

Diagnóstico de fallas para un sistema aeroespacial

Jeaustin Calderón-Quesada, Diana Cerdas-, Nicole Jiménez-Herrera, Alexander Solís-Quesada

2020027413 2020023718 2020035277 2020277449

Fecha de entrega: 6 de noviembre, 2024

Escuela de Ingeniería Mecatrónica

Instituto Tecnológico de Costa Rica

Resumen—Este artículo explora el pronóstico de la vida útil remanente (RUL) de motores aeronáuticos por medio de la técnica de Deep Learning llamada BGOA-EANNS. Se utilizaron datos del reto PHM08 para el entrenamiento del modelo computacional. Estos datos consisten en una serie de lecturas de sensores que antes de poder ser utilizados fueron analizados para verificar su aporte en la determinación de una falla en el motor. Posterior a la selección de los datos, se procede al entrenamiento del modelo computacional y la validación del mismo. Finalmente, se plantean algunas recomendaciones para aumentar la robustez del trabajo a futuro.

Palabras clave—Mantenimiento Predictivo, Vida Útil Remanente, Ingeniería Aeroespacial, Deep Learning, BGOA-EANNS.

I. INTRODUCCIÓN

El diagnóstico de fallas en sistemas aeroespaciales es una disciplina crítica que ha ido en aumento en las últimas décadas. Esta busca identificar, clasificar y predecir fallas tanto en componentes o en subsistemas aeroespaciales. Dado las aplicaciones de estos sistemas, las condiciones de operación son extremas y complejas, por tanto una falla no detectada o mal gestionada puede comprometer la seguridad no solo de la aeronave, sino de las personas. Por consiguiente, la detección temprana y precisa de fallas permite tomar acciones correctivas antes que el sistema se vea afectado, lo que a su vez optimiza la seguridad, la eficiencia operativa y los costos de mantenimiento.

Los sistemas de diagnóstico de fallas suelen consistir en múltiples lecturas de sensores, técnicas de monitorio en tiempo real y algoritmos de procesamiento de datos que detectan patrones de comportamiento y degradación de componentes. Tecnologías como *machine learning* e inteligencia artificial han cobrado relevancia en este campo, pues permiten desarrollar modelos predictivos que identifican anomalías o signos de falla con anticipación.

Del mismo modo, para este proyecto se aborda el uso de un modelo para predecir fallos de un motor en la base de datos del PHM08, donde se documenta el proceso de degradación de motores en turbinas de gas aeronáuticas. Esta es una oportunidad para el desarrollo de modelos predictivos que anticipen fallas en función de patrones identificados en los datos, brindando así un primer acercamiento a las técnicas avanzadas del mantenimiento predictivo, el cual es un área que cada vez se vuelve indispensable en sistemas

mecatrónicos.

El modelo seleccionado está basado en *Deep Learning* y consisten en dos etapas; la fase de selección de características por el Algoritmo binario de optimización de Grasshopper (BGOA por sus siglas en inglés, y la etapa de aprendizaje y predicción por Redes Neuronales Artificiales (ANNs por sus siglas en inglés) con método de conjunto de votación. Este modelo es llamado BGOA-EANNS y es propuesto por [1].

El objetivo de este trabajo es desarrollar un modelo predictivo basado en BGOA-EANNS que permita obtener resultados precisos y confiables en la predicción de fallas, utilizando la base de datos PHM08. No obstante, el proyecto enfrenta diversas limitaciones, como la falta de documentación sobre la vida útil remanente en el conjunto de datos, el uso de múltiples sensores que incrementa el volumen de datos a procesar, la presencia de ruido en las mediciones de los sensores y la insuficiencia de datos para una validación adecuada del modelo. Asimismo, está la complejidad de identificar características relevantes que pueden ser determinantes para obtener resultados óptimos.

II. DESCRIPCIÓN DEL SISTEMA SELECCIONADO

Para este proyecto se utiliza el conjunto de datos PHM08 (Prognostics and Health Management) de la NASA. Este reto se compone de una serie de datos simulados con C-MAPSS en Simulink, MATLAB [2]. Esta herramienta está diseñada para simular un motor turbofan comercial grande de manera realista. Además, incluye una serie de parámetros de entrada personalizables que permiten que el usuario ingrese valores específicos de su propia elección con respecto configuración operacional, controladores de lazo cerrado, condiciones ambientales, entre otras [2].

Ahora bien, estos datos simulados fueron diseñados para predecir la vida útil remanente (RUL) de una serie de motores. El sistema se compone de un set de datos de entrenamiento y uno de prueba. En set de datos de entrenamiento cada motor comienza en condiciones operativas normales, es decir, no presentan ningún modo de falla. No obstante, esto no significa que comienza en un estado nuevo, sino que inicia en algún punto desconocido de su vida útil. Luego, progresivamente experimenta un deterioro que eventualmente lo lleva a un punto de falla. Por su parte, en el set de datos de prueba,

se brindan datos antes de que ocurra la falla, y el objetivo es obtener la cantidad de ciclos restantes hasta la falla [3].

II-A. Variables presentes en el set de datos

En este caso, cada registro del set de datos representa un ciclo de operación de un motor, el cual se compone de 26 columnas [4], estas se identifican en el cuadro I.

Cuadro I: Variables de las columnas del set de datos. Datos tomados de [4]

Número de columna	Variable medida
Columna 1	Identificador del motor
Columna 2	Número de ciclo
Columna 3	Configuración de operación 1
Columna 4	Configuración de operación 2
Columna 5	Configuración de operación 3
Columnas restantes	Valores de 20 diferentes sensores

Es importante destacar que el set de datos utilizado para el entrenamiento y validación está compuesto por 218 motores con diferentes cantidades de ciclos de operación antes de la falla. Esto significa que no todos los motores inician en un mismo nivel de desgaste [4]. Además, es importante destacar que el set de datos utilizado no está etiquetados, es decir, no se indica directamente el valor que se desea predecir (RUL). Para hacer el etiquetado se considera que todos los motores terminan en falla, por ende el valor de RUL en el último ciclo debe ser de -1. Entonces, a partir de esto se utilizó la ecuación 1 para obtener los demás valores de vida útil remanente.

$$RUL = \text{ciclo máximo} - \text{ciclo actual} - 1 \quad (1)$$

II-B. Modos de falla en el sistema

Para identificar los modos de falla del sistema, se considera que los datos fueron generados por una simulación. Esto implica que para desarrollarlos se utilizó un modelo de degradación exponencial, el cual se muestra en la ecuaciones 2 y 3. Como se puede apreciar en estas ecuaciones, para el sistema en estudio los sensores son los encargados de detectar la pérdida de eficiencia y flujo en diversos componentes clave de los motores [2]. Entonces, en la generación de los datos, los modos de falla son simulados al ajustar parámetros específicos en la eficiencia y el flujo de cada componente, lo cual se traduce en patrones de deterioro en los datos de los sensores. De acuerdo con [5], conforme que el motor se degrada, las lecturas de los sensores muestran cambios, estos cambios se reflejan más significativamente en la temperatura y la presión.

$$e(t) = 1 - d_e - \exp \left\{ a_e(t) t^{b_e(t)} \right\} \quad (2)$$

$$f(t) = 1 - d_f - \exp \left\{ a_f(t) t^{b_f(t)} \right\} \quad (3)$$

donde,

$e(t)$ es la eficiencia

$f(t)$ es el flujo

d_e y d_f son los desgastes iniciales de eficiencia y flujo.

III. DESCRIPCIÓN DEL MODELO DE PRONÓSTICO SELECCIONADO

El modelo de pronóstico seleccionado [1] se compone de varias etapas clave que optimizan la predicción de la vida útil remanente (RUL) del componente. La primera etapa realiza una selección de componentes mediante un algoritmo de optimización, identificando aquellos factores relevantes para la modelación precisa de la RUL. Para ello, se emplea el Algoritmo de Optimización de Saltamontes Binario (BGOA), que permite reducir la dimensionalidad del problema al seleccionar las variables más informativas. Una vez definidos los componentes, se establecen seis modos de operación basados en los atributos que configuran la operación del sistema. Cada uno de estos modos se asigna a un sub-conjunto de datos con su propio sistema de predicción. Cada sistema de predicción está compuesto de un conjunto de tres redes neuronales artificiales (ANNs) con diferentes configuraciones de hiperparámetros, las cuales se integran mediante un método de votación para obtener el resultado final [6].

En el estudio de referencia [1], los investigadores emplean una predicción categórica de la RUL, clasificándola en categorías de urgencia, corta, media y larga. Entre las fortalezas de este modelo se destaca la selección de componentes, lo que aumenta su capacidad para aprender el comportamiento de los componentes realmente relevantes. Al contar con tres modelos de redes neuronales artificiales (ANNs), se compensan los pequeños errores de clasificación de alguno de los modelos. Además, al segmentar los datos según los regímenes de operación, se facilita un mejor aprendizaje del comportamiento de los datos, lo que contribuye a una mayor precisión y adaptabilidad del modelo. A continuación, se describen las etapas principales del modelo:

III-A. Selección de componentes

El BGOA, una adaptación del algoritmo inspirado en el comportamiento de enjambre de los saltamontes se utiliza para resolver problemas de optimización en un espacio de búsqueda binario, en el que cada variable es binaria (0 o 1). En este contexto, cada saltamonte representa una posible solución, expresada como un vector binario que indica la inclusión o exclusión de cada componente del conjunto de datos. Para simular el movimiento de los saltamontes en el espacio binario, el desplazamiento continuo se adapta a valores discretos mediante funciones de activación, como la función sigmoide.

Durante la iteración, los saltamontes ajustan su posición en respuesta a la influencia de sus vecinos, logrando un balance entre la exploración de nuevas combinaciones y la explotación de las soluciones ya encontradas. Este proceso iterativo permite que el enjambre converge en áreas que contienen soluciones óptimas o cercanas al óptimo, facilitando una búsqueda inicial amplia seguida de una concentración progresiva en las mejores áreas identificadas. Según los resultados obtenidos en el artículo original [1], los componentes seleccionados incluyen:

número de motor, tiempo en ciclos, configuración de operación 1, 2 y 3, así como 15 de las 21 señales de los sensores. Los sensores excluidos corresponden a los números 2, 5, 6, 7, 19 y 21.

III-B. Redes Neuronales Artificiales (ANNs)

Como se indicó, el modelo cuenta con tres ANNs entrenadas con los mismos datos, pero cada una con diferentes configuraciones de hiperparámetros. Una vez entrenadas, se aplican en conjunto mediante un esquema de votación para obtener una decisión final que integre los resultados de los tres modelos. En la tabla II se detallan los hiperparámetros específicos de cada ANN según los autores del artículo original [1]. Además, se implementó la técnica de detención temprana (early stopping) para optimizar el proceso de entrenamiento, permitiendo finalizarlo si no se observa mejora en las métricas de rendimiento después de 15 iteraciones.

Cuadro II: Descripción de los hiperparámetros de los modelos ANN

Hiperparámetros	Modelo 1	Modelo 2	Modelo 3
Tamaño del lote	8	16	32
Número de neuronas en la primera capa	800	1000	600
Función de activación de la primera capa	Relu	Softsign	Softplus
Número de neuronas en la segunda capa	700	600	400
Función de activación de la segunda capa	Relu	Softsign	Softplus
Tasa de Dropout de la primera capa	0.4	0.5	0.3
Tasa de Dropout de la segunda capa	0.3	0.4	0.2
Valor inicial de la tasa de aprendizaje	1e-4	1e-4	1e-4
Algoritmo de optimización	Adam	Adam	Adam
Función de pérdida	EC	EC	EC
Número de épocas de entrenamiento	100	100	100

III-C. Resultados del estudio de referencia

Los resultados expuestos en el cuadro III muestran que el enfoque BGOA-EANNs logra una alta precisión en diferentes regímenes operativos, con valores superiores al 93 % en todos los casos. Destacan especialmente los regímenes 2 y 3, donde la precisión alcanza el 97.86 % y el 99.02 %, respectivamente, lo cual indica una gran capacidad del modelo para adaptarse a diferentes condiciones y realizar predicciones confiables.

Este nivel de precisión sugiere que el modelo es robusto y tiene un desempeño consistente en la estimación de la vida útil restante (RUL) de los componentes, lo que resulta fundamental para aplicaciones en las que se requiere un diagnóstico preciso y confiable. La consistencia en el rendimiento a través de distintos regímenes refuerza la confiabilidad del enfoque BGOA-EANNs frente a condiciones variadas de operación.

Cuadro III: Precisión del enfoque BGOA-EANNs en distintos regímenes [1].

Régimen	Precisión BGOA-EANNs(%)
Régimen 1	96.93
Régimen 2	97.86
Régimen 3	99.02
Régimen 4	93.15
Régimen 5	96.02
Régimen 6	96.05

IV. RESULTADOS Y DISCUSIÓN

IV-A. Resultados intermedios

Previo a la obtención de los resultados, resulta de importancia las etapas que llevan a la selección de los parámetros y características tomados en cuenta por el modelo computacional. Los datos dados contienen una gran cantidad de lectura de sensores, posibles modos de operación y corresponden a diferentes motores; debido a esta gran variedad es que es necesario agrupar los datos en clusters que comparten características como parte del preprocesado. Cada cluster representa un modo de fallo y están definidos principalmente por sus *settings configurations*. Para esta agrupación se utiliza el método *k-means* y se obtienen 6 clusters bien definidos, tal y como se puede apreciar en la figura 1.

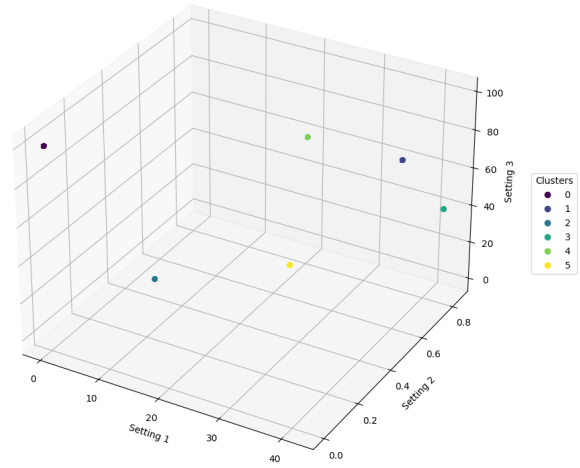


Figura 1: Clusters obtenidos por medio del algoritmo k-means.

Antes de realizar el entrenamiento de los modelos computacionales, se analizan los datos suministrados para identificar cuales de ellos aportan realmente a la determinación del RUL. Para ello se grafican los valores de las distintas categorías y mediciones de sensores en contra al RUL remanente, de esto modo se puede detectar si hay algún sensor o característica que no varíe en función al RUL. A continuación, en la figura 2, se muestran los resultados de este ejercicio.

A partir de la figura 2, se puede observar como no hay ningún sensor con medición constante a lo largo del RUL. Los sensores 13, 16 y 19 parecieran tener lecturas uniformes en relación al RUL pero tras un estudio de correlación de estas variables, se identifica que no son independientes y por ende podrían ser relevantes.

De las distribuciones y el estudio de correlación, se toma la decisión de utilizar el total de los sensores en cuenta para el entrenamiento de los modelos neuronales en la categorización y para la predicción se utiliza los componentes que sugieren los autores en [1].

IV-B. Resultados categorización

Se elaboró un modelo computacional que pudiera categorizar las distintas mediciones de los sensores en una de 4

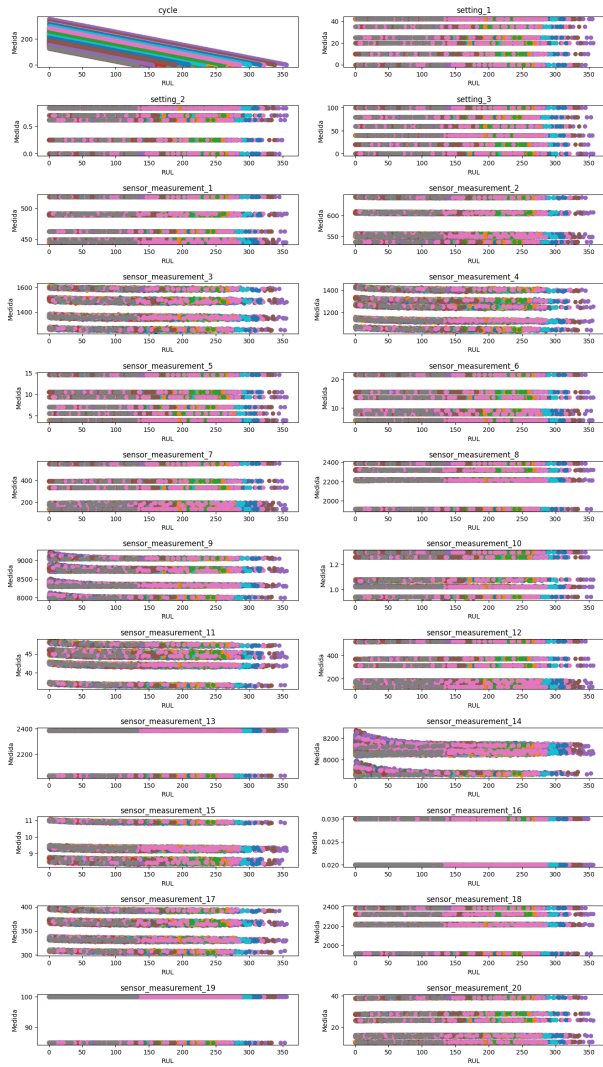


Figura 2: Relación de los valores de las características y la vida útil remanente.

categorías según el RUL restante. A continuación, en la tabla IV, se muestra la definición de las categorías.

Cuadro IV: Definición de categorías según RUL restante.

Categoría	RUL en ciclos restante
Urgente	1-10
Corto	11-50
Medio	51-150
Largo	> 150

Una vez que los modelos computacionales fueron entrenados con las características seleccionadas, se procede al análisis de los resultados de los mismos. A continuación, se presentan las métricas de precisión de los modelos en la tabla V. Además, en la figura 4, se muestran las matrices de confusiones para cada uno de los clúster de datos.

En esta tabla V se puede apreciar como, a pesar de que para cada clúster pueden haber modelos muy buenos y otros que tienen precisiones más bajas, por medio del método de conjunto de votación, se permite mantener un porcentaje de precisión aceptable para la decisión final, eliminando los extremos que

Cuadro V: Precisión de clasificación de los modelos computacionales según el clúster de datos.

	Modelo 1	Modelo 2	Modelo 3	Decisión Final
Clúster 0	90,20 %	88,00 %	81,36 %	87,13 %
Clúster 1	81,50 %	83,84 %	89,33 %	83,50 %
Clúster 2	94,06 %	96,50 %	84,59 %	95,55 %
Clúster 3	92,63 %	95,21 %	86,77 %	95,80 %
Clúster 4	89,87 %	91,80 %	86,26 %	91,09 %
Clúster 5	98,00 %	96,22 %	88,23 %	96,65 %
Promedio	91,04 %	91,93 %	86,09 %	91,62 %

pueden generarse en los modelos computacionales. Además, se puede apreciar como los clusters 2, 3, 4 y 5 contienen datos que para los modelos computacionales es más fácil clasificar y los clúster 0 y 1 son los que menores precisiones presentan.

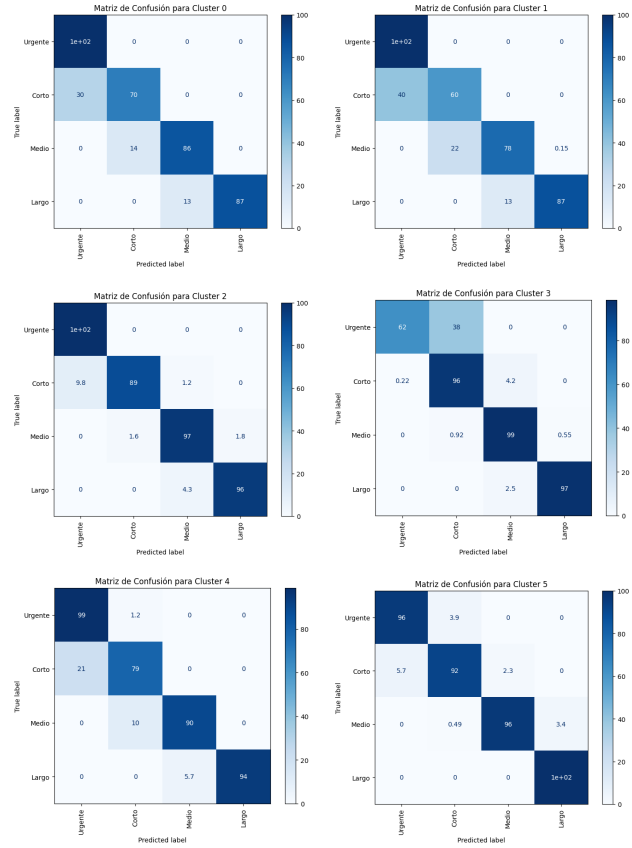


Figura 3: Matrices de confusión para cada uno de los clúster de datos.

La figura 4 ayuda a la identificación de debilidades y patrones de falla del modelo generado. Se puede observar como en el caso de los clúster 0 y 1, cuando la decisión final falla, se suele subestimar la urgencia del mantenimiento. Para el clúster 3 pasa lo contrario, cuando el modelo falla, suele sobrestimar la urgencia del mantenimiento. Además, se logra apreciar un patrón en todos los clúster, la precisión de clasificación suele decender cuando la urgencia aumenta. Es decir, la clasificación en las categorías urgente y corta son las que más fallas presentan.

IV-C. Resultados predicción

Aunque el modelo propuesto en el artículo de referencia [1] no hacía predicciones directamente del RUL, se modificó este para que pueda predecir valores exactos de vida útil remanente. A continuación se muestran las distribuciones obtenidas por clúster de RUL predicho con respecto al RUL real, donde se puede ver el desempeño del modelo generado.

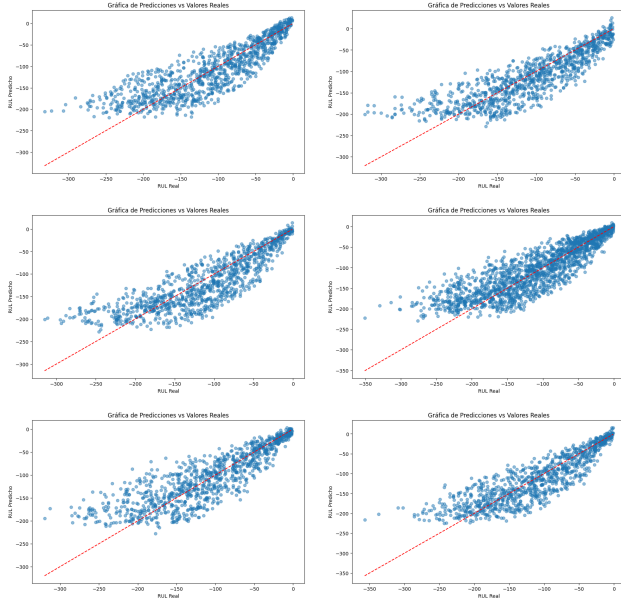


Figura 4: Distribuciones de predicción para cada uno de los clusters de datos.

En términos generales, la tendencia se repite con respecto a la clasificación. Los clusters 0 y 1 son los que menor precisión presentan mientras que los clusters 4 y 5 son los que más se aproximan. Además, se nota una disminución de la precisión en general ya que en predicciones la predicción promedio está al rededor de 61 %. Otra observación importante es que la predicción del RUL aumenta su precisión conforme se acerca más a la falla y son aquellos valores donde aún hacen por encima de 100 ciclos donde se ve una mayor dispersión.

En la tabla VI se muestran los resultados obtenidos con el conjunto de datos de la competencia.

Cuadro VI: Valores RUL para cada uno de los motores

Motor	RUL
1	37.52
2	113.16
3	38.71
4	39.71
5	9.14
6	157.60
7	29.63
8	117.56
9	50.97
10	170.85

V. CONCLUSIONES

1. La identificación de los seis modos de operación permitió segmentar el entrenamiento de las redes neuronales

artificiales (ANNs), lo que mejoró el modelado del comportamiento de las mediciones. Esta segmentación optimizó la precisión del modelo al enfocar el entrenamiento en patrones específicos de cada modo, logrando así resultados más consistentes.

2. En el artículo original, no se especifican los límites utilizados para segmentar las cuatro categorías de vida útil restante, por lo que se tomó una decisión basada en juicio experto para definir estos umbrales. Esta elección podría explicar las discrepancias observadas entre los resultados obtenidos en nuestro trabajo y los del estudio original.
3. La combinación del algoritmo de optimización BGOA con ANNs evolucionadas demostró ser una estrategia eficaz para el diagnóstico de fallas, al permitir una selección óptima de sensores y reducir la complejidad del modelo sin sacrificar precisión. Este enfoque representa una contribución valiosa para la optimización de modelos en sistemas de diagnóstico y monitoreo, especialmente en aplicaciones críticas como el diagnóstico de fallas en sistemas aeroespaciales.
4. Como resultado del proceso de selección mediante el BGOA, se descartaron cinco sensores que presentaban baja relevancia en el análisis. Esto permitió que el modelo de predicción concentrara su aprendizaje en el comportamiento de los componentes más significativos, mejorando su capacidad de generalización y reduciendo el ruido en las mediciones.
5. Fue necesario ajustar la arquitectura de las ANNs para adaptarlas a un problema de regresión en lugar de categorización, lo cual incluyó la modificación de funciones de activación, la función de pérdida y la cantidad de neuronas en las capas finales. Estos cambios mejoraron la precisión del modelo en la predicción continua de valores en los modos de operación.
6. El modelo de EANNs mostró una precisión general ligeramente inferior a la reportada por los autores del modelo original. Sin embargo, al incluir todas las características del conjunto de datos, la precisión mejoró en comparación con los resultados obtenidos al omitir las características que los autores recomendaron excluir a través de su análisis con GOA. Esta diferencia puede deberse en parte a la definición de los parámetros de categorización del RUL aparente en la fase de entrenamiento, ya que los autores no especificaron los criterios exactos que utilizaron en su modelo. A través de pruebas se observó que estos parámetros influyen significativamente en la precisión final del modelo.

VI. LINK AL VIDEO DE LA PRESENTACIÓN

<https://youtu.be/6VMrHRsxjuM>

REFERENCIAS

- [1] D. Ezzat, A. E. Hassanien, A. Darwish, M. Yahia, A. Ahmed and S. Abdelghafar, "Multi-Objective Hybrid Artificial Intelligence Approach for Fault Diagnosis of Aerospace Systems", in IEEE Access, vol. 9, pp. 41717-41730, 2021, doi: 10.1109/ACCESS.2021.3064976.

- [2] A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage Propagation Modeling for Aircraft Engine Prognostics", *NASA Technical Report*, pp. 1-9, 2008. Available: <https://ntrs.nasa.gov/citations/20090029214>
- [3] A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation", in *Proceedings of the 1st International Conference on Prognostics and Health Management (PHM08)*, Denver, CO, Oct. 2008.
- [4] NASA Prognostics Center of Excellence, "PHM08 Prognostics Data Challenge Dataset Description", 2016. [Online]. Available: <https://www.nasa.gov/content/prognostics-center-of-excellence-data-set-repository>
- [5] O. Bektas, F. Camci, and M. F. Anarim, "Reconstructing Secondary Test Database from PHM08 Challenge Data Set", *International Journal of Prognostics and Health Management*, vol. 9, no. 2, pp. 1-11, 2018. Available: https://www.researchgate.net/publication/329080682_Reconstructing_Secondary_Test_Database_from_PHM08_Challenge_Data_Set
- [6] Saremi, S., Mirjalili, S., & Lewis, A. (2017). "Grasshopper Optimisation Algorithm: Theory and application". *Advances in Engineering Software*, 105, 30-47.