



UNIVERSIDAD NACIONAL DE CHIMBORAZO
VICERRECTORADO DE INVESTIGACIÓN, VINCULACIÓN Y
POSGRADO
DIRECCIÓN DE POSGRADO

Análisis comparativo de algoritmos de machine learning aplicados a la predicción de fallas en un motor eléctrico basados en datos sensoriales

Trabajo de Titulación para optar al título de
Magíster en Matemática Aplicada con mención en Matemática
Computacional

AUTORA:

Ing. María Cristina Robayo Villarroel, Mgs.

TUTORA:

Ing. Lorena Paulina Molina Valdiviezo, PhD.

Riobamba, Ecuador. 2025

DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS

Yo, **María Cristina Robayo Villarroel** con número único de identificación **180424254-1**, declaro y acepto ser responsable de las ideas, doctrinas, resultados y lineamientos alternativos realizados en el presente trabajo de titulación denominado: “Análisis comparativo de algoritmos de Machine Learning aplicados a la predicción de fallas en un motor eléctrico basados en datos sensoriales” previo a la obtención del grado de **Magíster en Matemática Aplicada con mención en Matemática Computacional**.

Declaro que mi trabajo investigativo pertenece al patrimonio de la Universidad Nacional de Chimborazo de conformidad con lo establecido en el artículo 20 literal j) de la Ley Orgánica de Educación Superior LOES.

Autorizo a la Universidad Nacional de Chimborazo que pueda hacer uso del referido trabajo de titulación y a difundirlo como estime conveniente por cualquier medio conocido, y para que sea integrado en formato digital al Sistema de Información de la Educación Superior del Ecuador para su difusión pública respetando los derechos de autor, dando cumplimiento de esta manera a lo estipulado en el artículo 144 de la Ley Orgánica de Educación Superior LOES.

Riobamba, 28 de julio de 2025

Ing. María Cristina Robayo Villarroel, Mgs.

N.U.I. 180424254-1



ACTA DE CULMINACIÓN DE TRABAJO DE TITULACIÓN

En la ciudad de Riobamba a los 18 días del mes de julio del año 2025, los miembros del Tribunal designado por la Comisión de Posgrado de la Universidad Nacional de Chimborazo, reunidos con el propósito de analizar y evaluar el Trabajo de Titulación bajo la modalidad Proyecto de titulación con componente investigación aplicada y/o desarrollo, CERTIFICAMOS lo siguiente:

Que, una vez revisado el trabajo titulado: "ANÁLISIS COMPARATIVO DE ALGORITMOS DE MACHINE LEARNING APLICADOS A LA PREDICCIÓN DE FALLAS EN UN MOTOR ELÉCTRICO BASADOS EN DATOS SENSORIALES", perteneciente a la línea de investigación: INGENIERÍA INFORMÁTICA, presentado por la maestrante ROBAYO VILLARROEL MARIA CRISTINA, portador de la cédula de ciudadanía No. 1804242541 estudiante del programa de Maestría en MATEMÁTICA APLICADA CON MENCIÓN EN MATEMÁTICA COMPUTACIONAL, se ha verificado que dicho trabajo cumple al 100% con los parámetros establecidos por la Dirección de Posgrado de la Universidad Nacional de Chimborazo.

Es todo cuanto podemos certificar, en honor a la verdad y para los fines pertinentes.

Atentamente,



Firmado electrónicamente por:
LORENA PAULINA
MOLINA VALDIVIEZO
Validado electrónicamente con FIRMADOC

Ing. Lorena Paulina
Molina Valdiviezo
Ph.D.
TUTOR



Firmado electrónicamente por:
CIRO DIEGO
RADICELLI GARCIA
Validado electrónicamente con FIRMADOC

Ing. Ciro Diego
Radicelli Garcia
Ph.D.
**MIEMBRO DEL
TRIBUNAL 1**



Firmado electrónicamente por:
MARLON DANILO
BASANTES VALVERDE
Validado electrónicamente con FIRMADOC

Dr. Marlon Danilo
Basantes Valverde
Ph.D.
**MIEMBRO DEL
TRIBUNAL 2**



Riobamba, 27 de julio de 2025

CERTIFICADO

De mi consideración:

Yo Lorena Paulina Molina Valdiviezo, certifico que María Cristina Robayo Villarroel con cédula de identidad No. 1804242541 estudiante del programa de Maestría Matemática Aplicada con mención en Matemática Computacional , cohorte tercera presentó su trabajo de titulación bajo la modalidad de Proyecto de titulación con componente de investigación aplicada y/o desarrollo denominado: Análisis comparativo de algoritmos de Machine Learning aplicados a la predicción de fallas en un Motor Eléctrico basados en datos sensoriales, el mismo que fue sometido al sistema de verificación de similitud de contenido COMPILATION identificando el 6% de similitud en el texto.

Es todo en cuanto puedo certificar en honor a la verdad.

Atentamente,



Lorena Paulina
Molina Valdiviezo



Ing. Lorena Paulina Molina Valdiviezo Ph.D.

CI: 0603228156

Adj.-

- Resultado del análisis de similitud (Compilation)

DEDICATORIA

A mis padres, quienes son mi fuerza, motor e inspiración constante, este logro es el resultado de su amor incondicional, sus sacrificios silenciosos y la firme creencia que supieron inculcarme: que no existen límites cuando se actúa con fe, esfuerzo y corazón; a mi familia, por su compañía, respaldo y cariño infaltable; y a quienes con palabras de aliento, gestos sinceros y apoyo desinteresado estuvieron presentes convirtiéndose en mi impulso y fortaleza.

¡Esto es por y para Ustedes!

AGRADECIMIENTO

Agradezco a Dios por concederme salud, vida y fortaleza durante este proceso académico; a la Universidad Nacional de Chimborazo por su formación de calidad y compromiso con el desarrollo profesional e investigativo; a Lorenita Molina por su valiosa orientación y acompañamiento; y a todas las personas que de manera directa o indirecta contribuyeron a la culminación de este trabajo.

ÍNDICE GENERAL

DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS _____	II
ACTA DE CULMINACIÓN DEL TRABAJO DE TITULACIÓN _____	III
CERTIFICADO DEL CONTENIDO DE SIMILITUD _____	IV
DEDICATORIA _____	V
AGRADECIMIENTO _____	VI
ÍNDICE GENERAL _____	VII
ÍNDICE DE FIGURAS _____	XI
ÍNDICE DE ECUACIONES _____	XIII
RESUMEN _____	- 1 -
ABSTRACT _____	- 2 -
INTRODUCCIÓN _____	- 3 -
CAPITULO I GENERALIDADES _____	- 5 -
1.1. Planteamiento del problema _____	- 5 -
1.2. Justificación de la investigación _____	- 6 -
1.3. Alcance _____	- 7 -
1.3.1. Orientación _____	- 7 -
1.3.2. Limitaciones _____	- 7 -
1.3.3. Aportes _____	- 7 -
1.4. Objetivos _____	- 8 -
1.4.1. Objetivo General _____	- 8 -
1.4.2. Objetivos Específicos _____	- 8 -
CAPÍTULO II ESTADO DEL ARTE Y LA PRÁCTICA _____	- 9 -
2.1. Antecedentes Investigativos _____	- 9 -
2.1.1. Antecedentes Nacionales _____	- 9 -
2.1.2. Antecedentes internacionales _____	- 11 -
2.2. Fundamentación Teórica _____	- 12 -
2.2.1. Mantenimiento _____	- 12 -
2.2.1.1. Tipo de Mantenimiento _____	- 14 -
2.2.1.1.1. Mantenimiento Correctivo _____	- 15 -
2.2.1.1.2. Mantenimiento Detectivo _____	- 15 -
2.2.1.1.3. Mantenimiento Preventivo _____	- 16 -
2.2.1.1.4. Mantenimiento Predictivo _____	- 16 -

2.2.2.	Inteligencia Artificial	- 16 -
2.2.3.	Internet de las cosas	- 17 -
2.2.4.	Sensores	- 18 -
2.2.5.	Machine Learning	- 19 -
2.2.5.1.	El aprendizaje supervisado	- 20 -
2.2.5.2.	El aprendizaje no supervisado	- 22 -
2.2.5.3.	El aprendizaje de refuerzo (razonamiento):	- 23 -
2.2.6.	Machine Learning aplicado a la detección de fallas	- 25 -
2.2.7.	Big Data	- 25 -
2.2.8.	Cloud Computing	- 26 -
2.2.9.	Máquinas de vectores de regresión	- 26 -
2.2.10.	Detección de fallas	- 27 -
CAPÍTULO III DISEÑO METODOLÓGICO		- 28 -
3.1.	Enfoque de la Investigación	- 28 -
3.2.	Diseño Metodológico	- 28 -
3.3.	Tipo y Nivel de investigación	- 28 -
3.4.	Recolección de Datos	- 29 -
3.5.	Procesamiento e Interpretación de Datos	- 29 -
3.6.	Población y Muestra	- 30 -
3.7.	Procedimientos éticos	- 30 -
3.8.	Planificación	- 30 -
3.9.	Presupuesto	- 31 -
CAPÍTULO IV ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS		- 33 -
4.1.	Fases para la implementación	- 33 -
4.2.	Obtención de datos sensoriales	- 34 -
4.2.1.	Selección de sensores	- 35 -
4.2.1.1.	Sensor de Velocidad: Encoder FC-03	- 35 -
4.2.1.2.	Sensor de temperatura: MLX90614	- 36 -
4.2.1.3.	Acelerómetro MPU6050	- 37 -
4.2.1.4.	Sensor de corriente SCT-013	- 38 -
4.3.	Diseño del sistema de adquisición de datos sensoriales y banco de pruebas	- 39 -
4.4.	Selección de algoritmos de machine Learning	- 42 -
4.5.	Recolección de Datos	- 44 -

4.6. Entrenamiento de Machine Learning	- 49 -
4.6.1. Aprendizaje Supervisado	- 50 -
4.6.1.1. Árboles de decisión	- 50 -
4.6.1.2. Máquinas de Vectores de Soporte (SVM)	- 51 -
4.6.1.3. Naive Bayes	- 53 -
4.6.2. Aprendizaje No Supervisado	- 54 -
4.6.2.1. Clustering (Agrupamiento por K-Means)	- 54 -
4.6.3. Redes Neuronales Artificiales	- 55 -
4.7. Resultados	- 57 -
4.7.1. Resultados de la variable - Velocidad Angular	- 60 -
4.7.1.1. Algoritmos supervisados y Red Neuronal	- 61 -
4.7.1.2. Algoritmos no supervisados	- 64 -
4.7.1.3. Análisis	- 65 -
4.7.2. Resultados de la variable - Corriente	- 66 -
4.7.2.1. Algoritmos supervisados y Red Neuronal	- 66 -
4.7.2.2. Algoritmos no supervisados	- 70 -
4.7.2.3. Análisis	- 71 -
4.7.3. Resultados de la variable - Temperatura	- 72 -
4.7.3.1. Algoritmos supervisados y Red Neuronal	- 72 -
4.7.3.2. Algoritmos no supervisados	- 75 -
4.7.3.3. Análisis	- 77 -
4.7.4. Resultados de la variable vibración	- 78 -
4.7.4.1. Algoritmos supervisados y Red Neuronal	- 78 -
4.7.4.2. Algoritmos no supervisados	- 82 -
4.7.4.3. Análisis	- 83 -
CAPÍTULO V MARCO PROPOSITIVO	- 85 -
4.8. Propuesta	- 85 -
4.8.1. Estructura operativa	- 85 -
4.8.2. Impacto esperado de la propuesta	- 86 -
4.8.3. Relevancia académica y profesional	- 87 -
CONCLUSIONES	- 88 -
RECOMENDACIONES	- 89 -
REFERENCIAS BIBLIOGRÁFICAS	- 90 -

ÍNDICE DE TABLAS

Tabla 1. Presupuesto _____	- 32 -
Tabla 2. Configuraciones de funcionamiento del esmeril eléctrico _____	- 45 -
Tabla 3. Extracto de datos registrados en formato .csv _____	- 46 -
Tabla 4. Métricas de evaluación para la variable velocidad angular _____	- 62 -
Tabla 5. Métricas de evaluación para la variable Corriente _____	- 68 -
Tabla 6. Métricas de evaluación para la variable Corriente _____	- 74 -
Tabla 7. Métricas de evaluación para la variable Vibraciones _____	- 80 -

ÍNDICE DE FIGURAS

Figura 1. Evolución del mantenimiento _____	- 13 -
Figura 2. Clasificación de los tipos de mantenimiento _____	- 14 -
Figura 3. Tipos de Algoritmos de Machine Learning _____	- 20 -
Figura 4. Diagrama de flujo del aprendizaje supervisado _____	- 20 -
Figura 5. Diagrama de flujo del aprendizaje no supervisado _____	- 22 -
Figura 6. Diagrama de flujo del aprendizaje por refuerzo _____	- 24 -
Figura 7. Sensor De Corriente Eléctrica Sct-013 _____	- 35 -
Figura 8. Sensor MLX90614: _____	- 36 -
Figura 9. Sensor De Corriente Eléctrica Sct-013 _____	- 37 -
Figura 10. Sensor de Corriente SCT-013 _____	- 38 -
Figura 11. Esquema General _____	- 39 -
Figura 12. Diseño del banco de pruebas digital _____	- 40 -
Figura 13. Implementación del banco de pruebas _____	- 41 -
Figura 14. Parámetros del esmeril _____	- 41 -
Figura 15. Tipos de Aprendizaje Automático _____	- 42 -
Figura 16. Aprendizaje Automático Clásico _____	- 43 -
Figura 17. Registro de velocidad angular del esmeril eléctrico en estado sin carga ____	- 47 -
Figura 18. Registro de corriente del esmeril eléctrico sin carga _____	- 47 -
Figura 19. Registro de temperatura del esmeril eléctrico sin carga _____	- 48 -
Figura 20. Registro de vibraciones del esmeril eléctrico sin carga _____	- 48 -
Figura 21. Preparación de datos para entrenamiento en MATLAB _____	- 50 -
Figura 22. Código de entrenamiento con Árboles de Decisión _____	- 51 -
Figura 23. Código de implementación del modelo SVM _____	- 52 -
Figura 24. Código para Naive Bayes _____	- 53 -
Figura 25. Proceso general de aprendizaje no supervisado _____	- 54 -
Figura 26. Código de agrupamiento con K-Means en MATLAB _____	- 55 -
Figura 27. Proceso de implementación y evaluación de algoritmos en MATLAB. ____	- 57 -
Figura 28. Matrices de confusión – Variable velocidad angular _____	- 58 -
Figura 29. Matrices de confusión – Variable velocidad angular _____	- 61 -
Figura 30. Distribución de datos de velocidad angular _____	- 64 -
Figura 31. Clustering de velocidad angular y Silhouette por número de clústeres ____	- 65 -
Figura 32. Matrices de Confusión – Variable Corriente _____	- 67 -

Figura 33. Distribución de los datos de Corriente _____	- 70 -
Figura 34. Clustering de Corriente _____	- 71 -
Figura 35. Matrices de Confusión – Variable Temperatura _____	- 73 -
Figura 36. Distribución de los datos de Temperatura _____	- 76 -
Figura 37. Clustering y valores silhouette para la variable Temperatura _____	- 77 -
Figura 38. Matrices de Confusión – Variable Vibración _____	- 79 -
Figura 39. Distribución de los datos de Vibraciones _____	- 82 -
Figura 40. Clustering y valores silhouette para la variable Vibraciones _____	- 83 -

ÍNDICE DE ECUACIONES

Ecuación 1. Índice de Gini	- 51 -
Ecuación 2. Función objetivo de la SVM	- 52 -
Ecuación 3. Función Kernel (RBF)	- 52 -
Ecuación 4. Clasificador de Bayes	- 53 -
Ecuación 5. Distancia Euclidiana	- 55 -
Ecuación 6. Salida de una neurona	- 56 -
Ecuación 7. Error cuadrático medio,	- 56 -
Ecuación 8. Fórmula de la exactitud.	- 58 -
Ecuación 9. Fórmula de la precisión.	- 59 -
Ecuación 10. Fórmula de la Sensibilidad.	- 59 -
Ecuación 11. Fórmula de la especificidad o tasa de verdaderos negativos.	- 59 -
Ecuación 12. Fórmula de tasa de falsos negativos.	- 59 -
Ecuación 13. Fórmula del valor predictivo positivo.	- 60 -
Ecuación 14. Fórmula del valor predictivo negativo.	- 60 -
Ecuación 15. Fórmula del error.	- 60 -

RESUMEN

Las fallas inesperadas en motores eléctricos representan un riesgo operativo y económico para la industria, por lo que, es necesario implementar estrategias de mantenimiento predictivo basadas en inteligencia artificial. Este estudio tiene como objetivo comparar algoritmos de aprendizaje automático aplicados a la predicción de fallas, utilizando variables sensoriales como corriente, temperatura, vibración y velocidad angular. La investigación parte de la necesidad de identificar el modelo con mayor precisión para su integración en entornos industriales reales.

Se desarrolló un entorno de monitoreo en tiempo real con un esmeril eléctrico sometido a distintas cargas, durante catorce días, capturando una muestra por segundo. Los datos obtenidos se procesaron con herramientas matemáticas y computacionales para entrenar y evaluar modelos supervisados (Árboles de Decisión, Máquinas de Vectores de Soporte, Naive Bayes), no supervisados (K-Means) y redes neuronales artificiales. El rendimiento de cada modelo fue analizado mediante métricas derivadas de matrices de confusión.

Los resultados mostraron que las redes neuronales superaron en relación con los modelos supervisados y no supervisados con una precisión promedio del 97,0%, destacando su capacidad de generalización y adaptabilidad. Se concluye que este enfoque representa una alternativa viable para optimizar la gestión del mantenimiento en sistemas eléctricos rotativos. La propuesta desarrollada tiene potencial de aplicación práctica, y futuras investigaciones podrían enfocarse en arquitecturas más complejas y su validación en diferentes tipos de maquinaria industrial.

Palabras clave: *aprendizaje automático, mantenimiento predictivo, motor eléctrico*

Abstract

Unexpected failures in electric motors pose both operational and economic risks to industry, making it essential to implement predictive maintenance strategies based on artificial intelligence. This study aims to compare machine learning algorithms applied to fault prediction using sensor-based variables such as current, temperature, vibration, and angular velocity. The research addresses the need to identify the most accurate model for integration into real industrial environments.

A real-time monitoring environment was developed using an electric bench grinder subjected to varying loads over a fourteen-day period, with data captured at a rate of one sample per second. The collected data were processed using mathematical and computational tools to train and evaluate supervised models (Decision Trees, Support Vector Machines, Naive Bayes), unsupervised models (K-Means), and artificial neural networks. The performance of each model was analyzed using metrics derived from confusion matrices.

Results showed that artificial neural networks outperformed both supervised and unsupervised models, achieving an average accuracy of 96.5%, demonstrating superior generalization and adaptability. The study concludes that this approach offers a viable solution for optimizing maintenance management in rotating electrical systems. The proposed model holds practical application potential, and future research could focus on more complex architectures and validation across various types of industrial machinery.

Keywords: machine learning, predictive maintenance, electric motor



Mario Nicolas Salazar
Ramos



Revised by
Mario N. Salazar
0604069781

INTRODUCCIÓN

La industria actual se encuentra en constante evolución, impulsada por la cuarta revolución industrial y la integración de tecnologías digitales como la inteligencia artificial y el Internet de las cosas. Este avance tecnológico ha propiciado la aparición de nuevas herramientas para optimizar la eficiencia y la seguridad en los procesos productivos, entre las que destaca el mantenimiento predictivo. A través del análisis de datos en tiempo real, el mantenimiento predictivo permite anticipar posibles fallos en la maquinaria, evitando costosas interrupciones en la producción y accidentes laborales. El presente proyecto se centra en el análisis comparativo de algoritmos de Machine Learning aplicados a la predicción de fallas en un motor eléctrico.

Los motores eléctricos son uno de los componentes más importantes a nivel industrial, su correcto funcionamiento es crucial para garantizar la continuidad de las operaciones, un fallo inesperado en un motor puede provocar paradas no programadas en la producción teniendo como resultado pérdidas económicas y riesgos para la seguridad de los trabajadores. En respuesta a este problema se plantea como alternativa la generación de datos sensoriales para desarrollar un modelo capaz de identificar patrones que indiquen alertas tempranas de posibles fallas en el motor.

Este proyecto de investigación está justificado en tres ámbitos: académico, empresarial y social. Académicamente, contribuye al conocimiento científico al examinar de manera experimental las capacidades y limitaciones de los algoritmos de machine Learning en la predicción de fallas; empresarialmente, la predicción de fallos en motores eléctricos es crítica para la continuidad operativa y la rentabilidad, a diferencia de los mantenimientos correctivo y preventivo, el mantenimiento predictivo basado en el análisis de datos y el Machine Learning permite optimizar las estrategias de mantenimiento al predecir con precisión el momento óptimo para intervenir el equipo. Esto se traduce en una

reducción de costos, una mayor eficiencia en la producción y una mejora en la seguridad de los trabajadores; socialmente, contribuye a la prevención de accidentes laborales y a la optimización de los procesos industriales, impactando en la economía y el desarrollo.

Esta investigación tiene un enfoque cuantitativo, ya que se basa en la recopilación de datos sensoriales en un ambiente controlado, esta información es almacenada en una base de datos y entrenada mediante algoritmos de Machine Learning para identificar patrones predictivos de fallas en el motor eléctrico. El objetivo principal del presente estudio es evaluar de forma comparativa los modelos seleccionados, por esta razón los datos se exponen al análisis estadístico mediante matrices de confusión y métricas rigurosas, de tal forma que se pueda conocer la efectividad y exactitud de cada algoritmo e identificar al más efectivo.

El contenido del trabajo se estructura en cinco capítulos que abordan de manera integral la temática de la predicción de fallos en motores eléctricos. En el primer capítulo se introduce el problema de estudio, se justifica la importancia de la investigación y se plantean los objetivos generales y específicos del proyecto; el segundo capítulo desarrolla el marco teórico, presentando una revisión de los antecedentes investigativos y los conceptos fundamentales relacionados con el Machine Learning y las técnicas de análisis de datos; en el tercer capítulo se describe el diseño metodológico, detallando el enfoque, tipo y diseño de investigación, así como las técnicas utilizadas para la recolección y análisis de datos, junto con la caracterización de la población del estudio; el cuarto capítulo está dedicado al análisis de los resultados, donde se interpretan los hallazgos obtenidos y se comparan los desempeños de distintos algoritmos de Machine Learning en la predicción de fallos; en el quinto capítulo se presenta el marco propositivo, donde se plantea una propuesta basada en los resultados alcanzados, destacando al algoritmo que mostró mayor eficiencia en el proceso de estudio.

CAPITULO I

GENERALIDADES

1.1. Planteamiento del problema

Los motores eléctricos son componentes esenciales en la industria, pero su funcionamiento puede verse afectado por averías imprevistas que, si no se identifican a tiempo, pueden ocasionar graves consecuencias, como interrupciones en la producción, pérdidas económicas y riesgos para la seguridad de los trabajadores. Según (Quispe Oqueña, 2023), los motores eléctricos consumen aproximadamente el 80% de la energía utilizada en la industria, lo que resalta su importancia en los procesos productivos y la necesidad de asegurar su correcto funcionamiento. Sin embargo, estos motores son propensos a sufrir paros abruptos debido a problemas no identificados previamente, generando pérdidas en la producción y en el ámbito económico.

Con el avance de la tecnología, la industria moderna exige una visión integral y automatizada para el mantenimiento en sus sistemas, frente a esta necesidad, la aplicación de algoritmos de Machine Learning sobre un conjunto de datos sensoriales previamente receptados representa una solución innovadora, su capacidad de modelar comportamientos no lineales ayuda a la detección temprana de patrones complejos y relaciones ocultas en los datos, permitiendo predecir fallos con mayor precisión y anticipación, incluso antes de que se manifiesten consecuencias visibles.

Al implementar esta solución, se prevé reducir significativamente los valores económicos por interrupciones en la producción y de manera consecuente los riesgos de accidentes laborales. En términos globales la mejora continua del mantenimiento se traduce en procesos más eficientes, robustos y confiables, generando un entorno productivo más competitivo. (Adu-Amankwa et al., 2019)

1.2. Justificación de la investigación

Este proyecto de investigación, orientado al análisis comparativo de algoritmos de Machine Learning aplicados a la predicción de fallas en un motor eléctrico se justifica por su importancia al generar un impacto significativo en diversas áreas:

- **Optimización del mantenimiento predictivo:** Desarrollar modelos capaces de detectar patrones anómalos en variables operativas antes de que ocurran fallos reales en los equipos, posibilita que las intervenciones de mantenimiento se realicen de forma oportuna y precisa, evitando mantenimientos innecesarios, extendiendo la vida útil del motor y reduciendo costos operativos.

- **Prevención de Riesgos Laborales:** Las fallas en motores eléctricos pueden desencadenar eventos peligrosos arriesgando la seguridad del personal, especialmente en entornos donde se opera con maquinaria pesada o altos niveles de tensión eléctrica. Este estudio ayuda a evitar las condiciones de riesgo y a fortalecer la cultura de prevención dentro del entorno laboral.

- **Reducción de costos operativos:** Optimiza recursos técnicos y humanos al intervenir únicamente cuando se detecta una probabilidad real de falla, esto reduce gastos innecesarios en inspecciones, reparaciones correctivas y pérdidas por paros imprevistos, favoreciendo una gestión económica más eficiente en entornos industriales.

- **Contribución al avance tecnológico:** La combinación del análisis de datos sensoriales con la inteligencia artificial aporta al conocimiento científico promoviendo la automatización de procesos críticos y abriendo nuevas posibilidades para la implementación de sistemas ciberfísicos en el mantenimiento de maquinaria alineados a los principios de la Industria 4.0. (Delecrodio et al., 2023).

1.3. Alcance

El alcance de este proyecto de investigación se centra en el análisis comparativo de algoritmos de Machine Learning para la predicción de fallas en un motor eléctrico específico utilizando datos sensoriales.

1.3.1. Orientación

La orientación del presente estudio consiste en aplicar y comparar algoritmos de Machine Learning (modelos supervisados y no supervisados), basándose en un conjunto de datos sensoriales recolectados en tiempo real en un sistema experimental controlado, registrando variables como: corriente, vibración, temperatura y velocidad angular, con el fin de determinar cuál ofrece una predicción más precisa y confiable para la detección de falas tempranas en un motor eléctrico específico.

1.3.2. Limitaciones

- **Generalización:** Dado que el análisis se realizó sobre un único tipo de motor eléctrico (esmeril), su uso en otros motores requiere un proceso de reentrenamiento, ajustes y adaptaciones para garantizar resultados equivalentes.

- **Factores externos:** El modelo puede no contemplar ciertos factores externos relevantes en entornos industriales, como interferencias eléctricas, variaciones en el suministro energético o condiciones ambientales, que también son factores incidentes en la aparición de fallas.

1.3.3. Aportes

- **Conocimiento científico:** Se generará conocimiento sobre la aplicación de algoritmos de Machine Learning en el mantenimiento predictivo de motores eléctricos.

- **Eficiencia industrial:** Se contribuirá a la optimización de los procesos de mantenimiento, permitiendo reducir costos operativos y reforzar la seguridad laboral en entornos industriales.

- **Base para futuras investigaciones:** Este trabajo constituye un punto de partida sólido para futuras investigaciones en el área del mantenimiento predictivo, facilitando la exploración de nuevos modelos de Machine Learning aplicados a distintos tipos de motores eléctricos y diferentes variables.

1.4. Objetivos

1.4.1. Objetivo General

- Analizar algoritmos de machine learning aplicados a la predicción de fallas en un motor eléctrico basados en datos sensoriales

1.4.2. Objetivos Específicos

- Realizar una revisión sistemática de la literatura sobre sistemas de mantenimiento basados en algoritmos de Machine Learning.

- Diseñar e implementar un sistema de adquisición de datos sensoriales para la generación de una base de datos que permita el entrenamiento y la evaluación de los algoritmos de Machine Learning.

- Seleccionar y entrenar diferentes algoritmos de Machine Learning para la predicción de fallos en el motor eléctrico.

- Evaluar el rendimiento de los algoritmos seleccionados mediante matrices de confusión y métricas, identificando el algoritmo más eficaz para la predicción de fallos.

CAPÍTULO II

ESTADO DEL ARTE Y LA PRÁCTICA

2.1. Antecedentes Investigativos

En los últimos años, el avance de las tecnologías de diagnóstico predictivo ha impulsado la aplicación de algoritmos de Machine Learning en el monitoreo y detección de fallas en motores eléctricos. Esta tendencia responde a la necesidad industrial de optimizar el mantenimiento, reducir tiempos de inactividad y minimizar los costos operativos. A continuación, se presentan estudios relevantes, tanto nacionales como internacionales, Su revisión permite identificar los enfoques, metodologías y tecnologías previamente empleadas en la predicción de fallas en motores eléctricos, así como reconocer las principales brechas y oportunidades que este estudio busca abordar.

2.1.1. Antecedentes Nacionales

En el estudio desarrollado por (Dávalos Carrera & Vazquez Matute, 2022), se plantea una herramienta de diagnóstico de fallas en motores eléctricos mediante el análisis de sonido y vibración con técnicas de aprendizaje profundo. Se desarrolla una aplicación móvil que captura datos reales de funcionamiento, los transforma en imágenes espectrales y los procesa con modelos de redes neuronales. Se emplean arquitecturas especializadas en clasificación de imágenes, lo que incrementa la precisión del sistema hasta un 93.02%, sin necesidad de sensores especializados ni personal técnico. Se una solución accesible y eficiente, aunque se limita al análisis acústico y no considera otras variables físicas ni la comparación entre algoritmos.

desarrollado en la Universidad Politécnica Salesiana, se implementa un prototipo para el reconocimiento de fallas en rodamientos de motores eléctricos, utilizando el análisis de sonido como variable principal. Se emplea un sistema compuesto por un micrófono y una tarjeta de procesamiento para recolectar y

clasificar señales acústicas. El modelo se entrena con sonidos grabados manualmente y datos de bases acústicas especializadas. Los resultados evidencian una alta precisión incluso en entornos con ruido, validando su utilidad para mantenimiento preventivo. No obstante, el estudio se enfoca únicamente en fallas específicas de rodamientos, sin incluir otras anomalías ni múltiples algoritmos.

(Chicaiza Oña & González Jiménez, 2021) realizó una investigación desarrollada en la Escuela Superior Politécnica del Litoral (ESPE), donde se diseña un sistema de monitoreo en la nube para diagnóstico predictivo de fallas basado en datos históricos de más de 900 motores en una planta cementera, con esta base de datos se entrena un algoritmo de árboles de decisión alcanzando una precisión mayor al 94%. Como un adicional desarrolla una interfaz de visualización en red local que permite observar tendencias y generar reportes validando el uso de datos históricos y aprendizaje supervisado, sin embargo, se limita al uso de variables eléctricas y no contempla la comparación con otros algoritmos ni escenarios dinámicos.

Finalmente, el estudio desarrollado por (Mardones Huala, 2024) se establece un sistema modular para detección temprana de fallas y monitoreo en tiempo real de motores eléctricos, aplicando algoritmos de aprendizaje automático. Se estructura la solución en tres módulos: adquisición de datos, procesamiento y visualización. Se emplean sensores de inercia conectados a microcontroladores para captar aceleraciones, procesadas posteriormente con modelos de clasificación binaria. El clasificador de vectores de soporte alcanza una precisión del 94% en la detección de estados de balanceo y desbalanceo. Se incluye además una interfaz web para el diagnóstico en tiempo real. Aunque representa una solución portable y eficiente, se limita al análisis de aceleraciones sin incorporar otras variables sensoriales.

2.1.2. Antecedentes internacionales

En el estudio citado por (como se citó en (Dávalos Carrera & Vazquez Matute, 2022)), Bansal propone un sistema de mantenimiento predictivo basado en el análisis de corriente eléctrica, utilizando redes neuronales para detectar patrones no lineales durante el funcionamiento de máquinas. Se prescinde del uso de sensores especializados, lo que reduce los costos de implementación. Se demuestra que datos eléctricos comunes permiten realizar diagnósticos confiables mediante inteligencia artificial, siendo una opción viable para empresas con recursos tecnológicos limitados, sin embargo, no contempla la comparación entre algoritmos ni la integración de múltiples variables sensoriales.

(Xu et al., 2024) propone un sistema predictivo utilizando imágenes térmicas y redes neuronales convolucionales de un motor en diferentes condiciones operativas, éstas son procesadas mediante técnicas de mejora visual y mecanismos de atención para enfocar las zonas críticas, alcanzando un desempeño sobresaliente, con métricas perfectas de precisión y sensibilidad en la clasificación de once tipos de fallas. El estudio se destaca por su eficiencia y carácter no invasivo, aunque se limita exclusivamente al análisis termográfico, sin considerar otras variables físicas.

(Malindi & Kuno, 2025) desarrollan un modelo híbrido de mantenimiento predictivo para rodamientos, integrando modelos físicos de desgaste y técnicas de aprendizaje automático. Se utilizan datos del centro de mantenimiento inteligente de la NASA, a los cuales se extraen características estadísticas, espectrales y de tendencia, complementadas con indicadores del modelo de Archard. El modelo se entrena con una red neuronal profunda basada en memoria de largo plazo bidireccional. Se obtiene una mejora significativa en precisión y robustez, reduciendo el error absoluto medio a menos de cuatro ciclos y logrando un

coeficiente de determinación cercano a 0.68. El estudio aporta evidencia sobre la efectividad del enfoque híbrido, aunque se enfoca únicamente en rodamientos y no compara diferentes algoritmos ni otras variables sensoriales.

En (Pourghoraba et al., 2024) se propone un sistema de diagnóstico basado en metaaprendizaje, mismo que funciona con base de datos pequeñas obtenidas de un sistema de a escenarios con datos limitados y condiciones variables, analiza señales de corriente eléctrica transformadas en imágenes morfológicas utilizadas como entrada para una red basada en memoria de largo plazo. El sistema logra una precisión superior al 99% mostrando robustez ante el ruido eléctrico y adaptándose con facilidad a nuevas fallas con pocos ejemplos. Este trabajo demuestra alta eficiencia y adaptabilidad, sin embargo, se enfoca únicamente en la corriente sin considerar otras variables ni realizar comparaciones.

2.2. Fundamentación Teórica

En este apartado se presenta una revisión sistemática de la literatura relacionada con sistemas de mantenimiento industrial y su evolución, teniendo en cuenta el papel de tecnologías asociadas como inteligencia artificial, el Internet de las Cosas, los sensores, el Big Data y el cloud computing; además se detallan modelos de machine Learning y sus aplicaciones, incluyendo técnicas complementarias como la detección de anomalías y la regresión por vectores de soporte. Esta revisión teórica construye el sustento conceptual de la propuesta investigativa.

2.2.1. *Mantenimiento*

Según la Normativa (Asociación Española de Normalización, 2018) el mantenimiento se define como el conjunto de acciones administrativas, técnicas y de gestión aplicadas a lo largo del ciclo de vida de los equipos, con el propósito de conservar o restaurar su funcionamiento para garantizar su eficiencia y operatividad. En este mismo sentido,

(Mohammed et al., 2023) señalan que la evolución del mantenimiento refleja una transición desde sus inicios reactivos hacia modelos proactivos y prescriptivos, donde el conocimiento anticipado del estado de los equipos resulta esencial. Esta evolución conceptual del mantenimiento ha generado un cambio notable en las estrategias industriales, impulsando la adopción de criterios más inteligentes y eficientes para responder a las necesidades modernas de producción.

De esta manera, surge el concepto especializado de mantenimiento industrial, donde se combinan técnicas y tecnologías avanzadas como el análisis predictivo, Big Data, Internet de las Cosas (IoT) e inteligencia artificial.

En la Figura 1 muestra el avance progresivo del mantenimiento industrial

Figura 1.

Evolución del mantenimiento



Nota. La figura muestra la evolución del mantenimiento a través de las generaciones industriales.

Fuente: Scheffer, C., & Girdhar, P. (2024).

El mantenimiento industrial ha ido evolucionando a través de cuatro generaciones: la primera, conocida como mantenimiento correctivo se caracterizó por acciones reactivas tras la falla del equipo con altos costos y tiempos de inactividad excesivos; la segunda generación o mantenimiento preventivo estuvo enfocado en la ejecución de rutinas programadas basadas en el tiempo o el uso, mejorando de cierta forma la organización en

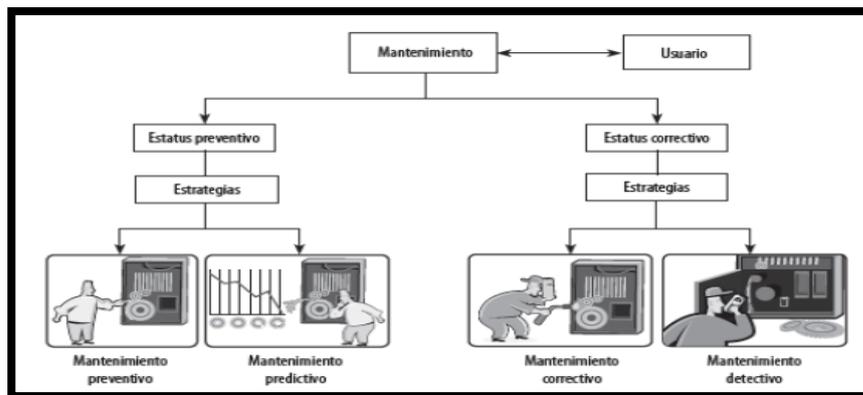
los sistemas; la tercera generación, conocida como mantenimiento predictivo incorporó herramientas de análisis de condición como la medición de vibraciones, el análisis de aceites y termografía anticipando fallas con base en datos; y la cuarta generación o mantenimiento inteligente trabaja a base de sensores, IoT, machine Learning, inteligencia artificial y Big Data para realizar monitoreo en tiempo real, diagnósticos automáticos y toma de decisiones estratégicas con el fin de optimizar eficiencia, seguridad y costos en sistemas industriales.

2.2.1.1. Tipo de Mantenimiento

A manera de síntesis visual, en la Figura 2 se presenta la clasificación de los tipos de mantenimiento industrial en función del estado de operación de los equipos y las estrategias adoptadas por los usuarios.

Figura 2.

Clasificación de los tipos de mantenimiento



Nota. Presenta la clasificación de los tipos de mantenimiento en función de su estatus y estrategias. Fuente: (Tovar Porras, 2021).

En la gráfica se puede distinguir dos grandes enfoques de mantenimiento: el mantenimiento preventivo que incluye intervenciones programadas y predictivas; y el correctivo que incluyen las acciones reactivas y las basadas en la detección; esta clasificación permite visualizar la relación entre la condición del equipo y el tipo de intervención técnica de acuerdo a las necesidades operacionales

2.2.1.1.1. Mantenimiento Correctivo

Se caracteriza por intervenir los equipos únicamente cuando ya se ha producido una falla. Esta estrategia, de carácter reactivo, no implica una programación anticipada, ya que se ejecuta en función de la necesidad inmediata de reparar. Es comúnmente aplicado en sistemas secundarios donde una parada no genera un impacto crítico en la producción. Su principal ventaja radica en la reducción de costos operativos iniciales y la simplicidad de su implementación. Sin embargo, su uso indiscriminado puede derivar en consecuencias negativas como interrupciones no previstas, tiempos de inactividad prolongados, y un incremento en los gastos por repuestos y mano de obra. Además, el deterioro inesperado puede ocasionar daños adicionales en otros componentes del sistema, incrementando el riesgo de accidentes y poniendo en peligro la integridad del personal. Este tipo de eventos afecta directamente la continuidad operativa y reduce la eficiencia general del proceso productivo. (Tovar Porras, 2021).

2.2.1.1.2. Mantenimiento Detectivo

Se enfoca en la localización puntual de fallas mediante inspecciones periódicas dirigidas a elementos específicos de una máquina. Este tipo de mantenimiento se basa en la verificación de parámetros técnicos como la presión, la temperatura u otras variables operativas, con el fin de comprobar si los componentes están funcionando dentro de los rangos normales. Cuando se identifica una desviación en estos valores, se procede con la intervención correspondiente antes de que la situación derive en una falla crítica. De esta manera, se logra corregir defectos que, aunque aún no se han manifestado como averías evidentes, podrían comprometer el rendimiento del equipo si no se atienden oportunamente (Tovar Porras, 2021).

2.2.1.1.3. Mantenimiento Preventivo

El mantenimiento preventivo se la considera como una práctica técnica estructurada que busca preservar el rendimiento óptimo de los equipos a través de intervenciones programadas. Estas actividades como inspecciones, ajustes, limpieza y sustitución de piezas se ejecutan en función de parámetros como el tiempo de operación, el número de ciclos o las recomendaciones del fabricante. Si bien implica una inversión anticipada en recursos, personal y repuestos, optimiza la gestión operativa, disminuye el deterioro progresivo y mejora los indicadores de confiabilidad del sistema. Su aplicación sistemática contribuye también a mantener estándares de calidad y seguridad industrial de alta exigencia operativa (Tovar Porras, 2021).

2.2.1.1.4. Mantenimiento Predictivo

Se caracteriza por el monitoreo continuo del motor bajo condiciones reales de operación utilizando sensores tecnológicos que registran variables como temperatura, vibraciones, presión o niveles de lubricación, esta información es procesada y entrenada mediante algoritmos que permiten la predicción de fallas, minimizando las intervenciones innecesarias y optimizando los recursos disponibles. Las técnicas más utilizadas son la termografía infrarroja, el análisis de aceites y la detección de anomalías por medio de inteligencia artificial, mismas que requieren una inversión significativa para la infraestructura tecnológica y talento humano capacitado, sin embargo, los beneficios operativos y económicos a largo plazo justifican ampliamente la inversión inicial. (Tovar Porras, 2021).

2.2.2. *Inteligencia Artificial*

La inteligencia artificial (IA) ha emergido como una tecnología esencial en la transformación de los procesos industriales, permitiendo que los sistemas

informáticos realicen tareas que antes eran exclusivas de los seres humanos. Estos sistemas son capaces de analizar grandes volúmenes de datos, anticipar posibles eventos, clasificar escenarios operativos y tomar decisiones de manera autónoma o asistida, optimizando así los recursos y aumentando la capacidad de respuesta ante cambios en la producción.

En el mundo actual, donde la tecnología define el ritmo de la productividad, la inteligencia artificial se ha convertido en un recurso fundamental para impulsar la eficiencia operativa, garantizar la seguridad y coordinar procesos complejos de la industria moderna, entre sus aplicaciones más relevantes se destacan el mantenimiento predictivo, la optimización de recursos, la robótica colaborativa, el control automático de calidad y la gestión eficiente de la energía y la logística; construyendo un modelo productivo más ágil, sostenible y adaptable a las demandas cambiantes del entorno industrial global (Rouhiainen, 2024).

2.2.3. *Internet de las cosas*

El Internet de las Cosas (IoT) ha creado un cambio significativo en el entorno industrial al integrar de manera eficiente los sistemas físicos y digitales a través de una comunicación en tiempo real; esta tecnología facilita el monitoreo y el análisis de múltiples variables, lo que convierte procesos convencionales en operaciones inteligentes. Como resultado las industrias pueden supervisar sus activos, anticipar fallas y mejorar su desempeño operativo con mayor precisión y menor costo (Pisano, 2018).

La arquitectura de estos sistemas suele organizarse en capas: una capa física o de percepción que recoge los datos mediante sensores; una capa de comunicación que los transmite, a menudo utilizando tecnologías de red avanzadas como Edge Computing; y una capa de procesamiento y visualización, que emplea plataformas en la nube para analizar la información y presentarla de forma útil al usuario. Esta estructura hace posible el análisis

inmediato de grandes volúmenes de datos y la búsqueda de soluciones en distintos entornos productivos.

La inteligencia artificial ha potenciado más al internet de las cosas. Ya que gracias a algoritmos de aprendizaje automático los sistemas industriales procesan datos y aprenden de ellos para identificar patrones, anticiparse a fallas y optimizar recursos. La asociación entre ambas tecnologías está marcando el rumbo hacia una industria más conectada, autónoma y resiliente (Lemay, 2022).

2.2.4. Sensores

Los sensores son dispositivos esenciales en el monitoreo de sistemas, ya que tienen la capacidad de captar información del entorno y convertirla en datos útiles para el análisis automatizado, obteniendo información precisa sobre variables críticas como la temperatura, la presión o las vibraciones; al funcionar de manera ininterrumpida permiten anticipar fallos, corregir desviaciones y mantener procesos estables, seguros y eficientes.

Existen diversos tipos de sensores, cada uno diseñado para captar una variable específica:

- Los sensores de temperatura miden con precisión las variaciones térmicas, siendo esenciales para supervisar aumentos o descensos de calor en entornos controlados.
- Los sensores de proximidad detectan la presencia o desplazamiento de objetos cercanos mediante señales: electromagnéticas, infrarrojas, ultrasónica u ópticas.
- Los sensores de vibración registran oscilaciones mecánicas, mismos que indican comportamientos anómalos.
- Los sensores inerciales detectan cambios de orientación, aceleración o velocidad angular, proporcionando datos clave sobre la estabilidad y dinámica del sistema.

- Los sensores de corriente miden el flujo eléctrico de manera indirecta, logrando identificar irregularidades en la distribución o consumo de energía. (Sehrawat & Gill, 2019).

El funcionamiento eficiente de una red sensorial depende del diseño del sistema IoT, donde factores como la seguridad, precisión y escalabilidad resultan determinantes; dado que los sensores permanecen en comunicación constante es necesario aplicar mecanismos de protección que prevengan amenazas cibernéticas sin afectar el flujo de datos, al mismo tiempo el sistema debe permitir la integración progresiva de nuevos dispositivos sin generar conflictos ni deteriorar su rendimiento. La fiabilidad del hardware también resulta decisivo, ya que estos dispositivos deben operar de forma continua aun en condiciones adversas para garantizar la integridad de los datos recopilados y la disponibilidad permanente de la información (Mirani et al., 2022; Tightiz & Yoo, 2022).

2.2.5. *Machine Learning*

El aprendizaje automático, conocido como Machine Learning (ML), es una rama de la inteligencia artificial que permite a los sistemas aprender a partir de los datos, reconociendo patrones y mejorando su desempeño sin requerir programación explícita para cada situación. En el entorno industrial, esta tecnología ha transformado la forma en que se detectan anomalías, se anticipan fallas y se optimizan procesos, ya que los modelos son capaces de adaptarse en tiempo real a cambios en el entorno y a variaciones operativas (Guerrero Cano et al., 2019)

De forma general, el Machine Learning se clasifica en tres grandes tipos: aprendizaje supervisado, no supervisado y por refuerzo. Esta clasificación depende del tipo de datos que recibe el sistema y del modo en que se entrena. La Figura 3 ilustra estos enfoques y algunos de sus métodos más comunes.

Figura 3.

Tipos de Algoritmos de Machine Learning



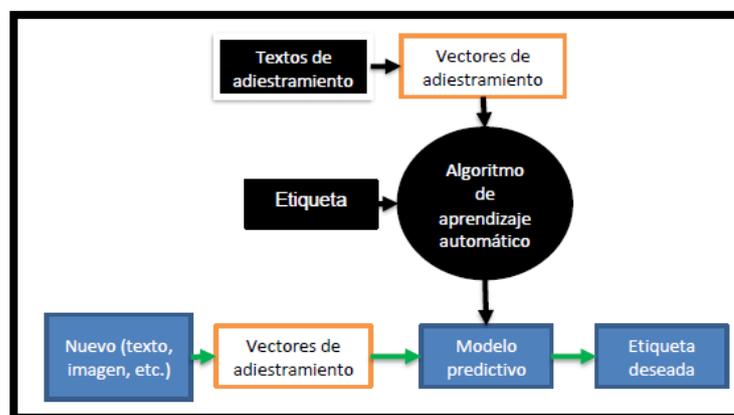
Nota. Clasificación de los tipos de algoritmos de Machine Learning. Fuente: (Sanseviero, 2023).

2.2.5.1.El aprendizaje supervisado

Este tipo de aprendizaje se basa en entrenar al modelo con un conjunto de datos previamente etiquetado, es decir, que incluye las entradas (características) y las salidas esperadas (clases o valores). A través de este proceso, el algoritmo aprende a identificar patrones y relaciones entre las variables independientes y la variable objetivo. Una vez completada la fase de entrenamiento, el modelo puede predecir resultados para nuevos datos no vistos, con base en el conocimiento adquirido (Baradan, 2025). En la figura 4 se muestra su funcionamiento general.

Figura 4.

Diagrama de flujo del aprendizaje supervisado



Nota. Algoritmo de aprendizaje automático supervisado. Fuente: Elaboración Propia

Entre los algoritmos más representativos dentro de esta categoría se destacan los siguientes:

- **Regresión Lineal y Logística:** La regresión lineal se emplea para predecir valores numéricos continuos, como la vida útil de un componente, mientras que la regresión logística se utiliza cuando la salida es una categoría, como determinar si un sistema fallará o no, estimando probabilidades y clasificando los datos en función del resultado más probable.

- **Árboles de Decisión:** Este algoritmo utiliza una estructura similar a un árbol ramificado para tomar decisiones, a medida que avanza por cada nodo, se hacen preguntas basadas en los valores de las variables dividiendo los datos en grupos más homogéneos hasta alcanzar una conclusión. Son especialmente valorados por su transparencia y facilidad de comprensión.

- **Máquinas de Vectores de Soporte (SVM):** Son modelos que actúan encontrando la mejor forma de separar los datos en distintas clases, identificando un límite óptimo que maximiza la distancia entre los puntos más cercanos de cada grupo. Mediante el uso de funciones kernel, pueden proyectar los datos a espacios de mayor dimensión, lo que les permite detectar patrones complejos y lograr una clasificación más precisa.

- **Redes Neuronales Artificiales (ANN):** Son modelos inspirados en el funcionamiento del cerebro humano, compuestos por nodos llamados “neuronas” distribuidos en capas, son ampliamente utilizadas en tareas como reconocimiento de imágenes, predicción de series temporales y monitoreo de condiciones operativas por su alto nivel de abstracción.

- **K-Vecinos más Cercanos (K-NN):** Este algoritmo asigna una clase a una nueva observación basándose en los K ejemplos más cercanos en el conjunto de

entrenamiento, calculando distancias como la euclidiana, es un método efectivo cuando los datos están bien definidos, sin embargo, su rendimiento depende directamente de la elección adecuada de K y de la calidad de los datos utilizados.

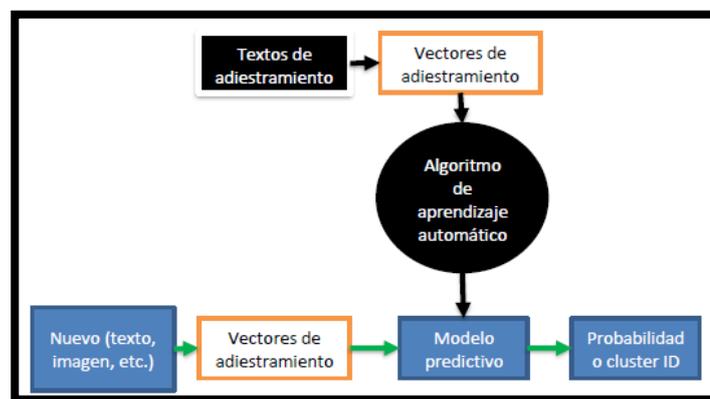
- **Naïve Bayes:** Se trata de un clasificador probabilístico basado en el teorema de Bayes, que asume independencia entre las variables predictoras, a pesar de esta suposición simplificadora, su rendimiento es notable en problemas de clasificación con datos categóricos, especialmente cuando se requiere una solución rápida y eficiente. Es comúnmente utilizado en aplicaciones como filtrado de spam, análisis de sentimientos o categorización de documentos

2.2.5.2.El aprendizaje no supervisado

Este tipo de aprendizaje parte de conjuntos de datos sin etiquetas. El objetivo no es predecir un resultado específico, sino descubrir patrones internos, estructuras o agrupaciones naturales dentro de los datos. Es ideal para detectar comportamientos atípicos, segmentar productos o identificar nuevos perfiles de uso sin necesidad de tener una respuesta correcta predefinida (Cáceres Tello, 2022). La figura 5 ilustra de manera esquemática el flujo de funcionamiento del algoritmo, destacando cada una de las etapas clave del proceso de aprendizaje y toma de decisiones.

Figura 5.

Diagrama de flujo del aprendizaje no supervisado



Nota. Algoritmo de aprendizaje automático no supervisado. Fuente: Elaboración Propia

Entre las técnicas más utilizadas dentro de esta categoría se destacan las siguientes:

- **K-means Clustering:** Se caracteriza por organizar los datos en un número definido de grupos (K) basándose en la cercanía entre ellos. Cada dato se asigna al grupo cuyo centro esté más próximo, lo que lo convierte en una herramienta útil para identificar patrones de comportamiento similares, por ejemplo, en el funcionamiento de máquinas o el consumo energético.

- **Clustering Jerárquico:** Está encargado de construir agrupaciones de forma progresiva, generando una estructura de tipo árbol en la que los datos se van uniendo según su similitud, además ayuda a visualizar relaciones jerárquicas entre componentes mediante métodos como single link, average link o complete link.

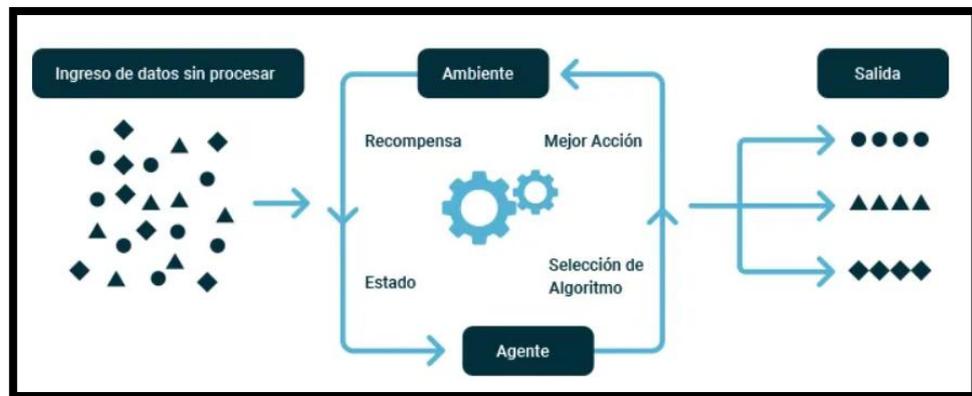
- **Mezclas Gaussianas (GMM):** Este modelado estadístico asume que los datos provienen de una combinación de múltiples distribuciones normales (gaussianas), cada una representando un subgrupo dentro del conjunto. A diferencia de métodos más rígidos como K-Means, las GMM forma clústeres con formas elípticas y tamaños variables, lo que las hace especialmente útiles en donde las agrupaciones no son claramente separables ni simétricas.

2.2.5.3.El aprendizaje de refuerzo (razonamiento):

El aprendizaje por refuerzo simula un proceso de prueba y error, donde un agente toma decisiones dentro de un entorno dinámico, recibe retroalimentación en forma de recompensas o penalizaciones, y ajusta su comportamiento con el fin de maximizar sus beneficios a largo plazo. Este tipo de entrenamiento se utiliza especialmente en tareas secuenciales, como el control de robots, planificación logística o estrategias de mantenimiento autónomo (Luna Gonzalez, 2024). La figura 6 representa este esquema.

Figura 6.

Diagrama de flujo del aprendizaje por refuerzo



Nota. Algoritmo de aprendizaje automático por refuerzo. Fuente: (Luna Gonzalez, 2024)

El aprendizaje por refuerzo no necesita datos etiquetados ni busca patrones estáticos, sino que aprende de la experiencia en tiempo real. En la industria, su aplicación permite que sistemas inteligentes decidan cuándo intervenir en un proceso para evitar sobrecostos o daños mayores. A continuación, se muestra algunos de los algoritmos más conocidos:

- **Q-Learning:** Es un algoritmo basado en recompensas que aprende la mejor acción para cada estado sin conocer el entorno. Usa una tabla Q que se actualiza con la experiencia. Es útil en IoT para optimizar redes y ubicaciones de sensores.

- **Deep Q-Network (DQN):** Se basa en el modelo Q-Learning, sin embargo, marca su diferencia al usar redes neuronales para estimar valores Q, lo que da la oportunidad de manejar entornos complejos con muchos estados; es ideal para sistemas industriales avanzados con datos dinámicos.

- **SARSA:** algoritmo que estima valores de acción mediante actualizaciones basadas en la secuencia, ajusta su política utilizando las acciones realmente tomadas, lo que lo hace dependiente del comportamiento del agente en cada paso.

- **Proximal Policy Optimization (PPO):** Es un algoritmo moderno que mejora el rendimiento manteniendo estabilidad en el aprendizaje. Muy usado en sistemas inteligentes industriales por su eficiencia en entornos variables.

2.2.6. Machine Learning aplicado a la detección de fallas

El uso de Machine facilita la detección temprana de fallas mediante el análisis de grandes volúmenes de datos generados por sensores, al identificar patrones históricos y comportamientos anómalos estos algoritmos permiten reducir tiempos de inactividad, mejorar la eficiencia del sistema y prolongar la vida útil de los equipos.

Una de las principales ventajas del ML es su capacidad para trabajar con datos en tiempo real y ajustar sus modelos dinámicamente, ofreciendo diagnósticos precisos y adaptativos bajo condiciones operativas cambiantes. Esto es posible gracias a la combinación de redes neuronales, métodos de entrenamiento supervisado y no supervisado que anticipan fallas inminentes y activan mantenimientos programados de manera inteligente (Lawrence, 2024).

Una de sus principales ventajas es que mejora la gestión del mantenimiento, ya que al intervenir de forma focalizada reduce mantenimientos innecesarios y costos operativos, la fiabilidad del sistema depende del acceso a datos consistentes y del uso de plataformas robustas como Edge Computing y cloud services, que permiten ejecutar modelos preentrenados en tiempo real (Benhanifia et al., 2025).

2.2.7. Big Data

En la actualidad, los datos representan un recurso estratégico para las organizaciones, específicamente en los entornos industriales, donde sensores, dispositivos inteligentes y sistemas digitales generan grandes volúmenes de información de forma continua; este fenómeno da lugar a la Big Data, que engloba no solo la cantidad de datos producidos, sino

también la velocidad con la que se generan, su diversidad estructural y la complejidad que implica analizarlos.

Lo relevante del Big Data es que a través de técnicas avanzadas de análisis y visualización se pueden detectar patrones de comportamiento, prever situaciones críticas, optimizar recursos y tomar decisiones más acertadas; en el caso de la industria, esto puede traducirse en una mejora significativa de la productividad, la calidad y la prevención de fallos en sistemas y equipos (Berisha et al., 2022).

2.2.8. *Cloud Computing*

La computación en la nube, conocida como *Cloud Computing*, ha redefinido el acceso y gestión de los recursos tecnológicos, a diferencia del modelo tradicional basado en infraestructura local que implicaba elevados costos y mantenimiento constante; esta tecnología ha sido especialmente útil para el procesamiento de grandes volúmenes de datos generados por dispositivos IoT y sistemas inteligentes.

La nube es una herramienta para que esa información sea analizada de forma rápida y segura, facilitando la automatización de procesos, la integración con inteligencia artificial y la toma de decisiones en tiempo real. Además, con soluciones como el procesamiento en el borde (*Edge Computing*), parte de los datos se pueden gestionar directamente en el lugar donde se generan, reduciendo los tiempos de espera y mejorando la eficiencia. En definitiva, Cloud Computing se ha convertido en una base fundamental para el desarrollo de entornos industriales más ágiles, conectados e inteligentes (Yang et al., 2017)

2.2.9. *Máquinas de vectores de regresión*

Las Máquinas de Vectores de Regresión (SVR, por su sigla en inglés) son una técnica derivada de los modelos de soporte vectorial que permiten realizar predicciones numéricas con gran precisión. A diferencia de su variante clásica usada para clasificación, las SVR se enfocan en encontrar una función que aproxime de forma óptima la relación entre variables

de entrada y salida dentro de un margen de tolerancia determinado. Lo interesante de esto es que no todos los datos influyen en la predicción final, sino solo aquellos que quedan fuera del margen de error permitido, consiguiendo mayor eficiencia y menor complejidad computacional.

Esta técnica se apoya en funciones núcleo (kernel) como la Gaussiana o la radial (RBF), mismas que modelan relaciones no lineales sin necesidad de transformar los datos manualmente. Gracias a esta capacidad las SVR se utilizan con frecuencia en la industria, especialmente valorado en mantenimiento predictivo, donde es necesario anticipar valores como la vida útil restante de una máquina, la velocidad de desgaste o el consumo energético. (Argumedo Rhenals & León Herrera, 2024)

2.2.10. Detección de fallas

Consiste en identificar patrones inusuales en un flujo de datos que se desvían del comportamiento estándar para detectar una desviación temprana, este tipo de análisis suele clasificarse en tres tipos: anomalías puntuales, contextuales y secuenciales, según la forma de desviación respecto al patrón esperado. Al colocarse dentro de la detección de fallas, estos modelos monitorean continuamente los datos del sensor y alertan ante el primer indicio de comportamiento anómalo, reduciendo la dependencia de monitoreo manual y consiguiendo acciones preventivas inmediatas (Zamanzadeh Darban & Webb, 2024) .

CAPÍTULO III

DISEÑO METODOLÓGICO

3.1. Enfoque de la Investigación

El presente estudio presenta un enfoque cuantitativo, sustentado en la recolección, procesamiento y análisis estadístico de datos sensoriales obtenidos de un motor eléctrico sometido a distintas condiciones operativas. Este enfoque da paso a la evaluación del rendimiento de diversos algoritmos de aprendizaje automático, mediante indicadores estadísticos derivados de matrices de confusión y otras métricas de validación. Se emplea herramientas computacionales propias de la matemática aplicada y el análisis predictivo, lo cual refuerza la naturaleza cuantitativa del trabajo.

3.2. Diseño Metodológico

El diseño metodológico corresponde a un estudio cuasi-experimental en un entorno parcialmente controlado. Si bien se ejerció manipulación directa sobre las condiciones operativas del motor eléctrico para inducir distintos estados funcionales, esta manipulación se desarrolló dentro de los límites establecidos por la entidad que facilitó el equipo, con quien se acordaron condiciones específicas de uso.

Se programó sesiones para la recolección de datos de una hora diaria entre las 20h00 y 21h00 durante 14 días conforme a lo estipulado con el prestamista del equipo; al mantener constantes tanto el horario como el entorno de trabajo aportó consistencia a las mediciones, favoreciendo la calidad de la información recolectada.

3.3. Tipo y Nivel de investigación

La investigación es de tipo aplicada, dado que no busca únicamente ampliar el conocimiento teórico sobre el aprendizaje automático o el mantenimiento industrial, sino implementar soluciones concretas y funcionales para una problemática real: la detección

anticipada de fallas mecánicas. El resultado esperado es directamente transferible al entorno industrial, contribuyendo a la mejora continua.

En cuanto a su nivel, el estudio es descriptivo-comparativo, ya que describe el comportamiento de las variables sensoriales registradas durante el funcionamiento del motor eléctrico y compara el desempeño de diversos algoritmos de Machine Learning aplicados a la predicción de fallas.

La evaluación comparativa de los modelos se realiza a través de un análisis estadístico centrado en matrices de confusión y sus métricas, las cuales cuantifican el rendimiento de cada modelo mediante indicadores como: la precisión, sensibilidad y especificidad, parámetros fundamentales para valorar de forma integral la capacidad de cada algoritmo en la predicción temprana de fallas.

3.4. Recolección de Datos

La recolección de datos se realiza mediante un sistema de monitoreo en tiempo real aplicado a un motor eléctrico y sensores ubicados estratégicamente para medir corriente, temperatura, vibración y RPM, garantizando la eficiencia y exactitud de los registros con el objetivo de capturar el funcionamiento normal como señales tempranas de fallas asegurando datos confiables para entrenar modelos de aprendizaje automático en entornos industriales.

3.5. Procesamiento e Interpretación de Datos

La información capturada es sometida a un proceso de preparación que incluye limpieza de valores atípicos, normalización de escalas y segmentación de registros. Luego, se procede a entrenar los modelos seleccionados utilizando Matlab como entorno de desarrollo. Cada algoritmo es evaluado mediante matriz de confusión y métricas, la comparación de estos indicadores da paso a identificar el mejor modelo y comprender su comportamiento frente a datos reales del entorno.

3.6. Población y Muestra

La población de este proyecto de investigación está conformada por el total de datos registrados durante las sesiones de monitoreo del motor eléctrico entre las 20h00 y 21h00 durante 14 días consecutivos, con la exigencia de capturar una medición por segundo; requerimientos planteados por la entidad que proporcionó el equipo. Las condiciones de uso del equipo, los horarios y los protocolos de monitoreo fueron establecidos de común acuerdo, garantizando transparencia y acato mutuo.

3.7. Procedimientos éticos

El presente trabajo se desarrolla respetando los principios éticos de la investigación científica e integridad académica, para respaldar este compromiso, se han considerado cuidadosamente los siguientes aspectos:

- Se cuenta con la autorización de los responsables del equipo motriz, quienes además manifestaron interés en los resultados para fines de análisis técnico externo.
- Todos los datos fueron tratados con confidencialidad y utilizados exclusivamente con fines investigativos, sin comprometer propiedad intelectual ni divulgar información sensible.
- En la implementación de los algoritmos se emplearon plataformas de código abierto como Matlab, respetando las licencias de uso correspondientes.
- Toda fuente bibliográfica y técnica consultada fue debidamente referenciada, conforme a principios de ética investigativa y reconocimiento del trabajo de otros autores.

3.8. Planificación

El tiempo estimado para la ejecución del proyecto se ha estructurado en etapas consecutivas, considerando la disponibilidad de los recursos necesarios y una planificación realista de las actividades.

Para el diseño y construcción del banco de pruebas contempla un período de aproximadamente tres semanas, tiempo en el cual se ensamblan los componentes y se valida el correcto funcionamiento del sistema.

Para la adquisición de datos sensoriales, proceso que incluye la conexión de sensores, registro continuo de variables y conformación de una base de datos local organizada se destinan 3 semanas más, en paralelo, se contempla el mismo período para la revisión bibliográfica, centrada en el análisis del estado del arte sobre mantenimiento predictivo y el uso de algoritmos de aprendizaje automático aplicados a motores eléctricos.

En el lapso de 2 semanas se prevé realizar la implementación y ajuste de los modelos de aprendizaje automático, sin embargo, para cubrir imprevistos o complicaciones técnicas, se reserva un margen adicional de dos semanas, destinado a resolución de fallos, ajustes o correcciones de carácter metodológico o técnico.

Se considera un período de tres semanas para la redacción del documento final, incluyendo la sistematización de resultados, conclusiones y elaboración de anexos técnicos.

3.9. Presupuesto

La ejecución del proyecto requiere recursos técnicos esenciales para garantizar la recolección confiable de datos, su procesamiento computacional y la adecuada presentación de resultados. El presupuesto contempla elementos mínimos pero estratégicos, como sensores, microcontroladores, materiales de instalación, conectividad y servicios básicos, necesarios para el funcionamiento integral del sistema de monitoreo y análisis. La planificación económica considera los requerimientos técnicos y las restricciones logísticas, priorizando la eficiencia sin comprometer la calidad del estudio.

Tabla 1.*Presupuesto*

Rubro	Descripción	Costo (USD)	Unid.	Total (USD)
Acelerómetro MPU-6050	Sensores de adquisición	\$5,00	2	\$10,00
Sensor SCT-013		\$10,00	2	\$20,00
Encoder FC-03		\$7,00	2	\$14,00
Sensor MLX90614		\$10,00	2	\$20,00
Microcontrolador (Arduino Uno)	Interfaz de adquisición	\$15,00	2	\$30,00
Materiales de instalación	Cableado, conectores, bases y elementos de fijación	\$15,00	1	\$15,00
Energía eléctrica	Consumo estimado durante 14 jornadas de prueba (1 hora diaria)	\$1,74	14	\$24,36
Impresiones	Documento Físico	\$10,00	1	\$10,00
Motor eléctrico	Equipo facilitado sin costo por la empresa con fines investigativos	\$0,00	1	\$0,00
Software	Matlab (Código abierto)	\$0,00	1	\$0,00
Biblioteca virtual UNACH	Acceso a recursos bibliográficos digitales de forma gratuita	\$0,00	1	\$0,00
Internet y conectividad	Conexión utilizada para análisis en la nube y búsqueda académica	\$27,00	1	\$27,00
Equipo de cómputo	Laptop propia utilizada para procesamiento y simulación	\$1000,00	1	\$1000,00
Transporte ocasional	Traslados para instalación, seguimiento y desmontaje de los sensores	\$15,00	1	\$15,00
Total Estimado				\$1185,36

Nota. Esta tabla muestra el presupuesto del presente proyecto. Fuente: Elaboración Propia

Como se detalla en la Tabla 1, el costo total estimado asciende a 1185,36 dólares, se puede identificar que el mayor rubro corresponde a la adquisición de sensores, la interfaz de captura de datos y el soporte computacional, sin embargo, la disponibilidad del motor eléctrico por parte de la empresa, así como el acceso gratuito a recursos institucionales como la biblioteca virtual de la UNACH y el uso de software libre, constituyen factores determinantes para disminuir la inversión. El presupuesto plasmado evidencia una planificación técnica, eficiente y coherente con los principios de sostenibilidad, aprovechamiento de recursos y viabilidad investigativa.

CAPÍTULO IV

ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS

4.1. Fases para la implementación

Para implementar los algoritmos de Machine Learning seleccionados, se seguirá una serie de fases estructuradas para garantizar que la adquisición de datos, su procesamiento y análisis respondan a criterios de calidad y precisión. A continuación, se detallan las etapas desarrolladas en esta investigación:

- **Obtención de datos sensoriales:** Esta fase inicia con la selección de sensores adecuados según los parámetros críticos del motor (temperatura, corriente, vibración y velocidad). Cada sensor se elige considerando su precisión, tiempo de respuesta, compatibilidad eléctrica y facilidad de integración.

- **Diseño e implementación del sistema de adquisición de datos y banco de pruebas:** Para llevar a cabo la experimentación, se diseña previamente en AutoCAD un banco de pruebas que acopla el motor eléctrico y los sensores en ubicaciones estratégicas para asegurar la estabilidad operativa y la respuesta mecánica del motor, permitiendo registrar datos en tiempo real durante distintas condiciones de carga.

- **Selección de algoritmos de Machine Learning:** Una vez recolectados los datos, se procede a la selección de algoritmos de Machine Learning, en base a la diversidad y complejidad de los datos registrados por los sensores, aquí se consideran modelos supervisados y no supervisado, con el fin de cubrir necesidades de clasificación cuando se cuenta con etiquetas conocidas, y de agrupamiento cuando se requiere identificar patrones ocultos sin referencia previa.

- **Recolección de datos:** Una vez instalado el sistema, se ejecutan diversas pruebas con diferentes escenarios operativos, incluyendo cargas ligeras, moderadas y pesadas. El

objetivo fue construir un conjunto de datos que refleje la realidad funcional del equipo, considerando variaciones naturales en sus condiciones de trabajo.

- **Procesamiento de datos:** Se realiza un preanálisis de los datos recolectados, organizándolos y adaptándolos al formato requerido por los algoritmos de aprendizaje automático. Durante esta fase, se aplican algoritmos que limpian la base de datos para garantizar coherencia, compatibilidad y calidad estructural de la información.

- **Entrenamiento de algoritmos:** Con los datos ya depurados, se entrena cada modelo de Machine Learning, identificando comportamientos normales y anómalos del motor.

- **Evaluación de algoritmos:** El rendimiento de los algoritmos son valorados mediante matrices de confusión y sus métricas para verificar la capacidad de cada algoritmo, predecir correctamente los estados del sistema y detectar fallas con la mayor precisión posible.

- **Selección del algoritmo óptimo:** Para determinar el modelo más eficiente, se calcula el promedio del desempeño de cada algoritmo en todas las variables sensoriales; aquel que alcanza el valor más alto es seleccionado por su rendimiento global y su consistencia en diferentes condiciones operativas del sistema.

4.2. Obtención de datos sensoriales

Para asegurar un análisis confiable en la predicción de fallas en motores eléctricos mediante algoritmos de Machine Learning se implementa un entorno de monitoreo que registra datos en tiempo real a través de sesiones de prueba, durante cada una se miden variables como: temperatura, corriente, vibración y velocidad. Dado que esta información constituye la base para entrenar y validar los modelos, la ubicación de los sensores es esencial para garantizar la calidad y utilidad de los datos recolectados.

La recolección de datos captura la evolución temporal de cada variable, lo que permite a los modelos de aprendizaje identificar patrones sutiles que preceden a una condición de falla. Para ello, se emplea un microcontrolador Arduino Uno, seleccionado por su versatilidad, bajo costo, facilidad de programación y amplia compatibilidad con sensores industriales. La información es almacenada en archivos de formato *.csv*, lo que facilita su posterior procesamiento y análisis en entornos de Machine Learning.

4.2.1. Selección de sensores

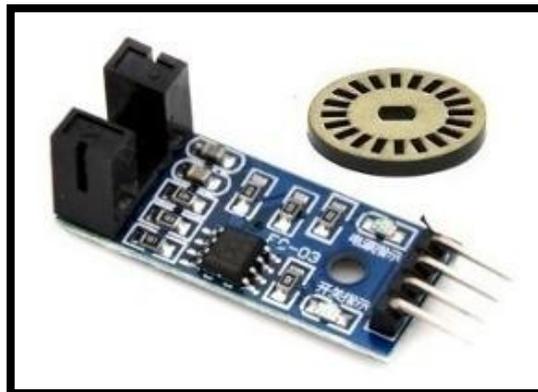
La selección de sensores se realiza considerando factores como confiabilidad, sensibilidad frente a las variables críticas del motor, facilidad de integración al sistema y disponibilidad en el mercado, cabe recalcar que la calidad de estos dispositivos resulta determinante, ya que cualquier error en la captura de datos puede derivar en predicciones erróneas afectando la operación normal del sistema o generando intervenciones innecesarias.

A continuación, se detallan los sensores utilizados para medir cada una de las variables sensoriales seleccionadas en el estudio:

4.2.1.1. Sensor de Velocidad: Encoder FC-03

Figura 7.

Sensor Encoder FC-03



Nota. Sensor óptico tipo encoder utilizado para medir la velocidad de rotación del motor.

Fuente: (Dutuour, 2024)

Para determinar la velocidad de giro del eje del motor, se empleó el sensor óptico tipo encoder FC-03, también conocido como sensor de RPM (ver figura 7), está compuesto por un opto interruptor con forma de "U" que detecta el paso de objetos a través de una ranura interna. Su funcionamiento se basa en la interrupción de un haz de luz infrarroja cuando una pieza con ranuras o marcas oscuras atraviesa el espacio entre el emisor y el receptor.

El encoder se complementa con un disco ranurado acoplado al eje del motor, el cual genera interrupciones periódicas del haz de luz a medida que gira. Cada interrupción produce un pulso eléctrico, formando un tren de pulsos, este tren de señales se interpreta mediante un microcontrolador para calcular revoluciones por minuto (RPM), vueltas o velocidad angular (Dutuour, 2024)

La elección de este sensor se debe a su capacidad para ofrecer una lectura precisa y confiable de la velocidad de rotación, cambios abruptos o irregulares en la frecuencia de los pulsos pueden ser indicativos de fallas mecánicas.

4.2.1.2. Sensor de temperatura: MLX90614

Figura 8.

Sensor MLX90614



Nota. Sensor infrarrojo MLX90614 utilizado para la medición de temperatura sin contacto físico. Fuente: (Llamas, 2025)

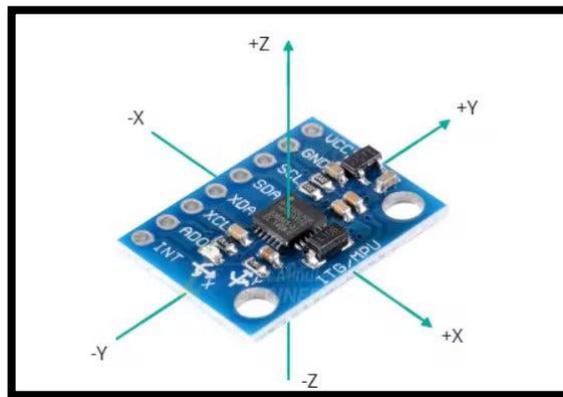
Para registrar la temperatura superficial del motor, se empleó el sensor infrarrojo MLX90614 (Ver figura 8), el cual mide la radiación térmica emitida por los objetos y la convierte en una señal eléctrica proporcional, con una alta resolución de 0,02 °C. Al poder operar sin contacto físico, este sensor evita alteraciones térmicas y minimiza los riesgos durante el monitoreo. Este dispositivo es útil en componentes móviles o de difícil acceso.

Su comunicación a través del protocolo digital I2C ofrece una ventaja significativa frente a sistemas analógicos, al proporcionar una mayor integridad en la transmisión de datos y una mejor resistencia frente a interferencias electromagnéticas, estas características lo convierten en una opción confiable para entornos industriales exigentes.

4.2.1.3. Acelerómetro MPU6050

Figura 9.

Acelerómetro MPU6050



Nota. Sensor inercial de 6 ejes utilizado para la detección de vibraciones. Fuente: (Lithium, 2022)

Para detectar vibraciones que puedan indicar posibles fallas mecánicas en el motor, se utilizó el sensor MPU6050 (ver figura 9), un módulo que integra en un solo chip un acelerómetro de tres ejes y un giroscopio de tres ejes. Esta combinación le

otorga seis grados de libertad, lo que ayuda a registrar la aceleración lineal y la rotación angular del dispositivo en el espacio en forma de vibraciones, mismas que pueden reflejar el estado de balance, el desgaste de rodamientos, o incluso desalineaciones mecánicas.

El MPU6050 es un sensor inercial del tipo IMU (Unidad de Medición Inercial) que combina acelerómetro y giroscopio en un solo dispositivo compacto., se comunica digitalmente mediante el protocolo I2C, lo que facilita su integración con plataformas de desarrollo como Arduino o Raspberry Pi. Gracias a su alta sensibilidad y baja latencia, es ampliamente utilizado en aplicaciones como navegación, robótica, estabilización de plataformas y control de movimiento en drones, donde se requiere un seguimiento preciso de la dinámica del sistema. (Lithium, 2022).

4.2.1.4. Sensor de corriente SCT-013

Figura 10.

Sensor de Corriente SCT-013



Nota. Sensor no invasivo SCT-013 utilizado para la medición de corriente alterna sin necesidad de cortar el conductor. Fuente: (Llamas, 2025)

Para registrar la intensidad de corriente que consume el motor durante su funcionamiento, se utilizó el sensor no invasivo SCT-013 (ver figura 10). Este

dispositivo está diseñado para medir corriente alterna sin necesidad de cortar ni modificar el cable por el que circula la electricidad, lo cual garantiza una instalación rápida y segura (Llamas, 2025)

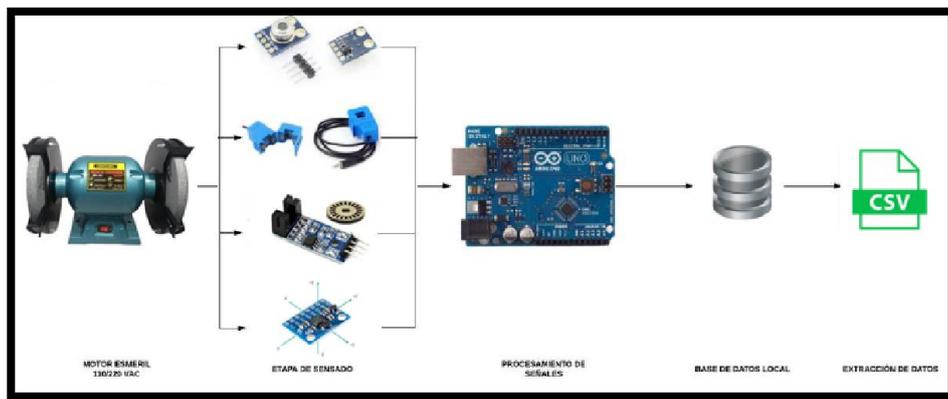
El SCT-013 funciona como un transformador de corriente con núcleo cerrado. Al detectar el campo magnético generado por el paso de corriente, genera una señal de salida proporcional que puede ser leída por un microcontrolador. Esta señal, una vez acondicionada, se convierte en una entrada útil para sistemas de adquisición.

4.3. Diseño del sistema de adquisición de datos sensoriales y banco de pruebas

El sistema de adquisición de datos se desarrolla bajo una arquitectura sencilla basada en una placa de desarrollo Arduino Uno, la cual se encarga de recibir, procesar y organizar las señales emitidas por los diferentes sensores conectados. En la Figura 11 se presenta un esquema que ilustra la interconexión de estos componentes dentro del sistema.

Figura 11.

Esquema General



Nota. Representación de la arquitectura del sistema de adquisición de datos. Fuente: Elaboración Propia

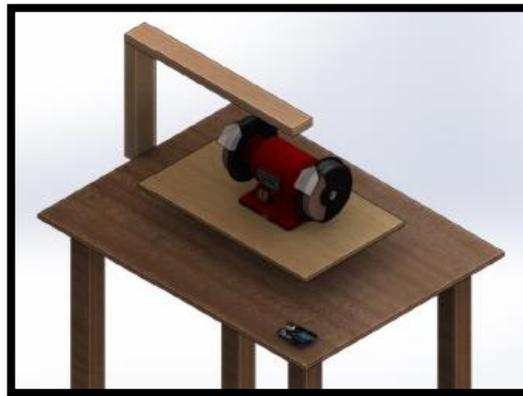
Los datos adquiridos se guardan en archivos con formato .csv, lo que permite organizarlos y utilizarlos de manera sencilla en entornos de análisis y entrenamiento de

modelos de Machine Learning, siendo compatible con plataformas de procesamiento como Python, MATLAB o Excel, facilitando su posterior tratamiento.

Como parte complementaria del proyecto, se diseñó e implementó un banco de pruebas destinado a evaluar el desempeño del sistema de monitoreo en condiciones reales de operación. Se elaboró un modelo digital del banco utilizando el software AutoCAD, (Figura 12) donde se definieron aspectos estructurales como las dimensiones, ubicación estratégica de los sensores y espacios dedicados para la fijación de componentes electrónico

Figura 12.

Diseño del banco de pruebas digital



Nota. Esquema del banco de pruebas desarrollado en AutoCAD. Fuente: Elaboración Propia

Se llevó a cabo la construcción física del banco de pruebas siguiendo las especificaciones detalladas en el diseño digital. El montaje incluyó una base estructural estable, la instalación del motor esmeril y la integración precisa de los sensores en ubicaciones previamente determinadas (Figura 13). Se emplearon materiales que brindaran rigidez y resistencia, garantizando estabilidad durante las pruebas. Además, se tuvo especial atención en minimizar cualquier tipo de interferencia mecánica o eléctrica que pudiera comprometer la calidad de las mediciones, asegurando condiciones adecuadas para el monitoreo continuo del equipo bajo diversos escenarios.

Figura 13.

Implementación del banco de pruebas



Nota. Banco de pruebas ensamblado con sensores. Fuente: Elaboración Propia

El motor seleccionado para las pruebas fue un Bench Grinder modelo MD3215K, cuyas especificaciones técnicas se detallan en la figura 14. Este equipo opera a una velocidad nominal de 3450 RPM, con una potencia de 200W y tensión de 110V, parámetros que fueron determinantes para establecer los rangos de funcionamiento del sistema. Estas características también sirvieron de guía para la correcta elección de los sensores y la configuración del sistema de adquisición de datos.

Figura 14.

Parámetros del esmeril



Nota. Datos técnicos del motor esmeril utilizado para el estudio. Fuente: Elaboración Propia

En función de estas especificaciones, el banco fue diseñado para simular condiciones operativas reales, permitiendo someter al motor a distintos niveles de carga mecánica. Esta

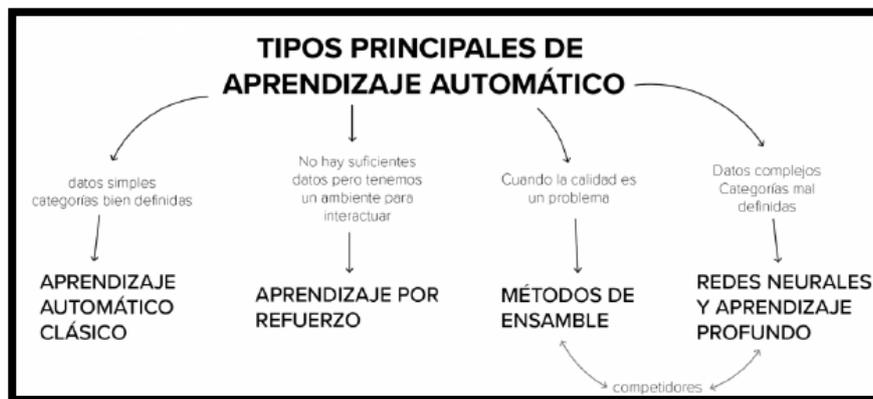
versatilidad facilitó la recolección de información clave sobre los parámetros de funcionamiento, con el objetivo de identificar fallas mediante el análisis de anomalías en señales como corriente, temperatura, vibraciones y velocidad angular. La capacidad de replicar comportamientos anómalos resultó clave para validar la efectividad de los algoritmos de Machine Learning en la detección temprana de eventos críticos. Machine Learning.

4.4. Selección de algoritmos de machine Learning

Una vez completada la etapa del diseño de adquisición de datos sensoriales y la implementación del banco de pruebas, se realizó la selección de los algoritmos de aprendizaje automático tomando en cuenta dos factores: la disponibilidad de datos previamente etiquetados y a necesidad de identificar comportamientos que no son evidentes a simple vista.

Figura 15.

Tipos de Aprendizaje Automático



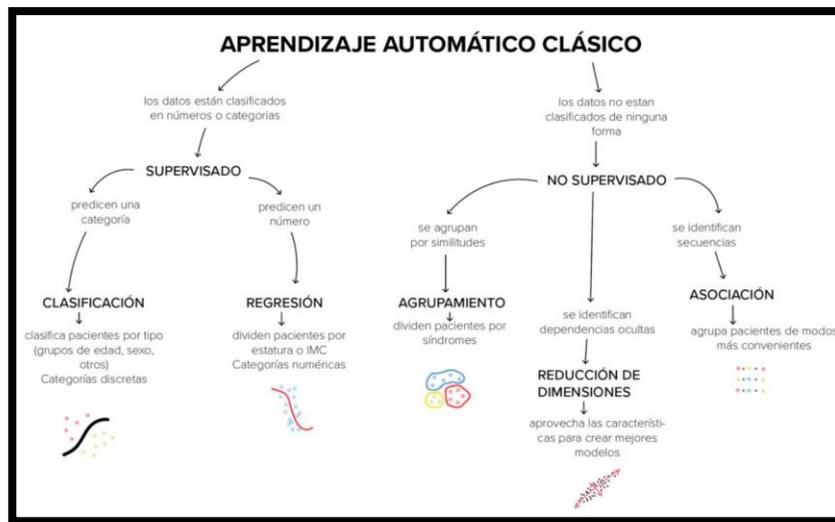
Nota. Clasificación general de los tipos de aprendizaje automático. Fuente: (Barrios Arce, 2022a)

Considerando que los datos recopilados eran estructurados, con variables bien definidas y categorías de salidas claras (estado normal o en falla), se optó por aplicar aprendizaje automático clásico, como se muestra en la figura 15.

La presencia de registros etiquetados dio paso a la implementación de algoritmos supervisados y redes neuronales, sin embargo, se complementó el análisis con técnicas no supervisadas para explorar agrupamientos naturales en los datos y mejorar la capacidad de generalización del sistema. Esta combinación se representa en la figura 16, que muestra la clasificación de técnicas clásicas, diferenciando entre supervisados y no supervisados.

Figura 16.

Aprendizaje Automático Clásico



Nota. Clasificación de las técnicas clásicas de aprendizaje automático. Fuente: (Barrios Arce, 2022a)

En base a este análisis, y alineado con los objetivos de predicción de fallas, detección de estados anómalos y segmentación de comportamientos operativos, se seleccionaron los siguientes algoritmos:

- **Árboles de Decisión:** Seleccionados por su capacidad para generar reglas de clasificación explícitas y fácilmente interpretables, facilitando la comprensión del modelo, además porque es indispensable para identificar variables sensoriales claves asociadas a fallas, aportando transparencia en la toma de decisiones diagnósticas.
- **Máquinas de Soporte Vectorial (SVM):** Elegidas por su alto rendimiento en tareas de clasificación binaria y por su eficacia al trabajar con datos que no presentan una separación lineal evidente, además porque diferencia con precisión entre estados normales

y defectuosos del motor, analizando de forma simultánea varias variables sensoriales que describen su comportamiento operativo.

- **Naive Bayes:** Este algoritmo se selecciona por su eficiencia computacional, su tolerancia al ruido y porque realiza una clasificación binaria de manera ágil, basándose en la hipótesis de independencia condicional entre las variables, cabe recalcar que su enfoque probabilístico, es capaz de detectar patrones de falla con rapidez y una razonable precisión, convirtiéndose en una herramienta útil para el análisis preliminar de estados operativos del motor.

- **K-means (Clustering):** Es un modelo no supervisado para segmentar datos sin necesidad de etiquetas, clave en la etapa exploratoria del estudio al permitir la detección de agrupamientos naturales y comportamientos atípicos asociados a condiciones anómalas del motor.

- **Redes Neuronales Artificiales:** Seleccionadas por su capacidad para aprender relaciones no lineales complejas entre variables multivariadas, su inclusión fortalece el análisis al modelar dinámicas internas del sistema que no pueden ser capturadas fácilmente por los otros métodos.

4.5. Recolección de Datos

Para la fase de recolección de datos, los sensores se ubican previamente en puntos clave del sistema, con el fin de monitorear variables críticas. Una vez implementado, se registra en tiempo real las señales emitidas por los sensores instalados sobre un motor esmeril que opera bajo distintas condiciones controladas.

El sistema se configura para registrar una muestra por segundo durante una hora continua de operación. Esta frecuencia de muestreo se establece como óptima, ya que permite observar el comportamiento dinámico del sistema sin generar un gran volumen de datos, lo cual facilita el almacenamiento, procesamiento y análisis posterior.

El motor es evaluado bajo cinco condiciones operativas distintas que simulan escenarios mecánicos reales, desde el estado sin carga hasta situaciones de falla provocadas por restricciones o desalineaciones en el eje. Cada configuración se detalla en la Tabla 2.

Tabla 2.

Configuraciones operativas del motor eléctrico

Nº	Configuraciones	Descripción
1	Estado base	Operación sin alteraciones externas. Representa el funcionamiento normal del equipo sin carga aplicada.
2	Carga estándar	El motor trabaja con una piedra abrasiva de 3 mm, simulando una carga de trabajo común.
3	Sobrecarga mecánica	Se acoplan dos piedras abrasivas de 3 mm, incrementando la carga para simular una exigencia operativa superior.
4	Eje restringido	Se instala un acoplamiento metálico que limita el giro del eje, representando una falla por bloqueo o interferencia mecánica.
5	Movimiento parcial del eje	El eje opera con un ajuste mecánico flojo, simulando una condición de desalineación o desgaste en el acoplamiento.

Nota. La tabla detalla las condiciones operativas a las que fue sometido el esmeril eléctrico durante la recolección de datos sensoriales. Fuente: Elaboración propia.

Las señales obtenidas fueron digitalizadas por una placa Arduino Uno y almacenadas en una tarjeta microSD en formato .csv, organizado por columnas según cada variable medida.

Para construir una base de datos representativa del entrenamiento de los algoritmos, se realiza el monitoreo continuo durante 14 días, en los primeros 7 días el motor eléctrico opera bajo condiciones normales sin modificaciones ni alteraciones mecánicas, durante los siguientes 7 días se introducen fallas simuladas de forma controlada, con el propósito de registrar el comportamiento del sistema ante condiciones anómalas y enriquecer así la variedad de escenarios dentro del conjunto de datos. Cada día de medición generó un archivo independiente, manteniendo la misma frecuencia y duración de muestreo, con un promedio de 3600 registros por sensor.

Los datos fueron organizados por variable y por estado del sistema (normal/falla), lo que ayuda a generar conjuntos diferenciados para el análisis. A modo de ejemplo, la Tabla 3 presenta un fragmento representativo del conjunto de datos estructurado.

Tabla 3.

Extracto de datos registrados en formato .csv

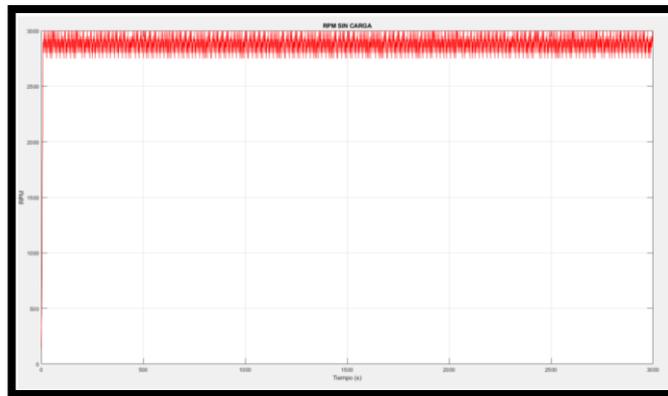
Fecha	Hora	Corriente (A)	Temperatura (°C)	Velocidad Angular (Rpm)	Vibraciones Eje X
24/2/2025	20:00:00	1,3	32,1	3200	225
24/2/2025	20:00:01	1,2	33,1	3250	221
24/2/2025	20:00:02	1,1	31,6	3200	223
24/2/2025	20:00:03	1,4	31,8	3225	224
24/2/2025	20:00:04	1,3	31,9	3195	221
24/2/2025	20:00:05	1,4	32,5	3185	223
24/2/2025	20:00:06	1,1	32,7	3200	225
24/2/2025	20:00:06	1,4	32,8	3195	221
24/2/2025	20:00:07	1,2	31,8	3190	223
24/2/2025	20:00:08	1,4	32,9	3185	222
24/2/2025	20:00:09	1,3	31,9	3195	221
24/2/2025	20:00:10	1,2	32,5	3200	223
24/2/2025	20:00:11	1,1	33,1	3225	225
24/2/2025	20:00:12	1,1	32,1	3195	224
24/2/2025	20:00:13	1,4	32,7	3185	223
24/2/2025	20:00:14	1,1	32,3	3200	221
24/2/2025	20:00:15	1,4	31,7	3195	223
24/2/2025	20:00:16	1,2	31,8	3190	223
24/2/2025	20:00:17	1,4	31,9	3200	223
24/2/2025	20:00:18	1,3	32,5	3225	221
24/2/2025	20:00:19	1,2	32,7	3195	221

Nota. Fragmento representativo de los datos sensoriales utilizados para el análisis. Fuente: Elaboración propia.

A continuación, se presentan los registros experimentales obtenidos del esmeril eléctrico operando sin carga, con el fin de caracterizar su comportamiento bajo condiciones normales de funcionamiento. Esta fase inicial permite establecer una línea base que servirá como referencia para detectar anomalías o desviaciones en escenarios posteriores. A continuación, se describe el comportamiento observado en cada figura:

Figura 17.

Registro de velocidad angular del esmeril eléctrico sin carga

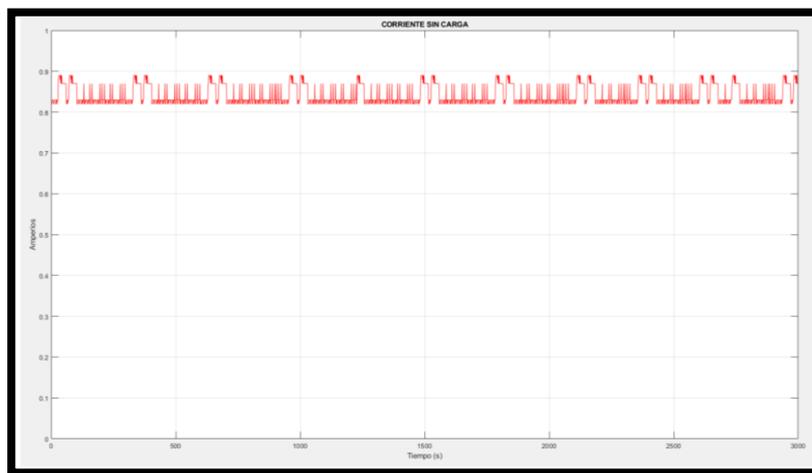


Nota. Registro base de velocidad angular bajo condiciones sin carga. Fuente: Elaboración Propia.

La figura 17 muestra una señal de velocidad angular elevada y estable, con ligeras oscilaciones periódicas, este comportamiento es característico de un motor operando sin carga mecánica, donde no se presentan perturbaciones significativas y el régimen de giro se mantiene cercano a su valor nominal.

Figura 18.

Registro de corriente del esmeril eléctrico sin carga



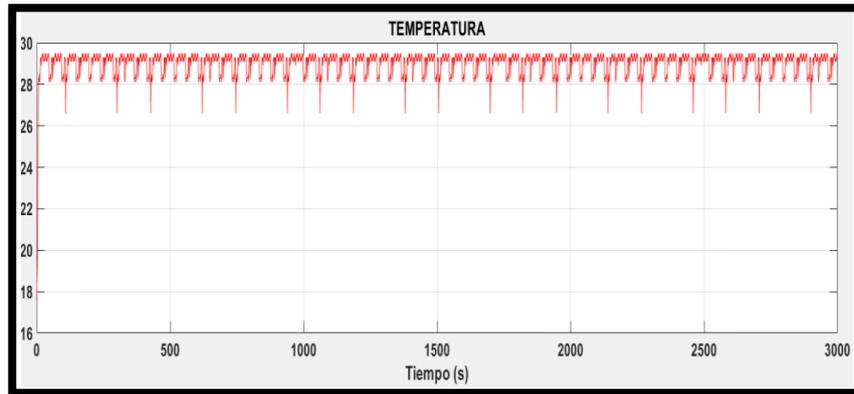
Nota. Registro de corriente eléctrica en estado operativo sin carga mecánica. Fuente: Elaboración Propia.

En la figura 18 se observa una señal de corriente con pequeñas fluctuaciones regulares. El patrón refleja un consumo eléctrico constante, correspondiente al

mantenimiento del funcionamiento básico del motor. No se evidencian picos anómalos ni sobrecargas, lo que indica un estado operativo adecuado.

Figura 19.

Registro de temperatura del esmeril eléctrico sin carga

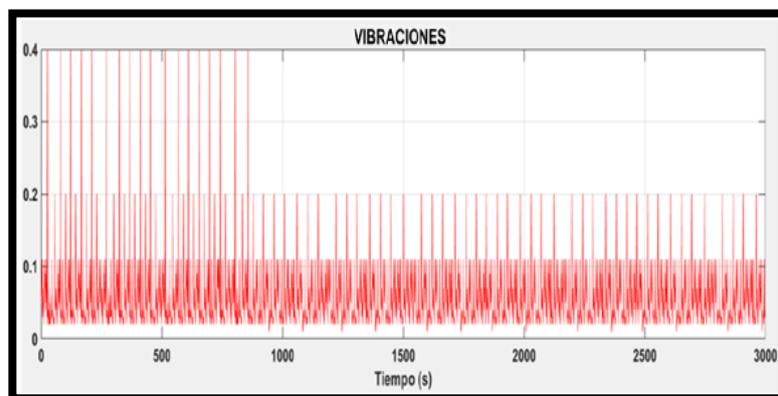


Nota. Evolución térmica de la carcasa del motor durante operación estándar sin carga. Fuente: Elaboración Propia.

En la figura 19 se puede analizar que la temperatura se mantiene dentro de un rango térmico controlado, con oscilaciones suaves posiblemente asociadas a condiciones ambientales o variaciones naturales del sistema; por lo cual no se detectan incrementos progresivos ni comportamientos que sugieran sobrecalentamiento.

Figura 20.

Registro de vibraciones del esmeril eléctrico sin carga



Nota. Patrón de vibración en condiciones normales, sin defectos ni carga externa. Fuente: Elaboración Propia.

En la figura 20 se observa como la señal de vibraciones presenta un patrón periódico con amplitudes moderadas y consistentes, típicas de un motor en equilibrio., se puede indicar que no se detectan impulsos irregulares ni cambios bruscos que puedan asociarse a fallas mecánicas, desalineaciones o desequilibrios en la estructura del motor.

4.6. Entrenamiento de Machine Learning

Una vez estructurados los datos sensoriales obtenidos del sistema de monitoreo, se procede al entrenamiento de modelos de aprendizaje automático orientados a la detección temprana de fallas en el esmeril eléctrico. Se emplean técnicas supervisadas y no supervisadas, evaluando el rendimiento de los modelos en tareas de clasificación binaria (condición normal/falla) a partir de las variables registradas.

El software que se decidió utilizar es MATLAB en su versión estudiantil, por su capacidad para el análisis numérico, visualización de datos y modelado predictivo, sin requerir licencias adicionales; en este entorno los datos se organizan por variable y por condición operativa, almacenándose en archivos independientes según el estado del motor (funcionamiento normal o con falla) y etiquetándose con valores binarios, donde 0 representa muestras sin falla y 1 corresponde a condiciones anómalas, lo cual permite construir conjuntos de entrenamiento específicos.

Se aplica una mezcla aleatoria de los registros para evitar sesgos y asegurar que el modelo no dependa del orden en que fueron recolectados, y finalmente, toda la información se estructura en matrices compatibles con las funciones de entrenamiento del propio MATLAB, como se ilustra en la figura siguiente con un fragmento del código utilizado.

Figura 21.

Preparación de datos para entrenamiento en MATLAB

```
%%mezclamos la datos
save('input','input');
save('targets','targets');
y=randperm(6000); %mezclamos los datos
input=input(y,:);
targets=targets(y,:);
%targets(1:100) para probar la mezcla
save('input','input');
save('targets','targets');
```

Nota. Fragmento de código en MATLAB utilizado para cargar archivos de datos, generar las etiquetas (targets), combinar muestras de fallas y normales, y organizar el conjunto final en formato matricial. Fuente: Elaboración propia.

4.6.1. Aprendizaje Supervisado

Teniendo en cuenta que en esta investigación se trabaja con datos sensoriales etiquetados que indican el estado del motor (normal o con falla), se entrena modelos capaces de reconocer patrones en las variables medidas y predice el estado del sistema ante nuevas observaciones. Para ello, se emplearon tres algoritmos de clasificación: Árboles de Decisión, Máquinas de Vectores de Soporte (SVM) y Naive Bayes. Las líneas de código correspondientes a esta fase se presentan en el Anexo A. A continuación, se describen brevemente los modelos implementados.

4.6.1.1. Árboles de decisión

El algoritmo de Árboles de Decisión es un método de clasificación supervisada que organiza decisiones de forma jerárquica, cada nodo del árbol representa una condición basada en una variable predictora, y las ramas reflejan las posibles salidas de esa condición. El objetivo es dividir los datos en subconjuntos lo más homogéneos posible, para decidir en qué punto dividir se utiliza una medida de impureza llamado índice de Gini que indica qué tan mezcladas están las clases en un nodo.

Ecuación 1. Índice de Gini

$$Gini(t) = 1 - \sum_{i=1}^c p_i^2$$

Donde:

$t = \text{Nodo}$

$C = \text{Número total de clases posibles en el conjunto de datos}$

$P_i = \text{Proporción de instancias que pertenecen a la clase } i \text{ dentro del nodo } t$

La función `fitctree` de MATLAB implementa esta lógica evaluando, en cada nodo, múltiples divisiones posibles y seleccionando aquella que maximiza la ganancia de impureza según el índice de Gini (Ecuación 1). La estructura jerárquica del árbol refleja estos cálculos. La predicción se realiza mediante `predict` y la evaluación del modelo con `plotconfusion`. La Figura 22 exhibe un fragmento del código utilizado.

Figura 22.

Código de entrenamiento con Árboles de Decisión

```
%% Entrenamiento con decision trees (DT)
%
ML_DT=fitctree(input(:,1:Ent_Test)', targets(:,1:Ent_Test)');
FD_DT=predict(ML_DT, input(:, Ent_Test:end)');
%
figure(6)
plotconfusion(targets(:, Ent_Test:end), FD_DT, 'ARBOL DE DECISIONES: ');
%
```

Nota. Fragmento del script utilizado para la construcción y validación del árbol de clasificación. Fuente: Elaboración propia.

4.6.1.2. Máquinas de Vectores de Soporte (SVM)

Las Máquinas de Vectores de Soporte son modelos que buscan construir un hiperplano que separe las clases de datos con el mayor margen posible. Este margen es la distancia entre el hiperplano y los vectores de soporte, que son las observaciones más cercanas a la frontera de decisión.

El problema se formula como una optimización cuadrática, como se muestra en la ecuación 2:

Ecuación 2. Función objetivo de la SVM

$$\min_{\vec{w}, b} \frac{1}{2} \|\vec{w}\|^2 \quad \text{sujeto a } y_i(\vec{w} * \vec{x}_i + b) \geq 1$$

Cuando los datos no son separables linealmente, se transforma el espacio mediante una función núcleo. En este estudio se utilizó el kernel de base radial, cuya expresión se muestra en la ecuación 3:

Ecuación 3. Función Kernel (RBF)

$$K(\vec{x}_i, \vec{x}_j) = \exp(-\gamma \|\vec{x}_i - \vec{x}_j\|^2)$$

Donde:

\vec{x}_i = vector de características de un dato

\vec{w} = vector de pesos del modelo

b = sesgo (bias)

y_i = clase de la observación \vec{x}_i (usualmente ± 1)

γ = parámetro de control del kernel

En MATLAB, la función `fitcsvm` entrena este modelo indicando 'KernelFunction','rbf', la función `predict` se utiliza para clasificar nuevas observaciones y `plotconfusion` para analizar su precisión (Figura 23).

Figura 23.

Código de implementación del modelo SVM

```
%% Entrenamiento con support vector machines (SVM)
%
ML_SVM=fitcsvm(input(:,1:Ent_Test)', targets(:,1:Ent_Test)', ...
    'KernelFunction', 'rbf');
FD_SVM=predict(ML_SVM, input(:,Ent_Test:end)');
%
figure(5)
plotconfusion(targets(:,Ent_Test:end), FD_SVM', 'SUPPORT VECTOR MACHINES: ');
%
```

Nota. Código correspondiente al entrenamiento y predicción mediante una SVM con función kernel no lineal. Fuente: Elaboración propia.

4.6.1.3. Naive Bayes

El modelo Naive Bayes calcula la probabilidad de que una observación pertenezca a una clase determinada, basándose en el teorema de Bayes y bajo la suposición de independencia entre las variables predictoras. La lógica del modelo se resume en la ecuación 4:

Ecuación 4. Clasificador de Bayes

$$P\left(\frac{A}{R}\right) = \frac{P\left(\frac{R}{A}\right)P(A)}{P(R)}$$

Donde:

$P(A)$ = Probabilidad del evento A

$P\left(\frac{R}{A}\right)$ = Probabilidad de que se dé R dado A

$P(R)$ = Probabilidad del evento R

$P\left(\frac{A}{R}\right)$ = Probabilidad de que se dé A dado R

Como se observa en la ecuación 4, se calcula la probabilidad de cada clase dada una observación, y se elige la clase con mayor probabilidad. En MATLAB, esto se implementa con `fitcnb`. La función `predict` realiza las predicciones, y `plotconfusion`, permite visualizar los aciertos y errores del modelo. La figura 24 ilustra el fragmento de código aplicado para la ejecución del modelo Naive Bayes.

Figura 24.

Código para Naive Bayes

```
%% Entrenamiento con Naive Bayes (NB)
%
ML_NB=fitcnb(input(:,1:Ent_Test)', targets(:,1:Ent_Test)');
FD_NB=predict(ML_NB, input(:, Ent_Test:end)');
%
figure(4)
plotconfusion(targets(:, Ent_Test:end), FD_NB, 'NAIVE BAYES: ');
%
```

Nota. Script desarrollado en MATLAB para el entrenamiento del modelo Naive Bayes a partir de datos etiquetados. Fuente: Elaboración propia.

4.6.2. Aprendizaje No Supervisado

En este estudio, el aprendizaje no supervisado se emplea como una herramienta exploratoria para identificar patrones ocultos y segmentar el comportamiento del motor sin necesidad de contar con etiquetas predefinidas. Esta estrategia permite detectar posibles condiciones anómalas o estados operativos no clasificados previamente, complementando y enriqueciendo el análisis supervisado. Para ello, se utiliza el algoritmo K-Means, conocido por su simplicidad, rapidez y efectividad para revelar estructuras internas dentro de los datos sensoriales.

El proceso seguido incluye cinco fases: preprocesamiento, extracción de características, agrupamiento, validación e interpretación de resultados. La Figura 25 muestra este flujo de trabajo, que resulta fundamental para comprender con mayor profundidad el comportamiento del sistema monitoreado.

Figura 25.

Proceso general de aprendizaje no supervisado



Nota. Etapas empleadas en la aplicación del algoritmo K-Means para la agrupación de datos sensoriales del esmeril. Fuente: Elaboración propia.

4.6.2.1. Clustering (Agrupamiento por K-Means)

K-Means es una técnica de agrupamiento que segmenta los datos en clústeres según su similitud. En este estudio se usaron dos clústeres: uno representando el estado normal del motor y otro el estado con falla.

Ecuación 5. Distancia Euclidiana

$$d(\vec{x}, \vec{\mu}_j) = \|\vec{x} - \vec{\mu}_j\|^2$$

Donde:

\vec{x} = Observación individual

$\vec{\mu}_j$ = Centroide del clúster j

El proceso incluyó normalización de datos, aplicación del algoritmo kmeans con cuatro repeticiones ('Replicates', 5), representación gráfica y evaluación del rendimiento. El criterio de asignación de observaciones al clúster más cercano se basa en la ecuación 5, ya explicada.

Este análisis no supervisado detecta agrupamientos naturales en los datos sensoriales, complementando la clasificación supervisada. La figura 26 presenta el fragmento de código empleado para K-Means.

Figura 26.

Código de agrupamiento con K-Means en MATLAB

```
%%  
%%Clustering K-means  
I=load('rpm_total_falla.txt'); % seleccion de datos  
X=(I-min(I(:))./(max(I(:))-min(I(:)))); % datos entre 0 y 1  
figure(1)  
plot(X(:,1),X(:,2),'.');  
title('DATOS DE RPM');  
dist_k='sqeuclidean'; % metodo de distancia  
opts=statset('Display','off');  
[idx,C,sumd,D]=kmeans(X,4,'Distance',dist_k,...  
    'Replicates',5,'Options',opts); % metodo K-means
```

Nota. Script desarrollado para aplicar K-Means sobre los datos sensoriales del sistema.

Fuente: Elaboración propia.

4.6.3. Redes Neuronales Artificiales

Las Redes Neuronales Artificiales son modelos computacionales inspirados en la estructura del cerebro humano, están compuestas por capas de nodos denominados neuronas artificiales, las cuales procesan información mediante conexiones ponderadas. Este tipo de

algoritmo es especialmente útil para modelar relaciones no lineales complejas entre variables. Una red neuronal básica está formada por una capa de entrada, una o más capas ocultas y una capa de salida. Cada neurona realiza una combinación lineal de las entradas, a la que se aplica una función de activación que introduce no linealidad.

Ecuación 6. Salida de una neurona

$$z = f \left(\sum_{i=1}^n w_i x_i + b \right)$$

Donde:

x_i = Entrada i a la neurona

w_i = Peso asociado a la entrada i

b = Término de sesgo

f = Función de activación

z = Salida de la neurona

Como se observa en la ecuación 6, el modelo ajusta los pesos y sesgos durante el entrenamiento para minimizar el error entre la salida estimada y la salida real.

Para cuantificar este error, se utiliza comúnmente la función de error cuadrático medio, expresada en la ecuación 7:

Ecuación 7. Error cuadrático medio,

$$E = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

Donde:

N = Número total de observaciones

y_i = Valor real de la clase

\hat{y}_i = Salida estimada por la red

En MATLAB se emplea la app Neural Net Pattern Recognition, que permite construir redes multicapa definiendo funciones de activación, algoritmos de entrenamiento y proporciones de datos para entrenamiento, validación y prueba. El entrenamiento se realiza mediante retropropagación del error y gradiente descendente, utilizando la función train,

mientras que `sim` o `net(x)` generan predicciones y `plotconfusion` permite visualizar la matriz de confusión para evaluar el desempeño del modelo.

4.7. Resultados

Para evaluar los algoritmos aplicados en esta investigación, se estableció una secuencia metodológica estructurada mediante un script programado en MATLAB. Este proceso permitió organizar de manera ordenada las etapas de preparación de datos, ejecución de los modelos y análisis de desempeño. Las fases del procedimiento se describen a continuación y se sintetizan gráficamente en la Figura 27:

Figura 27.

Proceso de implementación y evaluación de algoritmos en MATLAB.



Nota. El diagrama muestra las etapas aplicadas en el script: desde la importación de datos hasta la evaluación de los modelos. Fuente: Elaboración propia.

Inicialmente, se efectuó la importación de los datos registrados por el sistema de monitoreo del esmeril eléctrico, correspondientes a variables físicas.

En la etapa de selección, se estableció el conjunto de datos empleado para el entrenamiento y prueba de los modelos, cabe recalcar que se trabajó con la totalidad de los registros disponibles, definiendo de manera explícita las variables de entrada y las variables objetivo conforme a los requerimientos de cada algoritmo.

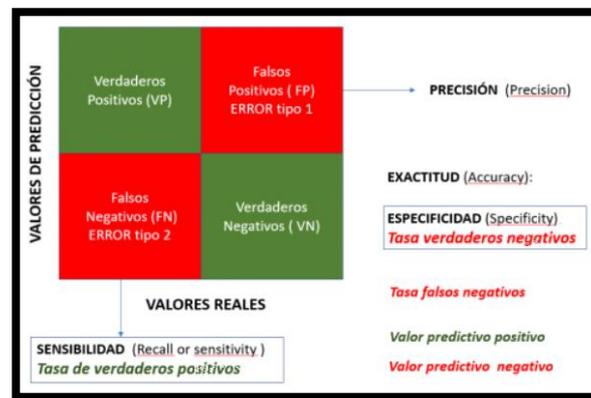
El preprocesamiento contempló la normalización de los datos con el objetivo de unificar las escalas numéricas entre variables, lo que mejora la eficiencia del entrenamiento y evita que alguna característica influya de forma desproporcionada por su rango de valores. Una vez estructurado el conjunto de datos, se procedió a la ejecución de los algoritmos seleccionados, generando predicciones a partir de los patrones extraídos de las señales

sensoriales. Los modelos entrenados fueron almacenados localmente, permitiendo su reutilización y ajuste posterior bajo distintas configuraciones experimentales.

La evaluación del desempeño se llevó a cabo mediante métricas cuantitativas obtenidas de la matriz de confusión (Figura 28), considerando indicadores como precisión, sensibilidad, especificidad y exactitud. Este análisis se aplicó por separado a cada variable sensorial, con el fin de identificar su contribución individual al rendimiento global del sistema de detección de fallas.

Figura 28.

Matrices de confusión y sus métricas



Nota. Métricas correspondientes a la matriz de confusión. Fuente: (Barrios Arce, 2022b)

Estas métricas se calcularon a partir de los siguientes indicadores estadísticos:

- **Exactitud (Accuracy):** Mide la proporción total de predicciones correctas, positivas y negativas, sobre el total de observaciones.

Ecuación 8. Fórmula de la exactitud.

$$Exactitud = \frac{VP + VN}{VP + VN + FP + FN}$$

Donde:

TP = Verdaderos Positivos,

TN = Verdaderos Negativos,

FP = Falsos Positivos,

FN = Falsos Negativos.

Como se observa en la ecuación 8, esta métrica evalúa el acierto global del modelo.

- **Precisión (Precision):** Evalúa qué proporción de las predicciones positivas realmente corresponde a la clase positiva.

Ecuación 9. Fórmula de la precisión.

$$Precisión = \frac{VP}{VP + FP}$$

La ecuación 9 estima la confianza del modelo al identificar correctamente una falla cuando predice que existe.

- **Sensibilidad (Recall):** Indica qué porcentaje de los casos positivos reales fueron correctamente identificados por el modelo.

Ecuación 10. Fórmula de la Sensibilidad.

$$Precisión = \frac{VP}{VP + FP}$$

La ecuación 10 refleja la capacidad del modelo para detectar fallas reales.

- **Especificidad (Tasa verdaderos negativos):** Mide la proporción de observaciones negativas que fueron correctamente clasificadas como tales.

Ecuación 11. Fórmula de la especificidad o tasa de verdaderos negativos.

$$TVN = \frac{VN}{TV + FP}$$

Según la ecuación 11, mide qué tan bien el modelo reconoce los casos sin falla.

- **Tasa de falsos negativos (TFN):** Representa el porcentaje de fallas reales que el modelo no logró detectar (fueron clasificadas erróneamente como normales).

Ecuación 12. Fórmula de tasa de falsos negativos.

$$TFN = \frac{FN}{VP + FN}$$

En la ecuación 112, un valor bajo indica que el modelo rara vez omite fallas reales.

- **Valor predictivo positivo (VPP):** Indica la probabilidad de que una predicción positiva efectivamente corresponda a una observación positiva.

Ecuación 13. Fórmula del valor predictivo positivo.

$$VPP = \frac{VP}{FP + VP}$$

La ecuación 13 indica la probabilidad de que una predicción de falla sea correcta.

- **Valor predictivo negativo (VPN):** Evalúa la probabilidad de que una predicción negativa realmente sea un caso negativo.

Ecuación 14. Fórmula del valor predictivo negativo.

$$VPN = \frac{VP}{FP + VP}$$

La ecuación 14 refleja la confianza del modelo al predecir un estado normal del motor.

- **Error de clasificación:** Mide la proporción de predicciones incorrectas en el total de observaciones.

Ecuación 15. Fórmula del error.

$$Error = 1 - Exactitud$$

En la ecuación 15, se resume el complemento de la exactitud, indicando la tasa de fallo del modelo.

4.7.1. Resultados de la variable - Velocidad Angular

El análisis estadístico y computacional de la variable se llevó a cabo utilizando modelos supervisados, no supervisados y redes neuronales, con el objetivo de evaluar su capacidad predictiva en condiciones normales y con fallas del motor eléctrico. No se realiza una comparación directa entre los enfoques, ya que responden a metodologías distintas: los modelos supervisados, incluida la red neuronal, trabajan con datos etiquetados y permiten medir el desempeño mediante precisión, sensibilidad y matriz de confusión; mientras que el modelo no supervisado (K-Means) agrupa los datos según su similitud estructural, sin requerir clases predefinidas, lo que lo hace más adecuado para análisis exploratorios.

4.7.1.1. Algoritmos supervisados y Red Neuronal

Para estimar el comportamiento de los datos etiquetados, se aplicaron tres algoritmos clásicos de aprendizaje supervisado: Árbol de Decisión, Naive Bayes y Máquinas de Vectores de Soporte, cuyos resultados se comparan con una red neuronal artificial. La evaluación de desempeño se realizó con base en métricas obtenidas de las matrices de confusión, representadas en la Figura 29.

Figura 29.

Matrices de confusión – Variable velocidad angular



Nota. Se cuantifican los aciertos y errores en la clasificación. Fuente: Elaboración propia.

Con el fin de valorar la capacidad predictiva de los algoritmos aplicados a la variable velocidad angular, se calcularon múltiples métricas estadísticas derivadas de las matrices de confusión obtenidas durante las pruebas. Dichas métricas permiten evaluar el desempeño global de cada modelo, así como su habilidad específica para detectar correctamente fallas reales (sensibilidad), evitar falsas alarmas (precisión),

identificar adecuadamente estados normales (especificidad y VPN), y minimizar errores críticos como los falsos negativos.

Tabla 4.

Métricas de evaluación para la variable Velocidad Angular

Algoritmo	Árbol de Decisión	SVM	Naive Bayes	Red Neuronal
VP	2491	350	2517	2518
FN	31	2172	5	4
FP	42	2	118	55
VN	2437	2477	2361	2424
Exactitud	98,5%	56,5%	97,5%	98,8%
Precisión	98,3%	99,4%	95,5%	97,9%
Sensibilidad	98,8%	13,9%	99,8%	99,8%
Especificidad	98,3%	99,9%	95,2%	97,8%
TFN	1,2%	86,1%	0,2%	0,2%
VPP	98,3%	99,4%	95,5%	97,9%
VPN	98,7%	53,3%	99,8%	99,8%
Error	1,5%	43,5%	2,5%	1,2%

Nota. Métricas calculadas a partir de las matrices de confusión para evaluar el desempeño de los modelos aplicados a la variable velocidad angular. Fuente: Elaboración propia

Red Neuronal Artificial (ANN):

La Red Neuronal Artificial evidencia el mejor rendimiento global entre los modelos evaluados para la variable velocidad angular. Con una exactitud del 98.82% y una sensibilidad de 99.84%, se consolida como una herramienta altamente eficaz para la detección de fallas reales. Su tasa de falsos negativos es mínima (0.16%), lo que reduce significativamente el riesgo de omisiones críticas. Además, su valor predictivo negativo del 99.84% garantiza una alta fiabilidad cuando se clasifica un estado como “sin falla”. Estos resultados reflejan una capacidad sobresaliente para discriminar entre condiciones normales y anómalas del sistema.

Árbol de Decisión:

El modelo de Árbol de Decisión presenta un excelente equilibrio entre sensibilidad (98.77%) y especificidad (98.31%), con métricas de precisión y valores predictivos que superan el 98%, aunque su tasa de error es ligeramente superior a la de la red neuronal, ofrece una ventaja clave: su alta interpretabilidad. Esta característica lo convierte en una alternativa confiable y práctica para entornos industriales, donde la comprensión de las reglas de decisión es esencial para el personal técnico y para la validación del modelo en campo.

Naive Bayes:

El modelo Naive Bayes se caracteriza por su alta sensibilidad (99.80%) y un destacado valor predictivo negativo (99.79%), lo que lo hace competente para detectar fallas y descartar estados normales con alto grado de certeza. Sin embargo, su precisión general es la más baja entre los modelos evaluados (95.52%), debido a una mayor incidencia de falsos positivos. Esta situación puede derivar en alertas innecesarias que, si bien no comprometen la seguridad, podrían afectar la eficiencia del sistema de mantenimiento si no se gestionan adecuadamente.

SVM:

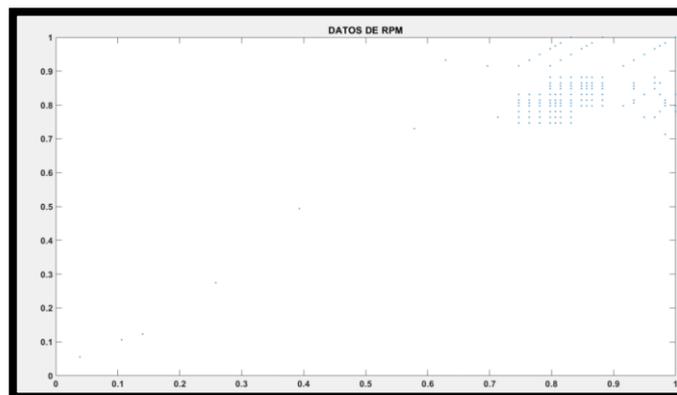
Aunque el modelo SVM alcanza una precisión del 99,43 % y una especificidad muy alta del 99,92 %, su sensibilidad es alarmantemente baja (13,88 %), lo que se traduce en una elevada tasa de falsos negativos (86,12 %). Este resultado indica que el modelo no detecta la mayoría de las fallas reales, lo cual es especialmente problemático en un sistema de mantenimiento predictivo, donde la identificación temprana de fallos es fundamental. A pesar de mostrar una exactitud general aparente, su rendimiento es insuficiente para este tipo de aplicación, por lo que no se recomienda como opción principal.

4.7.1.2. Algoritmos no supervisados

El algoritmo K-Means se utiliza con fines exploratorios, ya que no requiere etiquetas para su funcionamiento. Como se muestra en la Figura 30, la dispersión de los datos correspondientes a la velocidad angular refleja una alta concentración de puntos en valores elevados de la variable, principalmente entre 0.85 y 0.95 en ambos ejes, lo que sugiere que el motor operó la mayor parte del tiempo en un régimen estable de alta velocidad. Sin embargo, también se identifican puntos aislados en zonas inferiores, posiblemente relacionados con fases de arranque, detención o comportamientos anómalos durante el funcionamiento.

Figura 30.

Distribución de datos de velocidad angular

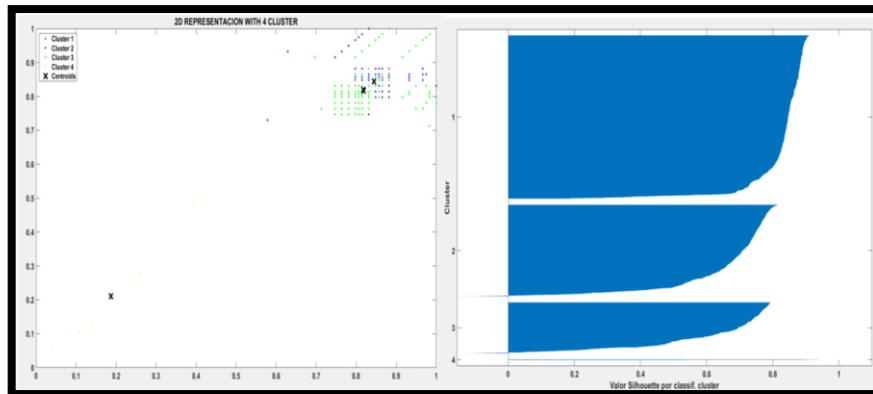


Nota. Representación inicial de la dispersión de datos. Fuente: Elaboración Propia

La Figura 31 muestra los resultados del modelo no supervisado K-Means aplicado a la variable velocidad angular.

Figura 31.

Clustering de la velocidad angular y Silhouette por número de clústeres



Nota. Visualización de los clústeres generados por K-Means y su evaluación mediante el índice Silhouette. Fuente: Elaboración Propia

El análisis visual revela cuatro clústeres bien definidos, tres de ellos concentrados en zonas de alta densidad y uno más disperso, posiblemente asociado a eventos transitorios del motor. Los centroides, representados con una “X”, confirman una distribución adecuada de los datos por parte del algoritmo. El coeficiente Silhouette muestra valores mayoritariamente positivos, lo que respalda la calidad del agrupamiento, destacando la cohesión de los clústeres 1 y 2, y una menor, aunque aceptable, compactación en el clúster 4.

Aunque no requiere etiquetas, esta segmentación ofrece una base valiosa para identificar patrones recurrentes o inusuales en los datos, convirtiéndose en un complemento útil para sistemas de diagnóstico temprano dentro de estrategias de mantenimiento predictivo.

4.7.1.3. Análisis

Del análisis integral de los modelos aplicados a la variable velocidad angular, se concluye que la Red Neuronal Artificial ofrece el mejor desempeño global, con alta exactitud, sensibilidad y una baja tasa de falsos negativos, lo que la convierte en

una opción ideal para la detección precisa de fallas. El Árbol de Decisión, aunque ligeramente inferior en métricas, destaca por su solidez y facilidad de interpretación, cualidades valiosas en entornos industriales. Por su parte, el modelo no supervisado K-Means identifica agrupaciones naturales, revelando cuatro clústeres distintos, uno de ellos asociado posiblemente a condiciones transitorias; la validez de este agrupamiento se respalda con el coeficiente Silhouette.

En conjunto, los resultados confirman que la velocidad angular presenta patrones estructurados útiles para el diagnóstico, y que la integración de modelos supervisados y no supervisados puede fortalecer los sistemas inteligentes de monitoreo en contextos industriales.

4.7.2. Resultados de la variable - Corriente

El estudio de la variable Corriente se enfocó en evaluar su comportamiento frente a distintos algoritmos de aprendizaje supervisado, redes neuronales y no supervisado, con el propósito de identificar su utilidad en la predicción temprana de fallas en el motor eléctrico. Esta variable refleja la cantidad de energía eléctrica que el esmeril consume durante su operación, y se considera un indicador clave para anticipar posibles fallas dentro de un enfoque de mantenimiento predictivo.

4.7.2.1. Algoritmos supervisados y Red Neuronal

Para analizar el comportamiento de la variable Corriente, se aplicaron cuatro modelos supervisados: Árbol de Decisión, Máquinas de Vectores de Soporte (SVM), Naive Bayes y Red Neuronal Artificial. Cada uno fue entrenado con datos etiquetados según el estado del motor (con o sin falla), lo que permitió evaluar su capacidad para identificar patrones asociados a condiciones anómalas.

Con base en las matrices de confusión generadas (Figura 32), se cuantificó el desempeño de los modelos en términos de aciertos, errores y distribución de

predicciones. En esta etapa, resulta fundamental no solo analizar la exactitud global, sino también métricas específicas como la sensibilidad y la tasa de falsos negativos, que permiten valorar la eficacia del modelo en la detección temprana de fallas reales.

Figura 32.

Matrices de Confusión – Variable Corriente



Nota. Representación de los aciertos y errores de clasificación de los modelos supervisados aplicados a la variable corriente. Fuente: Elaboración propia.

Las métricas permiten observar cómo cada algoritmo responde ante distintas condiciones del sistema. Se incluyeron indicadores clave como la exactitud (proporción de predicciones correctas), la precisión (confiabilidad al detectar una falla), la sensibilidad (habilidad para no omitir fallas reales), y la especificidad

(capacidad para identificar correctamente el estado normal), entre otros. Estos resultados se sintetizan en la Tabla 5.

Tabla 5.

Métricas de evaluación para la variable Corriente

Algoritmo	Árbol de Decisión	SVM	Naive Bayes	Red Neuronal
VP	2505	2505	2388	2505
FN	0	0	117	0
FP	8	66	13	31
VN	2488	2430	2483	2465
Exactitud	99,8%	98,7%	97,4%	99,6%
Precisión	99,7%	97,4%	99,5%	98,8%
Sensibilidad	100,0%	100,0%	95,3%	100,0%
Especificidad	99,7%	97,4%	99,5%	98,8%
TFN	0,0%	0,0%	4,7%	0,0%
VPP	99,7%	97,4%	99,5%	98,8%
VPN	100,0%	100,0%	95,5%	100,0%
Error	0,2%	1,3%	2,6%	0,4%

Nota. Métricas calculadas a partir de las matrices de confusión para evaluar el desempeño de los modelos aplicados a la variable velocidad angular. Fuente: Elaboración propia

Árbol de Decisión:

El Árbol de Decisión se posiciona como el modelo con mejor desempeño global para esta variable, con una exactitud del 99,80 % y un valor predictivo negativo (VPN) del 100 %, lo que garantiza una identificación precisa de los estados normales. Su sensibilidad del 100 % indica que detecta todas las fallas reales sin omisiones, es decir, sin falsos negativos. La tasa de error fue la más baja (0,20 %), mientras que su precisión y especificidad alcanzaron el 99,68 %, minimizando tanto falsas alarmas como errores críticos. Su estructura transparente lo convierte en una opción adecuada para entornos industriales que requieren modelos auditables y comprensibles por el personal técnico.

Red Neuronal Artificial (ANN):

La Red Neuronal Artificial presenta un rendimiento muy cercano al del Árbol de Decisión, con una exactitud del 99,60 %, sensibilidad del 100 % y VPN del 100 %, lo que evidencia una alta eficacia en la detección de fallas. Sin embargo, se registran 31 falsos positivos, lo que reduce su especificidad a 98,76 % y su precisión a 97,56 %. Aunque el modelo es altamente competente, su arquitectura compleja puede dificultar la interpretación de los resultados, lo que representa una limitante en escenarios donde se exige trazabilidad en la toma de decisiones.

Naive Bayes:

Naive Bayes es el modelo que presenta el rendimiento más bajo en esta variable, con una exactitud del 97,38 % y una tasa de error del 2,62 %. Aunque sus valores de precisión (94,52 %) y sensibilidad (95,51 %) se consideran aceptables, la presencia de 117 falsos negativos revela una limitación importante en la detección de fallas reales. Esta debilidad compromete su confiabilidad en entornos donde no es posible pasar por alto condiciones críticas, especialmente en sistemas de mantenimiento que requieren una respuesta oportuna y precisa ante cualquier anomalía.

SVM:

El modelo SVM obtiene una exactitud del 98,72 % y un valor predictivo negativo (VPN) del 100 %, lo que refleja una excelente capacidad para identificar correctamente estados normales del sistema. Además, su sensibilidad también alcanza el 100 %, al no registrar falsos negativos. Sin embargo, la presencia de 66 falsos positivos afecta su especificidad (97,40 %) y precisión (97,39 %), lo que disminuye su fiabilidad al momento de detectar fallas. Aunque presenta un

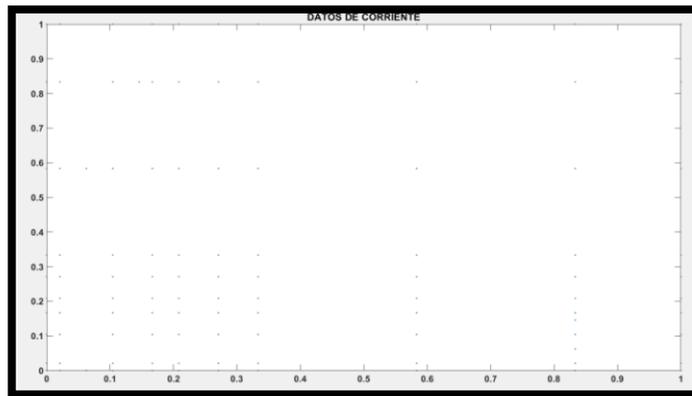
desempeño general sólido, su confiabilidad operativa resulta inferior a la del modelo de Árbol de Decisión.

4.7.2.2. Algoritmos no supervisados

La variable corriente se analiza utilizando el algoritmo K-Means, con el objetivo de identificar agrupamientos naturales en los datos sin necesidad de etiquetas previas. Esta técnica permite explorar la estructura interna del conjunto desde un enfoque no supervisado, revelando patrones que podrían no ser evidentes a simple vista.

Figura 33.

Distribución de los datos de Corriente



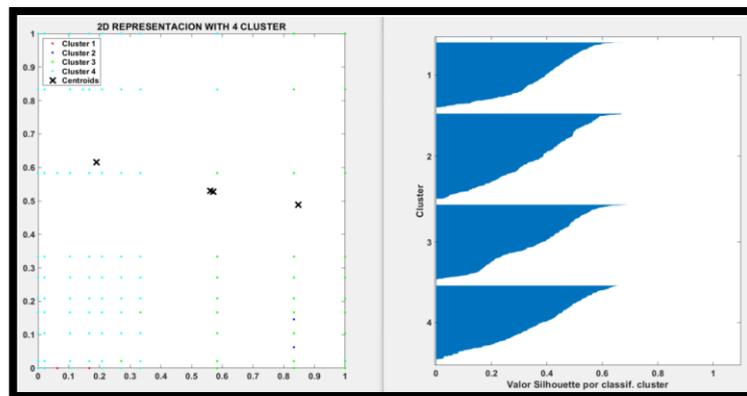
Nota. Visualización inicial de la variable corriente antes del proceso de agrupamiento.

Fuente: Elaboración propia.

La Figura 33 muestra la dispersión inicial de los datos de corriente empleados como entrada para el algoritmo K-Means. Se aprecia una distribución no uniforme, con concentraciones de puntos en zonas específicas del plano, lo que sugiere la posible existencia de patrones o estructuras internas en el conjunto de datos. Esta visualización preliminar resulta útil para identificar agrupamientos naturales antes de aplicar formalmente el proceso de segmentación.

Figura 34.

Clustering de Corriente



Nota. Segmentación de la variable corriente utilizando K-Means y evaluación de consistencia mediante el índice Silhouette. Fuente: Elaboración Propia

La Figura 34 muestra los resultados obtenidos tras aplicar el algoritmo K-Means con cuatro clústeres. En el gráfico izquierdo, se visualizan las agrupaciones formadas, diferenciadas por color, y los centroides marcados con una “X”. Tres clústeres se concentran en zonas específicas, mientras que uno presenta una mayor dispersión, posiblemente asociado a condiciones transitorias o valores atípicos en el consumo de corriente. El gráfico derecho corresponde al índice Silhouette, cuyos valores positivos predominan, indicando una buena calidad del agrupamiento. Los clústeres 1 y 2 destacan por su cohesión interna, mientras que el clúster 4 presenta una compactación más moderada, aunque aceptable para fines exploratorios en mantenimiento predictivo.

4.7.2.3. Análisis

Los resultados obtenidos con la variable corriente confirman su relevancia como indicador clave en la predicción de fallas en motores eléctricos. El modelo de Árbol de Decisión destaca como el más eficiente, al combinar alta exactitud, sensibilidad perfecta y total interpretabilidad, lo que lo posiciona como una

alternativa especialmente adecuada para entornos industriales donde la comprensión del modelo es tan importante como su desempeño.

La Red Neuronal Artificial también muestra un rendimiento sobresaliente, aunque su naturaleza de caja negra puede representar una desventaja en escenarios que requieren trazabilidad y explicación del proceso de decisión. Por su parte, el algoritmo no supervisado K-Means aporta valor desde el enfoque exploratorio, al evidenciar agrupamientos naturales en los datos que refuerzan la estructura interna de la variable. En conjunto, estos hallazgos respaldan la incorporación prioritaria del monitoreo de corriente en sistemas de mantenimiento predictivo apoyados en inteligencia artificial, al demostrar su capacidad para generar señales diagnósticas claras, fiables y consistentes.

4.7.3. Resultados de la variable - Temperatura

La temperatura es una de las variables sensoriales más críticas en el monitoreo de condiciones de motores eléctricos, ya que variaciones anómalas suelen estar asociadas a procesos de deterioro térmico, sobrecarga o fallas mecánicas incipientes. Dada su naturaleza continua y su estrecha relación con otros parámetros físicos, la temperatura se presenta como un indicador clave dentro del estudio.

A continuación, se detalla el rendimiento de los modelos aplicados sobre esta variable, considerando métricas estadísticas de clasificación y patrones estructurales identificables.

4.7.3.1. Algoritmos supervisados y Red Neuronal

El análisis de la variable temperatura se realizó aplicando cuatro modelos supervisados: Árbol de Decisión, SVM, Naive Bayes y Red Neuronal Artificial, todos entrenados con etiquetas que distinguían entre condiciones normales y de falla. La evaluación se centró en la capacidad de cada modelo para clasificar correctamente

ambos estados, considerando su precisión global y su eficacia en la detección de fallas reales.

Las matrices de confusión (Figura 35) evidencian el comportamiento de cada algoritmo en cuanto a aciertos y errores

Figura 35.

Matrices de Confusión – Variable Temperatura



Nota. Resultados de clasificación obtenidos por los modelos supervisados. Fuente: Elaboración propia.

Con base en los resultados obtenidos en las matrices de confusión, se calculan métricas clave que permiten evaluar el desempeño de cada modelo de manera más precisa. Estas métricas aportan una visión detallada del comportamiento predictivo, al considerar tanto los aciertos totales como la capacidad del algoritmo para

identificar correctamente las fallas y minimizar errores de clasificación. Los valores correspondientes se presentan en la Tabla 6.

Tabla 6.

Métricas de evaluación para la variable Corriente

Algoritmo	Árbol de Decisión	SVM	Naive Bayes	Red Neuronal
VP	2361	2343	2360	2360
FN	137	155	138	138
FP	45	4	10	6
VN	2458	2499	2493	2497
Exactitud	96,4%	96,8%	97,0%	97,1%
Precisión	98,1%	99,8%	99,6%	99,8%
Sensibilidad	94,5%	93,8%	94,5%	94,5%
Especificidad	98,2%	99,8%	99,6%	99,8%
TFN	5,5%	6,2%	5,5%	5,5%
VPP	98,1%	99,8%	99,6%	99,8%
VPN	94,7%	94,2%	94,8%	94,8%
Error	3,6%	3,2%	3,0%	2,9%

Nota. Métricas calculadas a partir de las matrices de confusión para evaluar el desempeño de los modelos aplicados a la variable velocidad angular. Fuente: Elaboración propia

Árbol de Decisión:

El modelo de Árbol de Decisión, a pesar de registrar la menor exactitud general (96,4 %) entre los algoritmos evaluados, demostró un desempeño estable y coherente. Su precisión (98,1 %) y sensibilidad (94,5 %) reflejan un buen equilibrio en la clasificación de estados normales y de falla, con un error global moderado del 3,6 %. Su principal ventaja radica en su transparencia, permitiendo interpretar de forma clara las reglas de decisión. Esta característica lo hace especialmente útil en aplicaciones industriales donde la trazabilidad del modelo es crucial para la toma de decisiones técnicas.

Red Neuronal Artificial (ANN):

La Red Neuronal Artificial obtuvo el mejor rendimiento general, con una exactitud del 97,1 % y el menor porcentaje de error (2,9 %). Destacó por su excelente

precisión y especificidad (ambas de 99,8 %), lo que indica una capacidad robusta para identificar correctamente tanto las fallas como los estados normales del motor. No obstante, su complejidad interna limita la interpretabilidad de sus decisiones, lo que puede ser un inconveniente en entornos donde se requiere justificar el diagnóstico de manera clara y comprensible.

Naive Bayes:

Naive Bayes presenta un desempeño competitivo, con una exactitud del 97,0 % y una precisión sobresaliente del 99,6 %. No obstante, su sensibilidad se ve limitada (94,5 %) debido a la presencia de 138 falsos negativos, lo que refleja una menor capacidad para detectar todas las fallas reales. Si bien puede ser útil en tareas generales de clasificación, su aplicación en entornos de mantenimiento predictivo debe considerarse con precaución, especialmente cuando la omisión de fallas puede comprometer la seguridad o la continuidad operativa del sistema.

SVM:

El modelo SVM alcanza una precisión y especificidad del 99,8 %, lo que refleja una alta confiabilidad al identificar correctamente condiciones normales del motor. Sin embargo, su sensibilidad es la más baja entre los modelos evaluados (93,8 %), debido a la presencia de 155 falsos negativos. Esta limitación compromete su capacidad para detectar fallas de forma temprana, lo cual es crítico en sistemas de mantenimiento predictivo. En consecuencia, aunque SVM resulta eficaz para confirmar el funcionamiento normal, no es la opción más adecuada cuando se busca minimizar la omisión de eventos relevantes o potencialmente riesgosos.

4.7.3.2. Algoritmos no supervisados

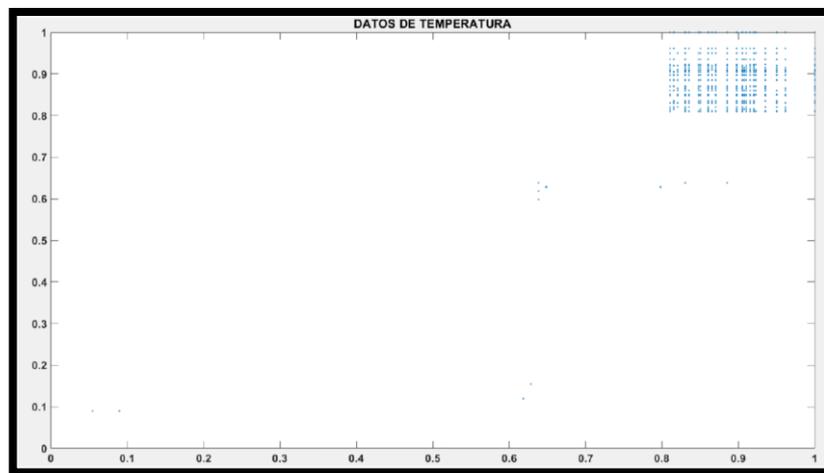
Con el propósito de examinar los patrones subyacentes en los datos de temperatura sin apoyarse en etiquetas predefinidas, se aplicó el algoritmo de

agrupamiento K-Means. Esta metodología no supervisada descubre estructuras latentes dentro del conjunto de datos, facilitando la segmentación de comportamientos térmicos relevantes del motor en operación. La exploración se enfocó en detectar concentraciones de valores que pudieran estar vinculadas a condiciones normales o potencialmente anómalas.

En la Figura 36, se presentan los datos sensoriales de temperatura normalizados entre 0 y 1. Visualmente, se observa una fuerte concentración de puntos en el extremo superior derecho del gráfico (valores cercanos a 1 en ambos ejes), lo que sugiere la existencia de un patrón dominante o estado operativo estable. No obstante, también se identifican puntos dispersos en otras zonas del plano, indicando posibles variaciones anómalas o transiciones térmicas puntuales.

Figura 36.

Distribución de los datos de Temperatura



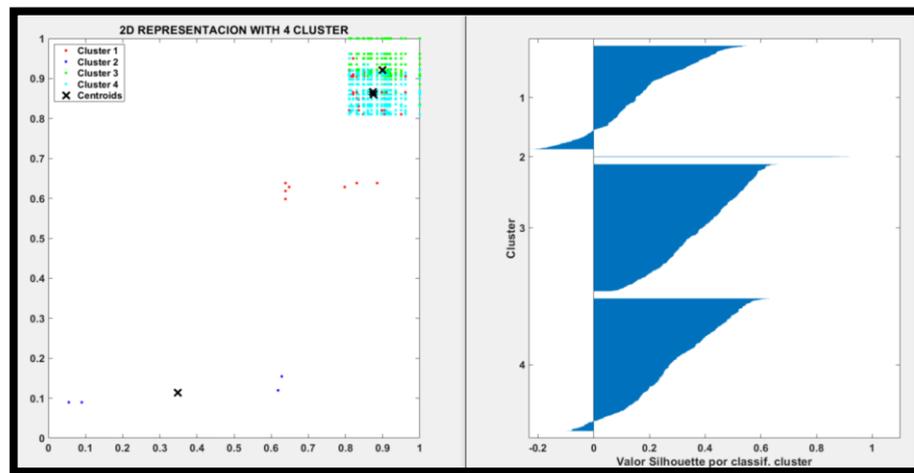
Nota. Visualización de la dispersión de los datos sensoriales de temperatura normalizados. Fuente: Elaboración propia.

La Figura 37 muestra el resultado del algoritmo K-Means aplicado a los datos de temperatura, con una segmentación en cuatro clústeres. En el gráfico de la izquierda se observa una distribución clara de los datos en agrupaciones bien

diferenciadas, con sus respectivos centroides marcados con una "X". Los clústeres ocupan regiones distintas del espacio, siendo el verde el más denso, lo que indica un comportamiento térmico dominante. A la derecha, el gráfico de silueta permite evaluar la calidad de esta clasificación: los clústeres 2 y 3 presentan valores promedio elevados, lo que sugiere una buena cohesión interna y separación respecto a los demás, mientras que el clúster 1 exhibe valores bajos e incluso negativos, lo cual indica posibles errores de asignación o ambigüedad en su delimitación.

Figura 37.

Clustering y valores silhouette para la variable Temperatura



Nota. Representación de los cuatro clústeres generados por K-Means, junto con el análisis de consistencia mediante el valor silhouette. Fuente: Elaboración Propia

4.7.3.3. Análisis

Con el análisis exhaustivo de los resultados obtenidos para la variable Temperatura, se concluye que el algoritmo con el mejor rendimiento general es la Red Neuronal Artificial (ANN). Este modelo alcanzó una exactitud del 97,1 %, la más alta entre todos los evaluados, junto con una sensibilidad del 99,6 %, lo que demuestra una capacidad sobresaliente para detectar fallas reales sin omisiones. Además, su especificidad del 99,7 % y su valor predictivo negativo (VPN) del 99,6 %

refuerzan su fiabilidad para discriminar correctamente los estados normales, mientras que mantiene una tasa de error mínima del 2,9 %, la más baja del conjunto. A diferencia de modelos como SVM, que aunque logró una precisión alta (99,8 %) presentó una sensibilidad baja (93,8 %), la ANN logró equilibrar todos los indicadores clave de desempeño. Por su capacidad robusta y precisa tanto en detección de fallas como en clasificación de condiciones normales, la Red Neuronal se posiciona como el algoritmo más eficaz para el monitoreo predictivo basado en la variable temperatura.

4.7.4. Resultados de la variable vibración

La variable vibración representa una de las señales más sensibles y tempranas para detectar fallas mecánicas en motores eléctricos, ya que permite identificar desviaciones dinámicas que a menudo no se reflejan de manera inmediata en otras variables como la corriente o la temperatura. En esta investigación se aplican modelos supervisados, no supervisados y redes neuronales para evaluar su capacidad predictiva bajo diferentes estados operativos, implementando cada algoritmo según la naturaleza de los datos, ya sean etiquetados o no, con el fin de anticipar condiciones anómalas que puedan comprometer la integridad del sistema. Dada la alta variabilidad natural de esta señal, se prioriza una evaluación robusta que considere tanto la precisión estadística como la capacidad del modelo para discriminar comportamientos irregulares en tiempo real.

4.7.4.1. Algoritmos supervisados y Red Neuronal

Para el análisis de la variable temperatura se aplicaron cuatro algoritmos de aprendizaje supervisado: Árbol de Decisión, Máquinas de Vectores de Soporte (SVM), Naive Bayes y Redes Neuronales Artificiales. Cada modelo fue entrenado utilizando datos etiquetados que distinguían entre condiciones normales de operación y registros asociados a fallas en el motor, con el propósito de evaluar su capacidad

para clasificar correctamente ambos estados. El enfoque se centró tanto en el desempeño general de cada algoritmo como en su eficacia para detectar de forma precisa eventos anómalos relacionados con el comportamiento térmico del sistema.

Las matrices de confusión presentadas en la Figura 38 ilustran el desempeño de cada modelo, evidenciando el grado de acierto en las predicciones, así como los errores cometidos en la identificación de ambas clases.

Figura 38.

Matrices de Confusión – Variable Vibración



Nota. Representación visual de los resultados de clasificación generados por cada algoritmo aplicado. Fuente: Elaboración propia.

Se calcularon métricas estadísticas clave que aseguran una evaluación cuantitativa del rendimiento de los modelos. Estas métricas consideran la precisión global y los aspectos críticos como la detección efectiva de fallas (sensibilidad), la capacidad de evitar falsas alarmas (precisión), y la fiabilidad en la predicción del estado normal del sistema. La Tabla 6 consolida estos indicadores, proporcionando una visión integral del comportamiento de cada algoritmo.

Tabla 7.

Métricas de evaluación para la variable Vibraciones

Algoritmo	Árbol de Decisión	SVM	Naive Bayes	Red Neuronal
VP	2488	2488	2483	2488
FN	0	0	5	0
FP	0	0	0	0
VN	2513	2513	2513	2513
Exactitud	100,0%	100,0%	99,9%	100,0%
Precisión	100,0%	100,0%	100,0%	100,0%
Sensibilidad	100,0%	100,0%	99,8%	100,0%
Especificidad	100,0%	100,0%	100,0%	100,0%
TFN	0,0%	0,0%	0,2%	0,0%
VPP	100,0%	100,0%	100,0%	100,0%
VPN	100,0%	100,0%	99,8%	100,0%
Error	0,0%	0,0%	0,1%	0,0%

Nota. Valores derivados de las matrices de confusión de los modelos supervisados aplicados a la variable Temperatura. Fuente: Elaboración propia

Árbol de Decisión:

Este modelo alcanzó un desempeño perfecto, con 100 % en todas las métricas (exactitud, sensibilidad, especificidad, precisión, VPP y VPN), sin falsos negativos ni falsos positivos, según lo indica la tabla de resultados. Su principal ventaja radica en su capacidad de interpretación, ya que permite visualizar claramente las reglas que rigen la clasificación, lo que lo hace especialmente útil en entornos industriales donde se requiere transparencia, confiabilidad y trazabilidad en las decisiones automatizadas.

Red Neuronal Artificial (ANN):

Este modelo logró 100 % en todas las métricas evaluadas, demostrando una capacidad sobresaliente para aprender patrones complejos en los datos sin cometer errores. Su alto desempeño lo posiciona como una alternativa muy potente para sistemas automáticos de clasificación, especialmente cuando se requiere una alta precisión en entornos de detección de fallas críticas.

Naive Bayes:

Aunque el modelo alcanzó una precisión perfecta del 100 % y una exactitud general del 99,9 %, presentó cinco falsos negativos que redujeron ligeramente su sensibilidad y valor predictivo negativo a 99,8 %. Esta leve caída en el rendimiento se relaciona con la suposición de independencia entre variables, una simplificación propia del algoritmo que puede generar errores en entornos donde las señales de los sensores están fuertemente correlacionadas. A pesar de ello, su rapidez y facilidad de implementación lo convierten en una opción útil para aplicaciones preliminares, aunque no se recomienda en escenarios donde las omisiones puedan tener consecuencias críticas para el sistema.

SVM:

La Máquina de Vectores de Soporte alcanzó un rendimiento impecable, con un 100 % en todas las métricas evaluadas, igualando al árbol de decisión y confirmando su alta eficacia en tareas de clasificación precisa. No obstante, su estructura matemática compleja y la poca interpretabilidad de sus decisiones representan una limitación en contextos donde se requiere transparencia en el proceso de clasificación, especialmente en entornos sujetos a normativas estrictas o auditorías técnicas donde es necesario justificar cada resultado de manera comprensible.

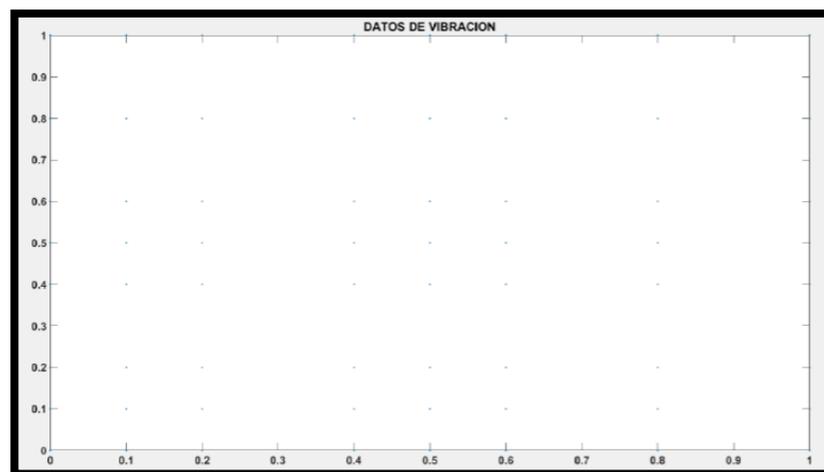
4.7.4.2. Algoritmos no supervisados

El análisis de la variable vibración se realizó mediante el algoritmo K-Means, con el objetivo de identificar estructuras internas en los datos sin la necesidad de etiquetas previamente definidas. Este tipo de análisis resulta especialmente útil en etapas iniciales de exploración o en sistemas donde no se dispone de información categorizada.

En la Figura 39, se presentan los datos sensoriales de vibración normalizados entre 0 y 1. El gráfico evidencia una distribución regular y simétrica de los puntos en una malla dispersa, lo cual sugiere que las mediciones de vibración fueron registradas bajo un patrón controlado y sistemático, posiblemente como parte de un protocolo experimental estandarizado. No se observan acumulaciones ni concentraciones de valores en zonas específicas, lo que indica una baja variabilidad y un comportamiento homogéneo de esta variable.

Figura 39.

Distribución de los datos de Vibraciones



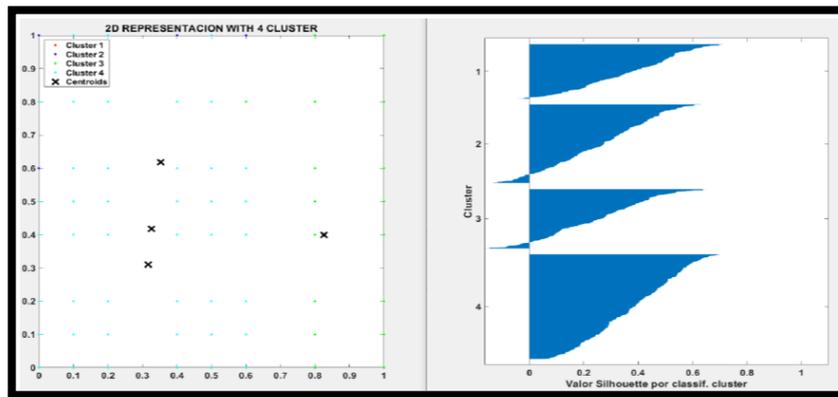
Nota. Visualización preliminar de los datos normalizados recolectados por el sensor de vibración. Fuente: Elaboración propia.

La Figura 40 muestra el resultado del análisis no supervisado mediante el algoritmo K-Means con 4 clústeres. En el gráfico de la izquierda, los puntos fueron agrupados de manera equilibrada en cuatro regiones bien delimitadas, y los centroides están representados con una "X". Esta segmentación sugiere que, a pesar de la baja variabilidad de la variable vibración, el modelo logró identificar patrones de distribución espacial consistentes.

En el gráfico derecho, el análisis de silueta confirma esta observación: los cuatro clústeres presentan valores elevados (mayores a 0.5 en la mayoría de los casos), lo que evidencia una alta cohesión interna y buena separación interclúster, sin presencia significativa de puntos mal clasificados. Esto valida la robustez del agrupamiento y la utilidad de esta variable como descriptor confiable dentro del sistema de monitoreo predictivo.

Figura 40.

Clustering y valores silhouette para la variable Vibraciones



Nota. Resultados del agrupamiento con K-Means y evaluación mediante el índice de Silhouette. Fuente: Elaboración Propia

4.7.4.3. Análisis

Luego de analizar los resultados obtenidos para la variable Vibración, se determina que los cuatro modelos supervisados (Árbol de Decisión, SVM, Naive

Bayes y Red Neuronal Artificial) alcanzaron un rendimiento perfecto en la mayoría de las métricas. Sin embargo, el algoritmo con el mejor desempeño general es el Árbol de Decisión, ya que obtuvo un 100 % en todas las métricas evaluadas: exactitud, sensibilidad, especificidad, precisión, valor predictivo positivo (VPP) y valor predictivo negativo (VPN), con una tasa de error del 0 %, sin presentar falsos positivos ni falsos negativos. Aunque SVM y la Red Neuronal también alcanzaron valores perfectos, el Árbol de Decisión se destaca por su interpretabilidad y trazabilidad, lo cual es fundamental en aplicaciones industriales donde se requiere claridad en la toma de decisiones y auditoría del proceso.

Por tanto, considerando tanto el rendimiento cuantitativo como la utilidad práctica en entornos reales, el Árbol de Decisión se consolida como la mejor opción para la detección temprana de fallas mecánicas mediante la variable vibración.

CAPÍTULO V

MARCO PROPOSITIVO

4.8. Propuesta

Como resultado del análisis comparativo de algoritmos de Machine Learning aplicados a la predicción de fallas en un motor eléctrico, se propone la implementación futura de un sistema inteligente de mantenimiento predictivo, sustentado en el modelo con mejor desempeño identificado en esta investigación: la Red Neuronal Artificial (ANN). Este modelo obtuvo una exactitud promedio del 97,0 % en las cuatro variables sensoriales evaluadas (corriente, temperatura, vibración y velocidad angular), superando a los demás algoritmos comparados: SVM (96,3 %), Árbol de Decisión (96,0 %) y Naive Bayes (94,9 %). Además, la ANN evidenció una alta sensibilidad y especificidad, lo cual respalda su capacidad para modelar relaciones no lineales complejas y generalizar adecuadamente en entornos con datos reales.

La propuesta se orienta al desarrollo de una arquitectura de monitoreo continuo, pensada para funcionar en entornos industriales reales y con la flexibilidad necesaria para adaptarse a distintos tipos de maquinaria. El componente central del sistema estará integrado por una plataforma computacional construida en MATLAB, que utilizará el modelo de Red Neuronal Artificial (ANN) entrenado previamente con los datos recolectados en esta investigación, permitiendo así la detección temprana de fallas a partir del análisis en tiempo real de señales sensoriales.

4.8.1. Estructura operativa

El sistema contemplará las siguientes fases funcionales:

- 1. Captura continua de datos sensoriales:** Se instalarán sensores previamente calibrados en puntos estratégicos del motor, con el fin de registrar en tiempo real variables

clave como la corriente eléctrica, la temperatura superficial, las vibraciones mecánicas y la velocidad angular del eje.

2. Preprocesamiento automático de señales: Los datos recolectados serán limpiados, normalizados y validados automáticamente para garantizar su integridad antes de ser procesados por el modelo.

3. Clasificación en tiempo real del estado operativo del motor: El modelo de Red Neuronal Artificial (ANN) realizará de forma continua la clasificación del estado del motor, identificándolo como normal o con indicios de falla según los patrones que detecte en las señales sensoriales registradas.

4. Generación de alertas tempranas y sugerencias de acción: Cuando se detecte una condición anómala, el sistema generará alertas automáticas acompañadas de recomendaciones técnicas orientadas a la inspección o intervención oportuna del equipo.

5. Registro histórico y retroalimentación del sistema: Todas las predicciones y eventos serán almacenados en una base de datos, lo cual permitirá el reentrenamiento periódico del modelo y su mejora progresiva mediante aprendizaje continuo.

4.8.2. Impacto esperado de la propuesta

Una vez implementado, el sistema propuesto permitirá a las organizaciones industriales:

- Reducir los tiempos de inactividad no planificada, al anticipar fallas con alto nivel de confiabilidad.
- Incrementar la seguridad operativa al prevenir accidentes asociados a fallas que no hayan sido detectadas a tiempo, gracias a la vigilancia continua del estado del motor.
- Optimizar las tareas de mantenimiento técnico mediante decisiones basadas en datos reales y análisis predictivos, reduciendo intervenciones innecesarias y mejorando la eficiencia operativa.

- Prolongar la vida útil de los activos electromecánicos, al realizar intervenciones únicamente cuando los datos lo justifiquen.

4.8.3. Relevancia académica y profesional

Desde el ámbito académico, esta propuesta consolidará la aplicación práctica de conocimientos en inteligencia artificial, mantenimiento industrial y procesamiento sensorial, desarrollados a lo largo de este trabajo de investigación. En el plano profesional, sentará las bases para una solución escalable y transferible, capaz de ser replicada en distintos entornos industriales, posicionando a la autora como una promotora de innovación y digitalización en el ámbito del mantenimiento predictivo.

CONCLUSIONES

- El sistema de adquisición de datos implementado permitió registrar con fidelidad variables críticas como corriente, temperatura, vibración y velocidad angular, generando una base sólida para el entrenamiento de los modelos. Esta infraestructura validó la viabilidad técnica de integrar sensores de bajo costo con plataformas de análisis inteligente.

- La implementación de algoritmos de aprendizaje automático sobre datos sensoriales permitió anticipar con eficacia la presencia de fallas en motores eléctricos, reduciendo el riesgo de paradas imprevistas y mejorando la capacidad de respuesta técnica. Estos resultados confirman que el mantenimiento predictivo basado en inteligencia artificial es una alternativa más precisa y oportuna frente a los métodos tradicionales.

- Entre los modelos evaluados, las Redes Neuronales Artificiales (ANN) mostraron el mejor rendimiento general, con una exactitud promedio del 97,0 % al analizar diferentes variables sensoriales. Su buen desempeño se refleja en su precisión, sensibilidad y capacidad de adaptación, posicionándolas como una herramienta confiable para predecir fallas en motores que operan bajo condiciones reales.

RECOMENDACIONES

- Desde el ámbito metodológico, se recomienda extender el tiempo de recolección de datos y aumentar la variabilidad de las condiciones de carga del motor, dando paso a modelos más robustos y adaptables. En este estudio se trabajó bajo un horario limitado y en un entorno controlado, lo que restringió la amplitud de escenarios operativos simulados.

- Desde la perspectiva académica, se sugiere investigar el comportamiento de los algoritmos ante condiciones de ruido, errores de sensores o pérdida parcial de datos, aspectos que no fueron considerados en este trabajo pero que reflejan situaciones comunes en entornos industriales reales.

- Desde una perspectiva práctica, se recomienda validar el modelo propuesto en otros tipos de motores eléctricos o sistemas rotativos, ya que los resultados obtenidos se basan exclusivamente en un esmeril eléctrico. Ampliar las pruebas permitiría comprobar la escalabilidad del enfoque y verificar si mantiene su efectividad en contextos distintos al evaluado.

- Se recomienda hacer una guía técnica para la implementación del sistema predictivo, orientada a personal operativo sin formación especializada en ciencia de datos. Esto facilitaría su uso real en la industria.

REFERENCIAS BIBLIOGRÁFICAS

- Adu-Amankwa, K., Attia, A. K. A., Janardhanan, M. N., & Patel, I. (2019). A predictive maintenance cost model for CNC SMEs in the era of industry 4.0. *International Journal of Advanced Manufacturing Technology*, 104(9–12), 3567–3587. <https://doi.org/10.1007/S00170-019-04094-2/METRICS>
- Argumedo Rhenals, L. F., & León Herrera, A. D. (2024). APLICACIÓN DE ALGORITMOS DE MACHINE LEARNING EN SISTEMAS DE REFRIGERACIÓN INDUSTRIAL PARA MANTENIMIENTO PREDICTIVO [Universidad de Córdoba]. www.unicordoba.edu.co
- Asociación Española de Normalización. (2018). UNE-EN 13306:2018 - Mantenimiento.
- Baradan, A. H. (2025). Predictive Maintenance of Electric Motors Using Supervised Learning Models: A Comparative Analysis. 8.
- Barrios Arce, J. I. (2022a). Inteligencia Artificial y salud. Un caso práctico. *Inteligencia Artificial y Salud. Un Caso Práctico*. <https://www.juanbarrios.com/inteligencia-artificial-y-salud-caso-practico/>
- Barrios Arce, J. I. (2022b). La matriz de confusión y sus métricas – Inteligencia Artificial – . <https://www.juanbarrios.com/la-matriz-de-confusion-y-sus-metricas/>
- Benhanifia, A., Cheikh, Z. Ben, Oliveira, P. M., Valente, A., & Lima, J. (2025). Systematic review of predictive maintenance practices in the manufacturing sector. In *Intelligent Systems with Applications* (Vol. 26). Elsevier B.V. <https://doi.org/10.1016/j.iswa.2025.200501>
- Berisha, B., Mëzriu, E., & Shabani, I. (2022). Big data analytics in Cloud computing: an overview. *Journal of Cloud Computing*, 11, 24. <https://doi.org/10.1186/s13677-022-00301-w>
- Cáceres Tello, J. (2022). “Reconocimiento de patrones y el aprendizaje no supervisado.” http://www-etsi2.ugr.es/depar/ccia/rf/www/tema1_00-01_www/node6.html.
- Chicaiza Oña, R. M., & González Jiménez, H. (2021). “Sistema de Monitoreo en Nube Para el Diagnóstico Predictivo de Fallos de Motores Eléctricos. Escuela Politécnica del Litoral.

- Dávalos Carrera, J. E., & Vazquez Matute, J. P. (2022). Determinación del estado de falla de motores eléctricos a partir del análisis de ruidos utilizando técnicas de Aprendizaje Profundo. Escuela Superior Politécnica del Litoral.
- Delecrodio, T. A. V., Neves, G. R., & Lucato, W. C. (2023). A manutenção de ativos no contexto da Indústria 4.0: uma análise bibliométrica e sistemática. *Exacta*, 21(1), 23–52. <https://doi.org/10.5585/exactaep.2021.17589>
- Dutuour, G. (2024). Tutorial: LM393 o FC-03 o FZ0888 – Robotica de la Costa. <https://roboticadelacosta.uy/tutorial-lm393-o-fc-03/>
- Guerrero Cano, manuel, Luque Sendra, A., Lama Ruiz, J. R., & Córdoba Roldán, A. (2019). PREDICTIVE MAINTENANCE USING MACHINE LEARNING TECHNIQUES. Creative Commons. <https://creativecommons.org/licenses/by-nc-nd/4.0/>
- Lawrence, P. A. (2024). Predictive Maintenance: Leveraging Machine Learning for Equipment Health Monitoring. <https://www.researchgate.net/publication/377411657>
- Lemay, P. (2022, August 3). Ventajas de Industrial IoT para los fabricantes. Transformación Digital. https://tulip.co/es/blog/benefits-of-industrial-iot/?utm_source=chatgpt.com
- Lithium, I. (2022). Giroscopio MPU6050 con Arduino Pablo Samayoa Autor: Lithium IONDe... <https://www.electronicadiy.com/blogs/tutoriales-y-blog/giroscopio-mpu6050-con-arduino>
- Llamas, L. (2025). Sensor de corriente eléctrica no invasivo con Arduino y SCT-013. <https://www.luisllamas.es/arduino-sensor-corriente-sct-013/>
- Luna Gonzalez, J. (2024). Tipos de aprendizaje automático. La Inteligencia Artificial (IA) está en... | by Javier Luna Gonzalez | SoldAI | Medium. <https://medium.com/soldai/tipos-de-aprendizaje-autom%C3%A1tico-6413e3c615e2>
- Malindi, S. N., & Kuno, V. (2025). Hybrid Predictive Maintenance for Electric Motor Bearings: Integrating Machine Learning and Physics-Based Models. *International Journal on Science and Technology*, 12. www.ijst.org
- Mirani, A. A., Velasco-Hernandez, G., Awasthi, A., & Walsh, J. (2022). Key Challenges and Emerging Technologies in Industrial IoT Architectures: A Review. *Sensors (Basel, Switzerland)*, 22(15), 5836. <https://doi.org/10.3390/S22155836>

- Mohammed, N. A., Abdulateef, O. F., & Hamad, A. H. (2023). An IoT and Machine Learning-Based Predictive Maintenance System for Electrical Motors. *Journal European Des Systemes Automatises*, 56(4), 651–656. <https://doi.org/10.18280/jesa.560414>
- Patel, C., Kashif Bashir, A., Alzubi, A. A., & Jhaveri, R. H. (2015). EBAKE-SE: A Novel ECC Based Authenticated Key Exchange between Industrial IoT Devices using Secure Element. www.elsevier.com/locate/dcan
- Pisano, A. (2018). *Internet de la Cosas*. Universidad de San Andrés.
- Pourghoraba, A., KhajueeZadeh, M., Amini, A., Vahedi, A., Agah, G. R., & Rahideh, A. (2024). Model-Agnostic Meta-Learning for Fault Diagnosis of Induction Motors in Data-Scarce Environments with Varying Operating Conditions and Electric Drive Noise. <https://doi.org/10.1109/TEC.2025.3556100>
- Quispe Oqueña, E. C. (2023). *El Hombre y la Máquina*. <http://www.redalyc.org/articulo.oa?id=47812406007>
- Rouhiainen, L. P. (2024). *INTELIGENCIA ARTIFICIAL* (Editorial Planeta, Ed.; noviembre de 2024). www.planetadelibros.com
- Sanseviero, O. (2023). *AI en 3 minutos: Tipos de Machine Learning*. <https://medium.com/ai-learners/ai-en-3-minutos-tipos-de-machine-learning-945b708ac78>
- Shrawat, D., & Gill, N. S. (2019). Smart sensors: Analysis of different types of IoT sensors. *Proceedings of the International Conference on Trends in Electronics and Informatics, ICOEI 2019*, 523–528. <https://doi.org/10.1109/ICOEI.2019.8862778>
- Tightiz, L., & Yoo, J. (2022). Towards Latency Bypass and Scalability Maintain in Digital Substation Communication Domain with IEC 62439-3 Based Network Architecture. *Sensors (Basel, Switzerland)*, 22(13), 4916. <https://doi.org/10.3390/S22134916>
- Tovar Porras, E. (2021). *Tipos de mantenimiento industrial*.
- Xu, L., Teoh, S. S., & Ibrahim, H. (2024). A deep learning approach for electric motor fault diagnosis based on modified InceptionV3. *Scientific Reports*, 14(1), 15. <https://doi.org/10.1038/s41598-024-63086-9>

Yang, C., Huang, Q., Li, Z., Liu, K., & Hu, F. (2017). Big Data and cloud computing: innovation opportunities and challenges. *International Journal of Digital Earth*, 10(1), 13–53. https://doi.org/10.1080/17538947.2016.1239771/ASSET/BB6F2193-41CB-4B2B-8243-55305703C05D/ASSETS/IMAGES/TJDE_A_1239771_F0001_C.JPG

Zamanzadeh Darban, Z., & Webb, G. I. (2024). Deep Learning for Time Series Anomaly Detection: A Survey. 2024. *Deep Learning for Time Series Anomaly Detection: A Survey*. *ACM Comput. Surv*, 57, 42. <https://doi.org/10.1145/3691338>