UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI

POSGRADO

MAESTRÍA EN ESTADÍSTICA APLICADA

“Predicción de la vida útil restante de turborreactores de doble flujo basado en las redes neuronales artificiales”

Trabajo de titulación previa la obtención del
Título de Magister en Estadística Aplicada

Autor: Rómulo Alexis Rivera Auqui
Tutor: Félix Antonio García Mora

Tulcán, 2025

CERTIFICADO DEL TUTOR

Certifico que el estudiante Rómulo Alexis Rivera Auqui con el número de cédula 0604692715 ha elaborado el Trabajo de Titulación: “Predicción de la vida útil restante de turborreactores de doble flujo basado en las redes neuronales artificiales”.

Este trabajo se sujeta a las normas y metodología dispuesta en el Reglamento de Régimen Académico y de Estudiantes de la Universidad Politécnica Estatal del Carchi con RESOLUCIÓN No. 183-CSUP-2024, por lo tanto, autorizo su presentación para la sustentación respectiva.

Firmado electrónicamente por:
FELIX ANTONIO GARCIA MORA
Validar únicamente con FirmaEC

f.....

M.Eng. Félix Antonio García Mora

Tulcán, septiembre de 2025

AUTORIA DEL TRABAJO

El presente trabajo de titulación constituye un requisito previo para la obtención del título de Magister en Estadística Aplicada.

Yo, Rómulo Alexis Rivera Auqui, ciudadano ecuatoriano con cédula de identidad número 0604692715 declaro: que la investigación es absolutamente original, autentica, personal y los resultados y conclusiones a los que he llegado son de mi absoluta responsabilidad.

Firmado electrónicamente por:
ROMULO ALEXIS
RIVERA AUQUI
Validar únicamente con FirmaEC

f.....

Rómulo Alexis Rivera Auqui

Tulcán, septiembre de 2025

ACTA DE CESIÓN DE DERECHOS DE LA TDT

Yo, Rómulo Alexis Rivera Auqui declaro ser autor de los criterios emitidos en el trabajo de titulación: “Predicción de la vida útil restante de turbo reactores de doble flujo basado en las redes neuronales artificiales” y eximo expresamente a la Universidad Politécnica Estatal del Carchi y a sus representantes legales de posibles reclamos o acciones legales.

f.....

Rómulo Alexis Rivera Auqui

Tulcán, septiembre de 2025

DEDICATORIA

A Dios, por ser mi luz y fortaleza en cada etapa de este camino, dándome la sabiduría y perseverancia necesarias para alcanzar esta meta.

A la memoria de mi padre, Rómulo Genaro Rivera Oleas, cuyo ejemplo de esfuerzo, dedicación y valores sigue siendo mi guía. Aunque no está físicamente, su legado de trabajo y amor perdura en cada uno de mis logros.

A mi querida madre, Aida Leonila Auqui Molina, por su infinito amor, sacrificio y apoyo incondicional. Su fortaleza y enseñanzas han sido el pilar fundamental en mi vida, inspirándome a seguir adelante con determinación y humildad.

A mis hermanos, quienes han estado a mi lado brindándome su aliento y motivación en cada desafío.

Con cariño y gratitud,
Rómulo Alexis Rivera Auqui

AGRADECIMIENTO

La culminación de este trabajo ha sido posible gracias al apoyo y colaboración de diversas personas, a quienes expreso mi más sincero agradecimiento.

A la Universidad Politécnica Estatal del Carchi, por brindarme la oportunidad de formarme en un ambiente de excelencia académica y por facilitar los recursos necesarios para el desarrollo de esta investigación.

Al M.Eng. Félix Antonio García Mora, tutor de este proyecto de investigación, por su invaluable asesoría, orientación y dedicación en cada etapa del proceso. Su guía ha sido fundamental para la estructuración y consolidación de este trabajo.

Finalmente, a todas aquellas personas que, de una u otra manera, contribuyeron con su asistencia técnica, científica para la realización de esta investigación. A todos ustedes, mi más profundo agradecimiento.

ÍNDICE

RESUMEN xv

ABSTRACT xvi

CAPÍTULO I 1

PROBLEMA 1

1.1. Planteamiento del problema 1

1.2. Preguntas de Investigación o Hipótesis 3

1.2.1 Pregunta general de investigación 3

1.2.2 Preguntas específicas de investigación 3

1.3. Objetivos de investigación 3

1.3.1 Objetivo General 3

1.3.2 Objetivos Específicos 4

1.4. Justificación 5

CAPÍTULO II 7

FUNDAMENTACIÓN TEÓRICA 7

2.1. Antecedentes de la investigación 7

2.2. Marco teórico 11

2.2.1 Turborreactores de doble flujo 11

2.2.2 Componentes principales y principio de funcionamiento del turbofán 12

2.2.3 Modelos de predicción de RUL 13

2.2.4 Modelos de predicción basados en Física 14

2.2.5 Modelos de predicción basados en Redes Neuronales Artificiales 14

2.2.6 Arquitectura de las redes neuronales artificiales (RNA) 20

2.2.7 Entrenamiento de la red neuronal artificial RNA 26

2.2.8 Normalización 29

2.2.9 Métricas de evaluación....32
2.3. Marco Legal....36
2.3.1 Normativas de Ingeniería Aeroespacial....36
2.3.2 Normativa sobre procesamiento de datos....37
2.3.3 Normativas sobre el uso de la inteligencia artificial....38
CAPÍTULO III....39
METODOLOGÍA....39
3.1. Descripción del área de estudio....39
3.2. Enfoque y tipo de investigación....40
3.2.1 Enfoque....40
3.2.2 Tipo de investigación....40
3.3. Definición y operacionalización de variables....40
3.4. Procedimientos....41
3.4.1 Descripción del problema....42
3.4.2 Obtención de la base de datos....43
3.4.3 Descripción general de la base de datos....45
3.4.4 Procesamiento de la base de datos....50
3.4.5 Preparación del conjunto de entrenamiento....56
3.4.6 Diseño de modelos de RNA....56
CAPÍTULO IV....79
RESULTADOS Y DISCUSIÓN....79
4.1. Evaluación de los modelos entrenados....79
4.1.1 Model_básico....79
4.1.2 Model_básico1....81
4.1.3 Model_básico2....82
4.1.4 Model_básico3....83

4.1.5 Model_básico4 84
4.1.6 Model_GRU 86
4.1.7 Model_GRU1 87
4.1.8 Modelo GRU2 88
4.1.9 Modelo GRU3 89
4.1.10 Modelo LSTM 91
4.1.11 Modelo LSTM1 92
4.2. Análisis de resultados en escala Original 93
4.3. Discusión de los resultados obtenidos 98
4.4. Respuesta a la pregunta de investigación 99
CONCLUSIONES Y RECOMENDACIONES 101
5.1. Conclusiones 101
5.2. Recomendaciones 101
REFERENCIAS 103
ANEXOS 106

ÍNDICE DE TABLAS

Tabla 1 Operacionalización de variables 41

Tabla 2 Estructura de las columnas de la base de datos del conjunto de entrenamiento y prueba 46

Tabla 3 Datos de entrenamiento.....47

Tabla 4 Datos de prueba 48

Tabla 5 Datos de RUL (Vida Útil Restante).....49

Tabla 6 Asignación de MachineID al conjunto RUL**¡Error! Marcador no definido.**

Tabla 7 Resumen de valores atípicos en el conjunto de entrenamiento 54

Tabla 8 Resumen de valores atípicos en el conjunto de prueba.....54

Tabla 9 Estadísticas descriptivas de RUL para filas con valores atípicos y no atípicos 55

Tabla 10 Diseño de una red neuronal Model_básico 57

Tabla 11 Diseño de una red neuronal Model_básico1 59

Tabla 12 Diseño de una red neuronal Model_básico2 61

Tabla 13 Diseño de una red neuronal Model_básico3 63

Tabla 14 Diseño de una red neuronal Model_básico4 65

Tabla 15 Diseño de una red neuronal de tipo GRU Model_GRU 67

Tabla 16 Diseño de una red neuronal de tipo GRU Model_GRU1 68

Tabla 17 Diseño de una red neuronal de tipo GRU Model_GRU2 70

Tabla 18 Diseño de una red neuronal de tipo GRU Model_GRU3 71

Tabla 19 Diseño de una red neuronal de tipo LSMT Model_LSTM 74

Tabla 20 Diseño de una red neuronal de tipo LSMT Model_LSTM1 76

Tabla 21 Métricas de evaluación de los modelos RNA en escala original.....94

ÍNDICE DE FIGURAS

Figura 1 Motor de propulsión, turbo reactor o turbofán 11

Figura 2 Neurona Biológica 15

Figura 3 Modelo de una red neuronal RNA básica 16

Figura 4 Funciones de activación 20

Figura 5 Red neuronal artificial RNA monocapa 21

Figura 6 Red neuronal artificial RNA multicapa 22

Figura 7 Red neuronal artificial RNA recurrente 24

Figura 8 Página WEB del Centro de investigación AMES DE la NASA 39

Figura 9 Diagrama de flujo para el desarrollo de la red neuronal artificial RNA 42

Figura 10 Porcentaje data train con respecto a Total train 48

Figura 11 Porcentaje data test por cada archivo con respecto a total test 49

Figura 12 Datos de RUL (Vida útil restante) 50

Figura 13 Cantidad de datos de los archivos combinados 51

Figura 14 Distribución de RUL para valores Atípicos y No atípicos 55

Figura 15 Evolución de la pérdida durante el entrenamiento Model_básico 58

Figura 16 Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_básico 59

Figura 17 Evolución de la pérdida durante el entrenamiento Model_básico1 ..60

Figura 18 Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_básico1 60

Figura 19 Evolución de la pérdida durante el entrenamiento Model_básico2 ..62

Figura 20 Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_básico2 62

Figura 21 Evolución de la pérdida durante el entrenamiento Model_básico3 ..63

Figura 22 Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_básico3 64

Figura 23 Evolución de la pérdida durante el entrenamiento Model_básico4 ..65

Figura 24 Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_básico4 66

Figura 25 Evolución de la pérdida durante el entrenamiento Model_GRU..... 67

Figura 26 Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_GRU 68

Figura 27 Evolución de la pérdida durante el entrenamiento Model_GRU1..... 69

Figura 28 Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_GRU1 69

Figura 29 Evolución de la pérdida durante el entrenamiento Model_GRU2..... 71

Figura 30 Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_GRU2 72

Figura 31 Evolución de la pérdida durante el entrenamiento Model_GRU3..... 72

Figura 32 Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_GRU3 72

Figura 33 Evolución de la pérdida durante el entrenamiento Model_LSTM 74

Figura 34 Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_LSTM..... 75

Figura 35 Evolución de la pérdida durante el entrenamiento Model_LSTM1 ...77

Figura 36 Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_LSTM1..... 77

Figura 37 Comparación entre el RUL Real y Predicted Rul del Model_básico.80

Figura 38 Comparación entre el RUL Real y Predicted Rul del model_básico1 81

Figura 39 Comparación entre el RUL Real y Predicted Rul del model_básico2 83

Figura 40 Comparación entre el RUL Real y Predicted Rul del model_básico3 84

Figura 41 Comparación entre el RUL Real y Predicted Rul del model_básico4 85

Figura 42 Comparación entre el RUL Real y Predicted Rul del Model_GRU ...86

Figura 43 Comparación entre el RUL Real y Predicted Rul del Model_GRU1 .88

Figura 44 Comparación entre el RUL Real y Predicted Rul del Model_GRU2 .89

Figura 45 Comparación entre el RUL Real y Predicted Rul del Model_GRU3 .90

Figura 46 Comparación entre el RUL Real y Predicted Rul del Model_LSTM .91

Figura 47 Comparación entre el RUL Real y Predicted Rul del Model_LSTM1 93

Figura 48 Análisis del rendimiento los modelos en base al MAE y RMSE en escala original 95

Figura 49 Análisis del rendimiento los modelos en base al porcentaje de error de MAE y RMSE 95

Figura 50 Interpretación del R^2 con respecto al porcentaje de error de MAE y RMSE 96

Figura 51 Análisis del rendimiento de los modelos en base a la precisión de MAE y RMSE.....97

ÍNDICE DE ANEXOS

Anexo A. Certificado de aprobación del abstract por parte del centro de idiomas 105
Anexo B. Estadísticas descriptivas de los archivos de la base de datos 107
Anexo C. Cantidad de datos por cada archivo ……..118

RESUMEN

La estimación precisa de la vida útil restante (Remaining Useful Life, RUL) es fundamental para optimizar las estrategias de mantenimiento predictivo y garantizar la seguridad operativa en la industria aeroespacial. La presente investigación desarrolló un modelo basado en redes neuronales artificiales para predecir el RUL de turborreactores de doble flujo, bajo un enfoque cuantitativo de tipo exploratorio. Se diseñaron y entrenaron diversas arquitecturas de redes neuronales, que incluyeron desde modelos básicos con capas densas hasta estructuras avanzadas como GRU y LSTM. La base de datos utilizada corresponde al repositorio oficial de la NASA, conformado por archivos de entrenamiento, prueba y RUL. Para mejorar la eficiencia de los modelos, se aplicó un preprocesamiento de los conjuntos de datos y, de manera paralela a la evaluación de los modelos, se realizaron ajustes iterativos de parámetros e hiperparámetros, además de modificaciones en la arquitectura y en las funciones de activación. Los resultados muestran que, aunque es posible predecir la vida útil restante de los turborreactores de doble flujo mediante redes neuronales, la precisión alcanzada no cumple con los estándares requeridos en la aviación civil y militar, donde se exige un margen de error de al menos ±5 ciclos y un coeficiente de determinación (R^2) igual o superior a 0,95. El modelo con mejor desempeño alcanzó una precisión de ±20 ciclos y un coeficiente de determinación de 0,73. En consecuencia, las redes neuronales aplicadas con esta base de datos no resultaron lo suficientemente efectivas. Se recomienda reformular el parámetro del RUL en unidades alternativas de tiempo, lo cual podría facilitar la modelación de series temporales y, en consecuencia, mejorar la exactitud de las predicciones.

Palabras clave: RUL, Red Neuronal Artificial, Dropout, Mantenimiento, Predicción.

ABSTRACT

Accurate estimation of Remaining Useful Life (RUL) is fundamental for optimizing predictive maintenance strategies and ensuring operational safety in the aerospace industry. This research developed an artificial neural network-based model to predict the RUL of turbofan engines, using an exploratory quantitative approach. Various neural network architectures were designed and trained, ranging from basic models with dense layers to advanced structures such as GRU and LSTM. The database used is the official NASA repository, consisting of training, test, and RUL files. To improve the models' efficiency, the datasets were preprocessed, and in parallel with model evaluation, iterative adjustments of parameters and hyperparameters were performed, along with modifications to the architecture and activation functions. The results show that, although it is possible to predict the remaining useful life of turbofan engines using neural networks, the accuracy achieved does not meet the standards required in civil and military aviation, where an error margin of at least ±5 cycles and a coefficient of determination (R^2) of 0.95 or higher are required. The best-performing model achieved an accuracy of ±20 cycles and a coefficient of determination of 0.73. Consequently, the neural networks applied to this dataset did not prove effective enough. It is recommended to reformulate the RUL parameter in alternative time units, which could facilitate time series modeling and, consequently, improve prediction accuracy.

Keywords: RUL, Artificial Neural Network, Dropout, Maintenance, Prediction.

CAPÍTULO I

PROBLEMA

1.1. Planteamiento del problema

En la industria aeronáutica mundial, los turborreactores de doble flujo son ampliamente utilizados en áreas como la aviación civil, aviación militar y en aeroespacial, debido en gran parte a las numerosas ventajas que presentan estos dispositivos, como el ahorro significativo de combustible, no obstante, el desempeño seguro y eficiente de estos dispositivos es fundamental para las aplicaciones antes mencionadas, por lo tanto, uno de los desafíos es lograr predecir la vida útil restante, considerando que el deterioro de sus elementos constitutivos no sigue modelos lineales y se ven influenciados por múltiples factores.

Los Turbo reactores de doble flujo requieren una fiabilidad cercana al 100 %, un desperfecto de este dispositivo podría ocasionar fallos catastróficos, que a su vez derivan en costos de mantenimiento elevados, tiempos de inactividad no planificados y, lo que es más crítico, riesgos para la seguridad de las operaciones (Widodo *et al*., 2025).

A nivel regional la aerolínea Avianca cuenta con un centro de mantenimiento conocido como MRO (Maintenance Repair and Overhaul) en Rionegro Perú, en el cual se realiza mantenimiento predictivo y correctivo con estrategias ya conocidas de la flota de aviones de esta aerolínea, a pesar de contar con una tecnología de punta no se evidencia que hagan uso de las herramientas Machine Learning (Aprendizaje de máquina), o Redes Neuronales Artificiales que permitan anticiparse a fallas o a su vez predecir su vida útil residual (Larenas, 2018).

Por otro lado, la aerolínea latinoamericana Latam cuenta con un MRO (Maintenance Repair and Overhaul), en donde se realiza labores de mantenimiento, además, se fabrica piezas que sirven de repuestos, no obstante, en esta empresa tampoco se

aplica, herramientas de la inteligencia artificial que permita predecir fallos o su vida útil restante (Latam Airlines, 2024).

En este contexto se puede decir que, a nivel de la región la totalidad de las empresas optan por utilizar métodos tradicionales para el mantenimiento de sus aeronaves, convirtiéndose en un proceso poco eficiente, además, es impensable que estos métodos puedan predecir con un alto nivel de exactitud la vida útil restante del dispositivo.

A nivel nacional el empleo de redes neuronales artificiales para predecir fallas de turborreactores de doble flujo es poco conocido, la mayoría de las aerolíneas realizan mantenimientos correctivos de emergencia en los aeropuertos, incluso los mantenimientos preventivos y predictivos se realizan fuera del país, por lo tanto, no se puede hablar del empleo de la inteligencia artificial para predecir la vida útil restante de estos dispositivos, eso sugiere la bibliografía disponible hasta el momento.

No obstante, a nivel nacional las investigaciones se han ido incrementando en la predicción de fallas mediante redes neuronales y otros mecanismos. Contreras Urgiles et al., (2018) exponen que se ha logrado exactitudes muy bajas de estos modelos con aplicaciones en el diagnóstico de fallas mecánicas en motores de combustión interna, lo que resulta poco satisfactorio para turborreactores de doble flujo en donde se requiere una precisión muy alta debido a la aplicación en que están inmersos estos elementos.

Con base en lo descrito surge la siguiente interrogante ¿Se puede desarrollar un modelo de redes neuronales artificiales que permita predecir con exactitud la vida útil restante de turborreactores de doble flujo, de tal manera que se pueda optimizar la gestión del mantenimiento y por ende reduciendo riesgos asociados a fallas imprevistas?

1.2. Preguntas de Investigación o Hipótesis

1.2.1 Pregunta general de investigación

¿Puede un modelo de red neuronal artificial predecir con un alto nivel de precisión la vida útil restante de turborreactores de doble flujo, empleando una base de datos del centro de excelencia de pronósticos AMES de la NASA?

1.2.2 Preguntas específicas de investigación

- ¿Qué arquitectura de red neuronal artificial ofrece el mejor desempeño en la predicción de la vida útil restante de turborreactores de doble flujo, utilizando los datos proporcionados por el Centro de Excelencia de Pronósticos Ames?

- ¿Cómo influye la selección y ajuste de parámetros e hiperparámetros en la capacidad de generalización y precisión de las predicciones realizadas por los modelos de redes neuronales artificiales?

- ¿Qué métricas cuantitativas son más adecuadas para evaluar el rendimiento predictivo de los modelos entrenados, y cómo se comparan entre distintas configuraciones de redes neuronales?

1.3. Objetivos de investigación

1.3.1 Objetivo General

Desarrollar un modelo de predicción para estimar la vida útil restante de turborreactores de doble flujo, empleando redes neuronales artificiales, en base a un conjunto de datos proporcionado por el Centro de Excelencia de Pronósticos Ames de la NASA.

1.3.2 Objetivos Específicos

- Seleccionar diversas arquitecturas de red neuronal artificial, valorando configuraciones y algoritmos de aprendizaje supervisado.

- Entrenar los modelos a través de técnicas de aprendizaje supervisado, ajustando parámetros e Hiperparámetros hasta alcanzar una adecuada generalización de los datos, de modo que el modelo pueda realizar predicciones precisas y eficaces en un nuevo conjunto de datos.

- Evaluar el rendimiento de los modelos mediante métricas cuantitativas para determinar su capacidad predictiva.

1.4. Justificación

Las aplicaciones de los turborreactores de doble flujo cubren el ámbito comercial, militar y aeronáutico, por lo tanto, la predicción exacta de la vida útil restante de estos dispositivos es una necesidad prioritaria, debido a que la eficiencia y seguridad de las aeronaves que tienen a su a ver está condicionando por el óptimo funcionamiento de los turborreactores, en este contexto, las metodologías tradicionales de mantenimiento resultan útiles pero no son suficientes, puesto que no consiguen anticipar una falla de forma exacta, lo cual se traduce en altos costos operativos por fallas imprevistas y el riesgo de catástrofes.

Las redes neuronales artificiales (RNA) presentan una gran versatilidad logrando analizar gran cantidad de datos e identificar patrones de deterioro de elementos de máquinas, superando por completo modelos estadísticos tradicionales que se emplean comúnmente en metodologías del mantenimiento en la actualidad, el modelo en cuestión que busca predecir la vida útil restante del dispositivo permitirá optimizar tareas de mantenimiento preventivo y aumentar la fiabilidad de los turborreactores de doble flujo.

Además, la presente investigación tiene relevancia en el desarrollo de nuevas estrategias para mejorar la planificación operativa y optimizar la gestión del mantenimiento, lo cual se traduce en una reducción sustancial de los recursos empleados, logrando maximizar la vida útil restante del dispositivo. La integración del modelo en la industria aeronáutica, civil o militar permitirá un cambio sustancial en la forma que se gestionan los activos físicos en los departamentos de mantenimiento asociados.

De igual forma, el empleo de este modelo será de gran utilidad en la seguridad operacional, puesto que poder anticipar fallas críticas puede asegurar una operación óptima de los turborreactores de doble flujo, logrando reducir riesgos asociados al funcionamiento del motor de Propulsión. El proyecto también favorecerá al adelanto científico-tecnológico, al concentrar nuevas técnicas de inteligencia artificial (IA) en el

mantenimiento de dispositivos de gran complejidad, formando conocimientos que se pueden emplear en otros sectores industriales.

El presente trabajo de investigación se enfoca en mejorar la fiabilidad de los turborreactores de doble flujo, a través de un enfoque innovador, siguiendo la línea de la inteligencia artificial (IA), que tiene como base imitar una red de neuronas capaces de aprender y con el tiempo analizar patrones no lineales complejos, permitiendo predecir la vida útil restante del dispositivo en cuestión, lo cual aportará beneficios económicos a mediano plazo a las industrias del sector que opten por aplicar estos modelos.

La presente investigación se enfoca en el uso de una de las ramas de la inteligencia artificial (IA), como son las redes neuronales artificiales a fin de predecir la vida útil restante de turborreactores de doble flujo, la cual se alinea con los ODS 9 (industria innovación e infraestructura), ya que se prevé estudiar la aplicación de las RNA como una estrategia que permita optimizar la gestión del mantenimiento reduciendo gastos derivados de fallos imprevistos. Además, contribuye al Plan Nacional de Desarrollo Para El Nuevo Ecuador 2024-2025, al fomentar la innovación tecnológica en el área de mantenimiento industrial, que a su vez forma parte de investigación de la UPEC, buscando soluciones innovadoras y rentables para el sector industrial y aeronáutico.

CAPÍTULO II

FUNDAMENTACIÓN TEÓRICA

2.1. Antecedentes de la investigación

Los turborreactores de doble flujo son componentes críticos en sus distintas aplicaciones como la aviación civil, militar y la propulsión aeroespacial. La determinación de la vida útil restante de estos dispositivos es de suma importancia para garantizar la seguridad operativa, optimizar el mantenimiento y evitar fallos imprevistos que pueden resultar catastróficos, en este contexto la inteligencia artificial (IA) es un gran aliado para lograr este cometido. A continuación, se revisaron algunos estudios que se han enfocado en el empleo (IA) y en especial las Redes Neuronales Artificiales (RNA) para predecir fallos o vida útil restante en diferentes materiales, equipos, máquinas y mecanismos.

El artículo publicado por Lemache *et al*. (2023) plantea como objetivo una comparación de modelos de aprendizaje de turborreactores de doble flujo a partir de un gran volumen de datos obtenidos del portal de la NASA, para tal efecto se hizo uso del modelo Random Forest optimizado con f1 score, obteniéndose resultados favorables con una precisión cercana al 100% en la predicción de fallas, además, el estudio sugiere que la creación de un modelo esquematizado para la predicción de fallos imprevistos en diversos en diversos áreas, sin embargo, hace énfasis en la importancia de emplear data confiable bajo diversas condiciones operativas de los elementos.

El estudio realizado por Carroll *et al*. (2019) se enfoca en la predicción de fallas y vida útil restante del tren de engranes empleadas en turbinas eólicas, mediante redes neuronales artificiales (RNA), para tal efecto se empleó una gran cantidad de data obtenidos a partir del control, supervisión y adquisición de datos (SCADA) y las vibraciones producidas por estos dispositivos. Los resultados obtenidos exponen un logro importante en la investigación, pudiendo anticiparse a los fallos hasta con un

mes de anticipación considerando una precisión casi del 100%, el estudio concluye resaltando la importancia de la data de vibraciones, debido a que se logra aumentar significativamente el tiempo de anticipación a la falla, mejorando así la fiabilidad del modelo.

La inteligencia artificial también ha incursionado en la industria petrolera, llevándose a cabo algunas investigaciones, como la realizada por Kumar *et al*. (2022) quien sugiere la aplicación de redes neuronales artificiales (RNA) como un método empírico en la predicción de fallas en tuberías API 56 X80, con referencias por defectos de corrosión, dicho trabajo se plantea como una alternativa a métodos convencionales y ampliamente difundidos que hoy en día resultan poco eficientes o a su vez costosos, para el entrenamiento de la red neuronal artificial se empleó el análisis de elementos finitos para diversas características de las tuberías, al final se evidenció un coeficiente de precisión de 0,99, por lo tanto, se logró demostrar que el modelo propuesto resulta útil y eficiente en la predicción de fallas en este tipo de elementos.

La aplicación de la inteligencia artificial (IA) en ingeniería ha sido ampliamente explorada, un estudio realizado por Shaheen *et al*. (2023), quien aborda la predicción de fallas basada en datos y estimación de la vida útil restante de componentes mecánicos a través de redes neuronales artificiales (RNA) de tipo acumulativas. En dicho trabajo se expone la necesidad de buscar nuevas técnicas o métodos que permita predecir fallos en virtud de optimizar la programación y planificación de las tareas de mantenimiento, lo cual se traduce en un aumento importante de la fiabilidad de los equipos, el estudio propone la predicción de fallas en función a una curva de degradación con un punto predefinido de previsión, para lo cual se empleó un modelo de redes neuronales artificiales con aprendizaje supervisado, los resultados obtenidos fueron concluyentes logrando predecir fallos con una precisión cercana al 100%.

Los motores son esenciales en cualquier tipo de industria, a su vez dispositivos susceptibles a entrar en fallo de forma inesperada, por lo tanto, es necesario buscar técnicas o métodos que permitan predecirlos, varios investigadores han explorado el

campo de la (IA) como una alternativa, en el trabajo realizado por Sampaio *et al*. (2019) el cual centra su estudio en predecir un tiempo de falla en el cual se espera que un motor de tipo eléctrico entre en fallo a través de redes neuronales artificiales (RNA), para ello hace uso de datos recopilados de un acelerómetro capaz de medir las vibraciones producidas por la máquina, los resultados arrojados muestran el modelo con una capacidad de predecir fallos de estos dispositivos con una precisión del 98%, siendo útil para mejorar la programación del mantenimiento en la industria.

En el mismo contexto, un estudio realizado por Amuzuvi y Warden (2020), analiza motores eléctricos de inducción magnética en condicion de trabajo severo para predecir fallas mediante RNA, los datos empleados se obtuvieron in situ en operación activa de la máquina. Se empleó el software MATLAB para desarrollar el modelo, además, se hizo uso de las funciones de Levenberg-Marquardt y regularización bayesiana, al final se obtuvo resultados favorables, teniendo un coeficiente de 0,98, es decir, la tasa de éxito es bastante alta, el trabajo concluye describiendo la importancia de este modelo en la predicción de fallas en este tipo de motores.

Varios estudios han denotado la importancia de la IA en ingeniería, los compuestos poliméricos no están exentos de aquello, el trabajo realizado por Wan *et al*. (2023) analiza la probabilidad de detección de fallas en este tipo de materiales, mediante el empleo de elementos finitos y haciendo uso de ciertas caracteristicas de degradación típico de estos compuestos, además, se consideró un total de 15 combinaciones biaxiales para mayor precisión del modelo, los resultados obtenidos se describen en función al número de neuronas y de capas ocultas, en tal virtud se confirma que para este caso en particular cuanto mayor sea el número de capas ocultas el desempeño del modelo en la predicción es mayor, acercándose casi al 100%, la investigación concluye la importancia de realizar un análisis para cada caso en función a la data disponible, ya que no siempre un mayor número de capas denota mayor precisión del modelo.

Hasta el momento se ha visto la aplicación de la inteligencia artificial (IA) en aplicaciones puntuales, sin embargo, el estudio realizado por Schneller *et al*. (2022)

aborda la aplicación de RNA en la predicción de fallas por fatiga de componentes mecanizados por arranque de viruta fabricados en acero para diversas aleaciones, para lo cual considera tres aspectos básicos como son: tensión a la carga, dureza del material y el tamaño de defecto letal. El desarrollo del modelo y los Hiperparámetros fueron optimizados mediante una validación cruzada, que ayuda notablemente en el aumento de la exactitud, los resultados obtenidos denotan una precisión superior al 91% de predicción de los materiales que tienden a fallas por fatiga, finalmente, se concluye exponiendo sobre la fiabilidad de la predicción de fallas de elementos mecánicos con la ayuda de la inteligencia artificial.

Los estudios de aplicación de inteligencia IA también ha explorado el sector minero, prueba de ello es el trabajo realizado por Rafezi y Hassani, 2023, quienes buscaban predecir fallas con miras a una automatización minera, para ello se recopiló los datos in situ buscando las características más relevantes que pudieran influir en el deterioro de los elementos mecánicos de las brocas empleadas para la perforación, una vez desarrollado y entrenado el modelo se pudo comprobar que podía predecir fallas con una tasa de éxito superior al 70%, considerando las condiciones severas de trabajo esta aproximación es superior a cualquier otro método de mantenimiento predictivo empleado en minería.

La predicción de fallas en oleoductos o gasoductos mediante IA se ha estudiado por diversas instituciones e investigadores autónomos, Al-Sabaeei *et al*. (2023) realiza una revisión sistemática de la eficacia de los modelos de RNA realizados para este campo específico, a lo largo del artículo se repite con frecuencia la capacidad de predecir fallas de diversa índole en este tipo de sistemas con una precisión superior en comparación con métodos convencionales, sin embargo, también se recomienda realizar múltiples combinaciones con los algoritmos disponibles lo cual ha demostrado en varios estudios mejorar notablemente la tasa de éxito en las predicciones de falla. El trabajo termina destacando que la precisión de los modelos está fuertemente influenciada por ciertos factores ajenos a esta industria que tienden a limitar la capacidad de predicción con redes neuronales artificiales.

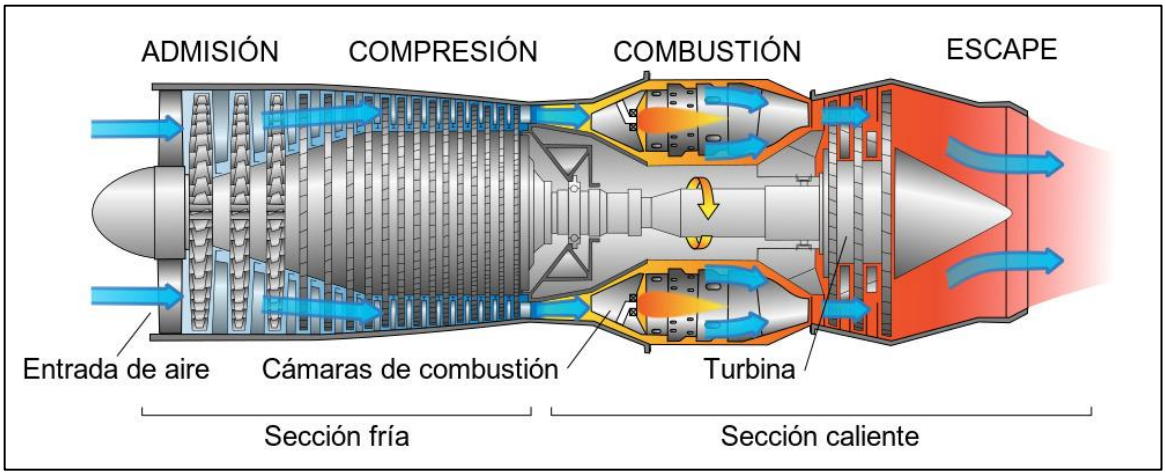
2.2. Marco teórico

2.2.1 Turborreactores de doble flujo

Los turborreactores de doble flujo o también conocidos como turbofán, son ampliamente empleados en áreas como la aviación civil y militar, debido a sus múltiples ventajas principalmente el ahorro de combustible que representa, las nuevas tecnologías y estrategias de mejora y repotenciación han permitido progresos notables en el rendimiento de estos motores de propulsión capaces de alcanzar velocidades subsónicas y supersónicas.

Figura 1

Motor de propulsión, turbo reactor o turbofán



Nota: en la Figura se puede apreciar los procesos al interior del núcleo del motor por donde circula un primer flujo de aire, además, se aprecia las secciones frías y calientes, antes y después de la combustión. Fuente Kaiser *et al*. (2022)

En términos generales se puede describir un turborreactor como una máquina térmica con capacidad de producir empuje gracias al aceleramiento de dos flujos de aire, un primer flujo se dirige al núcleo en donde atraviesa diversos componentes que le permiten generar empuje al pasar por la turbina y el escape y debido a la alta temperatura de los fluidos gaseosos producto de la combustión, entre tanto, el segundo flujo se direcciona por un ducto externo, lo cual genera empuje adicional,

este último proceso permite un aumento considerable en la eficiencia del motor (Kaiser *et al*., 2022). La Figura 1 ilustra de forma dinámica este proceso de tránsito del aire por el interior del propulsor hasta su salida.

2.2.2 Componentes principales y principio de funcionamiento del turbofán

El motor de propulsión es una máquina compleja, compuesta por una serie de dispositivos mecánicos, eléctricos y electrónicos capaces de funcionar en armonía y que en su conjunto generan empuje capaz de mover aeronaves de diversa índole, no obstante, para términos prácticos se centrará en describir sus seis principales componentes que hacen posible su funcionamiento.

De acuerdo con Contreras *et al*. (2018) los componentes principales de un turborreactor de doble flujo o turbofán son los siguientes:

- **Fan. -** es un componente que hace las veces de ventilador, su ubicación estratégica en la parte delantera del motor de propulsión le permite mover grandes volúmenes de aire.
- **Compresores. –** formado por un conjunto de alavés que tienen por objeto comprimir el primer flujo de aire previo a la combustión**.**
- **Cámara de combustión. –** ubicado en el centro del turborreactor, donde se combinan una serie de gases que producen una explosión controlada, dando como resultados fluidos gaseosos a alta temperatura.
- **Turbina. –** con la ayuda de la energía de los fluidos gaseosos a alta temperatura, tiene capacidad de mover el ventilador y compresor**.**
- **Escape. –** componente encargado de desprender los gases a alta velocidad con lo cual se consigue empuje adicional.
- **Bypass. –** ducto que tiene por objeto redireccionar un segundo flujo de aire, con capacidad de generar empuje, aumentando considerablemente la eficiencia del motor de propulsión.

Con los componentes que se describen anteriormente el turborreactor de doble flujo, basa su funcionamiento en dos corrientes de aire que al ingresar al motor toman dos direcciones posibles, una de ellas se dirige al núcleo hasta su combustión, a la salida se expulsa una mezcla de diferentes gases producto del proceso anterior, a muy alta velocidad, dando origen al primer empuje, otra corriente de aire se direcciona por el fan o bypass característico del motor que hace las veces de ventilador, generando un segundo empuje.

El funcionamiento característico del turborreactor de doble flujo le hace acreedor de una serie de ventajas en comparación con los turborreactores de un solo flujo, una de ellas es mayor eficiencia en vuelos subsónicos lo que los hace ideales para aviones comerciales que recorren grandes distancias, el combustible es mucho menor debido a los dos flujos de aire correctamente direccionados permite generar mayor empuje con menor consumo, al canalizar una segunda corriente de aire frio mediante un bypass se logra disminuir notablemente el ruido, finalmente, genera menor contaminación al reducir el consumo de combustible.

Las múltiples ventajas del turborreactor de doble flujo lo hacen ideal para aplicaciones comerciales, aerolíneas como Airbus y Boeing han optado por este motor de propulsión para sus modelos A320 y 747 respectivamente. Además, en la industria militar los modelos F22 Raptor y el casa de quinta generación F35 Lightning llevan incorporados este tipo de motores debido a su potencia y versatilidad en velocidades supersónicas (Kaiser *et al*., 2022).

2.2.3 Modelos de predicción de RUL

La vida útil restante o remanente es un aspecto de suma importancia en varias industrias, como la aeronáutica, manufactura, entre otras. Además, es esencial en la gestión y administración del mantenimiento, estos métodos tienen a su haber predecir el tiempo restante de un determinado mecanismo antes de un fallo, lo cual permite intervenciones oportunas que maximizan la seguridad operativa con un costo ahorro en recursos.

2.2.4 Modelos de predicción basados en Física

Estos modelos emplean ecuaciones de determinación a fin de tasar la evolución de un fallo progresivo, estos modelos en particular requieren de un entendimiento casi perfecto del mecanismo y su degradación de lo contrario los resultados pueden resultar erróneos. La combinación armónica de estos modelos juntamente con técnicas de modelado es ideales para estimar el desgaste de componentes principalmente mecánicos. No obstante, le efectividad de este método está condicionada por el entendimiento de la máquina en cuestión (Cuesta *et al*., 2025).

2.2.5 Modelos de predicción basados en Redes Neuronales Artificiales

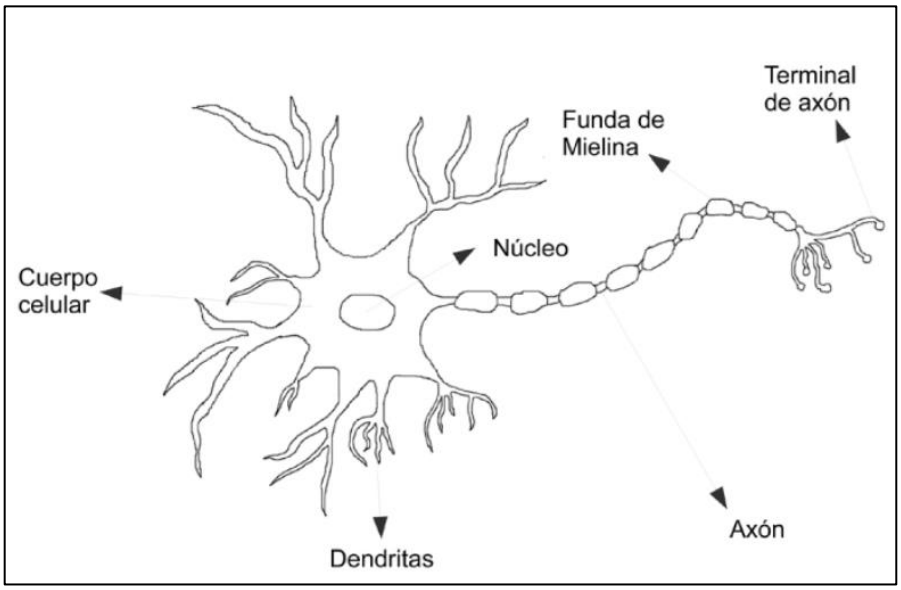
Antes de hablar sobre redes neuronales artificiales, se debe referir a una red neuronal biológica, que se compone de un número indeterminado de neuronas enlazadas entre sí, quienes a su vez conforman la estructura cerebral de un ser humano, cada neurona se estructura por tres componentes fundamentales como son: el axón, las dendritas y el cuerpo neuronal.

La capacidad de las neuronas de establecer una comunicación entre ellas mediante impulsos eléctricos tiende a facilitar los procesos de aprendizaje, este suceso tiende a ocurrir debido a la interacción entorno e individuo, esta serie de procesos logra adaptar estímulos provenientes del exterior favoreciendo la cognición

El proceso de comunicación entre neuronas inicia con una estimulación que un individuo logra percibir a través de estímulos que se percibe mediante los sentidos, esto genera señales electroquímicas que distinguen las dendritas, que dan paso al cuerpo celular, quien se encarga de procesar, distinguir, procesar y establecer umbrales efectivos, finalmente, el axón es el encargado de transferir a las neuronas interconectadas, en la Figura 2 se puede observar de forma clara una neurona biológica (Restrepo *et al*., 2021).

Figura 2

Neurona Biológica



Nota: en la Figura se puede apreciar los componentes de una neurona biológica, que al estar conectada con miles de millones de similares generan comunicación y forma la compleja estructura del cerebro humano. Fuente Restrepo *et al*. (2021)

El propósito de conocer el funcionamiento de una red neuronal biológica es debido a que una red neuronal artificial trata de emular dicho funcionamiento, en término general se puede describir una RNA como un algoritmo capaz de ajustar parámetros buscando la mejor solución o alternativas a problemáticas dadas en base a unos datos de entrada, y mediante el empleo de un software informático (Restrepo *et al*., 2021).

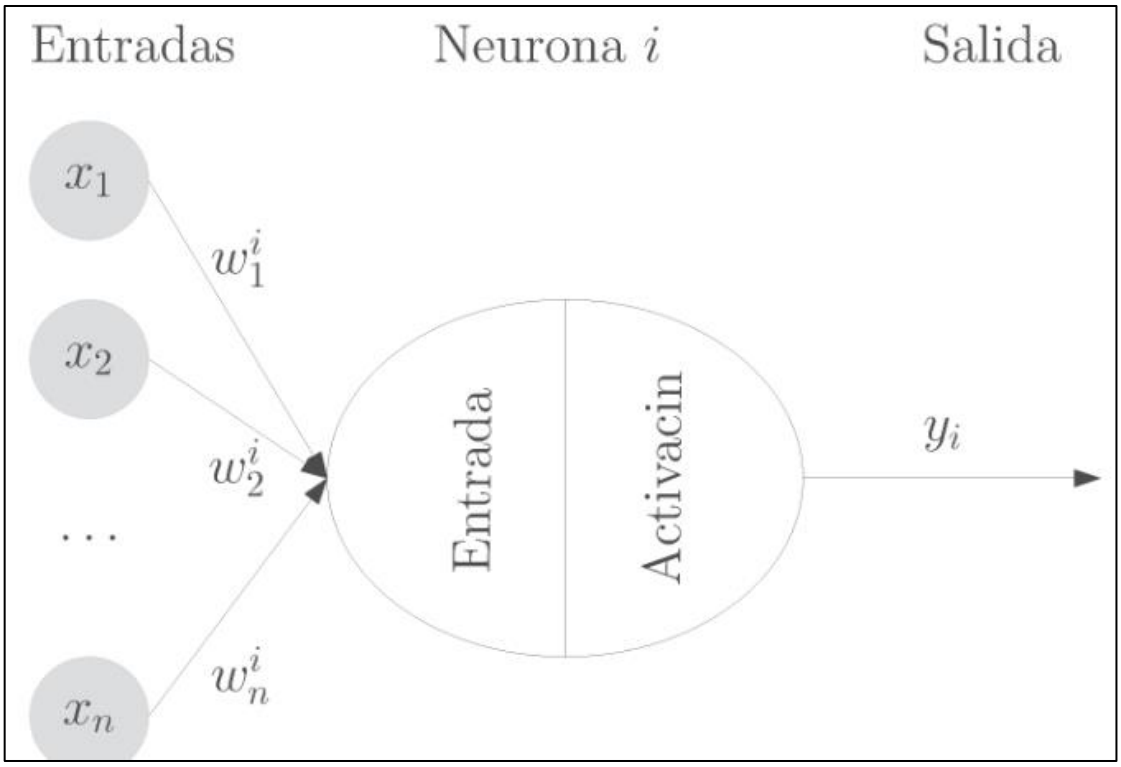
Una red neuronal de forma básica se compone de un enlace de estas quienes aplican una función a los datos provenientes de otras neuronas anteriores o en su defecto a los valores de entrada de la red, y trasmitir a las neuronas subsiguientes o su vez enviar esta información como salida o resultado, en función de su arquitectura.

La Figura 3 ilustra una red neuronal básica, que se compone de unas entradas asignadas con la letra “X”, a su vez cada entrada se le establece una importancia o también llamado peso sináptico “W”, cada neurona individualmente realiza ciertas operaciones de acuerdo a una función de entrada, el resultado de dicha acción se procesa a través de otra función llamada de activación, que tiene como propósito

armonizar el resultado que se visualizará a la salida de la red neuronal o se transmitirá hacia otras neuronas en función al número de capas constitutivas (Bosch *et al*., 2020).

Figura 3

Modelo de una red neuronal RNA básica



Nota: en la Figura se puede denotar un modelo básico de RNA, donde se puede apreciar los componentes de una red neuronal artificial, como: entrada (x), peso sináptico (w), umbral o bias, estímulo, función de activación y salida (y). Fuente: (Bosch *et al*., 2020)

Función de entrada

También conocida como función de combinación se encarga de concertar las entradas con el peso sináptico, y en el caso de las redes multicapa añadir valores alcanzados en los diversos enlaces o conexiones a fin de generar un único valor a transmitir como resultado “Y” o la subsiguiente neurona. Cabe resaltar que, el número de entradas “X” a la red tiene la misma cantidad de pesos sinápticos “W”. más adelante se describen estas funciones.

- **Función suma ponderada**

$$F(\mathrm{x}) = \sum_{j=1}^{n} X_j \, W_j^i \tag{1}$$

- **Función máxima**

$$F(\mathrm{x}) = \max(X_1 \, W_1^n \ldots . \; X_n W_n^n) \tag{2}$$

- **Función mínima**

$$F(\mathrm{x}) = \min(X_1 \, W_1^n \ldots . \; X_n W_n^n) \tag{3}$$

- **Función lógica AND y OR**

$$F(\mathrm{x}) = (X_1 \, W_1^n \; \wedge \ldots . \wedge X_n W_n^n) \tag{4}$$

$$F(\mathrm{x}) = (X_1 \, W_1^n \; \vee \ldots . \vee X_n W_n^n) \tag{5}$$

La aplicación de una u otra función es en base al criterio del problema planteado, sin embargo, en la mayoría de los casos se ha evidenciado el uso de la función suma ponderada, y las funciones lógicas and u or se emplean únicamente en casos donde los problemas su ajustan a salidas binarias (Bosch *et al*., 2020).

Función de activación

La función de activación o también conocida por varios autores como de transferencias se encarga de variar los valores generados por la función de combinación, para posteriormente transferirlos a otra neurona o bien generar la salida o resultado de la red neuronal.

Esta función se considera de suma importancia ya que si no se hace uso de esta se hablaría únicamente de un incremento lineal del resultado basados en una entrada, caso que corresponde más a una programación que al aprendizaje automático. Más adelante se describen las ecuaciones con su respectivo significado de las funciones más utilizadas para transferencia o activación.

- **Función escalón. –** la función escalón unitario genera resultados en el rango de [-1, 1], sin embargo, también lo hace en el rango de [0,1], considerando el valor del umbral o sesgo representado por "α" en la siguiente expresión matemática se describe esta particularidad.

$$\mathrm{F(X)} = \begin{cases} 1\ si\ X\ \geq\ \alpha \\ -1\ SI\ X\ \leq\ \alpha \end{cases} \tag{6}$$

- **Función lineal. –** es la función más sencilla de todas, incrementa de manera proporcional el valor de salida o resultado de la red a medida que lo hace la función de entrada o combinación de la neurona antecesora, la ecuación representativa es la siguiente:

$$\mathrm{F(X)} = Bx \tag{7}$$

- **Función sigmoide. -** también conocida como función logística, entre sus características incluye un factor representado por la "ρ" que determina la forma de la curva a obtener, con lo cual se consigue valores suavizados a la salida**.** La expresión matemática es la siguiente**:**

$$\mathrm{F(X)} = \frac{1}{1 + e^{\frac{-X}{\rho}}} \tag{8}$$

- **Función tangente hiperbólica. –** esta función al igual que la sigmoide comparten una característica relevante, es que presentan una tasa de cambio menor para valores extremos y mayor en datos intermedios, su representación matemática es la siguiente:

$$\mathrm{F(X)} = \mathrm{Tanh\ (x)} = \frac{senh(X)}{\cosh(X)} = \frac{1 - e^{-2X}}{1 + e^{-2X}} \quad (9)$$

- **Función rectificadora. –** los valores de esta función se obtienen a partir de los datos positivos del argumento (Bosch *et al*., 2020).

$$\mathrm{F(X)} = \max(0, X) \quad (10)$$

La selección de las funciones de activación, así como de las funciones de entrada o combinación, está determinada por la naturaleza del problema planteado. Esta elección resulta fundamental en el diseño de modelos basados en redes neuronales artificiales, ya que influye directamente en su desempeño. Por ello, es imprescindible realizar un análisis previo de las características y tendencias del modelo, considerando tanto las propiedades del conjunto de datos como los objetivos específicos del problema.

Por otro lado, se debe considerar el entendimiento de las funciones disponibles a fin de evaluar su funcionamiento bajo ciertos parámetros en casos de estudio real, esto tenderá a favorecer notablemente el rendimiento en el problema planteado logrando una selección correcta.

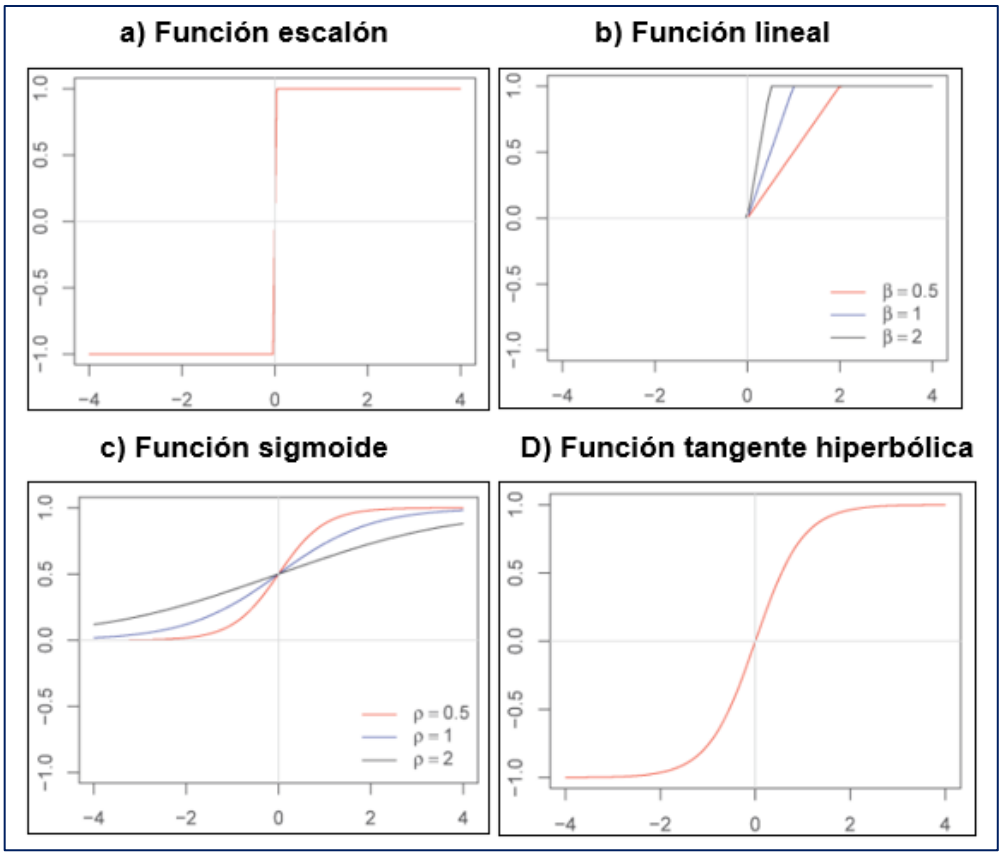
Las redes neuronales artificiales (RNA) están previstas para su aplicación en problemáticas de patrones no lineales, de alta complejidad, ya que pueden modelar funciones que tienden a seguir trayectorias diferentes a las de una línea recta, en base a esta descripción se puede afirmar que las funciones de activación juegan un papel indispensable en la resolución de estos problemas.

Sin embargo, en el ámbito del aprendizaje profundo, la función rectificadora, también conocida como ReLU (Rectified Linear Unit) y popularizada en 2011, se ha consolidado como una de las más empleadas. Esto se debe a que ofrece ventajas significativas en el entrenamiento de modelos de RNA, como la mitigación del

problema de desvanecimiento del gradiente y una mayor eficiencia computacional, lo que la hace especialmente adecuada para redes profundas (Bosch *et al*., 2020).

Figura 4

Funciones de activación



Nota: en la Figura se puede observar las funciones de activación más empleadas en modelos de redes neuronales artificiales (RNA), fuente: Bosch *et al*. (2020)

2.2.6 Arquitectura de las redes neuronales artificiales (RNA)

La arquitectura de una red neuronal artificial se refiere a la disposición interna de sus neuronas o a su vez el número de capas constitutivas, y está estrechamente relacionada con el algoritmo de aprendizaje seleccionado para su entrenamiento. Según la cantidad de capas presentes, las redes pueden clasificarse en monocapa o multicapa. Además, considerando el flujo de información como criterio de clasificación, es posible distinguir entre redes Feedforward y Recurrentes (Caicedo y López, 2020).

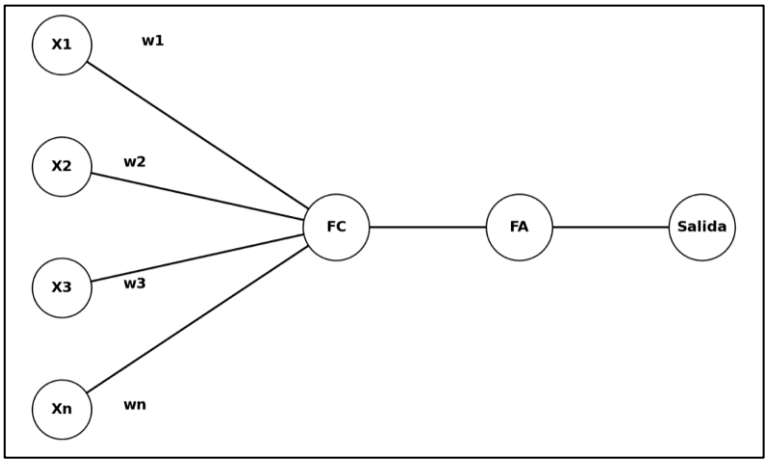
Redes neuronales artificiales monocapa

En este tipo de red neuronal, se distinguen dos niveles de neuronas. El primer nivel está relacionado con la entrada de datos; sin embargo, al no realizar procesamiento alguno, no se considera expresamente una capa, pues su función consiste únicamente en servir de interfaz entre los datos de entrada y las demás neuronas, además, posee la misma dimensión que el vector de entrada, permitiendo que la información sea transferida previo a una alteración en los pesos sinápticos. Dado que todas las neuronas en este nivel reciben la misma información, se le considera una capa, que corresponde a la salida de la red y se denomina “capa de salida”. Cabe destacar que, debido a que la arquitectura posee únicamente esta capa de procesamiento, se clasifica como una red monocapa (Caicedo y López, 2020).

La Figura 5 se puede observar una conexión entre las neuronas de entrada con una sola salida de la red, sin embargo, el número de salidas también se puede configurar de forma proporcional a los vectores de entrada, por otro lado, el subíndice n hace referencia que el número de entradas posibles no está definido y es diferente para cada caso de estudio, sucede lo mismo para las bias o el peso sináptico que es igual al número de entrada.

Figura 5

Red neuronal artificial RNA monocapa



Nota: la figura representa de forma esquemática una red neuronal monocapa, donde X1, X2, X3 y Xn representa las entradas, W1, W2, W3 y Wn son los pesos sinápticos, FC refiere la función de combinación o entrada, FA refiere la función de activación.

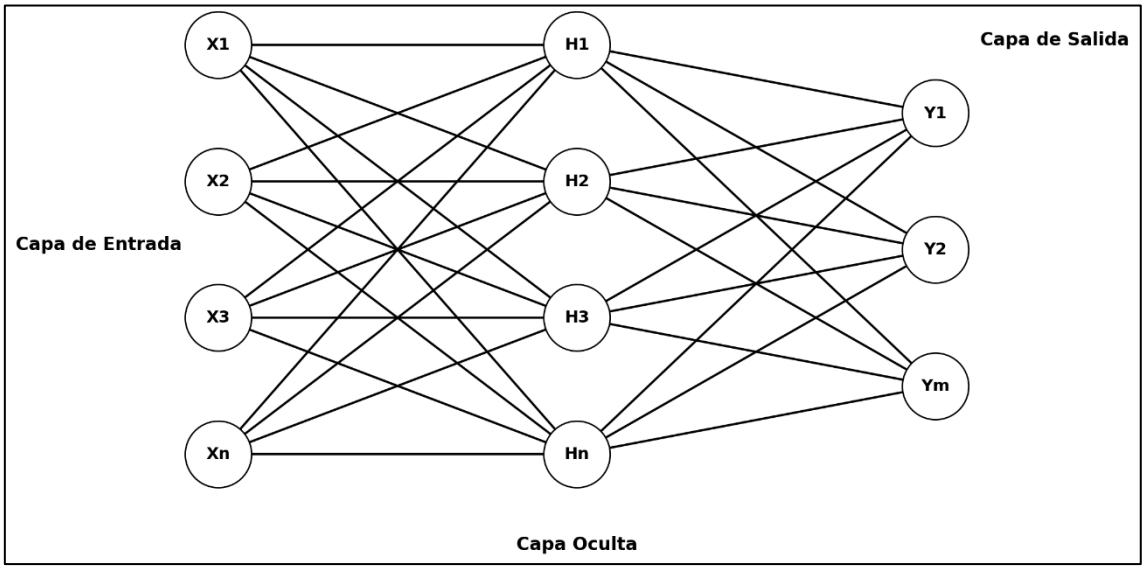
Red neuronal artificial multicapa

En este caso, la red neuronal se compone de una capa de entrada con “n” neuronas y una capa de salida con “m” neuronas, cuya estructura se asemeja al de un perceptrón monocapa (RNA monocapa). La principal diferencia radica en la incorporación de una capa intermedia entre la entrada y la salida, conocida como capa oculta, que está compuesta por “h” neuronas.

Similar a una red monocapa, el flujo de información ocurre en una única dirección, desde la capa de entrada hacia la capa oculta y, finalmente, hacia la capa de salida, partiendo de la premisa de la existencia de conectividad entre neuronas. Es importante señalar que, por cuestiones de simplicidad, esta red se describe con una sola capa oculta; no obstante, un perceptrón multicapa puede integrar múltiples capas ocultas según las necesidades del problema en cuestión (Caicedo y López, 2020).

Figura 6

Red neuronal artificial RNA multicapa



Nota: la figura muestra a manera de esquema una RNA multicapa, compuesta por entradas X1, X2, X3 y Xn, con una capa oculta como H1, H2, H3 y Hm neuronas, y salidas Y1, Y2, Ym.

En la Figura 6 se muestra el esquema de una red neuronal artificial multicapa, con una sola capa oculta, sin embargo, este tipo de configuración puede tener un sin número

de capas que se le asigna en función al problema que se pretende resolver con el modelo.

Redes neuronales artificiales Feedforward

En esta clase de red neuronal artificial, el flujo de información se desplaza en una única dirección, comenzando desde las neuronas de entrada, pasando por la capa de procesamiento en redes monocapa, o por múltiples capas de procesamiento en redes multicapa, hasta alcanzar la salida de la red neuronal.

Las redes feedforward se utilizan en múltiples áreas de aplicación, incluyendo la clasificación de imágenes, el reconocimiento de patrones, la detección de anomalías, y la predicción de series temporales. Debido a su simplicidad y eficiencia en problemas de clasificación y regresión, son una elección común para tareas supervisadas, especialmente cuando los datos son linealmente separables o cuando se busca una estructura sencilla sin interacciones complejas entre los datos (Caicedo y López, 2020).

Como esquema de esta red neuronal artificial se puede ver la Figura 6, si bien es cierto corresponde a una red multicapa, sin embargo, si se considera el flujo de información unidireccional se puede convertir en una red feedforward, además este tipo de red puede configurarse como perceptrones monocapa o multicapa en base al problema planteado.

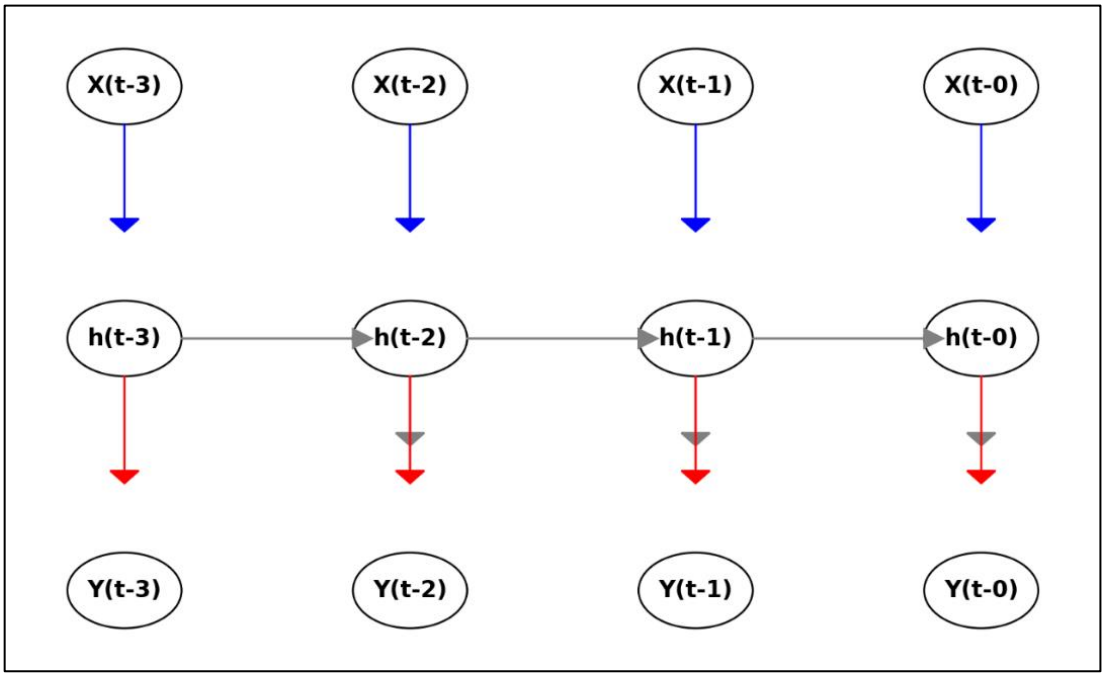
Redes neuronales artificiales recurrentes

En esta clase de red neuronal, el flujo de información no sigue una única dirección, ya que existe la posibilidad de retroalimentarse hacia capas anteriores mediante conexiones sinápticas. Este tipo de arquitectura puede presentar tanto una estructura de capa única como una estructura de múltiples capas (Caicedo y López, 2020).

En la Figura 7, se puede apreciar la arquitectura de una red neuronal artificial de tipo recurrente, la información fluye desde la capa de entrada representado por las letras “X”, hacia una capa oculta “h” que se encarga de recibir información y establecer una memoria a partir de las conexiones recurrentes, dando como resultado una salida “Y”, dado que el flujo de valores en este tipo red no es unidireccional.

Figura 7

Red neuronal artificial RNA recurrente



Nota: la figura ilustra una red neuronal recurrente, con una capa oculta “h” y las capas de entrada “X” y salidas “Y”, con las respectivas retroalimentaciones en cada una de las neuronas de la capa oculta donde se genera la memoria.

Redes neuronales artificiales recurrentes GRU

Las Redes Neuronales Artificiales tipo GRU (Gated Recurrent Units) son una arquitectura recurrente diseñada para modelar datos secuenciales de manera eficiente. Introducidas por Cho *et al*. 2014, las GRU simplifican la arquitectura de las redes LSTM combinando múltiples puertas en dos principales: la puerta de actualización y la puerta de reinicio. Esta estructura permite que las GRU manejen dependencias a largo plazo con menor complejidad computacional.

Las redes neuronales de tipo GRU tienen la capacidad de generalizar de manera más precisa y rápida en comparación con las LSTM, además los recursos computacionales que requiere para su entrenamiento son considerablemente menores lo cual resulta conveniente en cuanto a lo económico, por estas ventajas son útiles en aplicación de traducción automática y predicción de series temporales, las GRU representan una herramienta versátil para abordar problemas complejos en el ámbito de la inteligencia artificial.

El entrenamiento de las redes neuronales tipo GRU se basa, al igual que en las LSTM, en el algoritmo de retropropagación a través del tiempo (BPTT), lo que permite ajustar los pesos considerando la información secuencial que contienen los datos de entrada. Las GRU simplifican la arquitectura de las LSTM al combinar las puertas de entrada y olvido en una única puerta de actualización, lo cual reduce la complejidad computacional. Esta simplificación las hace especialmente atractivas en contextos donde se requiere eficiencia y rapidez en el entrenamiento durante el proceso de aprendizaje, también se emplean técnicas como regularización, descenso del gradientes y ajuste de hiperparámetros para evitar problemas de convergencia o sobreajuste (Caicedo y López, 2020).

Redes neuronales artificiales de tipo LSTM

Las Redes Neuronales de tipo LSTM (Long Short-Term Memory) son una arquitectura especializada dentro de las redes neuronales recurrentes, diseñada para capturar patrones en secuencias de datos. Introducidas por Hochreiter y Schmidhuber en 1997, las LSTM abordan el problema del desvanecimiento y explosión del gradiente mediante un sistema de celdas de memoria y puertas. Estos permiten conservar información relevante durante períodos prolongados, lo que las hace ideales para tareas que requieren modelar dependencias temporales a largo plazo.

Las LSTM funcionan a través de tres puertas principales: entrada, olvido y salida. La puerta de entrada decide qué nueva información se almacena en la celda de memoria, mientras que la puerta de olvido regula qué datos anteriores se descartan. Finalmente,

la puerta de salida determina qué partes de la memoria se utilizan para generar la salida actual. Este mecanismo jerárquico de control permite que las LSTM manejen datos secuenciales complejos con mayor precisión.

El entrenamiento de redes LSTM se basa en la retropropagación a través del tiempo (BPTT), permitiendo ajustar los pesos considerando dependencias temporales en los datos. Estas redes requieren una cuidadosa configuración de hiperparámetros como la tasa de aprendizaje y el tamaño del lote. Para evitar problemas en el descenso del gradiente, se realiza un tratamiento de la base de datos. Su capacidad de retener información durante secuencias largas las hace ideales para tareas complejas. No obstante, este proceso exige recursos computacionales y validación constante. El objetivo final es lograr un modelo que generalice adecuadamente y evite el sobreajuste (Caicedo y López, 2020).

2.2.7 Entrenamiento de la red neuronal artificial RNA

El entrenamiento de una red neuronal artificial es el proceso mediante el cual el modelo ajusta sus parámetros internos, específicamente los umbrales también conocidos como pesos o sesgos, a fin de instruirse sobre patrones concretos de un conjunto de datos y efectuar predicciones exactas sobre un conjunto de prueba. Durante la instrucción, se asignan datos de entrada marcados (cuando se trata de aprendizaje supervisado) y, a partir de ellos, va reformando sus parámetros internos mediante un algoritmo de optimización para reducir el error en sus predicciones.

El entrenamiento de una red involucra un número indefinido de iteraciones sobre el conjunto de datos de entrenamiento, admitiendo que los pesos se ajusten y que el modelo asimile esquemas más complejos con cada paso. Sin embargo, es necesario implementar estrategias a fin de evitar sobreajustes, es decir, que la red se ajuste desmedidamente a los datos de entrenamiento, reduciendo así su capacidad de generalización en datos de prueba.

El proceso de entrenamiento finaliza cuando el modelo alcanza ciertos criterios, que pueden incluir un umbral de error minúsculo o un número determinado de épocas. Una red neuronal debidamente entrenada tiene capacidad de generalización y eficiencia en la predicción en un conjunto denominado de prueba, mediante el cual se demuestra su utilidad práctica en la resolución de un problema (Incio *et al*., 2022).

Método del descenso del gradiente

El error en una red neuronal se define como la diferencia entre el resultado esperado en una instancia de entrenamiento y el resultado producido por la red. Para cuantificar este error, se emplean diversos métodos, entre los cuales se destaca el error cuadrático medio, que mide la magnitud de estas discrepancias promedio entre las predicciones y los valores reales (Bosch *et al*., 2020).

La ecuación que define el Error Medio Cuadrático (MSE) puede representarse de la siguiente manera: en esta fórmula, la letra c simboliza el valor correspondiente a una instancia específica de entrenamiento, mientras que Y representa el resultado obtenido en dicha instancia. Esta relación permite evaluar la discrepancia entre los valores reales y las predicciones del modelo.

$$\in = \frac{1}{2}\,(c - Y)^2 \qquad (11)$$

La instancia de entrenamiento de la red neuronal artificial se refiere a cada uno de los entrenamientos llevados a cabo, por cada entrenamiento se obtendrá unos valores de salida que se debe analizar si corresponden o está cerca de los datos esperados, y de ser el caso ajustar los pesos con la finalidad que la red vaya perfeccionando la función de salida (Bosch *et al*., 2020).

En casos de redes neuronales con funciones discontinuas no es posible aplicar la ecuación 11, es decir, el método del descenso de gradiente sobre los resultados obtenidos en cada instancia resulta inviable, sin embargo, existe la posibilidad de

aplicar a los valores de activación, esta técnica lleva el nombre de ADALINE por sus siglas en inglés, que en español significa “elementos adaptativos lineales”.

La ecuación 11 sufre una leve modificación para su aplicación en redes neuronales con funciones de entrada discontinuas, donde la letra “Y” se reemplaza por la “Z” que corresponde al valor de activación.

$$\in = \frac{1}{2}\,(c - Z)^2 \tag{12}$$

Método de retropropagación

En redes neuronales artificiales con múltiples capas, es inviable la aplicación de un algoritmo que contiene el descenso del gradiente, el inconveniente surge en los puntos de conexión o nodos de las capas ocultas, puesto que no es posible conocer con exactitud los valores de salida correctos (Bosch *et al*., 2020).

En base a lo anterior se plantea una solución para redes neuronales artificiales multicapa con función de activación sigmoide, consiste en una ecuación con regla de variación delta representada por la siguiente expresión matemática.

$$\Delta W_i^j = n\sigma'(Z^j)(C^j - Y^j)X_i^j \tag{13}$$

$\sigma'(\mathrm{Z^j})(\mathrm{C^j})$: indica la pendiente de la función de activación sigmoide
$(\mathrm{C^j} - \mathrm{Y^j})$: representa el error derivado de la neurona “j”
$\mathrm{X_i^j}$: representa el grado de importancia de la entrada “i” de la neurona “j”, en caso de que el valor sea cero no es necesario ningún ajuste del umbral, sólo cuando el valor es superior a cero es propicio su ajuste según sea necesario (Bosch *et al*., 2020).

El método de retropropagación o backpropagation por su denominación en inglés consiste en dos sencillos pasos que se enumeran a continuación:

- **Propagación hacia adelante. -** feedforward por su denominación en inglés, consiste en mover el conjunto de entrenamiento en una sola trayectoria, desde la capa de entrada hacia las capas ocultas y, finalmente, hasta la capa de salida, sin retroalimentación. Este flujo unidireccional es el primer paso en el entrenamiento de la red, donde se realiza un primer análisis de la predicción inicial que luego se coteja con el valor deseado con la finalidad de ajustar los parámetros internos de la red en instancias posteriores. En cada capa, la combinación de pesos, sesgos y funciones de activación permite a la red modelar relaciones no lineales y representar patrones complejos en el conjunto de entrenamiento (Carpio y Oñate, 2020).

- **Propagación hacia atrás. -** La propagación hacia atrás en redes neuronales artificiales es un algoritmo esencial para el ajuste de los parámetros internos de la red, específicamente de las bias o también conocidos como sesgos, y permite que el modelo aprenda de sus errores. Este proceso se lleva a cabo luego de la propagación hacia adelante feedforward por su denominación en inglés, cuando se ha generado una predicción inicial que se coteja con el valor deseado mediante una función de pérdida. La propagación hacia atrás computa los gradientes de la función de pérdida en relación con cada umbral de la red, determinando así la contribución de cada parámetro al error total (Carpio y Oñate, 2020).

El algoritmo de propagación hacia atrás calcula los gradientes de la función de pérdida con respecto a cada parámetro de la red neuronal, permitiendo determinar la contribución específica de cada uno al error total.

2.2.8 Normalización

En redes neuronales artificiales se puede tener un sin número de datos variados y a su vez discordantes entre sus métricas, es decir, se puede tener valores en el orden de los miles, y otros en el orden de las décimas o centésimas, a partir de esta premisa

surge la necesidad de realizar un análisis del conjunto de datos de entrenamiento y de ser necesario realizar la respectiva normalización, con la finalidad de facilitar la generalización durante aprendizaje de la red neuronal.

Normalizar un conjunto de datos tiene como finalidad asegurar una escala análoga en los datos de entrada, lo cual favorecerá cualquier proceso estadístico impidiendo que los valores más altos dominen un conjunto multivariable. Desde una perspectiva metodológica, la normalización favorece la reducción de problemas referentes al aprendizaje, mejorando notablemente la capacidad del modelo de generalizar datos no vistos y por ende la precisión en la predicción (Perdigón y González, 2022).

Durante los años de investigación de las redes neuronales artificiales se han propuesto algunos métodos, a continuación, se realiza una breve descripción de estos.

- **Normalización min – max. -** este método consiste en estandarizar los datos de tal manera que puedan ajustar a un rango predeterminado, generalmente entre 0 y 1

La siguiente ecuación representa la definición exacta de este primer método, donde $\mathrm{X}_{\max}$ y $\mathrm{X}_{\min}$ corresponde a los valores más altos y más pequeños respectivamente de un conjunto de datos.

$$X_{Normalizado} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{14}$$

Estandarización Z- score. - al igual que el caso anterior se considera un método de procesamiento inicial que consiste en ajustar los datos de entrada de tal forma que se tenga una desviación estándar de cero y media de cero (Montaño, 2002).

La ecuación característica de este método es la siguiente, donde la letra “X” representa el valor de entrada que se pretende ingresar a la red, entre tanto “μ” y "$\boldsymbol{\rho}$" corresponden a la media y desviación estándar respectivamente.

$$X_{Estandarizado} = \frac{X - \mu}{\rho} \tag{15}$$

- **Normalización por lotes. –** en este método se busca que un conjunto de datos organizado por lotes tenga valores con medias próximas a cero y una varianza de uno (Montaño, 2002).

La expresión matemática característica de este método es la siguiente, donde μ_{lote} y σ^2_{lote} hacen referencia al promedio y la varianza respectivamente, que, a su vez correspondiente a un lote en específico, entre tanto, $\gamma\, Y\, \beta$ son parámetros característicos aprendidos de la red, finalmente, ∈ corresponde a un error ponderado.

$$X_{normalizada} = \frac{X - \mu_{lote}}{\sqrt{\sigma^2_{lote} + \epsilon}} * \gamma * \beta \tag{16}$$

- **Normalización por capas. -** La normalización de capas o *Layer Normalization* por su denominación en inglés, es una técnica empleada ampliamente en arquitecturas que manejan sucesiones, como las redes neuronales recurrentes (RNN). En contraposición a la normalización por lotes que se aplica a un lote en específico de un conjunto de datos (Montaño, 2002).

La ecuación característica es similar a la empleada en la normalización por lotes cambiando únicamente las denominaciones de sus términos, donde X_i representa el dato de activación de una neurona de la red, σ^2 y μ identifican promedio y varianza respectivamente de la capa.

$$X_{normalizada} = \frac{X_i - \mu}{\sqrt{\sigma^2 + \epsilon}} * \gamma * \beta \quad (17)$$

- **Normalización por grupos. -** La normalización por grupos tiene como propósito dividir las características de una capa en diferentes grupos con el propósito de normalizarlas independientemente (Montaño, 2002).

La ecuación es similar a la empleada a la normalización por lotes, cambiando únicamente la denominación y alcance de sus términos como se muestra a continuación.

$$X_{normalizada} = \frac{X - \mu_{grupo}}{\sqrt{\sigma^2_{grupo} + \epsilon}} * \gamma * \beta \quad (18)$$

2.2.9 Métricas de evaluación

El propósito de las métricas de evaluación es medir el nivel o la eficacia de la solución alcanzada, lo cual permite llevar a cabo una comparativa entre diferentes métodos empleando los mismos conjuntos de datos.

Las métricas disponibles para esta evaluación varían, en gran medida en función a la naturaleza del problema abordado. En el contexto de la predicción de la vida útil restante o RUL (Remaining Useful Life) por su denominación en inglés se considera una regresión multivariada, para lo cual se emplea métricas como las que se describen a continuación:

Mean Absolute Error

El Mean Absolute Error (MAE) o Error Absoluto Medio calcula el promedio de las diferencias absolutas entre las predicciones y los valores reales. Al trabajar con diferencias absolutas, el MAE provee una métrica instintiva que evalúa la dimensión promedio de los errores en las mismas unidades que la variable objetivo. Esto lo hace

especialmente útil para interpretaciones simples y prácticas en problemas de regresión.

Una ventaja clave del MAE es su robustez frente a valores atípicos pequeños, ya que trata los errores de manera lineal. Sin embargo, no penaliza más los errores grandes, lo que puede ser una desventaja en contextos donde los errores significativos son especialmente importantes. Su fórmula se expresa como:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|Y_i - Y'_i| \quad (19)$$

Donde:

n: número total de observaciones

Y_i: Valor real de la observación i:

Y'_i: Valor predicho por el modelo para la observación i

Mean Squaered Error (MSE)

El Error Cuadrático Medio (Mean Squared Error, MSE) calcula el promedio de los errores al cuadrado entre las predicciones del modelo y los valores reales. Una de sus características distintivas es su capacidad para penalizar de manera más severa los errores grandes al elevarlos al cuadrado, lo que lo convierte en una métrica particularmente adecuada en contextos donde los errores significativos tienen un impacto crítico en las decisiones derivadas del modelo. Este enfoque permite priorizar la precisión en las predicciones al minimizar los desvíos considerables.

Sin embargo, su sensibilidad a valores atípicos puede ser un problema para considerar sobre todo en datos ruidosos. Su deducción, aunque menos instintiva que la del MAE, es de gran importancia para modelos que priorizan restar importancia a errores grandes. La fórmula característica de esta métrica es la siguiente:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - Y'_i)^2 \quad (20)$$

Donde:

n: número total de observaciones

Y_i: Valor real de la observación i:

Y'_i: Valor predicho por el modelo para la observación i

Root Mean Squared Error (RMSE)

El Root Mean Squared Error (RMSE) o raíz cuadrada del Error Cuadrático Medio, en términos sencillos corresponde a la raíz cuadrada del MSE, lo que restituye la métrica a las mismas unidades que la variable objetivo. Esto ayuda a su interpretación, fundamentalmente en problemas donde se desea reconocer el impacto práctico del error en el contexto del dominio.

El RMSE al igual que el MSE penaliza errores grandes debido al cuadrado previo en su cálculo, lo que puede resultar útil o perjudicial según el problema en cuestión. En general, es una métrica más instintiva que el MSE, aunque mantiene su sensibilidad a valores atípicos. Su fórmula es:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - Y'_i)^2} \quad (21)$$

Donde:

n: número total de observaciones

Y_i: Valor real de la observación i:

Y'_i: Valor predicho por el modelo para la observación i

R-Squared (R^2)

El R^2 o también conocido como coeficiente de determinación mide qué proporción de la diferenciación total en los datos reales es explicada por el modelo. Es una métrica relativa, por lo que valores cercanos a 1 indican un modelo que explica casi toda la variabilidad, mientras que valores cercanos a 0 reflejan un modelo poco útil.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(Y_i - Y'_i)^2}{\sum_{i=1}^{n}(Y_i - Y'_i)^2} \quad (22)$$

Donde:

n: número total de observaciones

Y_i: Valor real de la observación i:

Y'_i: Valor predicho por el modelo para la observación i

2.3. Marco Legal

El marco legal que se presenta para el trabajo en cuestión suministra el contexto normativo necesario para garantizar que el estudio ponga en práctica las regulaciones y estándares éticos y legales aplicables. Este apartado sirve para validar y legitimar los procedimientos metodológicos.

En el presente marco legal se considera normativas nacionales, que a su vez se fundamenta en normativas internacionales, además, la base de datos a emplearse se va a obtener del portal de la NASA, que tiene acceso libre para aplicaciones en diferentes investigaciones.

2.3.1 Normativas de Ingeniería Aeroespacial

En Ecuador, las actividades afines a ingeniería aeroespacial son controladas por la Dirección General de Aviación Civil (DGAC). Esta entidad se basa en normativas internacionales como la ICAO (Organización de Aviación Civil Internacional), o la FAA (Federal Aviación Administration), adaptando dichas normativas al contexto local, asegurando que las prácticas y tecnologías empleadas en la aviación, incluido el uso de modelos predictivos, se desempeñan bajo los más altos estándares de seguridad y eficiencia.

ICAO (2018) en su Anexo 8 instituye los lineamientos fundamentales para la aeronavegabilidad Ininterrumpida de las aeronaves de uso comercial, lo cual incluye la certificación, diseño y el mantenimiento de aeronaves y sus respectivos componentes dentro de lo cual se encuentra los motores de propulsión. Además, normaliza los procedimientos y obligaciones técnicas a fin de garantizar seguridad y fiabilidad de las aeronaves durante su vida útil.

Por otro lado, este Anexo destaca la importancia del monitoreo y la inspección de manera frecuente, con el objetivo de identificar y responder adecuadamente ante posibles fallos imprevistos o desgastes de los componentes de la aeronave, de tal

forma que los parámetros medibles permanezcan dentro de los intervalos de seguridad.

La Administración Federal de Aviación FAA por sus siglas en inglés de los EE. UU, establece ciertos lineamientos aplicables que tienen por objeto avalar aquellos modelos predictivos basados en redes neuronales RNA, siempre y cuando estos se alineen con los requerimientos de seguridad.

FAA (2020) en la parte 33 establece los requerimientos concretos para la certificación de motores de aeronaves, de esta forma se asegura seguridad y rendimiento óptimo en condiciones de operación severas. Este reglamento contiene pautas precisas sobre el diseño, construcción, pruebas de resistencia y durabilidad de los motores, así como los procedimientos para valorar el comportamiento frente a fallos imprevistos.

De igual forma, abarca una serie de requerimientos sobre los esquemas de seguridad durante el tiempo operativo de la aeronave, y sobre todo el control de emisiones de gases nocivos hacia al ambiente, con el propósito de avalar que este tipo de motores de propulsión cumplan estrictos estándares de fiabilidad.

2.3.2 Normativa sobre procesamiento de datos

El empleo de redes neuronales artificiales con fines de predicción de vida útil restante involucra la utilización de base de datos que contenga parámetros operativos y técnicos, que se deben procesar en base a la legislación ecuatoriana, para lo cual se va a considerar la Ley Orgánica de Protección de Datos Personales (LOPDP), publicada en el año 2021.

A pesar de que los datos a emplearse no son de carácter personal, más bien corresponde a datos técnicos es importante revisar el objeto y finalidad de la LOPDP en cuanto al manejo de datos personales, que dice lo siguiente:

El objeto y finalidad de la presente ley es garantizar el ejercicio del derecho a la protección de datos personales, que incluye el acceso y decisión sobre información y datos de este carácter, así como su correspondiente protección, Para dicho efecto regula, prevé y desarrolla principios, derechos, obligaciones y mecanismos de tutela (LOPDP, 2021. p.6).

Por otra parte, la General Data Protection Regulation (GDPR) correspondiente al año 2021 de la Unión Europea establece lineamientos importantes sobre el manejo de datos de carácter personal, haciendo énfasis en la necesidad imperiosa de contar con el consentimiento necesario para el manejo transparente de datos personales.

De acuerdo con lo antes descrito, se puede aseverar que tanto a nivel local como internacional las normativas vigentes se enfocan únicamente en el manejo o procesamiento de datos personales, mas no en datos técnicos, no obstante, se va a considerar algunos parámetros de la normativas locales e internacionales para el presente trabajo.

2.3.3 Normativas sobre el uso de la inteligencia artificial

En Ecuador no existe normativa alguna sobre el uso de la inteligencia artificial (IA), por lo tanto, se va a considerar las directrices éticas de la comisión europea, la cual establece la necesidad de brindar transparencia y sobre todo la rendición de cuentas sobre sistemas implementados con inteligencia artificial.

European Commission (2019) establece directrices para garantizar que el desarrollo y uso de sistemas de inteligencia artificial (IA) sean seguros, justos y transparentes. Estas directrices se basan en principios como la transparencia, que implica trazabilidad y explicabilidad; la robustez técnica, que asegura un funcionamiento confiable incluso en condiciones imprevistas; y la responsabilidad, que exige asumir consecuencias del uso de la IA. Además, se destaca la importancia de eliminar sesgos, proteger los datos y maximizar los beneficios sociales de la IA.

CAPÍTULO III

METODOLOGÍA

3.1. Descripción del área de estudio

El presente estudio se sitúa en el contexto de la industria aeroespacial, centrándose en el progreso de estrategias de mantenimiento predictivo para los sistemas de propulsión de aeronaves. La investigación se enfoca en diseñar un modelo capaz de predecir la vida útil restante de turborreactores de doble flujo mediante el empleo de redes neuronales artificiales (RNA). El modelo se desarrollará tomando como referencia una extensa base de datos operacionales de turborreactores de doble flujo, que se obtendrán del Centro de Excelencia de Pronósticos Ames de la NASA, ampliamente conocido por su experticia en este ámbito.

Figura 8

Página WEB del Centro de investigación AMES DE la NASA

Nota: en la imagen se puede apreciar la página WEB del centro de investigación AMES de la NASA, de donde se obtuvo la base de datos empleada en la presente investigación.

El sitio WEB ampliamente difundido por la NASA cuenta con un sin número de ramas de investigación, entre ellas los modelos de predicción de diferentes máquinas y mecanismos.

3.2. Enfoque y tipo de investigación

3.2.1 Enfoque

El enfoque de esta investigación es cuantitativo, centrado en el desarrollo y evaluación de un modelo de predicción basado en redes neuronales artificiales (RNA), que tiene por objeto estimar la vida útil restante de turbo reactores de doble flujo. Además, es cuantitativa puesto que se va a analizar los datos basándose en métricas numéricas que garantizan precisión y objetividad, generando conclusiones basadas en evidencia estadística (Fan *et al*., 2025).

3.2.2 Tipo de investigación

La presente investigación es de tipo exploratoria, dado que se basa en la utilización del conjunto de datos C-MAPSS de la NASA, reconocido en el ámbito del mantenimiento predictivo de sistemas aeronáuticos. El objetivo es implementar redes neuronales artificiales (RNA) para la predicción de la vida útil restante (RUL) de turborreactores de doble flujo. Este proceso incluye el diseño inicial del modelo, entrenamiento, y la posterior evaluación de su desempeño.

En función de los resultados obtenidos en las primeras iteraciones, se procederá a ajustar la estructura interna de la RNA, incluyendo modificaciones en la cantidad de capas ocultas, las funciones de activación, o los parámetros de regularización, con el propósito de optimizar su capacidad predictiva y evitar problemas de sobreajuste o subajuste. Este ciclo de mejora, que implica diseñar, entrenar, evaluar y ajustar, se repetirá de forma iterativa hasta alcanzar un modelo que ofrezca un alto nivel de generalización.

3.3. Definición y operacionalización de variables

La Variable dependiente del presente trabajo es la vida útil restante o RUL por sus siglas en inglés del turborreactor de doble flujo, es decir, se busca predecir el tiempo

en ciclos antes de que el mecanismo entre en fallo. Entre tanto, la variable independiente es la estructura del modelo de red neuronal artificial (RNA), la cual se irá ajustando conforme a los resultados obtenidos de la evaluación de los modelos entrenados.

Tabla 1

Operacionalización de variables

Variable	Dimensión	Indicador
Vida útil restante (VD)	Tiempo de operación	Tiempo de operación restante (Ciclos)
Estructura del modelo RNA (VI)	Arquitectura del modelo	• Número de capas ocultas • Número de neuronas por capa
	Parámetros de aprendizaje	• Función de activación • Algoritmos de optimización
	Hiperparámetros	• Learning rate • Tamaño de lote • Número de épocas • Tasa de regularización

Nota: en la tabla se describe a detalle la variable dependiente (VD) y la variable independiente (VI), con sus respectivas dimensiones e indicadores.

La estructura de un modelo de estas características se compone de diversos parámetros, sin embargo, se ha simplificado en tres parámetros principales representados por la variable independiente como se muestra en la Tabla1.

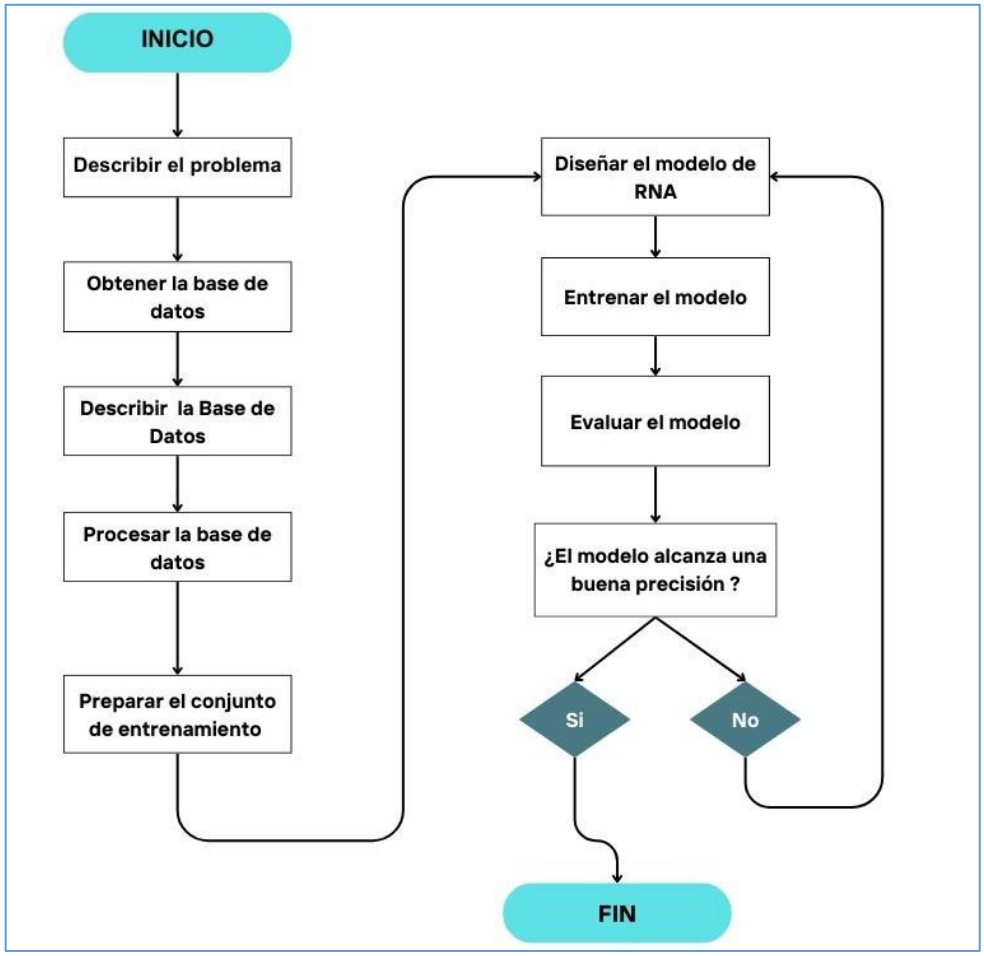
3.4. Procedimientos

El proceso por seguir para el desarrollo de la RNA abarca una serie de acciones que comienza por una breve descripción del problema planteado, esto facilita la comprensión de lo que se busca con la red neuronal artificial, y termina por encontrar un modelo capaz de predecir la vida útil restante de los turborreactores de

doble flujo con una alta precisión. En la Figura 8 se muestra a detalle el proceso.

Figura 9

Diagrama de flujo para el desarrollo de la red neuronal artificial RNA



Nota: la Figura muestra a detalle los pasos a seguir durante el desarrollo el desarrollo de la red neuronal artificial con el propósito de predecir la vida útil restante (RUL, Remaing Useful Life), de una base de datos de datos de la NASA.

3.4.1 Descripción del problema

La presente investigación pretendió desarrollar un modelo basado en redes neuronales artificiales (RNA) con capacidad de predecir la vida útil (RUL, Remaing Useful Life) de propulsores turbofán. Para ello, se utilizó la base de datos C-MAPSS (Commercial Modular Aero-Propulsion System Simulation), desarrollada por la NASA, la cual se compone de datos obtenidos en simulaciones de operación de motores turbofán bajo diferentes condiciones ambientales y modos de fallo, además incluye

variables de suma importancia como presiones, temperaturas, flujos de masa, velocidades de rotor, entre otros, que se miden mediante 21 sensores instalados estratégicamente en los componentes del motor, reflejando así el comportamiento de la máquina a los largo de la vida operativa.

Un modelo de red neuronal artificial implica una serie de procesos claramente definidos, que comenzaron por un correcto procesamiento de la base de datos, seguido del diseño y entrenamiento, y finalizaron por la evaluación de su desempeño, en la mayoría de los casos en los primeros modelos las métricas que se obtuvieron son por demás deficientes, por lo tanto fue necesario ajustar parámetros y aumentar la complejidad de la estructura, este proceso en muchas áreas se conoce como rediseño que se realizó de forma paulina tratando que el modelo pueda generalizar datos no vistos.

3.4.2 Obtención de la base de datos

A continuación, se describe el proceso de obtención de esta base de datos para su uso en la presente investigación:

Acceso a la fuente oficial

- La base de datos C-MAPSS está disponible públicamente y puede ser descargada a través de las plataformas oficiales de la NASA.
- Los archivos constitutivos de la base C-MAPSS se obtuvieron directamente del repositorio de la NASA, de esta manera se garantiza la veracidad de los conjuntos, así como el cumplimiento de las licencias de uso.

Revisión de licencias y términos de uso:

Antes de proceder con la descarga, se revisaron los términos de uso especificados por la NASA para el empleo de sus datos. La presente base de datos cuenta con una licencia de tipo libre, es decir, se permite su empleo para fines de investigación o educativos, considerando su referencia de origen.

Selección de conjuntos de datos específicos:

La base de datos contiene cuatro conjuntos principales (FD001, FD002, FD003 y FD004), cada uno diseñado para representar diferentes escenarios de operación y modalidades de fallo:

- **FD001**: Motores idénticos bajo condiciones operativas uniformes y un solo modo de fallo.
- **FD002**: Variaciones en las condiciones ambientales con un único modo de fallo.
- **FD003**: Motores idénticos con múltiples modalidades de fallo.
- **FD004**: Combinación de variaciones ambientales y modalidades múltiples de fallo.

En la presente investigación, se consideró el empleo de los cuatro conjuntos, que contienen información de funcionamiento de la máquina en diferentes condiciones operativas, lo cual favorecerá el proceso de entrenamiento y posterior validación.

Descarga de la base de datos:

Una vez seleccionados los conjuntos de datos, se procedió a la descarga de los archivos proporcionados por la NASA en formato .txt. Los 8 archivos se encuentran organizados y comprimidos en una carpeta comprimida en formato .ZIP. Además, los registros contienen datos tabulares organizados en columnas que representan variables críticas del funcionamiento del motor, como presión, temperatura, flujo de masa, velocidad del rotor, entre otras.

Validación de la integridad de los datos:

Tras la descarga, los archivos fueron revisados para garantizar que no se encontraran dañados o incompletos. Este proceso incluyó una verificación inicial del contenido y formato, confirmando que los datos eran consistentes con la documentación oficial proporcionada por la NASA.

3.4.3 Descripción general de la base de datos

La base de datos C-MAPSS (Commercial Modular Aero-Propulsion System Simulation) está compuesta por múltiples archivos que representan diversas condiciones operativas, configuraciones de motores turbofán y escenarios de fallos. Los archivos abarcan una serie de datos simulados de ciertas variables críticas que tienden a influir directamente en la operación eficiente del motor de propulsión, siendo una herramienta estratégica para el desarrollo de técnicas predictivas en el mantenimiento basado en la condición del elemento.

La base de datos se compone por un total de doce archivos, divididos en 4 archivos de entrenamiento (train.txt), prueba (test.txt), y vida útil restante (Remaing Useful Life, RUL.txt), respectivamente, cada uno de ellos con una denominación FD001, FD002, FD003 y FD004, para listar los archivos.

Los archivos RUL, contiene datos en un asola columna, que corresponden a la vida útil restante de los motores, en total se contabiliza 707 datos, que corresponde plenamente con los datos del archivo de prueba, por lo tanto, se puede aseverar que en total se analiza 707 máquinas correspondientes a motores diferentes, por otro lado, los archivos de entrenamiento unidos contienen en total alrededor de 160,000 filas y 26 columnas, entre tanto, los archivos de prueba contienen alrededor de 104,000 filas y el mismo número de columnas.

La base de datos C-MAPSS perteneciente al repositorio de la NASA que se emplea para la presente investigación está compuesta por una serie de disposiciones censadas durante el ciclo operativo del turbofán, el conjunto cuenta con archivos de entrenamiento (train.txt) y prueba (test.txt) ordenados en 26 columnas cada archivo.

Tabla 2

Estructura de las columnas de la base de datos del conjunto de entrenamiento y prueba

Columna	Tipo de datos
1	Número de máquina (Machine ID)
2	Tiempo en ciclos (Cycle)
3	Configuración 1 (Setting 1)
4	Configuración 2 (Setting 2)
5	Configuración 3 (Setting 3)
6	(Fan inlet temperature) (◦R) (Sensor 1)
7	(LPC outlet temperature) (◦R) (sensor2)
8	(HPC outlet temperature) (◦R) (sensor 3)
9	(LPT outlet temperature) (◦R) (sensor 4)
10	(Fan inlet Pressure) (psia) (sensor 5)
11	(bypass-duct pressure) (psia) (sensor6)
12	(HPC outlet pressure) (psia) (sensor 7)
13	(Physical fan speed) (rpm) (sensor 8)
14	(Physical core speed) (rpm) (sensor 9)
15	(Engine pressure ratio(P50/P2) (sensor 10)
16	(HPC outlet Static pressure) (psia) (sensor 11)
17	(Ratio of fuel flow to Ps30) (pps/psia) (sensor 12)
18	(Corrected fan speed) (rpm) (sensor 13)
19	(Corrected core speed) (rpm) 8sensor 14)
20	(Bypass Ratio) (sensor 15)
21	(Burner fuel-air ratio) (sensor 16)
22	(Bleed Enthalpy) (sensor 17)
23	(Required fan speed) (sensor 18)
24	(Required fan conversion speed) (sensor 19)
25	(High-pressure turbines Cool air flow) (sensor 20)
26	(Low-pressure turbines Cool air flow) (sensor 21)

Nota: la tabla contiene la denominación de las columnas tanto de los conjuntos de entrenamiento y los conjuntos de prueba.

En la primera de estos archivos se hace referencia al número de máquina o identificación (MachineID), en la segunda columna se enumera los ciclos de operación, considerando que un ciclo corresponde a un despegue y un aterrizaje, desde la tercera hasta la quinta columna concierne a las configuraciones operativas, finalmente, las columnas restantes abarcan a los 21 sensores instalados en lugares estratégicos del turborreactor y miden ciertas variables en operación del dispositivo, como la temperatura del aire a la entrada, y a la salida, presiones, velocidades de giro, entalpias entre otros parámetros. Esta disposición de los datos ofrece gran versatilidad a la hora de realizar un análisis de estos.

La Tabla 2 muestra la estructura de las columnas de la base de datos pertenecientes a los conjuntos train y test, estos archivos se disponen en un total de 26 columnas que detallan 3 configuraciones operativas y mediciones de diferentes variables medidos a través de sensores. Además, los conjuntos pertenecientes a RUL (Remaining Useful Life) se convierta en una sola columna con un único valor de vida útil restante perteneciente a una máquina en el orden que se establecen el conjunto de prueba, que suman en total 707.

Tabla 3

Datos de entrenamiento

Archivos	N° filas	N° Columnas	Total	Total, data train	Porcentaje data train (%)
train_FD001	20631	24	495144	3848616	12,87
train_FD002	53759	24	1290216		33,52
train_FD003	24720	24	593280		15,42
train_FD004	61249	24	1469976		38,19

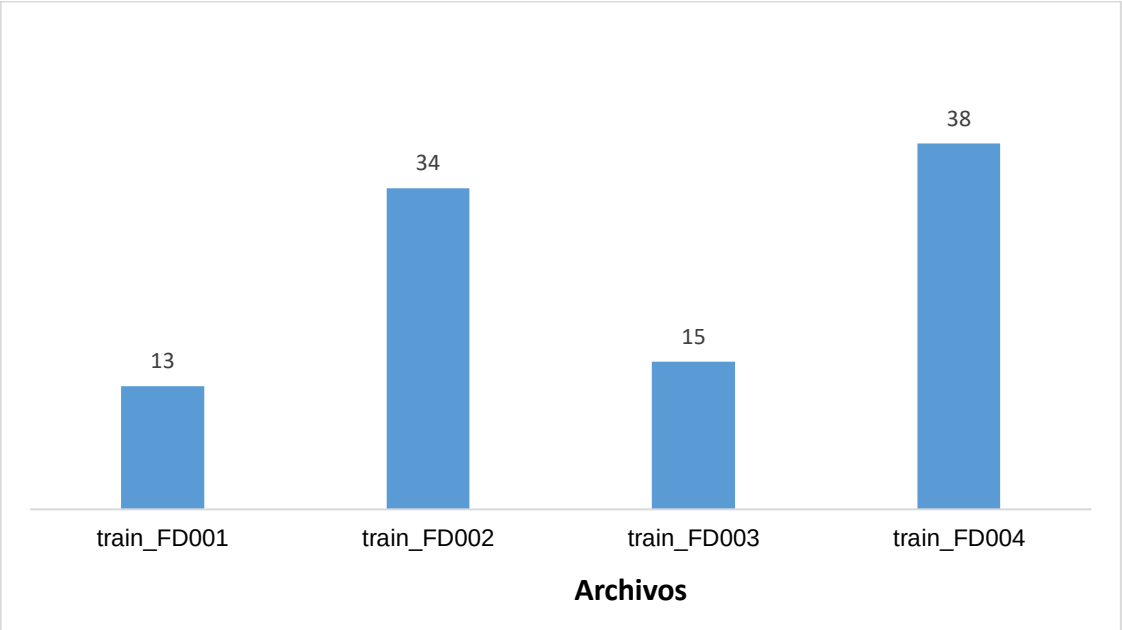
Nota: La tabla describe numéricamente los cuatro archivos de entrenamiento, identificados en su respectivo orden mediante la denominación FD00.

Para el entrenamiento de los modelos de redes neuronales artificiales se va a considerar un total de 24 columnas, las cuales corresponden a las configuraciones operativas de los turborreactores y la medición de ciertas variables obtenidas de los 21 sensores instalados estratégicamente en la estructura de los motores de

propulsión, de esta manera se puede asegurar que en el proceso de entrenamiento cuente con los datos suficientes que le permita generalizar datos no vistos en el proceso de evaluación, en la siguientes Tablas y figuras de muestra a detalle este particular.

Figura 10

Porcentaje data train con respecto a Total train



Nota: la gráfica se observa el porcentaje de datos de entrenamiento (porcentaje data train), con respecto al 100% de los datos (Total train data)

Tabla 4

Datos de prueba

Archivos	N° filas	N° Columnas	Total	Total, data test	Porcentaje data test (%)
test_FD001	13096	24	314304	2517528	12,48
test_FD002	33991	24	815784		32,40
test_FD003	16596	24	398304		15,82
test_FD004	41214	24	989136		39,29

Nota: La tabla describe numéricamente los cuatro archivos de prueba, identificados en su respectivo orden mediante la denominación FD00.

Figura 11

Porcentaje data test por cada archivo con respecto a total test

Nota: la gráfica se observa el porcentaje de datos de entrenamiento (porcentaje data test), con respecto al 100% de los datos (Total test data)

Tabla 5

Datos de RUL (Vida Útil Restante)

Archivos	N° filas	N° Columnas	Total	Total, data RUL	Porcentaje data RUL (%)
RUL_FD001	100	1	100	707	14,14
RUL_FD002	259	1	259		36,63
RUL_FD003	100	1	100		14,14
RUL_FD004	248	1	248		35,08

Nota: La tabla describe numéricamente los cuatro archivos de prueba, identificados en su respectivo orden mediante la denominación FD00.

Los valores de RUL (Remaining Useful Life) están ligados a los datos del conjunto de prueba, por lo tanto, se tienen 4 archivos de FD001 a FD004, que coincide con el número de datos del conjunto de prueba, cabe indicar que, se tiene un solo valor de vida útil restante por cada máquina, por otro lado, cada máquina tiene un numero indefinido de ciclos, contados desde el inicio de operación hasta el momento del fallo, valor definido en los archivos RUL.

Figura 12

Datos de RUL (Vida útil restante)

Nota: la gráfica se observa el porcentaje de datos de RUL (porcentaje data RUL), con respecto al 100% de los datos (Total RUL data)

3.4.4 Procesamiento de la base de datos

En el procesamiento de la base de datos C-MAPSS de la NASA, se procede a realizar ciertas operaciones con la finalidad de garantizar su utilidad en el empleo en redes neuronales artificiales, a continuación, se detallan las operaciones realizadas.

Integración de los conjuntos de datos

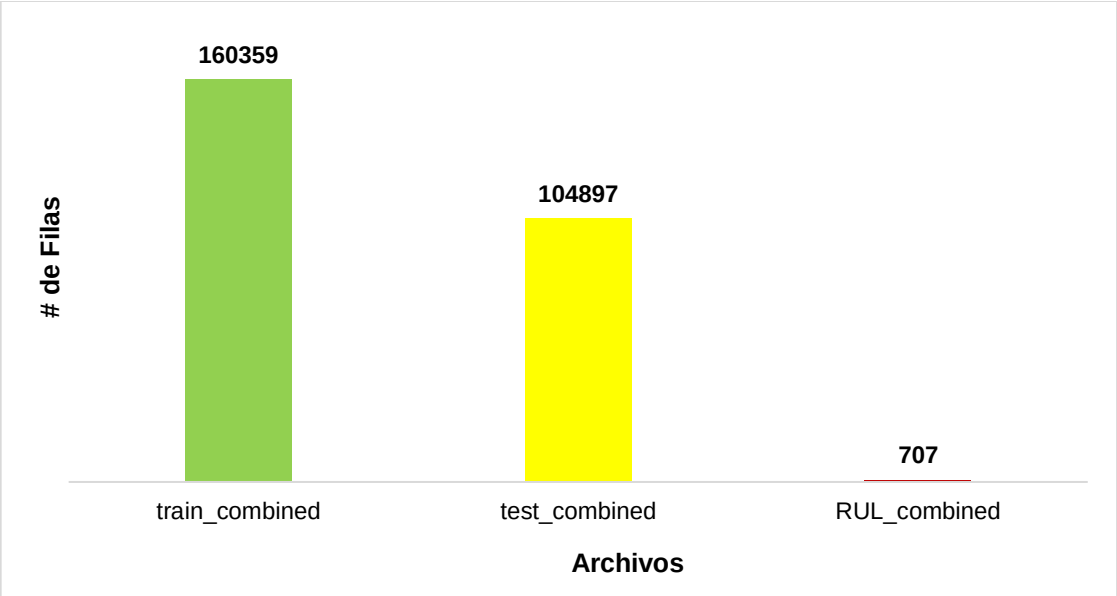
El diseño metodológico de este proyecto se basa en la integración de tres conjuntos de datos principales: entrenamiento, prueba y vida útil restante (RUL). El objetivo de esta etapa es preparar una estructura de datos unificada que permita un análisis consistente y eficiente durante el diseño y la implementación de la red neuronal.

Los archivos de entrenamiento fueron combinados en un único archivo consolidado, conteniendo los datos de los sensores registrados durante los ciclos operativos de los motores de reacción. Los archivos de prueba se unificaron en otro archivo independiente, con un esquema de datos similar al del archivo de entrenamiento, pero sin las etiquetas de RUL. Finalmente, los archivos correspondientes a RUL se combinaron en un archivo único, conteniendo las etiquetas que representan la vida

útil restante, además, se le asignó un número de máquina a cada valor del conjunto de datos combinado.

Figura 13

Cantidad de datos de los archivos combinados



Nota: en la figura se puede observar la cantidad de datos en función al número de filas de los archivos combinados, es de notar que, el archivo RUL contiene una cantidad de 707 datos que es el equivalente al número de máquinas o motores.

Cálculo de RUL para el conjunto de entrenamiento combinado

Este proceso es de suma importancia, puesto que la base de datos contiene un solo RUL, por lo tanto, esos valores se deben emplear para la evaluación del modelo, sin embargo, se necesita el RUL para el conjunto de entrenamiento, para lo cual se va considera la columna correspondiente a Cycle, de tal manera que se va a calcular un RUL instantáneo por cada ciclo para cada máquina en función a la siguiente expresión matemática

$$RUL_i = ciclo\ de\ fallo - ciclo\ actual_i \quad (23)$$

Donde:

$\mathrm{RUL_i}$: Rul correspondiente al $\mathrm{ciclo\ actual_i}$

ciclo de fallo: último ciclo de operación del motor antes que ocurra una falla

ciclo actual_i: ciclo inicial de la máquina cuando entra en operación por primera vez o después de haber entrenado a mantenimiento

Asignación de MachineID al conjunto RUL combinado

Se le asigno un numero de maquina MachineID al conjunto RUL, Con la finalidad de facilitar la evaluación de los modelos RNA, además facilita la comprensión e interpretación de los resultados obtenidos en la evaluación.

Tabla 6

Asignación de MachineID al conjunto RUL

Primeras filas			Últimas filas		
#	RUL	MachineID	#	RUL	MachineID
0	112	1	702	35	703
1	98	2	703	131	704
2	69	3	704	194	705
3	82	4	705	112	706
4	91	5	706	26	707

Nota: en la tabla se muestra las primeras 5 filas y las últimas 5 filas del conjunto RUL que se le asigno un número de máquina, contabilizando en total 707 motores.

Búsqueda de valores faltantes

La identificación de valores faltantes, también conocidos como *missing values*, se llevó a cabo en el entorno de Python mediante la función check_missing_values. Estos valores, que representan la ausencia de datos, se identifican en el entorno como NaN (Not a Number). El análisis se realizó de manera detallada por columna, evaluando cada variable del turborreactor en los archivos train_combined y test_combined. Como resultado de este procedimiento, se verificó que los conjuntos de datos correspondientes a entrenamiento, prueba y RUL no contienen valores faltantes, asegurando la integridad de la información para el modelo.

Búsqueda de valores atípicos

En el presente estudio, se llevó a cabo un proceso de identificación de valores atípicos en la base de datos C-MAPSS (Commercial Modular Aero-Propulsion System Simulation) de la NASA. Esta base de datos cuenta con un elevado número de observaciones y múltiples variables de medición provenientes de sensores.

Los valores atípicos o también conocidos como outliers por su denominación en inglés, se identificaron mediante el método de las distancias de Mahalanobis e Isolation forest, al identificar valores atípicos brinda cierta certeza en la búsqueda de estos valores obteniendo un resultado más preciso y robusto, cabe mencionar que, la identificación de los valores atípicos se realizó en los archivos combinados de entrenamiento y prueba.

La elección de estos métodos se fundamentó en las características específicas de la base de datos C-MAPSS:

- **Alta Dimensionalidad**: el número de variables que reflejan los archivos de prueba y entrenamiento, entre las cuales consta: temperatura de entrada de aire, temperatura de salida de aire, entalpias, presiones entre otras, es necesario recurrir a un método robusto que pueda relacionar dichas variables que se encuentran en diferentes unidades de medida
- **Volumen de Datos**: los datos de entrenamiento y prueba se componen de una gran cantidad de datos, solo en variables se contabiliza 24, sin contar las configuraciones operativas, por este motivo el método de Isolation Forest es ventajoso dado que no requiere un gran procesador para su ejecución.
- **Naturaleza Multivariada:** como ya se menciona en aparatados anteriores, las variables de los conjuntos de prueba y entrenamiento de la base de datos se compone de un total de 24 variables correspondientes a los sensores que toman medidas de forma estratégica en toda la máquina.

Tabla 7

Resumen de valores atípicos en el conjunto de entrenamiento

Método	Total, de outliers	Porcentaje de outliers (%)
Distancias de Mahalanobis	1604	1
Isolation forest	1604	1

Nota: en la tabla se describen los valores atípicos detectados por dos métodos, y se representa el porcentaje equivalente con respecto al total de datos del conjunto de entrenamiento.

Tabla 8

Resumen de valores atípicos en el conjunto de prueba

Método	Total, de outliers	Porcentaje de outliers (%)
Distancias de Mahalanobis	1049	1
Isolation forest	1049	1

Nota: en la tabla se describen los valores atípicos detectados por dos métodos, y se representa el porcentaje equivalente con respecto al total de datos del conjunto de prueba.

Para el método distancias de Mahalanobis se realizó en base a tres pasos principales que se detalla a continuación:

- Calcular la media y matriz de covarianza.
- Calcular la distancia de Mahalanobis para cada punto.
- Establecer un umbral basado en el percentil 99 para identificar outliers.

Para el método Isolation forest se realizó en base a 2 pasos principales que se detalla a continuación:

- Entrenar un modelo Isolation Forest con contamination=0.01.
- Clasificar los puntos como outliers si se aíslan rápidamente (predicción -1).

Incidencia de los valores atípicos en el RUL del conjunto de entrenamiento

De acuerdo con la Tabla 9, se puede describir que el conjunto RUL en las filas con

valores atípicos se mantiene dentro del rango de valores de las filas que no tienen valores atípicos, la media varía claramente, sin embargo, hay que considerar que el número de valores atípicos es significativamente menor

Tabla 9

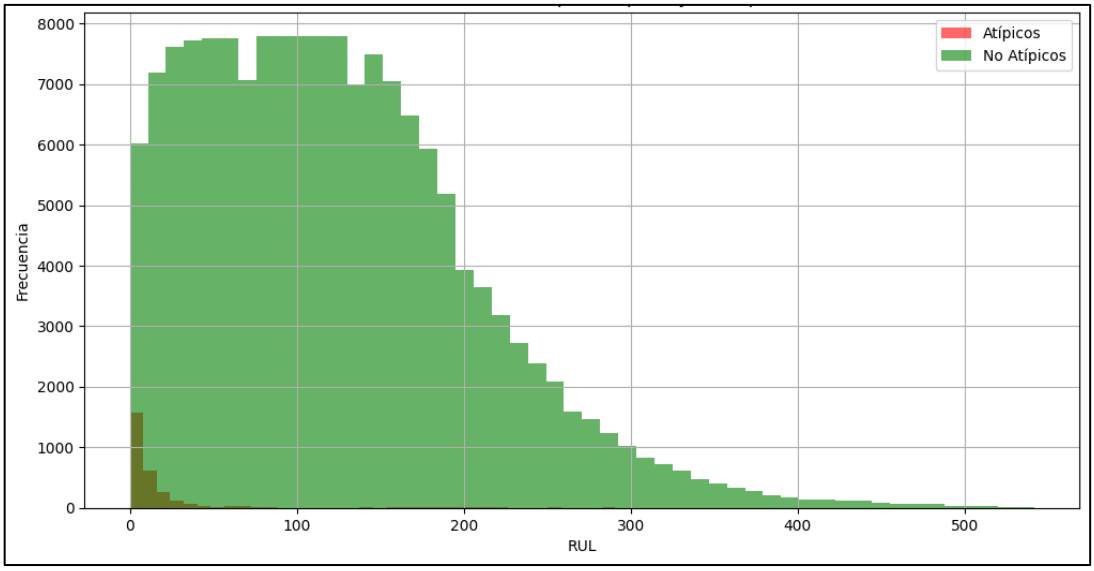
Estadísticas descriptivas de RUL para filas con valores atípicos y no atípicos

	Count	mean	std	min	25%	50%	75%	max
Atípicos	2858	19.5	45.6	< 0.001	3.00	7.00	16.0	404
No atípicos	157501	124	82.9	< 0.001	59.0	114	173	542

Nota: en la tabla se describe el RUL para las filas con valores atípicos y las filas con valores no atípicos observándose la variabilidad de este valor.

Figura 14

Distribución de RUL para valores Atípicos y No atípicos



Nota: en la gráfica se comprueba que el RUL con valores atípicos se mantienen dentro del rango de los valores de RUL sin valores atípicos, además se observa que sigue un patrón similar, considerando que el RUL de los valores atípicos tiene mejor cantidad de datos.

La eliminación o tratamiento de los valores atípicos podría sesgar al entrenar el modelo de RNA, por lo tanto, considerando que los valores atípicos tanto del conjunto de entrenamiento como del conjunto de prueba combinado son equivalentes, es decir, tienen el mismo porcentaje de atípicos respecto de sus totales, se mantendrá para su uso en redes neuronales artificiales.

Normalización de los conjuntos

Una vez que se ha terminado con el análisis de missing values y outliers, se procede a realizar una normalización de los conjuntos, dado que la base de datos C-MAPSS de la NASA está compuesta por múltiples variables con diferentes escalas y unidades de medida (por ejemplo, temperatura, presión, vibraciones y ciclos de operación).

La normalización se realiza con el objetivo de homogeneizar la escala de todas las variables y evitar que aquellas con valores numéricos más grandes dominan el proceso de modelado y análisis.

3.4.5 Preparación del conjunto de entrenamiento

El conjunto de entrenamiento compuesto por 4 archivos train, se debe dividir en dos subconjuntos, considerando un 80% para el entrenamiento y un 20% para la validación del modelo, además, se asigna una función de aleatoriedad, a fin de evitar sesgos, por otro lado, se etiqueta la columna del RUL calculado como la variable objetivo.

- Forma de X_train (128287, 25)
- Forma de Y_train (128287)
- Forma de X_val (32072, 25)
- Forma de Y_val (32072)

3.4.6 Diseño de modelos de RNA

El diseño de los modelos de redes neuronales artificiales tiene por objeto ajustar diversos modelos en conveniencia con la presente investigación, por lo tanto, considerando que se aborda por primera vez la aplicación de estos modelos de inteligencia artificial (IA) en la predicción de la vida útil restante de Turborreactores de doble Flujo, se prevé diseñar varios modelos seleccionando diversas arquitecturas de

RNA, configuraciones y algoritmos de aprendizaje empleados en diversas áreas de la inteligencia artificial.

Seleccionar diversas arquitecturas de red neuronal artificial, valorando configuraciones y algoritmos de aprendizaje supervisado

El diseño de los modelos de Redes Neuronales Artificiales (RNA) inicia con una configuración básica, compuesta por redes neuronales formadas por capas densas interconectadas. Esta configuración inicial sirve como punto de partida para explorar y ajustar la arquitectura del modelo.

Posteriormente, se incrementa gradualmente su complejidad, incorporando elementos adicionales y optimizando parámetros con el objetivo de mejorar su desempeño y adaptarlo a los requerimientos específicos del problema planteado. Este enfoque progresivo permite evaluar de manera sistemática el impacto de cada ajuste en la capacidad predictiva del modelo.

Tabla 10

Diseño de una red neuronal Model_básico

Arquitectura del modelo	Parámetros de aprendizaje	Hiperparámetros
• Red neuronal básica. • Dos capas ocultas de 64 y 32 neuronas respectivamente. • una capa de salida con una sola neurona.	• Función de activación relu • Algoritmo de optimización Adam	• Learning rate ajustado por el optimizador. • Tamaño de lote: 32 • Número de épocas: 50 • Sin tasa de regularización

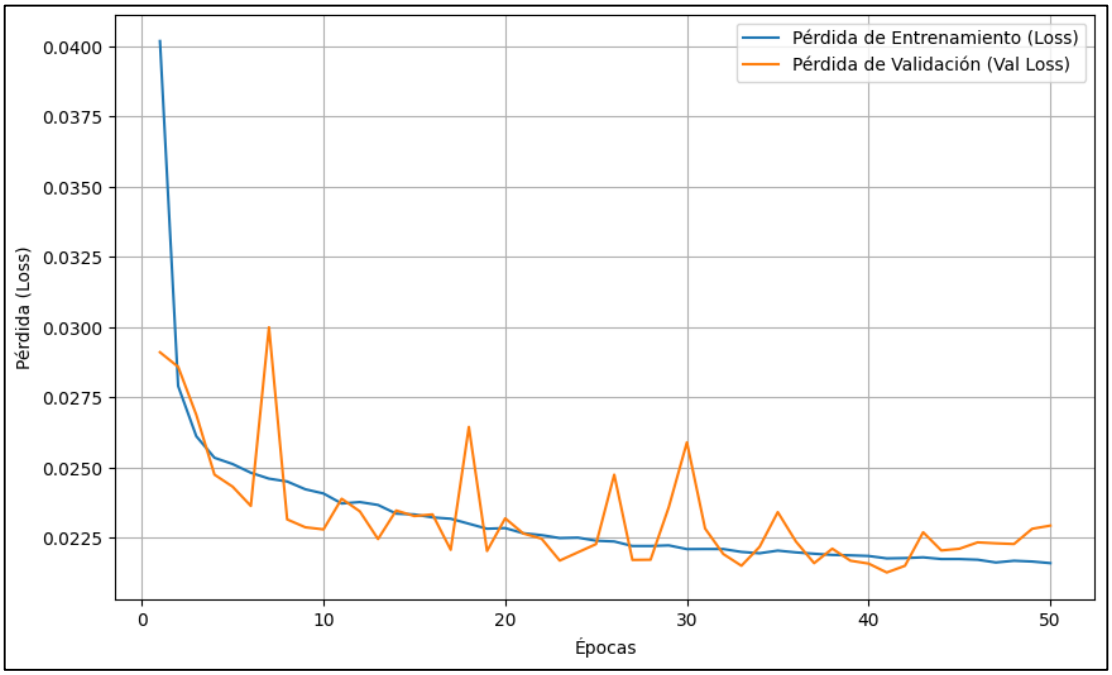
Nota: La tabla describe a detalle la estructura de la red neuronal artificial para el primer modelo denominado Model_básico

Entrenar los modelos a través de técnicas de aprendizaje supervisado, ajustando parámetros e Hiperparámetros hasta alcanzar una adecuada generalización de los datos, de modo que el modelo pueda realizar predicciones precisas y eficaces en un nuevo conjunto de datos.

Una vez realizado el diseño de los modelos, donde se establecen el tipo de arquitectura a emplear, número de capas ocultas y neuronas por capas, además de la configuración de parámetros e Hiperparámetros, se ejecuta el entrenamiento del modelo, para hacer un seguimiento del aprendizaje se configura métricas de entrenamiento tales como: Loss, Val Loos, Mae y Val Mae.

Figura 15

Evolución de la pérdida durante el entrenamiento Model_básico



Nota: la gráfica corresponde a la evolución de la pérdida de entrenamiento (Loss) y pérdida de validación (Val Loss) a lo largo de las 50 épocas establecidas para el entrenamiento.

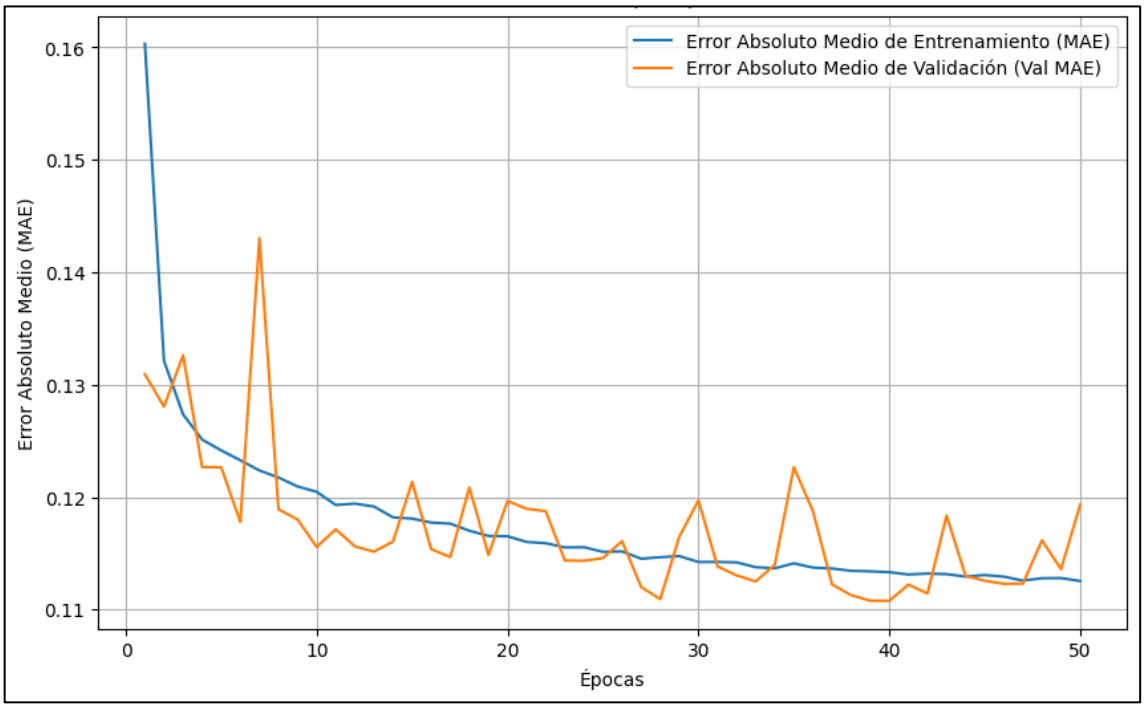
La Figura 15 correspondiente a la evolución de la pérdida durante el entrenamiento del Model_básico evidencia un entrenamiento poco estable a lo largo de las 50 épocas, con fluctuaciones considerables de pérdida de validación durante todo el entrenamiento. Por otro lado, la evolución del error presenta un comportamiento similar observándose fluctuaciones evidentes del Error Absoluto Medio en validación. Además, la pérdida de entrenamiento y Error Absoluto Medio de entrenamiento en ambas gráficas evidencia un avance estable casi sin fluctuaciones notorias.

Considerando los resultados obtenidos anteriormente, se considera implementar una maniobra que permita detener el entrenamiento del modelo en cuestión si este no

presenta una mejora notable durante 10 épocas consecutivas, por lo tanto, desde el Model_básico1 y demás modelos que se entrenen llevaran inmerso en el código la configuracion early stoping. Se implementa esta estrategia con la finalidad de optimizar el tiempo que se invierte entrenando los modelos.

Figura 16

Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_básico



Nota: la gráfica corresponde a la evolución del Error Absoluto Medio (MAE) y Error Absoluto Medio (Val MAE) a lo largo de las 50 épocas establecidas para el entrenamiento.

Tabla 11

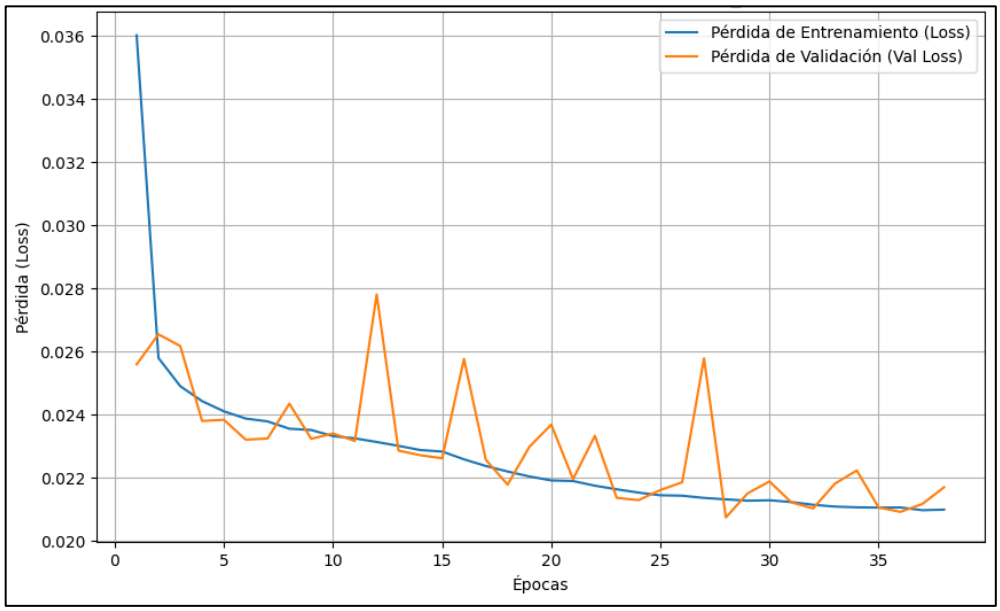
Diseño de una red neuronal Model_básico1

Arquitectura del modelo	Parámetros de aprendizaje	Hiperparámetros
• Red neuronal de tipo densa. • Cuatro capas ocultas de 128, 64, 32 y 16 neuronas respectivamente. • una capa de salida con una neurona.	• Función de activación relu • Algoritmo de optimización Nadam	• Learning rate: 0,001 • Tamaño de lote: 64 • Número de épocas: 100 • Sin tasa de regularización

Nota: La tabla describe a detalle la estructura de la red neuronal artificial para el segundo modelo de regresión denominado modelovt2.

Figura 17

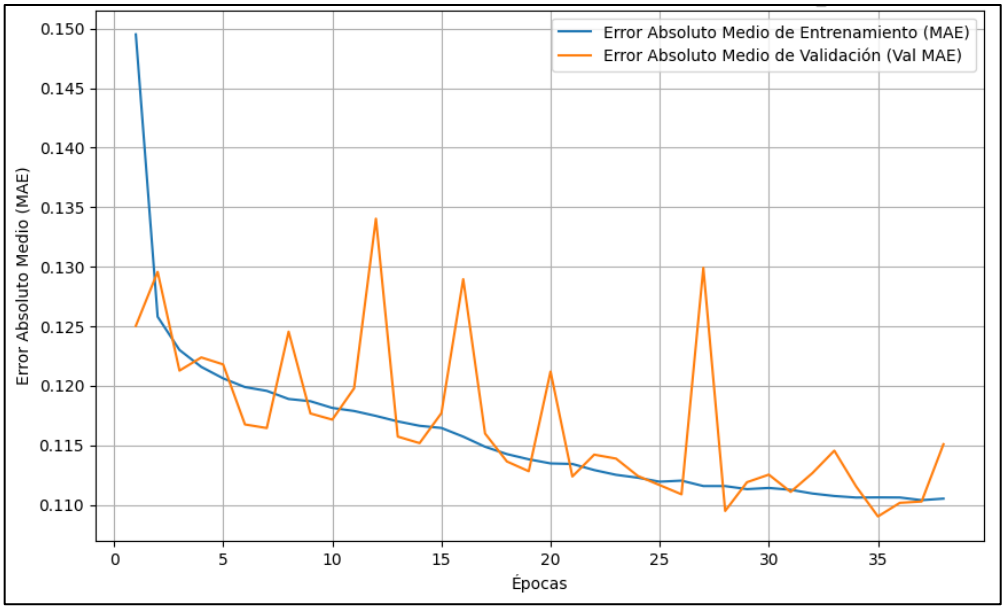
Evolución de la pérdida durante el entrenamiento Model_básico1



Nota: la gráfica corresponde a la evolución de la perdida de entrenamiento (Loss) y pérdida de validación (Val Loss) a lo largo de las 100 épocas establecidas para el entrenamiento, en este caso solo se completan 38 épocas dado que el modelo a partir de las 28 épocas no presenta mejoras notables.

Figura 18

Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_básico1



Nota: la gráfica corresponde a la evolución del Error Absoluto Medio (MAE) y Error Absoluto Medio (Val MAE) a lo largo de las 100 épocas establecidas para el entrenamiento., en este caso solo se completan 38 épocas dado que el modelo a partir de las 28 épocas no presenta mejoras notables.

La Figura correspondiente a la evolución de la pérdida durante el entrenamiento del Model_básico1 evidencia una disminución progresiva en entrenamiento mientras que en validación existe fluctuaciones considerables a lo largo de todas las épocas ejecutadas. Este comportamiento se repite para la Figura 18 la cual describe la Evolución del Error Absoluto Medio durante el entrenamiento del modelo en cuestión

Tabla 12

Diseño de una red neuronal Model_básico2

Arquitectura del modelo	Parámetros de aprendizaje	Hiperparámetros
• Red neuronal de tipo densa. • Siete capas ocultas de 256, 128, 128, 64, 64, 32 y 16 neuronas respectivamente. • una capa de salida con una neurona para regresión.	• Función de activación relu • Algoritmo de optimización Nadam	• Learning rate: 0,00001 • Tamaño de lote: 64 • Número de épocas: 100 • Sin tasa de regularización

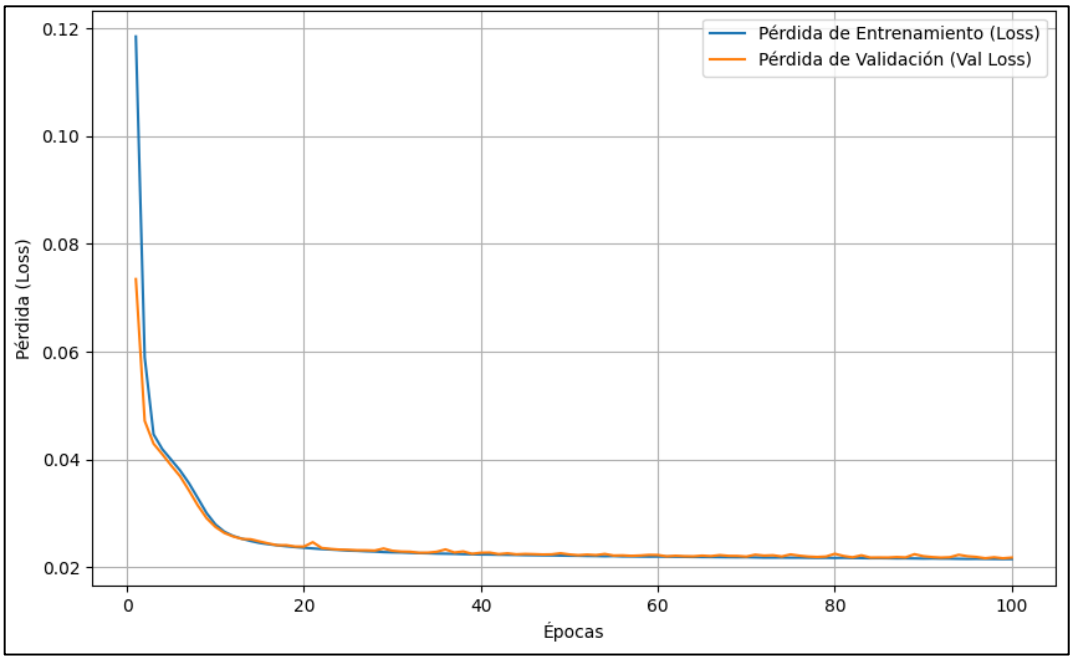
Nota: La tabla describe a detalle la estructura de la red neuronal artificial para el modelo de regresión denominado Model_básico2

La Figura 19 correspondiente a la pérdida durante el entrenamiento del Model_básico2 presenta un comportamiento estable, casi sin fluctuaciones tanto en pérdida de entrenamiento (Loss) como en pérdida de validación (Val Loss), sin embargo, la pérdida disminuye rápidamente en ambos durante las primeras 20 épocas, después de eso el modelo se mantiene en los mismos valores.

La Figura 20 que corresponde al Error Absoluto Medio del mismo modelo presenta un comportamiento casi similar con la diferencia que el MAE de validación oscila un poco más a lo largo de las 100 épocas en comparación con la pérdida de validación. Sin embargo, se evidencia que el modelo a pesar de que disminuye rápidamente en las primeras interacciones el modelo no muestra un progreso significativo.

Figura 19

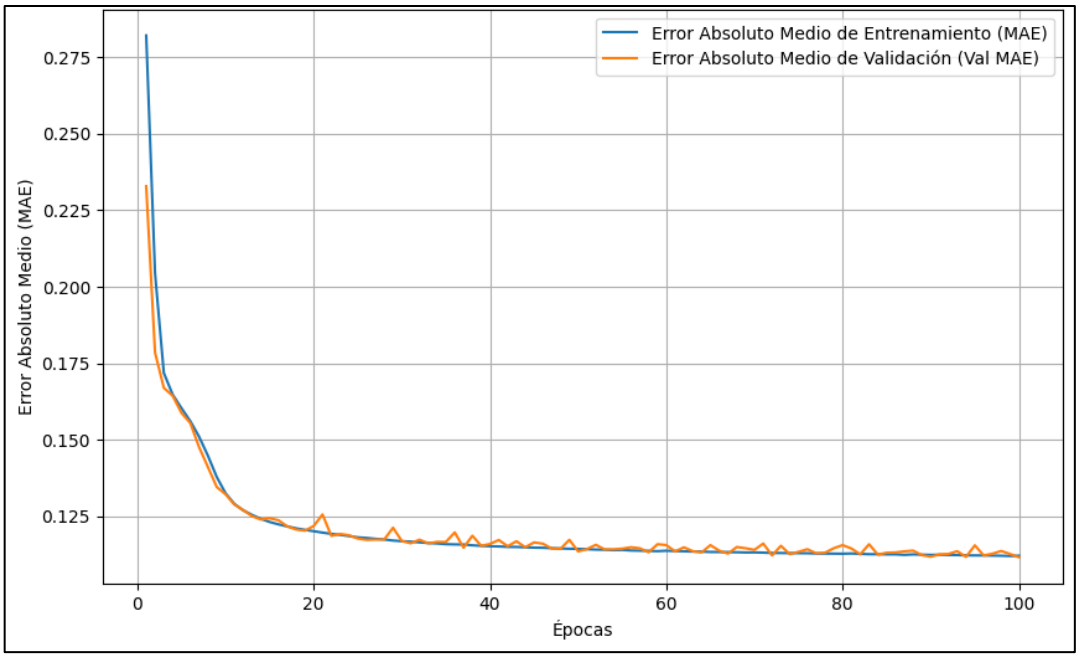
Evolución de la pérdida durante el entrenamiento Model_básico2



Nota: la gráfica corresponde a la evolución de la pérdida de entrenamiento (Loss) y pérdida de validación (Val Loss) a lo largo de las 100 épocas establecidas para el entrenamiento.

Figura 20

Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_básico2



Nota: la gráfica corresponde a la evolución del Error Absoluto Medio (MAE) y Error Absoluto Medio (Val MAE) a lo largo de las 100 épocas establecidas para el entrenamiento.

Tabla 13

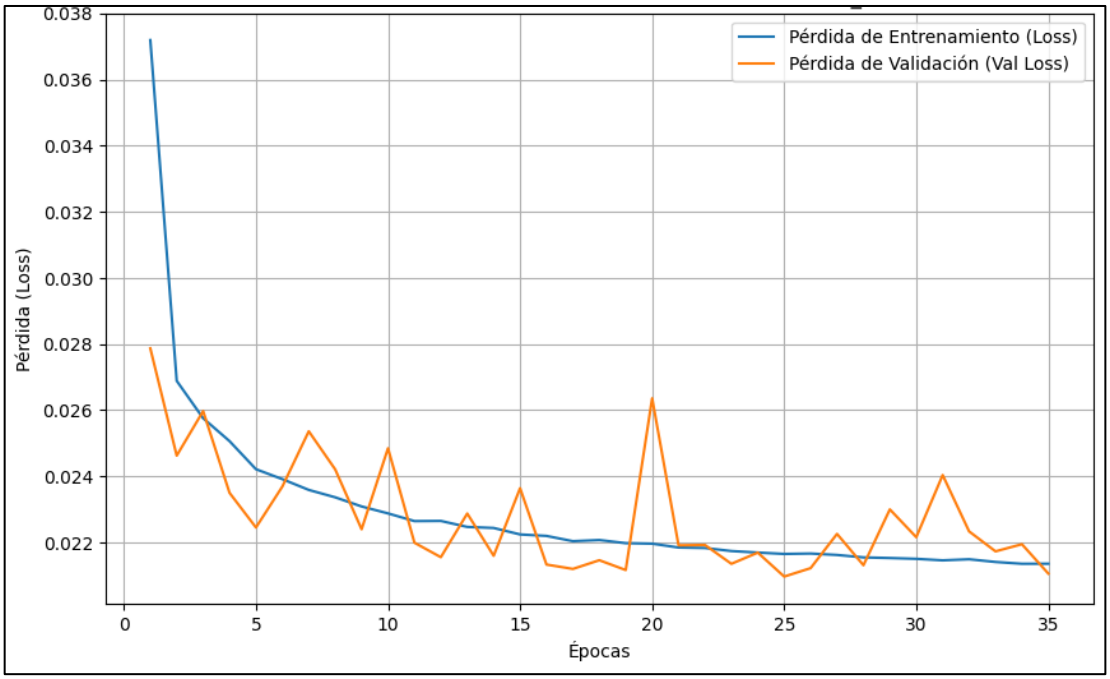
Diseño de una red neuronal Model_básico3

Arquitectura del modelo	Parámetros de aprendizaje	Hiperparámetros
• Red neuronal de tipo densa. • Ocho capas ocultas de 256, 128, 128, 64, 64, 32,32 y 16 neuronas respectivamente. • una capa de salida con una neurona para regresión.	• Función de activación Tanh (Tangente Hiperbólica) • Algoritmo de optimización Nadam	• Learning rate: 0,001 • Tamaño de lote: 64 • Número de épocas: 100 • Sin tasa de regularización

Nota: La tabla describe a detalle la estructura de la red neuronal artificial para el modelo de regresión denominado Model_básico3

Figura 21

Evolución de la pérdida durante el entrenamiento Model_básico3



Nota: la gráfica corresponde a la evolución de la pérdida de entrenamiento (Loss) y pérdida de validación (Val Loss) a lo largo de las 100 épocas establecidas para el entrenamiento, en este caso solo se completan 35 épocas dado que el modelo a partir de las 25 épocas no presenta mejoras notables.

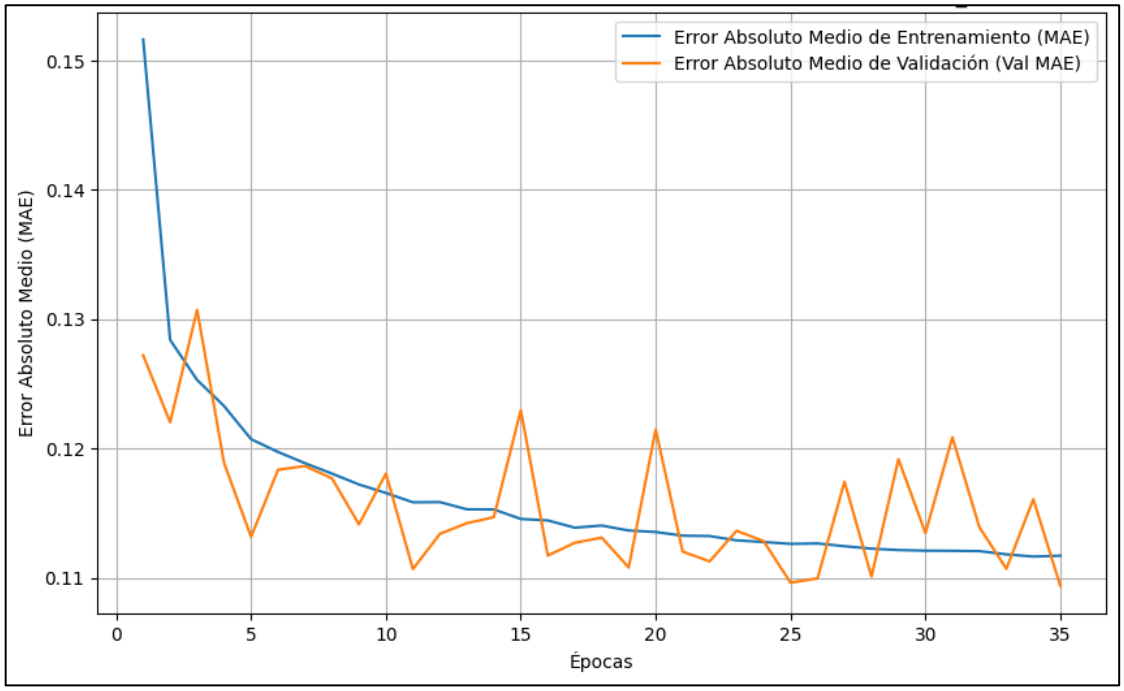
La Figura 21 correspondiente al Model_básico3 muestra un avance estable de la pérdida de entrenamiento (Loss), sin embargo, la pérdida de validación tiende a

presentar fluctuaciones significativas a lo largo de las 35 interacciones, las dos métricas disminuyen rápidamente durante las primeras 5 épocas.

Sin embargo, el avance es considerablemente más lento a partir de las 6 época. Por otro lado, la Figura 22 que representa la evolución del Error Absoluto Medio muestra fluctuaciones aún más evidentes en la validación, y mantiene su comportamiento estable en entrenamiento. Además, el modelo se detiene de manera temprana debido a que no mejora de forma notable desde la época 25.

Figura 22

Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_básico3



Nota: la gráfica corresponde a la evolución del Error Absoluto Medio (MAE) y Error Absoluto Medio (Val MAE) a lo largo de las 100 épocas establecidas para el entrenamiento., en este caso solo se completan 35 épocas dado que el modelo a partir de las 25 épocas no presenta mejoras notables.

Hasta el momento se han entrenado modelos basados en arquitecturas básicas con capas de tipo densas, además, los parámetros e Hiperparámetros de aprendizaje muestran variaciones considerables en las gráficas de pérdidas y el Error Absoluto obtenidas durante el entrenamiento.

Tabla 14

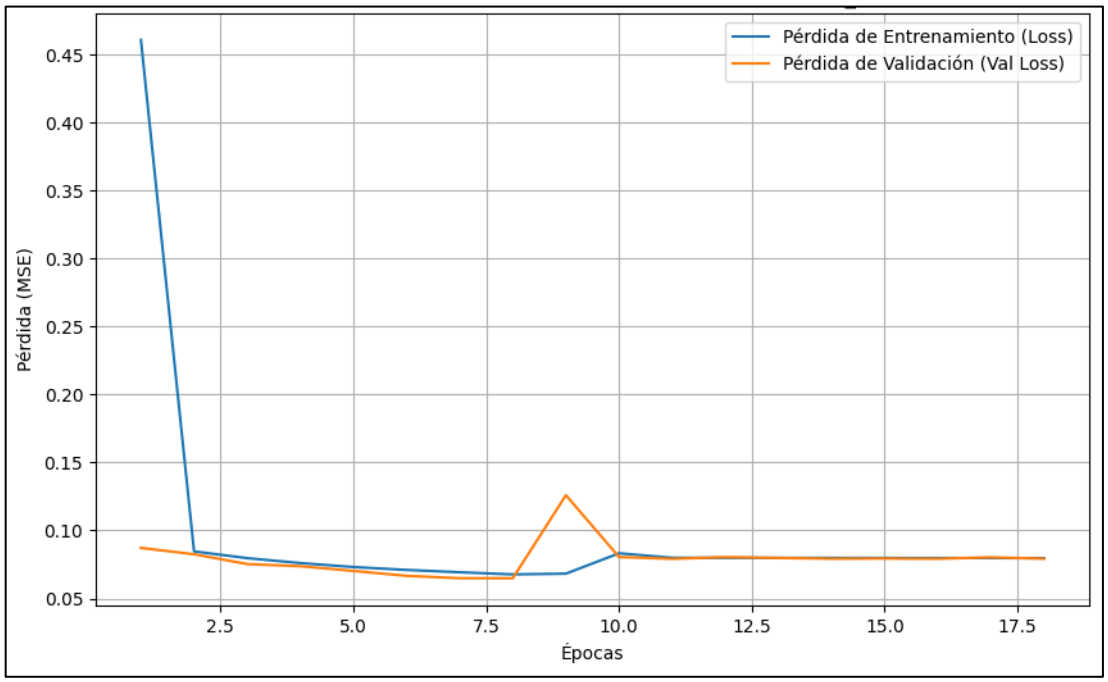
Diseño de una red neuronal Model_básico4

Arquitectura del modelo	Parámetros de aprendizaje	Hiperparámetros
• Red neuronal de tipo densa. • Ocho capas ocultas de 256, 128, 128, 64, 64, 32,32 y 16 neuronas respectivamente. • una capa de salida con una neurona para regresión.	• Función de activación Tanh (Tangente Hiperbólica) • Algoritmo de optimización Nadam	• Learning rate: 0,001 • Tamaño de lote: 64 • Número de épocas: 100 • Tasa de regularización L2 en 0,01 • Dropout: 0,02

Nota: La tabla describe a detalle la estructura de la red neuronal artificial para el modelo de regresión denominado Model_básico4

Figura 23

Evolución de la pérdida durante el entrenamiento Model_básico4



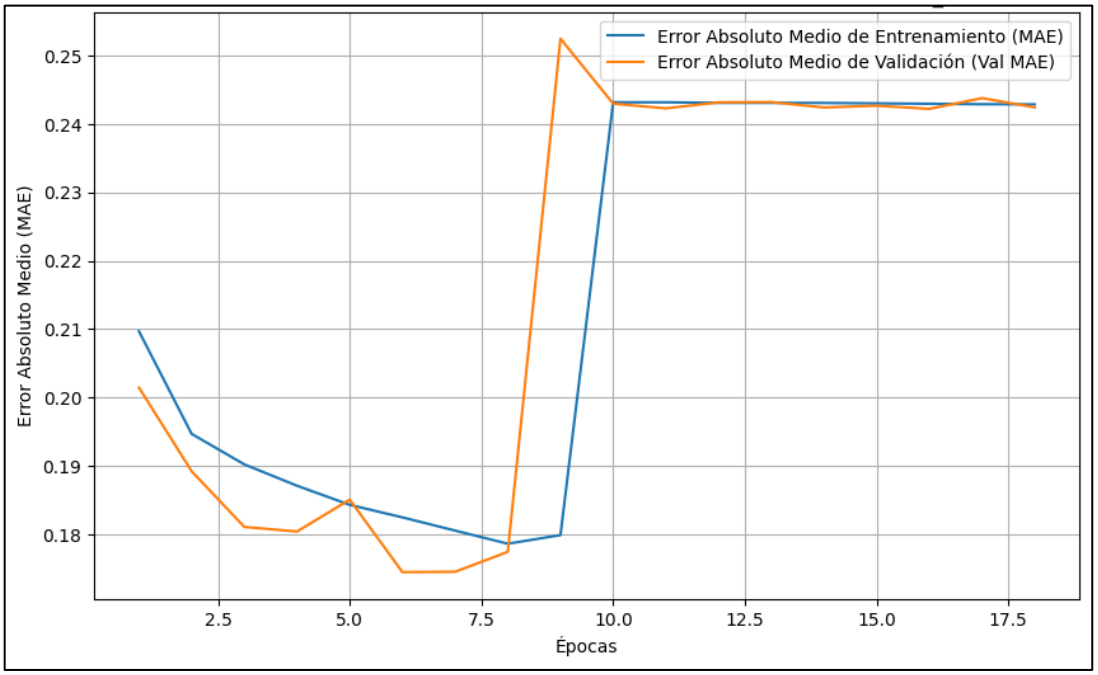
Nota: la gráfica corresponde a la evolución de la pérdida de entrenamiento (Loss) y pérdida de validación (Val Loss) a lo largo de las 100 épocas establecidas para el entrenamiento, en este caso solo se completan 18 épocas dado que el modelo a partir de las 8 épocas no presenta mejoras notables.

La evolución de la pérdida durante el entrenamiento del Model_básico4, ilustrada en la Figura 23, muestra una rápida disminución en el valor de la función de pérdida (Loss) al inicio del proceso, seguida de una estabilización en un valor constante a

medida que avanzan las épocas. No obstante, la pérdida correspondiente al conjunto de validación presenta ligeras fluctuaciones, especialmente alrededor de la mitad de las épocas totales ejecutadas. Este comportamiento sugiere una posible variabilidad en la capacidad del modelo para generalizar datos no vistos durante el proceso de entrenamiento.

Figura 24

Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_básico4



Nota: la gráfica corresponde a la evolución del Error Absoluto Medio (MAE) y Error Absoluto Medio (Val MAE) a lo largo de las 100 épocas establecidas para el entrenamiento., en este caso solo se completan 35 épocas dado que el modelo a partir de las 25 épocas no presenta mejoras notables.

La Figura 24 que representa la evolución del Error Absoluto Medio del Model_básico4, evidencia una rápida reducción durante las primeras épocas, no obstante, luego el modelo muestra un incremento notable en el error tanto en pérdida como en validación, luego se estabiliza en el punto superior alcanzado, luego de aquello no se presenta mejoras significativas, este comportamiento obliga a detener el entrenamiento antes de completar el número de épocas establecidas completándose únicamente 18 de un total de 100.

Tabla 15

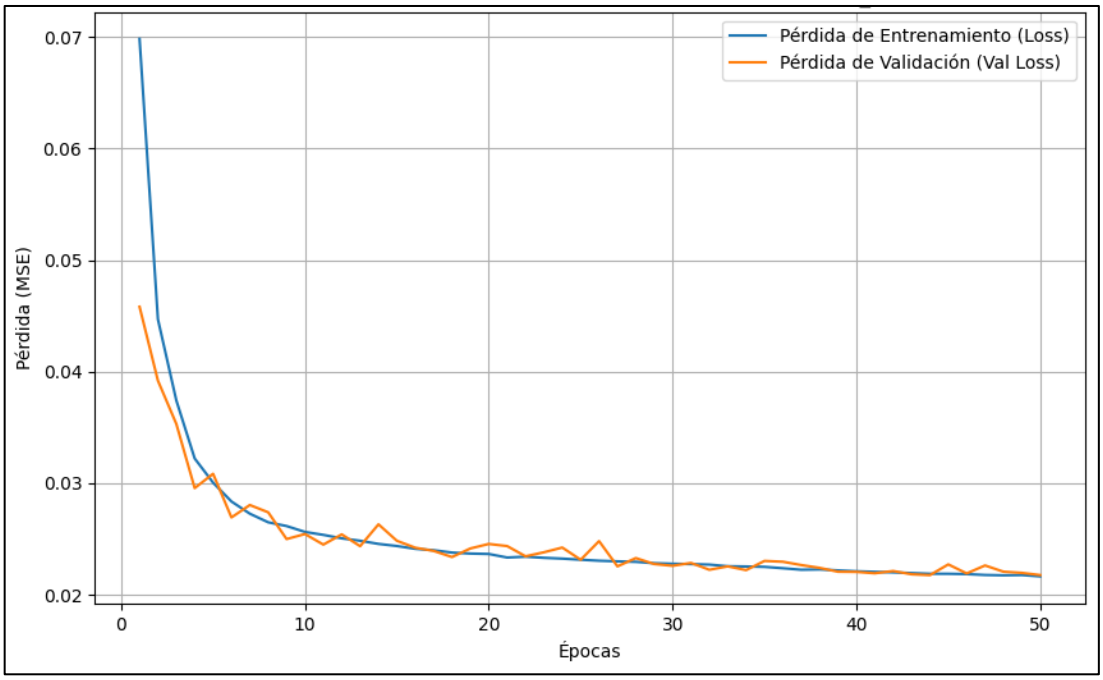
Diseño de una red neuronal de tipo GRU Model_GRU

Arquitectura del modelo	Parámetros de aprendizaje	Hiperparámetros
• Red neuronal de tipo GRU. • Una capa oculta de 64 neuronas • una capa de salida con una neurona para regresión.	• Función de activación Tanh • Algoritmo de optimización Adam	• Learning rate: 0,001 • Tamaño de lote: 32 • Número de épocas: 50 • Sin tasa de regularización

Nota: La tabla describe a detalle la estructura de la red neuronal artificial de tipo GRU para el modelo de regresión denominado Model_GRU

Figura 25

Evolución de la pérdida durante el entrenamiento Model_GRU



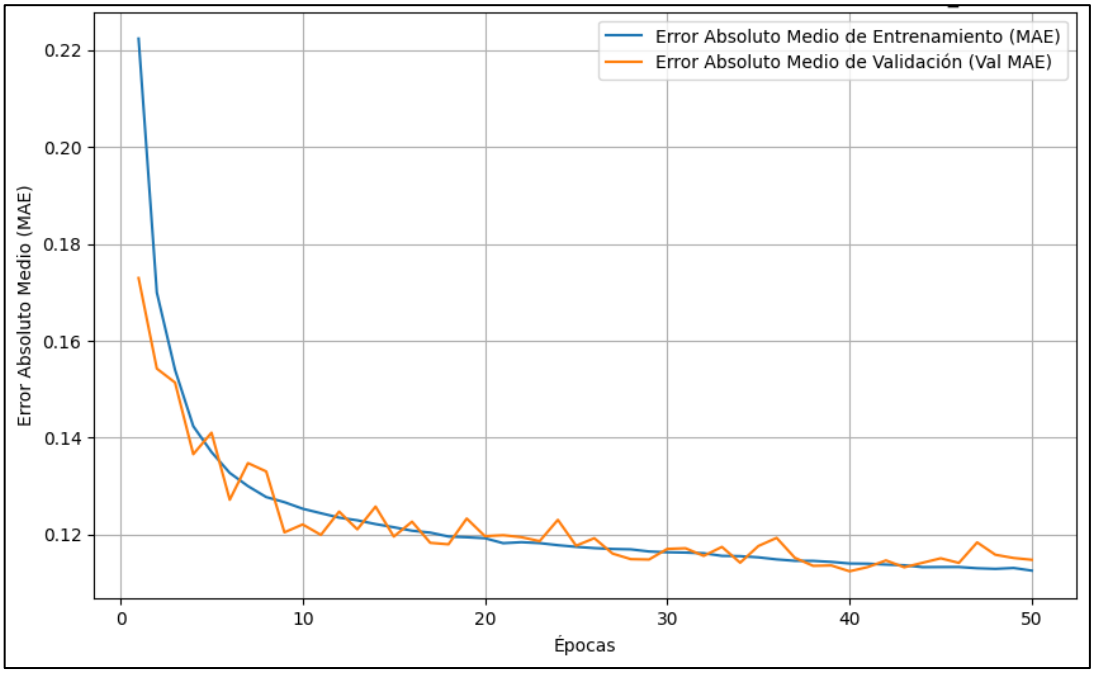
Nota: la gráfica corresponde a la evolución de la pérdida de entrenamiento (Loss) y pérdida de validación (Val Loss) a lo largo de las 100 épocas establecidas para el entrenamiento.

En la Figura 25 se muestra la gráfica correspondiente a la evolución de pérdida durante el entrenamiento, la tendencia evidencia una disminución significativa en las primeras épocas, posterior a ello, la disminución es más lenta y progresiva durante todo el entrenamiento, además, la curva de validación presenta pequeñas

oscilaciones durante las 50 épocas, siendo más pronunciadas en las primeras interacciones. Esta tendencia se evidencia de manera casi similar en la Figura 26 que representa la evolución del Error Absoluto Medio del Model_GRU.

Figura 26

Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_GRU



Nota: la gráfica corresponde a la evolución del Error Absoluto Medio (MAE) y Error Absoluto Medio (Val MAE) a lo largo de las 100 épocas establecidas para el entrenamiento.

Tabla 16

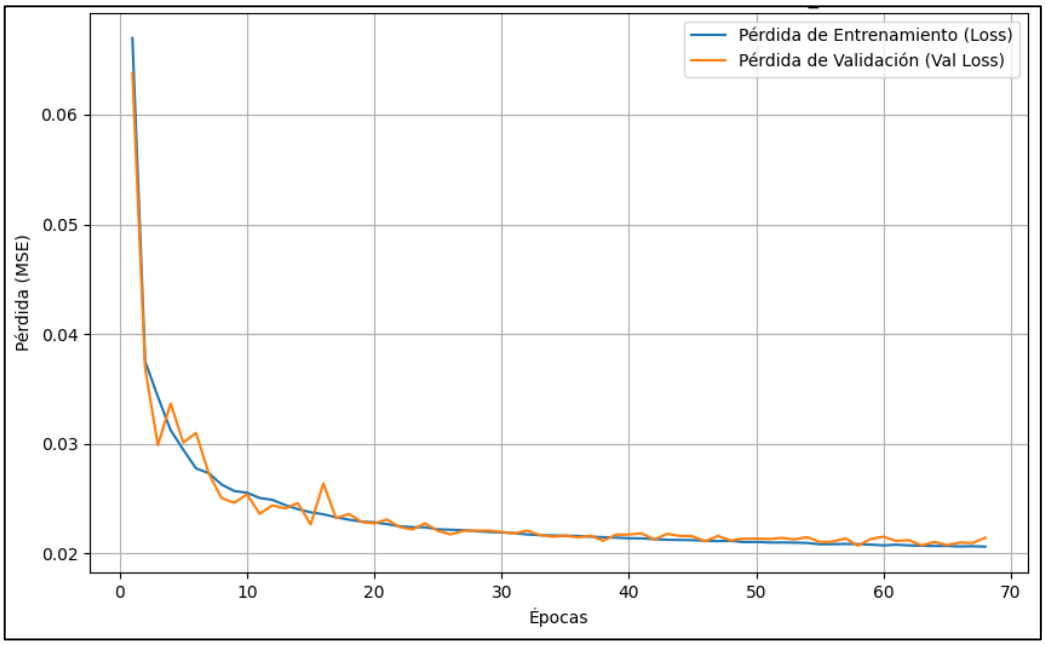
Diseño de una red neuronal de tipo GRU Model_GRU1

Arquitectura del modelo	Parámetros de aprendizaje	Hiperparámetros
• Red neuronal de tipo GRU. • Dos capas ocultas de 64 y 32 neuronas respectivamente. • una capa de salida con una neurona para regresión.	• Función de activación Tanh • Algoritmo de optimización Adam	• Learning rate: 0,001 • Tamaño de lote: 32 • Número de épocas: 100 • Sin tasa de regularización

Nota: La tabla describe a detalle la estructura de la red neuronal artificial de tipo GRU para el modelo de regresión denominado Model_GRU1.

Figura 27

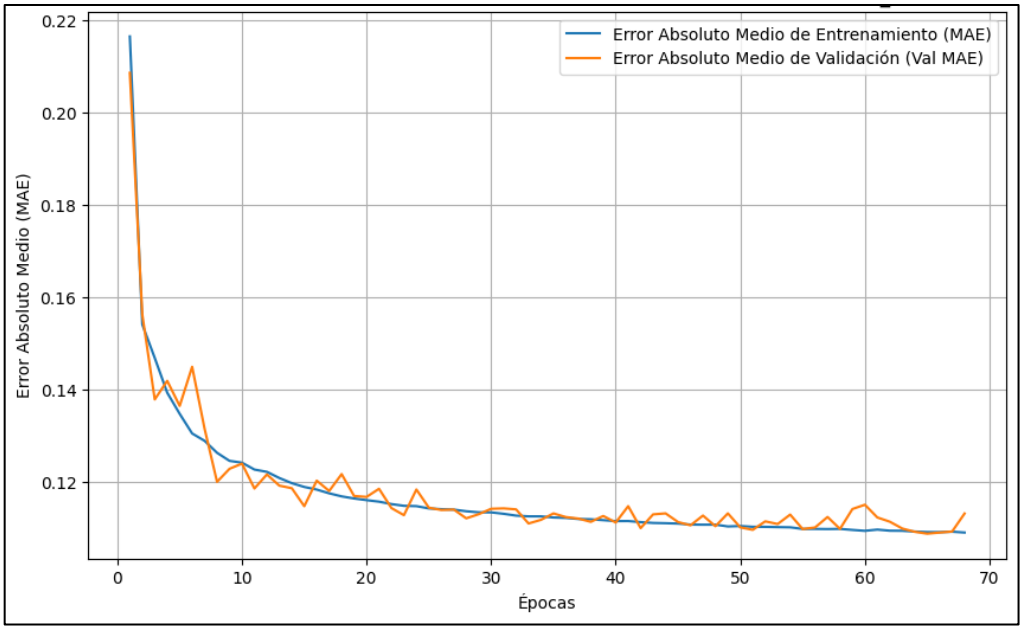
Evolución de la pérdida durante el entrenamiento Model_GRU1



Nota: la gráfica corresponde a la evolución de la pérdida de entrenamiento (Loss) y pérdida de validación (Val Loss) a lo largo de las 100 épocas establecidas para el entrenamiento, en este caso solo se completan menos de 70 épocas.

Figura 28

Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_GRU1



Nota: la gráfica corresponde a la evolución del Error Absoluto Medio (MAE) y Error Absoluto Medio (Val MAE) a lo largo de las 100 épocas establecidas para el entrenamiento., en este caso solo se completan 68 épocas dado que el modelo a partir de las 58 épocas no presenta mejoras notables.

La Figura 27, que representa la evolución de la pérdida durante el entrenamiento del modelo Model_GRU1, evidencia una notable estabilidad a lo largo de las épocas. Sin embargo, se observan pequeñas fluctuaciones en la pérdida de validación durante las primeras etapas del entrenamiento, las cuales se estabilizan a partir de la época 20, lo que sugiere una mejora en la estabilidad del entrenamiento.

Por su parte, la Figura 28, que representa la evolución del Error Absoluto Medio (Mean Absolute Error, MAE), también refleja un comportamiento relativamente estable. No obstante, el MAE correspondiente a la validación muestra fluctuaciones a lo largo de todo el proceso de entrenamiento, siendo estas más evidentes durante las primeras 20 épocas

Tabla 17

Diseño de una red neuronal de tipo GRU Model_GRU2

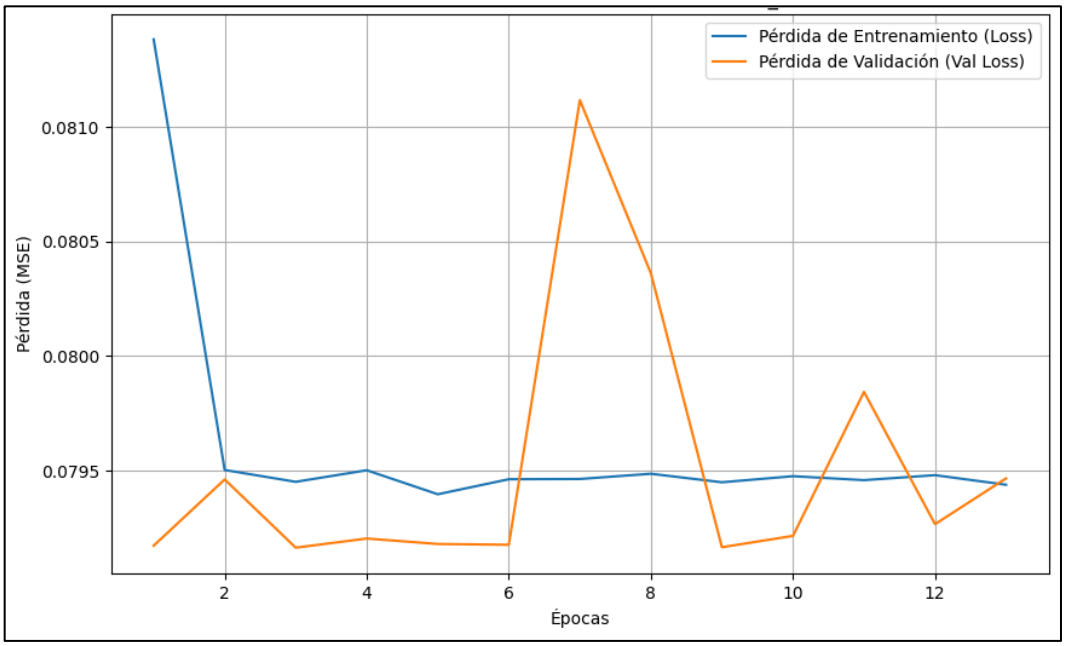
Arquitectura del modelo	Parámetros de aprendizaje	Hiperparámetros
• Red neuronal de tipo GRU. • Tres capas ocultas de 256, 128 y 64 neuronas respectivamente. • una capa de salida con una neurona para regresión.	• Función de activación Tanh para las capas GRU • Función de activación RELU, para las capas densas • Algoritmo de optimización Adam	• Learning rate: 0,01 • Tamaño de lote: 16 • Número de épocas: 50 • Dropout en 0,1 • Tasa de regularización L2

Nota: La tabla describe a detalle la estructura de la red neuronal artificial de tipo GRU para el modelo de regresión denominado Model_GRU2.

En la Figura 29, la cual corresponde la cual corresponde al Model_GRU2 demuestra que al añadir líneas de código que contienen reguladores que tienen como propósito mitigar el subajuste y sobreajuste afecta considerablemente el modelo, puesto que las curvas de entrenamiento y validación muestran fluctuaciones notables y no logran una estabilización en ninguna época, por lo tanto, el entrenamiento se detiene en la doceava interacción. Este comportamiento del modelo sugiere que las herramientas Dropout y L2 no son efectivas en este conjunto de datos.

Figura 29

Evolución de la pérdida durante el entrenamiento Model_GRU2



Nota: la gráfica corresponde a la evolución de la pérdida de entrenamiento (Loss) y pérdida de validación (Val Loss) a lo largo de las 100 épocas establecidas para el entrenamiento, en este caso solo se completan únicamente 13 épocas.

Tabla 18

Diseño de una red neuronal de tipo GRU Model_GRU3

Arquitectura del modelo	Parámetros de aprendizaje	Hiperparámetros
• Red neuronal de tipo GRU. • Cuatro capas ocultas de 256, 128, 64 y 32 neuronas respectivamente. • Dos capas densas de 128 y 64 neuronas respectivamente. • una capa de salida con una neurona para regresión.	• Función de activación Tanh para las capas GRU • Función de activación RELU, para las capas densas • Algoritmo de optimización Adam	• Learning rate: 0,001 • Tamaño de lote: 64 • Número de épocas: 50 • Sin tasa de regularización.

Nota: en la tabla 18 se representa las distintas configuraciones de la estructura de la red neuronal artificial de tipo GRU para el modelo de regresión denominado Model_GRU3

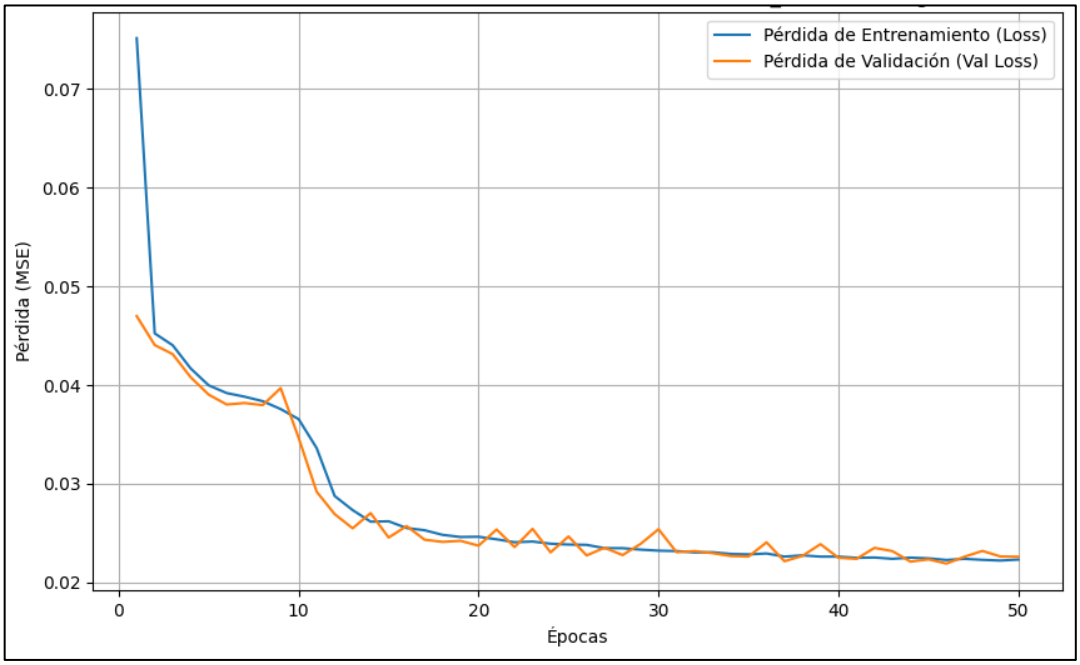
La evolución de la perdida durante el entrenamiento del Model_GRU3 se representa en la Figura 31, donde se muestra una tendencia a la baja en las primeras con una

pequeña meseta seguido una nueva disminución tanto en perdida como en validación, al final la reducción es más lenta y progresiva completándose el total de épocas establecidas.

La Figura 32 que representa la evolución del Error Absoluto Medio correspondiente al Model_GRU3 asume un comportamiento de similares caracteristicas a la Figura 30 que corresponde al a la evolución de pérdida durante el entrenamiento, es decir, la disminución es rápida en las primeras interacciones seguido de una meseta una estabilización con disminución progresiva y lenta.

Figura 30

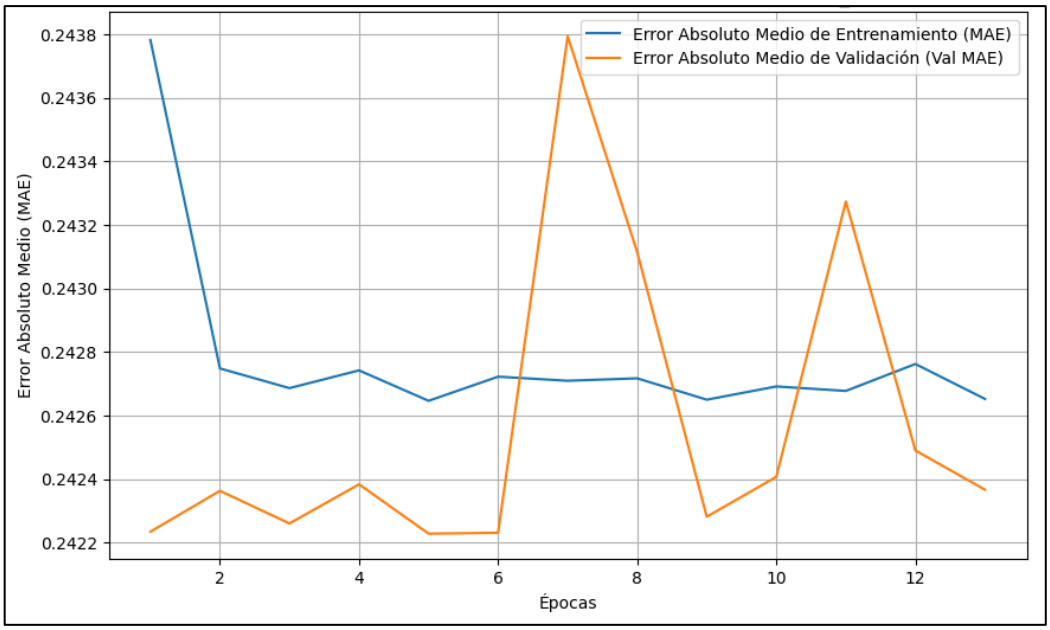
Evolución de la pérdida durante el entrenamiento Model_GRU3



Nota: la gráfica corresponde a la evolución de la pérdida de entrenamiento (Loss) y pérdida de validación (Val Loss) a lo largo de las 100 épocas establecidas para el entrenamiento, en este caso solo se completan 50 épocas.

Figura 31

Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_GRU2

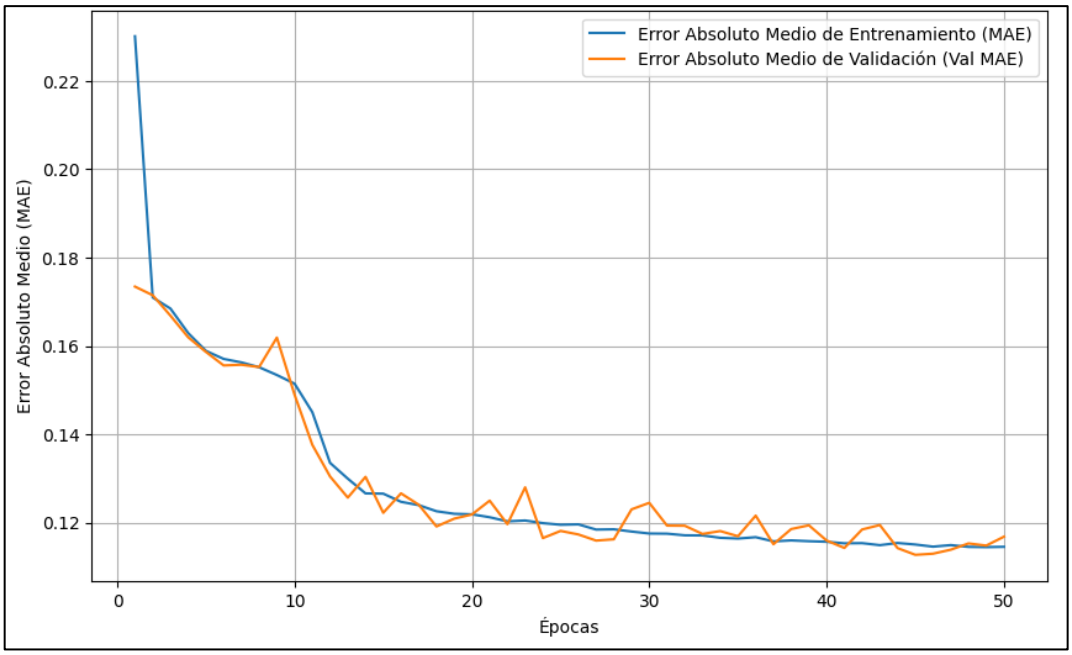


Nota: la gráfica corresponde a la evolución del Error Absoluto Medio (MAE) y Error Absoluto Medio (Val MAE) a lo largo de las 100 épocas establecidas para el entrenamiento., en este caso solo se completan 13 épocas dado que el modelo a partir de las 3 épocas no presenta mejoras notables.

A pesar de que la curva que representa la pérdida durante el entrenamiento evidencia menores variaciones en contraste con la curva de validación, el modelo claramente se ve comprometido. Este comportamiento también se refleja en la Figura 30, que representa la evolución del Error Absoluto Medio donde se observa una tendencia similar de inestabilidad y empeoramiento en el rendimiento del modelo.

Figura 32

Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_GRU3



Nota: la gráfica corresponde a la evolución del Error Absoluto Medio (MAE) y Error Absoluto Medio (Val MAE) a lo largo de las 50 épocas establecidas para el entrenamiento.

Tabla 19

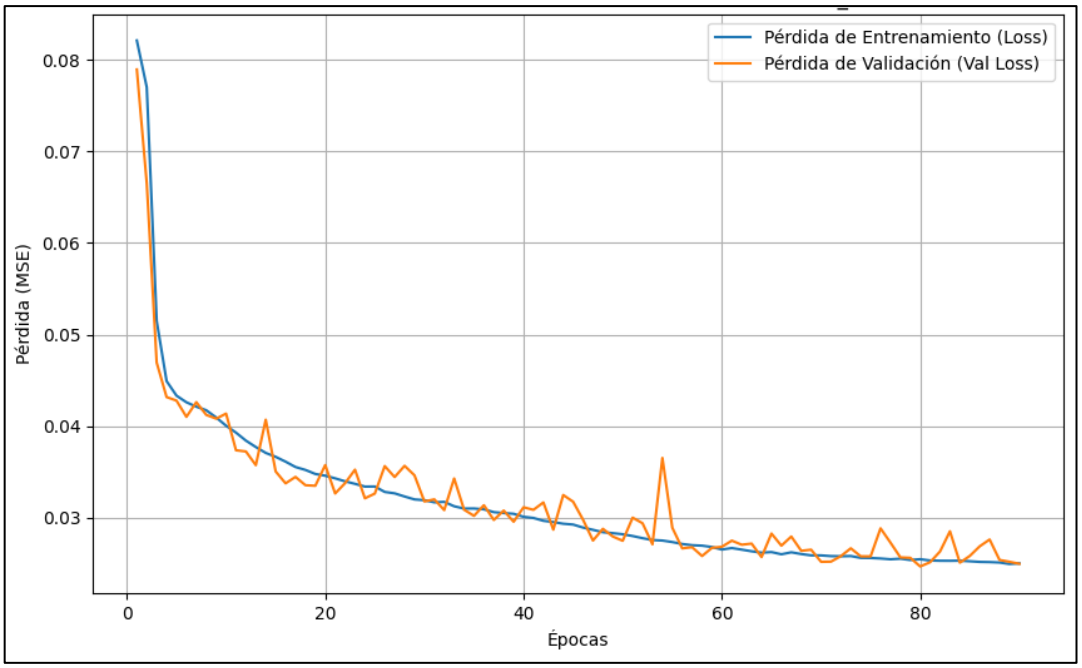
Diseño de una red neuronal de tipo LSMT Model_LSTM

Arquitectura del modelo	Parámetros de aprendizaje	Hiperparámetros
• Red neuronal de tipo GRU. • Dos capas ocultas de 64 y 32 neuronas respectivamente. • Una capa densa de 64 neuronas respectivamente. • una capa de salida con una neurona para regresión.	• Función de activación Tanh para las capas LSTM • Función de activación RELU, para las capas densas • Algoritmo de optimización Adam	• Learning rate: 0,0001 • Tamaño de lote: 32 • Numero de épocas: 100 • Sin taza de regularización.

Nota: la tabla que se muestra el detalle de la configuración estructural de una red neuronal artificial de tipo LSTM para el modelo de regresión que tiene por nombre Model_LSTM.

Figura 33

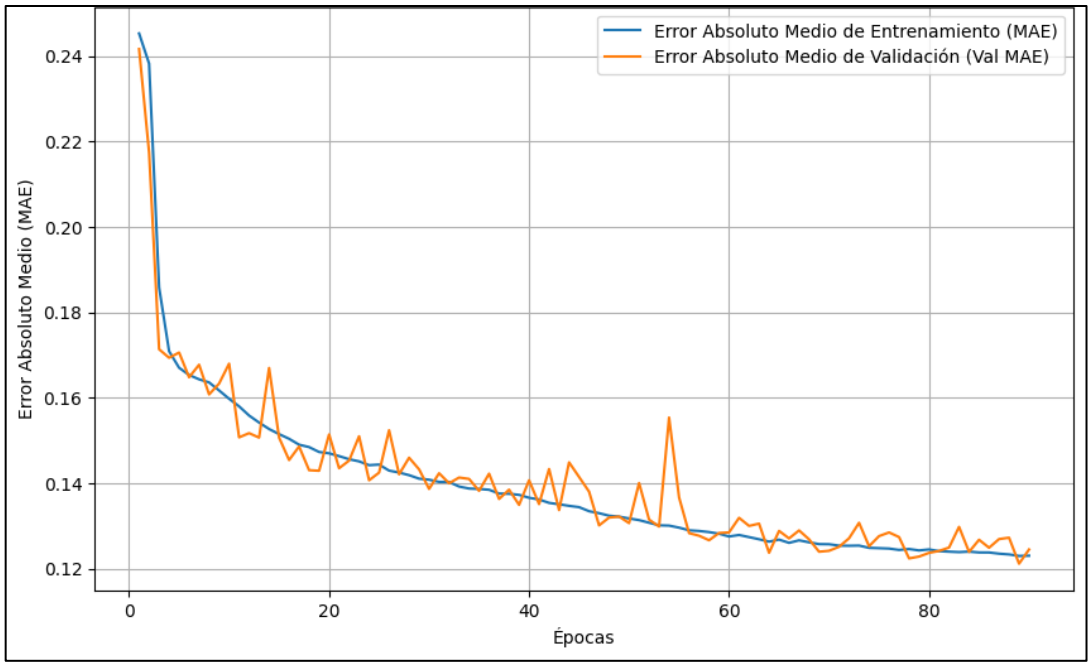
Evolución de la pérdida durante el entrenamiento Model_LSTM



Nota: la gráfica corresponde a la evolución de la pérdida de entrenamiento (Loss) y pérdida de validación (Val Loss) a lo largo de las 100 épocas establecidas para el entrenamiento, en este caso solo se completan 90 épocas dado que el modelo a partir de las 80 épocas no presenta mejoras notables.

Figura 34

Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_LSTM



Nota: la gráfica corresponde a la evolución del Error Absoluto Medio (MAE) y Error Absoluto Medio (Val MAE) a lo largo de las 100 épocas establecidas para el entrenamiento., en este caso solo se completan 90 épocas dado que el modelo a partir de las 80 épocas no presenta mejoras notables.

La pérdida durante el entrenamiento del Model_LSTM con su respectiva evolución se muestra en la Figura 33, la disminución es considerablemente rápida en las primeras interacciones, no obstante, a partir de las 40 épocas las curvas disminuyen de forma paulatina y estable, a pesar de las fluctuaciones existente en validación el entrenamiento del modelo muestra un buen desempeño hasta la época 70 luego este no presenta mejoras evidentes, lo cual obliga a la configuracion early stoping a detener el entrenamiento.

En la Figura 34 el comportamiento del Error Absoluto Medio muestra una tendencia similar al que se observó en la Figura 32 correspondiente a la pérdida en validación, de igual forma se muestra las fluctuaciones en validación que se hacen más evidentes aproximadamente a la mitad del total de épocas ejecutadas, en términos generales se puede decir que durante el entrenamiento este modelo presenta un comportamiento constante y poco variable.

Tabla 20

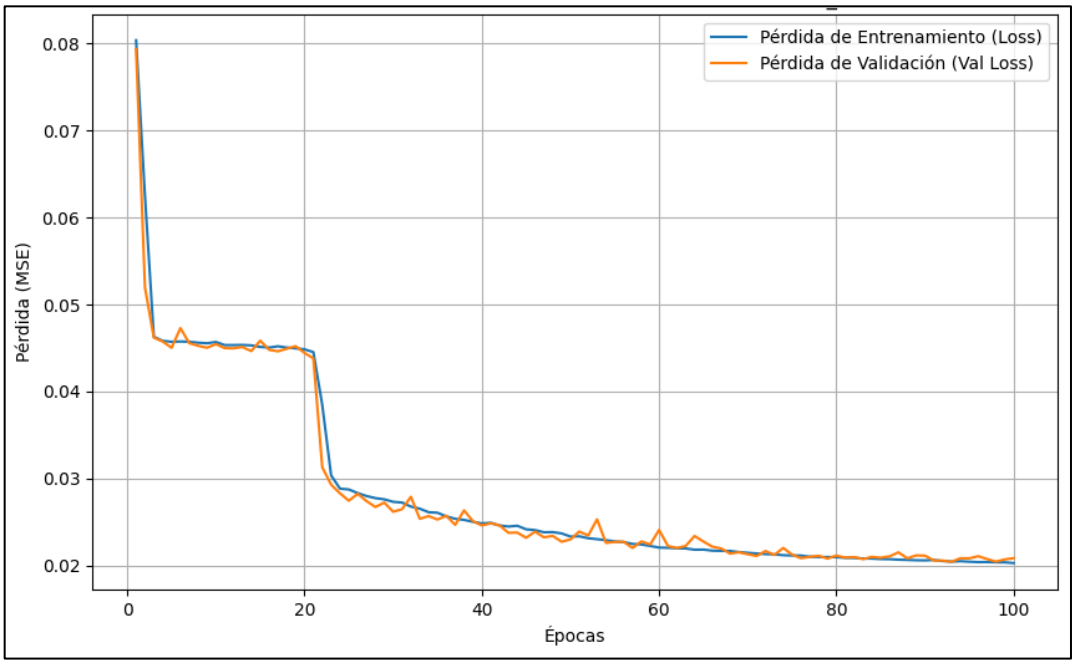
Diseño de una red neuronal de tipo LSMT Model_LSTM1

Arquitectura del modelo	**Parámetros de aprendizaje**	**Hiperparámetros**
• Red neuronal de tipo LSTM. • Dos capas ocultas de 256,128 y 64 neuronas respectivamente. • Tres capas densas de 128, 64 y 32 neuronas respectivamente. • una capa de salida con una neurona para regresión.	• Función de activación Tanh para las capas LSTM • Función de activación RELU, para las capas densas • Algoritmo de optimización Adam	• Learning rate: 0,001 • Tamaño de lote: 64 • Número de épocas: 100 • Sin tasa de regularización.

Nota: la tabla evidencia las configuraciones de la red neuronal de tipo LSTM para el modelo de regresión que lleva por nombre Model_LSMT1

Figura 35

Evolución de la pérdida durante el entrenamiento Model_LSTM1



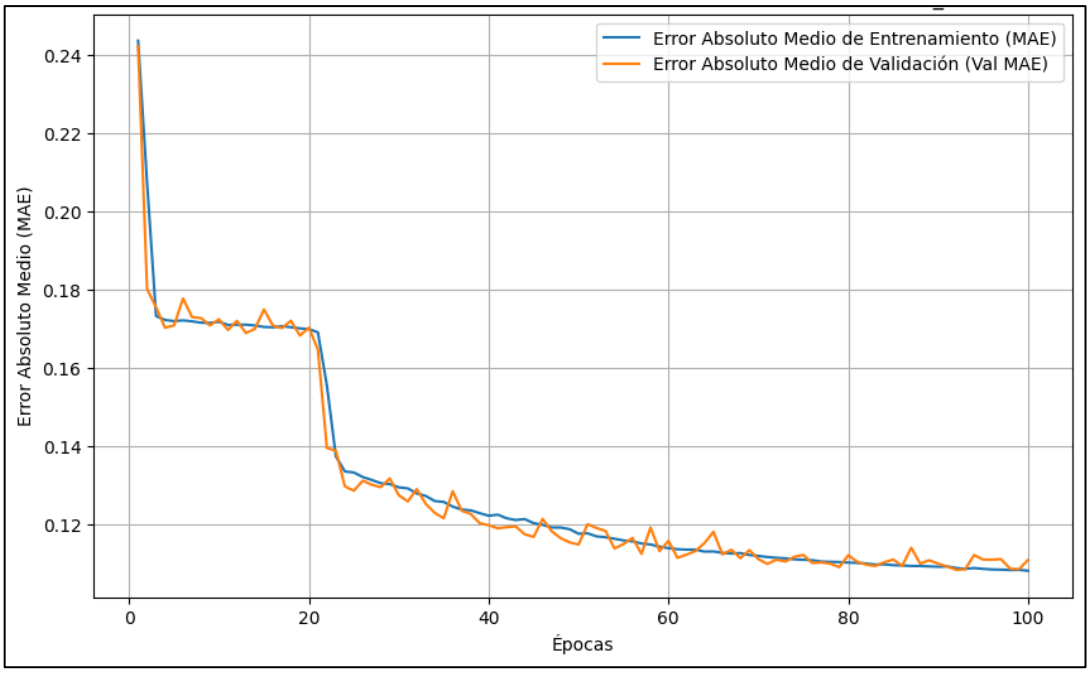
Nota: la gráfica corresponde a la evolución de la pérdida de entrenamiento (Loss) y pérdida de validación (Val Loss) a lo largo de las 100 épocas establecidas para el entrenamiento.

La Figura 35, que representa la evolución de la pérdida durante el entrenamiento del modelo Model_LSTM1, muestra un rápido descenso en las primeras épocas, seguido de la formación de una meseta inicial. Posteriormente, la pérdida experimenta una nueva disminución acelerada, tras lo cual el descenso tanto en la pérdida como en la validación se torna progresiva, acompañada de ligeras fluctuaciones en la métrica de pérdida de validación (Val Loss) a lo largo del entrenamiento.

La Figura 36 correspondiente a la evolución del Error Absoluto Medio (MAE), del Model_LSTM1 evidencia un comportamiento casi similar al descrito para la evolución de la pérdida durante el entrenamiento del modelo en cuestión. Cabe mencionar que, el presente modelo completó el total de épocas establecidas para el entrenamiento, lo cual infiere que el modelo mejora constantemente.

Figura 36

Evolución del Error Absoluto Medio (MAE) durante el entrenamiento Model_LSTM1



Nota: la gráfica corresponde a la evolución del Error Absoluto Medio (MAE) y Error Absoluto Medio (Val MAE) a lo largo de las 100 épocas establecidas para el entrenamiento.

Evaluar el rendimiento de los modelos a través de un conjunto de datos de prueba, a fin de avalar la precisión en la predicción de vida útil restante de turborreactores de doble flujo

Una vez terminado el entrenamiento de los modelos se procede a realizar la evaluación de cada uno de ellos empleando el conjunto de datos de prueba, a partir de allí se obtienen algunas métricas de evaluación que reflejan el desempeño de los modelos entrenados en la predicción de la vida útil restante de Turborreactores de doble flujo, para su posterior análisis y discusión en el CAPÍTULO IV.

CAPÍTULO IV

Resultados y discusión

4.1. Evaluación de los modelos entrenados

Para evaluar el desempeño de los modelos entrenados, se realizaron predicciones de la vida útil restante (RUL, Remaining Useful Life) utilizando el conjunto de prueba. El objetivo principal fue verificar la capacidad del modelo para estimar los valores de RUL, denominados Predicted_RUL. Posteriormente, se compararon estos valores predichos con los valores reales de RUL proporcionados por la base de datos.

La comparativa que se realiza para evaluar el desempeño de los modelos se realiza empleando los valores de RUL predicho en el último ciclo de cada máquina, esto debido a que los modelos realizan predicciones para cada ciclo, sin embargo, los valores de RUL real contienen un solo valor por máquina.

4.1.1 Model_básico

Las métricas de evaluación reflejan el desempeño del modelo de red neuronal artificial detallado en la Tabla 10 al predecir los valores de RUL (Remaining Useful Life). A continuación, se describe los valores obtenidos para este modelo y se interpreta cada uno de ellos,

- MSE: 0.0219
- MAE: 0.1155
- RMSE: 0.1479
- R^2 (Coeficiente de Determinación): 0.7004

El valor bajo del Error Cuadrático Medio (MSE) refleja que, en promedio, las predicciones se encuentran próximas a los valores reales. En este caso, el reducido

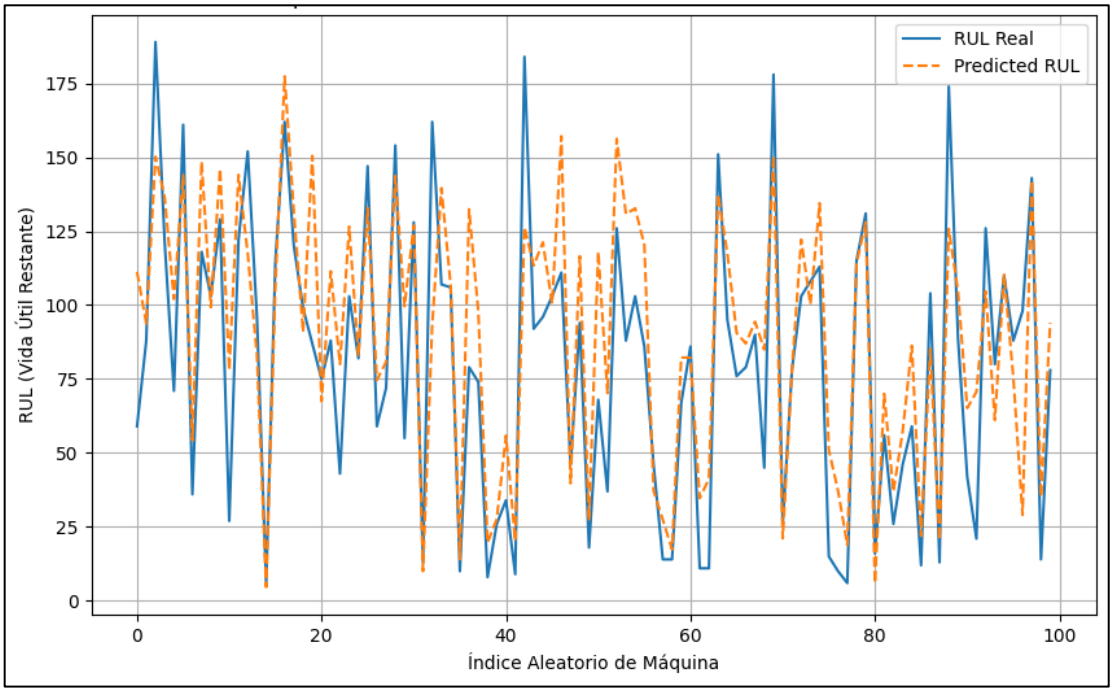
margen de error sugiere que el primer modelo básico presenta un ajuste adecuado a los datos.

En cambio, el Error Absoluto Medio indica una variabilidad ponderada de 0,1155 unidades en escala normalizada en referencia a los valores reales de RUL (Remaining Useful Life) que se muestran en la base de datos. La presente métrica establece una precisión medianamente aceptable, además, el Error Cuadrático Medio evidencia una disminución de la precisión de vida útil restante.

El coeficiente de determinación (R^2), muestra un valor por encima del 70 %, lo cual es un indicativo que el modelo en cuestión tiene capacidad para explicar ese porcentaje de variabilidad con respecto al 100 % existente. A pesar de que es una buena estimación no se logra la precisión requerida en aplicaciones críticas.

Figura 37

Comparación entre el RUL Real y Predicted Rul del Model_básico



Nota: La gráfica muestra una comparación entre las predicciones de RUL Real y los valores predichos por el modelo utilizando el conjunto de prueba, al cual se denominó Predicted RUL, para la gráfica se revirtió la normalización y se empleó de 14% del total de predicciones y valores reales.

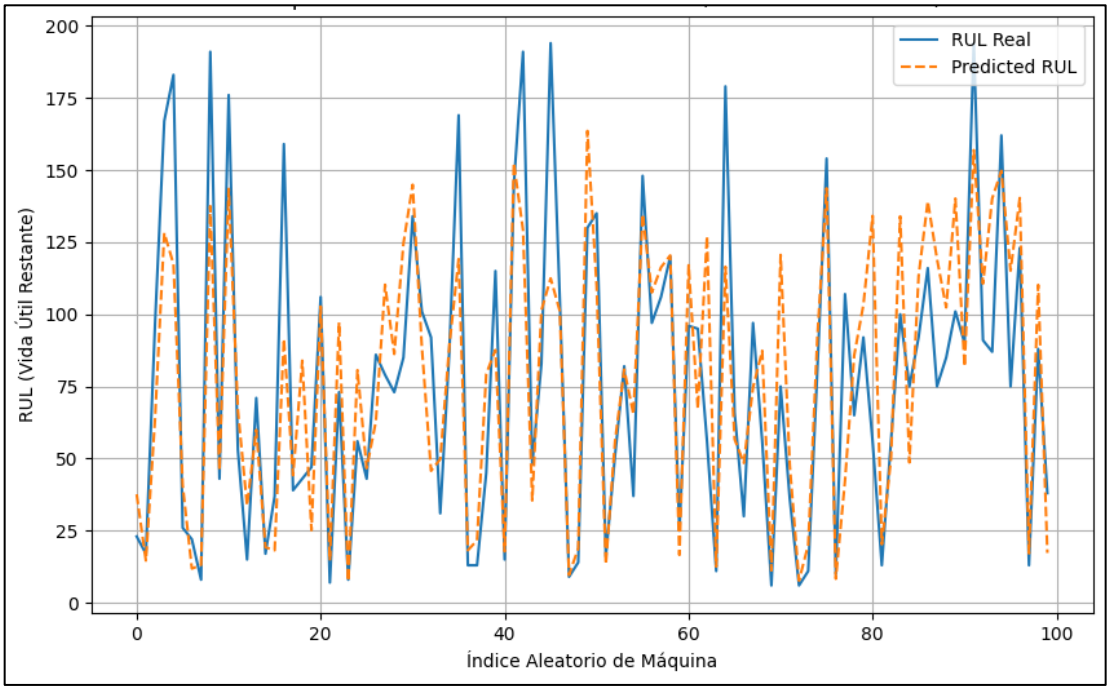
4.1.2 Model_básico1

Las métricas de evaluación que se muestran más adelante corresponden al desempeño del modelo descrito en la Tabla 11. A continuación, se detallan estas métricas y su evolución en comparación con el modelo inicial (Model_básico).

- MSE: 0.0194
- MAE: 0.1050
- RMSE: 0.1394
- R^2 (Coeficiente de Determinación): 0.7337

Figura 38

Comparación entre el RUL Real y Predicted Rul del model_básico1



Nota: La gráfica muestra una comparación entre las predicciones de RUL Real y los valores predichos por el modelo utilizando el conjunto de prueba, al cual se denominó Predicted RUL, para la gráfica se revirtió la normalización y se empleó el 14% del total de predicciones y valores reales.

El modelo presenta un avance palpable en contraste al Model_básico de acuerdo con las métricas de evaluación, los valores correspondientes al Error Cuadrático Medio (MSE), el Error Absoluto Medio (MAE) y la Raíz del Error Cuadrático Medio (RMSE)

se ven reducidas, además, el coeficiente de determinación (R^2) experimenta un aumento en su métrica fijándose en un valor levemente superior al 73%, con esto se deduce que el modelo tiene una capacidad de explicar ese porcentaje de variabilidad.

El aumento de la precisión alcanzado por el modelo responde al incremento de las capas ocultas densas, además de la incorporación de un mayor número de neuronas en ellas, también se realizó un ajuste interactivo de los parámetros e Hiperparámetros, modificando el tamaño de lote, número de épocas o interacciones, entre otros aspectos.

4.1.3 Model_básico2

Las siguientes métricas de evaluación corresponden al desempeño del modelo que se describen en la Tabla 12. A continuación se describen las métricas y su evolución con respecto al primer modelo (Model_básico).

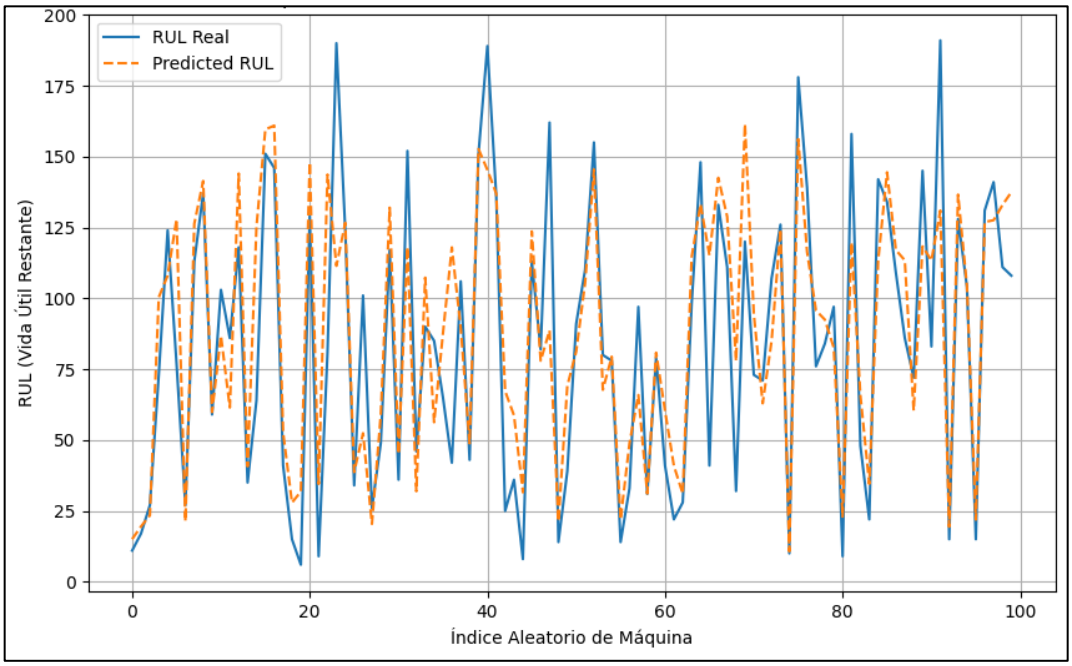
- MSE: 0.0204
- MAE: 0.1084
- RMSE: 0.1428
- R^2 (Coeficiente de Determinación): 0.7208

Las métricas de evaluación evidencian una mejora en comparación con el primer modelo (Model_básico); sin embargo, no se observa un avance significativo respecto al segundo modelo (Model_básico1). Este resultado se refleja en los valores de las métricas clave, como el Error Cuadrático Medio (MSE), el Error Absoluto Medio (MAE) y la Raíz del Error Cuadrático Medio (RMSE), las cuales no muestran una reducción adicional en relación con el desempeño del Model_básico1.

Como resultado de lo antes descrito se evidencia un coeficiente de determinación levemente inferior al modelo antecesor con una variación de un punto porcentual, lo cual indica que el presente modelo no supera lo hecho por el Model_básico1.

Figura 39

Comparación entre el RUL Real y Predicted Rul del model_básico2



Nota: La gráfica muestra una comparación entre las predicciones de RUL Real y los valores predichos por el modelo utilizando el conjunto de prueba, al cual se denominó Predicted RUL, para la gráfica se revirtió la normalización y se empleó el 14% del total de predicciones y valores reales.

4.1.4 Model_básico3

Las siguientes métricas de evaluación corresponden al desempeño del modelo que se describen en la Tabla 13. A continuación se describe las métricas y su evolución con respecto a los modelos anteriores.

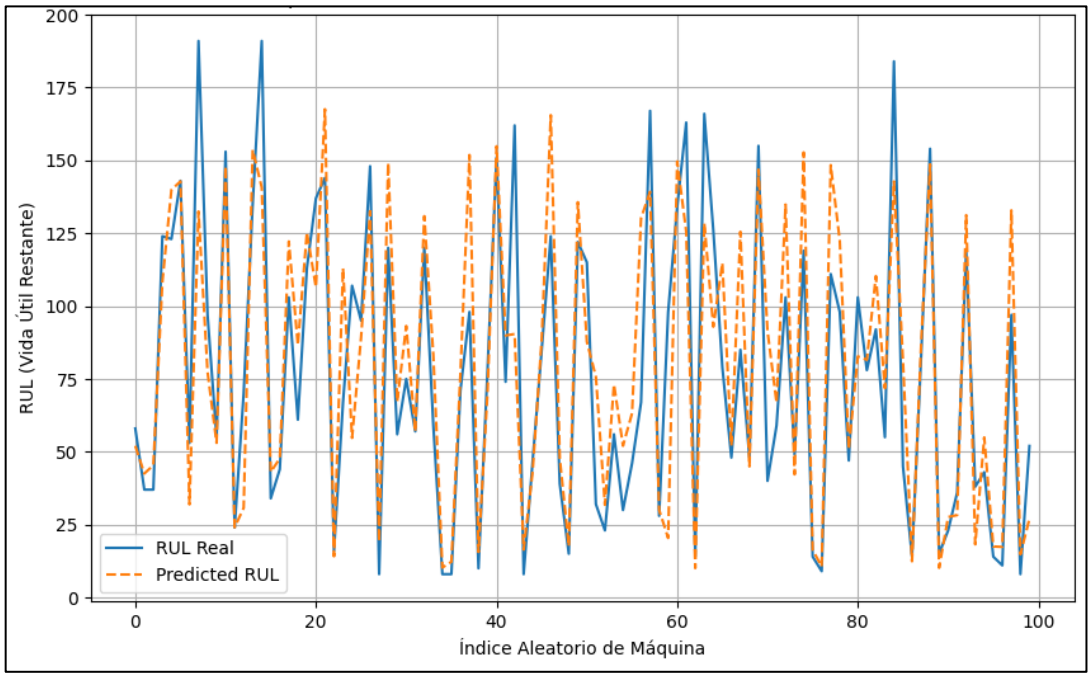
- MSE: 0.0195
- MAE: 0.1041
- RMSE: 0.1396
- R^2 (Coeficiente de Determinación): 0.7330

Las métricas de evaluación muestran una mejora únicamente en comparación con el Model_básico y el Model_básico2. No obstante, el desempeño del modelo presenta una disminución en precisión frente al Model_básico1. Esto se evidencia en un coeficiente de determinación del 73,3 %, ligeramente inferior al alcanzado por el

Model_básico1, lo que indica que las optimizaciones implementadas no lograron superar la efectividad de este último. Estos resultados subrayan la necesidad de explorar ajustes adicionales o enfoques alternativos para mejorar el rendimiento del modelo.

Figura 40

Comparación entre el RUL Real y Predicted Rul del model_básico3



Nota: La gráfica muestra una comparación entre las predicciones de RUL Real y los valores predichos por el modelo utilizando el conjunto de prueba, al cual se denominó Predicted RUL, para la gráfica se revirtió la normalización y se empleó el 14% del total de predicciones y valores reales.

4.1.5 Model_básico4

Las siguientes métricas de evaluación corresponden al desempeño del modelo que se describen en la Tabla 14. A continuación se describen las métricas y su evolución con respecto a los modelos anteriores.

MSE: 0.0447

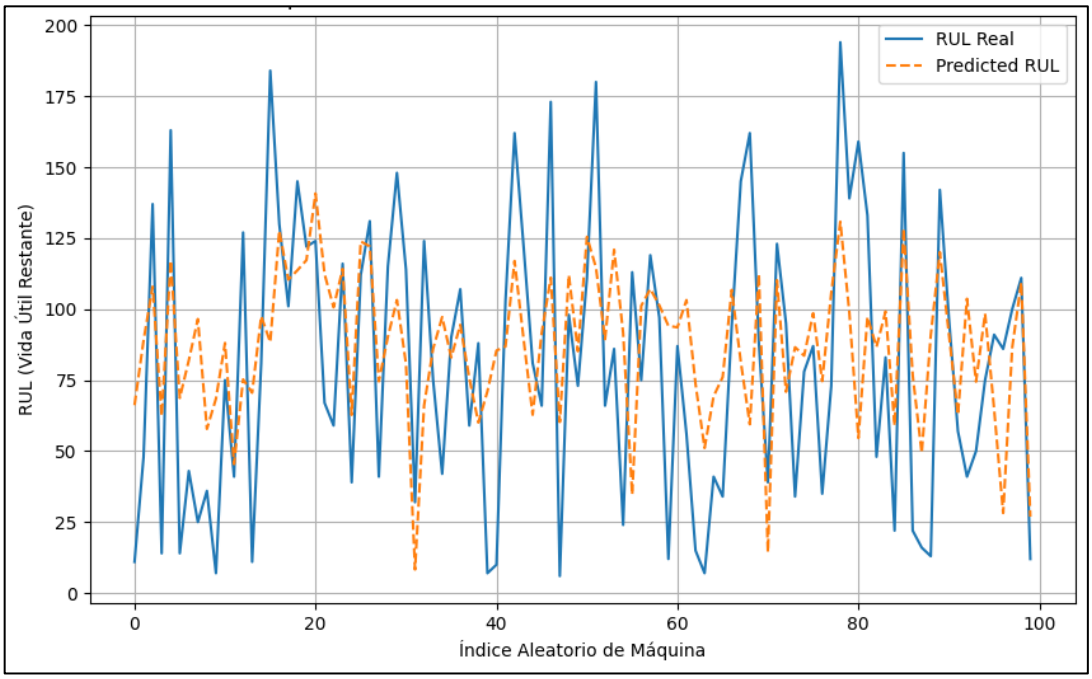
MAE: 0.1738

RMSE: 0.2115

R^2 (Coeficiente de Determinación): 0.3874

Figura 41

Comparación entre el RUL Real y Predicted Rul del model_básico4



Nota: La gráfica muestra una comparación entre las predicciones de RUL Real y los valores predichos por el modelo utilizando el conjunto de prueba, al cual se denominó Predicted RUL, para la gráfica se revirtió la normalización y se empleó el 14% del total de predicciones y valores reales.

Las métricas del presente modelo (Model_básico4) demuestran una reducción en la precisión de la predicción de RUL, evidenciando un aumento del Error Cuadrático Medio (MSE), Error Absoluto Medio (MAE) y la Raíz del Error Cuadrático Medio (RMSE), de igual forma el coeficiente de determinación sufre una disminución de la explicabilidad en la variabilidad presente en los datos, reduciendo en aproximadamente la mitad en relación con los otros modelos básicos.

Cabe señalar que, en el código de entrenamiento del presente modelo se incorporó técnicas de regularización como el Dropout y L2 a fin de evitar el sobreajuste y subajuste respectivamente. A pesar de que estas estrategias deberían mejorar el rendimiento del modelo, por el contrario, tienden a empeorar significativamente la capacidad de generalización con datos no vistos por el modelo. Lo cual hace deducir que estas estrategias implementadas son definitivamente poco efectivas en este tipo de arquitectura y base de datos.

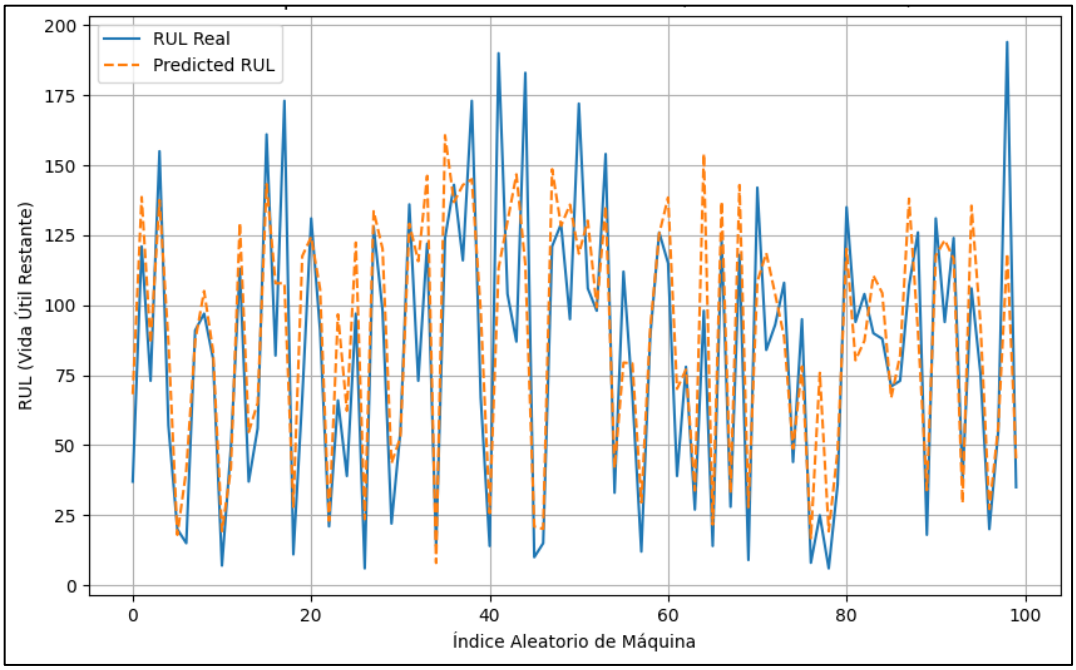
4.1.6 Model_GRU

Las siguientes métricas de evaluación corresponden al desempeño del modelo que se describen en la Tabla 15. A continuación se describen las métricas y su evolución con respecto a los modelos anteriores.

MSE: 0.0206
MAE: 0.1114
RMSE: 0.1434
R^2 (Coeficiente de Determinación): 0.7184

Figura 42

Comparación entre el RUL Real y Predicted Rul del Model_GRU



Nota: La gráfica muestra una comparación entre las predicciones de RUL Real y los valores predichos por el modelo utilizando el conjunto de prueba, al cual se denominó Predicted RUL, para la gráfica se revirtió la normalización y se empleó el 14% del total de predicciones y valores reales.

Las métricas de evaluación del modelo basado en la arquitectura GRU muestran un nivel de mejora en comparación con el Model_básico. Es importante señalar que ambos modelos comparten configuraciones básicas, caracterizadas por la inclusión de una sola capa oculta y un número reducido de neuronas. Esto contrasta con

configuraciones más avanzadas presentes en otros modelos, como el Model_básico1, el Model_básico2 y sus derivados, los cuales incluyen arquitecturas más complejas y mayor capacidad de aprendizaje. A pesar de su simplicidad, el modelo GRU logra un desempeño ligeramente superior, destacando su eficiencia en configuraciones básicas.

4.1.7 Model_GRU1

Las siguientes métricas de evaluación corresponden al desempeño del modelo que se describen en la Tabla 16. A continuación se describen las métricas y su evolución con respecto a los modelos anteriores.

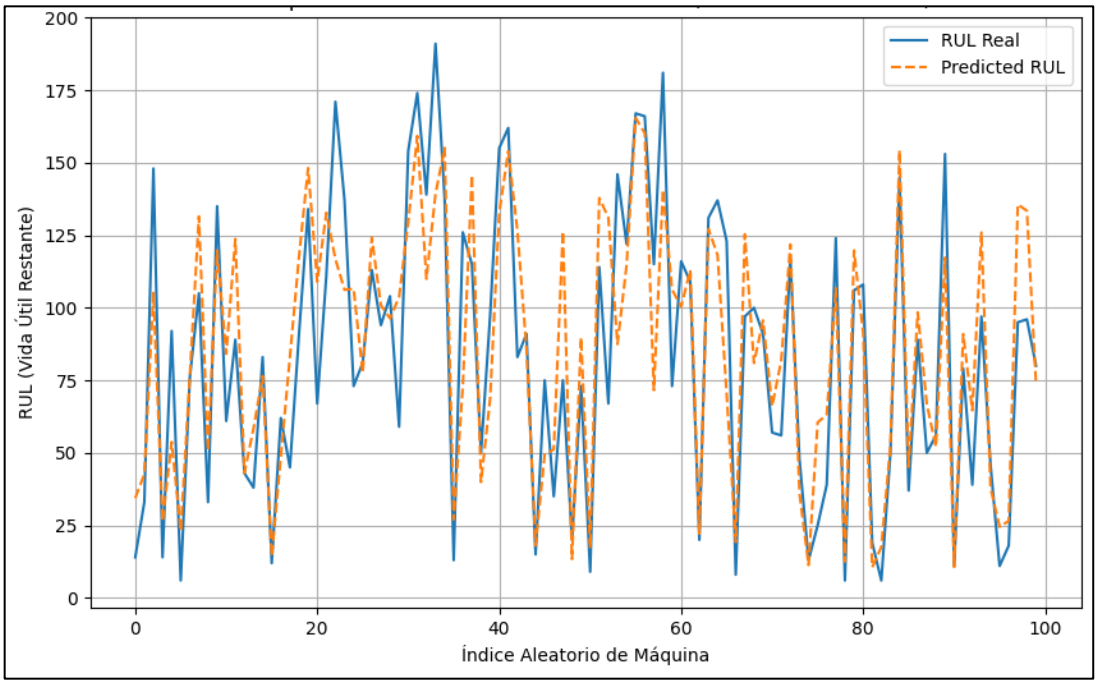
- MSE: 0.0196
- MAE: 0.1057
- RMSE: 0.1401
- R^2 (Coeficiente de Determinación): 0.7314

Las métricas de evaluación del Model_GRU1 reflejan una ligera mejora en comparación con su predecesor, el Model_GRU, que presenta una arquitectura similar. Se observa una reducción moderada en los valores de las métricas como el MSE, el MAE y el RMSE, acompañada de un incremento en el coeficiente de determinación, que pasa de casi un 72 % a un 73 %, lo que equivale a una mejora de un punto porcentual.

El presente modelo evidencia una leve mejora con respecto a su antecesor, no obstante, este no logra superar lo hecho por los modelos básicos como: Model_básico1 y Model_básico3 respectivamente, las variaciones son poco perceptibles considerando que el modelo cuenta con una arquitectura de tipo GRU, además la explicación de la variabilidad existente se mantiene entorno al 70%, barrera la cual hasta el momento ningún modelo logra superar con las variaciones realizadas.

Figura 43

Comparación entre el RUL Real y Predicted Rul del Model_GRU1



Nota: La gráfica muestra una comparación entre las predicciones de RUL Real y los valores predichos por el modelo utilizando el conjunto de prueba, al cual se denominó Predicted RUL, para la gráfica se revirtió la normalización y se empleó el 14% del total de predicciones y valores reales de RUL.

4.1.8 Modelo GRU2

Las siguientes métricas de evaluación corresponden al desempeño del modelo que se describen en la Tabla 17. A continuación se describen las métricas y su evolución con respecto a los modelos anteriores.

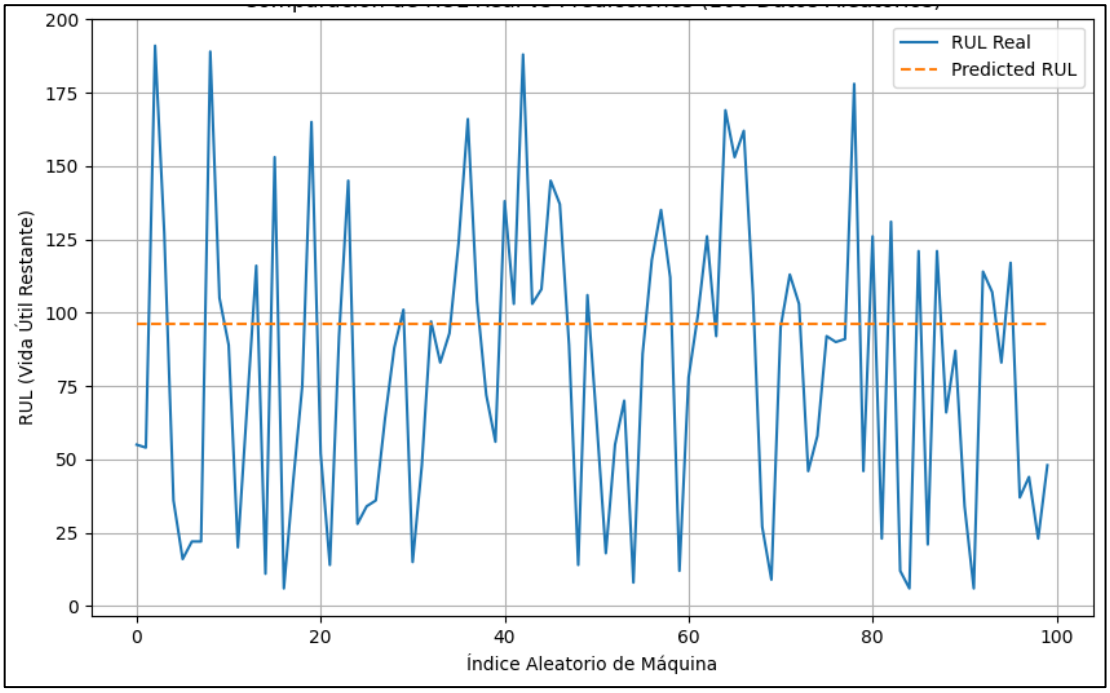
- MSE: 0.0791
- MAE: 0.2355
- RMSE: 0.2812
- R^2 (Coeficiente de Determinación): -0.0830

El modelo presenta un deterioro significativo en su rendimiento, como lo reflejan las métricas de evaluación: el Error Cuadrático Medio (MSE), el Error Absoluto Medio (MAE) y la Raíz del Error Cuadrático Medio (RMSE) son considerablemente peores

en comparación con los modelos de capas densas y sus predecesores basados en la arquitectura GRU. Además, el coeficiente de determinación es negativo, lo que indica un ajuste deficiente y un desempeño claramente insuficiente en la predicción.

Figura 44

Comparación entre el RUL Real y Predicted Rul del Model_GRU2



Nota: La gráfica muestra una comparación entre las predicciones de RUL Real y los valores predichos por el modelo utilizando el conjunto de prueba, al cual se denominó Predicted RUL, para la gráfica se revirtió la normalización y se empleó el 14% del total de predicciones y valores reales de RUL.

En términos generales el modelo empeora notablemente en su capacidad predictiva, obteniendo valores completamente invariables con respecto a la variabilidad presentada en los datos. Cabe indicar que este modelo al igual que en las capas densas se implementa técnicas de regularización como Dropout y L2 a fin de evitar sobreajuste y subajuste.

4.1.9 Modelo GRU3

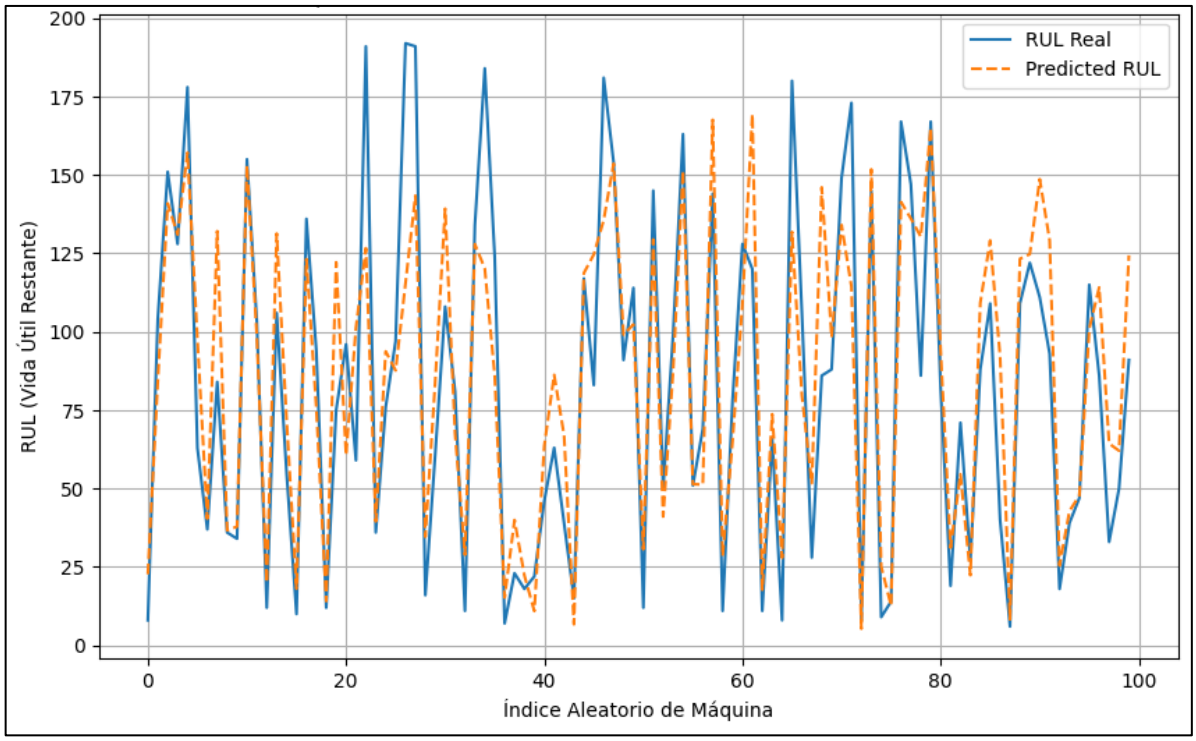
Las siguientes métricas de evaluación corresponden al desempeño del modelo que se describen en la Tabla 18. A continuación se describen las métricas y su evolución con respecto a los modelos anteriores.

- MSE: 0.0205
- MAE: 0.1085
- RMSE: 0.1432
- R^2 (Coeficiente de Determinación): 0.7192

Las métricas de evaluación del Model_GRU3 muestran un desempeño claramente superior en comparación con el Model_GRU2. Sin embargo, este modelo no logra superar los resultados obtenidos por el Model_GRU1 y apenas presenta una ligera mejora respecto al Model_GRU. Además, su rendimiento tampoco alcanza el de los modelos de capas densas, los cuales establecen un coeficiente de determinación cercano al 73 %, valor que también se registra para el Model_GRU3. Estos resultados destacan que, si bien el Model_GRU3 evidencia ciertas mejoras, aún no logra mejorar el desempeño de las arquitecturas densas en esta investigación.

Figura 45

Comparación entre el RUL Real y Predicted Rul del Model_GRU3



Nota: La gráfica muestra una comparación entre las predicciones de RUL Real y los valores predichos por el modelo utilizando el conjunto de prueba, al cual se denominó Predicted RUL, para la gráfica se revirtió la normalización y se empleó el 14% del total de predicciones y valores reales de RUL.

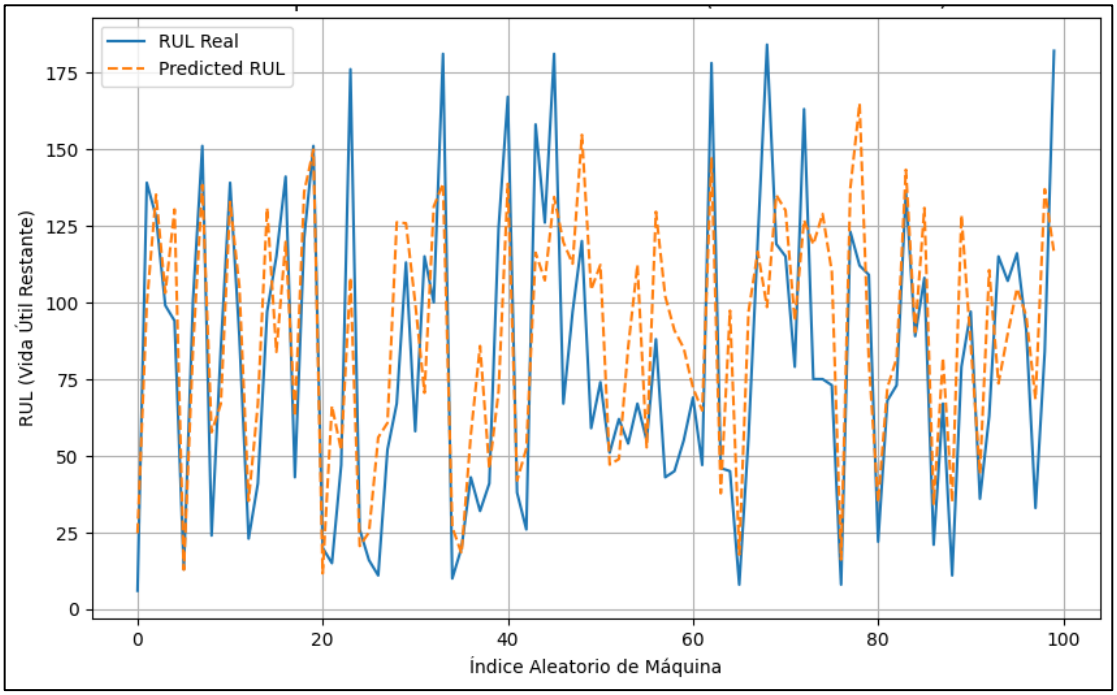
4.1.10 Modelo LSTM

Las siguientes métricas de evaluación corresponden al desempeño del modelo que se describen en la Tabla 19. A continuación se describen las métricas y su evolución con respecto a los modelos anteriores.

- MSE: 0.0245
- MAE: 0.1246
- RMSE: 0.1564
- R^2 (Coeficiente de Determinación): 0.6650

Figura 46

Comparación entre el RUL Real y Predicted Rul del Model_LSTM



Nota: La gráfica muestra una comparación entre las predicciones de RUL Real y los valores predichos por el modelo utilizando el conjunto de prueba, al cual se denominó Predicted RUL, para la gráfica se revirtió la normalización y se empleó el 14% del total de predicciones y valores reales de RUL.

El modelo que contiene capas ocultas de tipo LSTM muestra un desempeño razonable, no obstante, no logra superar los modelos con arquitecturas de tipo densas y GRU respectivamente, las métricas de evaluación en cuanto Error Cuadrático Medio

(MSE), el Error Absoluto Medio (MAE) y la Raíz del Error Cuadrático Medio (RMSE) con más elevadas en contraste con los mejores modelos entrenados en las arquitecturas antes descritas, además, el coeficiente de determinación desciende por debajo del 70%, este es un punto clave, puesto que los anteriores modelos no se desploman por debajo de este limite a excepción de aquellos con regularizadores Dropout y L2.

4.1.11 Modelo LSTM1

Las siguientes métricas de evaluación corresponden al desempeño del modelo que se describen en la Tabla 20. A continuación se describen las métricas y su evolución con respecto a los modelos anteriores.

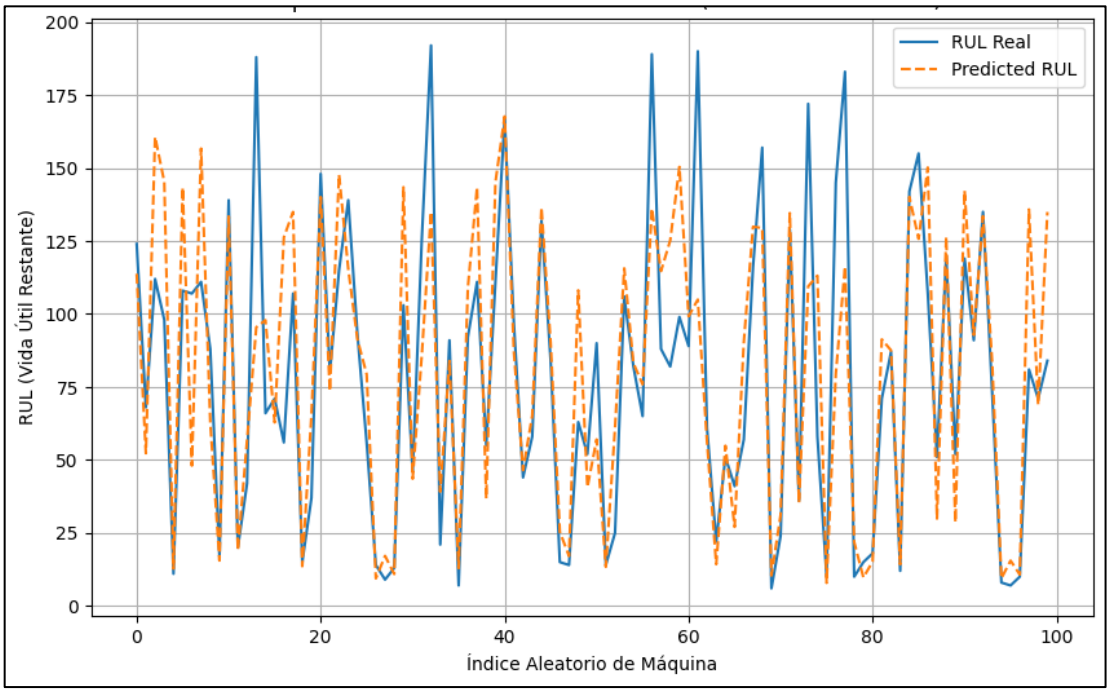
- MSE: 0.0198
- MAE: 0.1046
- RMSE: 0.1405
- R^2 (Coeficiente de Determinación): 0.7295

A partir del análisis de las métricas de evaluación del Model_LSTM1 permite concluir que este modelo muestra una mejora en su desempeño en comparación con su predecesor, el Model_LSTM. No obstante, su eficiencia predictiva no logra superar la alcanzada por el Model_GRU1 y el Model_básico1, lo que sugiere que, aunque las optimizaciones implementadas han tenido un efecto positivo respecto a su versión inicial, aún no son suficientes para igualar o superar los resultados de los modelos más efectivos.

El modelo logra explicar más del 70 % de variabilidad existente, superando lo hecho por su antecesor el Model_LSTM, este avance es considerablemente significativo, teniendo en cuenta que en arquitecturas como las de tipo GRU el avance en la precisión era más lento entorno a un máximo de 3 puntos porcentuales en el coeficiente de determinación, no obstante, este modelo requiere de un mayor costo computacional, e inversión de tiempo en el entrenamiento.

Figura 47

Comparación entre el RUL Real y Predicted Rul del Model_LSTM1



Nota: La gráfica muestra una comparación entre las predicciones de RUL Real y los valores predichos por el modelo utilizando el conjunto de prueba, al cual se denominó Predicted RUL, para la gráfica se revirtió la normalización y se empleó el 14% del total de predicciones y valores reales de RUL.

4.2. Análisis de resultados en escala Original

En este apartado se procede analizar las métricas de evaluación en base a la escala original de los datos, además, se considera que las métricas de evaluación MAE y RMSE están en las mismas unidades en las que se fija la variable objetivo, para la presente investigación es la Vida Útil Restante o RUL por sus siglas en inglés, en ciclos, entre tanto, la métrica correspondiente al coeficiente de determinación se mide en porcentaje (%), finalmente el MSE se mide en Ciclos^2.

Con base en la Figura 48, que ilustra el rendimiento de los modelos entrenados, se puede afirmar que, en términos generales, los modelos presentan la capacidad de predecir el Remaining Useful Life (RUL) con un margen de error que oscila entre ±20 y ±45 ciclos, según el Error Absoluto Medio (MAE). Dentro de este análisis, el modelo de peor desempeño es el Model_GRU2, mientras que el Model_básico3 destaca como el más preciso.

Tabla 21

Métricas de evaluación de los modelos RNA en escala original

	Métricas normalizadas				Métricas en escala original			
Modelos	**MSE**	**MAE**	**RMSE**	**R^2**	**MSE**	**MAE**	**RMSE**	**R^2 (%)**
Model_básico	0.022	0.116	0.148	0.700	782.290	21.830	27.953	70.040
Model_básico1	0.019	0.105	0.139	0.734	692.987	19.845	26.347	73.370
Model_básico2	0.020	0.108	0.143	0.721	728.708	20.488	26.989	72.080
Model_básico3	0.020	0.104	0.140	0.733	696.560	19.675	26.384	73.300
Model_básico4	0.045	0.174	0.212	0.387	1596.729	32.848	39.974	38.740
Model_GRU	0.021	0.111	0.143	0.718	735.853	21.055	27.103	71.840
Model_GRU1	0.020	0.106	0.140	0.731	700.132	19.977	26.479	73.140
Model_GRU2	0.079	0.236	0.281	-0.083	2825.531	44.510	53.147	-8.300
Model_GRU3	0.021	0.109	0.143	0.719	732.281	20.507	27.065	71.920
Model_LSTM	0.025	0.125	0.156	0.665	875.165	23.549	29.560	66.500
Model_LSTM1	0.020	0.105	0.141	0.730	707.276	19.769	26.555	72.950

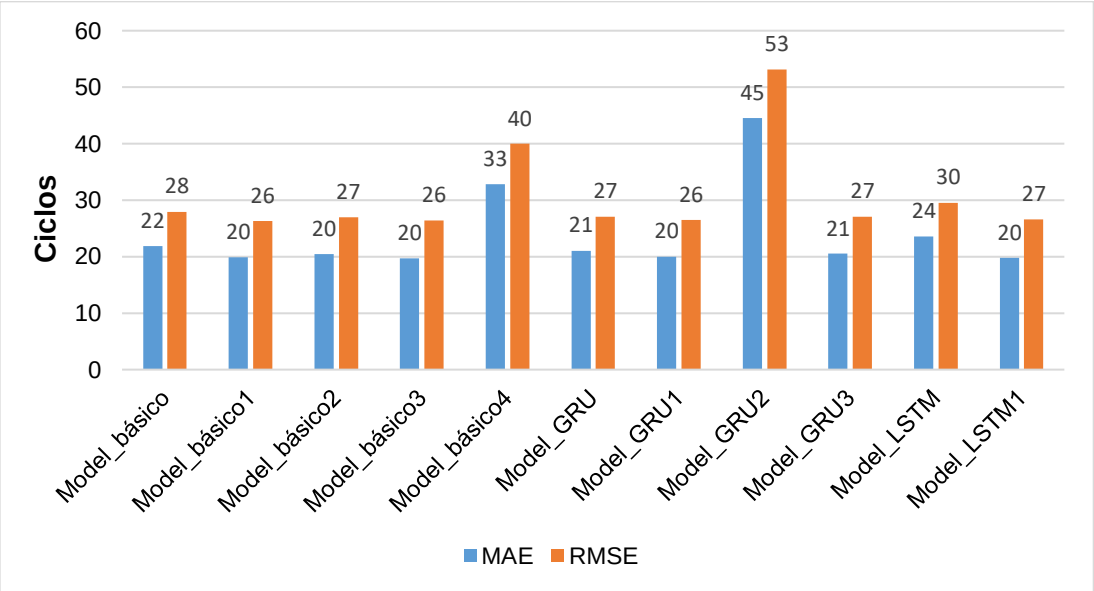
Nota: la tabla muestra las métricas de evaluación utilizadas para evaluar los modelos de RNA, describiéndose el Error cuadrático Medio (MSE), Error Absoluto Medio (MAE), Raíz del Error Cuadrático Medio (RMSE) y el Coeficiente de determinación (R^2).

A partir del análisis del Error Cuadrático Medio (RMSE), el margen de error en la predicción de RUL aumenta levemente con respecto al Error Absoluto Medio (MAE), fijándose alrededor de ±26 ciclos el mejor modelo y ±53 ciclos el peor modelo, además esta métrica es más realista puesto que penaliza los errores grandes arrojados durante la predicción de vida útil restante, a pesar que el mejor modelo logra una precisión aceptable esta no es suficiente en aplicaciones críticas como la propuesta en la presente investigación.

Las leves diferencias entre las métricas del Error Absoluto Medio (MAE) y la Raíz del Error Cuadrático Medio (RMSE) evidencian la presencia de errores considerables en las predicciones de RUL, la variabilidad de la precisión se fija en torno a ±6 ciclos para el mejor modelo y ±8 ciclos para el peor modelo, además las dos métricas sugieren que el peor modelo posible sigue siendo el Model_GRU2.

Figura 48

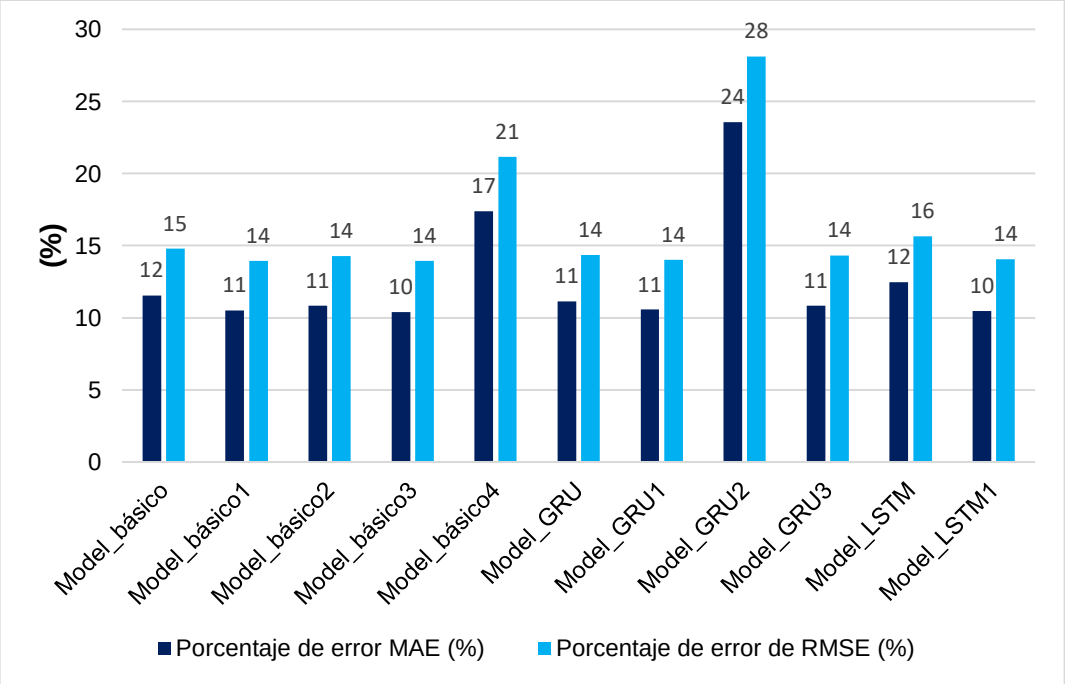
Análisis del rendimiento los modelos en base al MAE y RMSE en escala original



Nota: la gráfica muestra el rendimiento de los modelos entrenados, con las métricas de evaluación correspondientes al Error Absoluto Medio (MAE) y la Raíz del Error Cuadrático Medio (RMSE).

Figura 49

Análisis del rendimiento los modelos en base al porcentaje de error de MAE y RMSE



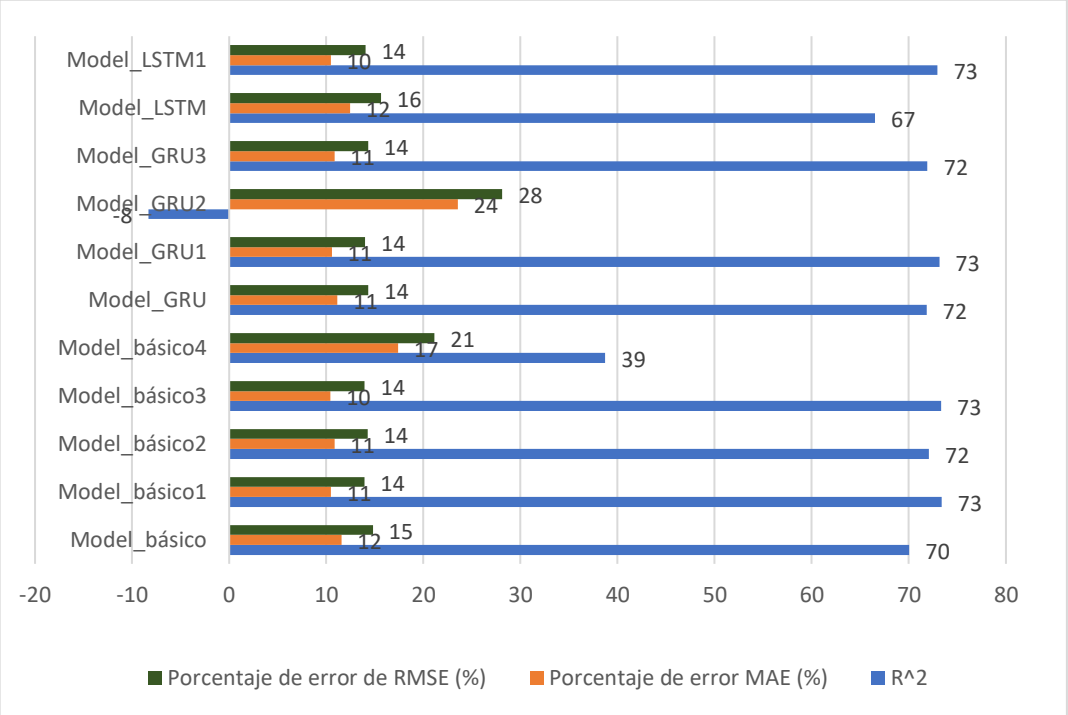
Nota: la gráfica muestra el rendimiento de los modelos entrenados, con las métricas de evaluación correspondientes al Error Absoluto Medio (MAE) y la Raíz del Error Cuadrático Medio (RMSE) en porcentaje considerando el rango de RUL real contenida en la base de datos.

La Figura 49 muestra el porcentaje de error en las predicciones, calculado a partir de las métricas de evaluación MAE y RMSE, en relación con el rango de valores del RUL establecido en la base de datos, que varía entre 6 y 195 ciclos. Esto implica que las predicciones realizadas por los modelos también se encuentran dentro de este rango de valores.

Según el MAE, las predicciones presentan un margen de error que oscila entre ±10 % y ±24 %, mientras que, de acuerdo con el RMSE, este margen varía entre ±14 % y ±28 %. En términos de desempeño, el mejor modelo es el Model_básico3 cuando se considera el MAE, y el Model_básico1 sobresale al evaluarse mediante el RMSE. En contraste, el Model_GRU2 se identifica como el modelo de menor rendimiento, al registrar el mayor porcentaje de error en ambas métricas. Este análisis evidencia que, aunque los modelos logran predecir el RUL con un nivel aceptable de precisión, existen diferencias en el comportamiento dependiendo de la métrica empleada para evaluar su desempeño

Figura 50

Interpretación del R^2 con respecto al porcentaje de error de MAE y RMSE



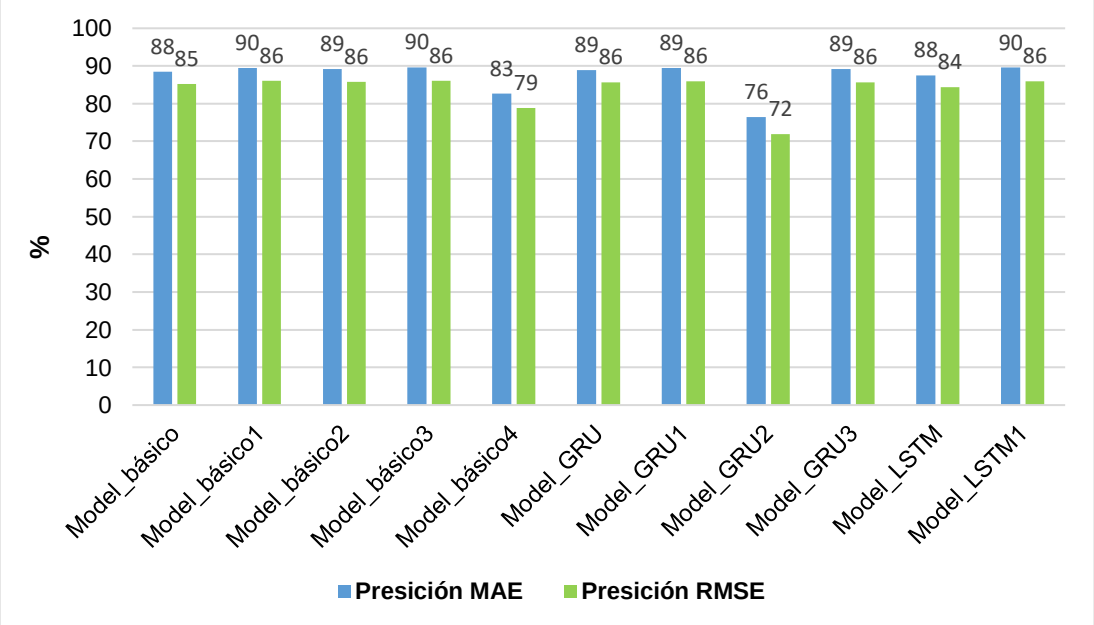
Nota: La gráfica muestra el coeficiente de determinación de todos los modelos entrenados y el porcentaje de error de MAE y RMSE.

En la Figura 50 se realiza la interpretación del coeficiente de determinación con respecto al porcentaje de error de MAE y RMSE expresados en porcentaje, en base a ello se visualiza que el modelo que mejor eficacia presenta es el Modelo_básico1 y Model_básico3, entre tanto, el Model_GRU2 evidencia un desempeño pobre con errores notables y explicabilidad nula, lo cual sugiere que los regularizadores que previenen sobreajuste y subajuste respectivamente son ineficaces en arquitecturas GRU con la base de datos empleada.

Al analizar el rendimiento de los modelos en base al porcentaje de error, se observa algunas discrepancias entre las métricas de evaluación, según la Raíz del Error Cuadrático Medio y el coeficiente de determinación el modelo mejor puntuado es el Model_básico1, no obstante, el Error Absoluto Medio indica que el mejor modelo es el Model_básico3, a pesar de estas discrepancias, las variaciones para elegir la mejor modelo varía ligeramente en poco más de un punto porcentual, por lo tanto, si se considera elegir un modelo para predicción debe ser seleccionado en base al RMSE que ya penaliza los errores grandes cometidos por el modelo en la predicción.

Figura 51

Análisis del rendimiento de los modelos en base a la precisión de MAE y RMSE



Nota: La gráfica muestra el coeficiente de determinación de todos los modelos entrenados y el porcentaje de error de MAE y RMSE.

4.3. Discusión de los resultados obtenidos

Los resultados obtenidos muestran un avance importante en cuanto al empleo de la Redes Neuronales Artificiales y su implementación en modelos predictivos en la industria aeroespacial, además, establece lineamientos necesarios para futuras investigaciones que opten por aplicar esta rama de la inteligencia artificial, en la presente línea de investigación se ensaya las arquitecturas más favorables en modelos de predicción de acuerdo con la evidencia actual.

Si bien es cierto las arquitecturas simples o también conocidas como densas en el ámbito de la programación, muestran ser razonablemente efectivas de acuerdo con los modelos entrenados, sin embargo, estas tienen cierto limite a partir del cual no logran mejorar al introducir a los modelos mayor complejidad. Por otra parte, las arquitecturas de tipo GRU se comportan de manera similar a las de tipo densas en cuanto a las métricas de evaluación, no obstante, hay que destacar que estas logran un rendimiento superior con un modelo de menor complejidad en comparación con las arquitecturas de tipo densas.

Las arquitecturas de tipo LSTM resultaron ser las más efectivas de acuerdo con los modelos entrenados, a pesar de que las métricas de evaluación indican un comportamiento similar a las demás arquitecturas (Densas y GRU), hay que destacar que, los modelos fueron de baja complejidad en comparación con las demás arquitecturas ensayadas, esto debido al costo computacional y el tiempo requerido para el entrenamiento.

Los modelos de predicción de vida útil restante en calderas industriales con Suport Vector Machine (SVM) empleadas para detectar anomalías y predecir el RUL evidencia una precisión superior al 90%. Además, resultaron ser efectivas con un conjunto de datos pequeño (Widodo *et al*., 2025).

Otras investigaciones que han optado por modelos híbridos combinando Redes Neuronales Recurrentes (RNN) y Random forest, con el propósito de predecir la vida

remanente de baterías empleadas en vehículos eléctricos lograron una mejora superior al 30% en comparación con modelos tradicionales y ampliamente difundidos (Bhupathi, 2023).

Las arquitecturas de tipo LSTM se han empleado en combinación con Autoencoders con el propósito de predecir el RUL en baterías de litio, este enfoque permitió una reducción del error en un 36 %, además, esta composición permitió mejorar notablemente la incertidumbre los datos (Li *et al*., 2025).

La literatura disponible en la actualidad evidencia la implementación de la inteligencia artificial como una estrategia para predecir la vida útil restante (RUL), no obstante, en la mayoría de las investigaciones evidencian una disminución del error en comparación con métodos tradicionales basados en física y estadística.

La precisión alcanzada en la presente investigación con los modelos entrenados supera el 70 %, lo cual no es suficiente para aplicaciones críticas como la industria aeroespacial, puesto que los motores de propulsión turbofán son componentes altamente críticos en las aeronaves de uso civil y militar, sin embargo, se pudo identificar que un modelo hibrido que combine las redes neuronales de tipo LSTM y de tipo GRU podrían resultar altamente efectivas y quizá alcanzar la precisión requerida en esta industria.

La base de datos obtenida del portal de la NASA no favorecía en gran medida sobre todo en cuanto al uso del tiempo, dado que se especifica los ciclos, donde cada ciclo corresponde a un despegue y un aterrizaje, no obstante, no menciona cuanto tiempo estuvo en operación dicha máquina, lo cual dificulta crear un modelo preciso para la predicción del RUL.

4.4. Respuesta a la pregunta de investigación

La pregunta de investigación que guio el presente trabajo fue la siguiente:

¿Puede un modelo de red neuronal artificial predecir con un alto nivel de precisión la vida útil restante de turborreactores de doble flujo, empleando una base de datos del centro de excelencia de pronósticos AMES de la NASA?

Las métricas de evaluación obtenidas en los modelos demuestran que si es posible predecir la vida útil restante (RUL) de turborreactores de doble flujo, ya sea con arquitecturas densas, GRU o LSTM. No obstante, la precisión alcanzada por los modelos entrenados no satisface los requerimientos en aplicaciones como las planteadas para esta problemática, donde se buscaba predecir el RUL de motores de Propulsion para su uso en aviación civil o militar, que requieren precisiones que no superen los ±5 ciclos previo al fallo o a su vez un coeficiente de determinación que logre explicar la variabilidad de al menos el 95%.

Es importante señalar que, para responder a esta pregunta, se llevó a cabo un exhaustivo proceso de entrenamiento y evaluación de múltiples modelos de RNA, utilizando diversas arquitecturas y explorando variaciones en los parámetros de aprendizaje y los Hiperparámetros. A pesar de los esfuerzos realizados, los resultados evidencian la necesidad de optimizar aún más los modelos o explorar enfoques alternativos para cumplir con los estrictos requerimientos de esta aplicación.

La métrica de evaluación RMSE (Raíz del Error Cuadrático Medio) sería la ideal para la selección de un modelo de predicción de RUL (Remaining Useful Life) dado que está ya considera los errores grandes cometidos por el modelo en la predicción, además, en un caso práctico al indicar ±26 ciclos antes del fallo, se debe elegir siempre con el signo negativo, esto permitirá tener un margen de maniobrabilidad para entrar en mantenimiento y evitar fallos imprevistos derivados de la operación de los motores de propulsión.

Conclusiones y Recomendaciones

5.1. Conclusiones

Se seleccionó tres tipos de arquitectura de redes neuronales artificiales (RNA) como las de capas densas empleadas en diversas problemáticas, de tipo GRU y LSTM ideales para modelar series temporales en problemas de regresión, resultando ser más eficientes las de capas densas, además las LSTM mostraron su potencial al mostrar avances significativos con pequeñas variaciones en los parámetros e Hiperparámetros.

Se entrenó los modelos de redes neuronales artificiales (RNA), ajustando interactivamente en base a los resultados los parámetros e Hiperparámetros. La tasa de aprendizaje más efectiva fue del 0,001, dado que al aumentar los modelos empeoraban y al disminuir no se evidenciaba avances positivos en el aprendizaje, las funciones de activación no evidenciaron influir notablemente en la mejora o empeoramiento del modelo, además al aumentar el tamaño de lote el modelo tendía a empeorar las métricas de evaluación, finalmente los algoritmos de optimización no mostraron su influencia en el entrenamiento.

Se evaluó el rendimiento de los modelos mediante métricas cuantitativas empleando el conjunto de prueba a fin de medir su capacidad predictiva, el modelo más eficiente resulto ser el Model_básico 1 de acuerdo con el RMSE (RAÍZ del Error Cuadrático Medio) y coeficiente de determinación (R^2), sin embargo, este no fue suficiente para aplicaciones criticas como la aviación civil y militar donde se requiere altísimas precisiones por motivos especialmente de seguridad.

5.2. Recomendaciones

Se recomienda añadir a la base de datos una columna con los valores de tiempo en otras unidades de medida diferente a ciclos, dado que para modelar series temporales se requiere el tiempo en horas. Además, incluir a cuantas horas de vuelo equivale un

ciclo, dado que se considera un ciclo únicamente a un aterrizaje y un despegue en conjunto.

Seleccionar arquitecturas de redes neuronales (RNA), de tipo hibridas, es decir, arquitecturas que combinen capas densas, GRU y LSTM, a fin de experimentar su funcionamiento con este tipo de combinaciones, dado que en la presente investigación se consideró únicamente arquitecturas densas con GRU y densas con LSTM, debido a la limitación de los recursos de software.

Entrenar los modelos de redes neuronales artificiales (RNA) empelando un software más sofisticado que reduzca notablemente el tiempo de entrenamiento de los modelos.

Se recomienda añadir más datos al conjunto de prueba de tal manera que se pueda diseñar modelos de redes neuronales artificiales (RNA) más robustas, esto permitirá evaluar los modelos en escenarios diversos de funcionamiento.

REFERENCIAS

Al-Sabaeei, A. M., Alhussian, H., Abdulkadir, S. J., & Jagadeesh, A. (2023). Prediction of oil and gas pipeline failures through machine learning approaches: A systematic review. *Energy Reports,10,* 1418–1422. https://doi.org/10.1016/j.egyr.2023.08.009

Amuzuvi, C. K., & Warden, H. (2020). Failure prediction of induction motors: A case study using CSLGH900/6-214, 5.8 MW, 11 kV/3ph/50 Hz sag mill motor at goldfields, Damang Mine. *Global Journal of Researches in Engineering*, 20(2), 45–53. https://doi.org/10.34257/gjrefvol20is2pg7

Bhupathi, H. (2023). Enhancing the battery life of electric vehicles through machine learning. *Social Science Journal.*

Bosch, A., Casas, J., & Lozano, T. (2020). *Deep learning: Principios y fundamentos.* Editorial UOC.

Caicedo, E., & López, J. (2020). *Una aproximación práctica a las redes neuronales artificiales.* Universidad Valle.

Carpio, K. P., & Oñate-Valdivieso, F. (2020). Redes neuronales artificiales aplicadas en sistemas de predicción para la seguridad vial. *Avances Investigación En Ingeniería*, *17*(2), 67–75. https://doi.org/10.18041/1794-4953/avances.2.6632

Carroll, J., Koukoura, S., McDonald, A., Charalambous, A., Weiss, S., & McArthur, S. (2019). Wind turbine gearbox failure and remaining useful life prediction using machine learning techniques. *Wind Energy*, *22*(3), 373–388. https://doi.org/10.1002/we.2290

Contreras Urgiles, W. R., Maldonado Ortega, J., & León Japa, R. (2018). Aplicación de una red neuronal feed-forward backpropagation para el diagnóstico de fallas mecánicas en motores de encendido provocado. *Ingenius*, *21*, 32–40. https://doi.org/10.17163/ings.n21.2019.03

Cuesta, J., Leturiondo, U., Vidal, Y., & Pozo, F. (2025). A review of prognostics and health management techniques in wind energy. *Reliability Engineering & System Safety*, Article 111004. https://doi.org/10.1016/j.ress.2025.111004

Fan, J., Zhao, L., & Li, M. (2025). Residual life prediction of rolling bearings driven by digital twins. *Symmetry*, *17*(3), 406. https://doi.org/10.3390/sym17030406

Incio-Flores, F. A., Capuñay-Sánchez, D. L., & Estela-Urbina, R. O. (2022). Modelo de red neuronal artificial para predecir resultados académicos en la asignatura matemática II. *Revista Electrónica Educare*, *27*(1). https://doi.org/10.15359/ree.27-1.14516

Kaiser, S., Schmitz, O., Ziegler, P., & Klingels, H. (2022). The water-enhanced turbofan as enabler for climate-neutral aviation. *Applied Sciences*, *12*(23). https://doi.org/10.3390/app122312431

Kumar, S. D. V., Karuppanan, S., & Ovinis, M. (2022). Artificial neural network-based failure pressure prediction of API 5L X80 pipeline with circumferentially aligned Interacting corrosion defects subjected to combined loadings. *Materials*, *15*(6). https://doi.org/10.3390/ma15062259

Latam Airlines. (2024). *Dentro de un centro de mantenimiento de aeronaves*.

Lemache, K., García, F., Valverde, V., & Velastegui, E. (2023). The machine learning approach to industrial maintenance management. *Revista Científica de La Universidad de Cienfuegos*, *15(10),* 2218–3620.

Li, W., Li, Y., Wang, N., Garg, A., Gao, L., Bose, B., & Shankhwar, K. (2025). A novel method based on hybridization of generative adversarial imputation nets and sdae-kriging for RUL prediction of lithium-ion battery in scenarios of missing and incomplete data. *IEEE Transactions on Industry Applications*, *12*(2), 1–12. https://doi.org/10.1109/TIA.2025.3549408

MA, Y., DU, X., & SUN, X. (2022). Adaptive modification of turbofan engine nonlinear model based on LSTM neural networks and hybrid optimization method. *Chinese Journal of Aeronautics*, *35*(9). https://doi.org/10.1016/j.cja.2021.11.005

Montaño, J. J. (2002). Redes neuronales artificiales aplicadas al análisis de datos. *Network*.

Nicolás Larenas. (2018, May 28). *MRO DE AVIANCA EN RIONEGRO.*

Perdigón-Llanes, R., & González-Benítez, N. (2022). Redes neuronales artificiales en el pronóstico de la producción de leche bovina. *Revista Colombiana de Computación*, *23*(1). https://doi.org/10.29375/25392115.4209

Rafezi, H., & Hassani, F. (2023). Drill bit wear monitoring and failure prediction for mining automation. *International Journal of Mining Science and Technology*, *33*(3). https://doi.org/10.1016/j.ijmst.2022.10.006

Restrepo, D., Viloria, J., & Roble, C. (2021). *El camino de las redes neuronales artificiales* (1st ed.). Unimagdalena.

Sampaio, G. S., Filho, A. R. de A. V., da Silva, L. S., & da Silva, L. A. (2019). Prediction of motor failure time using an artificial neural network. *Sensors*, *19*(19). https://doi.org/10.3390/s19194342

Schneller, W., Leitner, M., Maier, B., Grün, F., Jantschner, O., Leuders, S., & Pfeifer, T. (2022). Artificial intelligence assisted fatigue failure prediction. *International Journal of Fatigue*, *155*, Article*106580*. https://doi.org/10.1016/j.ijfatigue.2021.106580

Shaheen, B., Kocsis, Á., & Németh, I. (2023). Data-driven failure prediction and RUL estimation of mechanical components using accumulative artificial neural networks. *Engineering Applications of Artificial Intelligence*, *119*, Article 105749. https://doi.org/10.1016/j.engappai.2022.105749

Wan, L., Ullah, Z., Yang, D., & Falzon, B. G. (2023). Probability embedded failure prediction of unidirectional composites under biaxial loadings combining machine learning and micromechanical modelling. *Composite Structures*, *312,* Article 116837. https://doi.org/10.1016/j.compstruct.2023.116837

Widodo, A., Prahasto, T., Soleh, M., & Nugraha, H. (2025). Diagnostics and prognostics of boilers in power plant based on data-driven and machine learning. *International Journal of Prognostics and Health Management*, *16*(1). https://doi.org/10.36001/ijphm.2025.v16i1.4222

ANEXOS

Anexo A: Certificado de aprobación del abstract por parte del centro de idioma

UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI FOREIGN AND NATIVE LANGUAGES CENTER

ABSTRACT- EVALUATION SHEET

NAME: Rómulo Alexis Rivera Auqui
DATE: Jueves, 4 de septiembre de 2025.
Topic: "Predicción de la vida útil restante de turborreactores de doble flujo basado en las redes neuronales artificiales"

MARKS AWARDED **QUANTITATIVE AND QUALITATIVE**

VOCABULARY AND WORD USE	Use new learnt vocabulary and precise words related to the topic	Use a little new vocabulary and some appropriate words related to the topic	Use basic vocabulary and simplistic words related to the topic	Limited vocabulary and inadequate words related to the topic
	EXCELLENT: 2 ☐	GOOD: 1,5 ■	AVERAGE: 1 ☐	LIMITED: 0,5 ☐
WRITING COHESION	Clear and logical progression of ideas and supporting paragraphs.	Adequate progression of ideas and supporting paragraphs.	Some progression of ideas and supporting paragraphs.	Inadequate ideas and supporting paragraphs.
	EXCELLENT: 2 ■	GOOD: 1,5 ☐	AVERAGE: 1 ☐	LIMITED: 0,5 ☐
ARGUMENT	The message has been communicated very well and identify the type of text	The message has been communicated appropriately and identify the type of text	Some of the message has been communicated and the type of text is little confusing	The message hasn't been communicated and the type of text is inadequate
	EXCELLENT: 2 ■	GOOD: 1,5 ☐	AVERAGE: 1 ☐	LIMITED: 0,5 ☐
CREATIVITY	Outstanding flow of ideas and events	Good flow of ideas and events	Average flow of ideas and events	Poor flow of ideas and events
	EXCELLENT: 2 ☐	GOOD: 1,5 ■	AVERAGE: 1 ☐	LIMITED: 0,5 ☐
SCIENTIFIC SUSTAINABILITY	Reasonable, specific and supportable opinion or thesis statement	Minor errors when supporting the thesis statement	Some errors when supporting the thesis statement	Lots of errors when supporting the thesis statement
	EXCELLENT: 2 ■	GOOD: 1,5 ☐	AVERAGE: 1 ☐	LIMITED: 0,5 ☐
TOTAL/AVERAGE	9 - 10: EXCELLENT 7 - 8,9: GOOD 5 - 6,9: AVERAGE 0 - 4,9: LIMITED	**TOTAL 9**		

UNIVERSIDAD POLITÉCNICA ESTATAL DEL CARCHI- FOREIGN AND NATIVE LANGUAGES CENTER

Informe sobre el Abstract de Artículo Científico o Investigación.

Autor: Rómulo Alexis Rivera Auqui
Fecha de recepción del abstract: Viernes, 29 agosto de 2025.
Fecha de entrega del informe: Jueves, 4 de septiembre de 2025.

El presente informe validará la traducción del idioma español al inglés si alcanza un porcentaje de: 9 – 10 Excelente.

Si la traducción no está dentro de los parámetros de 9 – 10, el autor deberá realizar las observaciones presentadas en el ABSTRACT, para su posterior presentación y aprobación.

Observaciones:

Después de realizar la revisión del presente abstract, éste presenta una apropiada traducción sobre el tema planteado en el idioma Inglés. Según la rúbrica de evaluación de la traducción en Inglés, ésta alcanza un valor de 9; por lo cual se valida dicho trabajo.

Atentamente

MARTHA ARACELLY VIVEROS ALMEIDA

MA. Martha Viveros
Docente responsable del
CIDEN

Anexo B. Estadísticas descriptivas de los archivos de la base de datos

- train_FD001

	count	mean	std	min	25%	50%	75%	max
MachineID	20631	51.5	29.2	1.00	26.0	52.0	77.0	100
Cycle	20631	109	68.9	1.00	52.0	104	156	362
Setting 1	20631	< 0.001	0.002	< 0.001	< 0.001	< 0.001	0.002	0.009
Setting 2	20631	< 0.001	0.0003	< 0.001	< 0.001	< 0.001	0.0003	0.0006
Setting 3	20631	100	< 0.001	100	100	100	100	100
Sensor 1	20631	519	< 0.001	519	519	519	519	519
Sensor 2	20631	643	0.50	641	642	643	643	645
Sensor 3	20631	1591	6.13	1571	1586	1590	1594	1617
Sensor 4	20631	1409	9.00	1382	1402	1408	1415	1441
Sensor 5	20631	14.6	< 0.001	14.6	14.6	14.6	14.6	14.6
Sensor 6	20631	21.6	0.001	21.6	21.6	21.6	21.6	21.6
Sensor 7	20631	553	0.89	550	553	553	554	556
Sensor 8	20631	2388	0.071	2388	2388	2388	2388	2389
Sensor 9	20631	9065	22.1	9022	9053	9061	9069	9245
Sensor 10	20631	1.30	< 0.001	1.30	1.30	1.30	1.30	1.30
Sensor 11	20631	47.5	0.27	46.9	47.4	47.5	47.7	48.5
Sensor 12	20631	521	0.74	519	521	521	522	523
Sensor 13	20631	2388	0.072	2388	2388	2388	2388	2389
Sensor 14	20631	8144	19.1	8100	8133	8141	8148	8294
	count	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**

Sensor 15	20631	8.44	0.038	8.32	8.41	8.44	8.47	8.58
Sensor 16	20631	0.030	< 0.001	0.030	0.030	0.030	0.030	0.030
Sensor 17	20631	393	1.55	388	392	393	394	400
Sensor 18	20631	2388	< 0.001	2388	2388	2388	2388	2388
Sensor 19	20631	100	< 0.001	100	100	100	100	100
Sensor 20	20631	38.8	0.18	38.1	38.7	38.8	39.0	39.4
Sensor 21	20631	23.3	0.11	22.9	23.2	23.3	23.4	23.6

- train_FD002

	count	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**
MachineID	53759	131	74.5	1.00	68.0	131	195	260
Cycle	53759	109	69.2	1.00	52.0	104	157	378
Setting 1	53759	24.0	14.7	< 0.001	10.0	25.0	42.0	42.0
Setting 2	53759	0.57	0.31	< 0.001	0.25	0.70	0.84	0.84
Setting 3	53759	94.0	14.2	60.0	100	100	100	100
Sensor 1	53759	473	26.4	445	445	463	491	519
Sensor 2	53759	580	37.3	536	550	556	607	645
Sensor 3	53759	1420	106	1244	1353	1369	1499	1613
Sensor 4	53759	1205	119	1024	1124	1139	1307	1439
Sensor 5	53759	8.03	3.61	3.91	3.91	7.05	10.5	14.6
Sensor 6	53759	11.6	5.43	5.71	5.72	9.03	15.5	21.6
Sensor 7	53759	283	146	137	140	195	394	556
Sensor 8	53759	2229	145	1915	2212	2223	2324	2388
	count	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**

Sensor 9	53759	8525	336	7986	8322	8361	8778	9216
Sensor 10	53759	1.09	0.13	0.93	1.02	1.02	1.26	1.30
Sensor 11	53759	43.0	3.23	36.2	41.9	42.4	45.4	48.5
Sensor 12	53759	266	138	129	132	183	371	523
Sensor 13	53759	2335	128	2028	2388	2388	2388	2390
Sensor 14	53759	8067	84.8	7848	8062	8083	8127	8268
Sensor 15	53759	9.33	0.75	8.34	8.68	9.31	9.39	11.1
Sensor 16	53759	0.023	0.005	0.020	0.020	0.020	0.030	0.030
Sensor 17	53759	348	27.8	303	331	335	369	399
Sensor 18	53759	2229	145	1915	2212	2223	2324	2388
Sensor 19	53759	97.8	5.36	84.9	100	100	100	100
Sensor 20	53759	20.8	9.87	10.2	10.9	14.9	28.5	39.3
Sensor 21	53759	12.5	5.92	6.01	6.55	8.93	17.1	23.6

- train_FD003

	count	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**
MachineID	24720	48.6	29.3	1.00	23.0	47.0	74.0	100
Cycle	24720	139	98.8	1.00	62.0	124	191	525
Setting 1	24720	< 0.001	0.002	< 0.001	< 0.001	< 0.001	0.002	0.009
Setting 2	24720	< 0.001	0.0003	< 0.001	< 0.001	< 0.001	0.0003	0.0007
Setting 3	24720	100	< 0.001	100	100	100	100	100
Sensor 1	24720	519	< 0.001	519	519	519	519	519
Sensor 2	24720	642	0.52	641	642	642	643	645
	count	**mean**	**std**	**min**	**25%**	**50%**	**75%**	**max**

Sensor 3	24720	1588	6.81	1564	1583	1588	1592	1615
Sensor 4	24720	1404	9.77	1377	1397	1403	1411	1441
Sensor 5	24720	14.6	< 0.001	14.6	14.6	14.6	14.6	14.6
Sensor 6	24720	21.6	0.018	21.4	21.6	21.6	21.6	21.6
Sensor 7	24720	555	3.44	550	553	554	556	570
Sensor 8	24720	2388	0.16	2387	2388	2388	2388	2389
Sensor 9	24720	9064	20.0	9018	9052	9060	9070	9234
Sensor 10	24720	1.30	0.003	1.29	1.30	1.30	1.30	1.32
Sensor 11	24720	47.4	0.30	46.7	47.2	47.4	47.6	48.4
Sensor 12	24720	523	3.26	518	521	522	524	537
Sensor 13	24720	2388	0.16	2387	2388	2388	2388	2389
Sensor 14	24720	8144	16.5	8100	8135	8141	8149	8291
Sensor 15	24720	8.40	0.061	8.16	8.36	8.40	8.44	8.57
Sensor 16	24720	0.030	< 0.001	0.030	0.030	0.030	0.030	0.030
Sensor 17	24720	393	1.76	388	391	392	394	399
Sensor 18	24720	2388	< 0.001	2388	2388	2388	2388	2388
Sensor 19	24720	100	< 0.001	100	100	100	100	100
Sensor 20	24720	39.0	0.25	38.2	38.8	39.0	39.1	39.9
Sensor 21	24720	23.4	0.15	22.9	23.3	23.4	23.5	24.0

- train_FD004

	count	mean	std	min	25%	50%	75%	max
MachineID	61249	124	72.0	1.00	60.0	126	185	249
Cycle	61249	134	89.8	1.00	62.0	123	191	543
Setting 1	61249	24.0	14.8	< 0.001	10.0	25.0	42.0	42.0
Setting 2	61249	0.57	0.31	< 0.001	0.25	0.70	0.84	0.84
Setting 3	61249	94.0	14.3	60.0	100	100	100	100
Sensor 1	61249	473	26.4	445	445	463	491	519
Sensor 2	61249	579	37.3	535	549	556	607	644
Sensor 3	61249	1418	106	1243	1351	1368	1497	1613
Sensor 4	61249	1202	119	1024	1119	1137	1303	1441
Sensor 5	61249	8.03	3.62	3.91	3.91	7.05	10.5	14.6
Sensor 6	61249	11.6	5.44	5.67	5.72	9.03	15.5	21.6
Sensor 7	61249	283	147	136	143	195	394	571
Sensor 8	61249	2229	145	1915	2212	2223	2324	2389
Sensor 9	61249	8525	337	7985	8321	8363	8777	9197
Sensor 10	61249	1.10	0.13	0.G93	1.02	1.03	1.26	1.32
Sensor 11	61249	42.9	3.24	36.0	41.8	42.3	45.2	48.4
Sensor 12	61249	267	138	128	135	183	371	537
Sensor 13	61249	2334	128	2028	2388	2388	2388	2390
Sensor 14	61249	8068	85.7	7846	8063	8084	8128	8262
Sensor 15	61249	9.29	0.75	8.18	8.65	9.26	9.37	11.1
Sensor 16	61249	0.023	0.005	0.020	0.020	0.020	0.030	0.030
Sensor 17	61249	348	27.8	302	330	334	368	399

	count	mean	std	min	25%	50%	75%	max
Sensor 18	61249	2229	145	1915	2212	2223	2324	2388
Sensor 19	61249	97.8	5.37	84.9	100	100	100	100
Sensor 20	61249	20.9	9.94	10.2	10.9	14.9	28.6	39.9
Sensor 21	61249	12.5	5.96	6.08	6.57	8.96	17.1	23.9

- test_FD001

	count	mean	std	min	25%	50%	75%	max
MachineID	13096	51.5	28.3	1.00	28.0	52.0	76.0	100
Cycle	13096	76.8	53.1	1.00	33.0	69.0	113	303
Setting 1	13096	< 0.001	0.002	< 0.001	< 0.001	< 0.001	0.002	0.008
Setting 2	13096	< 0.001	0.0003	< 0.001	< 0.001	< 0.001	0.0003	0.0007
Setting 3	13096	100	< 0.001	100	100	100	100	100
Sensor 1	13096	519	< 0.001	519	519	519	519	519
Sensor 2	13096	642	0.40	641	642	642	643	644
Sensor 3	13096	1588	5.00	1569	1585	1588	1591	1608
Sensor 4	13096	1405	6.69	1384	1400	1404	1409	1433
Sensor 5	13096	14.6	< 0.001	14.6	14.6	14.6	14.6	14.6
Sensor 6	13096	21.6	0.002	21.6	21.6	21.6	21.6	21.6
Sensor 7	13096	554	0.68	551	553	554	554	556
Sensor 8	13096	2388	0.057	2388	2388	2388	2388	2388
Sensor 9	13096	9058	11.4	9025	9051	9057	9064	9155
Sensor 10	13096	1.30	< 0.001	1.30	1.30	1.30	1.30	1.30
Sensor 11	13096	47.4	0.20	46.8	47.3	47.4	47.5	48.3

	count	mean	std	min	25%	50%	75%	max
Sensor 12	13096	522	0.56	519	521	522	522	524
Sensor 13	13096	2388	0.057	2388	2388	2388	2388	2388
Sensor 14	13096	8139	10.2	8108	8132	8138	8144	8220
Sensor 15	13096	8.43	0.029	8.33	8.41	8.42	8.44	8.54
Sensor 16	13096	0.030	< 0.001	0.030	0.030	0.030	0.030	0.030
Sensor 17	13096	393	1.23	389	392	393	393	397
Sensor 18	13096	2388	< 0.001	2388	2388	2388	2388	2388
Sensor 19	13096	100	< 0.001	100	100	100	100	100
Sensor 20	13096	38.9	0.14	38.3	38.8	38.9	39.0	39.4
Sensor 21	13096	23.3	0.084	22.9	23.3	23.3	23.4	23.6

- test_FD002

	count	mean	std	min	25%	50%	75%	max
MachineID	33991	128	74.1	1.00	65.0	128	189	259
Cycle	33991	81.2	58.9	1.00	34.0	70.0	119	367
Setting 1	33991	24.0	14.8	< 0.001	10.0	25.0	35.0	42.0
Setting 2	33991	0.57	0.31	< 0.001	0.25	0.70	0.84	0.84
Setting 3	33991	94.1	14.2	60.0	100	100	100	100
Sensor 1	33991	473	26.5	445	449	463	491	519
Sensor 2	33991	580	37.3	535	549	556	607	644
Sensor 3	33991	1419	106	1245	1352	1367	1498	1610
Sensor 4	33991	1203	119	1030	1123	1134	1305	1434
Sensor 5	33991	8.04	3.62	3.91	5.48	7.05	10.5	14.6

	count	mean	std	min	25%	50%	75%	max
Sensor 6	33991	11.6	5.45	5.71	8.00	9.03	15.5	21.6
Sensor 7	33991	283	146	137	174	195	394	556
Sensor 8	33991	2229	145	1915	2212	2223	2324	2388
Sensor 9	33991	8522	336	7988	8320	8356	8776	9180
Sensor 10	33991	1.10	0.13	0.93	1.02	1.02	1.26	1.30
Sensor 11	33991	42.9	3.23	36.2	41.9	42.2	45.3	48.3
Sensor 12	33991	266	138	129	164	183	371	523
Sensor 13	33991	2335	128	2028	2388	2388	2388	2390
Sensor 14	33991	8064	83.8	7851	8061	8080	8126	8243
Sensor 15	33991	9.32	0.75	8.33	8.66	9.30	9.37	11.0
Sensor 16	33991	0.023	0.005	0.020	0.020	0.020	0.030	0.030
Sensor 17	33991	348	27.8	303	330	334	369	398
Sensor 18	33991	2229	145	1915	2212	2223	2324	2388
Sensor 19	33991	97.8	5.36	84.9	100	100	100	100
Sensor 20	33991	20.8	9.91	10.2	14.0	14.9	28.5	39.3
Sensor 21	33991	12.5	5.94	6.12	8.41	8.94	17.1	23.6

- test_FD003

	count	mean	std	min	25%	50%	75%	max
MachineID	16596	51.9	29.9	1.00	24.0	52.0	78.0	100
Cycle	16596	106	83.3	1.00	42.0	87.0	149	475
Setting 1	16596	< 0.001	0.002	< 0.001	< 0.001	< 0.001	0.002	0.008
Setting 2	16596	< 0.001	0.0003	< 0.001	< 0.001	< 0.001	0.0003	0.0006

	count	mean	std	min	25%	50%	75%	max
Setting 3	16596	100	< 0.001	100	100	100	100	100
Sensor 1	16596	519	< 0.001	519	519	519	519	519
Sensor 2	16596	642	0.42	641	642	642	643	644
Sensor 3	16596	1586	5.54	1565	1582	1585	1589	1607
Sensor 4	16596	1401	7.15	1379	1395	1400	1405	1430
Sensor 5	16596	14.6	< 0.001	14.6	14.6	14.6	14.6	14.6
Sensor 6	16596	21.6	0.016	21.6	21.6	21.6	21.6	21.6
Sensor 7	16596	555	2.24	550	553	554	555	568
Sensor 8	16596	2388	0.15	2387	2388	2388	2388	2388
Sensor 9	16596	9056	12.0	9023	9049	9055	9062	9119
Sensor 10	16596	1.30	0.002	1.29	1.30	1.30	1.30	1.31
Sensor 11	16596	47.3	0.22	46.6	47.1	47.3	47.4	48.2
Sensor 12	16596	522	2.11	518	521	522	523	536
Sensor 13	16596	2388	0.15	2387	2388	2388	2388	2389
Sensor 14	16596	8138	9.97	8105	8132	8138	8143	8191
Sensor 15	16596	8.40	0.041	8.20	8.37	8.40	8.42	8.53
Sensor 16	16596	0.030	< 0.001	0.030	0.030	0.030	0.030	0.030
Sensor 17	16596	392	1.39	387	391	392	393	398
Sensor 18	16596	2388	< 0.001	2388	2388	2388	2388	2388
Sensor 19	16596	100	< 0.001	100	100	100	100	100
Sensor 20	16596	39.0	0.17	38.3	38.9	39.0	39.1	39.8
Sensor 21	16596	23.4	0.11	23.0	23.3	23.4	23.5	23.8

- test_FD004

	count	mean	std	min	25%	50%	75%	max
MachineID	41214	120	71.7	1.00	57.0	118	181	248
Cycle	41214	109	83.7	1.00	43.0	91.0	155	486
Setting 1	41214	24.1	14.7	< 0.001	10.0	25.0	42.0	42.0
Setting 2	41214	0.57	0.31	< 0.001	0.25	0.70	0.84	0.84
Setting 3	41214	94.1	14.2	60.0	100	100	100	100
Sensor 1	41214	473	26.3	445	445	463	491	519
Sensor 2	41214	579	37.1	535	549	556	607	644
Sensor 3	41214	1415	105	1241	1349	1365	1495	1608
Sensor 4	41214	1197	118	1025	1117	1130	1299	1429
Sensor 5	41214	8.00	3.60	3.91	3.91	7.05	10.5	14.6
Sensor 6	41214	11.5	5.41	5.66	5.72	9.02	15.5	21.6
Sensor 7	41214	282	146	136	141	195	394	569
Sensor 8	41214	2229	145	1915	2212	2223	2324	2389
Sensor 9	41214	8517	334	7987	8318	8354	8774	9131
Sensor 10	41214	1.09	0.13	0.93	1.02	1.03	1.26	1.32
Sensor 11	41214	42.7	3.21	36.0	41.7	42.1	45.1	48.2
Sensor 12	41214	265	137	128	133	183	371	537
Sensor 13	41214	2335	128	2028	2388	2388	2388	2389
Sensor 14	41214	8063	84.1	7850	8061	8080	8126	8197
Sensor 15	41214	9.28	0.74	8.20	8.64	9.26	9.35	11.0
Sensor 16	41214	0.023	0.005	0.020	0.020	0.020	0.030	0.030
Sensor 17	41214	347	27.6	302	330	334	368	397

	count	mean	std	min	25%	50%	75%	max
Sensor 18	41214	2229	145	1915	2212	2223	2324	2388
Sensor 19	41214	97.8	5.35	84.9	100	100	100	100
Sensor 20	41214	20.8	9.88	10.2	10.9	14.9	28.6	39.7
Sensor 21	41214	12.5	5.93	6.15	6.54	8.96	17.1	23.8

- RUL_FD001

	count	mean	std	min	25%	50%	75%	max
RUL	100	75.5	41.8	7.00	32.8	86.0	112	145

- RUL_FD002

	count	mean	std	min	25%	50%	75%	max
RUL	259	81.2	53.9	6.00	35.0	80.0	121	194

- RUL_FD003

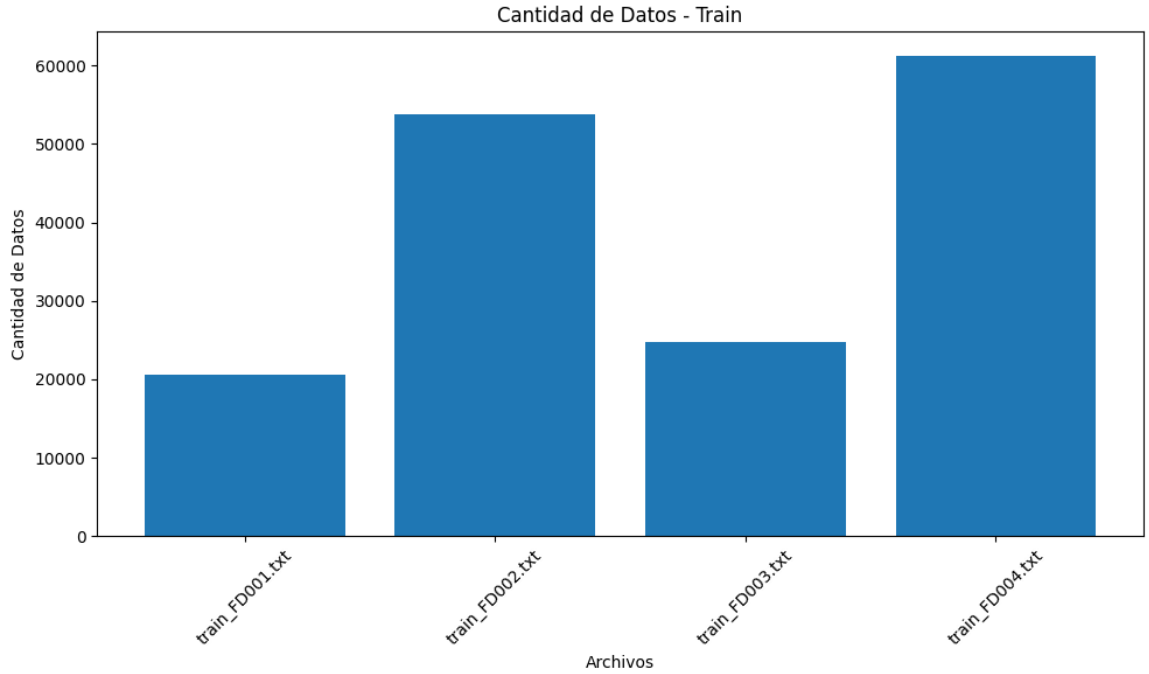
	count	mean	std	min	25%	50%	75%	max
RUL	259	81.2	53.9	6.00	35.0	80.0	121	194

- RUL_FD004

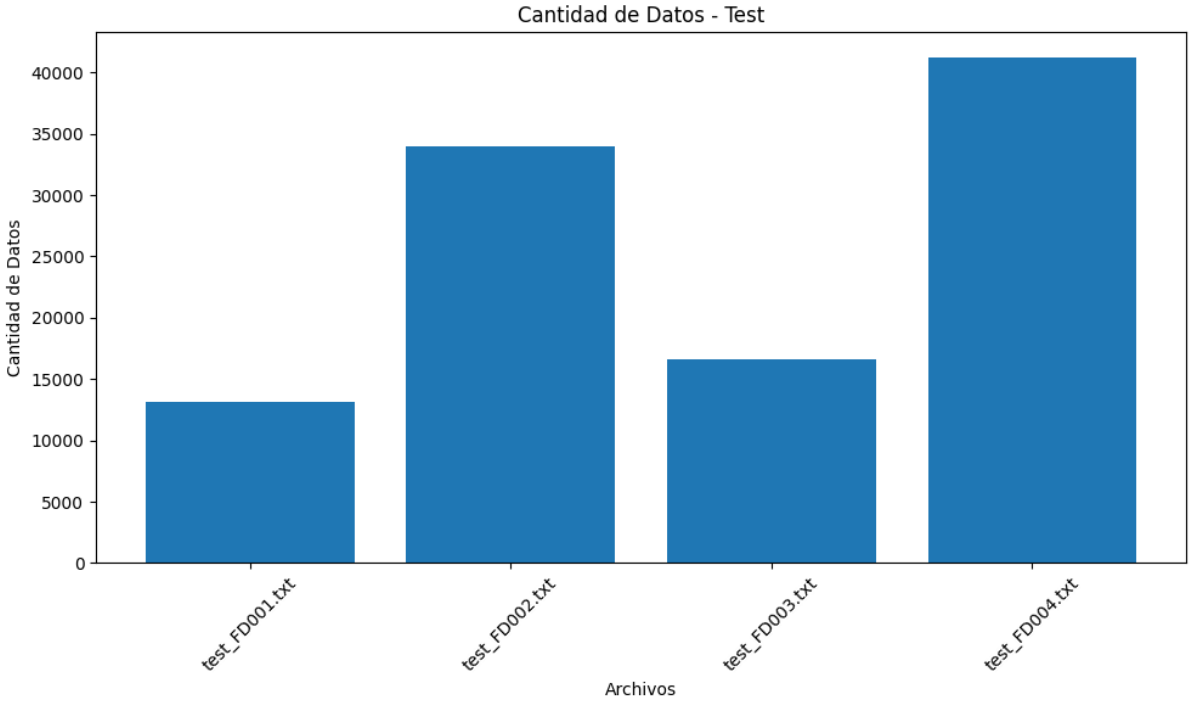
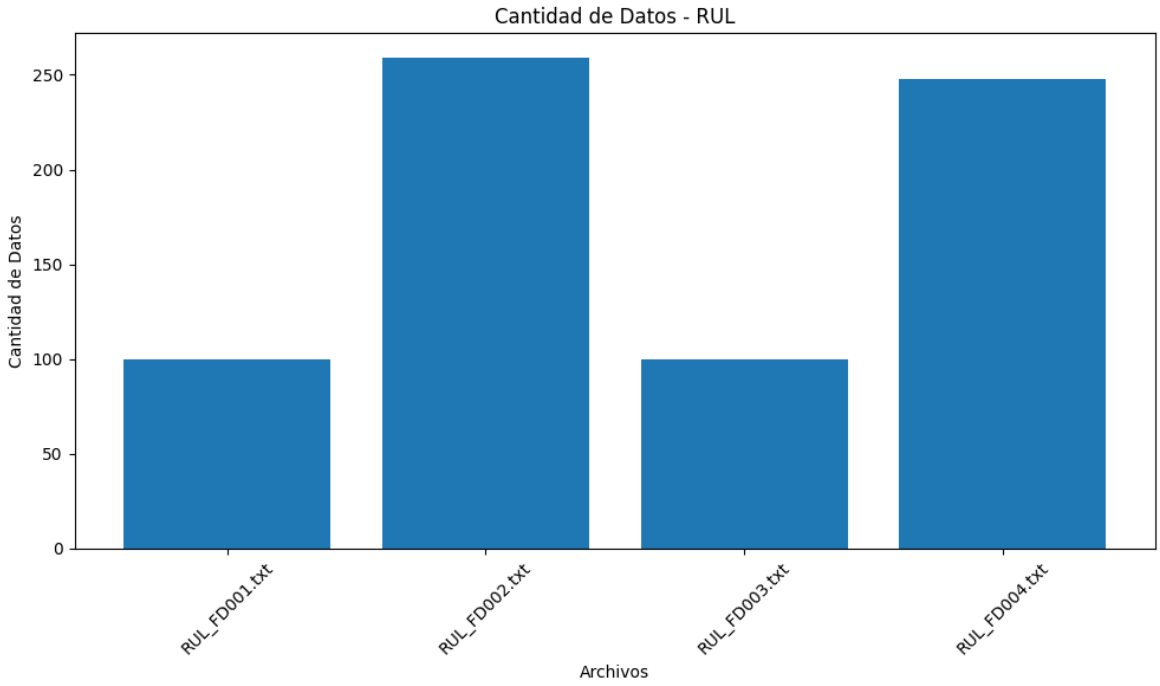
	count	mean	std	min	25%	50%	75%	max
RUL	100	75.3	41.6	6.00	43.2	77.5	115	145

Anexo C. Cantidad de datos por cada archivo

Archivos de entrenamiento



- Archivos de prueba



- Archivos RUL