EVALUACIÓN DE MODELOS DE APRENDIZAJE AUTOMÁTICO PARA PREDECIR EL RIESGO DE DESARROLLAR LA ENFERMEDAD DE PARKINSON

JHON MICHAEL ORTIZ DIAZ
SERGIO DANIEL BELTRÁN FORERO

UNIVERSIDAD ANTONIO NARIÑO
FACULTAD DE INGENIERÍA DE SISTEMAS
PROGRAMA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
BOGOTÁ
2021

EVALUACIÓN DE MODELOS DE APRENDIZAJE AUTOMÁTICO PARA PREDECIR EL RIESGO DE DESARROLLAR LA ENFERMEDAD DE PARKINSON

JHON MICHAEL ORTIZ DIAZ
SERGIO DANIEL BELTRÁN FORERO

DIRECTOR:
JUAN CAMILO RAMIREZ

UNIVERSIDAD ANTONIO NARIÑO
FACULTAD DE INGENIERÍA DE SISTEMAS
PROGRAMA DE INGENIERÍA DE SISTEMAS Y COMPUTACIÓN
BOGOTÁ
2021

TÍTULO DEL TRABAJO DE GRADO

Evaluación de modelos de aprendizaje automático para predecir el riesgo de desarrollar la enfermedad de Parkinson

CONTENIDO

Pág.

TABLA DE ILUSTRACIONES

RESUMEN

Las enfermedades que llegan a padecer los seres humanos en muchos casos no tienen un origen claro, pero cuando afecta a una persona impide que tenga una vida adecuada. Tal es el caso de la enfermedad de Parkinson, se trata de un padecimiento neurodegenerativo que les da a las personas en cierta etapa de su vida, pese a que los medicamentos podrían o no ayudar a vivir con esa condición, en la actualidad no existe una cura como tal y cada vez las cifras sobre la cantidad de personas que padecen esta enfermedad crecen porque no hay un tratamiento efectivo. Algo que podría ayudar a predecirlo tiene que ver con el uso del aprendizaje automático, a partir de datos que se tienen, se puede determinar que condición tendrá esa persona, es decir si desarrollará o no la EP (Enfermedad del Parkinson), con ayuda de estudios y datos obtenidos relacionados al tema se pueden entrenar modelos para hacer una mejor predicción.

La metodología utilizada para la investigación, consta de 3 partes, la parte de análisis donde se obtuvo el dataset y se identificó la cantidad de datos que se tienen, seguido de una fase de implementación donde se llevó a cabo el entrenamiento con ayuda de los modelos elegidos en este caso redes neuronales y bosques aleatorios, y se concluye con una fase de evaluación a partir de los resultados obtenidos para determinar entre otras cosas cual es el mejor modelo para obtener la mejor predicción posible.

Todo esto fue posible utilizando el lenguaje de programación Python y con ayuda de la librería de Scikit learn, y de los modelos de entrenamiento ya mencionados, se pudo llegar a obtener resultados de predicción de una manera acorde a lo que

se esperaba, y así obtener resultados lo más preciso posibles para brindar una información más clara en cuanto a detección de posibles desarrolladores de la enfermedad conocida como E.P

ABSTRACT

The diseases that human beings suffer in many cases do not have a clear origin, but when this (the disease) affects a person it prevents me from having an adequate life. Medicines could help to live with this condition, there is no cure as such and every time the figures on the number of people suffering from this disease grow there is no effective treatment, something that could help predict it has to do with the use of learning Automatic, from the data we have, it can be determined what that person will have, that is, whether or not they will develop PD (Parkinson's Disease), with the help of studies and data obtained related to the subject, models can be trained to make a better prediction.

The methodology used for the research consists of 3 parts, the analysis part where the dataset was obtained and the amount of data available was identified, followed by an implementation phase where training was carried out with the help of the models. chosen in this case neural networks and random forests, and concludes with an evaluation phase from the results obtained to determine, among other things, which is the best model to obtain the best possible prediction.

All this was possible using the Python programming language and with the help of the Scikit learn library, and the training models already mentioned, it was possible to

obtain prediction results in a way that was expected, and thus obtain results as accurate as possible to provide clearer information regarding the detection of potential developers of the disease known as EP.

INTRODUCCIÓN

El Parkinson es una anomalía que ocurre en los seres humanos, es decir es una enfermedad que puede aparecer por herencia (genética), y otros factores de salud (enfermedades previas); esta enfermedad consiste en el movimiento involuntario de las articulaciones o partes del cuerpo humano (manos, pies, cabeza) (Medlineplus, 2020).

Según estudios de la Federación Española del Parkinson, más de siete millones de personas en el mundo padecen de este síndrome; en unos años esta cifra pasará a ser de doce millones y para el año 2040 según proyecciones de esta misma organización la EP(Enfermedad de Parkinson) será la más grave y más común (Federacion Española, s/f). En el caso de Colombia las cifras hablan de que un poco más de 220.000 personas tienen esta enfermedad según el Ministerio de Salud (conexion capital, 2018).

Pese a que esta enfermedad como tal no tiene cura, existen algunos ejercicios realizados con ayuda de un fisioterapeuta, los cuales ayudan a que esta situación mejore. También existen algunos medicamentos que al consumirlos ayudan a liberar más dopamina (sustancia la cual carecen quien tenga este tipo de situación) como son la levodopa y la carbidopa. No existe como tal una manera de prevenir el Parkinson debido a que no es claro por qué aparece en las personas (Itziar Gastón Zubimendi, 2018).

Para ayudar a predecir mejor cuando un paciente tiene Parkinson, el aprendizaje automático logra ser una gran alternativa en esta materia, ya que este se aplica en detección de otras enfermedades como es la diabetes o el cáncer, esto se podría realizar con modelos, en este caso, redes neuronales o bosques aleatorios; por ejemplo, en el estudio titulado “A comparative analysis of speech signal processing algorithms for Parkinson’s disease classification and the use of the tunable Q-factor wavelet transform. Applied Soft Computing Journal”, se utilizan modelos de transformada de ondículas de factor Q sintonizable (TQWT) y los coeficientes cepstrales de frecuencia Mel (MFCC, para ayudar a predecir esta enfermedad por medio de la señal transmitida a partir de la voz de un paciente).

Lo que se pretende con el proyecto en curso, es utilizar el conjunto de datos que se obtuvo a partir del estudio mencionado, utilizando otros modelos que no se han empleado en la predicción, como las redes neuronales y los bosques aleatorios, ya que a partir de ellos se podrían encontrar mejores resultados a nivel de preprocesamiento de datos de pacientes.

1. PLANTEAMIENTO DEL PROBLEMA

1.1. DESCRIPCIÓN DEL PROBLEMA

El Parkinson es una enfermedad neurodegenerativa que impide la perfecta armonía del sistema nervioso del cuerpo humano, es la segunda de su clase después del Alzheimer y se puede presentar en personas mayores de 60 años; además, que no existe una cura para esta enfermedad, únicamente puede ser tratada. La EP se caracteriza por la pérdida de una sustancia química conocida como Dopamina, esta sustancia es producida por las células nerviosas. Los síntomas más comunes de esta enfermedad son el temblor en manos, brazos, piernas, cambios en el habla, el equilibrio y la coordinación (Balestrino & Schapira, 2020).

En algunos casos la EP es tratada con medicamentos para contrarrestar los efectos de la enfermedad, pero aun así no es suficiente, ocasionando que las personas gasten su dinero intentando recuperarse. Algunas investigaciones arrojan que aproximadamente el 90% de los pacientes presentan problemas de habla (comunicación), como consecuencia las personas son discriminadas por esta condición. Para sintetizar, un gran porcentaje tendrá esta enfermedad en algún momento de su vida sin saberlo, bien sea porque alguien de su familia la tuvo o por sus condiciones físicas (Sakar et al., 2019).

El machine Learning o aprendizaje automático se utiliza para que, a partir de ciertos datos, basados en estudios o situaciones, las máquinas o sistemas

son capaces de ser entrenadas, y a partir de este proceso, se logre clasificar si ocurre un fenómeno con determinado resultado. Estas técnicas se han usado para ayudar a predecir, por ejemplo, el resultado de un tratamiento contra el virus del Papiloma humano (Rossum & Boer, 1991). En el artículo que se mencionará más adelante se realizó un estudio sobre la EP, el cual está basado en temas de modelos matemáticos con los cuales se obtuvieron datos que son relevantes en el desarrollo de la detección de la EP.

Sería ideal diseñar algún programa inteligente capaz de identificar a una temprana edad (a partir de los 20 años), si una persona podría desarrollar la EP y así tendría tiempo para prevenir que entrada a la tercera edad más o menos a los 60 años, no tenga este problema que afecte su diario vivir, que pueda resultar ser difícil de tratar y de tener una buena evolución con su enfermedad. Por lo tanto, la investigación se basa en realizar un entrenamiento en modelos que tiene como fin principal, predecir si en algún momento una persona puede o no desarrollar la enfermedad de Parkinson.

1.2. FORMULACIÓN DEL PROBLEMA

Las enfermedades neurodegenerativas afectan drásticamente al ser humano sin distinción, es decir, no importa si es hombre o mujer, en ocasiones no es claro por qué a una persona en su vida le apareció una enfermedad bastante conocida en el mundo como lo es el Parkinson. Es aquí donde surge la pregunta a partir de la evaluación de dos algoritmos ¿Cuál modelo podrá predecir la enfermedad del Parkinson utilizando aprendizaje automático, a

partir de la evaluación de al menos dos modelos, y obtener resultados con mayor precisión que en estudios previos?

1.3. JUSTIFICACIÓN

La EP es un problema que puede afectarnos a todos en algún momento de la vida, es por eso que se necesita tener como bien podría ser “un as bajo la manga” con el cual defendernos, es por ello que se propone el entrenamiento de modelos que permitan al ser humano tener una idea clara de cómo ayudar a cualquier tipo de persona, que en cualquier momento de su vida lleguen a presentar la enfermedad.

Se hablaría de un alivio económico a mediano y largo plazo, tanto para los pacientes como para sus familiares, quienes llevarían toda la carga del coste de la enfermedad y de los posibles, y rigurosos tratamientos a los que son sometidos. Por lo tanto, esos recursos serán invertidos en necesidades que llegasen a tener en el futuro.

De encontrarse el modelo adecuado, a partir de datos sobre la salud de un paciente, podrían las entidades promotoras de salud (EPS) tener un ahorro significativo en la parte económica debido a que no se tendría que realizar un tratamiento tan estricto para un paciente, sino más bien este tendría una menor duración y un menor costo invertido tanto por el personal de salud como por la persona que tiene la enfermedad.

Haciendo uso de la tecnología y del aprendizaje automático por parte de las variables de riesgo a analizar, se estudió un modelo automatizado para predecir la enfermedad, la cantidad de datos de pacientes no influye ya que se espera que el servidor donde corra el programa sea de una capacidad alta para que la cantidad de datos que se va a entrenar no tenga problemas en dar los resultados necesarios para esta evaluación.

Con el desarrollo se aprendió sobre tecnologías que están a la vanguardia de la inteligencia artificial, en este caso sobre el aprendizaje automático, la inserción de conocimiento de nuevos lenguajes como lo es Python, a relacionar la Ingeniería de Sistemas y Computación con otras ciencias interdisciplinarias como el estudio del cuerpo humano; por otra parte, también a desarrollar una mejor formar de pensar y opinar sobre temas de la actualidad.

1.4. OBJETIVOS

1.4.1. Objetivo general

Evaluar modelos de aprendizaje automático para predecir diagnósticos de la enfermedad de Parkinson a partir de datos de pacientes recopilados, utilizando técnicas de aprendizaje y parametrizaciones no contempladas sobre este conjunto de datos anteriormente.

1.4.2. Objetivos específicos

- Pre procesar los datos como paso de entrenamiento previo, para clasificar y unificar la información.
- Implementar modelos de predicción tales como redes neuronales y bosques aleatorios en la predicción de Parkinson, para observar el comportamiento de los datos.
- Evaluar los resultados obtenidos por cada uno de los modelos por medio de una matriz de confusión que ayude a visualizar su desempeño.

1.5. ALCANCE Y LIMITACIONES DEL PROYECTO

1.5.1. Alcance

A partir de un conjunto de datos obtenidos en la investigación mencionada previamente, se pretende realizar el diseño y evaluación de dos modelos de aprendizaje automático, para predecir el riesgo de desarrollar la enfermedad de Parkinson con los datos clínicos recopilados en el artículo "*A comparative analysis of speech signal processing algorithms for Parkinson's disease classification and the use of the tunable Q-factor wavelet transform. Applied Soft Computing Journal*" (Sakar et al., 2019).

Se va a hacer el entrenamiento de los dos modelos con el conjunto de datos, utilizando el lenguaje de programación Python y *Scikit learn*, que es una librería compuesta por algoritmos que ayudan a realizar la clasificación, análisis y entrenamiento de datos para hacer predicciones basado en la ciencia de los datos (Molina Ríos et al., 2016).

1.5.2. Limitaciones

Ya que se tiene un dataset con datos ya recopilados, para esta investigación no se tendrá en cuenta información de investigaciones adicionales a ese conjunto de datos a entrenar.

2. MARCO DE REFERENCIA

2.1 MARCO TEORICO

2.1.1. El Parkinson (EP)

Se caracteriza por el movimiento constante de manera involuntaria que afecta directamente el sistema nervioso, dificultad al expresar ideas de manera oral y/o escrita; esta alteración en el sistema nervioso no solo es por cosas del azar como se cree, sino por factores genéticos que pueden repercutir en el desarrollo de la EP (Balestrino & Schapira, 2020).

Desde épocas pasadas, alrededor del siglo XIX, cuando James Parkinson un doctor en la ciudad de Londres desarrolló un pequeño estudio sobre un síndrome que causaba un movimiento atípico en el cuerpo del ser humano. No fue sino hasta después de 60 años que Jean Martin Charcot tuvo en cuenta a James, encontrando dos síntomas: la rigidez y los temblores; luego Guillermo Gowers logró establecer que esta enfermedad ataca más a los hombres que a las mujeres. Entre los años 50´s y 60's del siglo XX se evidencia que esta particularidad en el cerebro ocurre de una manera más agresiva en el encéfalo causando niveles bajos de Dopamina; se crean algunos medicamentos que ayudan a controlar este mal, pero no en su totalidad, dejando secuelas en la persona pese a que la enfermedad no es mortal (Tagle, s/f). Es por todo esto que los investigadores alrededor del mundo han intentado encontrar con ayuda del Aprendizaje automático una solución que permitiera controlar la presencia de la enfermedad en las personas.

2.1.2. Inteligencia artificial (IA)

Es un concepto amplio, se refiere al intento por conseguir que una máquina o un sistema pudiera imitar la inteligencia humana. Pero, quien verdaderamente fue el padre de este majestuoso término fue Alan Turing hacia el año de 1950, donde publica en un artículo una pregunta que cambiaría la forma de pensar de muchos: ¿pueden las máquinas pensar? (Berzal, s/f).

La inteligencia artificial es importante ya que ayuda a realizar un proceso de aprendizaje de manera automatizada y descubrimientos de forma repetitiva con el uso de datos. Ayuda a realizar tareas que son frecuentes haciendo uso del computador y de una gran cantidad de información, aunque para esto la intervención humana sigue siendo fundamental porque es quien realiza la adecuada conformación del sistema y formulando adecuadamente las preguntas acordes a la problemática a solucionar con ayuda de esta rama de la ciencia de la computación.

La IA hace que un producto logre aprender (poseer un nivel de inteligencia),es decir estos productos sufrirán un cambio para que funcionen de una mejor manera haciendo una combinación con una cantidad amplia de datos logrando así mejorar la tecnología de una vivienda y la seguridad del sitio por ejemplo, esto se logra a través del entrenamiento de modelos de aprendizaje quienes realizan una predicción a partir de datos sobre un fenómeno, también esto es posible gracias al uso de redes neuronales que

entre más se usen así mismo aprenderán; otra de las áreas de la IA es la medicina allí emplean técnicas de agrupación de imágenes e inspección de objetos que logran ser usadas para detectar el cáncer de manera que pude resultar muy confiable como si se tratase un radiólogo el que realiza el procedimiento (Adamssen, 2020)

A partir de estos temas se desglosan un sinfín de términos relacionados que ayudarán a entender un poco más cómo el aprendizaje automático se ve involucrado al momento de predecir la aparición de problemas bilógicos en las personas.

2.1.3. Aprendizaje automático

Se refiere al desarrollo de sistemas que aprenden, estos se alimentan de datos útiles que son proporcionados de forma no supervisada, es decir, este aprendizaje es como enseñarle a un niño acciones y tareas que debe ir conociendo e identificando al pasar el tiempo(Oracle, s/f). Este término en el año de 1959, cuando Arthur Samuel quien fue pionero en los juegos de computadora, acuñó el término en la búsqueda de lo que hoy en día es la inteligencia artificial.

El tema de aprendizaje tiene que ver con el almacenamiento de nuevo conocimiento, el desarrollo de habilidades, organización de ideas y el descubrimiento, para resolver este problema se ha decidido optar por hacer uso de las computadoras logrando así que estas puedan tener las capacidades de aprendizaje automático en términos de modelado y conocimiento de fenómenos. Esto gira sobre tres ejes que son los estudios

orientados a las tareas, la simulación cognoscitiva y el análisis teórico (Gramajo et al., s/f).

2.1.4. Modelos de clasificación

Se denominan modelos de clasificación a todo análisis capaz de predecir si se puede dar un resultado positivo o negativo, dependiendo el caso y el contexto desde donde se tome (Estevez, s/f). Existen diferentes formas de clasificar los datos siniestrados, dentro de las cuales podemos mencionar a las redes neuronales y los bosques aleatorios; y es por eso que existen lenguajes de programación que ayudan a que esta tarea sea más sencilla, un ejemplo significativo es el caso de Python.

Para llegar a este punto se debe de entender los datos que se tienen, se debe preparar (realizar la respectiva reducción si es el caso) y luego se selecciona el modelo más acorde para poder realizar la clasificación; también se define que variable resulta más apropiada para el modelo, luego de esto se puede ir definiendo que modelo a usar, en este caso puede ser redes neuronales y bosques aleatorios para aprendizaje de manera supervisada (Solutions, 2018).

- **Redes neuronales**

Las redes neuronales artificiales nacen a partir de la idea de que una máquina (en este caso un computador) se asemeje mucho a la capacidad con que cuenta el cerebro humano, esta se compone por un cuerpo o soma,

un axón que es la conexión hacia otra neurona y sinapsis que es una ramificación, por medio de las dendritas(otra ramificación en forma de árbol) se recibe la señal que es procesada y ocurre una salida a través del axón, así mismo se esperaba que fuera un nivel de computación, se utilizan términos numéricos y probabilísticos que determinan la dirección de una neurona a otra (Alonso Romero & Calonge Cano, s/f).

Las redes neuronales en computación funcionan de manera similar a las que tienen los animales, estas ayudan a comunicación interna de una conexión de neuronas (red neuronal) que ayuda a entender lo percibido en un instante imágenes o sonidos por ejemplo. En un sistema estas redes lo que hacen es aprender de manera supervisada (donde se tienen unos datos y se sabe cuál puede ser el resultado del entrenamiento) y no supervisada (donde se tienen unos datos y no se sabe cuál puede ser el resultado del entrenamiento) (Pardo, 2020).

- **Bosques aleatorios**

Es una manera de hacer aprendizaje supervisado, combina varios árboles de decisión causando un bosque aleatorio, realiza una predicción estable mayormente confiable dado que separa los datos en un subconjunto de datos donde están presentes características aleatorias y de ahí se tendrá la mejor opción para el modelo (IA, s/f).

2.1.5. Python

El lenguaje de programación denominado Python, es aquel software libre, amigable con el usuario, es decir, es práctico y sencillo de entender; con una gran cantidad de aplicativos en el mundo de la programación (Holguín et al., 2014). Por lo tanto, se decide que el estudio sea realizado en este lenguaje, porque se sabe que una de sus librerías está enfocada en realizar este tipo de análisis e investigaciones.

- **Scikit learn**

La librería Scikit learn del lenguaje de programación Python, está enfocada en realización de análisis a través de algoritmos de aprendizaje estadístico predictivo (Scikit-Learn: Aprendizaje Automático En Python - Documentación de Scikit-Learn 0.23.2, n.d.). Es en esta librería que se realizó la evaluación de los datos obtenidos y básicamente los resultados que arroje son de vital importancia para la investigación que se realizó sobre la enfermedad del Parkinson.

2.1.6 Principal Components Analysis (PCA)

El análisis de componentes principales**,** es el procedimiento usado en estadística, el cual permite reducir los espacios muéstrales, también con gran cantidad de dimensiones presentes en una muestra donde se conserva la información contenida. Es un método de aprendizaje no supervisado**,** el objetivo no es predecir como ta**l** sino extraer la información e identificar subgrupos en un conjunto de datos (Molina Ríos et al., 2016) (Joaquín Amat Rodrigo, 2017).

2.1.7 Recursive Feature Elimination

La eliminación de características recursivas, es un algoritmo el cual permite la selección de ciertas características (columnas), en un conjunto de datos que va a hacer entrenado y son mayormente relevantes, para ayudar a predecir la variable fin, además el hacer esto ayuda a manejar de manera más adecuada los datos y tiempo de ejecución (Jason Brownlee, 2020).

2.1.8 LinearSVC

Es un clasificador de vectores de soporte lineal el cual ayuda a la caracterización de los datos, después de esto se podría hacer un envió de características al clasificador para ver cuál es la clase a predecir (Pythonprogramming.net, s/f).

Algunos de los parámetros a tener en cuenta son:

- **Penalty:** es la norma utilizada para la penalización. La penalización 'l2' es un estándar utilizado en vectores de soporte lineal, mientras que la penalización 'l1' conduce a un coeficiente de vectores que son escasos.
- **Dual:** “True:”, se selecciona el algoritmo a optimizar. Se prefiere False cuando n_samples> n_features.
- **C:** Es un parámetro de regularización que debe ser positivo, esta evaluado entre 0 y 1.

- **fit_intercept:** True, se utiliza para calcular la intersección del modelo. Si se manejara como falso no habrá intersección para los cálculos (datos centrados).

2.1.9 Variables ordinales

Son variables donde existe un orden, estas expresan cualidades, pero estas variables no contienen números (José Francisco López, s/f-b).

2.1.10 Variables nominales

Son variables donde no existe un orden, estas expresan cualidades categóricas, pueden contener números (José Francisco López, s/f-a).

2.1.11 One hot encoding

Es una manera de representar variables categóricas en forma de vectores binarios, es decir, los valores categóricos primero tendrán valores enteros y estos se representan con valores de 0 y 1 de acuerdo a lo que se quiera representar (Jason Brownlee, 2017).

2.1.12 Feature selection

Se utiliza para la elección de características y reducción de las dimensiones en conjuntos de datos (muestras), para mejorar el valor de precisión de estimadores de un dataset muy grande. Existen varias maneras de llevar a cabo este proceso como lo es el método de eliminación de características de baja variación o el RFE (scickit learn, s/f-a).

2.1.13 Gridsearchcv

Es una clase disponible en Scikit-learn, que le permite evaluar y seleccionar sistemáticamente los parámetros del modelo. Al proporcionarle el modelo y los parámetros a probar, puede evaluar el desempeño del primero en función del desempeño del segundo a través de la validación cruzada. Esto permite seleccionar los parámetros más apropiados para ese modelo (Daniel Rodríguez, 2018a).

2.1.14 Matriz de confusión

Es una manera para ver el rendimiento de un algoritmo en aprendizaje supervisado, permite observar los aciertos que ha tuvo el modelo. Existen cuatro opciones que pueden ocurrir en relación al rendimiento de los datos: verdaderos positivos (VP), verdaderos negativos (VN), falsos negativos (FN), falsos positivos (FP).

Las métricas contenidas son:

- **Exactitud (accuracy):** se refiere a que tan cerca está el valor obtenido en una medición del dato verdadero.
- **Precisión (precision):** realiza mediciones repetitivas de datos, se obtiene una dispersión, donde entre menor dispersión exista, mayor es la precisión.
- **Sensibilidad (Recall o sensibility):** es la cantidad de casos positivos que el modelo identificó que son ciertos.
- **Especificidad (Especifity):** cantidad de datos que el modelo identificó que realmente son negativos.
- **F1 Score**: tiene en cuenta la precisión y sensibilidad como una sola métrica, ayuda a detectar datos que no son realmente ciertos. Es ampliamente usado en dataset que presentan desequilibrio (Juan Ignacio Barrios Arce, s/f).

2.2 ESTADO DEL ARTE

El estudio denominado “A comparative analysis of speech signal processing algorithms for Parkinson’s disease classification and the use of the tunable Q-factor wavelet transform”, se realizó con un grupo de un poco más de 180 personas donde se tuvo en cuenta la frecuencia de su voz; en él se puede observar que se menciona que a partir de este factor en la comunicación, este puede ser relevante para predecir la enfermedad, los autores de este artículo utilizaron algoritmos de redes neuronales, de recesión y arboles decisión, en donde se obtuvo una tasa de precisión superior al 95%; el entrenamiento de algoritmos realizado en esta investigación fue basado en aprendizaje de árboles de decisiones y de entrenamiento de instancias (Sakar et al., 2019).

Siguiendo con la búsqueda de más información, se encontró otro artículo muy parecido; el estudio fue realizado para un trabajo de grado sobre la EP en el 2016, denominado “Detección automática del grado de Parkinson a partir de la señal de voz”, para los estudios de una serie de características encontradas en cada uno de los 50 pacientes que participaron en la investigación; se utilizó un software llamado WEKA que es gratuito para realizar entrenamiento de datos y se usaron arboles de decisión; se extrajeron datos de gran relevancia para ayudar en la detección temprana y adicionalmente no se tuvo que utilizar una gran cantidad de datos, puesto que al analizar las características a partir de ciertos datos basado en el aprendizaje automático se logró una clasificación adecuada, a través del ruido que transmite la voz de una persona y con solo un 8% de análisis de características a entrenar se conocieron resultados prometedores en cómo detectar a tiempo esta enfermedad (Piñeiro, s/f). Aunque esta investigación es muy parecida con el estudio anterior, las diferencias son amplias dentro de lo que se puede destacar está el tiempo de ejecución de ambos estudios, el nivel de granularidad con que realizaron las pruebas, las herramientas utilizadas para la investigación y la cantidad de información empleada para garantizar que cada uno de los estudios podía predecir o detectar la EP.

Otro de los artículos hallados, indagando en la web, es el artículo denominado “Estudio y selección de las técnicas de Inteligencia Artificial para el diagnóstico de enfermedades”, el cual aborda un estudio de caracterización realizado a 20 especialistas en temas de Medicina. En él se les menciona algunas técnicas para realizar un adecuado tratamiento de una enfermedad como por ejemplo, las redes bayesianas, las redes neuronales artificiales y el razonamiento basado en casos; el método que se uso fue la Teoría de la Decisión Multicriterio Discreta (DMD), ayudando a describir el

mejor método para ayudar a la predicción de enfermedades, se encontró que basado en el análisis de las respuestas obtenidas en la clasificación, según los especialistas, la mejor opción son las redes bayesianas, ya que están ofrecen mayor capacidad de detección temprana debido a la facilidad de manejo de datos para realizar modelos de clasificación, adicionalmente por su almacenamiento en una base de datos se obtiene un razonamiento más acorde para desarrollar una predicción con alta efectividad(Neilys González Benítez, Vivian Estrada Sentí, s/f). Entendiendo que los autores decidieron escoger diferentes modelos de IA para diagnosticar enfermedades entre ellas el Parkinson, nuestro estudio base se centró en encontrar factores que se producían por medio de las señales de la frecuencia vocal.

Los artículos anteriormente mencionados se diferencian del trabajo de investigación, porque lo que se pretende con la investigación, es utilizar el conjunto de datos que se obtuvo a partir del estudio mencionado, utilizando otros modelos que no se han empleado en la predicción, como las redes neuronales y los bosques aleatorios, ya que a partir de ellos se pueden encontrar mejores resultados a nivel de preprocesamiento de datos.

2.3 MARCO LEGAL

A continuación, se presenta un breve resumen de algunas normas que están directamente relacionadas con la investigación que se está realizando y que se respetarán en el presente trabajo:

2.3.1 Habeas data

Por medio de ley establecida en la constitución colombiana 1581 de 2012 y el decreto 1377 de 2013, establecen que toda persona tiene derecho de conocer, borrar, actualizar y rectificar toda información personal recolectada en bases de datos o archivos. Si la información allí contenida es suministrada a terceros, el titular estará informado del tratamiento que se les den a los datos, mientras que el responsable de los datos, deberá tener una copia de lo que realiza con los datos, por si alguna vez el titular le solicita un reporte (Función Publica, s/f).

2.3.2 Derecho a la intimidad

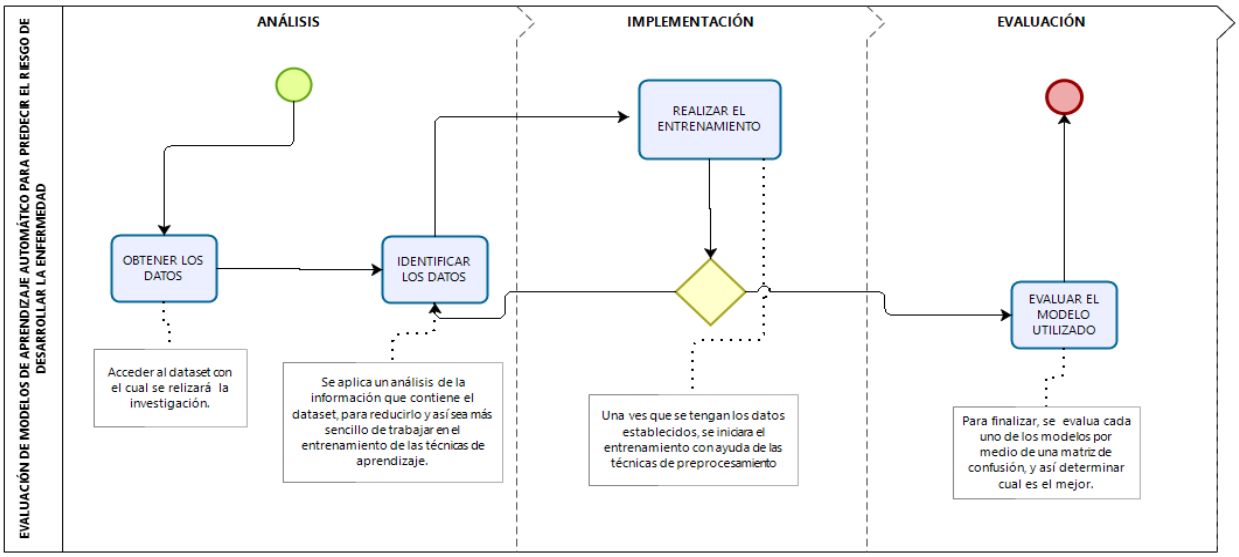
Según el artículo 15 de la constitución política de Colombia, se estipula que cualquier persona tiene el derecho a su intimidad personal, familiar y al buen nombre; por ende, el estado debe respetarlo. Además tienen el derecho de actualizar, consultar, sobrescribir la información de sí mismas recolectadas en bases de datos, que el estado podrá acceder a dicha información solo y únicamente con una orden judicial(Constitución Politica de Colombia, s/f).

3 METODOLOGÍA Y DESARROLLO

3.1. METODOLOGIA

El trabajo de investigación está principalmente en los resultados que se tienen a partir del entrenamiento realizado con ayuda de los modelos, para poder evaluar los datos obtenidos del artículo científico del que se habló en capítulos anteriores. Se utiliza la herramienta Bizagi *Modeler* para ilustrar de una manera sencilla la forma en que se realizará la obtención de los resultados, como se puede observar en la ilustración 1.

Ilustración 1



Fuente: elaboración propia.

3.1.1. Análisis de datos

En esta primera sección se abordarán todas las actividades concernientes a fin de encontrar los datos necesarios para realizar la predicción de la enfermedad del Parkinson, por tal motivo se nombran las actividades a ejecutar para este proceso:

- Obtener el conjunto de datos para hacer el desarrollo del proyecto de Investigación.
- Identificar si la información que se extrajo es clara para realizar el estudio de los datos.
- Evidenciar en el dataset las situaciones de cada paciente; es decir, ver qué se espera de la información que se tiene, en cuanto a su predicción.

Los entregables dispuestos para esta sección son:

- Clasificación de los datos para el análisis.
- Agrupación de información de los pacientes.

3.1.2. Implementar los modelos

En esta sección de la metodología, a partir de la información del conjunto de datos y de los modelos seleccionados para realizar la predicción a temprana edad de la sintomatología asociada al Parkinson, se empezará a realizar el estudio de los mismos, basándose en la información que se tiene de manera previa sobre pacientes que podrían enfrentar esta novedad en su salud. Las actividades a realizar son las siguientes:

- Aplicar los modelos adecuados para realizar la clasificación
- Continuar con el aprendizaje de modelos para tener resultados a corto y/o mediano plazo.

Los entregables de esta etapa serán:

- Mostrar el resultado de los entrenamientos de los modelos de clasificación, para observar su comportamiento con los datos.
- Informar sobre lo encontrado hasta el momento con base en los modelos seleccionados.

3.1.3. Evaluar resultados

Para este proceso, lo que se tiene en cuenta es que el entrenamiento de los modelos a partir de la información que se tiene haya sido desarrollado de una manera lo más precisa posible, para brindar una adecuada evaluación de los resultados, siendo capaces de lograr el impacto que se quiere en cuanto a los tiempos de detección oportuna y de resultados. Las actividades a realizar son las siguientes:

- Clasificación de pacientes.
- Realizar análisis al implementar los modelos.
- Reportar ventajas y desventajas de este entrenamiento de datos clínicos.

Los entregables de esta etapa serán:

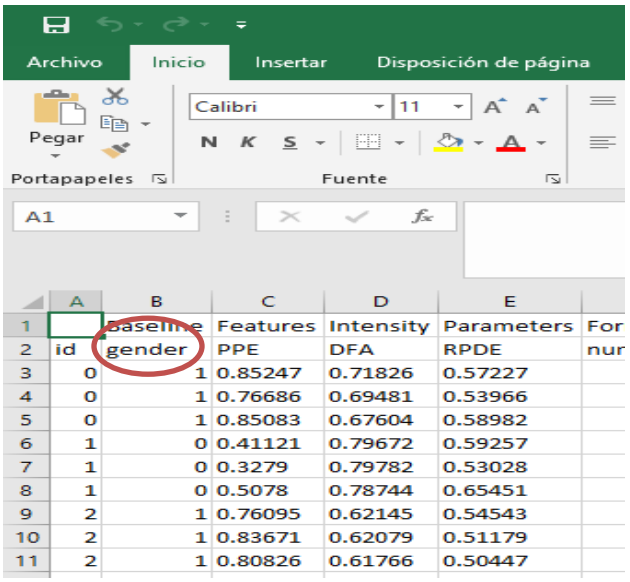
- Informar sobre los pacientes que puede desarrollar o no la enfermedad.
- Mostrar por medio de una matriz confusión los resultados obtenidos de cada modelo.
- Comunicar cuál es el mejor modelo para realizar la predicción.

4. DESARROLLO

Para desarrollar todo lo propuesto en la metodología, es importante entender lo siguiente, el dataset que fue suministrado para la realización de la investigación, contaba con exactamente 755 columnas, el total de pacientes que participaron en el dataset es de 180, esto nos da alrededor de 136.000 características a analizar.

En este caso se analiza el dataset para diferenciar qué datos se encuentran en variables cualitativas y cuáles en cuantitativas, esto con el fin de tener toda la información del estudio unificada con solo variables cuantificables; al verificar esta información dentro del dataset, se encontró que una las columnas (columna “gender” como se muestra en la ilustración 2), posee una variable nominal, es decir, dicha variable no puede expresarse como un tipo de medida o cantidad cuantitativa, sino que esta expresada en términos de una variable categórica (nominal).

Ilustración 2



	A	B	C	D	E
1		Baseline	Features	Intensity	Parameters
2	id	gender	PPE	DFA	RPDE
3	0	1	0.85247	0.71826	0.57227
4	0	1	0.76686	0.69481	0.53966
5	0	1	0.85083	0.67604	0.58982
6	1	0	0.41121	0.79672	0.59257
7	1	0	0.3279	0.79782	0.53028
8	1	0	0.5078	0.78744	0.65451
9	2	1	0.76095	0.62145	0.54543
10	2	1	0.83671	0.62079	0.51179
11	2	1	0.80826	0.61766	0.50447

Fuente: raw_data.csv

Una vez realizado el análisis anterior, se procede a reducir el tamaño de los datos, con el fin de obtener un número más pequeño deseado de características o componentes principales; para lograr esta reducción dentro del dataset con el que se encuentra trabajando, se necesitan de técnicas tales como RFE (eliminación de características recursivas), esta técnica tiene como objetivo seleccionar un conjunto inicial de características para que por medio de un estimador se entrene el conjunto inicial en donde las características menos importantes se eliminan hasta reducir el conjunto donde se encuentran el número deseado de características a seleccionar (la ilustración 3, muestra el código utilizado para realizar la reducción de dimensionalidad) (Jason Brownlee, 2020).

Ilustración 3

```
from sklearn.base import BaseEstimator, TransformerMixin
from yellowbrick.model_selection import RFECV

class CustomRFE(TransformerMixin, BaseEstimator):

  def __init__(self, step=5, cv=3):
      self.model=None
      self.step = step
      self.cv = cv
      self.x_new = None

  def fit(self, X, y):
      estimador = LogisticRegression(C=0.1)
      estimador.fit(X,y)
      self.model = RFECV(estimador, step=self.step, cv=self.cv)
      self.model.fit(X, y)
      return self

  def transform(self, X):
      global dim
      self.x_new=self.model.transform(X)
```

Fuente: creación propia (Google Colab)

Otra de las técnicas que se emplearon fue el PCA (análisis de componentes principales), está centrado en la formación de un conjunto de datos no correlacionado, se emplea principalmente para analizar datos y la construcción de modelos predictivos (Galarnyk, s/f). Esta técnica la podemos encontrar alojada en la librería de Scikit learn y podemos hacer uso de ella importándola directamente en el código fuente (la ilustración 4, muestra la librería de donde podemos obtener la técnica PCA).

Ilustración 4

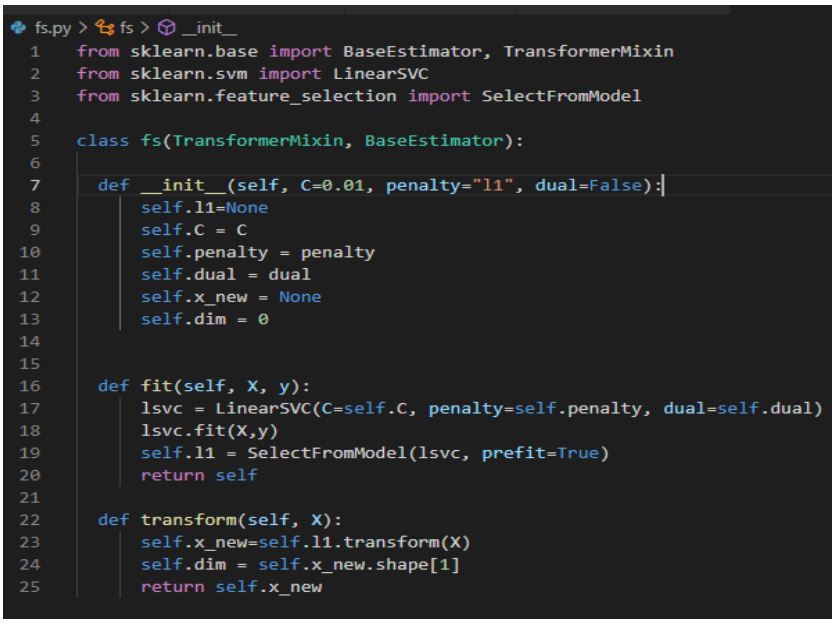
```
from sklearn.decomposition import PCA
```

Fuente: librería Scikit Learn

LinearSVC (clasificador de soportes linear), es otra de las técnicas utilizadas para la reducción de dimensionalidad del conjunto de datos inicial, se caracteriza por adecuarse a los datos que se le proporcionan que a su vez devuelve un hiperplano

que divide o categoriza los datos (la ilustración 5, muestra el código que se utilizó para hacer uso del clasificador de soportes linear) (pythonprogramming.net, s/f).

Ilustración 5



Fuente: creación propia (Visual Studio Code)

Dentro de la investigación se emplea cada uno de las técnicas de reducción de dimensionalidad descritas anteriormente, en algunos casos se utilizaron dentro de los modelos más de una técnica debido a que la dimensionalidad obtenida era bastante considerable y exigía que el entrenamiento de los modelos se prolongara en cuanto a tiempo de ejecución.

Después de reducir considerablemente la dimensión del dataset, se procede a diseñar los modelos de aprendizaje automático, que en este caso son redes neuronales y bosques aleatorios; ambas técnicas de clasificación capaces de ser entrenadas con el dataset resultante.

5 RESULTADOS

Con lo mencionado en capítulos anteriores, se diseñan tres modelos denominados “oh_l1fs”, “pca_rfe_nn” y “pca_rfe_rf”, cada uno de ellos estructurado de manera secuencial para ejecutarse, los resultados obtenidos por los modelos se presentan a continuación:

- **OH_L1FS**

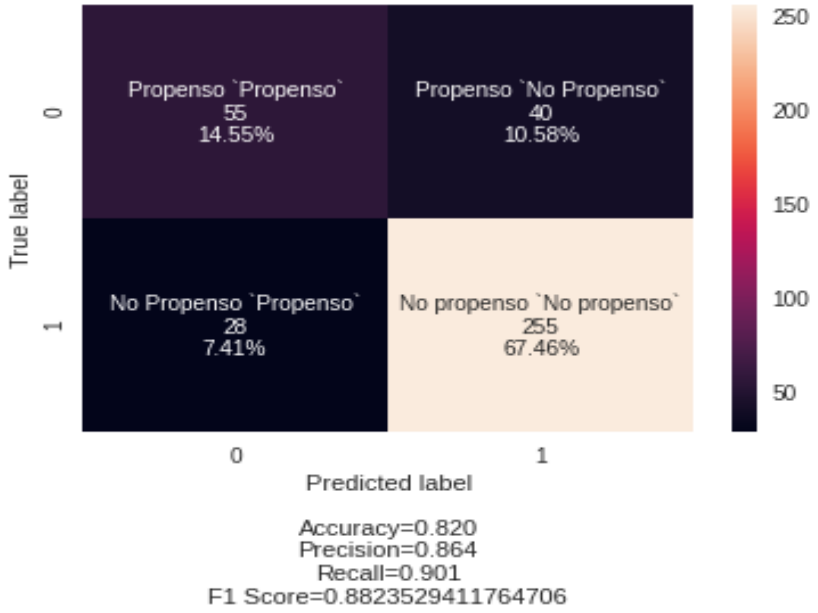
  El siguiente modelo se basa primeramente en eliminar la columna “id” del dataset inicial, esta columna no representa ninguna característica importante a estudiar; una vez eliminada el one hot encoding se aplica a la columna “gender” (genero), para convertir esa variable nominal a números binarios.

  Como paso a seguir se ejecuta la técnica de preprocesamiento que se encuentra alojada en la librería de Scikit learn utilizando el método de standardScaler, con el fin de realizar una estandarización de los datos y así determinar que la varianza sea igual a 1.

  El siguiente paso es aplicar el feature selection, en este caso se implementó el método de clasificación de soporte de vectores y con un parámetro de regularización “C” igual a 0.01. El siguiente paso es aplicar la red neuronal con el método gridsearchcv para encontrar los parámetros considerados los más óptimos.
  El resultado obtenido de este proceso se ve generado ya sea por un archivo .csv o una lista de métricas de crossvalidation que ayudaran en la evaluación del modelo.

La siguiente ilustración nos determina las métricas que son evaluadas por el modelo, por medio de una matriz de confusión en donde se visualiza el desempeño del algoritmo y su tasa de error,

Ilustración 6



Fuente: creación propia (Colab)

La matriz nos muestra que el conjunto datos entrenado se encuentra desbalanceado, es decir, la cantidad de Falsos negativos y de falsos positivos no es igual, debido a que muchos de los datos que deberían ser propensos a desarrollar la enfermedad, no lo son. Otra forma de ver cuando los datos están balanceados es cuando la tonalidad de las casillas es igual.

De acuerdo a la predicción realizada con el modelo, se determina que:

- los pacientes propensos a sufrir la enfermedad, pueden desarrollarla con un porcentaje del 14.55%.

- los pacientes propensos a sufrir la enfermedad, no la desarrollaran con un porcentaje del 10.58%.
- los pacientes no propensos a sufrir la enfermedad, la desarrollaran con un porcentaje del 7.41%
- los pacientes no propensos a sufrir la enfermedad, no la desarrollaran poniéndolos como el porcentaje más alto. Equivalente al 67.40%.

Se considera entonces a la métrica F1 Score encargada de resumir la precision y sensibilidad del modelo, que en este caso tiene una tasa 88.23%.

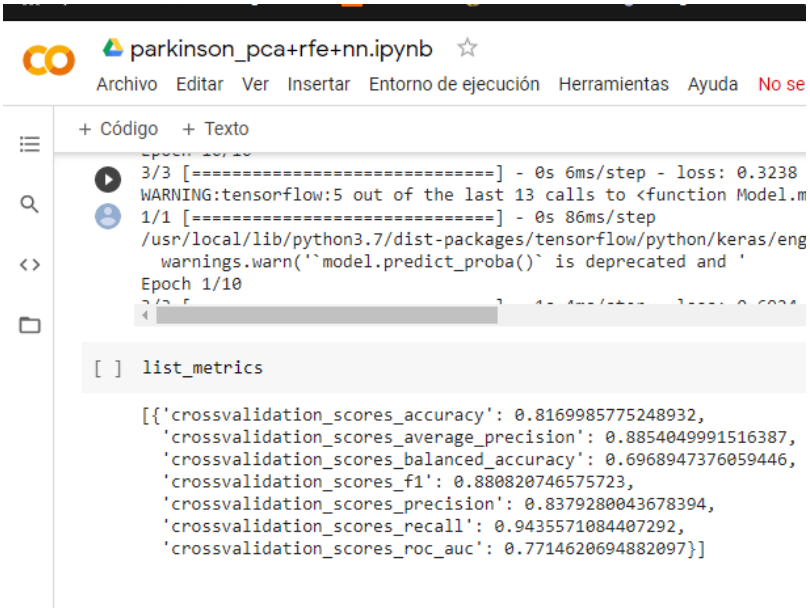
- **PCA_RFE_NN**

El segundo modelo comienza con la eliminación de la columna “id” y conversión de la columna “gender” a números binarios, seguido se implementa la librería de Scikit Learn de preprocesamiento para utilizar el método de StandardScaler para realizar una estandarización de los datos y así determinar que la varianza sea igual a 1.

Se utiliza la librería de scikit learn que importara el método PCA con el objetivo de analizar los componentes principales y así reducir la dimensión del dataset, debido a que esta reducción de componentes no es lo bastante considerable, se emplea otra técnica de reducción de dimensionalidad que es el RFE, el cual se encarga de eliminar las características recursivas, además de lograr reducir el consumo de los recursos como el tiempo de ejecución del entrenamiento del modelo.

Después se ejecuta la red neuronal junto con el método gridsearchcv, para intentar encontrar las variables óptimas del conjunto de datos. El resultado que se obtiene es un archivo con extensión .csv o una tabla de validación cruzada donde se observan datos relevantes del modelo.

Ilustración 7



Fuente: creación propia (Colab)

La ilustración 7 muestra una lista de métricas, que ayudaron a evaluar el modelo resultante, de estas métricas se toma como base: “presicion” y ”balance_accuracy” para determinar la precisión del modelo, y se obtiene que el porcentaje para este modelo es del 73.74%.

- **PCA_RFE_RF**

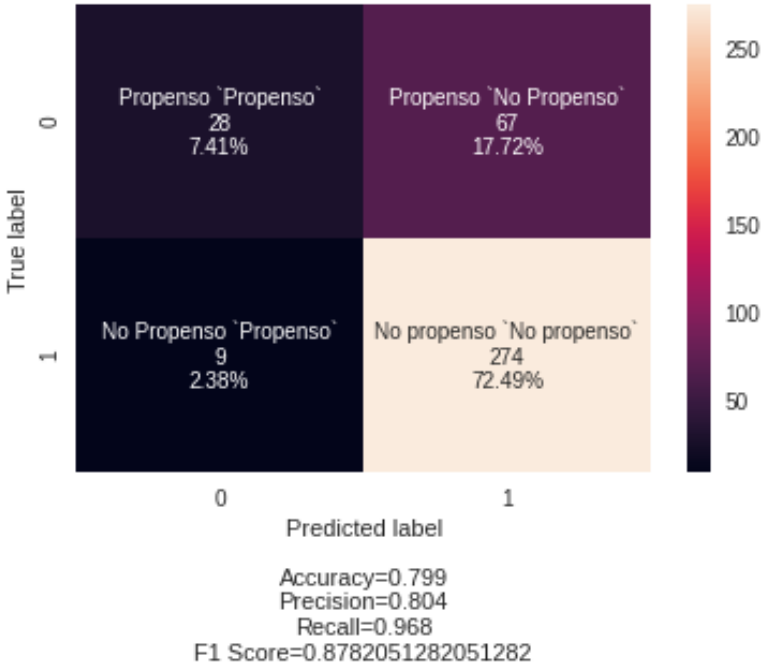
El tercer y último modelo esta secuencialmente distribuido para ejecutarse de la siguiente manera: primero los valores de la columna “gender” tendrán

una conversión a números binarios y la columna “id” será eliminada, segundo se utiliza StandarScaler con el fin de estandarizar los datos para que la varianza sea igual a 1.

Se utiliza la librería de Scikit Learn para importar el método PCA, el cual tiene como objetivo analizar los componentes principales ubicados en el dataset para reducir los datos, después se utiliza el método RFE para reducir aún más la dimensión de los datos, así mismo sucede con el modelo anterior, esto permite que el entrenamiento de los datos no consuma tantos recursos ni tanto tiempo al ejecutarse.

Por último se importa de la librería de Scikit Learn el método de RandomforestClassifier, que es un clasificador de bosques aleatorios que busca ajustar varios clasificadores de árboles de decisión en varias submuestras del conjunto de datos que se esté manejando, lo que ayuda a mejorar la precisión predictiva el resultado obtenido; de este proceso se obtiene la matriz de confusión que se observa en la ilustración 8.

Ilustración 8



Fuente: creación propia (Colab)

La matriz de la ilustración 8 muestra que el conjunto de datos estudiado se encuentra desbalanceado, es decir, el número de falsos ya sean positivos o negativos no se encuentran en equilibrio, debido a que muchos de los datos que deberían ser propensos a desarrollar la enfermedad no lo son, se determina que:

- los pacientes propensos a sufrir la enfermedad pueden desarrollarla con un porcentaje del 7.41%
- los pacientes propensos a sufrir la enfermedad, no la desarrollaran con un porcentaje del 17.72%
- los pacientes no propensos a sufrir la enfermedad, si la desarrollaran con un porcentaje del 2.38%
- los pacientes no propensos a sufrir la enfermedad, no la podrán desarrollar con un porcentaje del 72.49%.

Al igual que en el primer modelo de considera la métrica F1 Score, cuando la distribución de los datos es desigual, por lo cual se determina que la precisión y sensibilidad del modelo es del 87.88%.

Una vez obtenidos todos estos resultados, se puede decir que el mejor de los tres modelos descritos anteriormente es “oh_l1fs”, al tener una tasa de precisión del 88.23%, de acuerdo a lo anterior se determina que el modelo que mejor se comportó durante la investigación fue el de redes neuronales, también se puede concluir que la técnica de reducción de dimensionalidad es la más apropiada para esta investigación.

6 CONCLUSION Y RECOMENDACIONES

6.1 CONCLUSION

A partir de un análisis de métricas donde se puede decir que los datos se encuentran desbalanceados, como lo es la métrica de “roc auc” y la métrica “balanced accuraccy” en cada uno de los modelos propuestos en el capítulo anterior; el mejor resultado lo dará el entrenamiento basado en redes neuronales que en este caso sería superior al modelo de bosque aleatorios.

La reducción de datos ayuda a que los modelos consuman menos recursos y tiempo de ejecución en su entrenamiento. Utilizar la técnica de matriz de confusión genera los mejores resultados ya que se define una predicción de manera más correcta, debido a que se evalúa lo que realmente llega a pasar con un paciente.

Estos modelos de predicción complementarían la tarea de ayudar a predecir la EP a una edad temprana, siendo más tratable y de menor impacto en las personas que son diagnosticadas con esta enfermedad.

6.2 RECOMENDACIONES

Se recomienda que el código de entrenamiento sea ejecutado en Google Colab, debido a que al hacerlo de manera local se pueden tener dificultades ya sea con actualizaciones o con versiones de las librerías necesarias, si no se tienen las especificaciones adecuadas cuando se ejecuten los requerimientos para correr el modelo este empieza a generar error, por lo tanto, se recomienda para no tener estos inconvenientes hacer la ejecución en Colab.

En la carpeta compartida en Google drive se encontrará un execute de la herramienta Google Colab, en él se puede encontrar más a detalle los pasos que deben seguir para la ejecución de los modelos de aprendizaje automático.

Se debe de tener una conexión a Internet lo más estable posible para que la aplicación funcione de la mejor manera con Colab. Aunque no es recomendable probar el Software de manera local debido a las limitaciones que se pueden presentar, se debe de tener instalado Visual Studio Code o similar para poder realizar esa operación.

7 GLOSARIO

A continuación, se definen algunos métodos, librerías y términos utilizados durante el desarrollo del proyecto:

- **GOOGLE COLAB**: Es una herramienta para programar online en lenguaje de programación Python, donde ya se tiene todo lo relacionado a librerías y no se requiere mayor configuración para su uso, es muy útil para empezar con el tema de aprendizaje de inteligencia artificial (Collab, s/f).

- **KERAS IMPORT REGULARIZERS**: Los regularizadores ayudan a aplicar penalizaciones al momento de estar en proceso de optimización cierta capa de una red neuronal teniendo en cuenta por ejemplo el valor del peso en cierta parte de la red neuronal (The TensorFlow Authors, 2020).

- **KERAS.LAYERS IMPORT DENSE**: Es la función con la que se entrena por capas teniendo en cuenta datos relevantes como lo es el peso de los mismos para optimizarlos y dar un resultado (tuthorials point, 2021).

- **KERAS.LAYERS IMPORT DROPOUT**: El dropout permite realizar una regularización de los datos ya que permite reducir el sobreajuste que se pudiera presentar al momento de entrenar el modelo eliminando neuronas al azar (Christian Versloot, s/f).

- **KERAS.MODELS IMPORT SEQUENTIAL**: Es una API que se utiliza para realizar el entrenamiento de una red neuronal de manera secuencial es decir de acuerdo al modelo (stephen_mugisha, 2020).

- **KERAS.WRAPPERS.SCIKIT_LEARN IMPORT KERASCLASSIFIER**: Con esta API se implementa el clasificador de Scikit learn en la red neuronal (Keras) (TensorFlow, s/f).

- **MATPLOTLIB.PYPLOT AS PLT**: Estas funciones de pyplot sirve para realizar cambios en una figura en cuanto a trazados y decoración entre otras cosas, realiza un seguimiento de figuras (John Hunter, s/f).

- **NUMPY AS NP**: Es la biblioteca principal para realizar computación científica en el lenguaje de programación Python. Brinda una matriz de varias dimensiones y varias herramientas para trabajar con determinada matriz (Justin Johnson, s/f).

- **OS**: Este módulo nos permite acceder a funciones que dependen del sistema operativo. Lo más importante es que proporcionan información sobre su entorno y nos permiten manipular la estructura de directorios (Eugenia Bahit, 2013).

- **PANDAS AS PD**: Es una herramienta para análisis de datos la cual se utiliza en Python, aparte de lograr analizar diversos formatos de archivos adicionalmente puede convertir una tabla de datos en una matriz Numpy (Educative, 2021).

- **SKLEARN**: Es una biblioteca usada en Python la cual se usa para hacer aprendizaje automático, cuenta con varios algoritmos para este fin como es bosques aleatorios y máquinas de soporte vectorial (JournalDev, s/f).

- **SKLEARN.BASE IMPORT BASEESTIMATOR, TRANSFORMERMIXIN**: La parte de la clase estimador. Permite establecer y obtener parámetros para el mismo, es la clase que procede para los demás estimados en Python. El

transformador es un estimador que implementa una clase que herede ambos elementos (Vighnesh Birodkar, 2016).

- **SKLEARN.DECOMPOSITION IMPORT PCA**: Se importa para realizar el análisis de componentes principales (PCA),es decir se usa con el fin de comprender mejor el conjunto de datos y reducir la dimensionalidad de los mismos (Galarnyk, s/f).

- **SKLEARN.MODEL_SELECTION IMPORT CROSS_VAL_SCORE**: Con esta función se hace uso de la matriz de evaluación cruzada en el estimador y en el conjunto de datos, realizando la división de parte de entrenamiento y parte de experimento (Scikit learn, n.d.).

- **SKLEARN.MODEL_SELECTION IMPORT GRIDSEARCHCV**: Se utiliza para seleccionar y evaluar ciertos parámetros de un modelo, es decir se indica que se va a probar también se pueden evaluar de forma cruzada (Daniel Rodríguez, 2018b).

- **SKLEARN.MODEL_SELECTION IMPORT STRATIFIEDKFOLD**: Esta librería ayuda a preservar el balance que tengan los datos, por lo general la mayoría de conjuntos de datos están en desequilibrio y gracias a esto se puede informar al modelo la probabilidad de que ocurra o no cierto fenómeno (Analypeup, s/f).

- **SEABORN AS SNS**: Con esta biblioteca se tiene la opción de visualizar de una manera estadística y grafica información relacionado a determinados datos (Michael Waskom ., s/f).

- **SKLEARN.FEATURE_SELECTION IMPORT SELECTFROMMODEL**: Ayuda a realizar una adecuada caracterización en búsqueda de determinar

el dato más relevante, basado en un cálculo donde se tiene en cuenta el valor de la media (Cornellius Yudha Wijaya, 2021).

- **SKLEARN.METRICS IMPORT CONFUSION_MATRIX**: Con esta herramienta se tiene una matriz (tabla), a partir de la cual se puede evaluar el modelo de clasificación y se determina cuales casos son realmente acertados (positivos) y cuales son considerados incorrectos (negativos) (Terence Shin, 2020).

- **SKLEARN.MODEL_SELECTION IMPORT TRAIN_TEST_SPLIT**: Permite dividir un conjunto de datos en dos bloques, uno será el bloque de pruebas y el otro el bloque de entrenamiento (Daniel Burrueco, s/f-a).

- **SKLEARN.PIPELINE IMPORT PIPELINE**: El uso de tuberías (pipelines) es para realizar de una mejor manera el proceso de transformación y estimación en un conjunto de datos, hay que recordar que en pipelines la salida de un dato es la entrada del siguiente (Rodríguez, 2019).

- **SKLEARN.PREPROCESSING IMPORT STANDARDSCALER**: Realiza un estándar de acuerdo con los datos que se tengan, eliminando la media y entrenando los datos cuando el valor de varianza tenga como resultado 1 (Daniel Burrueco, s/f-b).

- **SKLEARN.SVM IMPORT LINEARSVC**: Se usa para traer el clasificador de vectores de soporte, con este se obtiene una clasificación binaria, para este caso 1 es el resultado de la predicción que es positivo y 0 será que el resultado de la predicción es negativo (pythonprogramming.net, s/f).

- **TRANSFORM(X):** Arroja un marco de datos que se procesa de manera automática, con datos que han sufrido una transformación luego de tener en cuenta una función que se asignara como parámetro (ALAKH SETHI, s/f).

8 REFERENCIAS BIBLIOGRÁFICAS

Adamssen, J. (2020). *Inteligencia artificial: Cómo el aprendizaje automático, la robótica y la ... - John Adamssen - Google Libros*.

ALAKH SETHI. (s/f). *Transformar función en Python, Pandas*. analyticsvidhya.

Alonso Romero, L., & Calonge Cano, T. (s/f). *Capítulo 1.-Redes Neuronales y Reconocimiento de Patrones*.

Analypeup. (s/f). *Tutorial de KFold estratificado | AnalyseUp.com*.

Berzal, F. (s/f). *Breve historia de la inteligencia artificial: el camino hacia la empresa*. Recuperado el 8 de septiembre de 2020, de https://asesoresdepymes.com/breve-historia-la-inteligencia-artificial-camino-hacia-la-empresa/

Christian Versloot. (s/f). *¿Cómo utilizar Dropout con Keras? - MachineCurve*.

Collab. (s/f). *Te damos la bienvenida a Colaboratory - Colaboratory*.

conexion capital. (2018). *Cerca de 1.000 pacientes con párkinson fueron atendidos en 2018 en Bogotá*. 26 de diciembre de 2018. https://conexioncapital.co/panorama-parkinson-bogota/

Constitución Politica de Colombia. (s/f). *Artículo 15 de la Constitución Política de Colombia*. Recuperado el 10 de septiembre de 2020, de https://www.constitucioncolombia.com/titulo-2/capitulo-1/articulo-15

Cornellius Yudha Wijaya. (2021). *5 método de selección de funciones de Scikit-Learn que debe conocer | de Cornellius Yudha Wijaya | Mar, 2021 | Hacia la ciencia de datos*.

Daniel Burrueco. (s/f-a). *sklearn.model_selection.train_test_split | Interactive Chaos*.

Daniel Burrueco. (s/f-b). *Standard Scaler | Interactive Chaos*.

Daniel Rodríguez. (2018a). *GridSearchCV - Analytics Lane*. https://www.analyticslane.com/2018/07/02/gridsearchcv/

Daniel Rodríguez. (2018b). *GridSearchCV - Analytics Lane*. analyticslane.com/2018/07/02/gridsearchcv/

Educative. (2021). *What is pandas in Python?*

Estevez, M. (s/f). *Un acercamiento a los modelos de clasificación - Inteligencia Analítica*. Recuperado el 10 de septiembre de 2020, de https://inteligencia-analitica.com/acercamiento-modelos-clasificacion/

Eugenia Bahit. (2013). *10.1. Módulos de sistema (Python para principiantes)*.

Federacion Española. (s/f). *DÍA MUNDIAL DEL PÁRKINSON 2019 - Federación Española de Parkinson*. Recuperado el 9 de marzo de 2021, de https://www.esparkinson.es/diamundialdelparkinson/

Función Publica. (s/f). *Ley 1581 de 2012 - EVA - Función Pública*. Recuperado el 10 de septiembre de 2020, de https://www.funcionpublica.gov.co/eva/gestornormativo/norma.php?i=49981

Galarnyk, M. (s/f). *PCA usando Python (scikit-learn) | de Michael Galarnyk | Hacia la ciencia de datos*.

Gramajo, E., -Martínez, G., Rossi, R., Claverie, B., & Totongi, P. Y. (s/f). *UNA VISION GLOBAL DEL APRENDIZAJE AUTOMATICO*.

Holguín, C., Díaz-Ricardo, Y., & Antonio Becerra-García, R. (2014). Ciencias Holguín. *El lenguaje de programación Python*, *XX*, 1–3. http://www.linuxjournal.com/article/2959

IA, A. (s/f). *Bosque Aleatorios Regresión - Teoría - ? Aprende IA*.

Itziar Gastón Zubimendi. (2018). *Enfermedad de Parkinson: prevención y tratamiento precoz – Zona Hospitalaria*.

https://zonahospitalaria.com/enfermedad-de-parkinson-prevencion-y-tratamiento-precoz/

Jason Brownlee. (2017). *How to One Hot Encode Sequence Data in Python*. https://machinelearningmastery.com/how-to-one-hot-encode-sequence-data-in-python/

Jason Brownlee. (2020). *Recursive Feature Elimination (RFE) for Feature Selection in Python*. https://machinelearningmastery.com/rfe-feature-selection-in-python/

John Hunter. (s/f). *Tutorial de Pyplot - documentación de Matplotlib 2.0.2*.

José Francisco López. (s/f-a). *Variable nominal - Qué es, definición y concepto | 2021 | Economipedia*.

José Francisco López. (s/f-b). *Variable ordinal - Qué es, definición y concepto | 2021 | Economipedia*.

JournalDev. (s/f). *Tutorial de aprendizaje de Python SciKit - JournalDev*.

Juan Ignacio Barrios Arce. (s/f). *La matriz de confusión y sus métricas – Inteligencia Artificial –*.

Justin Johnson. (s/f). *Tutorial de Python Numpy (con Jupyter y Colab)*. https://cs231n.github.io/python-numpy-tutorial/

Medlineplus. (2020). *Enfermedad de Parkinson: MedlinePlus en español*. 6 de marzo de 2020. https://medlineplus.gov/spanish/parkinsonsdisease.html

Michael Waskom . (s/f). *Introducción a seaborn - documentación de seaborn 0.11.1*.

Neilys González Benítez, Vivian Estrada Sentí, A. F. E. (s/f). *Estudio y selección de las técnicas de Inteligencia Artificial para el diagnóstico de enfermedades*. Recuperado el 8 de septiembre de 2020, de http://scielo.sld.cu/scielo.php?pid=S1561-

31942018000300014&script=sci_arttext&tlng=pt

Oracle. (s/f). *¿Qué es el aprendizaje automático? | Oracle Colombia*. Recuperado el 8 de septiembre de 2020, de https://www.oracle.com/co/artificial-intelligence/what-is-machine-learning.html

Pardo, C. (2020, julio). *¿Qué son las redes neuronales y cómo se aplican?* https://blog.enzymeadvisinggroup.com/redes-neuronales-que-son-y-aplicaciones

Piñeiro, M. A. (s/f). *DETECCIÓN AUTOMÁTICA DEL GRADO DE PARKINSON A PARTIR DE LA SEÑAL DE VOZ*. Recuperado el 8 de septiembre de 2020, de http://castor.det.uvigo.es:8080/xmlui/bitstream/handle/123456789/136/Andrés Piñeiro Martín.pdf?sequence=1

pythonprogramming.net. (s/f). *Tutoriales de programación de Python*. https://pythonprogramming.net/

Pythonprogramming.net. (s/f). *Python Programming Tutorials*. https://pythonprogramming.net/

Rodriguez, D. (2019). *Automatización del procesado de datos en Scikit-learn con Pipeline - Analytics Lane*.

Sakar, C. O., Serbes, G., Gunduz, A., Tunc, H. C., Nizam, H., Sakar, B. E., Tutuncu, M., Aydin, T., Isenkul, M. E., & Apaydin, H. (2019). A comparative analysis of speech signal processing algorithms for Parkinson's disease classification and the use of the tunable Q-factor wavelet transform. *Applied Soft Computing Journal*, *74*, 255–263. https://doi.org/10.1016/j.asoc.2018.10.022

scickit learn. (s/f-a). *1.13. Feature selection — scikit-learn 0.24.2 documentation*.

scickit learn. (s/f-b). *3.1. Cross-validation: evaluating estimator performance — scikit-learn 0.24.1 documentation*.

Solutions, M. (2018). *Diseño y Maquetación Dpto. Marketing y Comunicación Management Solutions-España*.

stephen_mugisha. (2020). *¿Cuál es la diferencia entre "from keras.models import Sequential" y "from tensorflow.python.keras.models import Sequential"? - Desbordamiento de pila*.

Tagle, P. (s/f). *Historia de la enfermedad de Parkinson*.

TensorFlow. (s/f). *tf.keras.wrappers.scikit_learn.KerasClassifier*.

Terence Shin. (2020). *Comprender la matriz de confusión y cómo implementarla en Python | de Terence Shin | Hacia la ciencia de datos*.

The TensorFlow Authors. (2020). *Regularizadores de peso de capa*. https://runebook.dev/es/docs/tensorflow/keras/regularizers/regularizer

tuthorials point. (2021). *Keras - Capas - Tutorialspoint*.

Vighnesh Birodkar. (2016). *Guía del usuario: contenido - documentación de sklearn-template 0.0.3*.

ANEXOS

Se anexa a la monografía en un documento aparte un Manual de Usuario, que contendrá las instrucciones para el uso del ejecutable de la investigación.