



GRADO EN INGENIERÍA EN TECNOLOGÍAS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

Aplicación de técnicas de Machine Learning para el diagnóstico del comportamiento de bombas de agua de circulación de una central de generación eléctrica.

Autor: Inés Rodríguez Aguilar

Director: Miguel Ángel Sanz Bobi

Madrid

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título

Aplicación de técnicas de Machine Learning para el diagnóstico del comportamiento de bombas de agua de circulación de una central de generación eléctrica

en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el

curso académico 2024/25 es de mi autoría, original e inédito y

no ha sido presentado con anterioridad a otros efectos.

El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido tomada de otros documentos está debidamente referenciada.



Fdo.: Inés Rodríguez Aguilar

Fecha: 18/ 06/ 2025

Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO

Firmado por SANZ BOBI MIGUEL ANGEL - ***6599** el día
20/06/2025 con un certificado emitido por AC FNMT Usuarios

Fdo.: Miguel Ángel Sanz Bobi

Fecha://



GRADO EN INGENIERÍA EN TECNOLOGÍAS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

Aplicación de técnicas de Machine Learning para el diagnóstico del comportamiento de bombas de agua de circulación de una central de generación eléctrica.

Autor: Inés Rodríguez Aguilar

Director: Miguel Ángel Sanz Bobi

Madrid

APLICACIÓN DE TÉCNICAS DE MACHINE LEARNING PARA EL DIAGNÓSTICO DEL COMPORTAMIENTO DE BOMBAS DE AGUA DE CIRCULACIÓN DE UNA CENTRAL DE GENERACIÓN ELÉCTRICA.

Autor: Inés Rodríguez Aguilar.

Director: Miguel Ángel Sanz Bobi.

Entidad Colaboradora: ENDESA

RESUMEN DEL PROYECTO

Este proyecto presenta el desarrollo de un sistema de diagnóstico predictivo aplicado a las bombas de circulación de agua en una central de ciclo combinado, basado en aprendizaje por refuerzo (TD3). Se emplean datos reales (2020–2023) para entrenar agentes que predicen valores de temperatura, presión y desplazamiento, y se traduce el error de predicción en curvas de estrés acumulado. Con ello, se permite detectar anomalías con antelación y optimizar las tareas de mantenimiento, lo que resulta en una reducción de los costes operativos.

Palabras clave: Diagnóstico predictivo, aprendizaje por refuerzo, TD3, estrés acumulado

1. Introducción

La disponibilidad y fiabilidad de las bombas de circulación de agua son esenciales para el funcionamiento continuo de una central de ciclo combinado, ya que garantizan el retorno del condensado a las calderas y mantienen la estabilidad del proceso térmico. Cuando estas bombas fallan sin aviso, pueden desencadenarse paradas imprevistas que provocan costes elevados, retrasos en la generación y riesgos en el resto de los equipos. Los métodos convencionales basados en límites estáticos o inspecciones programadas suelen generar falsas alarmas o no detectar degradaciones progresivas.

Frente a estas limitaciones, este trabajo explora un enfoque basado en inteligencia artificial, donde un gemelo digital integrado con un agente de aprendizaje por refuerzo aprende el comportamiento normal de la bomba a partir de datos reales. En lugar de necesitar etiquetar los fallos manualmente, el agente se entrena para predecir la señal siguiente y reporta anomalías cuando su predicción difiere de la medida real. De este modo, se consigue una monitorización adaptativa capaz de anticipar modos de fallo antes de que se conviertan en incidentes críticos.

2. Definición del proyecto

El propósito principal de este proyecto es construir un sistema capaz de anticipar modos de fallo en las bombas de circulación sin depender de datos etiquetados manualmente ni de inspecciones constantes. Para ello, se dispone de registros históricos de la planta correspondientes al periodo 2020–2023. En una fase inicial de preprocesado, se descartan los tramos en los que las bombas arranquen o se detengan y se limpian los datos

eliminando valores anómalos y normalizando las señales para que todas las variables compartan una escala común.

Una vez preparados los datos, se definen entornos de simulación en los que el agente TD3 toma como entrada el conjunto de mediciones disponibles en cada instante y genera una predicción sobre el valor siguiente. Cada vez que su predicción difiere del valor real, se añade un incremento al indicador llamado estrés. Al acumular todos estos incrementos, se obtiene la curva de estrés acumulado.

Cuando esta curva muestra un comportamiento ascendente prolongado, se interpreta como una señal de que el sistema comienza a desviarse de su funcionamiento normal. De esta forma, se obtiene un mecanismo de alerta temprana que permite programar tareas de mantenimiento antes de que el fallo se materialice.

3. Descripción del modelo

El modelo está basado en el algoritmo TD3, una técnica de aprendizaje por refuerzo que ha demostrado funcionar muy bien en entornos continuos. En este caso, el entorno está formado por secuencias temporales de variables como potencia, temperatura y presión. Se ha optado por una arquitectura simple, con un modelo por cada señal relevante, para facilitar la interpretación.

El entrenamiento se realiza exclusivamente con datos del año 2020. A partir de ahí, los modelos se aplican a los años posteriores (2021–2023) para evaluar su capacidad de generalización. En cada paso, el agente recibe un conjunto de observaciones y devuelve una predicción, que se compara con el valor real para calcular el error. Este error se normaliza y se compara con un umbral estadístico definido por $\pm 3\sigma$ (tres veces la desviación estándar). Solo cuando la desviación supera ese límite, se interpreta como señal de estrés y se acumula en la curva correspondiente.

El proceso general seguido por el sistema se resume en la Ilustración 1, donde se representa cómo las señales de planta se transforman en predicciones, errores y finalmente en la curva de riesgo.

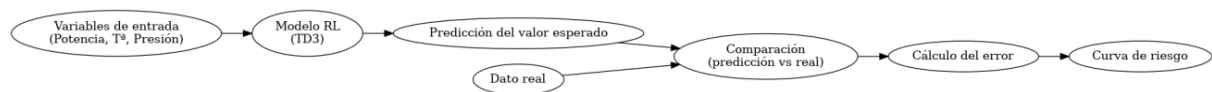


Ilustración 1 - Esquema conceptual del sistema de diagnóstico basado en aprendizaje por refuerzo

4. Resultados

A lo largo del proyecto se han desarrollado seis modos de fallo diferentes, cada uno centrado en un comportamiento anómalo concreto de una bomba. Para cada modo se han entrenado uno o varios modelos independientes, y en total se han evaluado diez modelos. En todos los casos, las curvas de estrés acumulado muestran una evolución coherente con la aparición y desarrollo de los fallos.

En los modos donde las señales responden de forma más clara ante una desviación (por ejemplo, un aumento de temperatura), los modelos detectan el fallo de forma rápida y consistente. En otros casos, donde los cambios son más graduales o sutiles, el sistema es capaz de acumular pequeñas desviaciones hasta generar una alerta progresiva.

Además, se ha observado que al combinar varios modelos que analizan distintas señales del mismo modo de fallo, se obtiene una visión más completa de lo que está ocurriendo. La Ilustración 2 muestra un ejemplo de cómo los distintos modelos acumulan estrés en momentos similares, pero con diferentes intensidades, reforzando la idea de que el sistema es capaz de integrar distintas perspectivas del mismo problema.

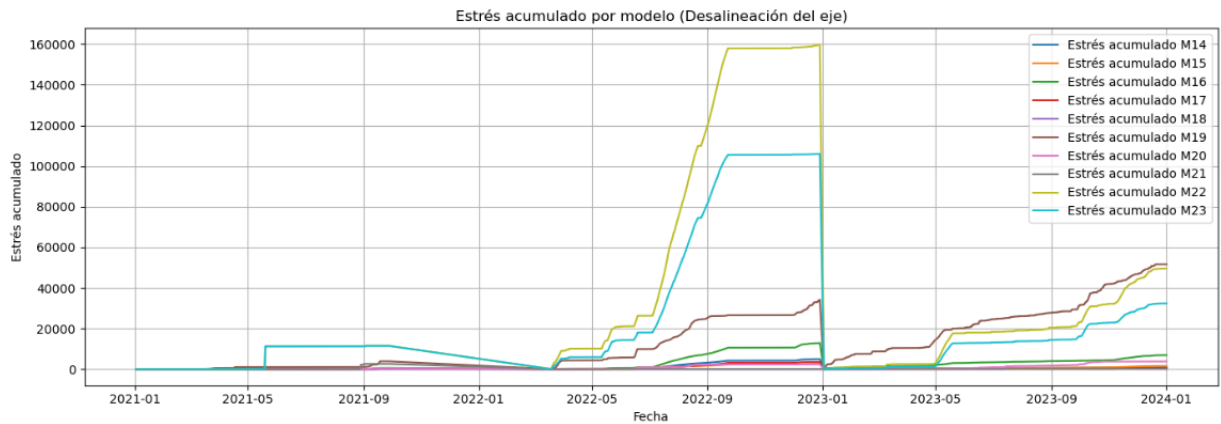


Ilustración 2 - Estrés acumulado total por modelo del modo del fallo “Desalineación del eje” en el periodo 2021–2023

5. Conclusiones

El sistema propuesto ha demostrado ser capaz de detectar con antelación comportamientos anómalos en señales clave de las bombas de circulación, usando únicamente datos históricos sin etiquetas. A lo largo del proyecto se ha comprobado que el enfoque basado en aprendizaje por refuerzo no solo es viable, sino que resulta eficaz incluso en condiciones reales de planta, donde los datos presentan ruido, interrupciones y otras complicaciones habituales.

Una de las principales aportaciones del proyecto ha sido validar que este tipo de modelos pueden adaptarse a fallos muy distintos, y que los indicadores de estrés acumulado ofrecen una forma intuitiva de seguir su evolución. Además, se ha conseguido que el sistema sea fácil de integrar en entornos reales, al apoyarse en señales que ya se recogen de forma rutinaria.

Se han cumplido los objetivos marcados: desarrollar un sistema flexible, validado con datos reales, y que ofrezca valor en la toma de decisiones de mantenimiento. El resultado es un sistema que no pretende sustituir al criterio experto, pero sí aportar una herramienta adicional para anticiparse a los fallos y actuar antes de que se materialicen.

6. Referencias

- [1] Sanz-Bobi, M. A., et al. “Anomaly Detection of a Cooling Water Pump of a Power Plant Based on its Virtual Digital Twin Constructed with Deep Learning Techniques.” *PHM Society European Conference*, 2024.
- [2] Stable Baselines3 Documentation. <https://stable-baselines3.readthedocs.io/en/master/>
- [3] Fracttal. “Guía completa del mantenimiento industrial.” <https://www.fracttal.com/es/blog/guia-completa-del-mantenimiento-industrial-estrategias-herramientas-y-mejores-practicas>

APPLICATION OF MACHINE LEARNING TECHNIQUES FOR THE DIAGNOSTICS OF CIRCULATION WATER PUMP BEHAVIOR IN A POWER GENERATION PLANT

Author: Rodríguez Aguilar, Inés.

Supervisor: Sanz Bobi, Miguel Ángel.

Collaborating Entity: ENDESA

ABSTRACT

This project presents the development of a predictive diagnostics system for circulation water pumps in a combined cycle power plant, based on reinforcement learning (TD3). Real operational data (2020–2023) are used to train agents that predict key signals such as temperature, pressure and displacement. The prediction error is translated into cumulative stress curves, allowing early detection of anomalies and optimization of maintenance tasks, which leads to reduced operational costs.

Keywords: Predictive diagnostics, reinforcement learning, TD3, cumulative stress.

1. Introduction

The availability and reliability of the water circulation pumps are essential for the continuous operation of a combined-cycle power plant, as they ensure the condensate's return to the boilers and maintain the stability of the thermal process. When these pumps fail without warning, unplanned shutdowns can occur, resulting in high costs, generation delays, and risks to other equipment. Conventional methods based on static thresholds or scheduled inspections often produce false alarms or fail to detect gradual degradations.

In contrast to these limitations, this work explores an artificial intelligence based approach in which a digital twin, integrated with a reinforcement learning agent, learns the pump's normal behavior from real data. Instead of requiring manual fault labeling, the agent is trained to predict the next signal and flags anomalies when its prediction deviates from the actual measurement. This enables adaptive monitoring capable of anticipating failure modes before they evolve into critical incidents.

2. Project definition

The main goal is to build a system capable of anticipating pump failures without requiring labeled data or frequent inspections. The project uses historical data from 2020 to 2023. In the preprocessing phase, segments where pumps are starting or stopping are excluded, and the data is cleaned by removing outliers and normalizing the variables to a common scale.

Once the data are prepared, simulation environments are created in which the TD3 agent receives the set of measurements available at each time step and produces a prediction of the next value. Whenever its prediction deviates from the actual measurement, an increment is

added to the indicator called stress. By accumulating all of these increments, the cumulative stress curve is obtained.

When this curve exhibits a prolonged upward trend, it is interpreted as a signal of deviation from normal behavior. This results in an early warning mechanism that allows maintenance to be planned before a fault manifests.

3. Model description

The model is based on the TD3 algorithm, a reinforcement learning technique well suited to continuous environments. In this application, the environment consists of time series of plant variables such as power, temperature and pressure. A simple architecture is used, there is one model per signal, to keep results interpretable.

Training is done using only data from 2020. The models are then applied to data from 2021 to 2023 to evaluate generalization. At each step, the agent receives observations and returns a prediction, which is compared to the actual value to compute the error.

This error is normalized and compared to a statistical threshold of $\pm 3\sigma$ (three times the standard deviation). Only when the deviation exceeds this threshold is it considered stress and added to the cumulative curve.

The overall process is summarized in Figure 1, which shows how plant signals are transformed into predictions, errors, and ultimately, a risk curve.



Figure 1 - Conceptual diagram of the diagnostics system based on reinforcement learning

4. Results

The project covers six different failure modes, each focused on a specific abnormal pump behavior. For each mode, one or more independent models have been trained, with a total of ten models evaluated. In every case, the cumulative stress curves reflect the onset and progression of faults with notable consistency.

When the signal reacts clearly to a fault (e.g., an increase in temperature), the model responds quickly and reliably. In more subtle cases, the system gradually accumulates small deviations until an alert is generated.

Combining several models that analyze different signals for the same failure mode enhances diagnostic coverage. Figure 2 shows an example where different models accumulate stress during similar periods but with varying intensity, reinforcing the idea that the system provides a more complete picture when integrating multiple perspectives.

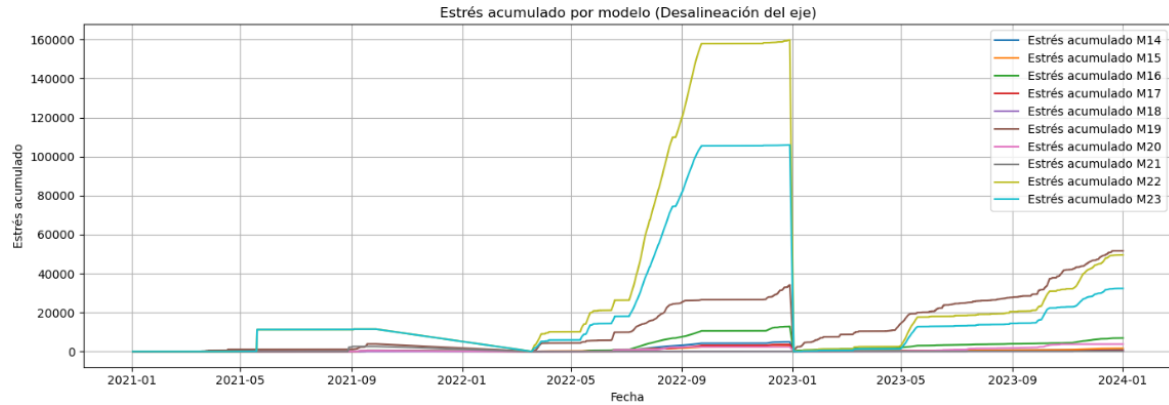


Figure 2 - Cumulative stress by model for the failure mode “Shaft misalignment” (2021–2023)

5. Conclusion

The system has proven capable of detecting early-stage anomalies in critical pump signals using only historical data, and no manual labeling is required. It has been shown that reinforcement learning, when properly applied, is not only viable but effective even under real industrial conditions, where data is noisy and imperfect.

A key achievement is validating that these models can be adapted to very different failure modes, and that cumulative stress indicators provide an intuitive way to track fault development over time. The system is also designed for practical integration, relying only on signals already available in the plant.

All project goals have been met: a flexible diagnostics system has been built, validated on real data, and shown to provide actionable value. While not intended to replace expert judgement, it adds a valuable tool to support maintenance planning and avoid unexpected failures.

6. References

- [1] Sanz-Bobi, M. A., et al. “Anomaly Detection of a Cooling Water Pump of a Power Plant Based on its Virtual Digital Twin Constructed with Deep Learning Techniques.” *PHM Society European Conference*, 2024.
- [2] Stable Baselines3 Documentation. <https://stable-baselines3.readthedocs.io/en/master/>
- [3] Fractal. “Guía completa del mantenimiento industrial.” <https://www.fractal.com/es/blog/guia-completa-del-mantenimiento-industrial-estrategias-herramientas-y-mejores-practicas>

Índice de la memoria

Capítulo 1. Introducción	6
1.1 Contexto	6
1.2 limitaciones de los métodos de mantenimiento actuales	7
1.3 ventajas del aprendizaje por refuerzo	8
1.4 Aplicación al caso real	9
1.5 Aplicación académica, técnica y profesional	10
1.6 relevancia económica, social y ambiental	10
1.7 objetivo.....	12
Capítulo 2. Descripción de las Tecnologías.....	13
2.1 aprendizaje por refuerzo	13
2.2 gymnasium	14
2.3 Stable-baselines3	15
2.4 algoritmo td3	15
2.5 Estructura general.....	16
Capítulo 3. Estado de la Cuestión	17
Capítulo 4. Definición del Trabajo	20
4.1 Justificación.....	20
4.1.1 organización del sistema	21
4.1.2 Por qué este enfoque tiene sentido	21
4.2 Objetivos	22
4.3 Metodología.....	23
Capítulo 5. Sistema/Modelo Desarrollado.....	25
5.1 visión general del sistema.....	25
5.2 preparación de los datos	26
5.3 diseño del entorno de aprendizaje	27
5.4 configuración del algoritmo td3	28
5.5 relación entre modelos y modos de fallo.....	29
5.6 relación entre modelos y modos de fallo.....	33

Capítulo 6. Análisis de Resultados.....	35
6.1 análisis por modos de fallo	35
6.1.1 Desgaste del cojinete del lado opuesto al acoplamiento del motor	36
6.1.2 Cojinete desgastado del lado del acoplamiento del motor.....	39
6.1.3 Cojinete desgastado del lado opuesto al acoplamiento de la bomba.....	42
6.1.4 Cojinete desgastado del lado del acoplamiento de la bomba	46
6.1.5 Cojinete de empuje desgastado	50
6.1.6 Desalineación del eje	53
Capítulo 7. Conclusiones y Trabajos Futuros.....	56
Capítulo 8. Bibliografía.....	59
ANEXO I: ALINEACIÓN DEL PROYECTO CON LOS ODS	61
ANEXO II: Implementación del sistema para el caso de desalineación del eje	63

Índice de figuras

Figura 1. Comparativa entre diferentes estrategias de mantenimiento.....	8
Figura 2. Esquema conceptual del sistema de diagnóstico basado en aprendizaje por refuerzo	10
Figura 3. Esquema de un sistema de aprendizaje por refuerzo [4].....	14
Figura 4. Error de predicción y bandas de confianza del modelo M22 durante el entrenamiento	37
Figura 5. Estrés acumulado del modelo M22 durante el año 2021	37
Figura 6. Estrés acumulado del modelo M22 durante el año 2022	38
Figura 7. Error normalizado y bandas de confianza durante el año 2023	38
Figura 8. Estrés acumulado total del modo de fallo 2 en el periodo 2021–2023	39
Figura 9. Error de predicción y bandas de confianza del modelo M21 durante el entrenamiento	40
Figura 10. Estrés acumulado del modelo M21 durante el año 2021	40
Figura 11. Estrés acumulado del modelo M21 durante el año 2022	41
Figura 12. Estrés acumulado del modelo M21 durante el año 2023	41
Figura 13. Estrés acumulado total del modo de fallo 3 en el periodo 2021–2023	42
Figura 14. Error de predicción y bandas de confianza del modelo M16 durante el entrenamiento	43
Figura 15. Estrés acumulado del modelo M16 durante el año 2021	44
Figura 16. Estrés acumulado del modelo M17 durante el año 2022	44
Figura 17. Estrés acumulado del modelo M20 durante el año 2023	45
Figura 18. Estrés acumulado total por modelo del modo de fallo 4 en el periodo 2021–2023	46
Figura 19. Error de predicción y bandas de confianza del modelo M14 durante el entrenamiento	47
Figura 20. Estrés acumulado del modelo M19 durante el año 2021	47
Figura 21. Estrés acumulado del modelo M15 durante el año 2022	48
Figura 22. Estrés acumulado del modelo M14 durante el año 2023	48

Figura 23. Estrés acumulado total por modelo del modo de fallo 5 en el periodo 2021–2023	49
Figura 24. Estrés acumulado total del modo de fallo 5 en el periodo 2021–2023	49
Figura 25. Error de predicción y bandas de confianza del modelo M18 durante el entrenamiento	50
Figura 26. Estrés acumulado del modelo M23 durante el año 2021	51
Figura 27. Estrés acumulado del modelo M23 durante el año 2022	51
Figura 28. Estrés acumulado del modelo M18 durante el año 2023	52
Figura 29. Estrés acumulado total por modelo del modo de fallo 5 en el periodo 2021–2023	52
Figura 30. Estrés acumulado total del modo de fallo 6 en el periodo 2021–2023	53
Figura 31. Estrés acumulado total por modelo del modo de fallo 8 en el periodo 2021–2023	54
Figura 32. Estrés acumulado total del modo de fallo 8 en el periodo 2021–2023	55

Índice de tablas

Tabla 1. Relación entre modos de fallo, modelos entrenados y variable predicha.....	32
---	----

Capítulo 1. INTRODUCCIÓN

En una central eléctrica de ciclo combinado, el funcionamiento eficiente del sistema depende de la coordinación de muchos elementos. Uno de ellos, que a menudo no recibe tanta atención como otros más visibles, pero que resulta fundamental, son las bombas de agua de circulación. Su función principal es enfriar el vapor que sale de la turbina haciendo circular agua por el condensador. Gracias a este proceso se consigue que el vapor se condense correctamente y el ciclo térmico continúe funcionando de manera estable y eficiente.

Estas bombas, aunque a simple vista puedan parecer componentes auxiliares, tienen un impacto enorme en la operación de la planta. Si alguna de ellas comienza a comportarse de forma anómala, pueden surgir desde caídas en la eficiencia hasta paradas no programadas que, en muchos casos, acaban derivando en pérdidas económicas importantes y en daños adicionales a otros equipos. Por eso, contar con herramientas que ayuden a detectar cualquier comportamiento anómalo lo antes posible es una necesidad real en la industria.

En este proyecto se plantea una solución basada en inteligencia artificial, más concretamente en aprendizaje por refuerzo, para desarrollar un sistema de diagnóstico inteligente que permita anticipar posibles fallos en estas bombas. Este enfoque parte del uso de datos reales recopilados en planta, y busca construir un modelo que actúe como un "gemelo digital" del equipo. A lo largo de este capítulo se contextualiza el problema, se explican las limitaciones de las soluciones actuales, y se justifica por qué se ha optado por este tipo de algoritmo.

1.1 CONTEXTO

Dentro del ciclo de funcionamiento de una central de generación eléctrica, las bombas de agua de circulación tienen un rol clave. Su tarea es mantener en movimiento el caudal necesario para el proceso de condensación, algo que influye directamente en el rendimiento

térmico de la instalación. Si esta parte del sistema no cumple bien su función, toda la cadena se ve afectada.

El problema no es solo técnico, sino también económico y organizativo. Un fallo en una bomba obliga a realizar paradas que no estaban previstas, lo que interrumpe la producción y puede suponer pérdidas de miles de euros. A esto hay que sumarle los costes de reparación, la disponibilidad limitada de recambios, y la posible afectación a la planificación de tareas de mantenimiento del resto de la instalación. Por eso, las empresas del sector están cada vez más interesadas en implementar soluciones predictivas, que no solo informen de lo que ha fallado, sino que avisen antes de que ocurra.

1.2 LIMITACIONES DE LOS MÉTODOS DE MANTENIMIENTO ACTUALES

Los métodos de mantenimiento más utilizados hoy en día suelen clasificarse como correctivos, preventivos o predictivos. El correctivo actúa cuando ya ha ocurrido el fallo. Aunque es el más simple de implementar, también es el más costoso y arriesgado. El preventivo, por otro lado, programa revisiones o reemplazos de forma periódica, pero a menudo se realizan intervenciones innecesarias o demasiado tarde. En cuanto al predictivo, se basa en señales como vibraciones, temperatura, presión o consumo, con el objetivo de anticiparse al fallo.

Este último es el enfoque más avanzado, pero aún presenta limitaciones importantes. En muchos casos, se depende de que un técnico defina manualmente los umbrales de alerta. Además, el análisis suele hacerse variable por variable, sin tener en cuenta la relación entre ellas. Si bien se han desarrollado sistemas de análisis más complejos basados en machine learning supervisado, estos requieren que el fallo ya haya ocurrido antes para poder etiquetar los datos, lo cual no siempre es posible ni deseable.

Esto hace que sea muy difícil detectar fallos que se desarrollan de forma lenta, progresiva y silenciosa, sin síntomas evidentes. Y es precisamente en estos casos donde los modelos

inteligentes que aprenden del comportamiento global del equipo pueden marcar la diferencia.

La Figura 1 muestra una comparativa visual entre los distintos tipos de mantenimiento, representando de forma simplificada dos aspectos clave: el nivel de anticipación que permite cada estrategia y su coste relativo. Esto permite entender por qué las técnicas basadas en inteligencia artificial resultan especialmente interesantes desde el punto de vista operativo y económico.

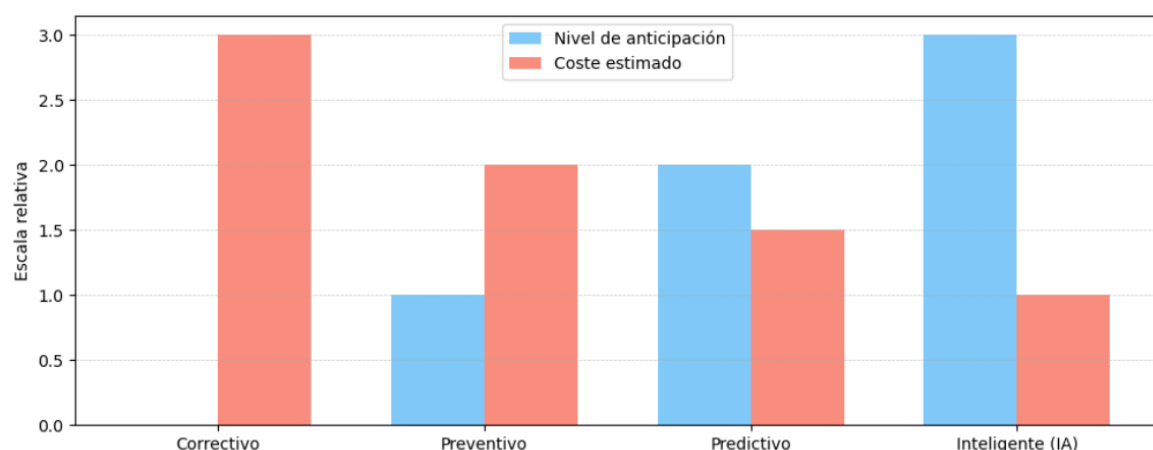


Figura 1. Comparativa entre diferentes estrategias de mantenimiento

1.3 VENTAJAS DEL APRENDIZAJE POR REFUERZO

El aprendizaje por refuerzo es una técnica dentro de la inteligencia artificial que permite que un agente aprenda a actuar dentro de un entorno interactivo, a través de prueba y error. No necesita saber desde el principio qué está bien o qué está mal, sino que va ajustando sus decisiones en función de las recompensas que va recibiendo.

En este proyecto se entrena un agente para que aprenda a predecir cómo debería comportarse una bomba en condiciones normales, utilizando únicamente datos reales de funcionamiento. El agente observa un estado del sistema (por ejemplo, la potencia térmica, la temperatura del motor, la presión de descarga...) y predice una variable objetivo, como la temperatura de un

cojinete. A medida que compara su predicción con el valor real, se le asigna una recompensa, que puede ser positiva si ha acertado o negativa si ha fallado. Así, con el tiempo, el agente aprende a hacer predicciones más precisas.

Para esto se ha utilizado el algoritmo TD3 (Twin Delayed Deep Deterministic Policy Gradient), que está especialmente pensado para entornos donde las acciones son continuas y las relaciones entre variables no son lineales. Este tipo de algoritmo ya se ha aplicado con éxito en problemas de robótica o control autónomo, pero su aplicación en entornos industriales como este todavía es poco común, lo que añade valor al trabajo realizado.

1.4 APLICACIÓN AL CASO REAL

El punto de partida son los modos de fallo descritos en una metodología previa basada en análisis de fallos y efectos (FMEA). Cada modo de fallo se caracteriza por una o varias variables que indican su aparición o evolución. Por ejemplo, si el cojinete del motor empieza a desgastarse, es probable que la temperatura o la vibración aumenten de forma gradual.

Lo que se hace en este proyecto es entrenar uno o varios modelos para cada modo de fallo. Estos modelos aprenden a predecir cuál sería el valor esperado de la variable objetivo si el sistema estuviera funcionando de forma normal. Si en algún momento la predicción y el valor real empiezan a diferir de forma significativa, se interpreta como un comportamiento anómalo.

A partir de estas diferencias (errores de predicción) se construyen curvas de riesgo que permiten hacer un seguimiento del estado de salud de la bomba a lo largo del tiempo. Estas curvas son especialmente útiles para comprobar si una acción de mantenimiento ha tenido efecto, ya que permiten visualizar si el riesgo ha bajado después de la intervención.

Para entender mejor cómo se articula todo el proceso, desde la entrada de datos hasta la obtención de las curvas de riesgo, se muestra en la Figura 2 un esquema conceptual del sistema propuesto. En él puede verse cómo el modelo basado en aprendizaje por refuerzo

recibe datos del sistema, genera una predicción, la compara con el valor real y, a partir del error, genera una estimación del riesgo de fallo.



Figura 2. Esquema conceptual del sistema de diagnóstico basado en aprendizaje por refuerzo

1.5 APLICACIÓN ACADÉMICA, TÉCNICA Y PROFESIONAL

Desde el punto de vista académico, el proyecto aporta valor porque explora la aplicación práctica del aprendizaje por refuerzo en un caso real de mantenimiento industrial, algo poco documentado hasta el momento. No se trata de un entorno de simulación, sino de datos reales tomados de una planta en operación, lo que representa un reto añadido por el ruido, los datos perdidos o las condiciones variables.

A nivel técnico, la motivación es mejorar la capacidad de detectar fallos antes de que generen daños, haciendo el mantenimiento más eficiente, adaptativo y específico. Es un paso más hacia la automatización de tareas de supervisión y hacia la toma de decisiones basada en datos.

También tiene una motivación personal y profesional. Poder aplicar técnicas avanzadas de IA a un problema real, con impacto en la industria energética, supone una oportunidad de aprendizaje y una forma de contribuir a un sector que está en plena transformación tecnológica.

1.6 RELEVANCIA ECONÓMICA, SOCIAL Y AMBIENTAL

Este proyecto no solo tiene implicaciones técnicas o académicas, sino también un impacto directo en varios aspectos clave del funcionamiento industrial, como la economía, la sociedad y el medio ambiente.

Desde un punto de vista económico, anticipar fallos en las bombas de agua de circulación permite reducir de forma significativa los costes operativos. Si se consigue detectar un problema antes de que se convierta en un fallo grave, se pueden evitar paradas no programadas, intervenciones de urgencia o sustituciones de componentes que podrían haberse alargado en el tiempo. Además, se optimiza el uso de recursos humanos y materiales, ya que las tareas de mantenimiento pueden planificarse con más criterio, evitando inspecciones innecesarias o reparaciones preventivas que no aportan valor real.

En el ámbito social, un sistema de mantenimiento más eficiente y fiable contribuye directamente a la estabilidad del suministro eléctrico. Las centrales eléctricas que sufren menos incidencias ofrecen una mayor disponibilidad, lo que se traduce en una red más robusta y con menos riesgo de cortes. Esta estabilidad energética no solo beneficia a los consumidores individuales, sino también a sectores clave como la sanidad, la industria o los servicios. Además, la implementación de sistemas inteligentes basados en IA en entornos industriales fomenta la modernización del sector, abre nuevas oportunidades laborales y contribuye a la formación de perfiles profesionales más especializados.

Por último, el impacto ambiental de este tipo de soluciones también es muy relevante. Cuando una bomba está funcionando fuera de sus condiciones óptimas, suele consumir más energía de la necesaria, genera más desgaste en otras partes del sistema y, en general, reduce la eficiencia del proceso. Detectar a tiempo estos desajustes permite actuar antes de que se produzca un daño mayor, evitando consumos excesivos y emisiones asociadas. Además, al alargar la vida útil de los componentes y reducir la necesidad de sustituciones, también se minimiza la generación de residuos industriales.

En conjunto, este proyecto busca no solo mejorar el mantenimiento de un equipo concreto, sino contribuir a un modelo industrial más eficiente, sostenible y alineado con las necesidades reales de las plantas energéticas actuales.

1.7 OBJETIVO

El objetivo final de este proyecto es construir un sistema de diagnóstico inteligente basado en aprendizaje por refuerzo que actúe como un gemelo digital de las bombas de agua de circulación. Este sistema debe ser capaz de aprender cómo debería comportarse la bomba en condiciones normales, detectar cualquier desviación que pueda indicar un fallo, y proporcionar herramientas que ayuden a tomar decisiones de mantenimiento más acertadas.

También se pretende demostrar que este tipo de enfoque puede adaptarse a otros equipos de la planta o incluso a otras industrias. La idea es que este trabajo sirva como base para futuros desarrollos, y como ejemplo de cómo la inteligencia artificial puede integrarse poco a poco en procesos industriales reales.

Capítulo 2. DESCRIPCIÓN DE LAS TECNOLOGÍAS

Antes de explicar cómo se ha desarrollado el sistema de diagnóstico, es importante detenerse a explicar qué tecnologías se han utilizado y qué papel cumple cada una dentro del proyecto. No hace falta entrar en detalles técnicos muy complejos, pero sí resulta útil saber qué herramientas hay detrás, sobre todo si no se está familiarizado con el aprendizaje por refuerzo o con los entornos de simulación. Este capítulo está pensado para que lo que viene después se entienda mejor y con menos barreras técnicas.

2.1 APRENDIZAJE POR REFUERZO

El aprendizaje por refuerzo es una rama de la inteligencia artificial que se basa en un principio muy intuitivo: aprender a partir de la experiencia. Un agente interactúa con un entorno, toma decisiones, observa el resultado y recibe una señal (positiva o negativa) que le indica si lo que ha hecho le ha ido bien o mal. A partir de ahí, y tras muchos intentos, va ajustando su comportamiento para maximizar su recompensa.

En este proyecto, se ha utilizado el aprendizaje por refuerzo no para que el agente tome decisiones de control, como ocurre en muchos casos, sino para que aprenda a predecir cómo debería comportarse una determinada variable del sistema bajo condiciones normales. Por ejemplo, si una temperatura determinada suele moverse entre ciertos valores cuando todo va bien, el agente puede aprender a identificar ese patrón. Cuando lo que ocurre en la planta empieza a desviarse demasiado de esa predicción, se interpreta como una posible señal de fallo.

Este enfoque tiene una ventaja importante: el modelo no necesita reglas fijas ni umbrales predefinidos, sino que aprende directamente del comportamiento real del sistema. Esto lo hace más flexible y más capaz de adaptarse a distintos modos de funcionamiento. En la Figura 3 se muestra un esquema simplificado del ciclo típico del aprendizaje por refuerzo,

en el que el agente observa el estado del entorno, toma una acción, recibe una recompensa y vuelve a observar el entorno. A través de este bucle constante, el agente va aprendiendo qué es lo que le conviene hacer.

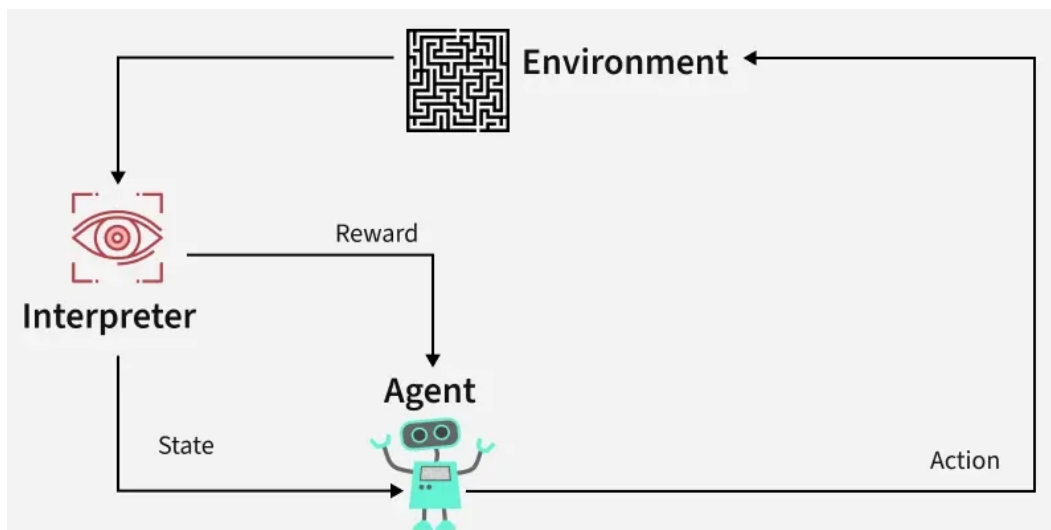


Figura 3. Esquema de un sistema de aprendizaje por refuerzo [4]

2.2 GYMNASIUM

Gymnasium es una librería de Python que permite crear entornos para entrenar agentes de aprendizaje por refuerzo. En este tipo de proyectos, el entorno es una parte fundamental, porque es lo que define lo que el agente puede observar, qué acciones puede tomar y cómo se le recompensa en función de su comportamiento. Si no existe un entorno bien definido, el agente no tiene un espacio sobre el que aprender.

En este trabajo se ha creado un entorno personalizado utilizando Gymnasium a partir de datos reales de operación. Cada episodio de entrenamiento recorre un tramo temporal de esos datos, y el agente va observando el estado del sistema, tomando decisiones (en este caso, predicciones), y recibiendo una recompensa según lo cerca o lejos que haya estado del valor real.

Una de las ventajas de Gymnasium es que permite estructurar todo esto de manera clara. Todo el proceso, desde cómo se organizan los datos hasta cómo se estructura el episodio o cómo se calcula la recompensa, se puede definir de forma clara y controlada. Esto ha sido clave para poder adaptar el entorno a las características reales del sistema y para asegurarse de que el agente aprende de forma coherente.

2.3 STABLE-BASELINES3

Stable-Baselines3 (SB3) es una librería que incluye implementaciones listas para usar de varios algoritmos de aprendizaje por refuerzo. Es una herramienta muy útil porque te permite centrarte en diseñar el entorno y analizar los resultados, sin tener que programar cada algoritmo desde cero. Además, está diseñada para funcionar perfectamente con entornos creados con Gymnasium, lo que hace que todo se integre sin complicaciones.

En este proyecto, se ha utilizado SB3 para entrenar al agente dentro del entorno creado. Una vez que el entorno estaba definido, bastaba con elegir el algoritmo y ajustar sus parámetros (como el número de pasos, la tasa de aprendizaje, el tamaño de las redes, etc.). La librería se ha encargado del entrenamiento, del almacenamiento de los modelos y de la evaluación posterior. También ha permitido hacer pruebas con distintas configuraciones de una forma bastante sencilla, lo que ha facilitado mucho el trabajo.

2.4 ALGORITMO TD3

Una vez definido el entorno y elegida la herramienta de entrenamiento, hay que seleccionar un algoritmo. En este caso, se ha utilizado TD3, que son las siglas de *Twin Delayed Deep Deterministic Policy Gradient*. Es un algoritmo pensado para trabajar en entornos donde las variables cambian de forma continua y no se pueden dividir en categorías discretas. Esto es justo lo que ocurre en este proyecto, donde las variables clave son cosas como temperaturas o potencias, que pueden tomar cualquier valor dentro de un rango.

TD3 está pensado para que el aprendizaje sea más estable. Para ello, utiliza dos redes que ayudan a tomar decisiones más seguras, introduce algo de ruido para que el agente explore diferentes opciones, y actualiza sus decisiones de forma más lenta para evitar que se confunda con cambios aleatorios en los datos.

En resumen, TD3 permite entrenar modelos precisos incluso cuando se trabaja con datos complejos y variables que cambian continuamente, como ocurre en este proyecto.

2.5 ESTRUCTURA GENERAL

Todas las herramientas que se han descrito en este capítulo se combinan para formar el sistema completo de diagnóstico. La idea general es la siguiente: a partir de los datos reales de la planta, se construye un entorno de entrenamiento que permite enseñar a un modelo cómo debería comportarse el sistema cuando todo funciona bien.

Primero, los datos históricos alimentan el entorno personalizado creado con Gymnasium. Este entorno presenta al agente secuencias reales del funcionamiento de la bomba, como si fueran episodios. El agente, entrenado con el algoritmo TD3 utilizando la librería Stable-Baselines3, va aprendiendo a predecir el valor que debería tener una determinada variable en cada momento.

Una vez el modelo ha sido entrenado, se pueden comparar sus predicciones con los valores reales medidos en la planta. Si hay una diferencia importante entre ambos, eso puede ser señal de que algo está empezando a ir mal. A partir de ese error, se calculan indicadores que permiten visualizar de forma clara cuándo el sistema se está alejando de su comportamiento habitual.

Capítulo 3. ESTADO DE LA CUESTIÓN

Antes de poner en marcha cualquier desarrollo técnico, es necesario pararse a investigar qué soluciones existen ya, tanto en el ámbito académico como en el industrial. Es una forma de entender en qué punto se encuentra la tecnología, qué métodos se han probado y qué huecos podrían quedar por explorar. Este proyecto parte precisamente de esa reflexión: ¿hay sistemas similares que sirvan para diagnosticar el estado de bombas de agua de circulación? ¿Existen trabajos anteriores que hayan aplicado inteligencia artificial, y en particular aprendizaje por refuerzo, a problemas similares?

El diagnóstico de fallos en maquinaria industrial es un campo que lleva años desarrollándose, con muchas líneas de investigación abiertas. En el caso de bombas hidráulicas, se han aplicado estrategias basadas en umbrales, en análisis estadístico y, más recientemente, en técnicas de aprendizaje automático. La mayoría de estas últimas se enmarcan dentro del aprendizaje supervisado, donde se parte de un conjunto de datos etiquetado y se entrena un modelo para predecir directamente el estado del sistema o la probabilidad de fallo.

Ese es el enfoque que sigue un estudio previo centrado en la predicción de variables críticas en bombas de agua de circulación, desarrollado a partir de datos reales de operación de una planta de ciclo combinado. En ese trabajo, se identifican varios modos de fallo asociados a diferentes variables, como la temperatura del cojinete de empuje o la presión de descarga, y se construyen modelos predictivos utilizando técnicas de aprendizaje supervisado como regresión lineal o redes neuronales. El objetivo es estimar el valor esperado de cada variable bajo condiciones normales, y a partir de ahí detectar desviaciones que podrían interpretarse como síntomas de deterioro.

Este planteamiento es muy similar al que se sigue en el presente proyecto, con la diferencia de que aquí se ha optado por aplicar aprendizaje por refuerzo. En lugar de entrenar un modelo directamente con pares entrada-salida, se construye un entorno donde el agente aprende a

predecir el valor de una variable a partir de la experiencia, guiado únicamente por una señal de recompensa que penaliza los errores.

El uso de aprendizaje por refuerzo en este tipo de aplicaciones es menos común, pero ya existen algunos trabajos que han explorado esta vía. El más cercano y directamente relacionado es el estudio titulado *Digital Twins for Predictive Maintenance using Reinforcement Learning*, presentado en la *European Conference of the Prognostics and Health Management Society 2024*. En él, se desarrollan dos modelos digitales del comportamiento de una bomba de refrigeración: uno utilizando redes neuronales profundas y otro utilizando precisamente TD3, el mismo algoritmo empleado en este proyecto.

En ese estudio, el agente se entrena con datos reales de planta y aprende a predecir el comportamiento esperado de una variable de interés. Las predicciones se comparan con los valores reales y, cuando se observa una desviación significativa, se interpreta como una posible señal de fallo. Además, se define una banda de confianza para filtrar pequeñas variaciones no relevantes, lo que permite evitar falsas alarmas. Este enfoque, tanto en la estructura como en el tratamiento del error, es muy similar al que se ha implementado aquí.

El estudio también pone de manifiesto algunas ventajas del aprendizaje por refuerzo frente a otros métodos: no requiere etiquetado manual de los datos, se adapta mejor a comportamientos no lineales y permite entrenar modelos que aprenden directamente del comportamiento del sistema. Aunque su implementación puede ser más compleja, su flexibilidad y capacidad de adaptación lo hacen especialmente interesante en entornos industriales reales, donde las condiciones pueden cambiar con el tiempo y no siempre es fácil definir reglas claras.

En conjunto, los trabajos revisados confirman que el diagnóstico automático de bombas es un problema de interés y que ya se han propuesto soluciones viables utilizando inteligencia artificial. También muestran que el uso de aprendizaje por refuerzo, aunque todavía no es mayoritario, está empezando a dar resultados prometedores. El presente proyecto se sitúa en esa línea emergente, combinando ideas de ambos enfoques: por un lado, sigue la estructura

basada en modos de fallo que ya ha demostrado su utilidad en trabajos anteriores, y por otro, apuesta por un tipo de aprendizaje más autónomo y flexible, capaz de ajustarse a las condiciones reales del sistema sin depender de etiquetas externas.

Capítulo 4. DEFINICIÓN DEL TRABAJO

4.1 JUSTIFICACIÓN

Cuando uno se plantea desarrollar una solución técnica, lo primero que debería preguntarse es si realmente aporta algo nuevo. No tiene sentido construir algo si ya hay otras herramientas que resuelven el mismo problema igual o mejor. Por eso, antes de empezar este proyecto, lo primero fue investigar qué se había hecho ya en el ámbito del diagnóstico automático de bombas industriales y hasta qué punto las soluciones existentes resolvían bien el problema.

El análisis del estado de la cuestión muestra claramente que este es un campo en el que se han hecho avances importantes, pero donde aún hay espacio para mejorar. Existen trabajos que aplican técnicas de aprendizaje automático para predecir fallos, y algunos incluso utilizan datos reales de planta. Pero la mayoría de estas propuestas se basan en enfoques supervisados, donde el modelo aprende a partir de datos etiquetados o con fallos ya identificados. Esto presenta varias limitaciones: hace falta una cantidad significativa de datos clasificados, los modelos suelen estar muy ligados a un contexto específico y resulta complicado adaptarlos a otras condiciones de operación sin reentrenarlos por completo.

Por otro lado, han empezado a surgir algunas propuestas que aplican técnicas de aprendizaje por refuerzo al mantenimiento predictivo. Estos trabajos han demostrado que es posible entrenar agentes que aprendan directamente del comportamiento del sistema sin necesidad de etiquetas, lo cual es especialmente útil cuando se trabaja con sistemas reales donde los datos de fallo no son abundantes o no están claramente definidos. Sin embargo, la mayoría de estas propuestas se centran en modelos generalistas que tratan de replicar el comportamiento global del sistema, sin una separación clara por variables ni por modos de fallo, lo que limita su aplicación en un entorno industrial real.

Este proyecto surge de esa doble observación: por un lado, que los enfoques supervisados funcionan pero no siempre son lo suficientemente flexibles; y por otro, que los modelos basados en refuerzo tienen mucho potencial, pero necesitan una estructura más concreta y orientada a casos de uso reales para ser realmente útiles.

4.1.1 ORGANIZACIÓN DEL SISTEMA

La idea de este proyecto es construir modelos que aprendan a predecir el comportamiento normal de ciertas variables clave de la bomba, y detectar cuándo empiezan a desviarse. No se trata de modelar el sistema completo, sino de centrarse en lo que realmente aporta información útil para el diagnóstico.

Para eso, se han agrupado los posibles fallos en diferentes modos, y se han elegido variables representativas para cada uno. En algunos casos, un mismo modo de fallo necesita más de un modelo, porque afecta a más de una variable. En otros, hay modelos que sirven para más de un modo, porque esa variable es relevante en varios escenarios. Esa organización, aunque sencilla, permite tener una estructura modular y flexible, que puede crecer si se quiere añadir más modelos en el futuro.

Todos los modelos se entrenan con datos reales de la planta. No se necesita ningún sensor nuevo ni cambiar la infraestructura existente. Lo único que se hace es usar los datos que ya se están registrando para enseñar al agente cómo debería comportarse una variable en condiciones normales. Si en algún momento se aleja de eso más de lo esperable, se interpreta como una señal de posible fallo.

4.1.2 POR QUÉ ESTE ENFOQUE TIENE SENTIDO

Lo que hace diferente a este proyecto no es solo que use aprendizaje por refuerzo, sino cómo se ha aplicado. En lugar de construir un modelo genérico y difícil de llevar a la práctica, se ha planteado una solución que pueda encajar con facilidad en una planta real. Todo el sistema está pensado para funcionar con los datos que ya se recogen.

Además, el modelo no da simplemente una predicción. A partir del error que comete, se construyen indicadores que ayudan a interpretar el resultado: se ve si la predicción se aleja más de lo normal, si eso ocurre de forma puntual o sostenida en el tiempo, y si el comportamiento empieza a ser preocupante. La idea no es solo predecir, sino ofrecer información que ayude a tomar decisiones de mantenimiento antes de que aparezca un fallo real.

El sistema también está montado de forma modular, así que es fácil ampliarlo si se quieren añadir más variables o modos de fallo. Y como todo el entrenamiento se hace offline, no interfiere con el funcionamiento habitual de la planta. En resumen, es una solución que se puede usar, adaptar y escalar sin grandes complicaciones.

4.2 OBJETIVOS

El objetivo principal de este proyecto es desarrollar un sistema capaz de detectar comportamientos anómalos en una bomba de agua de circulación, utilizando aprendizaje por refuerzo y datos reales de operación. La idea no es predecir directamente si hay un fallo o no, sino construir modelos que aprendan a identificar cómo debería comportarse el sistema cuando todo funciona bien, y detectar cuándo eso deja de cumplirse.

Para conseguirlo, se va a diseñar un entorno de entrenamiento personalizado basado en Gymnasium, en el que un agente pueda aprender a predecir el valor esperado de determinadas variables relevantes para el diagnóstico. Estas variables se han elegido a partir de los modos de fallo definidos previamente, y representan señales físicas clave como temperaturas, presiones o potencias.

El proyecto no se plantea como un único modelo general, sino como un conjunto de modelos independientes. Cada uno se entrena con una variable distinta, lo que permite tratar los modos de fallo de forma más específica. Además, como ya se ha explicado antes, algunos de estos modelos podrán ser útiles para más de un modo de fallo, o bien complementar a otros dentro del mismo.

Una vez entrenados los modelos, se evaluará su capacidad para detectar desviaciones significativas. Para eso, se analizará el error entre lo predicho y el valor real, y se utilizarán herramientas como bandas de confianza o curvas de error acumulado para identificar posibles situaciones de riesgo. El sistema se validará no solo por su precisión, sino también por su utilidad práctica: debe ser capaz de generar señales claras, interpretables y útiles en un contexto real de operación.

Además, se tendrá en cuenta la escalabilidad del enfoque. Parte del objetivo es comprobar si esta forma de trabajar, con modelos independientes, entrenados con datos reales y sin necesidad de etiquetado, puede ampliarse fácilmente a otros equipos, otras variables o incluso otras plantas con condiciones similares.

En definitiva, este proyecto busca construir una herramienta de diagnóstico que no solo funcione, sino que tenga sentido implementar en la práctica: que no requiera grandes cambios, que se base en datos que ya existen, y que dé resultados que realmente ayuden a tomar decisiones.

4.3 METODOLOGÍA

Para poder desarrollar el sistema propuesto, lo primero ha sido revisar y limpiar los datos. Se ha trabajado directamente con los registros históricos de la planta, eliminando outliers evidentes y tramos en los que la bomba no estaba en condiciones normales de operación. A partir de ahí, se ha definido qué variables se iban a modelar, basándose en los modos de fallo descritos y en lo que tiene sentido desde un punto de vista técnico. Se ha intentado centrar el proyecto en lo que realmente puede ayudar a anticipar un fallo.

Con esa base, se ha creado un entorno de entrenamiento personalizado utilizando la librería Gymnasium. En ese entorno, cada episodio representa un tramo temporal del funcionamiento real de la bomba, y el agente recibe observaciones, hace predicciones, y obtiene una recompensa en función de lo cerca que ha estado del valor real.

El algoritmo elegido para el entrenamiento ha sido TD3, que se adapta bien a variables continuas como las que se manejan aquí. El agente aprende únicamente a partir de los datos, sin necesidad de etiquetarlos ni de introducir reglas externas. Una vez entrenado, el modelo se valida utilizando un conjunto de test independiente, para comprobar si realmente es capaz de generalizar su comportamiento y detectar desviaciones cuando se le presentan datos que no ha visto antes.

Después del entrenamiento, se analiza el error entre la predicción del modelo y el valor real de cada variable. Ese análisis se representa en el tiempo para ver cómo evoluciona y si llega a salirse de los límites esperables. A partir de ahí, se construyen indicadores que permiten visualizar de forma clara cuándo hay un cambio en el comportamiento normal del sistema.

Todo el proceso se ha planteado de forma modular. Cada modelo se entrena y se evalúa por separado, lo que permite ver con claridad cuáles funcionan mejor y cuáles se podrían mejorar. Además, esta estructura hace que sea fácil ampliar el sistema más adelante, añadiendo nuevos modelos o adaptándolo a otras bombas sin necesidad de rehacer todo desde cero.

Capítulo 5. SISTEMA/MODELO DESARROLLADO

Este capítulo describe el desarrollo completo del sistema de diagnóstico basado en aprendizaje por refuerzo aplicado a una bomba de agua de circulación. A diferencia de lo que se verá en el siguiente capítulo, aquí no se analizan resultados ni se interpretan desviaciones, sino que se explica cómo se ha construido el sistema paso a paso, desde el tratamiento de los datos hasta la organización final de los modelos y su evaluación.

El objetivo ha sido diseñar una solución modular, basada en datos reales, que permita detectar de forma anticipada comportamientos anómalos en señales asociadas a distintos modos de fallo.

5.1 VISIÓN GENERAL DEL SISTEMA

El sistema desarrollado se basa en un conjunto de modelos que aprenden, de forma individual, cómo se comportan determinadas variables de la bomba en condiciones normales de funcionamiento. Cada modelo se centra en una sola señal, y su objetivo no es detectar directamente un fallo, sino aprender a predecir qué valor debería tomar esa variable si todo está funcionando como debería. Si el modelo deja de acertar de forma constante, eso suele indicar que la variable ya no se comporta como lo hacía en condiciones normales.

Como ya se ha mencionado antes, se ha optado por una estructura modular. En lugar de construir un único modelo general que intente abarcar todo el sistema, se ha entrenado cada modelo por separado. Además, trabajar con modelos independientes hace que el sistema sea más flexible, ya que permite ampliar o ajustar el sistema únicamente modificando una parte, sin tener que rehacer todo.

Más adelante, en este mismo capítulo, se explicará cómo se han relacionado estos modelos con los distintos modos de fallo definidos. En algunos casos, un modelo puede servir para más de un modo de fallo, si la variable que predice es relevante en distintas situaciones. En

otros, un mismo modo de fallo requiere varios modelos para estar bien cubierto, porque afecta a más de una señal.

El entrenamiento de cada modelo se ha hecho en un entorno personalizado construido con Gymnasium. Todos los datos utilizados en la fase de entrenamiento han sido filtrados previamente para asegurar que corresponden a situaciones de funcionamiento normal, sin desviaciones. No se usan etiquetas, ni reglas externas, el agente simplemente observa, predice, y ajusta su comportamiento en función del error que comete.

Una vez entrenado, el modelo se valida con datos nuevos que no ha visto antes. Se calcula el error entre lo predicho y lo real, y se analiza si ese error se mantiene dentro de los márgenes normales o si empieza a crecer. Aunque ese análisis se verá con más detalle en el capítulo siguiente, aquí ya se deja planteado cómo se estructura la detección de posibles desviaciones.

5.2 PREPARACIÓN DE LOS DATOS

Todo el sistema parte de los datos reales registrados por la planta durante su funcionamiento normal. Estos datos recogen la evolución temporal de distintas variables, como temperaturas, presiones, potencias o caudales, medidas en diferentes puntos del sistema. El conjunto es amplio y contiene información útil, pero también incluye tramos que no son representativos o que podrían afectar negativamente al entrenamiento, por lo que el primer paso ha sido preparar los datos con cuidado.

En primer lugar, se han eliminado los periodos en los que la planta no estaba funcionando de manera estable. Esto incluye arranques, paradas o fases de operación con potencias muy bajas. Para asegurarlo, se ha filtrado el conjunto de datos manteniendo solo las muestras en las que tanto las potencias de la turbina de gas y de la turbina de vapor estaban por encima de un cierto umbral. Así se garantiza que los datos utilizados reflejen situaciones normales y repetitivas, que son justo las que interesan para que el modelo aprenda bien el patrón habitual de comportamiento.

Una vez aplicado ese primer filtro, los datos se han ordenado por fecha y divididos de forma temporal. Se ha utilizado el año 2020 como conjunto de entrenamiento, mientras que los datos de los años siguientes (2021 a 2023) se han reservado para la fase de validación. Esta separación garantiza que el modelo no haya visto previamente los datos con los que se va a evaluar su capacidad para detectar desviaciones. Además, permite comprobar cómo se comporta cuando se enfrenta a situaciones más recientes, con condiciones que pueden haber cambiado ligeramente respecto al periodo de entrenamiento.

Antes de entrenar, se ha aplicado un filtrado adicional para eliminar valores extremos o outliers que pudieran distorsionar el aprendizaje. Este paso se ha hecho únicamente sobre los datos de entrenamiento, utilizando el rango intercuartílico (IQR) de cada variable. Se eliminaron las muestras que quedaban demasiado alejadas de la distribución central. El objetivo de este filtrado es que el modelo se entrene solo con comportamientos realmente representativos, evitando que aprenda sobre datos atípicos o poco frecuentes. El conjunto de test, en cambio, se ha mantenido sin modificar, precisamente para evaluar si el modelo es capaz de detectar de forma autónoma esos mismos valores como desviaciones.

Una vez seleccionados y limpiados los datos, se ha aplicado un escalado de todas las variables. Esto permite normalizar los rangos de las señales entre 0 y 1, lo que facilita el entrenamiento del modelo y evita que ciertas variables dominen sobre otras por tener valores numéricamente mayores.

Este proceso de preparación es una parte esencial del sistema. Si los datos no están bien filtrados, organizados y escalados, el agente puede aprender patrones que no representan el comportamiento real de la planta.

5.3 DISEÑO DEL ENTORNO DE APRENDIZAJE

En lugar de usar una simulación genérica o un entorno prediseñado, se ha optado por crear un entorno específico para cada modelo, utilizando directamente los datos reales de

funcionamiento de la planta. Esto permite que el agente aprenda en un entorno que refleja exactamente las condiciones que se quieren modelar.

El funcionamiento del entorno es bastante directo. En cada paso, el agente recibe como entrada una observación que incluye las variables necesarias para predecir la señal objetivo. A partir de esa observación, el agente genera una acción, que en este caso es una predicción del siguiente valor de la variable que se está modelando.

Una vez realizada la predicción, el entorno la compara con el valor real de la señal en ese instante. En función del error cometido, se calcula una recompensa: cuanto más se acerque la predicción al valor correcto, mayor es la recompensa. Esta forma de plantear el problema permite que el agente aprenda a mejorar su precisión con el tiempo, sin necesidad de usar etiquetas ni marcar explícitamente qué es un fallo y qué no. Solo se basa en su capacidad para acertar.

Cada episodio de entrenamiento recorre un tramo temporal completo de los datos disponibles para ese modelo. Al finalizar el episodio, el entorno se reinicia y vuelve al principio, de modo que el agente puede realizar múltiples pasadas por el mismo conjunto de datos y seguir ajustando su comportamiento. Esto es especialmente útil cuando se trabaja con datos reales, ya que permite aprovecharlos al máximo sin necesidad de generar datos sintéticos.

Además, el entorno se ha planteado de forma flexible para que sea fácil adaptarlo a cada modelo. Cambiar la variable objetivo o las señales de entrada no requiere modificar toda la estructura, lo que ha permitido entrenar modelos diferentes sin complicaciones. Esto ha hecho que el sistema sea más manejable y que se pueda extender con facilidad si se quisiera aplicar el mismo enfoque a otras variables o a otras bombas.

5.4 CONFIGURACIÓN DEL ALGORITMO TD3

Todos los modelos del sistema se han entrenado utilizando el algoritmo TD3 (Twin Delayed Deep Deterministic Policy Gradient). Se trata de un método basado en aprendizaje por

refuerzo que resulta especialmente útil cuando se trabaja con espacios de acción continuos, como es el caso de este proyecto, donde el agente debe predecir un valor numérico en cada paso. Frente a otras alternativas más simples, TD3 ofrece mayor estabilidad durante el entrenamiento y mejores resultados cuando el objetivo es aproximar una función de comportamiento real a partir de datos.

En todos los modelos se ha utilizado la misma estructura de red neuronal. Esta red tiene dos capas ocultas con 256 neuronas cada una, y se ha activado con funciones ReLU. Esta configuración es bastante habitual y funciona bien en la mayoría de los casos. No se ha buscado complicar más el diseño porque lo importante no es tener un modelo muy grande, sino uno que sea estable, aprenda rápido y se entienda bien. Usar la misma estructura para todos también ayuda a que el sistema sea más fácil de mantener y de comparar.

También se ha añadido un poco de ruido a las predicciones durante el entrenamiento. Esto puede parecer contraproducente, pero es útil porque obliga al modelo a explorar distintas formas de actuar antes de asentarse en una estrategia. El ruido que se ha usado es muy pequeño, simplemente para que el agente no se quede desde el principio en una solución cómoda que luego no mejora.

Este planteamiento no busca complicar el entrenamiento más de lo necesario, sino que cada modelo aprenda bien el comportamiento normal de la variable que le corresponde. A partir de ahí, será posible detectar si en algún momento ese comportamiento cambia, que es justo lo que se busca con este sistema.

5.5 RELACIÓN ENTRE MODELOS Y MODOS DE FALLO

El sistema está formado por varios modelos independientes, y cada uno de ellos se ha entrenado para predecir el valor de una variable concreta del sistema. Estas variables están directamente relacionadas con el comportamiento de la bomba y permiten tener una referencia clara de cómo deberían evolucionar en condiciones normales. Si más adelante esa

variable empieza a desviarse, el error del modelo también crecerá, y eso puede servir como aviso de que algo está empezando a cambiar.

En algunos casos, una sola variable resulta suficiente para detectar si un modo de fallo está empezando a aparecer. Por ejemplo, si un fallo afecta directamente a la presión de descarga o a la temperatura de un cojinete, basta con que el modelo encargado de esa variable detecte una desviación para tener una señal clara. En otros casos, el fallo no se refleja en una sola señal, sino que afecta al comportamiento de varias variables al mismo tiempo. En esos casos, el diagnóstico se apoya en varios modelos distintos, y se interpreta el conjunto de forma más completa.

También se da la situación contraria: hay variables que tienen relación con más de un modo de fallo. En esos casos, el modelo que predice esa variable no solo contribuye al diagnóstico de un fallo concreto, sino que sirve como referencia en varios puntos del sistema. Esto hace que el sistema sea más eficiente, ya que un mismo modelo puede utilizarse en contextos distintos sin necesidad de repetir cálculos ni duplicar estructura.

Esta organización permite adaptar fácilmente el sistema. Si en el futuro se detectan nuevos modos de fallo o se quiere aplicar esta misma lógica a otra bomba o a una planta diferente, bastaría con entrenar nuevos modelos o redefinir las asociaciones ya existentes, sin necesidad de rehacer todo desde cero.

A continuación, en la Tabla 1, se muestra una tabla que resume la relación entre los distintos modos de fallo y los modelos. En ella se indica qué modelos están asociados a cada modo de fallo y qué variable predice cada modelo. Esta estructura ayuda a entender cómo se organiza el sistema y qué papel juega cada componente dentro del diagnóstico.

Es importante tener en cuenta que cada modelo representa el comportamiento específico de una bomba concreta. No se trata de modelos genéricos que puedan aplicarse indistintamente a cualquier equipo similar. Si se quisiera aplicar este mismo enfoque a otra bomba, sería necesario volver a entrenar los modelos utilizando sus propios datos de funcionamiento.

Modo de fallo	Modelo	Variables predichas
MF2 - Cojinete desgastado del lado opuesto al acoplamiento del motor	M22	Temperatura del cojinete de empuje
MF3 - Cojinete desgastado del lado del acoplamiento del motor	M21	Temperatura del cojinete de empuje del motor
MF4 - Cojinete desgastado del lado opuesto al acoplamiento de la bomba	M16	Desplazamiento del cojinete del lado opuesto al acoplamiento - dirección 1
	M17	Desplazamiento del cojinete del lado opuesto al acoplamiento - dirección 2
	M20	Temperatura del cojinete del lado acoplamiento del motor
MF5 - Cojinete desgastado del lado del acoplamiento de la bomba	M14	Desplazamiento del cojinete del lado acoplamiento - dirección 1
	M15	Desplazamiento del cojinete del lado acoplamiento - dirección 2
	M19	Temperatura del cojinete del lado acoplamiento del motor
MF6 - Cojinete de empuje desgastado	M18	Desplazamiento del cojinete de empuje axial
	M23	Temperatura del cojinete de empuje

MF8 - Desalineación del eje	M14	Desplazamiento del cojinete del lado acoplamiento - dirección 1
	M15	Desplazamiento del cojinete del lado acoplamiento - dirección 2
	M16	Desplazamiento del cojinete del lado opuesto al acoplamiento - dirección 1
	M17	Desplazamiento del cojinete del lado opuesto al acoplamiento - dirección 2
	M18	Desplazamiento del cojinete de empuje axial
	M19	Temperatura del cojinete del lado acoplamiento del motor
	M20	Temperatura del cojinete del lado acoplamiento del motor
	M21	Temperatura del cojinete de empuje del motor
	M22	Temperatura del cojinete de empuje
	M23	Temperatura del cojinete de empuje

Tabla 1. Relación entre modos de fallo, modelos entrenados y variable predicha

5.6 RELACIÓN ENTRE MODELOS Y MODOS DE FALLO

Una vez entrenados los modelos, la siguiente fase del proyecto ha consistido en comprobar si realmente son capaces de detectar desviaciones cuando las variables que predicen empiezan a comportarse de forma distinta a lo habitual. Para eso, se ha validado cada modelo con datos reales de años posteriores al entrenamiento, sin modificar ni filtrar esos datos más allá del escalado que ya se aplicó.

La evaluación se ha centrado en el error de predicción, es decir, en la diferencia entre lo que dice el modelo y el valor real de la señal. Cuando el sistema funciona con normalidad, ese error tiende a ser bajo y relativamente estable. Sin embargo, cuando la señal se empieza a alejar del comportamiento esperado, el modelo se equivoca más, y ese error puede aumentar de forma significativa.

Para que la interpretación de ese error sea coherente en todos los modelos, se ha aplicado una normalización basada en la desviación estándar del error de entrenamiento. Así, el error de test se expresa en función de cuántas veces se aleja del margen que el modelo considera normal. A partir de ahí, se han definido unas bandas de confianza situadas en $\pm 3\sigma$ (tres veces la desviación estándar). Si el error se mantiene dentro de ese rango, se entiende que el comportamiento es habitual. Si se sale de forma repetida, se considera que hay una desviación clara.

Además de ver si el error supera o no ese umbral, también se ha definido un indicador acumulativo de estrés. La idea es la siguiente: si el modelo se equivoca mucho, durante muchos pasos seguidos, eso puede indicar que el fallo no es puntual, sino que la variable lleva tiempo alejándose del comportamiento que el modelo considera normal. Este tipo de indicador ayuda a distinguir entre errores aislados y desviaciones más sostenidas en el tiempo.

Todos estos indicadores se han calculado primero a nivel de modelo, de forma independiente, pero también se han agrupado para obtener una visión más amplia del estado

de cada modo de fallo. En los casos en los que un mismo fallo afecta a varias señales, el estrés acumulado se ha combinado para representar mejor la situación general. De esta forma, no solo se evalúa si un modelo concreto empieza a fallar, sino también si, en conjunto, las señales asociadas a un mismo modo de fallo están mostrando un cambio sostenido en el tiempo.

Capítulo 6. ANÁLISIS DE RESULTADOS

Una vez entrenados los modelos, el siguiente paso lógico es ver cómo se comportan al aplicarlos sobre datos reales. Este capítulo recoge los resultados obtenidos durante la validación del sistema y analiza si las señales que emite el modelo tienen sentido y si responden bien ante los cambios que se producen en cada modo de fallo.

Más allá de si las predicciones son exactas o no en cada momento, lo importante es saber si el sistema detecta de forma coherente cuándo una variable empieza a alejarse de su comportamiento habitual. Por eso, en lugar de fijarse únicamente en el error puntual, el análisis se centra en los errores normalizados y en la evolución del estrés acumulado, como se explicó en el capítulo anterior.

Los resultados se han organizado por modos de fallo. En cada uno se explican los modelos que intervienen, cómo evolucionan sus errores y en qué momento se detectan desviaciones significativas.

6.1 ANÁLISIS POR MODOS DE FALLO

Para poder evaluar el comportamiento del sistema de forma más clara, se ha analizado por separado cada uno de los modos de fallo incluidos en el proyecto. Aunque todos siguen la misma lógica de funcionamiento, hay diferencias importantes entre ellos. En algunos casos hay un solo modelo implicado, mientras que en otros intervienen varios. También hay señales que responden de forma muy clara cuando algo empieza a ir mal, y otras en las que los cambios son más sutiles. Además, varía el momento en el que se produce la desviación y lo fácil que resulta detectarla.

Por eso, en lugar de presentar todos los resultados juntos, se ha preferido tratar cada modo por separado. Esto permite entender mejor qué ha ocurrido en cada caso, interpretar los

errores de forma más precisa y ver con más claridad cuándo empiezan a aparecer señales de alarma.

En cada uno de los siguientes apartados se comentan los modelos que intervienen, los resultados que se han obtenido al aplicar el sistema sobre datos sin filtrar, y la evolución del estrés acumulado a lo largo del tiempo. Además, se incluyen las gráficas más relevantes para ilustrar los puntos clave del análisis.

6.1.1 DESGASTE DEL COJINETE DEL LADO OPUESTO AL ACOPLAMIENTO DEL MOTOR

Este modo de fallo está relacionado con un deterioro progresivo del cojinete situado en el lado opuesto al acoplamiento del motor. Para su diagnóstico se ha utilizado el modelo M22, que predice la temperatura del cojinete de empuje del motor en la bomba 1. El modelo se entrena utilizando datos históricos del año 2020, una vez eliminados los valores extremos que podrían distorsionar el aprendizaje. La validación, en cambio, se lleva a cabo sobre datos reales y sin filtrar de los tres años siguientes, lo que permite evaluar cómo responde el sistema cuando las condiciones comienzan a desviarse de lo habitual.

Durante el entrenamiento, el modelo muestra un comportamiento razonablemente estable. Como se observa en la Figura 4, los errores más pronunciados se concentran en el primer tramo de los datos. Aun así, el modelo consigue adaptarse rápidamente y mantener sus predicciones dentro de las bandas de $\pm 3\sigma$. Esto se traduce en un nivel de estrés acumulado que se estabiliza pronto y no vuelve a incrementarse de forma significativa, lo que indica que el modelo ha captado correctamente la dinámica general de la señal bajo condiciones normales.

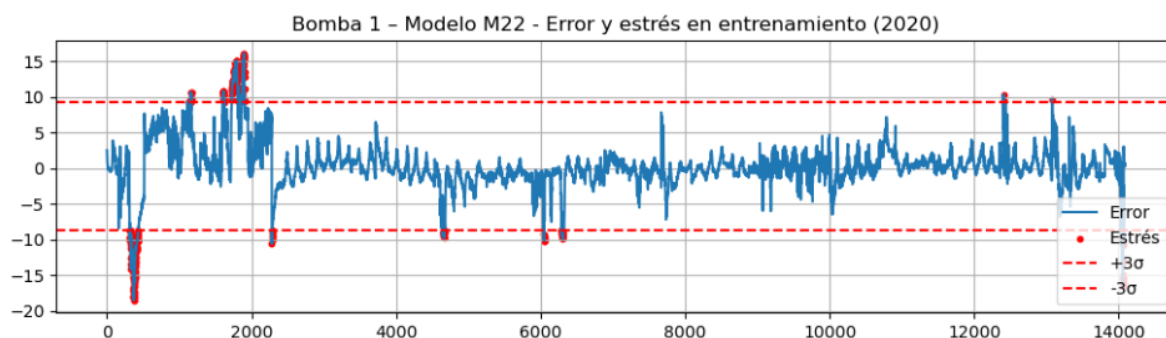


Figura 4. Error de predicción y bandas de confianza del modelo M22 durante el entrenamiento

Los resultados de la fase de validación muestran un comportamiento más variable entre años. En 2021, la señal se mantiene estable durante los primeros meses, pero a partir de mayo se produce una desviación muy marcada. El error predicho supera con claridad el umbral superior, activando de forma inmediata el indicador de estrés acumulado. Como se ve en la Figura 5, el crecimiento es muy brusco, pero se estabiliza poco después y no vuelve a incrementarse de forma significativa en lo que queda de año. Esto sugiere que el modelo detecta una anomalía clara en ese momento concreto, pero no lo interpreta como una desviación sostenida en el tiempo.

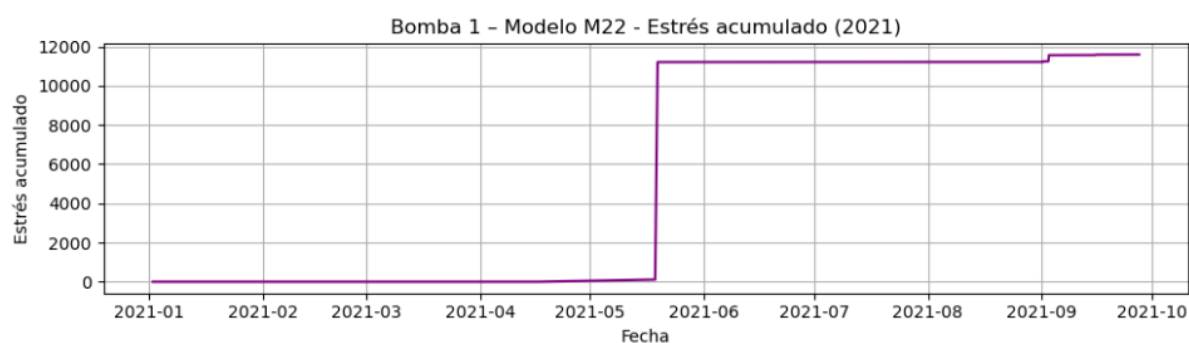


Figura 5. Estrés acumulado del modelo M22 durante el año 2021

En 2022 el comportamiento de la variable cambia. En lugar de mostrar una desviación brusca, se detecta una degradación progresiva. El error se va acumulando de forma más moderada, pero continua, y el estrés crece a lo largo del año. En la Figura 6 se aprecia que la pendiente del estrés acumulado empieza a ganar intensidad a partir del mes de junio, con un crecimiento especialmente fuerte entre julio y octubre. Esta evolución apunta a un fallo

que avanza poco a poco, sin grandes sobresaltos, pero de manera sostenida, y el modelo lo refleja con bastante precisión.

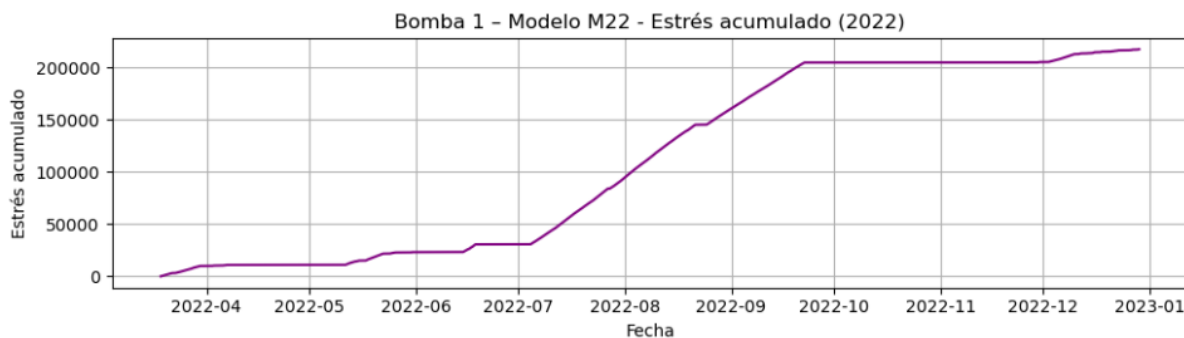


Figura 6. Estrés acumulado del modelo M22 durante el año 2022

En 2023, el patrón cambia de nuevo. El error es más irregular, con muchas oscilaciones alrededor del valor real. En la Figura 7 se muestra cómo el modelo reacciona ante estos desajustes puntuales: hay varios episodios en los que se supera el umbral superior, pero sin una tendencia clara. A pesar de estas salidas, el sistema no interpreta el comportamiento como una desviación sostenida, lo que se traduce en un estrés acumulado menos pronunciado. Es posible que durante este periodo la señal se mantuviera cerca de los límites normales, aunque con cierta inestabilidad.

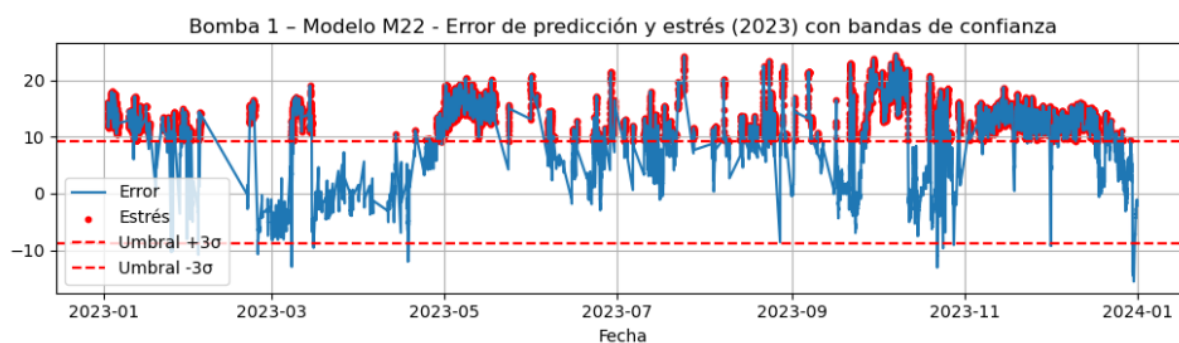


Figura 7. Error normalizado y bandas de confianza durante el año 2023

Por último, la Figura 8 muestra el estrés acumulado total del modo de fallo a lo largo del periodo completo de validación. Esta gráfica permite ver la evolución general del sistema desde una perspectiva más amplia. Durante la primavera de 2022 comienza una subida clara, que se intensifica en verano y alcanza un pico en octubre. A partir de ese momento, el sistema

entra en una fase de relativa estabilidad que se mantiene hasta agosto de 2023. Después, el estrés vuelve a crecer, aunque con una pendiente mucho más suave. Esta gráfica es muy útil porque ofrece una visión acumulada y continua que ayuda a interpretar el estado general de la bomba con una sola curva.



Figura 8. Estrés acumulado total del modo de fallo 2 en el periodo 2021–2023

6.1.2 COJINETE DESGASTADO DEL LADO DEL ACOPLAMIENTO DEL MOTOR

Este modo de fallo está asociado a un deterioro en el cojinete del lado del acoplamiento del motor. Para su diagnóstico se ha utilizado el modelo M21, que predice la temperatura del cojinete de empuje en la bomba 1. El modelo se entrena con datos de 2020, después de aplicar un filtrado para eliminar los valores más extremos, y se valida sobre datos reales y sin filtrar de los tres años siguientes.

Durante el entrenamiento, el comportamiento del modelo es bastante estable. La mayoría del error se mantiene dentro de las bandas de confianza, aunque hay algunos picos aislados. En la Figura 9 se muestra el error junto con las bandas de $\pm 3\sigma$. Se puede ver que, salvo en momentos muy concretos, el modelo aprende bien la dinámica general de la señal. Esto se refleja también en la gráfica del estrés acumulado (no incluida), donde apenas se observan episodios de estrés a lo largo del entrenamiento.

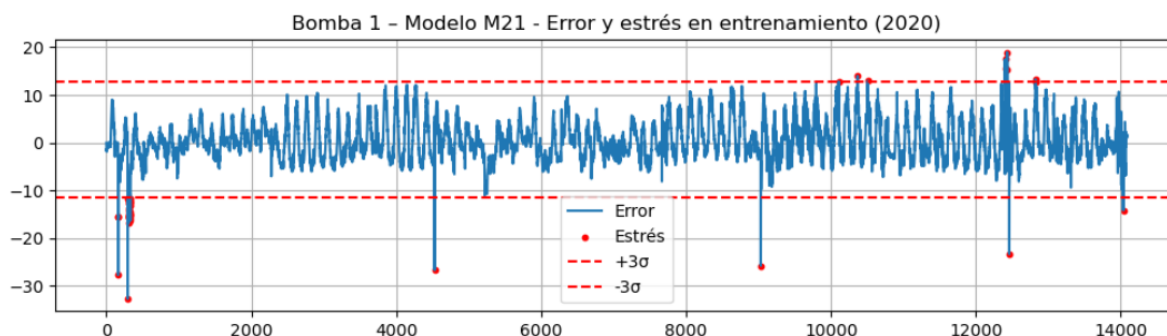


Figura 9. Error de predicción y bandas de confianza del modelo M21 durante el entrenamiento

En 2021, la señal se mantiene estable durante la mayor parte del año, pero en septiembre aparece una desviación puntual bastante clara. El error crece de forma brusca durante unos días y eso se refleja en un aumento repentino del estrés acumulado. Como se ve en la Figura 10, ese pico es muy marcado, pero se estabiliza enseguida y no vuelve a subir. Esto muestra que el sistema responde correctamente: detecta el cambio cuando ocurre, pero no sigue sumando estrés si la señal vuelve a su comportamiento normal.

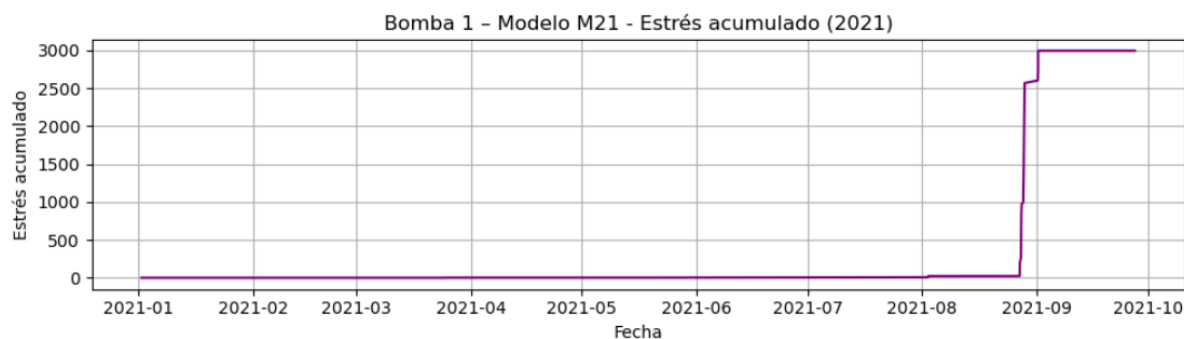


Figura 10. Estrés acumulado del modelo M21 durante el año 2021

En 2022, el error se comporta de forma más irregular. Aunque no hay picos tan pronunciados como en 2021, sí que se observan varias salidas de las bandas de confianza repartidas entre junio y septiembre. La Figura 11 muestra cómo el estrés acumulado crece en tramos: entre estos meses se van acumulando pequeñas activaciones que, aunque moderadas, indican que el sistema percibe una pérdida de estabilidad en la señal. Este tipo de comportamiento puede ser representativo de un fallo que avanza poco a poco.

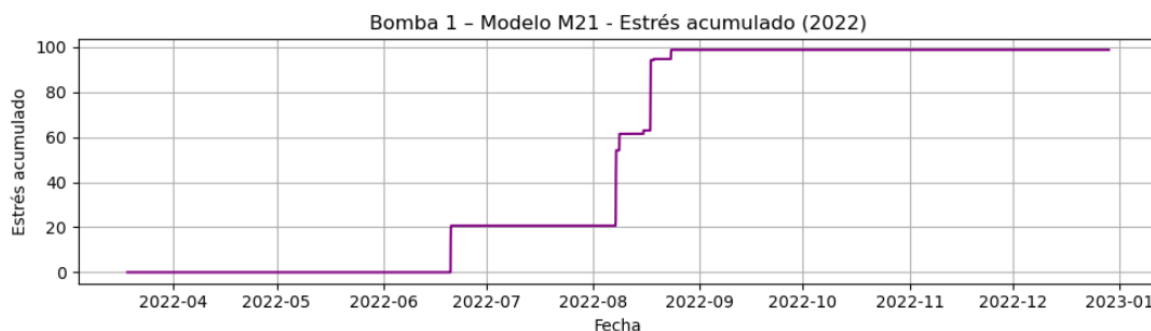


Figura 11. Estrés acumulado del modelo M21 durante el año 2022

En 2023, el modelo responde de forma más suave. El error es más irregular, con muchas oscilaciones, y aparecen varios tramos con activaciones del estrés. Como se observa en la Figura 12, hay una tendencia creciente desde mayo hasta finales de año. Aunque la pendiente no es tan pronunciada como en otros casos, la acumulación es constante, lo que sugiere un deterioro más difuso, sin un punto de inflexión claro, pero con señales de inestabilidad mantenida.

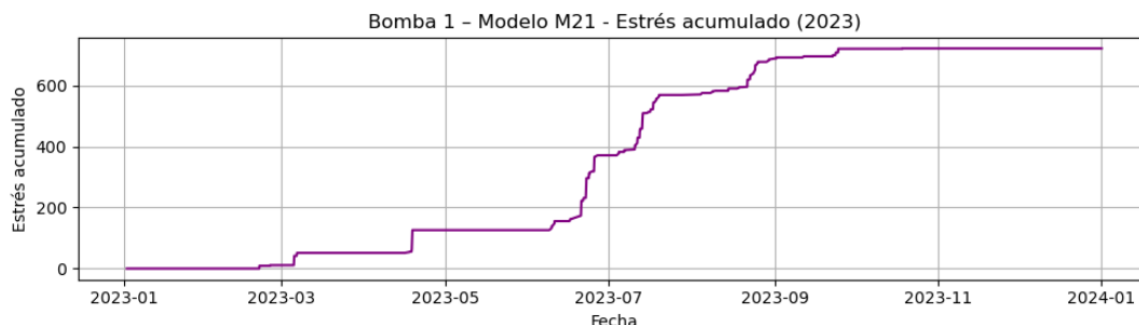


Figura 12. Estrés acumulado del modelo M21 durante el año 2023

Finalmente, la Figura 13 muestra el estrés acumulado total del modo de fallo a lo largo del periodo completo. Lo más destacado es el primer salto en septiembre de 2021, seguido de una fase más estable durante buena parte de 2022, y un crecimiento sostenido en 2023, especialmente a partir de septiembre. Esta gráfica permite ver cómo el sistema percibe el avance del fallo en el tiempo, con fases más y menos activas, y muestra que el modelo no solo responde a eventos puntuales, sino también a desviaciones más prolongadas.

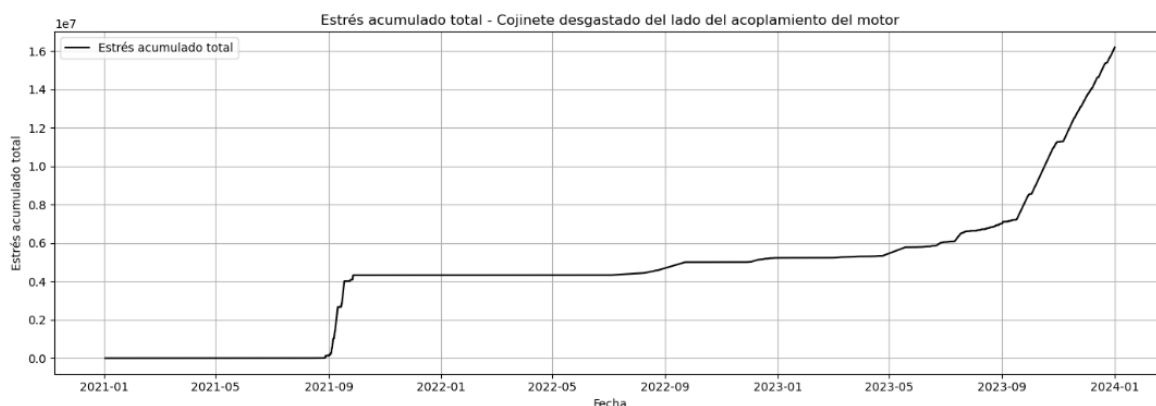


Figura 13. Estrés acumulado total del modo de fallo 3 en el periodo 2021–2023

6.1.3 COJINETE DESGASTADO DEL LADO OPUESTO AL ACOPLAMIENTO DE LA BOMBA

Este modo de fallo está relacionado con un desgaste del cojinete situado en el lado opuesto al acoplamiento de la bomba. Para su diagnóstico se han utilizado tres modelos diferentes: M16, M17 y M20. Todos ellos predicen la temperatura de componentes directamente relacionados con el cojinete, aunque difieren ligeramente en las variables de entrada. Los modelos M16 y M17 utilizan las mismas señales, mientras que el modelo M20 añade una variable adicional asociada a la presión de descarga.

El entrenamiento de los tres modelos se ha realizado utilizando los datos del año 2020, filtrando previamente los valores más extremos. En general, los tres modelos muestran un comportamiento estable durante esta fase, con errores moderados y pocas salidas de las bandas de confianza. En la Figura 14 se representa el caso del modelo M16, donde se observa que la mayor parte del error se mantiene dentro del margen de $\pm 3\sigma$. Aun así, aparecen

algunas desviaciones aisladas, sobre todo hacia el final del conjunto, que provocan una ligera acumulación de estrés. Los modelos M17 y M20 presentan un patrón bastante similar.

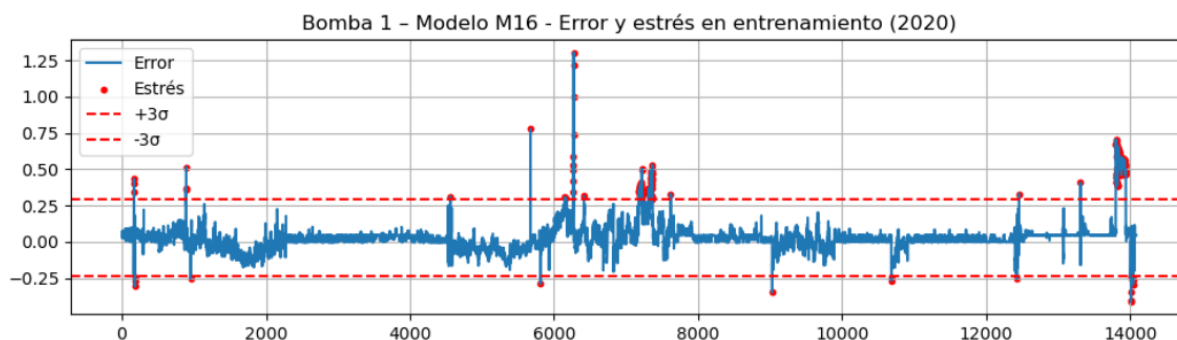


Figura 14. Error de predicción y bandas de confianza del modelo M16 durante el entrenamiento

Durante la validación en 2021, las diferencias entre modelos comienzan a hacerse más evidentes. El modelo M16 es el primero en reaccionar, acumulando estrés de forma más clara entre agosto y septiembre. La Figura 15 muestra cómo esta subida ocurre en una zona concreta del año, con un crecimiento rápido pero que no se prolonga demasiado. El modelo M17 también registra desviaciones en ese mismo periodo, aunque con menor intensidad. El modelo M20, por su parte, se mantiene estable durante la mayor parte del año, pero empieza a acumular estrés de manera significativa en septiembre, coincidiendo con los otros dos modelos. Aunque reacciona algo más tarde, acaba siendo el que más estrés acumula en ese tramo. Esta diferencia en el comportamiento puede deberse a cómo cada modelo interpreta los cambios más bruscos de nivel o la tolerancia al ruido de cada modelo. En cualquier caso, los tres modelos coinciden en señalar ese periodo como anómalo, lo que refuerza la fiabilidad del sistema.

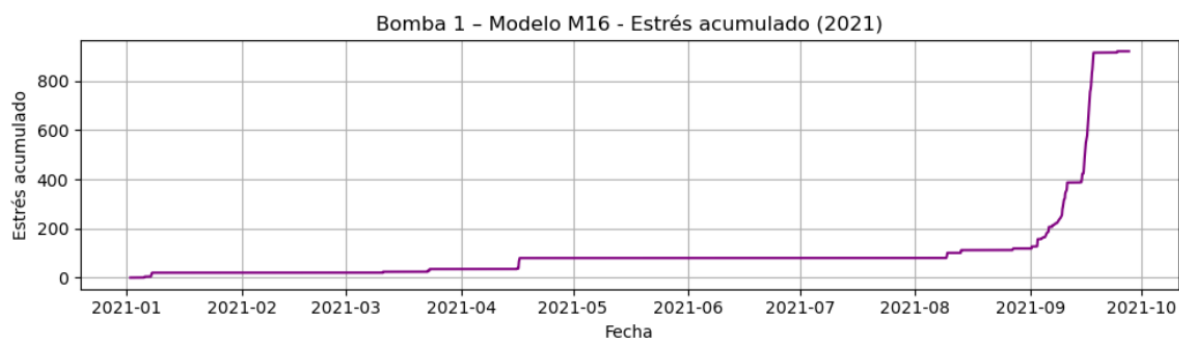


Figura 15. Estrés acumulado del modelo M16 durante el año 2021

En 2022, la situación cambia. Es el modelo M17 el que acumula más estrés a lo largo del año. Como se ve en la Figura 16, la pendiente del estrés comienza a subir a partir de junio y lo hace de forma relativamente sostenida hasta octubre. No se trata de un fallo repentino, sino de una serie de pequeñas desviaciones que se van acumulando con el tiempo. El sistema interpreta que, aunque la señal no se sale bruscamente de lo esperado, sí se mantiene en un rango poco habitual durante muchos meses. Los modelos M16 y M20 también detectan esta inestabilidad, aunque en menor medida: M16 con una pendiente más moderada y M20 con algunas activaciones puntuales.

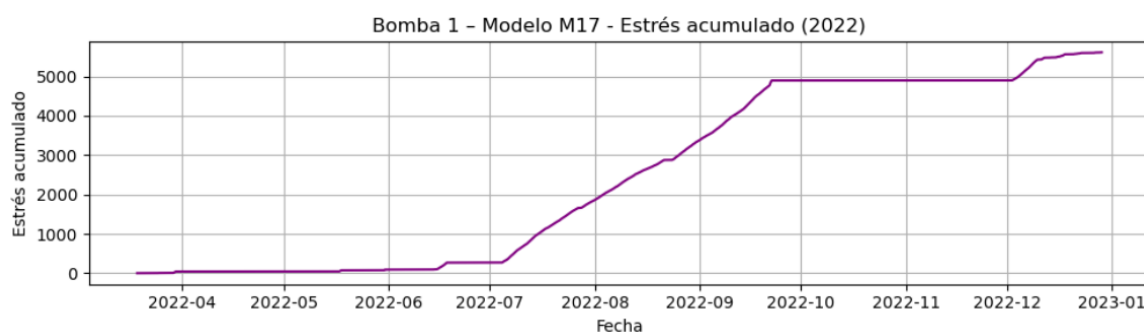


Figura 16. Estrés acumulado del modelo M17 durante el año 2022

Durante 2023, el modelo que más estrés acumula es claramente M20. Como se muestra en la Figura 17, el estrés empieza a aumentar en marzo y continúa creciendo de forma bastante constante hasta final de año. En este caso, el modelo detecta una desviación prolongada, sin grandes picos, pero con una tendencia mantenida. Esta sensibilidad puede deberse a la

variable adicional que utiliza, que podría estar captando mejor ciertos cambios lentos en las condiciones de operación. Por su parte, M16 y M17 siguen activándose en algunos tramos, pero sus curvas son menos pronunciadas y muestran más estabilidad general.

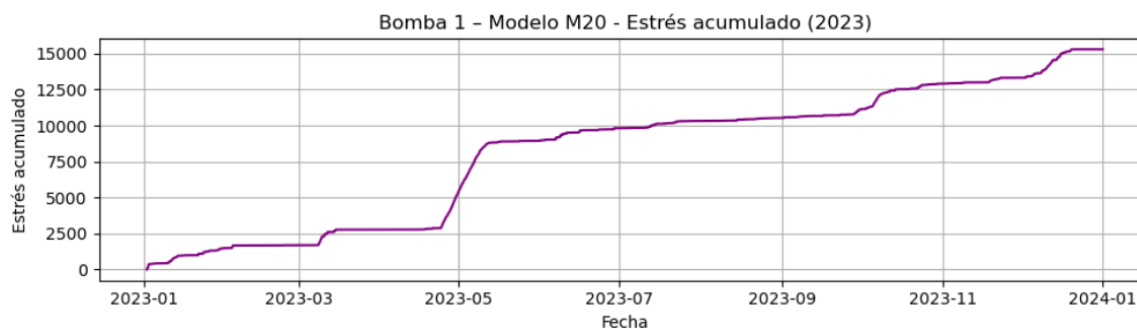


Figura 17. Estrés acumulado del modelo M20 durante el año 2023

La Figura 18 representa el estrés acumulado individual de los tres modelos utilizados para este modo de fallo: M16, M17 y M20. Aunque todos analizan señales relacionadas con el mismo componente, no reaccionan exactamente igual. En algunos tramos un modelo responde antes o de forma más marcada, mientras que otros lo hacen con más suavidad o algo más tarde. Aun así, el patrón general es muy parecido en los tres, lo que sugiere que todos están captando los mismos cambios, pero cada uno con una sensibilidad distinta.

Esta pequeña variación en la forma de responder añade solidez al sistema. Ver que los tres modelos detectan las mismas desviaciones, aunque con ligeras diferencias, refuerza la fiabilidad del diagnóstico. Esto demuestra que los modelos aportan información complementaria en la forma de cuantificar el fallo, y que el sistema gana robustez al integrar varias perspectivas sobre un mismo comportamiento.

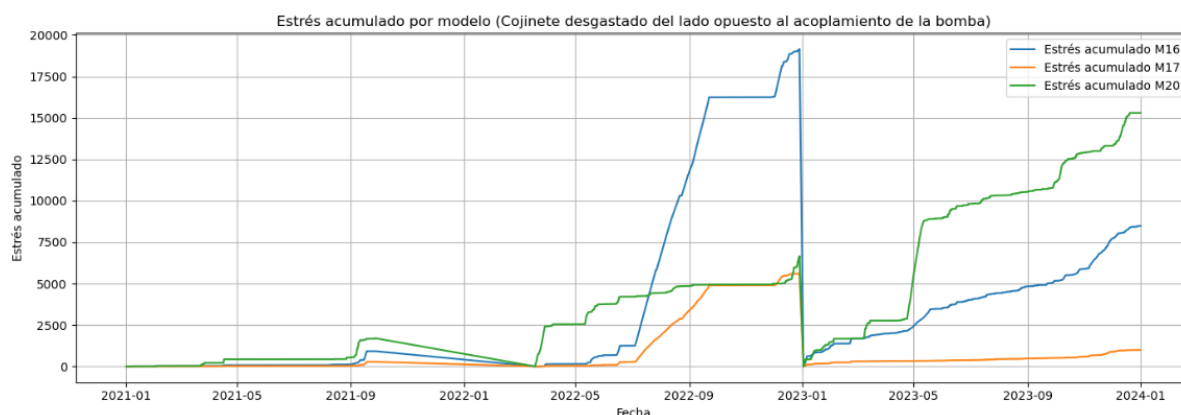


Figura 18. Estrés acumulado total por modelo del modo de fallo 4 en el periodo 2021–2023

6.1.4 COJINETE DESGASTADO DEL LADO DEL ACOPLAMIENTO DE LA BOMBA

Este modo de fallo se ha abordado utilizando tres modelos distintos: M14, M15 y M19. Cada uno de ellos predice una señal diferente, pero todas están relacionadas con el estado del cojinete situado en el lado del acoplamiento. El modelo M14 estima el desplazamiento en dirección 1, M15 en dirección 2 y M19 la temperatura del cojinete LAM. Aunque las señales son distintas, los tres modelos trabajan con datos de la misma bomba, y siguen el mismo planteamiento de entrenamiento y validación aplicado en los modos anteriores.

Durante el entrenamiento con los datos de 2020, los tres modelos muestran un comportamiento bastante estable. En la Figura 19, correspondiente al modelo M14, se observa que la mayoría del error se mantiene dentro de las bandas de $\pm 3\sigma$, aunque hay algunas salidas al final del conjunto que generan algo de estrés. En el caso de M15 y M19 ocurre algo similar, los errores están bastante contenidos, aunque hay algunos momentos concretos en los que se producen pequeñas acumulaciones de estrés. En conjunto, los modelos aprenden correctamente el comportamiento esperado de cada señal en condiciones normales de funcionamiento.

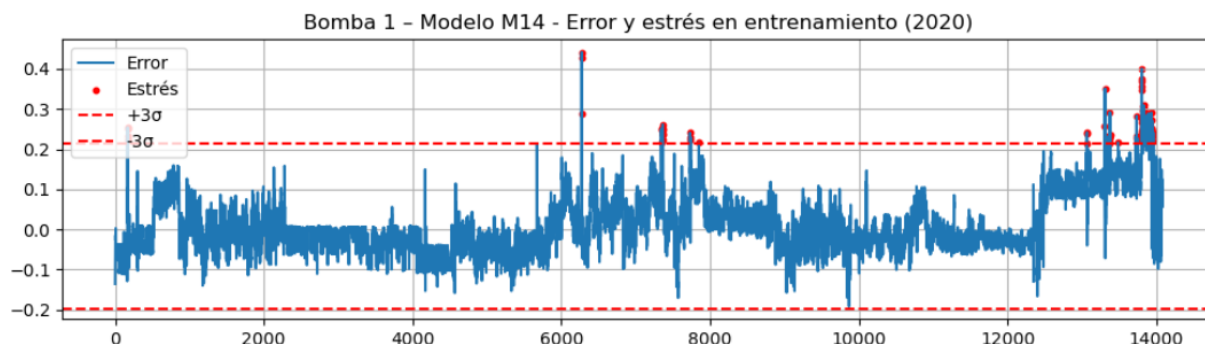


Figura 19. Error de predicción y bandas de confianza del modelo M14 durante el entrenamiento

En 2021, los tres modelos empiezan a detectar desviaciones en un tramo bastante parecido del año. En todos los casos, el estrés acumulado crece especialmente a partir de agosto, como puede verse en la Figura 20. Aunque los niveles son distintos, los tres modelos reflejan que las señales empiezan a separarse de lo que el modelo considera normal. En este caso no hay discrepancias importantes entre modelos: todos coinciden tanto en el momento en que surge la desviación como en su duración.

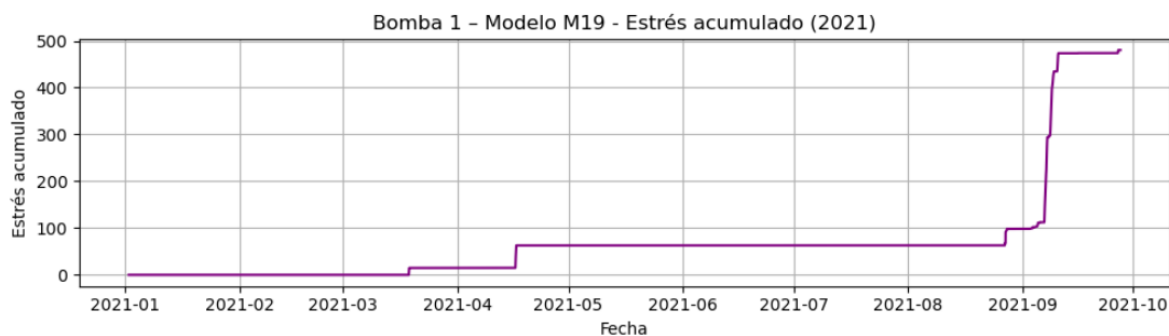


Figura 20. Estrés acumulado del modelo M19 durante el año 2021

Durante 2022, el comportamiento es más variado. En este caso los tres modelos acumulan estrés en distintos momentos del año. La Figura 21, que corresponde al modelo M15, muestra una subida especialmente fuerte entre junio y octubre. M14 también empieza a acumular estrés en ese periodo, aunque con una pendiente más suave. En cambio, el modelo M19 reacciona antes: su curva empieza a subir en primavera y se mantiene prácticamente plana durante casi todo el segundo semestre. Esta diferencia en el ritmo de crecimiento del estrés

no es necesariamente una contradicción, sino que puede reflejar que cada modelo está captando de forma distinta la misma tendencia general.

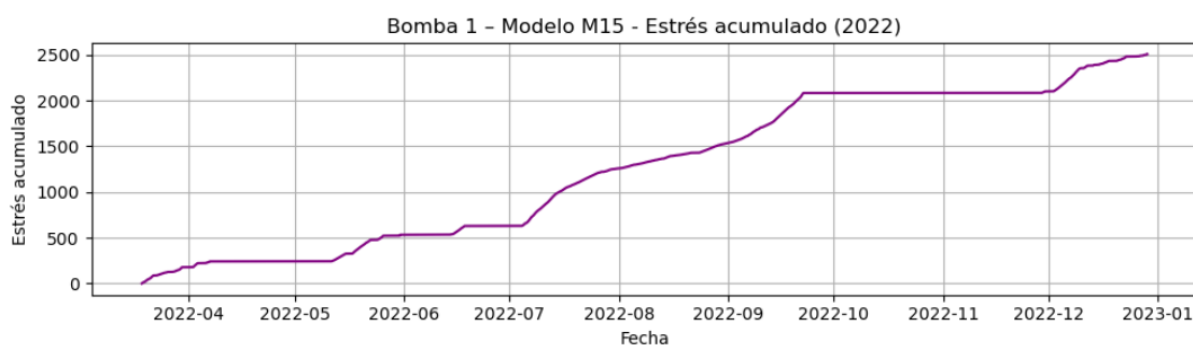


Figura 21. Estrés acumulado del modelo M15 durante el año 2022

En 2023, los tres modelos siguen acumulando estrés, aunque con comportamientos más diferenciados. Como se muestra en la Figura 22, M14 tiene una curva más gradual y sostenida, con pocas pausas. M15, en cambio, muestra un crecimiento más irregular, con tramos planos y saltos más bruscos. Por su parte, M19 tiene varias subidas bruscas en el primer semestre y después su pendiente se estabiliza. Aunque los tres modelos responden de forma distinta, todos terminan el año con un estrés acumulado alto, lo que confirma que el fallo sigue activo.

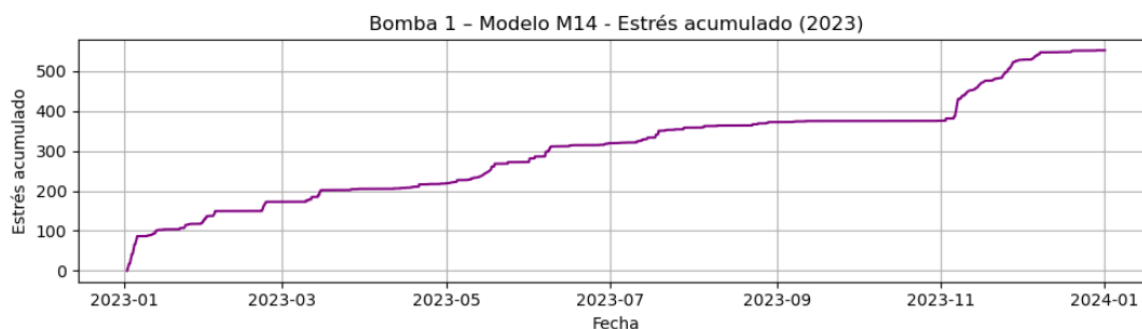


Figura 22. Estrés acumulado del modelo M14 durante el año 2023

La Figura 23 muestra de forma conjunta cómo ha evolucionado el estrés acumulado por modelo desde 2021. Aunque las curvas no se combinan, permiten ver de una manera clara

cómo se alterna el protagonismo de cada modelo según el periodo. En algunos tramos es M14 el que más crece, en otros es M15, y en otros M19. Las diferencias en la pendiente o en el momento en que cada uno reacciona aportan contexto y refuerzan la confianza en el sistema.

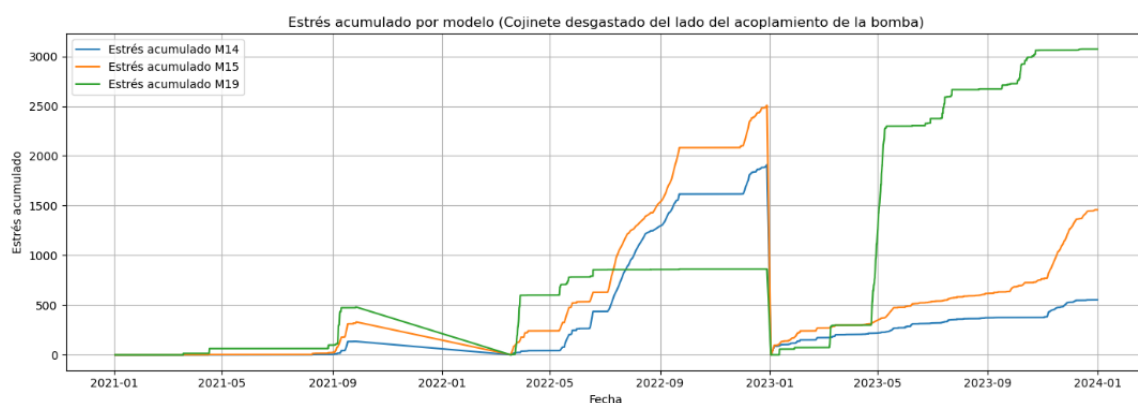


Figura 23. Estrés acumulado total por modelo del modo de fallo 5 en el periodo 2021–2023

Por último, en la Figura 24, se representa el estrés acumulado total del modo de fallo. Esta curva recoge la evolución conjunta del sistema, sumando el estrés detectado por los tres modelos. En ella se ve con claridad cómo a partir de mayo de 2022 comienza una subida constante del estrés que continúa sin interrupciones a lo largo de 2023. Este comportamiento confirma que el sistema detecta un cambio sostenido en el funcionamiento del cojinete.

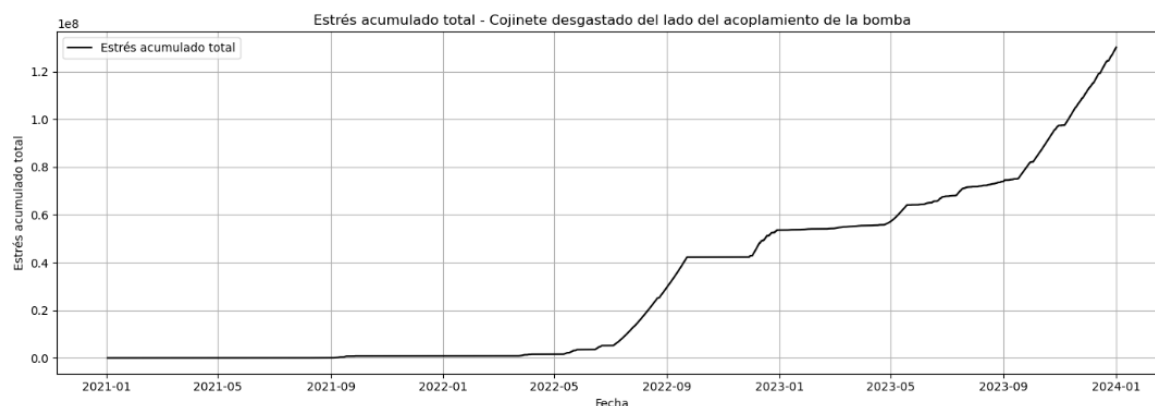


Figura 24. Estrés acumulado total del modo de fallo 5 en el periodo 2021–2023

6.1.5 COJINETE DE EMPUJE DESGASTADO

Este modo de fallo se ha abordado utilizando dos modelos: M18, que predice el desplazamiento del cojinete de empuje, y M23, centrado en su temperatura. Aunque se trata de señales diferentes, ambas están estrechamente relacionadas con el mismo componente y, por tanto, con el mismo fallo. Ambos modelos han sido entrenados y validados con los datos de la misma bomba, siguiendo el mismo procedimiento que en los modos anteriores.

Durante el entrenamiento con datos de 2020, tanto M18 como M23 muestran un comportamiento bastante estable. En la Figura 25, correspondiente al modelo M18, se observa que el error se mantiene mayoritariamente dentro de los márgenes esperados. Aun así, hay algunas activaciones puntuales, sobre todo en las zonas más irregulares de la señal, que provocan pequeños incrementos en el estrés acumulado. El patrón en el modelo M23 es similar, en general es estable, pero tiene algunas salidas en los tramos iniciales. En conjunto, ambos modelos aprenden con solidez el comportamiento típico de sus respectivas señales.

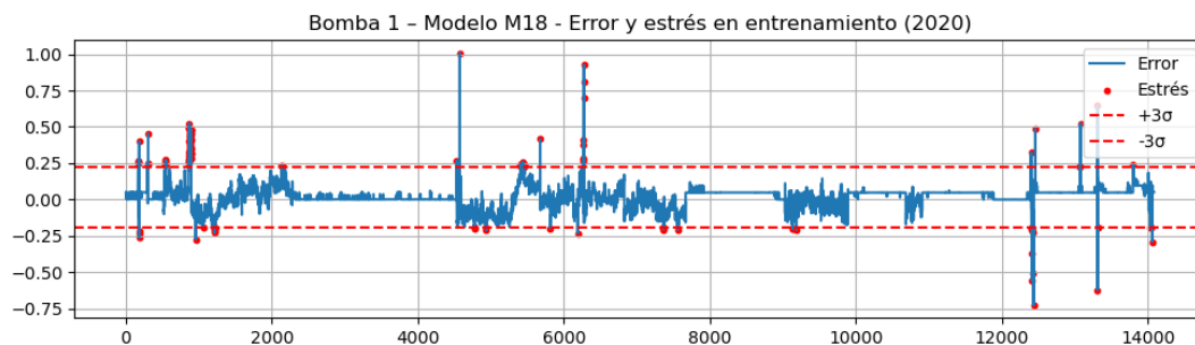


Figura 25. Error de predicción y bandas de confianza del modelo M18 durante el entrenamiento

En 2021, los dos modelos muestran patrones bastante distintos. En el caso del modelo M23, el estrés se acumula de forma muy abrupta a finales de mayo y permanece prácticamente constante durante el resto del año, como se ve en la Figura 26. Por otra parte, el modelo M18 comienza a activarse más tarde, en septiembre, y lo hace de manera más gradual. Aunque el

momento y el ritmo de reacción son diferentes, ambos modelos identifican desviaciones claras en la señal.

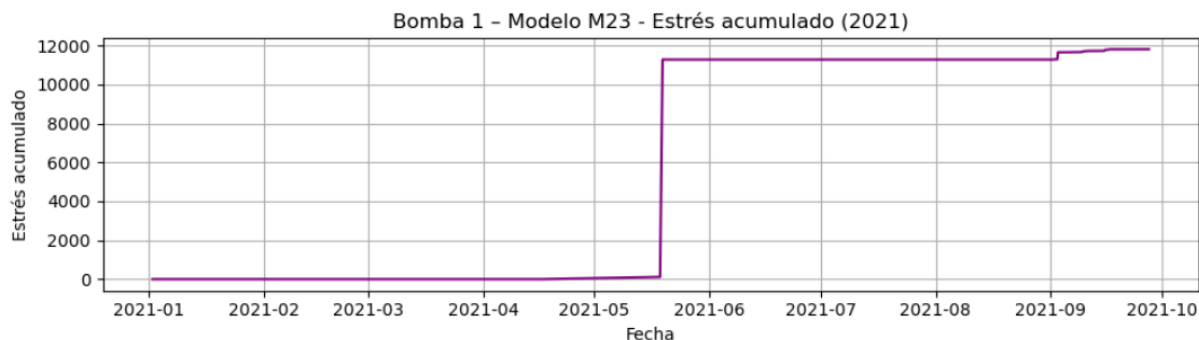


Figura 26. Estrés acumulado del modelo M23 durante el año 2021

Durante 2022, ambos modelos muestran un crecimiento del estrés claro y prolongado. En la Figura 27, que corresponde a el modelo M23, se aprecia una acumulación de estrés bastante pronunciada que empieza en mayo y se mantiene hasta septiembre, donde se estabiliza. En el caso de M18, el crecimiento es bastante parecido, aunque empieza un poco más tarde. Lo más relevante es que los dos modelos coinciden en esta desviación, lo que refuerza la robustez del diagnóstico.

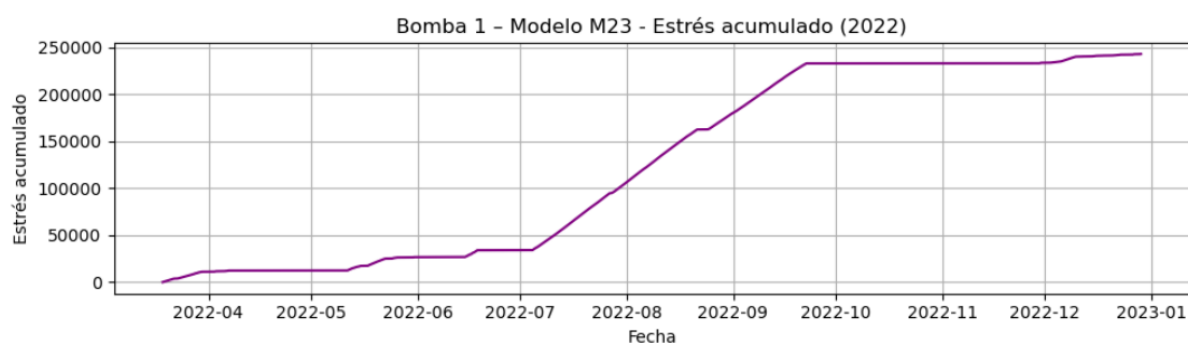


Figura 27. Estrés acumulado del modelo M23 durante el año 2022

En 2023, los dos modelos siguen activos, aunque con curvas de estrés algo distintas. En la Figura 28, se observa cómo el modelo M18 acumula estrés de forma continua, con una curva bastante regular que crece a lo largo de todo el año. En el caso de M23, la pendiente vuelve

a subir a partir de marzo, aunque con más oscilaciones. Aun así, ambos modelos reflejan que el sistema no ha vuelto a una situación estable, sino que persisten las desviaciones.

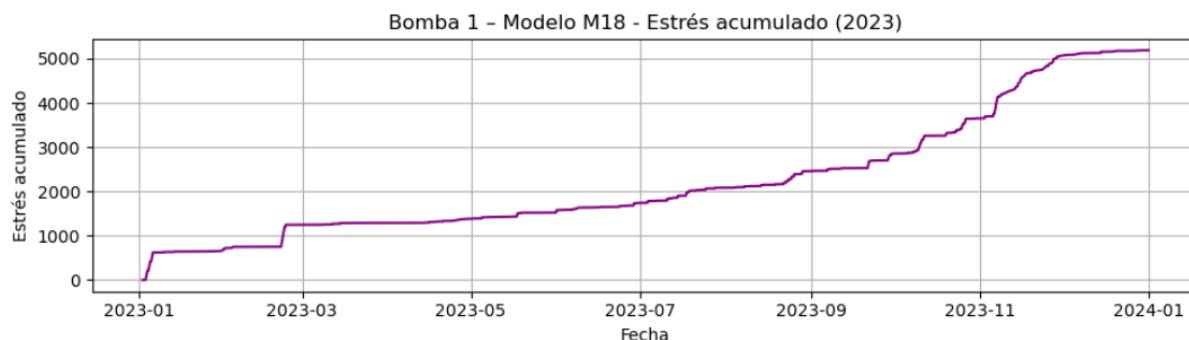


Figura 28. Estrés acumulado del modelo M18 durante el año 2023

La Figura 29 compara el estrés acumulado de los dos modelos a lo largo del periodo de validación. Las curvas permiten observar cómo ambos reaccionan en los mismos tramos de tiempo. La intensidad del crecimiento del estrés varía de un modelo a otro, pero el hecho de que coincidan en los momentos clave refuerza la confianza en el sistema desarrollado.

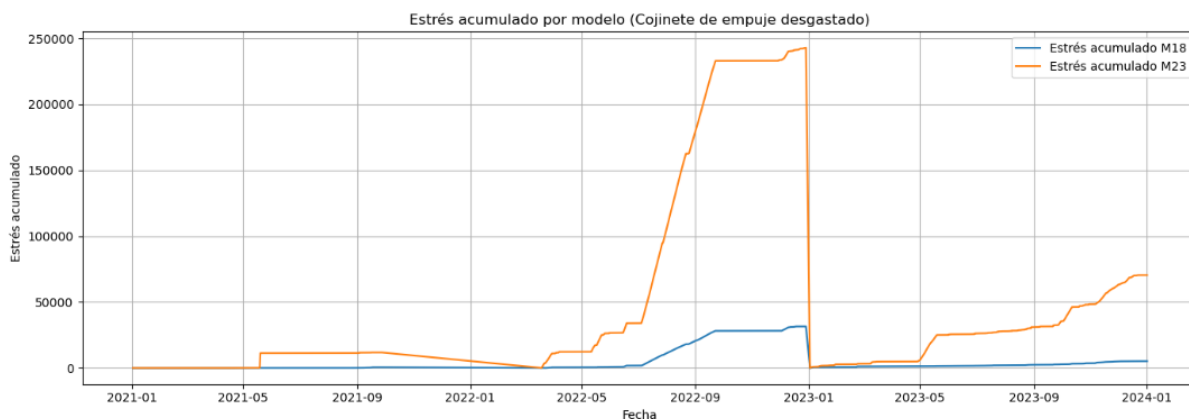


Figura 29. Estrés acumulado total por modelo del modo de fallo 5 en el periodo 2021–2023

Por último, en la Figura 30 se muestra la evolución del estrés acumulado total. Esta curva recoge la información combinada de los dos modelos. A partir de mayo de 2022 se observa un crecimiento muy acusado, que alcanza su punto más alto en octubre de ese año. En los primeros meses de 2023, la curva se estabiliza parcialmente, con una pendiente mucho más

suave, aunque sin llegar a detenerse del todo. A partir de septiembre, vuelve a aumentar de forma más progresiva. Esta evolución confirma que el sistema ha identificado un cambio sostenido en el comportamiento del cojinete.

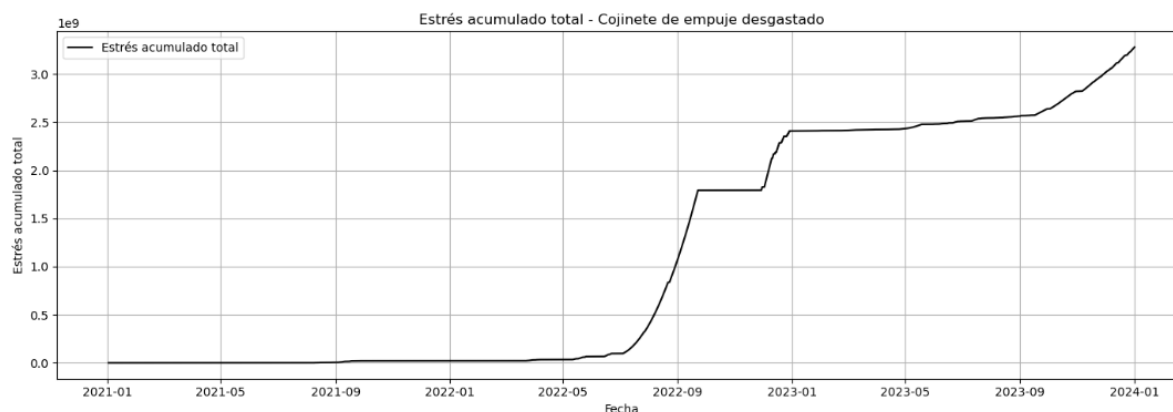


Figura 30. Estrés acumulado total del modo de fallo 6 en el periodo 2021–2023

6.1.6 DESALINEACIÓN DEL EJE

A diferencia del resto, este último modo de fallo ofrece una visión más global, combinando todos los modelos planteados durante el proyecto. A diferencia de los anteriores, en los que se analizaba el comportamiento de uno o varios modelos concretos, aquí se han combinado todos los modelos desarrollados durante el proyecto, del M14 al M23. Esto permite observar cómo responde el sistema completo ante un fallo más complejo, asociado a una posible desalineación del eje. Como los modelos individuales ya han sido evaluados en los apartados anteriores, en este caso el análisis se centra únicamente en las dos gráficas finales: el estrés acumulado por modelo y el estrés acumulado total del sistema.

La Figura 31 muestra cómo evoluciona el estrés en cada uno de los modelos que forman parte del diagnóstico. Aunque el ritmo y la magnitud del crecimiento varían entre ellos, lo más relevante es que todos reaccionan de forma coherente en los mismos tramos de tiempo. A partir de mediados de 2022, la mayoría empieza a acumular estrés con mayor rapidez, y esta tendencia se mantiene en 2023, aunque con algunas pausas y periodos más estables. El hecho de que haya cierta variabilidad entre modelos no es un problema, ya que cada uno está

especializado en una señal distinta, y es normal que algunos reaccionen antes o que lo hagan de forma más intensa que otros. Esta diversidad de respuestas, como ya se ha visto en modos de fallo anteriores, refuerza la solidez del sistema debido a que permite captar matices distintos de una misma situación.

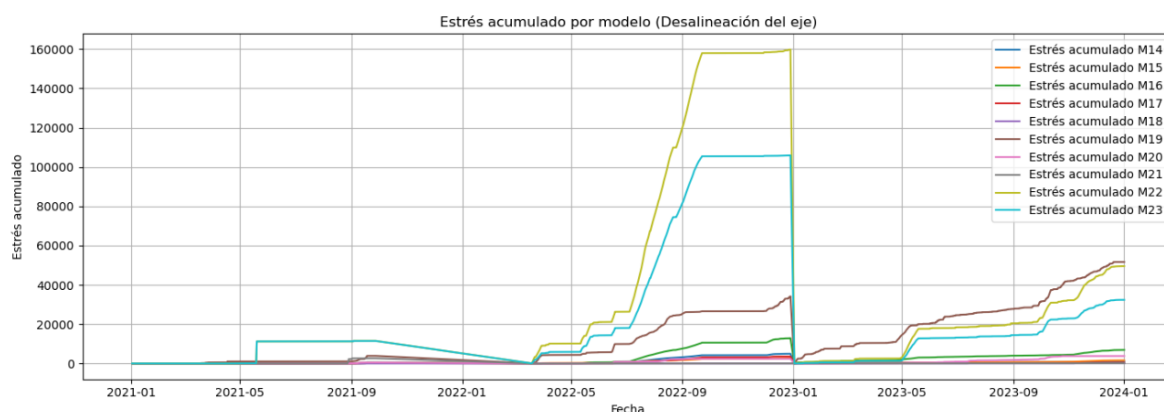


Figura 31. Estrés acumulado total por modelo del modo de fallo 8 en el periodo 2021–2023

La Figura 32 representa la evolución del estrés acumulado total del sistema para este modo de fallo. Es decir, el resultado combinado de todos los modelos implicados. Durante los primeros meses, la gráfica se mantiene prácticamente plana: no se detecta ninguna desviación relevante. Pero a partir del mayo de 2022 empieza a crecer con rapidez, marcando un punto de inflexión claro. Después de ese periodo de crecimiento repentino, el estrés se estabiliza brevemente en la primera mitad de 2023, y más adelante vuelve a subir, aunque de forma más progresiva. No es un crecimiento constante, pero sí sostenido. Y eso es precisamente lo que se busca con este enfoque: que el sistema no se active por una anomalía puntual, sino que sea capaz de reconocer cuándo una desviación se consolida en el tiempo.

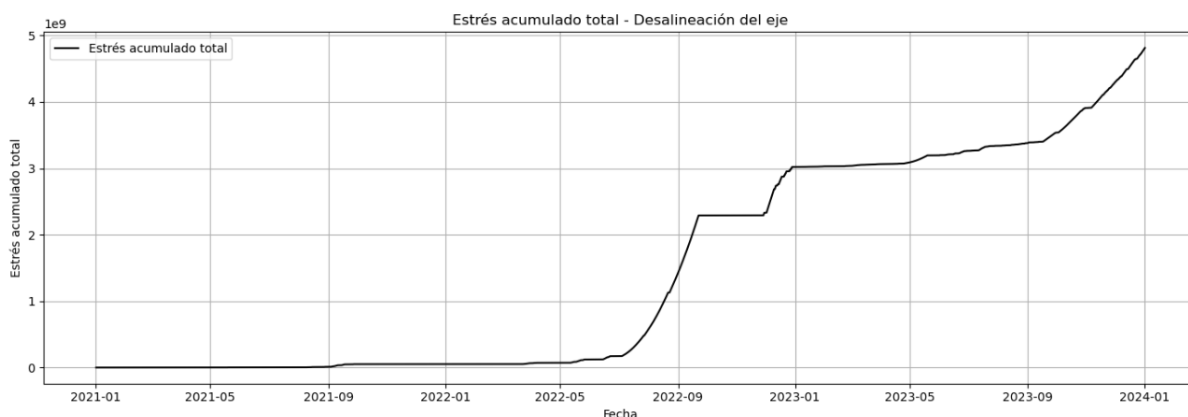


Figura 32. Estrés acumulado total del modo de fallo 8 en el periodo 2021–2023

Este modo de fallo pone en valor la idea central del sistema: diagnosticar a través de múltiples perspectivas complementarias. Al combinar modelos entrenados sobre señales distintas, como temperaturas, presiones o potencias, el sistema es capaz de reconocer patrones anómalos de forma más robusta, captando fallos incluso en fases tempranas. Que todos los modelos apunten a los mismos tramos como problemáticos, es una señal clara de que el enfoque funciona.

Además, este tipo de análisis no solo permite detectar que algo va mal, sino también seguir su evolución en el tiempo. Esto permite plantear un enfoque de mantenimiento más preventivo, en el que las decisiones no se tomen cuando el problema ya ha estallado, sino cuando el sistema lleva tiempo dando señales de que algo no va del todo bien.

Capítulo 7. CONCLUSIONES Y TRABAJOS FUTUROS

A lo largo de este proyecto se ha desarrollado y validado un sistema de diagnóstico basado en aprendizaje por refuerzo, capaz de detectar desviaciones en el comportamiento de señales clave asociadas al estado de distintas bombas de circulación. La idea ha sido entrenar un conjunto de modelos con datos reales de planta y utilizar sus predicciones para calcular un indicador de estrés acumulado, que permita identificar posibles fallos antes de que se hagan evidentes.

Uno de los objetivos principales era comprobar si el uso de técnicas como TD3 tenía sentido en un entorno industrial real. A diferencia de otros trabajos más teóricos o basados en simulaciones, aquí se ha trabajado con datos reales, tal y como se recogen en planta, con todo lo que eso implica: ruido, valores inesperados y condiciones que no siempre son ideales. A pesar de esas dificultades, los resultados han sido buenos. El sistema ha detectado patrones anómalos de forma robusta, incluso cuando los cambios no eran obvios a simple vista.

También se quería validar que el sistema fuera capaz de adaptarse a distintos modos de fallo, no solo a un caso concreto. Para ello, se han definido seis modos diferentes y se han entrenado diez modelos, cada uno centrado en una señal específica. En algunos modos los fallos han sido más claros y fáciles de detectar; en otros, las desviaciones han sido más sutiles y han aparecido de forma progresiva. Pero en todos los casos se ha conseguido construir un diagnóstico coherente, con curvas de estrés que reflejan bastante bien la evolución del fallo.

En conjunto, el proyecto ha cumplido sus objetivos: se ha desarrollado un sistema de diagnóstico que funciona con datos reales, que se adapta a distintos modos de fallo y que permite visualizar la evolución del comportamiento de cada señal de una forma bastante intuitiva. Y todo eso con una arquitectura sencilla, fácil de entender y de integrar.

A partir de aquí, hay muchas formas de seguir trabajando. Una de las más evidentes sería aplicar esta misma lógica a otros equipos dentro de la planta. Ahora que la metodología está

validada, se podría extender fácilmente a otros sistemas que también generen datos de forma continua. También sería interesante estudiar modelos más complejos, que combinen varias señales en lugar de analizarlas por separado. En algunos casos, eso podría ayudar a detectar ciertas desviaciones antes o a entender mejor lo que está ocurriendo.

Otra línea de trabajo sería ajustar la forma en que se calcula el estrés y se entrena el modelo. En este proyecto se ha seguido un enfoque general, con márgenes fijos y funciones de recompensa comunes. Pero en el futuro podrían probarse estrategias más específicas, ajustadas a cada tipo de señal o incluso adaptativas, según el estado del equipo o el tipo de comportamiento que se quiera detectar.

Además, en los modos de fallo en los que hay varios modelos implicados, sería útil estudiar cómo combinar mejor la información que aporta cada uno. Ahora se acumula el estrés de forma independiente, pero quizá podría integrarse de forma más estructurada para generar un diagnóstico conjunto más preciso.

Por último, queda todo lo relacionado con su integración real en planta. Qué tipo de alertas emitir, cómo presentar la información para que se entienda bien, cómo enlazarlo con los sistemas de gestión de mantenimiento que ya existen y cómo asegurarse de que los técnicos lo interpreten fácilmente. Ese paso sería clave para que el sistema se convierta en una herramienta que realmente se utilice.

Por otro lado, una vez validado el enfoque en una planta de generación eléctrica, sería interesante explorar su aplicación en otros sectores industriales con necesidades similares. Muchas industrias, como las plantas químicas, las refinerías o las instalaciones de tratamiento de agua, cuentan también con equipos críticos que generan datos de forma continua y donde los fallos pueden tener consecuencias importantes. En todos esos casos, un sistema como este, que no depende de etiquetados manuales ni de modelos excesivamente complejos, podría ayudar a anticiparse a los problemas y planificar el mantenimiento de forma más inteligente.

En definitiva, lo que se ha hecho en este proyecto no es una solución cerrada, sino una base sólida. Una forma de demostrar que este enfoque tiene sentido, que puede adaptarse a la realidad y que tiene potencial para seguir creciendo.

Capítulo 8. BIBLIOGRAFÍA

- [1] Farama Foundation. “Gymnasium Documentation.” 2023. <https://gymnasium.farama.org/>
- [2] Fujimoto, S., van Hoof, H., & Meger, D. “Addressing Function Approximation Error in Actor-Critic Methods.” *Proceedings of the 35th International Conference on Machine Learning*, PMLR 80:1587–1596, 2018. <https://proceedings.mlr.press/v80/fujimoto18a.html>
- [3] Fracttal. “Guía completa del mantenimiento industrial: estrategias, herramientas y mejores prácticas.” *Blog de Fracttal*. 2023. <https://www.fracttal.com/es/blog/guia-completa-del-mantenimiento-industrial-estrategias-herramientas-y-mejores-practicas>
- [4] *GeeksforGeeks*. “What is Reinforcement Learning?” *GeeksforGeeks*. <https://www.geeksforgeeks.org/machine-learning/what-is-reinforcement-learning/>.
- [5] Heng, A., Zhang, S., Tan, A. C., & Mathew, J. “Rotating machinery prognostics: State of the art, challenges and opportunities.” *Mechanical Systems and Signal Processing*, 23(3), 724–739, 2009. <https://doi.org/10.1016/j.ymssp.2008.06.009>
- [6] J. D. Hunter, "Matplotlib: A 2D Graphics Environment," in *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90-95, May-June 2007, doi: 10.1109/MCSE.2007.55.
- [7] Jardine, A. K. S., Lin, D., & Banjevic, D. “A review on machinery diagnostics and prognostics implementing condition-based maintenance.” *Mechanical Systems and Signal Processing*, 20(7), 1483–1510, 2006. <https://doi.org/10.1016/j.ymssp.2005.09.012>
- [8] Lee, J., Wu, F., Zhao, W., Ghaffari, M., Liao, L., & Siegel, D. “Prognostics and health management design for rotary machinery systems—Reviews, methodology and applications.” *Mechanical Systems and Signal Processing*, 42(1–2), 314–334, 2014. <https://doi.org/10.1016/j.ymssp.2013.06.004>

- [9] OpenAI. “Creating Custom Environments.” *Stable-Baselines3 Documentation*. 2021. https://stable-baselines3.readthedocs.io/en/master/guide/custom_env.html
- [10] OpenAI. “Stable-Baselines3: Reliable implementations of reinforcement learning algorithms in PyTorch.” GitHub. 2021. <https://github.com/DLR-RM/stable-baselines3>
- [11] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. “Scikit-learn: Machine Learning in Python.” *Journal of Machine Learning Research*, 12, 2825–2830, 2011. <https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html>
- [12] Sanz-Bobi, M. A., Orbach, S., Bellido-López, F. J., Muñoz, A., González-Calvo, D., & Álvarez-Tejedor, T. “Anomaly Detection of a Cooling Water Pump of a Power Plant Based on its Virtual Digital Twin Constructed with Deep Learning Techniques.” *PHM Society European Conference*, 8(1), 9. 2024. <https://doi.org/10.36001/phme.2024.v8i1.4004>
- [13] Sutton, R. S., & Barto, A. G. *Reinforcement Learning: An Introduction*. 2ª edición, MIT Press, 2018. <https://www.andrew.cmu.edu/course/10-703/textbook/BartoSutton.pdf>
- [14] F. Tao, H. Zhang, A. Liu and A. Y. C. Nee, "Digital Twin in Industry: State-of-the-Art," in *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405-2415, April 2019, doi: 10.1109/TII.2018.2873186.
- [15] S. van der Walt, S. C. Colbert and G. Varoquaux, "The NumPy Array: A Structure for Efficient Numerical Computation," in *Computing in Science & Engineering*, vol. 13, no. 2, pp. 22-30, March-April 2011, doi: 10.1109/MCSE.2011.37.
- [16] IBM Think. “What is Predictive Maintenance?” *IBM Think*, 9 May 2023. <https://www.ibm.com/think/topics/predictive-maintenance>

ANEXO I: ALINEACIÓN DEL PROYECTO CON LOS ODS

Aunque este proyecto tiene un enfoque claramente técnico, también se relaciona de forma directa con algunos de los grandes retos que plantea el desarrollo sostenible. La mejora del diagnóstico y del mantenimiento en entornos industriales no solo implica una mayor eficiencia operativa, sino que también puede tener un impacto positivo en el uso de recursos, en la forma en que se gestiona la energía y en la organización del trabajo dentro de una planta. Por eso, se puede decir que este trabajo está alineado con varios de los Objetivos de Desarrollo Sostenible (ODS) definidos por la ONU en la Agenda 2030.

A continuación, se explican brevemente los ODS con los que el proyecto tiene una relación más directa.

- ODS 7: Energía asequible y no contaminante. Cuando las bombas de agua funcionan dentro de sus condiciones normales, la planta puede mantener su rendimiento sin consumir más energía de la necesaria. El sistema propuesto ayuda a detectar cuándo empiezan a surgir desviaciones, lo que permite actuar a tiempo y evitar que el equipo se deteriore. Esto mejora la eficiencia global del sistema y, por tanto, reduce el consumo de energía y los costes asociados a paradas o a pérdidas de rendimiento. Aunque no se trata de una solución que genere energía limpia por sí sola, sí que ayuda a que la producción sea más estable y eficiente.
- ODS 9: Industria, innovación e infraestructura. Este objetivo busca promover una industria más sostenible, basada en la innovación y en el desarrollo de infraestructuras resilientes y eficientes. El proyecto encaja muy bien aquí, ya que propone introducir técnicas de inteligencia artificial en el mantenimiento de equipos críticos dentro de una planta eléctrica. Se trata de una aplicación real, no simulada,

en la que se demuestra que es posible usar algoritmos como TD3 con datos reales, sin necesidad de entornos artificiales.

Además, el hecho de que el modelo aprenda directamente de los datos históricos de la planta permite crear sistemas de diagnóstico más flexibles, adaptados a cada instalación concreta, en lugar de depender de modelos genéricos que no siempre reflejan la realidad del proceso. Esto supone un paso adelante en la forma de gestionar el mantenimiento industrial.

- ODS 12: Producción y consumo responsables. Uno de los objetivos de este trabajo es mejorar la forma en que se toman decisiones de mantenimiento. En lugar de actuar por calendario o cuando el fallo ya ha ocurrido, se propone intervenir cuando los datos indican que algo está empezando a ir mal. Esto evita sustituir piezas en buen estado, reduce el número de inspecciones innecesarias y mejora la eficiencia general del sistema.

Además, al reducir el número de intervenciones urgentes o no planificadas, también se optimiza el uso de los recursos técnicos, materiales y humanos. Todo esto encaja dentro del enfoque de consumo responsable que promueve el ODS 12, tanto en lo que se refiere a materiales como a energía o tiempo de trabajo.

En conjunto, este proyecto va más allá de una mejora técnica aplicada al mantenimiento industrial. A través del uso de inteligencia artificial y datos reales de operación, se contribuye a una gestión más eficiente de la energía, a la modernización de procesos clave en la industria y a una utilización más responsable de los recursos disponibles. Esto demuestra que la ingeniería puede ser una gran herramienta para avanzar hacia un modelo de desarrollo más sostenible y coherente con los retos actuales.

ANEXO II: IMPLEMENTACIÓN DEL SISTEMA PARA EL CASO DE DESALINEACIÓN DEL EJE

A continuación, se incluye el código correspondiente al modo de fallo 8 (desalineación del eje), que representa el caso más completo y general del sistema desarrollado. En él se entrenan y evalúan diez modelos distintos (M14 a M23), siguiendo la misma lógica empleada en el resto de los modos. El resto de las implementaciones difieren únicamente en los modelos utilizados para cada fallo.

Librerías y carga de datos

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import MinMaxScaler
from sklearn.metrics import mean_squared_error
import gymnasium as gym
import gym.spaces as spaces
from stable_baselines3 import TD3
from stable_baselines3.common.noise import NormalActionNoise

# Cargar y preparar datos
df = pd.read_csv("BESO_ST51_AC_BAC1.csv", sep=';', decimal=',')
df['index'] = pd.to_datetime(df['index'], dayfirst=True)
df = df.sort_values('index')
df = df[(df['BESC05S1.dwatt'] > 5) & (df['BESC05G1.DWATT'] > 5)]
```

Configuración de modelos

```
modelos_config = {
    "M14": {
        "target": "BESC05SB2.10PAC10CY002XJ31", # Desp_Coj_LAB_Dir1
        "inputs": [
            "BESC05S1.dwatt", # Pot_Turb_Vapor
            "BESC05G1.DWATT", # Pot_Turb_Gas
            "BESC05SB2.10PAC10CP001XJ01" # Presion_Descarga
        ]
    },
    # ... otros modelos ...
}
```

```
"M15": {
  "target": "BESC05SB2.10PAC10CY002XJ41", # Desp_Coj_LAB_Dir2
  "inputs": [
    "BESC05S1.dwatt", # Pot_Turb_Vapor
    "BESC05G1.DWATT", # Pot_Turb_Gas
    "BESC05SB2.10PAC10CP001XJ01" # Presion_Descarga
  ]
},
"M16": {
  "target": "BESC05SB2.10PAC10CY002XJ11", # Desp_Coj_LAOB_Dir1
  "inputs": [
    "BESC05S1.dwatt", # Pot_Turb_Vapor
    "BESC05G1.DWATT", # Pot_Turb_Gas
    "BESC05SB2.10PAC10CP001XJ01" # Presion_Descarga
  ]
},
"M17": {
  "target": "BESC05SB2.10PAC10CY002XJ21", # Desp_Coj_LAOB_Dir2
  "inputs": [
    "BESC05S1.dwatt", # Pot_Turb_Vapor
    "BESC05G1.DWATT", # Pot_Turb_Gas
    "BESC05SB2.10PAC10CP001XJ01" # Presion_Descarga
  ]
},
"M18": {
  "target": "BESC05SB2.10PAC10CY001XJ21", # Desp_Coj_Emp_Axial
  "inputs": [
    "BESC05SB2.10PAC10CT001XJ11", # Temp_Coj_Emp
    "BESC05S1.dwatt", # Pot_Turb_Vapor
    "BESC05G1.DWATT", # Pot_Turb_Gas
    "BESC05SB2.10PAC10CP001XJ01" # Presion_Descarga
  ]
},
"M19": {
  "target": "BESC05SB2.10PAC10CT002XJ81", # Temp_Coj_LAM
  "inputs": [
    "BESC05SB2.10PAC10CT002XJ11", # Temp_Fase_R_Motor
    "BESC05S1.dwatt", # Pot_Turb_Vapor
    "BESC05G1.DWATT", # Pot_Turb_Gas
    "BESC05SB2.10PAC10CP001XJ01" # Presion_Descarga
  ]
},
"M20": {
  "target": "BESC05SB2.10PAC10CT002XJ81", # Temp_Coj_LAM
  "inputs": [
    "BESC05SB2.10PAC10CT002XJ11", # Temp_Fase_R_Motor
    "BESC05S1.dwatt", # Pot_Turb_Vapor
    "BESC05G1.DWATT", # Pot_Turb_Gas
    "BESC05SB2.10PAC10CP001XJ01" # Presion_Descarga
  ]
},
"M21": {
  "target": "BESC05SB2.10PAC10CT002XJ61", # Temp_Coj_Emp_Motor
```

```

    "inputs": [
        "BESC05SB2.10PAC10CT002XJ11",    # Temp_Fase_R_Moto
        "BESC05S1.dwatt",                  # Pot_Turb_Vapor
        "BESC05G1.DWATT",                  # Pot_Turb_Gas
        "BESC05SB2.10PAC10CP001XJ01"      # Presion_Descarga
    ]
},
"M22": {
    "target": "BESC05SB2.10PAC10CT001XJ11", # Temp_Coj_Emp
    "inputs": [
        "BESC05SB2.10PAC10CT002XJ11",    # Temp_Fase_R_Moto
        "BESC05S1.dwatt",                  # Pot_Turb_Vapor
        "BESC05G1.DWATT",                  # Pot_Turb_Gas
        "BESC05SB2.10PAC10CP001XJ01"      # Presion_Descarga
    ]
},
"M23": {
    "target": "BESC05SB2.10PAC10CT001XJ11", # Temp_Coj_Emp
    "inputs": [
        "BESC05SB2.10PAC10CT002XJ11",    # Temp_Fase_R_Motor
        "BESC05S1.dwatt",                  # Pot_Turb_Vapor
        "BESC05G1.DWATT",                  # Pot_Turb_Gas
        "BESC05SB2.10PAC10CP001XJ01"      # Presion_Descarga
    ]
}
}

```

Separar, filtrar y escalar los datos

```

# Separar por año antes de filtrar
df_train_raw = df[df['index'].dt.year == 2020].copy()
df_test_all = df[df['index'].dt.year >= 2021].copy() # No se filtra

# Filtrar outliers en entrenamiento
for col in input_cols + [target_col]:
    Q1 = df_train_raw[col].quantile(0.25)
    Q3 = df_train_raw[col].quantile(0.75)
    IQR = Q3 - Q1
    lower = Q1 - 1.5 * IQR
    upper = Q3 + 1.5 * IQR
    df_train_raw = df_train_raw[(df_train_raw[col] >= lower) &
(df_train_raw[col] <= upper)]

df_train = df_train_raw.dropna()
scaler_x = MinMaxScaler()
scaler_y = MinMaxScaler()
X_train = scaler_x.fit_transform(df_train[input_cols])
y_train = scaler_y.fit_transform(df_train[[target_col]])

```

Definición del entorno

```
class PumpEnv(gym.Env):
    def __init__(self, X, y, max_steps=200):
        super().__init__()
        self.X = X
        self.y = y
        self.n = len(X)
        self.max_steps = max_steps
        self.observation_space = spaces.Box(low=0, high=1,
shape=(X.shape[1],), dtype=np.float32)
        self.action_space = spaces.Box(
            low=np.array([0.0], dtype=np.float32),
            high=np.array([1.0], dtype=np.float32),
            dtype=np.float32
        )
        self.prev_error = None

    def reset(self, seed=None, options=None):
        self.step_idx = 0
        self.idx = np.random.randint(0, self.n - self.max_steps)
        self.prev_error = None
        return np.array(self.X[self.idx], dtype=np.float32)

    def step(self, action):
        real = self.y[self.idx][0]
        pred = np.clip(action[0], 0, 1)
        error = abs(real - pred)
        peso = 1 + real
        reward = - peso * (10 * error + 50 * error**2)
        if self.prev_error is not None:
            reward += 20 * (self.prev_error - error)
        self.prev_error = error
        self.idx += 1
        self.step_idx += 1
        terminated = self.step_idx >= self.max_steps or self.idx >= self.n
        truncated = False
        done = terminated or truncated
        obs = np.array(self.X[self.idx], dtype=np.float32) if not terminated
        else np.zeros_like(self.X[0], dtype=np.float32)
        return obs, reward, done, {'real': real, 'pred': pred}
```

Entrenamiento de los modelos y evaluación

```
env = PumpEnv(X_train, y_train)
action_noise = NormalActionNoise(mean=np.zeros(1), sigma=0.2 * np.ones(1))
model = TD3("MlpPolicy", env, action_noise=action_noise, verbose=0,
batch_size=256, learning_rate=1e-3, gamma=0.60, policy_kwargs=dict(net_arch=[256,
256]))
```

```
episodes = 150
timesteps_per_episode = 150
for _ in range(episodes):
    obs = env.reset()
    for _ in range(timesteps_per_episode):
        action, _ = model.predict(obs, deterministic=False)
        obs, _, done, _ = env.step(action)
        if done:
            break
    model.learn(total_timesteps=timesteps_per_episode,
reset_num_timesteps=False)

def predict_and_inverse(model, X, y_true):
    y_pred = []
    for x in X:
        a, _ = model.predict(x, deterministic=True)
        y_pred.append(a[0])
    y_pred = np.array(y_pred).reshape(-1, 1)
    return scaler_y.inverse_transform(y_true),
scaler_y.inverse_transform(y_pred), y_pred

y_train_inv, y_train_pred, _ = predict_and_inverse(model, X_train, y_train)
error_train = y_train_inv.flatten() - y_train_pred.flatten()
std_error = np.std(error_train)
rmse_train = np.sqrt(mean_squared_error(y_train_inv, y_train_pred))
```

Cálculo del estrés

```
mean_error_train = np.mean(error_train)
lim_sup_train = mean_error_train + 3 * std_error
lim_inf_train = mean_error_train - 3 * std_error
stress_points_train = np.where((error_train > lim_sup_train) | (error_train <
lim_inf_train))[0]
stress_contribution_train = np.where(error_train > lim_sup_train, error_train
- lim_sup_train, 0)
cumulative_stress_train = np.cumsum(stress_contribution_train)
```