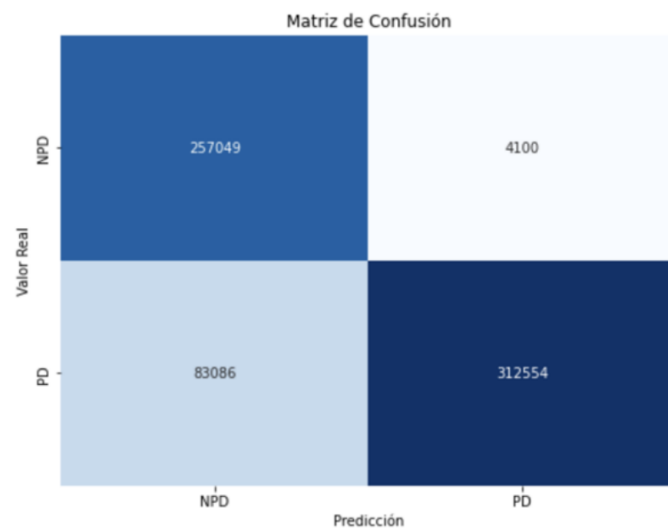
Configuración	Clase	Precisión	Recall	F1-score	Accuracy
1	HC	1.00	0.60	0.75	0.52
	LPD	0.00	0.00	0.00	
	RPD	0.33	1.00	0.49	
2	HC	1.00	0.86	0.93	0.64
	LPD	0.00	0.00	0.00	
	RPD	0.40	1.00	0.57	

Para el tercer y cuarto subexperimentos, el valor de `n_estimators` fue 100 y en `random_state` fue 55 y 70 respectivamente, lo que permitió obtener resultados cuyas métricas superaron el 70% (ver Tabla 2), obteniendo un accuracy del 86% y 87% respectivamente.

Tabla 2

Reporte Métricas de Clasificación Sub-exp 3, Sub-exp 4

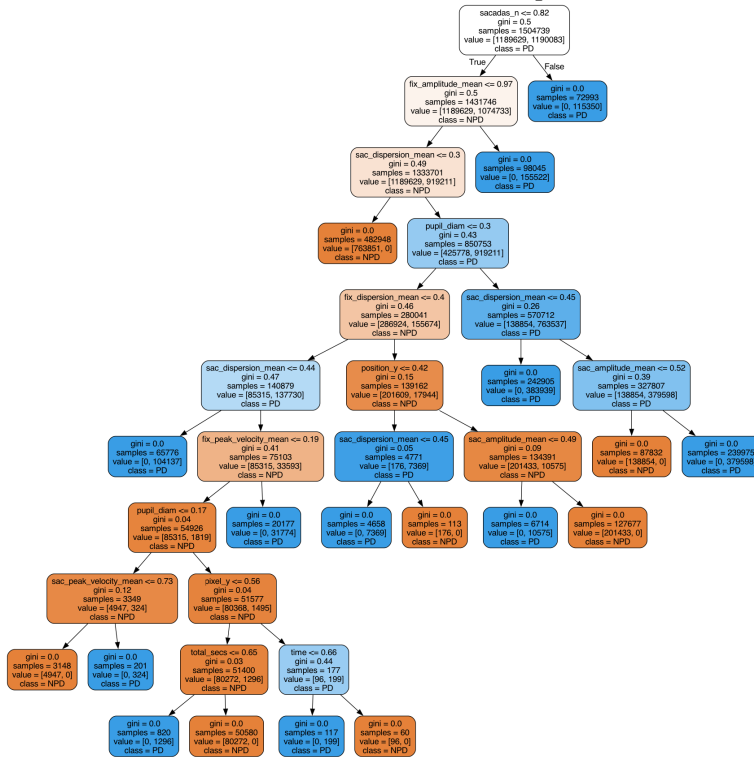
Configuración	Clase	Precisión	Recall	F1-score	Accuracy
1	NPD	0.75	0.96	0.84	0.86
	PD	0.97	0.79	0.87	
2	NPD	0.76	0.98	0.86	0.87
	PD	0.99	0.79	0.88	

Figura 17*Matriz de Confusión del Sub-exp.3***Figura 18***Matriz de Confusión del Sub-exp.4*

A continuación, se muestra la gráfica de uno de los árboles generados por el modelo de RandomForestClassifier del sub-experimento 4 donde se describen las relaciones y estructura del árbol, incluyendo nodos que representan a las decisiones y las flechas que representan las

conexiones entre los nodos (ver Figura 19). En los nodos azules, es donde se hacen preguntas sobre una característica y valor específicos para dividir el conjunto de datos, conteniendo el número de muestras que llegan al nodo, el valor de la clase que predomina en ese nodo, y la impureza del nodo; mientras que los nodos naranjas son las predicciones finales.

Figura 19
Estructura del Árbol de Decisión del Sub-exp. 4



En base a la clasificación de datos en donde se contenía solamente dos clase: NPD y PD, se obtuvieron los siguientes resultados de métricas de calidad los modelo mencionados en la sección 2.3.1:

Tabla 3

Otras Métricas de Calidad

Configuración	Clase	Precisión	Recall	F1-score	Accuracy
Regresión Logística	NPD	0.67	0.67	0.67	
	PD	0.86	0.86	0.86	0.80
2	NPD	0.64	0.99	0.78	
	PD	0.99	0.63	0.77	0.77

3.4. Modelos de Deep Learning

Los modelos de DL fueron generados por medio de las librerías Pytorch y Tensorflow. Para la ejecución de los experimentos con estos modelos se utilizó la supercomputadora del CEDIA, la cual cuenta con GPUs.

El proceso de entrenamiento de cada modelo se realizó con los mapas de calor de la mirada del paciente y la trayectoria del movimiento ocular del paciente. Se contó con 8 HC, 11 LPD, 11 RPD. Por medio de la librería Pymovements se extrajeron las características que se mencionan en la sección 2.2.2 y se generaron las imágenes los mapas de calor y las trayectorias. Para el entrenamiento se etiquetaron las imágenes con '0' para pacientes de HC, '1' para pacientes LPD y '2' RPD, luego se procedió a balancear el número de imágenes por cada etiqueta, ya que en cada paciente se generó un número de eventos distintos. Se destinaron 1200 imágenes que fueron repartidas en entrenamiento, test y validación, tanto para mapas como para trayectorias. La división del dataset de imágenes fue de 80% para entrenamiento, 10% de testeo y 10% de validación.

Para cada modelo de DL se usó Transfer Learning (TL), en donde se configuró la entrada del modelo para recibir las imágenes obtenidas del experimento. Se entrenó los modelos de VGG-16, VGG-19 y ResNet50 tanto para mapas de calor y la trayectoria del movimiento con 3 configuraciones de la arquitectura. Para InceptionV3 y MobileNet solo para mapas de calor y solo se realizó una configuración en la arquitectura. La primera configuración es una la modificación del Dropout en 0.5, la segunda configuración es de un Dropout de 0.5 y el uso de la función de activación Leaky-Relu, y en la tercera configuración el Dropout de 0.3 y la función de activación de ReLU.

3.4.1. Resultados de los experimentos

Las métricas de entrenamiento obtenidas en los modelos son exactitud (accuracy) y precisión (precision). Se construyó dos tablas, una con los resultados obtenidos del experimento del mapa de calor y otra tabla para el experimento de la trayectoria del movimiento ocular. El modelo con más alta precisión para los mapas de calor fue el modelo de RestNet50 en el entrenamiento con un 41.13% de precisión y un 1.0868% error. Para la trayectoria del movimiento ocular el modelo con mayor precisión fue el modelo de RestNet50 con una 33.80% de precisión y un 1.1026% de error (Ver Tabla 4 y Tabla

Tabla 4

Resultados de experimentos en imágenes de mapa de calor

Configuración	Modelo	Entrenamiento		Validación	
		Precisión	Error	Precisión	Error
1	VGG-16	0.2077	0.0173	0.4708	0.0344
	VGG-19	0.2081	0.0174	0.2081	0.0372
	RestNet50	0.39994	1.0912	0.4708	1.0672
	InceptionV3	0.4158	1.5172	0.4708	1.0638
	MobileNet	0.4007	1.1140	0.4631	1.050
2	VGG-16	0.1513	0.0175	0.2646	0.1513
	VGG-19	0.1468	0.0175	0.2646	0.0351
	RestNet50	0.4088	1.0871	0.4708	1.0662
3	VGG-16	0.1562	0.0175	0.2646	0.0350
	VGG-19	0.2150	0.0173	0.4708	0.0345
	RestNet50	0.4113	1.0868	0.4708	1.0659

Tabla 5

Resultados de experimentos en imágenes de trayectoria

Configuración	Modelo	Entrenamiento		Validación	
		Precisión	Error	Precisión	Error
1	VGG-16	0.3472	0.0113	0.7014	0.0218
	VGG-19	0.3278	0.0122	0.6708	0.0239
	ResNet50	0.3306	1.1044	0.3431	1.1013
2	VGG-16	0.1625	0.0174	0.3333	0.0352
	VGG-19	0.1685	0.0173	0.3333	0.0351
	ResNet50	0.3069	1.1048	0.3097	1.1026
3	VGG-16	0.1708	0.0173	0.3333	0.0351
	VGG-19	0.1653	0.0173	0.3333	0.0352
	ResNet50	0.3380	1.1026	0.3403	1.1012

Figura 20

VGG-16 – Mapa de calor precisión y error

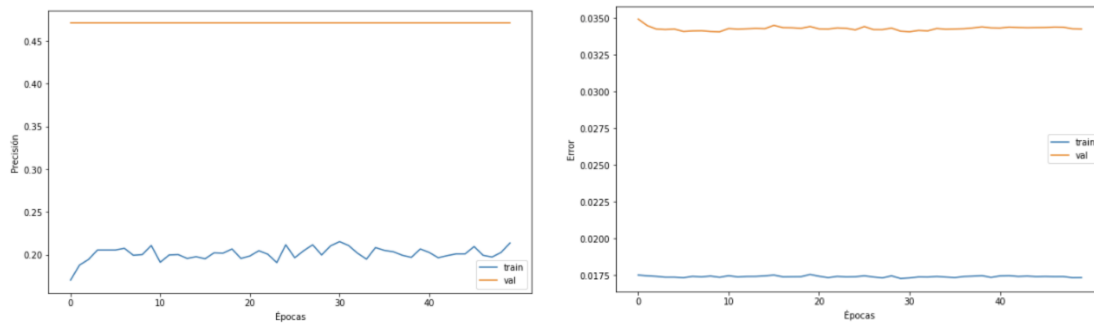
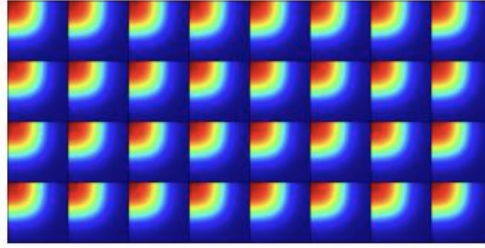


Figura 21

Evaluación del modelo VGG-16 en mapa de calor

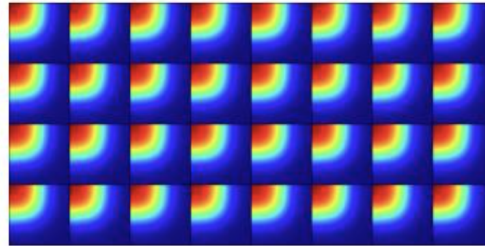
Ground truth:

['0', '0', '1', '0', '2', '1', '2', '1', '2', '1', '2', '2', '2', '1', '1', '0', '0', '2', '1', '0', '1', '2', '1', '2', '1', '0', '2', '1', '2', '2', '2', '1']



Prediction:

['1', '1']

**Figura 22**

VGG-16 – Trayectoria precisión y error

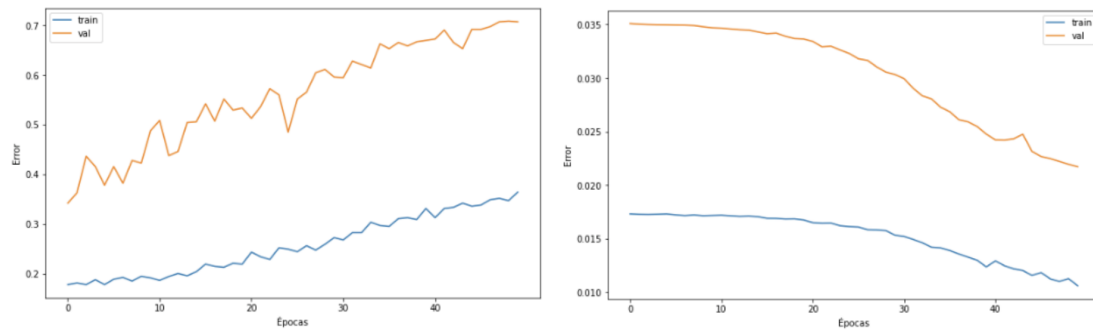
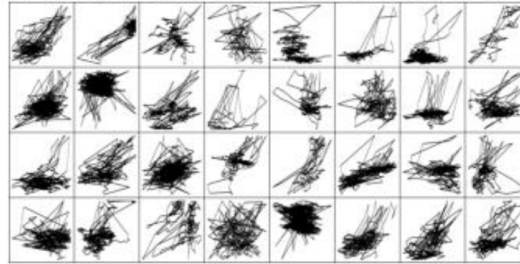


Figura 23

Evaluación del modelo VGG-16 en trayectoria

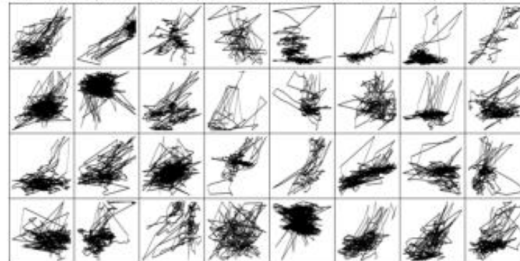
Ground truth:

['0', '0', '2', '0', '1', '2', '2', '2', '1', '1', '2', '2', '0', '1', '0', '1', '2', '0', '1', '2', '2', '0', '0', '0', '0', '2', '0', '2', '1', '1', '2', '2']



Prediction:

['2', '0', '2', '0', '2', '2', '2', '2', '1', '1', '0', '0', '1', '2', '2', '2', '1', '2', '1', '2', '0', '0', '0', '0', '0', '2', '0', '2', '1', '2', '2', '2']



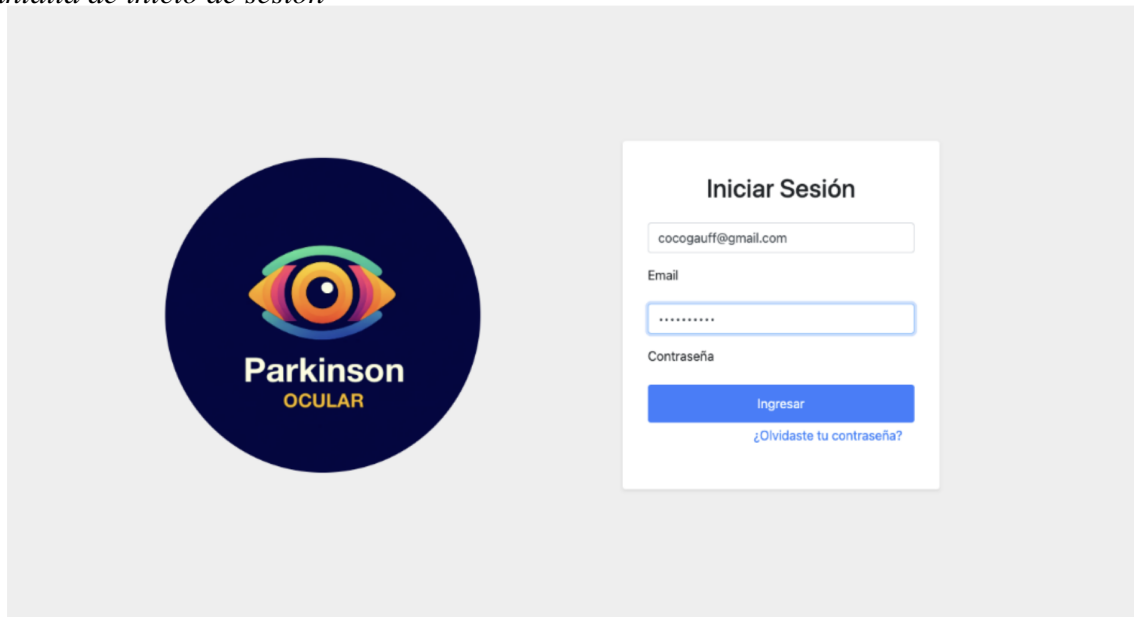
Los motivos por los cuales no se llevaron a cabo las demás configuraciones en las otras redes yace en que no se nota una mejora en base a los demás modelos.

3.5. Prototipo funcional

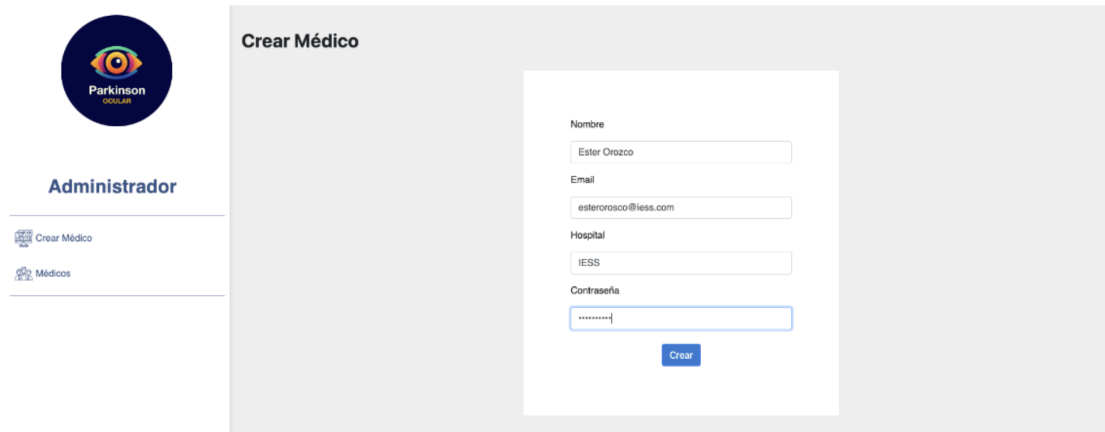
El prototipo funcional se desarrolló con Django y PgAdmin. Django es una herramienta que permitió crear el sitio web de una manera rápida facilitando la autenticación de usuarios y administración de su contenido. El sistema consta de dos roles, el cliente y el Administrador que inician la sesión en un mismo punto de acceso al sitio web. El inicio de sesión posee dos entradas de texto, uno para el correo electrónico y la contraseña (ver Figura 24).

Figura 24

Pantalla de inicio de sesión

**3.5.1. Rol de Administrador**

En el rol Administrador se provee una pantalla que le permite registrar a un nuevo Médico al sistema. Para el registro del Médico es necesario llenar los campos de nombre, email, hospital y contraseña. Una vez llenados los campos de forma exitosa, el botón “Crear” permite terminar el registro del Médico (Ver Figura 25).

Figura 25*Pantalla de creación del Médico*

The screenshot shows a web interface for creating a doctor. On the left, there is a sidebar with the Parkinson Ocular logo and the role 'Administrador'. The main content area is titled 'Crear Médico' and contains a form with the following fields:

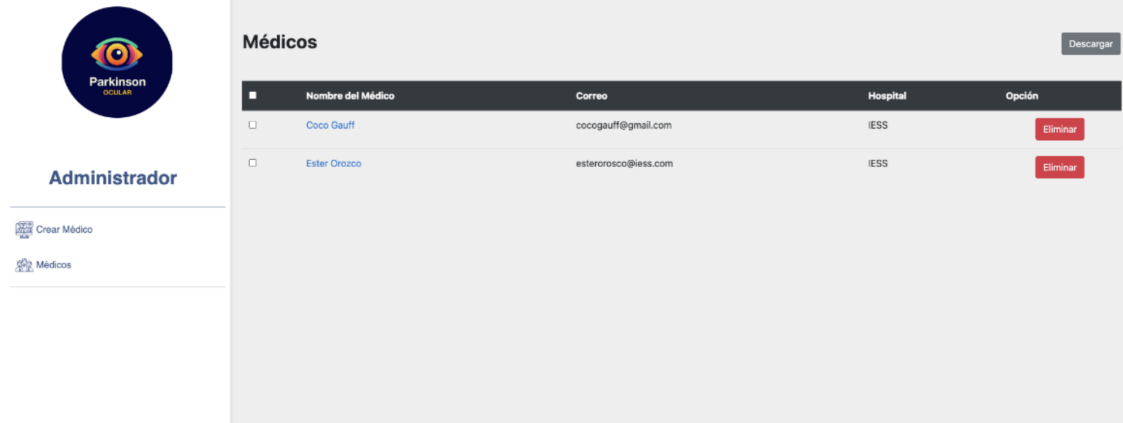
- Nombre: Ester Orozco
- Email: esterorozco@iess.com
- Hospital: IESS
- Contraseña: [masked]

A blue 'Crear' button is positioned below the password field.

Para ver el registro de Médicos agregados al sistema, en el menú del rol Administrador, al seleccionar la opción Médicos, se muestra la lista de Médicos, donde se visualiza la información de los médicos registrados en el sistema. Para cada Médico se muestra el nombre, correo, hospital y la opción para eliminarlo, pero también el botón descargar (ver Figura 26). El Administrador también cuenta con la opción de visualizar los archivos de cada Médico, desde la pantalla de “Médicos”, al hacer clic en el nombre del Médico se muestra una pantalla de archivos, en la que se muestra el nombre del archivo, la fecha de cuando fue subido el mismo y las opciones de eliminar y analizar; así como el botón de descargar (ver Figura 27).

Figura 26

Médicos del sistema



Administrador

Crear Médico

Médicos

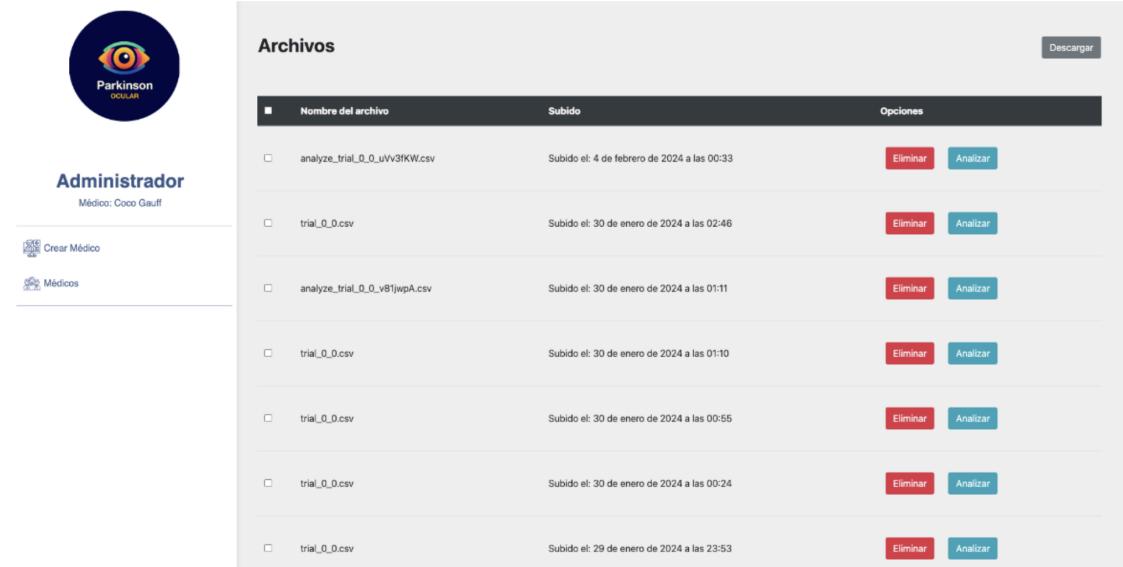
Médicos

Descargar

Nombre del Médico	Correo	Hospital	Opción
<input type="checkbox"/> Coco Gauff	cocogauff@gmail.com	IESS	Eliminar
<input type="checkbox"/> Ester Orozco	esterorosco@iess.com	IESS	Eliminar

Figura 27

Archivos del Médico



Administrador

Médico: Coco Gauff

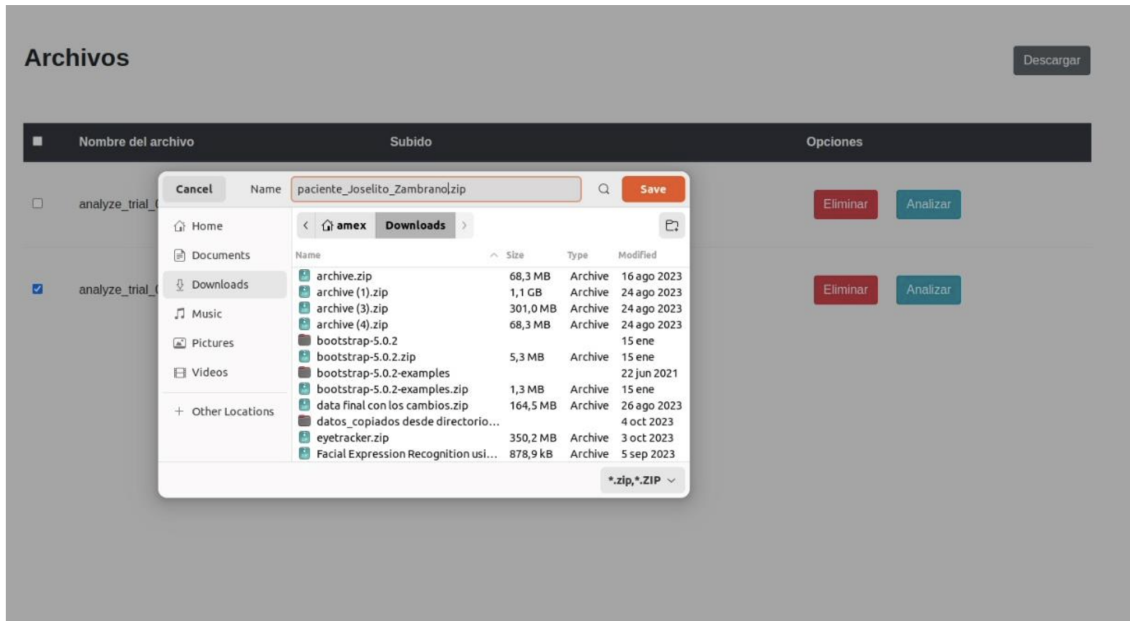
Crear Médico

Médicos

Archivos

Descargar

Nombre del archivo	Subido	Opciones
<input type="checkbox"/> analyze_trial_0_0_uVv2IKW.csv	Subido el: 4 de febrero de 2024 a las 00:33	Eliminar Analizar
<input type="checkbox"/> trial_0_0.csv	Subido el: 30 de enero de 2024 a las 02:46	Eliminar Analizar
<input type="checkbox"/> analyze_trial_0_0_v8TjwpA.csv	Subido el: 30 de enero de 2024 a las 01:11	Eliminar Analizar
<input type="checkbox"/> trial_0_0.csv	Subido el: 30 de enero de 2024 a las 01:10	Eliminar Analizar
<input type="checkbox"/> trial_0_0.csv	Subido el: 30 de enero de 2024 a las 00:55	Eliminar Analizar
<input type="checkbox"/> trial_0_0.csv	Subido el: 30 de enero de 2024 a las 00:24	Eliminar Analizar
<input type="checkbox"/> trial_0_0.csv	Subido el: 29 de enero de 2024 a las 23:53	Eliminar Analizar

Figura 28*Descargar Archivos del Médico*

3.5.2. Rol del Médico

En el rol del Médico, una vez que se ingrese al sistema, se muestra la pantalla de diagnóstico, en la sección de subir archivos, se puede cargar los archivos en formato .xls y subirlos al sistema por medio del botón subir (ver Figura 29). También se muestra un botón analizar que permite obtener el diagnóstico del paciente y crear observaciones que pueden ser guardadas con los resultados del análisis, así como el botón cancelar (ver Figura 30).

Figura 29*Subir Archivos*

Figura 30

Archivos Seleccionados

Diagnóstico

Subir archivos Subir

Arrastrar y soltar o buscar archivos

- 2009-04-06-18.57.55_juan_nuiz_RPD_TS_de_eyd.xls - 8.76 MB
- 2009-04-06-18.15.53_juan_nuiz_RPD_TS_la_eyd.xls - 8.81 MB
- 2009-04-06-19.33.03_juan_nuiz_RPD_TS_bim_eyd.xls - 9.86 MB

Figura 31

Diagnóstico de Parkinson

Gráfico de Movimiento Ocular

Diámetro de la Pupila (mm)

Tiempo (s)

Tiempo seleccionado (s): -

- Menor a 2 mm
- 2 - 3.2 mm
- 3.2 - 3.5 mm
- 3.5 - 4 mm
- Mayor a 4 mm

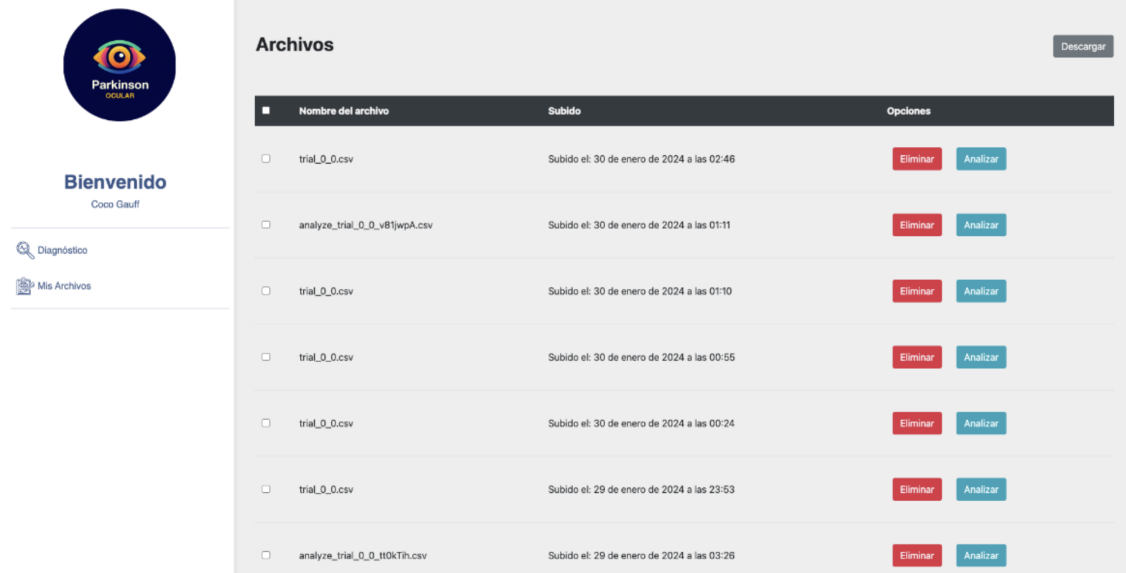
Coordenadas Y

Coordenadas X

Al igual que en el rol del Administrador, en el rol del Médico se puede ver la lista de archivos que se han cargado al sistema y se pueden descargar los análisis anteriores, o eliminarlos (ver Figura 32).

Figura 32

Lista de archivos del Médico



The screenshot displays the 'Archivos' (Files) section of a web application. On the left, there is a sidebar with a logo for 'Parkinson' and a welcome message: 'Bienvenido Coco Gaufl'. Below the welcome message are two menu items: 'Diagnóstico' and 'Mis Archivos'. The main content area is titled 'Archivos' and features a 'Descargar' button in the top right corner. A table lists the uploaded files with the following columns: 'Nombre del archivo', 'Subido', and 'Opciones'. Each row includes a checkbox, the file name, the upload date and time, and two buttons: 'Eliminar' (red) and 'Analizar' (blue).

Nombre del archivo	Subido	Opciones
<input type="checkbox"/> trial_0_0.csv	Subido el: 30 de enero de 2024 a las 02:46	Eliminar Analizar
<input type="checkbox"/> analyze_trial_0_0_v81jwpA.csv	Subido el: 30 de enero de 2024 a las 01:11	Eliminar Analizar
<input type="checkbox"/> trial_0_0.csv	Subido el: 30 de enero de 2024 a las 01:10	Eliminar Analizar
<input type="checkbox"/> trial_0_0.csv	Subido el: 30 de enero de 2024 a las 00:55	Eliminar Analizar
<input type="checkbox"/> trial_0_0.csv	Subido el: 30 de enero de 2024 a las 00:24	Eliminar Analizar
<input type="checkbox"/> trial_0_0.csv	Subido el: 29 de enero de 2024 a las 23:53	Eliminar Analizar
<input type="checkbox"/> analyze_trial_0_0_ttk0Th.csv	Subido el: 29 de enero de 2024 a las 03:26	Eliminar Analizar

Capítulo 4

4. Conclusiones y Recomendaciones

El trabajo realizado representa un avance importante en el área de análisis de movimientos oculares en la EP. Igualmente, la aportación de las características obtenidas en los experimentos y el uso de ellas de forma distinta para cada experimento contribuye a la mejora del estado del arte.

4.1. Conclusiones

Con el experimento Perceptive, el modelo Random Forest fue fundamental para clasificar datos de manera efectiva. Los sub-experimentos que conducen a métricas de calidad superiores al 70% son significativos en la mejora de la precisión, las métricas como Recall y f1-score, fueron esenciales para evaluar la eficacia de los modelos de clasificación.

En el sub-experimento 3, el número de falsos positivos fue más alto en comparación con el sub-experimento 4, lo que podría indicar que el modelo del sub-experimento 3 tuvo una tendencia a predecir la clase positiva más a menudo que el modelo del sub-experimento 4, lo cual puede no ser deseable dado que, bajo el contexto médico, un alto número de falsos positivos podría llevar a tratamientos innecesarios. En el sub-experimento 4, el número de verdaderos negativos fue notablemente más alto y el número de falsos positivos considerablemente más bajo, lo que sugiere que es más conservador al predecir la clase positiva y posiblemente más eficaz en la clasificación general, con una mejor capacidad para identificar y clasificar correctamente las instancias de cada clase.

Con respecto al experimento Stop&Go, ciertamente se buscó el estado del arte usando las diferentes arquitecturas propuestas de DL. Sin embargo, todos los modelos no llegaron a superar el 40% de precisión a pesar de la variedad de experimentos diseñados. Uno de los motivos por el cual los modelos no lograron extraer los patrones de la enfermedad, podría deberse a la complejidad de las arquitecturas para el dataset que se tiene.

La aplicación ParkinsonOcular desarrollada, brinda a los profesionales de la salud la capacidad de realizar análisis oculares detallados, permitiéndoles evaluar la presencia de anomalías en el movimiento ocular de sus pacientes. La visualización de una gráfica inicial ayuda a los médicos a observar las variaciones en el diámetro pupilar a lo largo del tiempo,

utilizando una herramienta de deslizamiento temporal para un análisis minucioso.

Adicionalmente, una segunda gráfica ofrece una representación de las posiciones de la pupila donde se identifican cambios en el diámetro, categorizados en tres rangos: menor a 3,2mm, entre 3.2mm y 4mm, y mayor a 4mm.

La capacidad de diagnóstico de la aplicación se ve reforzada por la inclusión de un modelo de clasificación RF (Random Forest), que proporciona un diagnóstico preliminar sobre la presencia de la Enfermedad de Parkinson con una precisión del 87%. Este nivel de precisión destaca la eficacia del modelo inteligente en la contribución al diagnóstico temprano y seguimiento de la enfermedad.

Además, facilita la descarga de datos enriquecidos con nuevas características médicas obtenidas a través de Pymovements, incluyendo métricas como sacadas, microsacadas y fijaciones. Esta funcionalidad permite a los Médicos realizar análisis adicionales, profundizando en el estudio del comportamiento ocular de sus pacientes. Por otro lado, admite la adición de observaciones personalizadas sobre los resultados obtenidos, lo que contribuye a la creación de un historial completo de análisis y resultados para cada paciente. Esta característica es esencial para el seguimiento a largo plazo y la evaluación continua de la condición del paciente.

Desde una perspectiva de gestión, el Administrador del sistema tiene la capacidad de crear cuentas para nuevos Médicos, así como de monitorear y revisar los datos, análisis y observaciones realizados por cada profesional. Esta funcionalidad de supervisión asegura un control efectivo y facilita el seguimiento de los casos manejados por el equipo médico.

4.2. Recomendaciones

- Para implementaciones futuras se recomienda generar otras características médicas para ser empleadas en los modelos de ML que permitan aumentar la precisión al momento de obtener los resultados.
- Para los modelos de DL, se recomienda balancear la complejidad de imagen con el número de capas en cada modelo. Esto es porque las redes neuronales más complejas capturan mejor imágenes más complejas, pero pueden caer en underfitting con imágenes

no complejas.

Referencias

- [1] J. Jankovic, “Parkinson’s disease: clinical features and diagnosis,” *J. Neurol. Neurosurg. Psychiatry*, vol. 79, no. 4, pp. 368–376, 2008.
- [2] M. C. Rodriguez-Oroz, “Initial clinical manifestations of Parkinson’s disease: features and pathophysiological mechanisms,” *Lancet Neurol*, vol. 8, no. 12, pp. 1128–1139, 2009.
- [3] E. P. Optican and L. M., “Eye Movements in Parkinson’s Disease and Inherited Parkinsonian Syndromes,” *Front. Neurol*, vol. 8, no. 592, pp. 1664–229, 2017.
- [4] L. R. de Pedro, “Movimiento humano, trastornos motores y núcleo caudado,” Ph.D. dissertation, Universidad de la Laguna, San Cristóbal, España, 2018.
- [5] M. W. y M. D. P., “Probing basal ganglia functions by saccade eye movements,” *European Journal of Neuroscience*, vol. 33, pp. 2070–2090, 2011.
- [6] R. J. L. y C. Kennard, “Using saccades as a research tool in the clinical neurosciences,” *Brain*, vol. 127, no. 3, pp. 460–477, 2004.
- [7] R. J. L. y D. S. Zee, “The neurology of Eye Movements,” 2015.
- [8] M. P. y S. E. D. P. Munoz M. C. Dorris, “On your mark, get set: Brainstem circuitry underlying saccadic initiation,” *Canadian journal of physiology and pharmacology*, vol. 78, no. 11, pp. 934–944, 2000.
- [9] D. G. Krakowczyk, D. R. Reich, J. Chwastek, *et al.*, “Pymovements: A python 46 package for eye movement data processing,” *Symposium on Eye Tracking Research and Applications*, 2023.
- [10] O. M. de la Salud. “Enfermedad de parkinson.” (2021), [Online]. Available: <https://www.who.int/es/news-room/fact-sheets/detail/parkinson-disease..>

- [11] A. Srivastava, R. Sharma, S. K. Sood, G. Shukla, V. Goyal, and M. Behari, “Saccadic eye movements in parkinson’s disease,” *Indian Journal of Ophthalmology*, vol. 62, no. 5, pp. 538–544, 2014.
- [12] L. Höglund, “Analysis of eye tracking data from parkinson’s patients using machine learning,” *Dissertation*, 2021.
- [13] D. C. Brien, H. C. Riek, R. Yep, and J. Huang, “Classification and staging of parkinson’s disease using video-based eye tracking,” *Parkinsonism Related Disorders*, vol. 110, p. 105 316, 2023.
- [14] R. Zemblys, D. C. Niehorster, and K. Holmqvist, “End-to- end eye- movement event detection with deep neural networks,” *Behavior Research Methods*, vol. 51, no. 2, pp. 840–864, 2019.
- [15] D. Bank, N. Koenigstein, and R. Giryes, “Autoencoders,” *Arxiv*, 2003.
- [16] I. Goodfellow, Y. Bengio, and A. Courville. “Deep learning.” (), [Online]. Available: <http://www.deeplearningbook.org>.
- [17] W. .-. T. F. Encyclopedia. “Support vector machine.” (2023), [Online]. Available: https://en.wikipedia.org/wiki/Support_vector_machine.
- [18] Krakowczyk, D. G., Reich, *et al.*, “A python package for processing eye movement data,” *ACM Digital Library*, no. 53, pp. 1–2, 2023.
- [19] S. Learn. “Scikit learn.” (2023), [Online]. Available: https://scikit-learn.org/stable/auto_examples/ensemble/plot_forest_importances.html.
- [20] A. Subasi, “Chapter 3 - machine learning techniques,” *Practical Machine Learning for Data Analysis Using Python*, pp. 91–202, 2020.
- [21] Scikit-learn. “Decision trees - scikit-learn 0.24.2 documentation.” (2023), [Online]. Available: <https://scikit-learn.org/stable/modules/tree.html>.
- [22] S. Zivkovic. “Machine learning - decision tree.” (2023), [Online]. Available: <https://datahacker.rs/011-machine-learning-decision-three/>.

- [23] S. Zivkovic. "Machine learning - introduction to random forest." (2023), [Online]. Available: <https://datahacker.rs/012-machine-%20learning-introduction-to-random-forest/>.
- [24] L. M. Gladence, M. Karthi, and V. M. Anu, "A statistical comparison of logistic regression and different bayes classification methods for machine learning," *Asian Research Publishing Network*, no. 14, 2015.
- [25] S. Zivkovic. "Machine learning – logistic regression models." Data Hacker. (2022), [Online]. Available: <https://datahacker.rs/004-machine-%20learning-logistic-regression-model/>.
- [26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *ICLR*, 2015.
- [27] K. Le. "An overview of vgg-16 and nin models." (2021), [Online]. Available: <https://medium.com/mllearning-ai/an-overview-of-VGG-16-and-nin-models-96e4bf398484>.
- [28] S. Mascarenhas and M. Agarwal, "A comparison between vgg-16, vgg-19 and resnet-50 architecture frameworks for image classification," *International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*, 2021.
- [29] Y. Zheng, C. K. Yang, and A. Merkulov, "Breast cancer screening using convolutional neural network and follow-up digital mammography," 2018.
- [30] S. Mukherjee. "The annotated resnet-50." (2022), [Online]. Available: <https://towardsdatascience.com/the-annotated-resnet-50-a6c536034758..>
- [31] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

- [33] S. Barratt and R. Sharma, “A note on the inception score,” *Proc. ICML 2018 Workshop on Theoretical Foundations and Applications of Deep Generative Models*, 2018.
- [34] “Inception-v3.” (2022), [Online]. Available:
<https://paperswithcode.com/method/inception-v3#:~:text=Inception-v3%20is%20a%20convolutional,use%20of%20batch%20normalization%20for..>
- [35] A. G. Howard, M. Zhu, B. Chen, T. W. D. Kalenichenko W. Wang, M. Andreetto, and H. Adam, “Mobilenets: Efficient convolutional neural networks for mobile vision applications,” *Computer Vision and Pattern Recognition*, 2017.
- [36] K. Kadam, S. Ahirrao, and K. Kotecha, “Efficient approach towards detection and identification of copy move and image splicing forgeries using mask r-cnn with mobilenet v1,” *Hindawi*, pp. 1–21, 2022.
- [37] M. G. Lanjewar and J. S. Parab, “Machine learning approach with data normalization technique for early stage detection of hypothyroidism,” *Min-Max Technique*, pp. 95–96, 2022.
- [38] “Minmaxscalar preprocessing.” (2023), [Online]. Available:
<https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.MinMaxScaler.html>.
- [39] V. J and S. M. PRIYA, “Hyper tuning using gridsearchcv on machine learning models for prognosticating dementia,” *Research Square*, 2022.