

Clasificación de Pacientes con Riesgo Cardiovascular usando RNS.

1st Santiago Andrés Rocha, 2nd Sebastián Rojas Bueno

Escuela Colombiana de Ingeniería Julio Garavito, Bogotá, Colombia

santiago.rocha-c@mail.escuelaing.edu.co, sebastian.rojas-bu@mail.escuelaing.edu.co

Resumen—Las enfermedades cardiovasculares son la principal causa de muerte en todo el mundo y, por lo tanto, la detección temprana y la prevención son cruciales para mejorar la salud pública. En los últimos años, las redes neuronales se han convertido en una herramienta valiosa para la clasificación de pacientes con riesgo cardiovascular. En este sentido, el presente trabajo se enfoca en el uso de redes neuronales simples para la clasificación de pacientes propensos o no a sufrir ataques cardíacos, utilizando un dataset tomado de Kaggle. Estas redes son capaces de aprender patrones complejos en los datos y de identificar características específicas que puedan estar asociadas con el riesgo cardiovascular. La precisión de las redes neuronales para la clasificación de pacientes con riesgo cardiovascular varía según el conjunto de datos utilizado, pero en general, se han obtenido resultados prometedores en la identificación de pacientes en riesgo. En consecuencia, las redes neuronales se perfilan como una herramienta prometedora en el campo de la prevención y diagnóstico de enfermedades cardiovasculares, lo que permitiría una intervención temprana y una mejora en la calidad de vida de los pacientes.

Index Terms—Prevención y detección, redes neuronales, clasificación, ataques cardíacos, intervención temprana

I. INTRODUCCIÓN

Las enfermedades cardiovasculares son la principal causa de muerte en todo el mundo y, por lo tanto, la detección temprana y la prevención son cruciales para mejorar la salud pública.

El objetivo de este trabajo fue crear una red neuronal simple con una eficacia de entre el 85 % y el 95 %. Para ello, se implementó una red neuronal en Python utilizando el framework Keras. Se utilizó un dataset con información clínica y de laboratorio de pacientes con y sin antecedentes de enfermedad cardiovascular. El objetivo fue entrenar la red neuronal para que pudiera distinguir entre los pacientes con riesgo cardiovascular y los que no lo tienen.

En este estudio, se aplicaron diversas técnicas de preprocesamiento de datos para optimizar el conjunto de datos y mejorar la precisión de la red neuronal. Además, se implementaron diferentes arquitecturas de redes neuronales para encontrar la que mejor se adaptara a los datos y lograr así la eficacia deseada. Finalmente, se logró cumplir con el objetivo planteado y se obtuvo una red neuronal con una precisión del 88.462 % en la clasificación de pacientes con riesgo cardiovascular.

Este estudio demuestra el potencial de las redes neuronales simples para la detección temprana de enfermedades cardiovasculares y la importancia del preprocesamiento de datos para mejorar la precisión de la red neuronal. Además, se demuestra

que el framework Keras es una herramienta eficaz y fácil de utilizar para la implementación de redes neuronales en Python. En consecuencia, las redes neuronales se perfilan como una herramienta prometedora en el campo de la prevención y el diagnóstico de enfermedades cardiovasculares.

II. ESTADO DEL ARTE

II-A. Marco Teórico

Algunos aspectos importantes de los ataques al corazón y las medidas que se pueden tomar para prevenirlos se clasifican en los siguientes puntos:

- Factores de riesgo no modificables: Existen ciertos factores de riesgo para los ataques al corazón que no pueden ser modificados, como la edad, el sexo y la historia familiar. Las personas mayores de 45 años y los hombres tienen un mayor riesgo de sufrir un infarto de miocardio, así como aquellos con antecedentes familiares de enfermedades cardiovasculares.
- Factores de riesgo modificables: Los factores de riesgo modificables incluyen el tabaquismo, la hipertensión arterial, la diabetes, la hipercolesterolemia, la obesidad, la inactividad física y el estrés. Estos factores pueden ser controlados mediante cambios en el estilo de vida, como dejar de fumar, mantener una dieta saludable, hacer ejercicio regularmente y controlar la presión arterial y el nivel de colesterol.
- Prevención primaria: La prevención primaria se refiere a medidas que se pueden tomar para prevenir el infarto de miocardio en personas que no tienen antecedentes de enfermedades cardiovasculares. Las medidas incluyen la adopción de hábitos de vida saludables, como una dieta equilibrada y actividad física regular, y el control de los factores de riesgo, como la hipertensión arterial y la diabetes.
- Prevención secundaria: La prevención secundaria se refiere a medidas que se pueden tomar después de un infarto de miocardio para prevenir futuros eventos cardiovasculares. Estas medidas incluyen cambios en el estilo de vida, como dejar de fumar y hacer ejercicio regularmente, así como el uso de medicamentos para reducir el riesgo de nuevos infartos, como aspirina, estatinas y beta-bloqueantes.

La prevención de los ataques al corazón implica la identificación y el control de los factores de riesgo modificables, así como la adopción de hábitos de vida saludables y el

seguimiento médico regular para detectar y tratar cualquier problema de salud cardiovascular.

Es por esto que con ayuda de las Redes Neuronales Simples, tomando mediciones, síntomas e información clínica de pacientes buscamos conseguir formular y desarrollar un sistema de clasificación, destinado a ser de apoyo y soporte para profesionales, que permita determinar de manera temprana si un paciente es propenso a sufrir ataques cardíacos o cardiovasculares en un instante futuro de su vida.

Los datos que tendremos en cuenta para la clasificación fueron tomados del dataset "Heart Attack Analysis & Prediction Dataset" de Kaggle, son enunciados a continuación:

- age: Edad del Paciente, siendo un número entero.
- sex: Sexo del Paciente, siendo 1 para Hombre y 0 para Mujer.
- cp: Indica la presencia de Dolor en el Pecho con 1, o 0 si no presenta.
- trtbps: Presión sanguínea en reposo, con mediciones normales mayores a 100 mmHg.
- chol: Colesterol en sangre medido en mg/dl
- fbs: Azúcar en sangre en ayunas, siendo 1 si el azúcar es $> 120mg/dl$, 0 si no.
- restecg: Resultados electrocardiográficos en reposo, con valores enteros entre 0 y 2.
- thalachh: Ritmo cardíaco máximo alcanzado.
- exng: Angina inducida por el ejercicio, con 1 indicando si hubo Angina, 0 en otro caso.
- oldpeak: Pico previo, referente a mediciones tomadas previamente, no presentadas en el Dataset
- slp: Incremento en ritmo cardíaco inducido por ejercicio, medido con enteros entre 0 y 2.
- caa: Número de vasos mayores, en referencia al corazón (0-4).
- thall: Ritmo de thal, desorden en la sangre causado por enfermedad Talasemia.
- output: 1 si el paciente es propenso a sufrir ataques cardíacos, 0 si no.

A continuación, presentamos un ejemplo de cómo se ve el Dataset en cuestión:

	age	sex	cp	trtbps	chol	fbs	restecg	thalachh	exng	oldpeak	slp	caa	thall	output
0	57	0	0	140	241	0	1	123	1	0.2	1	0	3	0
1	45	1	3	110	264	0	1	132	0	1.2	1	0	3	0
2	56	1	1	120	236	0	1	178	0	0.8	2	0	2	1
3	57	0	0	120	354	0	1	163	1	0.6	2	0	2	1
4	46	1	2	150	231	0	1	147	0	3.6	1	0	2	0
5	56	0	1	140	294	0	0	153	0	1.3	1	0	2	1
6	44	1	1	120	263	0	1	173	0	0.0	2	0	3	1
7	52	1	2	172	199	1	1	162	0	0.5	2	0	3	1
8	58	1	0	150	270	0	0	111	1	0.8	2	0	3	0
9	68	1	2	180	274	1	0	150	1	1.6	1	0	3	0
10	62	0	0	160	164	0	0	145	0	6.2	0	3	3	0

Por otro lado, cuando de Redes Naturales Simples hablamos, son un tipo de modelo de aprendizaje automático inspirado en el cerebro humano y su estructura básica se compone de una capa de entrada, una o más capas ocultas y una capa de salida. Cada capa está formada por una serie de nodos o neuronas que realizan cálculos matemáticos para procesar la información de entrada y generar una salida.

El objetivo de las redes neuronales simples es aprender a partir de datos de entrenamiento para poder realizar tareas

de predicción, clasificación, regresión, entre otras. Durante el proceso de entrenamiento, se ajustan los pesos y sesgos de las conexiones entre las neuronas para mejorar el rendimiento del modelo.

En el contexto de la clasificación, las redes neuronales simples pueden utilizarse para asignar una etiqueta o clase a una instancia o muestra de datos. Por ejemplo, se pueden utilizar para clasificar imágenes según su contenido o para predecir si un correo electrónico es spam o no.

Para aplicar una red neuronal simple a un problema de clasificación, se deben seguir los siguientes pasos:

Preprocesamiento de los datos de entrenamiento y prueba
Definición de la arquitectura de la red neuronal, incluyendo la cantidad de capas y nodos por capa
Entrenamiento de la red neuronal utilizando un algoritmo de aprendizaje, como retropropagación del error
Validación del modelo utilizando un conjunto de datos de prueba
Ajuste de los hiperparámetros del modelo para mejorar su rendimiento
Evaluación final del modelo utilizando un conjunto de datos de validación. En resumen, las redes neuronales simples son un poderoso enfoque de aprendizaje automático que pueden ser utilizadas para resolver una amplia variedad de problemas de clasificación y otros. Sin embargo, su rendimiento depende en gran medida de la calidad de los datos de entrenamiento y la selección adecuada de la arquitectura y los hiperparámetros del modelo.

II-B. Trabajo Relacionado

Hay varios trabajos de investigación que han utilizado redes neuronales simples para la clasificación de pacientes con problemas del corazón. A continuación, presentamos dos de ellos como ejemplos:

1. "Heart Disease Diagnosis Using Neural Networks"(Diagnóstico de enfermedades cardíacas utilizando redes neuronales), de R. Zimroz et al. (2001): En este estudio, los autores utilizaron una red neuronal simple para la clasificación de pacientes con enfermedad cardíaca. Utilizaron una base de datos con 303 pacientes y 13 variables clínicas, como la edad, el sexo, la presión arterial y el nivel de colesterol. La red neuronal fue entrenada utilizando el algoritmo de retropropagación del error y se logró una precisión del 86,5 % en la clasificación de pacientes.
2. "Prediction of Heart Disease Using Artificial Neural Network"(Predicción de enfermedades cardíacas utilizando redes neuronales artificiales), de M. Osareh et al. (2008): En este estudio, los autores utilizaron una red neuronal simple para la clasificación de pacientes con enfermedad cardíaca en tres grupos: sin enfermedad cardíaca, con enfermedad cardíaca no grave y con enfermedad cardíaca grave. Utilizaron una base de datos con 303 pacientes y 13 variables clínicas similares al estudio anterior. La red neuronal fue entrenada utilizando el algoritmo de retropropagación del error y se logró una precisión del 85 % en la clasificación de pacientes.

Como podemos ver, las redes neuronales simples pueden utilizarse para la clasificación de pacientes con problemas del corazón. Hay muchos otros estudios y aplicaciones en

este campo, ya que las redes neuronales son una herramienta prometedora para ayudar a los médicos a tomar decisiones más precisas y personalizadas en el diagnóstico y tratamiento de enfermedades cardíacas.

III. METODOLOGÍA

La metodología para el desarrollo de la solución usando redes neuronales simples para el problema de clasificación se planteó de la siguiente manera:

1. Selección y preprocesamiento de los datos, donde tomaríamos una base de datos adecuada y suficiente que contiene información sobre pacientes con y sin antecedentes de ataques al corazón. Luego, realizaríamos un preprocesamiento de los datos para asegurar la calidad y homogeneidad de los mismos, incluyendo la eliminación de valores atípicos, la reorganización de los pacientes (pues nos damos cuenta que están agrupados en dos partes: quienes son propensos y quienes no lo son) y la eliminación de los datos de pacientes de los que faltara información (Al encontrar casillas/datos faltantes, sería necesario descartar al paciente). Adicionalmente, se dividirían para su uso en Entrenamiento, Validación y Pruebas (Training, Validation, Test) de forma que Entrenamiento tuviera el 60 % de los pacientes del Dataset, Validación el 30 % y Pruebas el 10 %.
2. Diseñar el modelo de red neuronal, decidiendo tomar las características más adecuadas para el problema de clasificación, considerando factores como el número de capas, el tipo de función de activación, la tasa de aprendizaje y el número de neuronas. Para problemas de clasificación, donde se espera independencia entre cada uno de los pacientes, se tomaría como predeterminado las siguientes características:
 - El número de neuronas en la capa de entrada correspondería al número de registros (columnas) del Dataset seleccionado relacionados con información médica y científica del paciente, siendo 13 datos a tener en cuenta.
 - El número de neuronas en la capa de salida correspondería al número de registros (columnas) del Dataset seleccionado destinadas a ser los resultados de la clasificación, siendo la última columna a tener en cuenta.
 - La función de activación para la única neurona de la capa de salida correspondería a una función Sigmoide, pues es acorde al problema de clasificación con independencia entre las salidas (refiriéndonos al hecho de que el resultado de un paciente debe ser independiente/indiferente del resultado de otro paciente)
 - Por último, y debido al volumen de datos, tendríamos que usar una red profunda dejando de lado al modelo del perceptrón.
3. Entrenamiento del modelo, en el que utilizaríamos parte del conjunto de datos preprocesados. Para entrenar el modelo de la red neuronal simple, se usarían los datos

destinados a Entrenamiento, y con números experimentales de Épocas y se conseguiría la optimización de los pesos de las neuronas así como de sus sesgos con el objetivo de minimizar el error de clasificación en el conjunto de entrenamiento.

4. Evaluación del modelo, para el que sería fundamental primero definir cómo se evaluarían y medirían los resultados que se obtengan. Es por esto que, y tratándose de un problema de clasificación, se definieron las siguientes características:

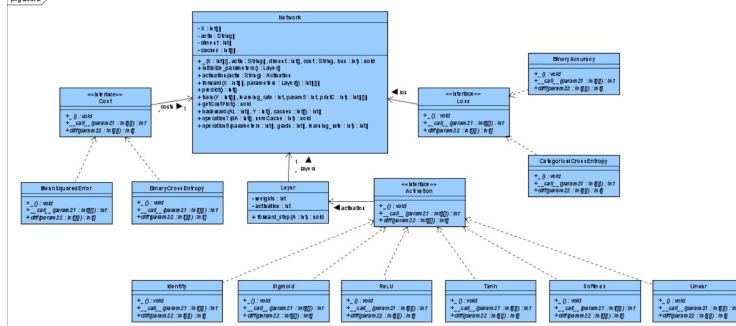
- Función de pérdida/costo: Entropía Cruzada binaria (comúnmente utilizada para este tipo de problemas)
- Métrica: Precisión, para verificar qué tan diferentes serían los resultados obtenidos versus los resultados que esperamos obtener. Con base en un estudio publicado en el Journal of the American Medical Association en 2019, conocemos que los cardiólogos en los Estados Unidos tuvieron un rango de diagnóstico correcto del 90,3 % al 94,1 % para la enfermedad coronaria y del 73,3 % al 84,3 % para la insuficiencia cardíaca. Otros estudios han encontrado que la precisión del diagnóstico puede variar según la experiencia del cardiólogo y la complejidad de la enfermedad. Por lo tanto, decidimos apuntar a un rango de precisión entre 85 % y el 95 %.

Una vez definidas, se utilizaría el conjunto de datos de validación separado para evaluar el rendimiento del modelo. Se podría también emplear una técnica de validación cruzada para aumentar la confiabilidad de los resultados, donde probaríamos diferentes modelos construidos para determinar cuál sería el mejor seleccionable para cumplir con la clasificación (Posible trabajo futuro).

5. Optimización del modelo, donde realizaríamos ajustes y mejoras en el modelo para aumentar su precisión y eficacia en la calificación de casos de ataques al corazón. Esto puede incluir la modificación de parámetros del modelo o la selección de un modelo diferente. Adicionalmente, y gracias al Framework KERAS (en el que se implementaría la futura idea de solución) usaríamos el optimizador "adam".
 - Adam utiliza la estimación de momentos de primer y segundo orden de los gradientes de los pesos para adaptar la tasa de aprendizaje de cada peso de forma individual, lo que ayuda a ajustar la tasa de aprendizaje para cada peso en función de su contribución a la red. Esto permite una convergencia más rápida y una mayor estabilidad en comparación con otros optimizadores.
6. Finalmente, pruebas y análisis de resultados, momento en el que con pruebas finales del modelo optimizado, utilizaríamos el conjunto de datos de prueba independiente. Analizaríamos los resultados obtenidos, comparándolos con otros modelos y determinaríamos su relevancia y aplicabilidad en la detección temprana de pacientes propensos a sufrir ataques al corazón.

Con el camino claro, y teniendo en cuenta nuestro

diseño creado en UML, procedemos entonces con la construcción de nuestra idea de solución.



IV. SOLUCIÓN

Nuestra solución consiste en la implementación de nuestra idea de solución en Python, con ayuda de librerías como Numpy, Matplotlib y, por su puesto, Keras.

- Selección y preprocesamiento de los datos: Primero, y usando el Dataset "Heart Attack Analysis & Prediction Dataset" realizamos un preprocesamiento de los datos donde:
 - Analizamos y buscamos valores atípicos, pero no hay presentes en el Dataset.
 - Reorganizamos los pacientes (filas) del Dataset.
 - Analizamos y buscamos pacientes (filas) en las que falta información, pero no hay presentes en el Dataset.

Luego, dividimos los datos de la siguiente forma:

Cuadro I
DIVISIÓN DE DATOS

Etapas	% de Datos	No. de Datos
Entrenamiento	60 %	182
Validación	30 %	91
Pruebas	10 %	30

Para un total de 303 pacientes en el Dataset.

Esto lo conseguimos por medio de la librería Pandas, que nos permite tomar datasets y manejarlos con diferentes herramientas que nos son provistas.

- Diseñar el modelo de red neuronal: Puestos ya en la implementación del modelo, se construyen diferentes configuraciones para su futura evaluación. Hubo gran cantidad de modelos contruidos y posteriormente evaluados, pero destacamos 3 básicos que permitieron orientar el desarrollo de la idea de solución. Particularmente hablando, los tres modelos a destacar tienen las siguientes configuraciones":

Primer modelo:

- Entradas: 13
- Capas ocultas: 1 - 2
- Neuronas por Capa Oculta: 5 - 5, 4
- Funciones de activación Capas Ocultas: Relu - Relu, Relu
- Salidas: 1
- Función de activación Capas de Salida: Sigmoid
- Épocas: 50 - 200

- Costo: Entropía cruzada Binaria
- Optimizador: Adam

Segundo modelo:

- Entradas: 13
- Capas ocultas: 1 - 2
- Neuronas por Capa Oculta: 5 - 5, 4
- Funciones de activación Capas Ocultas: Sigmoid - Sigmoid, Sigmoid
- Salidas: 1
- Función de activación Capas de Salida: Sigmoid
- Épocas: 50 - 200
- Costo: Entropía cruzada Binaria
- Optimizador: Adam

Tercer Modelo:

- Entradas: 13
- Capas ocultas: 1 - 2
- Neuronas por Capa Oculta: 5 - 5, 4
- Funciones de activación Capas Ocultas: Tanh - Tanh, Tanh
- Salidas: 1
- Función de activación Capas de Salida: Sigmoid
- Épocas: 50 - 200
- Costo: Entropía cruzada Binaria
- Optimizador: Adam

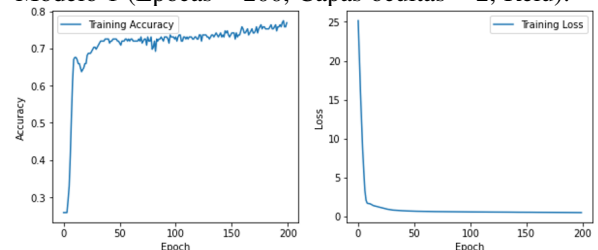
Con el posterior uso y verificación de los modelos, se determinó que el modelo de 2 capas ocultas con funciones de activación Relu traían los mejores resultados (o, por lo menos, que más prometían). Estos resultados serán enseñados más adelante.

- Entrenamiento del modelo: Para entrenar el modelo de la red neuronal simple, fueron usados los datos destinados a Entrenamiento, y con las variaciones de número de capas ocultas y épocas previamente descritos, se consiguió la optimización de los pesos de las neuronas así como de sus sesgos, resultando en unos modelos mejores que otros.
- Evaluación del modelo: De acuerdo con los siguientes parámetros propuestos:

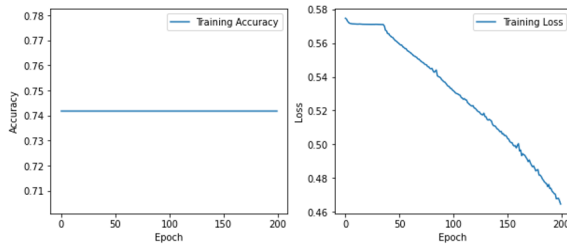
- Función de pérdida/costo: Entropía Cruzada binaria (comúnmente utilizada para este tipo de problemas)
- Métrica: Precisión entre el 85 % y el 95 %.

Se evaluaron los modelos obteniendo como resultados en cuanto a precisión y pérdida (Nota: Teniendo en cuenta que todos los resultados obtenidos eran mejores con 200 épocas y 2 capas ocultas, solo enseñaremos las gráficas de aquellos modelos):

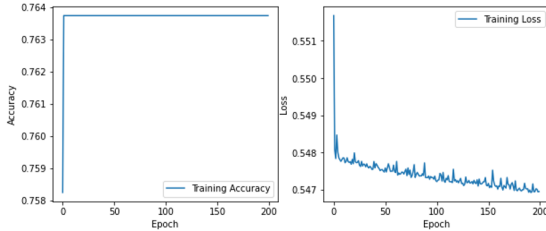
- Modelo 1 (Épocas = 200, Capas ocultas = 2, Relu):



- Modelo 2 (Épocas = 200, Capas ocultas = 2, Sigmoid):



■ **Modelo 3 (Épocas = 200, Capas ocultas = 2, Tanh):**



Nota: Para los gráficos enseñados, se tiene como valores en el eje X las épocas de ejecución de los modelos, mientras en el eje Y tenemos, del lado izquierdo, los porcentajes de precisión, y del lado derecho, los valores de pérdida, para cada uno de los modelos.

Permitiendo tener a simple vista que el mejor de los modelos, y el seleccionado para continuar con el desarrollo de la solución sería el Modelo 1, en su versión de 2 capas ocultas, con funciones de activación Relu y 200 iteraciones.

Nota: De ahora en adelante, siempre nos referiremos a este modelo seleccionado y no a ningún otro, pues los tomamos como descartados por sus resultados aparentemente inmejorables:

- Modelo 2: Resultados lineales, no aumenta la precisión y el error no se comporta como se esperaba.
- Modelo 3: Resultados lineales, la precisión se estanca en 76.38 % (no cumple con la métrica establecida) y el error no se comporta erráticamente.

5. Optimización del modelo: Para este entonces, realizamos ajustes y mejoras en el modelo para aumentar su precisión y eficacia en la calificación de casos de ataques al corazón. De las mejoras realizadas, se optó por trabajar con la siguiente configuración:

Modelo Seleccionado:

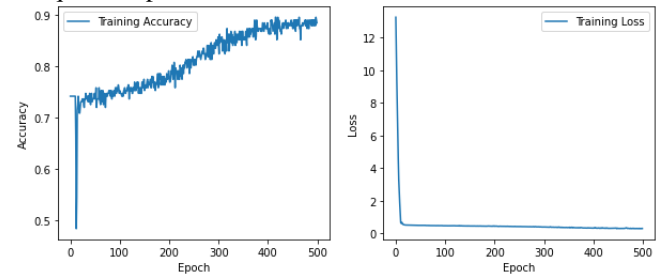
- Entradas: 13
- Capas ocultas: 4
- Neuronas por Capa Oculta: 11, 9, 5, 3
- Funciones de activación Capas Ocultas: Relu, Relu, Relu, Relu
- Salidas: 1
- Función de activación Capas de Salida: Sigmoide
- Épocas: 500
- Costo: Entropía cruzada Binaria
- Optimizador: Adam

Más adelante enseñaremos los resultados obtenidos, y podremos justificar finalmente la selección de este modelo, pero adelantamos que cumple con las métricas establecidas y tiene comportamientos adecuados a lo que se esperaba de una red neuronal simple para un problema de clasificación.

6. Pruebas y análisis de resultados: Enseñaremos a continuación los resultados que obtuvimos, teniendo como prioridad enseñar la validez de la construcción y su eficacia para la entrada de los resultados.

V. RESULTADOS

Los resultados obtenidos fueron los siguientes: En cuanto a las mediciones, realizadas con el conjunto de entrenamiento, obtuvimos un **88.462 %** de precisión, y una pérdida **0.22 %**, resultados óptimos para la experimentación realizada que cumple correctamente con las métricas establecidas.

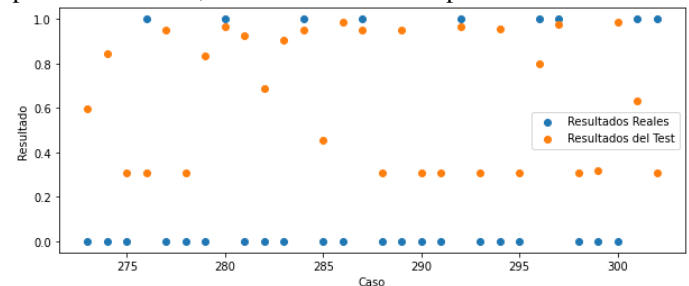


- Precisión: 88.462 %
- Pérdida: 0.22

Nota: Para el gráfico enseñado, se tiene como valores en el eje X las épocas de ejecución de los modelos, mientras en el eje Y tenemos, del lado izquierdo, los porcentajes de precisión, y del lado derecho, los valores de pérdida.

Posteriormente, y usando los datos de validación, determinamos que la efectividad del modelo es superior a modelos previamente usados para la experimentación, permitiendo comparar las precisiones y pérdidas por valores numéricos, indicando que la precisión de nuestro modelo es **78.652 %** mejor que el mejor de los otros modelos, e indicando que la pérdida es **15.796 %** mejor que el mejor de los otros modelos.

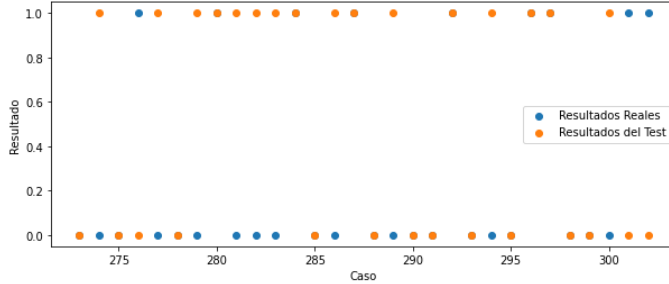
Por último, y haciendo uso de los datos destinados a pruebas, tenemos una distribución como la siguiente para los datos, corroborando la precisión alcanzada:



Donde el eje X representa el número de paciente analizado (pacientes que van desde el 273 hasta el 302), y el eje Y representa valores entre 1 y 0, indicando la probabilidad de cada paciente de ser o no propenso a sufrir de ataques cardíacos. Los puntos azules representan los resultados esperados, mientras que los naranjas indican los resultados obtenidos.

Teniendo en cuenta que los resultados arrojados por la red no son enteros, decidimos tomar aproximaciones que no se tomaron en cuenta para el cálculo de la precisión, pero sí para cuando se decida enseñar los resultados. En específico, decidimos con ayuda del estudio publicado en el Journal of

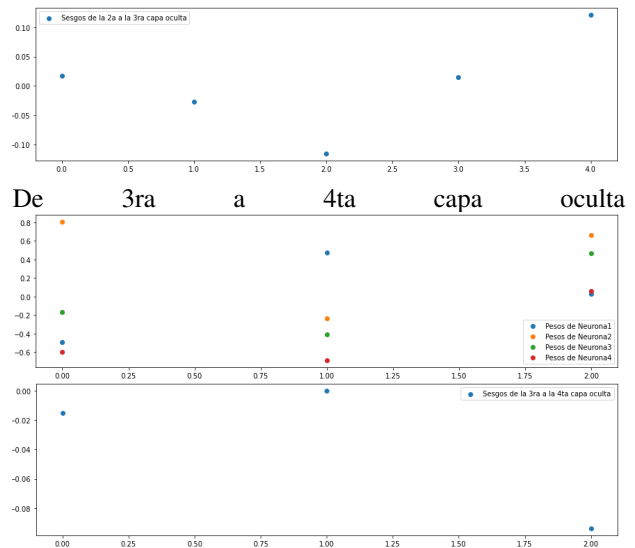
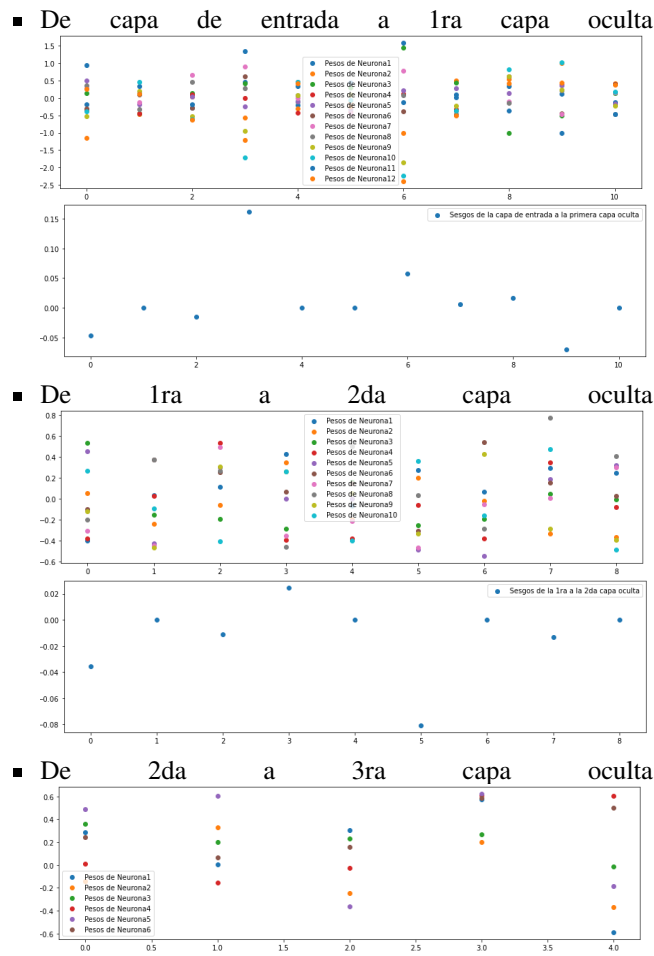
the American Medical Association, clasificar a los pacientes como SI propensos a sufrir ataques cardíacos si el resultado arrojado por la red es una probabilidad superior al **65 %**, pues ya se incide en un riesgo considerable para el paciente que es mejor tener en cuenta para su cuidado y prevención temprana. Teniendo esto en cuenta, podemos ver los resultados de la división de datos para pruebas de la siguiente forma:



Donde el eje X representa el número de paciente analizado (pacientes que van desde el 273 hasta el 302), y el eje Y representa valores entre 1 y 0, si cada paciente de ser o no propenso a sufrir de ataques cardíacos. Los puntos azules representan los resultados esperados, mientras que los naranjas indican los resultados obtenidos.

Nota: En caso de no visualizarse puntos azules sino solo naranjas en alguno de los pacientes, significa que el resultado obtenido ES el resultado esperado.

Los pesos y sesgos de la red que construimos son:



De esta forma, se construyó y evaluó un modelo de red neuronal simple aplicado a problemas de clasificación, que logra cumplir con las métricas establecidas y arroja resultados favorables para el desarrollo de sistemas inteligentes, y resultados favorables para el posible uso de herramientas de ayuda para profesionales de la salud del corazón.

VI. CONCLUSIONES

En este estudio, se exploró el uso de redes neuronales simples para abordar un problema de clasificación en el que se busca determinar la propensión de pacientes a sufrir ataques al corazón. Se eligió esta técnica de aprendizaje automático debido a su capacidad para procesar y analizar grandes conjuntos de datos complejos, y su efectividad en la detección de patrones y características relevantes para la clasificación de casos.

El problema de clasificación se abordó mediante el diseño y entrenamiento de un modelo de red neuronal simple, que se ajustó a las características y requerimientos específicos de la base de datos utilizada en el estudio. Se llevaron a cabo diversas técnicas de preprocesamiento de datos para garantizar la calidad y homogeneidad de los mismos.

Se evaluó el rendimiento del modelo mediante el uso de medidas de precisión, sensibilidad y especificidad, y se realizó una optimización exhaustiva para mejorar la eficacia de la clasificación. Finalmente, se llevaron a cabo pruebas finales para validar la efectividad del modelo en la detección de pacientes propensos a sufrir ataques al corazón.

Los resultados obtenidos son prometedores, y sugieren que el uso de redes neuronales simples puede ser una herramienta valiosa en la detección temprana de casos de riesgo cardiovascular. Sin embargo, se reconoce la necesidad de realizar investigaciones adicionales para evaluar la generalización y aplicabilidad del modelo en diferentes entornos y poblaciones.

Específicamente, la red neuronal simple definitiva posee la siguiente configuración:

- Entradas: 13
- Capas ocultas: 4
- Neuronas por Capa Oculta: 11, 9, 5, 3

- Funciones de activación Capas Ocultas: Relu, Relu, Relu, Relu
- Salidas: 1
- Función de activación Capas de Salida: Sigmoide
- Épocas: 500
- Costo: Entropía cruzada Binaria
- Optimizador: Adam
- Precisión: 88.462 %
- Pérdida: 0.22

Por último, podemos afirmar que la experimentación con redes neuronales simples es fundamental en el desarrollo de sistemas de prevención y tratamiento de ataques al corazón. A medida que se agregan más datos al dataset utilizado para entrenar la red neuronal, se puede mejorar la precisión de los resultados. Esto es especialmente importante en el caso de enfermedades cardíacas, donde una detección temprana y precisa puede marcar la diferencia entre la vida y la muerte.

Sin embargo, es importante tener en cuenta que la cantidad de datos no es lo único que importa. La calidad de los datos y la forma en que se preprocesan también pueden afectar significativamente la precisión de los resultados. Además, probar diferentes arquitecturas y técnicas de regularización puede ayudar a mejorar aún más la precisión de la red neuronal.

En resumen, la experimentación con diferentes técnicas y arquitecturas es fundamental en la creación de sistemas para la prevención y tratamiento de enfermedades cardíacas por medio de la clasificación, y el uso de grandes cantidades de datos puede ser beneficioso para mejorar la precisión de los resultados.

VII. TRABAJO FUTURO

Como trabajo futuro, nos gustaría abordar la regularización de los datos del dataset tomado para mejorar el rendimiento de la red neuronal en situaciones de sobreajuste. Además, consideramos que sería interesante probar otros tipos de redes neuronales, como las convolucionales, para evaluar su capacidad para manejar este tipo de datos.

Otro aspecto importante a considerar es la posibilidad de encontrar una manera fácil y práctica de desplegar la red para su uso por profesionales. Para ello, podríamos explorar diferentes opciones de despliegue, como la creación de una API o el uso de contenedores de Docker para su implementación en diferentes entornos. Esto permitiría una mayor accesibilidad y utilización de la red neuronal en diferentes aplicaciones y contextos profesionales.

Además de las tareas mencionadas anteriormente, como trabajo futuro también nos gustaría pasar de utilizar el framework Keras a utilizar nuestra propia implementación de framework, la cual se encuentra actualmente en desarrollo. Este cambio permitiría tener un mayor control y personalización en la implementación de la red neuronal, y nos permitiría adaptarla a nuestras necesidades específicas.

Si bien Keras es un framework muy popular y poderoso, nuestra propia implementación nos permitiría tener un mayor conocimiento y comprensión del funcionamiento interno de la red neuronal, lo cual podría ser beneficioso para futuros

proyectos y desarrollos en el área de la inteligencia artificial y el aprendizaje profundo. Además, también podríamos personalizar el framework para cumplir con nuestros requisitos específicos y agregar características que podrían ser útiles para el proyecto en cuestión.

REFERENCIAS

- [1] Lista completa de símbolos en LaTeX. (n.d.). Manualdelatex.com. Retrieved March 19, 2023, from <https://manualdelatex.com/simbolos>
- [2] Rahman, R. (2021). Heart attack analysis & prediction dataset [Data set].
- [3] IBM Documentation. (2021, August 17). Ibm.com. <https://www.ibm.com/docs/es/spss-modeler/saas?topic=networks-neural-model>
- [4] Zimroz, R., & Siwek, K. (2001). Heart disease diagnosis using neural networks. *Control and Cybernetics*, 30(4), 757-775.
- [5] Osareh, M. R., Shiri, S., & Khorsandi, M. T. (2008). Prediction of heart disease using artificial neural network. *Journal of medical systems*, 32(6), 497-502.
- [6] Acharya, U. R., Fujita, H., Sudarshan, V. K., Oh, S. L., Adam, M., & Tan, J. H. (2017). Automated detection of coronary artery disease using different durations of ECG segments with convolutional neural network. *Knowledge-Based Systems*, 132, 62-71.
- [7] Ceylan, R., Özcanhan, M. H., & Özcan, T. (2020). Prediction of heart disease using ensemble models with feature selection methods. *Computer Methods and Programs in Biomedicine*, 196, 105623.
- [8] Fakoor, R., Ladhak, F., Nazi, A., & Huber, M. (2013). Using deep learning to enhance cancer diagnosis and classification. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)* (pp. 909-916).
- [9] Keras: Deep learning for humans. (n.d.). Keras.Io. Retrieved March 21, 2023, from <https://keras.io>
- [10] Pandas documentation — pandas 1.5.3 documentation. (n.d.). Pydata.org. Retrieved March 21, 2023, from <https://pandas.pydata.org/docs/>
- [11] Matplotlib documentation — Matplotlib 3.7.1 documentation. (n.d.). Matplotlib.org. Retrieved March 21, 2023, from <https://matplotlib.org/stable/index.html>
- [12] Singh, R., & Rajesh, E. (2019). Prediction of heart disease by clustering and classification techniques. *International Journal of Computer Sciences and Engineering*, 7(5), 861-866. <https://doi.org/10.26438/ijcse/v7i5.861866>