



**COMILLAS**  
UNIVERSIDAD PONTIFICIA

ICAI

# GRADO EN INGENIERÍA EN TECNOLOGÍAS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

## **Evaluación de técnicas de Deep Learning para la detección de anomalías en máquinas rotativas**

Autor: Carlota López Argote

Director: Antonio Muñoz San Roque

Codirector: Miguel Ángel Sanz Bobi

Madrid

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

**Evaluación de técnicas de Deep Learning para la detección de anomalías en máquinas rotativas**

en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el curso académico 2022/23 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.: Carlota López Argote

Fecha: 04/06/2023

Autorizada la entrega del proyecto

**DIRECTORES DEL PROYECTO**



Fdo.: Antonio Muñoz San Roque

Fecha: 04/06/2023

Fdo.: Miguel Ángel Sanz Bobi

Fecha: 04/06/2023





**COMILLAS**  
UNIVERSIDAD PONTIFICIA

ICAI

# GRADO EN INGENIERÍA EN TECNOLOGÍAS DE TELECOMUNICACIÓN

TRABAJO FIN DE GRADO

## **Evaluación de técnicas de Deep Learning para la detección de anomalías en máquinas rotativas**

Autor: Carlota López Argote

Director: Antonio Muñoz San Roque

Codirector: Miguel Ángel Sanz Bobi

Madrid



# Agradecimientos

En primer lugar, quiero agradecer a mi familia por su constante apoyo, paciencia y motivación.

También quiero agradecer a mis directores, Antonio y Miguel Ángel, por su guía experta, su dedicación y su invaluable orientación a lo largo de esta investigación. Su visión y conocimiento han sido fundamentales para el desarrollo de este trabajo.

Agradezco de corazón a mis amigos por su apoyo incondicional a lo largo de estos 5 años, siempre animándome a dar lo mejor de mí.



# **Evaluación de técnicas de Deep Learning para la detección de anomalías en máquinas rotativas**

**Autor: López Argote, Carlota.**

Director: Muñoz San Roque, Antonio.

Sanz Bobi, Miguel Ángel.

Entidad Colaboradora: ICAI – Universidad Pontificia Comillas

## **RESUMEN DEL PROYECTO**

Este trabajo evalúa técnicas de Deep Learning para la detección de anomalías en maquinaria rotativa. Se desarrollan modelos combinando aprendizaje profundo y aprendizaje profundo reforzado, logrando resultados prometedores. Los modelos propuestos ofrecen una herramienta eficiente y confiable para el monitoreo y mantenimiento predictivo en entornos industriales.

**Palabras clave:** Aprendizaje Profundo, Maquinaria Rotativa, Aprendizaje por Refuerzo, CNN

### **1. Introducción**

En este proyecto se ha llevado a cabo una evaluación exhaustiva de diversas técnicas de Deep Learning (DL) para la detección de anomalías en máquinas rotativas, utilizando datos de sensores de vibración. Se destaca la importancia de los rodamientos ya que son la causa de entre el 40% y el 70% de los fallos en maquinaria [1] [2]. Por esta razón, es esencial mantener una vigilancia continua de la condición de los rodamientos en estas máquinas rotativas para detectar cualquier anomalía lo más pronto posible y diagnosticar sus causas. La detección temprana de fallos en estas máquinas es fundamental para garantizar su operación y mantenimiento óptimos en entornos industriales, como las centrales de generación eléctrica.

El procesamiento de los datos de los sensores de vibración proporciona información valiosa para identificar patrones y señales de alarma que indican posibles fallos en las máquinas rotativas [3]. En este proyecto se han explorado diferentes enfoques y técnicas de DL, como redes neuronales convolucionales (CNN) y aprendizaje por refuerzo profundo (DRL), con el objetivo de desarrollar un modelo preciso y fiable para la detección de fallos.

Se han realizado pruebas y comparaciones exhaustivas para evaluar el rendimiento y la efectividad de cada modelo en la detección y clasificación de anomalías. El objetivo final es contar con un modelo de DL implementable en entornos industriales reales, mejorando así la seguridad, la confiabilidad y la eficiencia energética de las operaciones.

Este proyecto se justifica por la necesidad de superar las limitaciones de los enfoques tradicionales [4] de detección de fallos en máquinas rotativas y ofrecer soluciones más efectivas. La combinación de técnicas de DL, adaptabilidad a diferentes velocidades de rotación y enfoque en la eficiencia energética posiciona este proyecto como una propuesta atractiva y altamente beneficiosa para las empresas, contribuyendo al crecimiento económico, la competitividad industrial y la sostenibilidad ambiental.



## 2. Definición del proyecto

En este proyecto, se ha utilizado un conjunto de datos público con el propósito de entrenar y validar los modelos de DL para la detección de anomalías en maquinaria rotativa. Esto ha permitido utilizar los resultados obtenidos como benchmark y así poder realizar una comparación con estudios similares. Los datos utilizados en este trabajo provienen de un estudio realizado en la Universidad de Paderborn [5].

El estudio aborda el procesamiento de datos, el desarrollo de modelos y el análisis de resultados obtenidos. Para visualizar de manera concisa el flujo de trabajo seguido en el proyecto, se ha incluido la Figura 1. A través de esta figura, se pueden observar de manera visual las etapas clave del proceso y su interconexión.

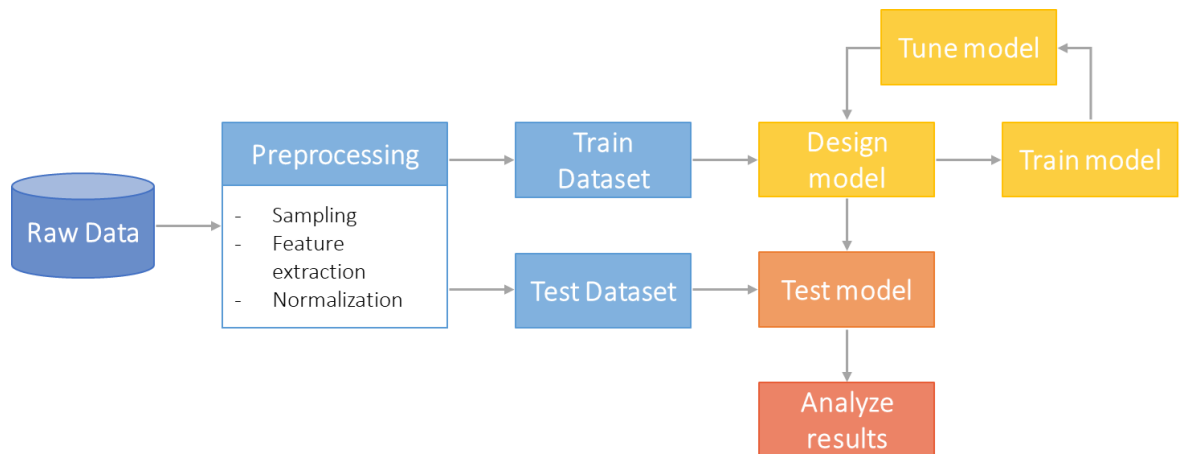


Figura 1 - Diagrama de flujo del desarrollo del proyecto

Para la realización de este trabajo se ha tomado la decisión de utilizar datos de todos los experimentos con el objetivo de diseñar un modelo que pueda aplicarse de manera generalizada a cualquier tipo de fallo. Se toma una muestra representativa del conjunto de datos, garantizando el equilibrio de las clases de fallo. Por otro lado, se emplean los datos de la señal de vibración ya que se espera que estos proporcionen un mayor rendimiento en los modelos y se transforman al espectro de la frecuencia para facilitar la identificación de patrones. Además de utilizar los datos de la señal de vibración, también se ha añadido la velocidad de rotación como dato de entrada en los modelos ya que influye en las frecuencias de la señal así que incluirla permite diseñar un modelo capaz de funcionar bajo diferentes configuraciones.

Una vez que los datos han sido procesados, se procede al desarrollo de diversos modelos de Deep Learning (DL) y Deep Reinforcement Learning (DRL) para la detección de anomalías en maquinaria. Estos modelos se entrenan utilizando el conjunto de datos obtenido y se evalúan utilizando datos de prueba independientes. Sin embargo, los detalles específicos de los modelos serán abordados en la siguiente sección.

Con estos enfoques y consideraciones, el proyecto busca implementar un sistema eficaz de detección de anomalías en maquinaria, utilizando modelos de DL y DRL entrenados con datos relevantes y representativos.

### 3. Descripción de los modelos

Se han diseñado tres modelos diferentes para abordar la detección de anomalías en maquinaria. Estos modelos son una Convolutional Neural Network (CNN), una Double Deep Q-Network (DDQN) como método de RL, y un modelo conjunto que combina los dos anteriores para construir un enfoque más robusto.

El primer modelo es una CNN, que ha demostrado ser altamente efectiva en el procesamiento de datos secuenciales, como pueden ser las señales de vibración de la maquinaria. La CNN utiliza capas convolucionales para extraer características relevantes de los datos de entrada y capas completamente conectadas para realizar la clasificación. La ventaja principal de la CNN es su capacidad para capturar patrones complejos en los datos, lo que resulta especialmente útil para detectar anomalías en señales de vibración.

El segundo modelo es una DDQN, que es una variante del algoritmo de Q-Learning utilizado en RL. Esta técnica utiliza una red neuronal para tomar decisiones basadas en la información recopilada mediante la interacción con el entorno. En el contexto de la detección de anomalías en maquinaria rotativa, la DDQN se utiliza para analizar los datos y detectar posibles fallos o irregularidades en el funcionamiento de la maquinaria. Esta técnica es efectiva para adaptarse entornos dinámicos y tomar decisiones informadas sobre la presencia de anomalías.

Finalmente, se ha desarrollado un modelo conjunto que combina los beneficios de la CNN y la DDQN. Este modelo aprovecha las capacidades de la CNN para capturar patrones complejos en los datos de vibración, y utiliza la DDQN para aprender las mejores acciones a tomar en función de las características extraídas por la CNN. Al combinar estos dos enfoques, se logra un modelo más robusto y preciso en la detección de anomalías en maquinaria rotativa.

Para una mejor comprensión de la estructura y el diseño del modelo conjunto, se incluye la Figura 2, que ilustra de manera visual el proceso seguido para el desarrollo del modelo conjunto mediante el ajuste de los pesos de los modelos previos.

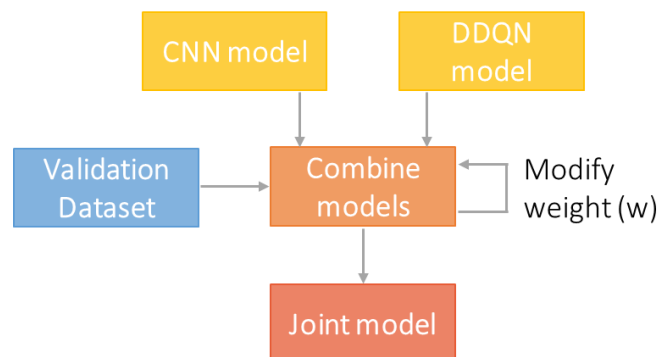


Figura 2 - Proceso de diseño del modelo conjunto

En resumen, se han diseñado tres modelos para abordar la detección de anomalías en maquinaria: una CNN, una DDQN, y un modelo conjunto que combina ambos enfoques.

Cada modelo tiene sus propias ventajas y características únicas, y juntos forman un enfoque completo y robusto para la detección de anomalías en maquinaria rotativa.

#### 4. Resultados

Se evaluaron los modelos individuales (CNN y DDQN) y el modelo conjunto en la detección de anomalías en maquinaria rotativa. El modelo conjunto mostró el mayor accuracy con 94.03%, como se muestra en la Tabla 1, ya que aprovecha las fortalezas de ambos modelos.

Tabla 1 - Accuracy de los modelos: análisis comparativo

	<i>CNN</i>	<i>DDQN</i>	<i>Conjunto</i>
<b>Accuracy</b>	0.9391	0.8481	<b>0.9403</b>

En términos de recall, se observa una disminución en la detección de maquinaria saludable (clase "healthy"), pero esto se justifica para maximizar la detección de fallos, que es crucial en este contexto.

Tabla 2 - Recall de los modelos: análisis comparativo

	<i>CNN</i>	<i>DDQN</i>	<i>Conjunto</i>
<b>Healthy</b>	<b>0.9589</b>	0.9347	0.9558
<b>OR fault</b>	0.9469	0.8156	<b>0.9514</b>
<b>IR fault</b>	0.9101	0.7919	<b>0.9122</b>

El modelo conjunto logra un mejor equilibrio entre la precisión y el recall, mejorando el F1-score en todas las clases de fallos. Además, las matrices de confusión, como la Figura 3, muestran que el modelo conjunto reduce los errores de clasificación para las clases de fallos, aunque también presenta algunos falsos positivos en la categorización de maquinaria sana. En resumen, el enfoque de modelo conjunto, combinando la CNN y la DDQN, resulta efectivo en la detección de anomalías en maquinaria rotativa, mejorando el rendimiento en términos de accuracy, recall y F1-score en comparación con los modelos individuales.

		Predicted Label		
		Healthy	OR fault	IR fault
True Label	Healthy	3395 (32.2%)	31 (0.3%)	126 (1.2%)
	OR fault	35 (0.3%)	3425 (32.5%)	140 (1.3%)
	IR fault	192 (1.8%)	106 (1.0%)	3095 (29.4%)

Figura 3 - Matriz de confusión modelo conjunto

## 5. Conclusiones

En resumen, los modelos desarrollados en este proyecto, basados en técnicas de DL y DRL, han demostrado su eficacia en la detección de anomalías en maquinaria rotativa, superando en muchos aspectos a los modelos tradicionales. Estas contribuciones son relevantes tanto desde el punto de vista académico como para las empresas que buscan mejorar la eficiencia y la seguridad de sus sistemas de monitorización de maquinaria rotativa.

## 6. Referencias

- [1] S. Kim, D. An and J.-H. Choi, "Diagnostics 101: A Tutorial for Fault Diagnostics of Rolling Element Bearing Using Envelope Analysis in MATLAB," *Applied Sciences*, vol. 10, no. 20, p. 7302, Oct. 2020.
- [2] J. van den Hoogen, S. Bloemheuvel and M. Atzmueller, "Classifying Multivariate Signals in Rolling Bearing Fault Detection Using Adaptive Wide-Kernel CNNs," *Applied Sciences*, vol. 11, no. 23, p. 11429, Dic. 2021.
- [3] G. Qian and J. Liu, "Development of deep reinforcement learning-based fault diagnosis method," *Progress in Nuclear Energy*, vol. 152, Sep. 2022.
- [4] M. Hakim, A. A. Borhana Omran, A. Najah Ahmed, M. Al-Waily and A. Abdellatif, "A systematic review of rolling bearing fault diagnoses based on deep learning and transfer learning: Taxonomy, overview, application, open challenges, weaknesses and recommendations," *Ain Shams Engineering Journal*, vol. 14, no. 4, p. 101945, 2023.
- [5] C. Lessmeier, J. K. Kimotho, D. Zimmer and W. Sextro, "Condition Monitoring of Bearing Damage in Electromechanical Drive Systems by Using Motor Current Signals of Electric Motors: A Benchmark Data Set for Data-Driven Classification," *European Conference of the Prognostics and Health Management Society*, 2016.

# **Evaluation of Deep Learning techniques for anomaly detection in rotating machinery**

**Author: López Argote, Carlota.**

Supervisor: Muñoz San Roque, Antonio.

Sanz Bobi, Miguel Ángel.

Collaborating Entity: ICAI – Universidad Pontificia Comillas

## **ABSTRACT**

This work evaluates Deep Learning techniques for anomaly detection in rotating machinery. Models combining deep learning and deep reinforcement learning are developed, achieving promising results. The proposed models provide an efficient and reliable tool for monitoring and predictive maintenance in industrial environments.

**Keywords:** Deep Learning, Rotating Machinery, Reinforcement Learning, CNN

## **1. Introduction**

This project aims to conduct a comprehensive evaluation of various Deep Learning (DL) techniques for anomaly detection in rotating machinery using vibration sensor data. The importance of rolling bearings is highlighted as they are the cause of 40% to 70% of machinery failures [1] [2]. Early detection of faults in these machines is crucial to ensure optimal operation and maintenance in industrial environments such as power generation plants.

Processing vibration sensor data provides valuable insights into identifying patterns and alarm signals indicating potential faults in rotating machinery [3]. This project will explore different DL approaches and techniques, including Convolutional Neural Networks (CNN) and Deep Reinforcement Learning (DRL), with the goal of developing an accurate and reliable model for fault detection.

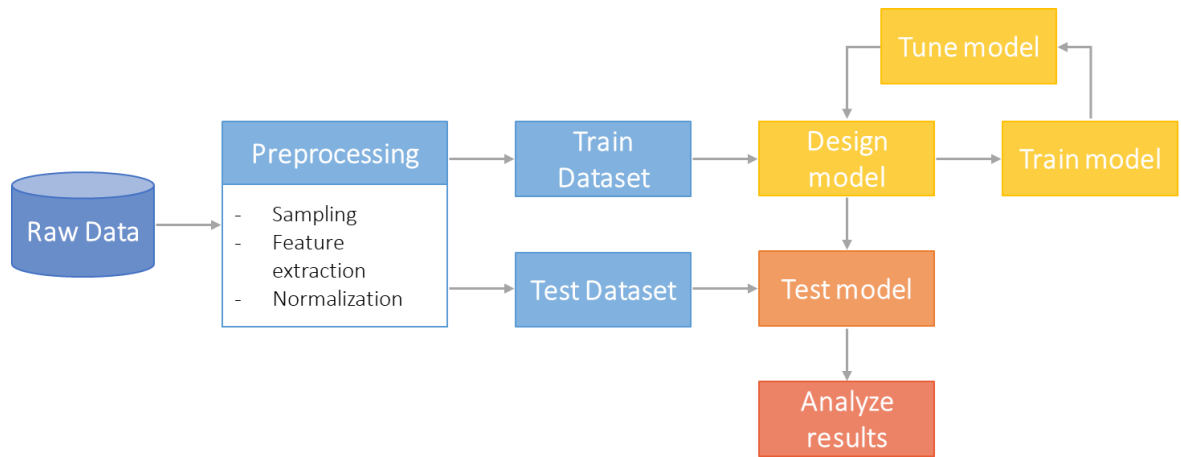
Extensive testing and comparisons will be performed to evaluate the performance and effectiveness of each model in detecting and classifying anomalies. The ultimate objective is to have a DL model that can be implemented in real-world industrial settings, enhancing safety, reliability, and energy efficiency in operations.

This project is justified by the need to overcome the limitations of traditional approaches to fault detection in rotating machinery and provide more effective solutions [4]. The combination of DL techniques, adaptability to different rotation speeds, and a focus on energy efficiency positions this project as an attractive and highly beneficial proposal for companies, contributing to economic growth, industrial competitiveness, and environmental sustainability.

## **2. Project definition**

In this project, a publicly available dataset has been used to train and validate DL models for anomaly detection in rotating machinery. This has allowed the use of the obtained results as a benchmark to compare with similar studies. The data used in this work comes from a study conducted at the University of Paderborn [5].

The study addresses data processing, model development, and analysis of the obtained results. Figure 4 has been included to provide a concise visualization of the workflow followed in the project, showcasing the key stages of the process and their interconnection.



*Figura 4 - Project development workflow diagram*

For this work, the decision has been made to use data from all experiments with the aim of designing a model that can be applied in a generalized manner to any type of fault. A representative sample of the dataset is taken, ensuring balance in the fault classes. Additionally, vibration signal data is used as it is expected to provide better performance in the models and is transformed into the frequency spectrum to facilitate pattern identification. In addition to using vibration signal data, the rotation speed is also included as input data in the models as it influences the signal frequencies, allowing the design of a model capable of functioning under different settings.

Once the data has been processed, various Deep Learning (DL) and Deep Reinforcement Learning (DRL) models are developed for anomaly detection in machinery. These models are trained using the obtained dataset and evaluated using independent test data. However, specific details of the models will be addressed in the following section.

With these approaches and considerations, the project aims to implement an effective system for anomaly detection in machinery, utilizing DL and DRL models trained with relevant and representative data.

### 3. Model description

Three different models have been designed to address anomaly detection in machinery. These models are a Convolutional Neural Network (CNN), a Double Deep Q-Network (DDQN) as a RL method, and a joint model that combines the two previous models to build a more robust approach.

The first model is a CNN, which has proven to be highly effective in processing sequential data, such as vibration signals from machinery. The CNN uses convolutional layers to extract relevant features from the input data and fully connected layers for classification. The main advantage of the CNN is its ability to capture complex patterns in the data, which is particularly useful for detecting anomalies in vibration signals.

The second model is a DDQN, which is a variant of the Q-Learning algorithm used in RL. This technique uses a neural network to make decisions based on the collected information through its interaction with the environment. In the context of anomaly detection in rotating machinery, the DDQN is used to analyze the data and detect possible faults or irregularities in the machinery's operation. This technique is effective in adapting to dynamic environments and making informed decisions about the presence of anomalies.

Finally, a joint model has been developed that combines the benefits of the CNN and the DDQN. This model leverages the CNN's capabilities to capture complex patterns in vibration data and uses the DDQN to learn the best actions to take based on the features extracted by the CNN. By combining these two approaches, a more robust and accurate model is achieved in detecting anomalies in rotating machinery.

For a better understanding of the structure and design of the joint model, Figure 5 is included, which visually illustrates the process followed to develop the joint model by adjusting the weights of the previous models.

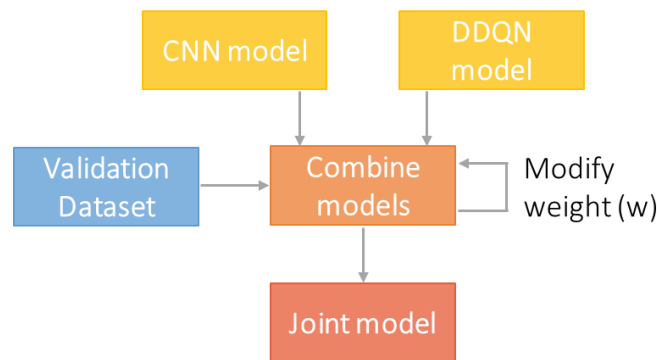


Figura 5 - Design process for the joint model

In summary, three models have been designed to address anomaly detection in machinery: a CNN, a DDQN, and a joint model that combines both approaches. Each model has its own advantages and unique characteristics, and together they form a comprehensive and robust approach to detecting anomalies in rotating machinery.

#### 4. Results

The individual models (CNN and DDQN) and the joint model were evaluated for anomaly detection in rotating machinery. The joint model exhibited the highest accuracy at 94.03%, as shown in Table 3, as it leverages the strengths of both models.

Tabla 3 - Accuracy of the models: comparative analysis

	<i>CNN</i>	<i>DDQN</i>	<i>Conjunto</i>
<b>Accuracy</b>	0.9391	0.8481	<b>0.9403</b>

In terms of recall, a decrease in the detection of healthy machinery is observed, but this is justified to maximize the detection of faults, which is crucial in this context.

Tabla 4 - Recall of the models: comparative analysis

	<b>CNN</b>	<b>DDQN</b>	<b>Conjunto</b>
<b>Healthy</b>	<b>0.9589</b>	0.9347	0.9558
<b>OR fault</b>	0.9469	0.8156	<b>0.9514</b>
<b>IR fault</b>	0.9101	0.7919	<b>0.9122</b>

The combined model achieves a better balance between precision and recall, improving the F1-score in all fault classes. Additionally, the confusion matrices, such as Figure 6, show that the combined model reduces classification errors for fault classes, although it also presents some false positives in the categorization of healthy machinery. In summary, the approach of the ensemble model, combining CNN and DDQN, proves effective in the detection of anomalies in rotary machinery, improving performance in terms of accuracy, recall, and F1-score compared to the individual models.

Joint Model

		Predicted Label		
		Healthy	OR fault	IR fault
True Label	Healthy	3395 (32.2%)	31 (0.3%)	126 (1.2%)
	OR fault	35 (0.3%)	3425 (32.5%)	140 (1.3%)
	IR fault	192 (1.8%)	106 (1.0%)	3095 (29.4%)

Figura 6 - Confusion matrix for the joint model

## 5. Conclusions

In summary, the models developed in this project, based on DL and DRL techniques, have proven their effectiveness in detecting anomalies in rotary machinery, surpassing traditional models in many aspects. These contributions are relevant both from an academic standpoint and for companies seeking to enhance the efficiency and safety of their rotary machinery monitoring systems.

## 6. References

- [1] S. Kim, D. An and J.-H. Choi, "Diagnostics 101: A Tutorial for Fault Diagnostics of Rolling Element Bearing Using Envelope Analysis in MATLAB," Applied Sciences, vol. 10, no. 20, p. 7302, Oct. 2020.



- [2] J. van den Hoogen, S. Bloemheugel and M. Atzmueller, "Classifying Multivariate Signals in Rolling Bearing Fault Detection Using Adaptive Wide-Kernel CNNs," *Applied Sciences*, vol. 11, no. 23, p. 11429, Dic. 2021.
- [3] G. Qian and J. Liu, "Development of deep reinforcement learning-based fault diagnosis method," *Progress in Nuclear Energy*, vol. 152, Sep. 2022.
- [4] M. Hakim, A. A. Borhana Omran, A. Najah Ahmed, M. Al-Waily and A. Abdellatif, "A systematic review of rolling bearing fault diagnoses based on deep learning and transfer learning: Taxonomy, overview, application, open challenges, weaknesses and recommendations," *Ain Shams Engineering Journal*, vol. 14, no. 4, p. 101945, 2023.
- [5] C. Lessmeier, J. K. Kimotho, D. Zimmer and W. Sextro, "Condition Monitoring of Bearing Damage in Electromechanical Drive Systems by Using Motor Current Signals of Electric Motors: A Benchmark Data Set for Data-Driven Classification," *European Conference of the Prognostics and Health Management Society*, 2016..

## *Índice de la memoria*

<b>Capítulo 1. Introducción</b> .....	<b>7</b>
1.1 Motivación del proyecto .....	7
<b>Capítulo 2. Estado de la Cuestión</b> .....	<b>10</b>
<b>Capítulo 3. Definición del Trabajo</b> .....	<b>14</b>
3.1 Justificación .....	14
3.2 Objetivos.....	15
3.3 Metodología .....	16
3.4 Herramientas Utilizadas.....	17
3.5 Planificación y Estimación Económica .....	18
<b>Capítulo 4. Modelos de Deep Learning para la Detección de Anomalías</b> .....	<b>20</b>
4.1 Redes Neuronales Convolucionales (CNN).....	21
4.1.1 Componentes de una red neuronal convolucional.....	21
4.1.2 Funciones de activación.....	24
4.1.3 Aprendizaje en redes neuronales.....	25
4.2 Aprendizaje por Refuerzo y Doble DQN.....	27
4.2.1 Componentes del aprendizaje por refuerzo.....	27
4.2.2 Redes principal y objetivo en la Doble DQN .....	28
4.2.3 Estrategias de exploración y explotación en la Doble DQN.....	29
4.2.4 Actualización de los Q-valores en la Doble DQN.....	30
4.2.5 Replay Memory en la Doble DQN.....	31
4.3 Resumen del capítulo.....	32
<b>Capítulo 5. Caso de Aplicación</b> .....	<b>34</b>
5.1 Descripción y Procesamiento de Datos.....	34
5.2 Modelos Desarrollados .....	38
5.2.1 Modelo de redes neuronales convolucionales.....	39
5.2.2 Modelo de doble DQN .....	45
5.2.3 Modelo conjunto .....	51
5.3 Análisis de Resultados.....	52

5.3.1 Modelo conjunto .....	53
5.3.2 Análisis comparativo con datos de referencia.....	60
<b>Capítulo 6. Conclusiones y Trabajos Futuros .....</b>	<b>66</b>
<b>Capítulo 7. Bibliografía.....</b>	<b>69</b>
<b>ANEXO I: ALINEACIÓN DEL PROYECTO CON LOS ODS.....</b>	<b>71</b>
<b>ANEXO II: Glosario de Términos .....</b>	<b>74</b>
<b>ANEXO III: Repositorio de Código .....</b>	<b>76</b>
7.1 Introducción al repositorio.....	76
7.2 Estructura del repositorio.....	76

## *Índice de figuras*

Figura 1 - Diagrama de flujo del desarrollo del proyecto .....	9
Figura 2 - Proceso de diseño del modelo conjunto .....	10
Figura 3 - Matriz de confusión modelo conjunto .....	12
Figura 4 - Project development workflow diagram .....	14
Figura 5 - Design process for the joint model .....	15
Figura 6 - Confusion matrix for the joint model .....	16
Figura 7 - Componentes de un rodamiento .....	8
Figura 8 - Diagrama de Gantt de la planificación del trabajo .....	19
Figura 9 - Principales tipos de aprendizaje automático .....	20
Figura 10 - Arquitectura de una CNN.....	22
Figura 11 - Convolución de una imagen .....	23
Figura 12 - Función de activación ReLU .....	24
Figura 13 - Función de activación Softmax .....	25
Figura 14 - Aprendizaje por refuerzo: interacción entre agente y entorno .....	28
Figura 15 - Diagrama de flujo del desarrollo del proyecto .....	34
Figura 16 - Sistema de pruebas modular del Paderborn dataset [7] .....	35
Figura 17 - Visualización de las señales en el dominio del tiempo y de la frecuencia.....	37
Figura 18 - Esquema del modelo CNN .....	42
Figura 19 - Aprendizaje del modelo CNN .....	43
Figura 20 - Esquema del modelo DDQN.....	46
Figura 21 - Aprendizaje del modelo DDQN .....	50
Figura 22 - Proceso de diseño del modelo conjunto .....	52
Figura 23 - Matriz de confusión modelo CNN.....	56
Figura 24 - Matriz de confusión modelo DDQN.....	56
Figura 25 - Matriz de confusión modelo conjunto .....	57
Figura 26 - Curvas ROC modelo CNN.....	58

Figura 27 - Curvas ROC modelo DDQN.....	58
Figura 28 - Curvas ROC modelo conjunto.....	59
Figura 29 - Rendimiento de los modelos en escenarios de pocas muestras de datos .....	64
Figura 30 - Objetivos de Desarrollo Sostenible.....	71

## *Índice de tablas*

Tabla 1 - Accuracy de los modelos: análisis comparativo .....	11
Tabla 2 - Recall de los modelos: análisis comparativo .....	11
Tabla 3 - Accuracy of the models: comparative analysis .....	15
Tabla 4 - Recall of the models: comparative analysis .....	16
Tabla 5 - Daños artificiales en el rodamiento.....	36
Tabla 6 - Parámetros de operación bajo distintas configuraciones.....	36
Tabla 7 - Resumen comparativo de los modelos DDQN .....	51
Tabla 8 - Accuracy de los modelos: análisis comparativo .....	53
Tabla 9 - Precisión de los modelos: análisis comparativo .....	54
Tabla 10 - Recall de los modelos: análisis comparativo .....	54
Tabla 11 - F1-score de los modelos: análisis comparativo .....	54
Tabla 12 - Rendimiento de los modelos entrenados con datos de daños artificiales y testeados con datos de daños reales .....	61
Tabla 13 - Rendimiento de los modelos entrenados y testeados con datos reales utilizando 5-fold cross-validation.....	61
Tabla 14 - Rendimiento de los modelos entrenados y testeados con datos reales utilizando 5-fold cross-validation incluyendo experimentos con múltiples fallos.....	62
Tabla 15 - Rendimiento de los modelos mostrando la media del accuracy $\pm$ la desviación típica de 10 experimentos.....	64

## *Índice de ecuaciones*

Ecuación 1 - Función de activación ReLU.....	24
Ecuación 2 - Función de activación Softmax .....	25
Ecuación 3 - Función de pérdida de la DDQN.....	30
Ecuación 4 - Decaimiento exponencial de $\epsilon$ en la estrategia Epsilon-Greedy para el modelo DDQN.....	48

## **Capítulo 1. INTRODUCCIÓN**

En este trabajo, se llevará a cabo una evaluación exhaustiva de diversas técnicas de Deep Learning (DL) para la detección de anomalías en máquinas rotativas. Se utilizarán datos de sensores de vibración para procesar, diseñar y probar diferentes modelos de Deep Learning.

La utilización de máquinas rotativas está muy extendida en la industria para realizar funciones clave de diversa índole que permiten la transformación de unos tipos de energía en otros. La detección temprana de fallos en máquinas rotativas es esencial para garantizar la operación y el mantenimiento óptimos de las centrales de generación eléctrica y otros sistemas industriales. El procesamiento de datos de sensores de vibración proporciona una fuente de información valiosa para identificar patrones y señales de alarma que indican posibles fallos.

A lo largo de este proyecto, se explorarán diferentes enfoques y técnicas de Deep Learning, como redes neuronales convolucionales (CNN) y aprendizaje por refuerzo profundo (DRL), con el objetivo de desarrollar un modelo de detección de fallos lo más tempranamente posible con precisión y fiabilidad. Se realizarán pruebas y comparaciones exhaustivas para evaluar el rendimiento y la efectividad de cada modelo en la detección y clasificación de anomalías en la maquinaria rotativa.

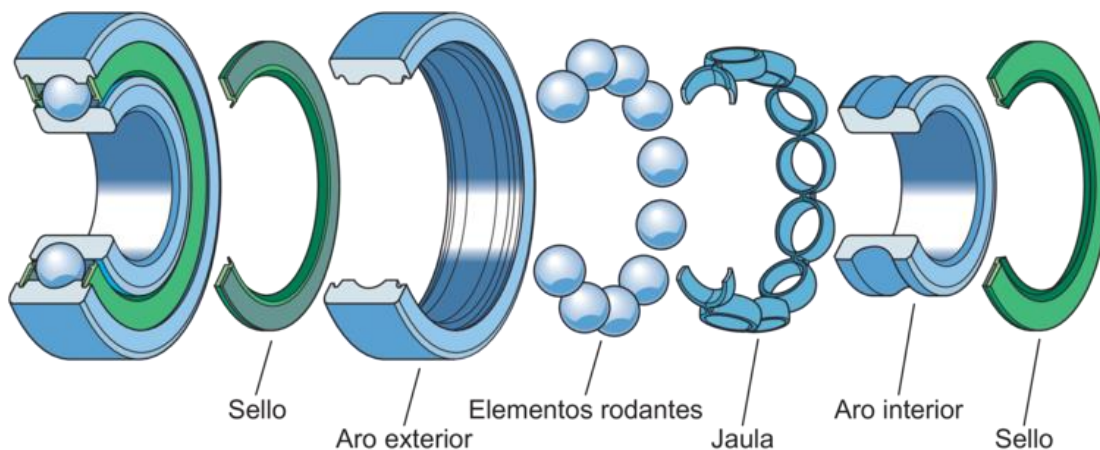
Al finalizar este trabajo, se espera contar con un modelo de Deep Learning que pueda ser implementado en entornos industriales reales, brindando una herramienta de detección de fallos lo más anticipadamente posible, que contribuya a mejorar la seguridad y la confiabilidad de las operaciones industriales.

### ***1.1 MOTIVACIÓN DEL PROYECTO***

Las centrales de generación eléctrica utilizan ampliamente máquinas rotativas en sus operaciones, es decir, dispositivos mecánicos que convierten la energía eléctrica en energía



mecánica y viceversa a través de la rotación de un eje. Estas máquinas incluyen generadores eléctricos, turbinas de gas o vapor y motores compresores, entre otros. Los fallos en estas maquinarias pueden tener un gran impacto en la operación y mantenimiento de las centrales, así como en los sistemas de producción industrial en general, afectando la fiabilidad de la operación, su mantenimiento y el uso de recursos económicos y humanos disponibles. Estudios previos han revelado que, en un entorno industrial, entre el 40% y el 70% de fallos en maquinaria se deben a defectos en los rodamientos [1] [2]. Por esta razón, es esencial mantener una vigilancia continua de la condición de los rodamientos en estas máquinas rotativas para detectar cualquier anomalía lo más pronto posible y diagnosticar sus causas. En la Figura 7 se muestra una caja de rodamientos [3] que consiste en una serie de bolas posicionadas entre un anillo interior y exterior. Cuando se aplica una carga a los anillos, las bolas ruedan sobre las superficies, lo que ayuda a reducir la fricción, permitiendo que la pieza se mueva o rote con mayor suavidad y eficiencia.



*Figura 7 - Componentes de un rodamiento*

Las centrales de generación eléctrica deben cumplir requisitos estrictos en cuanto a la seguridad de su operación, dado que cualquier fallo en una de estas máquinas puede resultar en importantes pérdidas económicas tanto de operación como de mantenimiento, e incluso en accidentes graves. Es por ello que la cantidad de fallos que se producen en este tipo de máquinas es muy reducida. Como resultado, la aplicación de algoritmos de inteligencia

artificial y métodos de Machine Learning (ML) tradicionales se convierte en un desafío particularmente complicado para empresas que desean realizar esta tarea de detección de fallos debido al reducido número de casos de fallo. [4]

El Deep Learning es una rama de la inteligencia artificial que utiliza redes neuronales profundas para procesar y analizar datos, con el objetivo de identificar patrones y características ocultas que no son detectables por métodos convencionales de Machine Learning. Esto lo hace ideal para aplicaciones como el seguimiento continuo y diagnóstico de anomalías en máquinas rotativas, donde los datos pueden ser muy complejos y los patrones no son evidentes a simple vista. Este trabajo pretende estudiar y evaluar la aplicación de varias técnicas de Deep Learning para este propósito usando datos públicos existentes a modo de benchmarking.

## **Capítulo 2. ESTADO DE LA CUESTIÓN**

En el contexto de la monitorización y diagnóstico de fallos en maquinaria rotativa, las técnicas de inteligencia artificial y aprendizaje automático están adquiriendo cada vez más importancia. Estos avances representan un enfoque prometedor para superar las limitaciones de los métodos tradicionales, que requerían el análisis manual de señales por expertos. Al explorar la literatura científica actual, se examinan las contribuciones clave y, a través de esta revisión exhaustiva, se pretende comprender el panorama actual de investigación y destacar las oportunidades y desafíos en este campo en constante evolución.

El artículo [5] realiza un repaso de los distintos estudios que se han realizado hasta el momento sobre técnicas de detección de fallos en maquinaria rotativa. Antes de la aparición del aprendizaje automático, la detección de fallos en maquinaria rotativa se realizaba mediante el análisis manual de señales de vibración por parte de expertos. Estos fallos se manifestaban en forma de vibraciones anormales, con amplitudes que fluctuaban en el dominio del tiempo y picos que destacaban en frecuencias características del fallo. Se propusieron diversos métodos de detección de fallos basados en el análisis de vibraciones con el fin de eliminar el ruido y la interferencia armónica de las señales, mejorar la detección de pulsos relacionados con los fallos y aumentar la eficacia global del proceso de detección. No obstante, estos métodos requerían la observación manual de las señales y el conocimiento experto para interpretar la presencia de un fallo, sin la capacidad de clasificar el tipo específico de fallo que había ocurrido.

Por lo tanto, el artículo destaca la importancia de la detección automática de fallos utilizando modelos de DL. A través del aprendizaje automático, estos modelos pueden aprender características directamente de las señales originales y entrenar clasificadores para lograr un reconocimiento de fallos efectivo sin intervención humana. Esta capacidad de detección automática y clasificación de fallos representa un avance significativo en el campo de la monitorización y diagnóstico de maquinaria rotativa.

Sin embargo, es importante destacar los avances significativos en el análisis de señales aplicado a la detección de fallos en maquinaria rotativa. El artículo [6] proporciona importantes hallazgos al respecto. En este estudio se resalta el uso de técnicas de procesamiento de señales, que se complementan para obtener un mejor rendimiento en el diagnóstico. Estas técnicas permiten extraer y separar las señales moduladoras generadas por los defectos en los rodamientos y su combinación cuidadosa ha demostrado ser beneficiosa para superar los desafíos presentados por las diferentes condiciones a las que están sometidas estas máquinas. Este enfoque en el procesamiento de señales complementa la aplicación de técnicas de aprendizaje profundo (DL) mencionadas anteriormente, permitiendo a las redes neuronales aprender de manera más precisa y eficaz las características de las señales de vibración y lograr así una detección y clasificación de fallos más efectiva. Además, se resalta el uso de métodos ensemble en el ámbito del diagnóstico de máquinas rotativas, los cuales combinan varios modelos de aprendizaje automático para lograr un mejor rendimiento predictivo. Estos avances representan un paso adelante en el campo de la monitorización y diagnóstico automático de maquinaria rotativa, proporcionando nuevas oportunidades para mejorar la eficiencia y la fiabilidad de los sistemas de detección de fallos.

El artículo de la Universidad de Paderborn, [7], tiene como objetivo principal la creación de una base de datos que pueda servir como referencia en investigaciones futuras. En particular, se enfoca en el análisis de fallos en sistemas electromecánicos y motores, donde los rodamientos desempeñan un papel crucial. Para ello, se llevan a cabo experimentos para generar tanto datos artificiales como reales de daños en rodamientos, registrando a través de sensores la vibración y las señales de corriente del motor. Posteriormente, se comparan diferentes modelos de Machine Learning tradicionales, como SVM, Random Forest o k-NN, en su capacidad para detectar estos fallos, obteniendo resultados prometedores para el uso de vibraciones como fuente de datos y cierto potencial para las señales de corriente del motor.

Sin embargo, en el campo de la detección de fallos en rodamientos, el uso de técnicas de ML tradicionales presentan ciertas limitaciones, mientras que el DL ha demostrado ser altamente efectivo. En el [8] se realiza una exhaustiva revisión de estudios previos sobre la detección

de fallos en rodamientos. Resalta la notable mejora lograda mediante el uso de DL, mencionando modelos como Convolutional Neural Networks (CNN) y Recurrent Neural Networks (RNN). Las CNN son una elección popular debido a su capacidad de reducir la complejidad del modelo y detectar patrones de fallos en los rodamientos independientemente de su tamaño, posición o distorsión en la señal de vibración capturada. Por otro lado, las RNN, con arquitecturas como Long Short-Term Memory (LSTM), son capaces de modelar dependencias temporales. Sin embargo, las RNN también presentan desventajas, como problemas frecuentes de aprendizaje debido a la desaparición o explosión del gradiente. Esto significa que la información se puede perder gradualmente o amplificar enormemente a medida que se propaga a través de la red, lo que dificulta que la red aprenda y capture patrones a largo plazo de manera efectiva.

El artículo explica cómo el Deep Learning ofrece características distintivas en comparación con el Machine Learning tradicional. Permite el aprendizaje simultáneo de características y clasificación, eliminando la necesidad de conocimientos expertos previos. La extracción de características se automatiza, lo que resulta en un procesamiento más efectivo. Sin embargo, requiere conjuntos de datos grandes para un rendimiento óptimo, y el costo computacional del entrenamiento puede ser elevado. A pesar de estas desventajas, la precisión diagnóstica es muy alta, lo que la convierte en una poderosa herramienta para la detección y diagnóstico de fallos en maquinaria rotativa.

En los últimos años, la detección de anomalías en maquinaria rotativa ha experimentado avances significativos gracias al uso de técnicas de DL. Sin embargo, en el artículo [4] se busca ir un paso más allá y propone una perspectiva innovadora en forma de Deep Reinforcement Learning (DRL). En particular, propone la vibración como método efectivo para la monitorización del estado de la maquinaria rotativa en centrales nucleares. El artículo compara el rendimiento de modelos de DL como las CNN y las Gated Recurrent Units (GRU), y modelos de DRL que combinan el aprendizaje por refuerzo (RL) con el Deep Learning, como las CNN-RL y GRU-RL, con un modelo de SVM. Los resultados revelan que, aunque los modelos de DRL requieren más tiempo para converger, son más estables y demuestran un mejor desempeño cuando se dispone de pocas muestras de datos. Esta

característica los convierte en una aproximación mejor a escenarios reales donde se tienen pocos datos de fallos en comparación con datos de la máquina funcionando correctamente.

El presente trabajo se basa en un marco conceptual sólido y se inspira en estudios previos sobre técnicas de detección de fallos en maquinaria rotativa. Se ha identificado un estudio relevante [4] que ha servido como punto de referencia para el desarrollo de este trabajo. Si bien se ha tomado como base ese estudio, es importante destacar que este trabajo va más allá al proponer enfoques innovadores y realizar contribuciones originales en el ámbito de la detección de anomalías.

Como conclusión del estado de la cuestión, se puede destacar que el enfoque del aprendizaje automático tradicional muestra limitaciones en la detección de fallos en rodamientos. En este sentido, se observa que las CNN parecen ser una de las opciones más apropiadas en este caso de uso específico. Sin embargo, es importante señalar que existe una falta de investigación en el campo del aprendizaje por refuerzo aplicado a problemas de clasificación de fallos en maquinaria rotativa. A pesar de esto, el potencial del DRL en este contexto resulta prometedor y puede ser objeto de futuras investigaciones y desarrollos. Además, es importante destacar que todos los artículos utilizan datos de laboratorio para entrenar y validar los modelos propuestos. Sin embargo, se enfatiza la importancia de aplicar estos modelos a datos reales de la maquinaria en funcionamiento para evaluar su eficacia en escenarios reales. Esto puede proporcionar información muy útil para la toma de decisiones en la gestión de activos y mantenimiento predictivo en la industria.

## **Capítulo 3. DEFINICIÓN DEL TRABAJO**

### ***3.1 JUSTIFICACIÓN***

A la luz del análisis crítico realizado sobre los trabajos previos en el campo de la detección de fallos en maquinaria rotativa, surge la necesidad de desarrollar un proyecto que supere las limitaciones de los enfoques tradicionales y ofrezca soluciones más efectivas. En este contexto, el proyecto se presenta como una propuesta atractiva que combina métodos de Deep Learning para abordar esta problemática de manera innovadora y con resultados significativamente mejorados.

Una de las principales fortalezas de este proyecto reside en la capacidad de adaptarse a distintas velocidades de rotación. Esta característica es esencial en entornos industriales, donde la maquinaria rotativa puede operar a diferentes velocidades según las necesidades del proceso. Al incorporar esta flexibilidad, garantiza una detección de fallos precisa y confiable en una amplia gama de escenarios industriales reales.

Además, el enfoque combina dos poderosas técnicas de aprendizaje automático: la CNN y el aprendizaje por refuerzo. Esta combinación es una estrategia innovadora y altamente efectiva, ya que permite aprovechar la capacidad de la CNN para extraer características relevantes de las señales de vibración y, al mismo tiempo, utilizar el RL para optimizar el proceso de clasificación y obtener resultados más precisos y confiables. La integración de estas técnicas es lo que distingue este trabajo de otros trabajos en el campo.

En términos de impacto y relevancia, es importante destacar cómo se alinea con los desafíos actuales relacionados con la eficiencia energética y la sostenibilidad en la industria. La detección temprana y precisa de fallos en la maquinaria rotativa a través de Deep Learning puede prevenir costosos tiempos de inactividad y reducir el consumo energético innecesario. Esto se traduce en ahorros significativos para las empresas, así como en una producción más responsable y respetuosa con el medio ambiente.

Además, este trabajo se posiciona como una solución valiosa para las empresas, ya que aborda la necesidad de un monitoreo continuo y fiable de la maquinaria rotativa. Al proporcionar un enfoque automatizado y de alta precisión, permite a las empresas adoptar un enfoque proactivo en el mantenimiento y minimizar los riesgos asociados con fallos.

En resumen, mediante la combinación de métodos de Deep Learning, adaptabilidad a distintas velocidades de rotación y enfoque en la eficiencia energética, el proyecto se posiciona como una solución atractiva y altamente beneficiosa para las empresas, contribuyendo al crecimiento económico, la competitividad industrial y la sostenibilidad ambiental.

### **3.2 OBJETIVOS**

El objetivo principal de este proyecto es desarrollar y evaluar modelos de DL y DRL capaces de detectar anomalías en máquinas rotativas. Para alcanzar este objetivo global, se plantean los siguientes objetivos específicos:

- **Recopilar, preprocesar y utilizar datos de benchmarking:** Para este trabajo se hará uso de un conjunto de datos públicos existente. Esto permitirá comparar el rendimiento de los modelos que se desarrollen con otros en la literatura y establecer una referencia sólida.
- **Desarrollo y entrenamiento de modelos de Deep Learning:** Desarrollar modelos de DL que utilicen arquitecturas como redes neuronales convolucionales para la detección de anomalías en máquinas rotativas.
- **Desarrollo y entrenamiento de modelos de DRL:** Implementar modelos que combinen técnicas de aprendizaje profundo y algoritmos de refuerzo para mejorar la capacidad de detección de anomalías en máquinas rotativas.
- **Evaluación de los modelos diseñados:** Realizar experimentos y evaluaciones exhaustivas para medir la precisión, la sensibilidad y la especificidad de los modelos propuestos en la detección de anomalías en máquinas rotativas.



- **Comparación de resultados con la literatura:** Comparar los resultados obtenidos por los modelos desarrollados con los resultados reportados en los artículos previamente estudiados. Esto permitirá evaluar la eficacia y el avance logrado en este proyecto en relación con el estado del arte.

A través de estos objetivos, se busca contribuir al campo de la detección de anomalías en máquinas rotativas mediante la evaluación de distintas técnicas de Deep Learning, y establecer nuevas perspectivas en este campo de investigación.

### **3.3 METODOLOGÍA**

Para conseguir los objetivos propuestos, en la realización de este proyecto se ha propuesto desarrollar un enfoque ágil, similar a la metodología Agile, en el que se trabaja en ciclos cortos de desarrollo con reuniones semanales con los directores del proyecto para revisar el progreso, recibir feedback y ajustar los planes. Durante estas reuniones, se ha expuesto en común el progreso de la semana anterior, se establecieron objetivos para la próxima semana y se realizaron ajustes en función del feedback recibido. Este enfoque ágil ha permitido adaptarse rápidamente a los cambios en los requisitos del proyecto y mejorar continuamente la calidad del software desarrollado.

En primer lugar, se descargaron los datos que se han utilizado a partir de una base de datos creada por un estudio de la Universidad de Paderborn [7]. Estos datos se sometieron a un proceso de preprocesamiento con el objetivo de obtener una versión limpia y adecuada para su análisis. Durante este proceso, se extrajeron las características relevantes para el caso de estudio, seleccionando las señales de vibración e incluyendo datos de todos los posibles fallos de la maquinaria a diferentes velocidades de rotación para poder diseñar modelos robustos y generalizables.

Una vez se dispuso de los datos procesados, se procedió al desarrollo de modelos de Deep Learning. Se optó por utilizar arquitecturas de CNN para la detección de anomalías en máquinas rotativas. Se llevaron a cabo pruebas con distintas configuraciones y parámetros, con el objetivo de obtener un modelo fiable y preciso en la detección de dichas anomalías.

Además de los modelos de Deep Learning, se investigó la aplicación de técnicas de DRL. Se optó por implementar un modelo basado en Double Deep Q-Network (DDQN) y se realizaron ajustes en los parámetros para mejorar su rendimiento en la detección de anomalías en máquinas rotativas.

Con el fin de obtener un modelo más robusto y fiable, también se creó un tercer modelo que combina las capacidades de los modelos de Deep Learning y Deep Reinforcement Learning desarrollados anteriormente. Esto permitió aprovechar las ventajas de ambas técnicas y mejorar la capacidad general de detección de fallos en maquinaria rotativa.

Finalmente, se realizó un análisis exhaustivo de los resultados obtenidos por los modelos diseñados. Se compararon los diferentes enfoques utilizados y se examinaron las fortalezas y debilidades de cada uno. Se extrajeron conclusiones sobre la eficacia de los modelos propuestos y se discutió su contribución en relación con los objetivos planteados y el estado del arte en el campo de la detección de anomalías en máquinas rotativas.

En resumen, la metodología seguida en este proyecto permitió abordar los objetivos planteados de manera eficiente y efectiva. El enfoque ágil adoptado facilitó la adaptación a los cambios y la mejora continua del trabajo realizado. A través de los diferentes pasos descritos, se logró desarrollar y evaluar modelos de Deep Learning y Deep Reinforcement Learning capaces de detectar anomalías en máquinas rotativas, contribuyendo al avance en este campo de investigación.

### **3.4 HERRAMIENTAS UTILIZADAS**

En el desarrollo de este proyecto se han utilizado diversas herramientas y recursos para el análisis de datos y el diseño de los modelos. En primer lugar, se ha empleado el lenguaje de programación Python, debido a que cuenta con una amplia variedad de librerías para aplicar técnicas de Machine Learning y manipulación de datos. En particular, se ha utilizado TensorFlow, una potente librería que permite construir, entrenar y evaluar redes neuronales de distinta complejidad de manera eficiente y altamente escalable. Para ello, se ha hecho uso

de Keras, una interfaz de alto nivel construida sobre TensorFlow para simplificar la construcción y entrenamiento de modelos de Deep Learning.

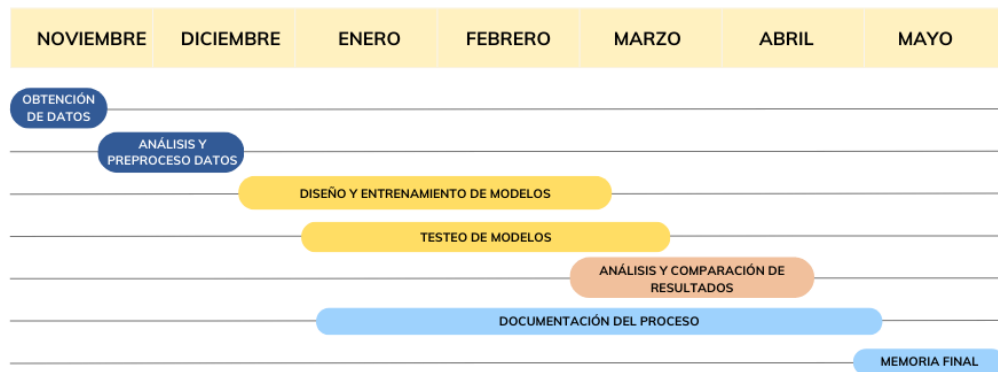
Además, se ha optado por el entorno de desarrollo integrado Visual Studio Code como IDE principal. Esta elección ha permitido escribir y ejecutar los Jupyter Notebooks de forma sencilla y eficiente, brindando un entorno de trabajo cómodo y flexible. La combinación de Python, TensorFlow y Visual Studio Code ha resultado clave para alcanzar los objetivos del proyecto de manera efectiva y eficiente.

Estas herramientas han proporcionado una base sólida para el análisis de datos, la implementación de los modelos y la evaluación de resultados, garantizando un enfoque riguroso y preciso en el desarrollo de este trabajo.

### ***3.5 PLANIFICACIÓN Y ESTIMACIÓN ECONÓMICA***

Con el objetivo de tener una planificación clara y organizada a lo largo del proyecto, se diseñó un diagrama de Gantt que se muestra en la Figura 8. Este diagrama detalla las diferentes fases del trabajo que se han llevado a cabo para alcanzar los objetivos específicos propuestos. El proyecto se dividió en varias etapas, comenzando con la obtención y análisis de los datos, seguido del diseño, entrenamiento y testeo de los modelos, el análisis de los resultados obtenidos y, finalmente, la documentación del proceso en la memoria final.

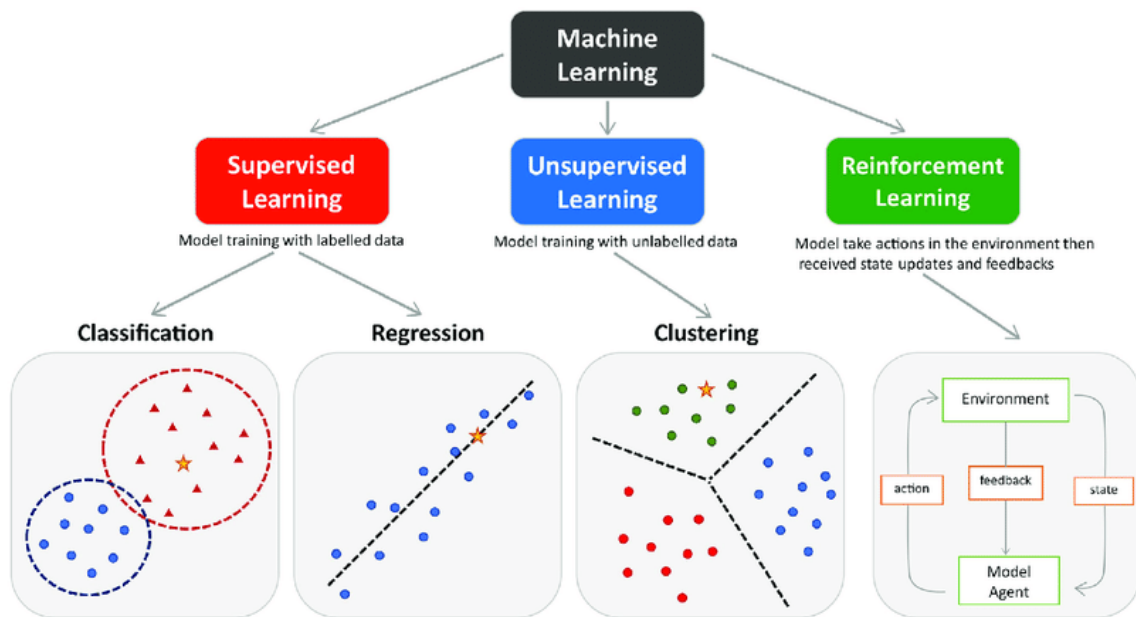
En cuanto a la estimación económica, este proyecto ha sido desarrollado sin incurrir en ningún coste económico significativo, dado que se han utilizado recursos y herramientas de código abierto disponibles de manera gratuita. Además, se ha empleado tiempo personal para llevar a cabo el proyecto, considerándolo como un recurso no remunerado pero valioso en la ejecución del mismo.



*Figura 8 - Diagrama de Gantt de la planificación del trabajo*

## Capítulo 4. MODELOS DE DEEP LEARNING PARA LA DETECCIÓN DE ANOMALÍAS

En este capítulo se llevará a cabo una exposición teórica de los modelos de Deep Learning más relevantes utilizados en la detección de anomalías. Se abordarán tanto modelos de aprendizaje supervisado como de aprendizaje por refuerzo. La Figura 9 detalla los principales tipos de aprendizaje automático [9]: aprendizaje supervisado, aprendizaje no supervisado y aprendizaje por refuerzo. El aprendizaje supervisado se caracteriza por el uso de datos etiquetados, mientras que el aprendizaje no supervisado utiliza datos sin etiquetar, y el aprendizaje por refuerzo implica la interacción de un agente con un entorno.



*Figura 9 - Principales tipos de aprendizaje automático*

Se explorarán en detalle las Convolutional Neural Networks (CNN) y las Double Deep Q-Networks (DDQN), brindando una comprensión profunda de sus principios fundamentales y su aplicación en el campo de estudio. Además, se analizarán críticamente los aspectos

positivos y negativos de cada uno de estos modelos, permitiendo una evaluación objetiva de sus fortalezas y debilidades. Esta revisión teórica sentará las bases necesarias para comprender mejor el enfoque adoptado en este trabajo y permitirá una apreciación más completa de las contribuciones posteriores en el mismo.

## **4.1 REDES NEURONALES CONVOLUCIONALES (CNN)**

Una red neuronal convolucional es un tipo de modelo de Deep Learning diseñado específicamente para procesar datos con una estructura espacial, como por ejemplo imágenes aunque también se pueden aplicar a señales de una dimensión. Su propósito general es extraer características relevantes de los datos de entrada mediante el uso de capas convolucionales y pooling, seguidas de capas completamente conectadas para realizar la clasificación. Estos modelos son ampliamente utilizados en la detección de anomalías debido a su capacidad para aprender representaciones de alto nivel y capturar patrones complejos en los datos.

### **4.1.1 COMPONENTES DE UNA RED NEURONAL CONVOLUCIONAL**

La Figura 10 muestra las distintas etapas que conforman una red neuronal convolucional en un problema de clasificación [10]. En esta arquitectura, las capas de convolución y de pooling desempeñan un papel fundamental en el aprendizaje de características (feature learning) mientras que las capas completamente conectadas se encargan de la etapa de clasificación final.

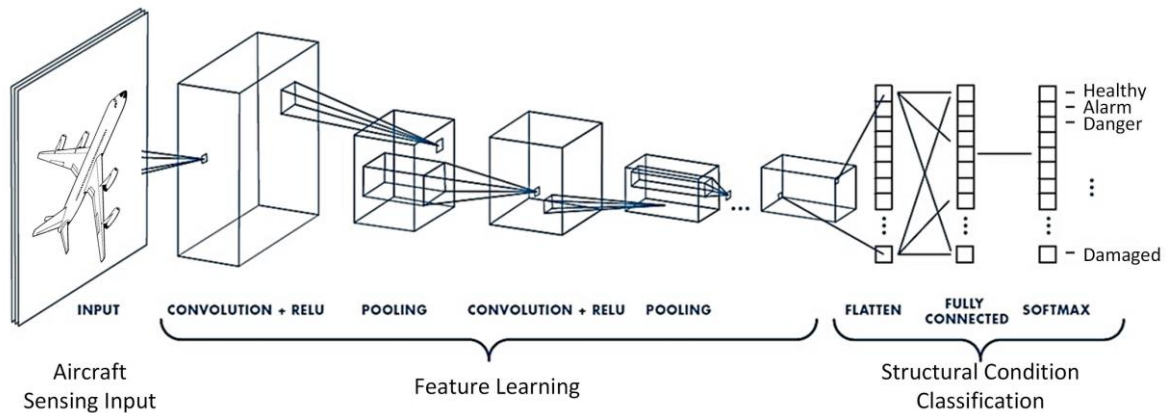


Figura 10 - Arquitectura de una CNN

A continuación, se presentan y describen en detalle los componentes fundamentales de una CNN:

- Capa de convolución:** La capa de convolución es la parte central de una CNN. Utiliza filtros o kernels que se desplazan a través de los datos de entrada para realizar operaciones de convolución [11] como se muestra en la Figura 11. Estas operaciones permiten extraer características locales y aprender representaciones más abstractas de los datos. Cada filtro calcula productos escalares entre los valores del filtro y las ventanas de entrada correspondientes. Esto genera un mapa de características, capturando diferentes aspectos de los datos como podría ser el borde de un objeto en el caso de una imagen, o un pico o patrón inusual en la frecuencia de la señal de vibración en el caso de detección de anomalías en maquinaria. La capa de convolución utiliza múltiples filtros para obtener múltiples mapas de características y aumentar la capacidad de representación de la red.

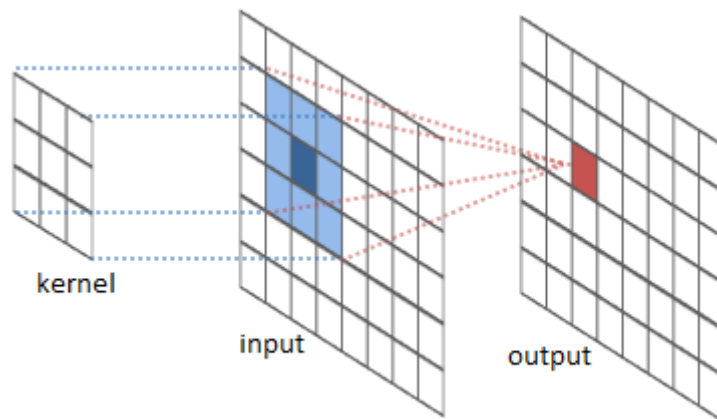


Figura 11 - Convolución de una imagen

- **Capa de agrupación (pooling):** La capa de agrupación, también conocida como capa de pooling, reduce la dimensionalidad de los mapas de características generados por la capa de convolución. El objetivo principal de esta capa es preservar las características más relevantes mientras se reduce la cantidad de parámetros y la complejidad computacional. Los métodos más comunes de agrupación son:
  - Agrupación máxima (max pooling): Selecciona el valor máximo dentro de una región de tamaño predefinido en cada mapa de características.
  - Agrupación promedio (average pooling): Selecciona el valor promedio dentro de una región de tamaño predefinido en cada mapa de características.Esto permite conservar las características dominantes y reducir el tamaño espacial de los datos.
- **Capa completamente conectada:** La capa completamente conectada, también conocida como capa densa (dense), se encuentra al final de la arquitectura de la CNN. Esta capa toma las características extraídas por las capas anteriores y las procesa para realizar la clasificación. En una capa completamente conectada, cada unidad neuronal está conectada a todas las unidades de la capa anterior. Estas conexiones permiten una combinación más compleja y no lineal de las características extraídas. Además, se pueden aplicar funciones de activación para mejorar la capacidad de la red para aprender patrones más complejos y no lineales en los datos.



#### 4.1.2 FUNCIONES DE ACTIVACIÓN

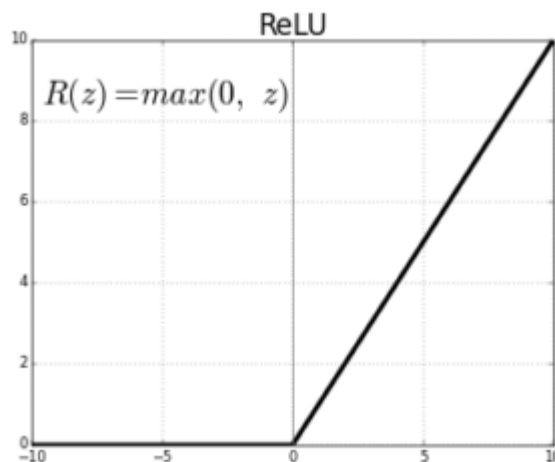
En las redes neuronales, las funciones de activación desempeñan un papel fundamental al introducir no linealidad en el proceso de cálculo e influyen en la propagación de la información a través de la red. Dos funciones de activación comúnmente utilizadas en las redes neuronales convolucionales para la detección de anomalías en maquinaria rotativa son ReLU y Softmax [12].

- ReLU (Rectified Linear Unit): Es una función de activación no lineal ampliamente utilizada. Su fórmula es simple, devuelve el valor de entrada si es mayor que cero y cero en caso contrario. Matemáticamente, se puede representar de la siguiente manera:

$$ReLU(z) = \max(0, z)$$

*Ecuación 1 - Función de activación ReLU*

La función ReLU introduce no linealidad en la red y se aplica a la salida de las capas convolucionales y en las capas densas. Su principal ventaja reside en su eficiencia computacional, ya que es computacionalmente más barata de calcular en comparación con otras funciones de activación.



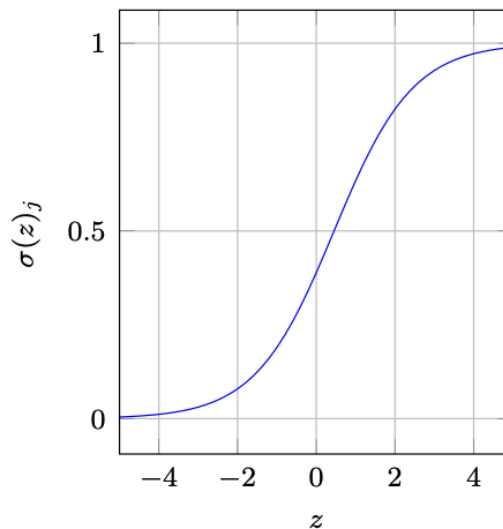
*Figura 12 - Función de activación ReLU*

- Softmax: Esta función se utiliza en la capa de salida para generar una distribución de probabilidades sobre las clases en problemas de clasificación. Permite obtener la

clase con mayor probabilidad de pertenencia. Su ecuación devuelve la probabilidad de que la muestra pertenezca a la clase  $i$  dado un vector de entrada  $x$ , donde  $z_i$  es la puntuación asociada a la clase  $i$  y la sumatoria se realiza sobre todas las clases posibles:

$$\text{Softmax}(z_j) = \frac{e^{z_j}}{\sum_{k=1}^n e^{z_k}}$$

*Ecuación 2 - Función de activación Softmax*



*Figura 13 - Función de activación Softmax*

### **4.1.3 APRENDIZAJE EN REDES NEURONALES**

Las redes neuronales son un tipo de algoritmo de aprendizaje supervisado. Esto significa que utilizan datos etiquetados para entrenar y predecir. Los datos de entrenamiento contienen ejemplos con entradas y salidas deseadas, permitiendo que la red aprenda a realizar predicciones precisas. Después del entrenamiento, la red puede predecir resultados en datos no vistos previamente.

Las redes neuronales aprenden durante su entrenamiento según el algoritmo de backpropagation. Durante este proceso, la red neuronal recibe datos de entrada y realiza cálculos para generar una salida al pasar por las distintas etapas como se ha explicado

anteriormente. Luego, se compara la salida con las respuestas correctas y se calcula el error según la función de pérdida escogida. En el contexto de la detección de anomalías, se utiliza la función de pérdida categorical crossentropy para problemas de clasificación. Esta función mide la discrepancia entre la distribución de probabilidad predicha por la red neuronal y la distribución de probabilidad real de las clases. El objetivo es minimizar esta diferencia durante el entrenamiento.

El backpropagation consiste en propagar este error hacia atrás a través de la red, ajustando los pesos en dirección opuesta al gradiente del error. Esto se logra mediante el descenso de gradiente. El gradiente del error indica la dirección en la que hay que modificar para disminuir el error de la red neuronal. En el descenso de gradiente, se realiza un ajuste en los pesos de la red en dirección opuesta al gradiente, es decir, se actualizan los pesos en la dirección que minimiza el error.

La actualización de los pesos utiliza una tasa de aprendizaje que determina cómo de grandes son los pasos del ajuste. Si la tasa de aprendizaje es muy pequeña, el proceso de entrenamiento puede volverse muy lento. Sin embargo, una tasa de aprendizaje muy grande puede hacer que se salte el mínimo óptimo de la función de error. Por lo tanto, es crucial encontrar una tasa de aprendizaje adecuada que permita un entrenamiento eficiente y una convergencia óptima de la red neuronal.

Existen diversos optimizadores diseñados para actualizar los pesos de manera eficiente y mejorar el rendimiento del aprendizaje de la red. ADAM es un optimizador muy utilizado en el entrenamiento de redes neuronales debido a sus mejoras respecto al descenso de gradiente normal. Incorpora dos puntos clave de mejora: el uso de momento y la adaptación de la tasa de aprendizaje. El momento acelera la convergencia considerando la dirección y velocidad de los ajustes anteriores, mientras que la adaptación de la tasa de aprendizaje ajusta automáticamente la velocidad del entrenamiento. De esta manera, se consigue mejorar la eficiencia y efectividad del aprendizaje de la red neuronal.

## **4.2 APRENDIZAJE POR REFUERZO Y DOBLE DQN**

El aprendizaje por refuerzo (RL) es una rama del aprendizaje automático que se basa en que un agente aprenda a través de la interacción con un entorno, tomando acciones y recibiendo recompensas. Este enfoque es típicamente usado en el desarrollo de videojuegos, pero también se está extendiendo a problemas de clasificación, como el desarrollo de modelos para la detección de anomalías en maquinaria. A lo largo de este capítulo, se analizarán en detalle las diferentes componentes del aprendizaje por refuerzo y se estudiará la implementación y aplicaciones del algoritmo Doble DQN.

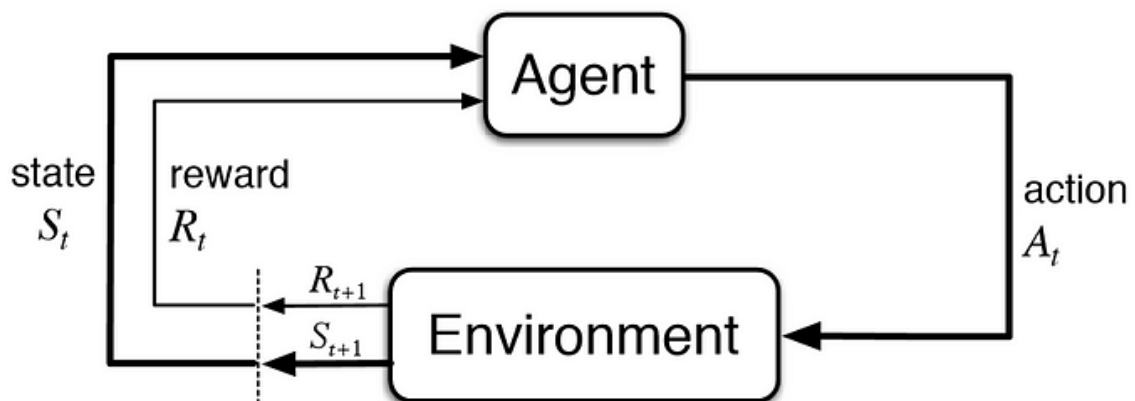
### **4.2.1 COMPONENTES DEL APRENDIZAJE POR REFUERZO**

El aprendizaje por refuerzo se basa en la interacción entre un agente y su entorno. En la Figura 14 se muestra un esquema de los componentes principales del aprendizaje por refuerzo [13]. Estos componentes son fundamentales para comprender cómo el agente se relaciona con su entorno, desarrolla estrategias y optimiza su desempeño a través de la retroalimentación recibida. En este apartado, se analizará en detalle cada uno de estos componentes:

- **Entorno:** Es el espacio en el que el agente interactúa y del cual recibe observaciones y recompensas.
- **Agente:** El agente representa al modelo de RL que toma acciones en el entorno basándose en su estado actual y en las recompensas recibidas. Utiliza algoritmos y estrategias para aprender a tomar decisiones óptimas y maximizar las recompensas a largo plazo.
- **Estado/observación:** Es una representación del entorno en un momento concreto. El estado contiene información relevante sobre el entorno que el agente utiliza para tomar decisiones. En el caso de detección de anomalías estos estados pueden incluir variables sensoriales como la vibración o la corriente del motor, la velocidad de rotación de la máquina, la fuerza radial u otras características que describan la situación actual del entorno.

- **Acción:** En cada paso del tiempo el agente toma una acción como respuesta al estado en el que se encuentra el entorno. Se selecciona una acción de un conjunto de posibles acciones disponibles en el entorno, en base a su política de toma de decisiones.
- **Recompensa:** Es una señal que el entorno proporciona al agente para evaluar la calidad de sus acciones. La recompensa puede ser positiva, negativa o neutra, y ayuda al agente a determinar qué acciones debe tomar.

Estos componentes trabajan conjuntamente para permitir que el agente aprenda a tomar decisiones óptimas y se adapte a diferentes entornos y tareas.



*Figura 14 - Aprendizaje por refuerzo: interacción entre agente y entorno*

#### 4.2.2 REDES PRINCIPAL Y OBJETIVO EN LA DOBLE DQN

Para comprender el algoritmo DDQN es fundamental tener claros algunos conceptos básicos del RL como son el Q-learning y las DQN.

El Q-learning es un algoritmo de RL muy conocido que se basa en aprender una función, conocida como Q-valor. Esta función evalúa la calidad de tomar una acción según un estado asignando un valor a cada combinación de estado y acción. De esta manera, es capaz de representar la recompensa agregada que se espera que el agente pueda obtener al tomar una acción concreta según el estado del entorno. El objetivo del Q-learning es determinar la función Q-valor óptima, es decir, que maximice la recompensa acumulada a largo plazo. Las

DQN (Deep Q-Networks) son una ampliación del Q-learning que utiliza redes neuronales profundas.

La Doble DQN es una variante de la DQN para mejorar la estabilidad y la convergencia del aprendizaje. Utiliza dos redes neuronales: la red principal (o red de política) y la red objetivo. La red principal es la red neuronal que se entrena y actualiza en cada paso del tiempo y se encarga de seleccionar la acción del agente. Por otra parte, la red objetivo no se actualiza en cada paso de tiempo, si no que se mantiene fija durante un tiempo determinado y se utiliza en la evaluación del valor Q de esa acción.

La razón de tener dos redes es evitar que el agente se quede atascado en un bucle infinito de acciones que no sean óptimas. Tener la red objetivo ayuda a guiar al agente hacia acciones que pueden ser mejores en el futuro, ya que se evalúan con el estado siguiente. Además, al actualizarse los pesos de la red objetivo con menos frecuencia, mejora la estabilidad del aprendizaje. En resumen, el uso de dos redes proporciona una perspectiva más objetiva y estable que ayuda al agente a explorar y aprender de manera más efectiva en entornos complejos de aprendizaje por refuerzo.

### **4.2.3 ESTRATEGIAS DE EXPLORACIÓN Y EXPLOTACIÓN EN LA DOBLE DQN**

La exploración y la explotación son dos estrategias fundamentales en el aprendizaje por refuerzo que buscan encontrar un equilibrio entre descubrir nuevas acciones y aprovechar las acciones conocidas. La exploración consiste en que el agente elija acciones aleatoriamente con el fin de encontrar nuevas acciones óptimas que no se hayan probado antes. En cambio, la explotación se refiere a aprovechar el conocimiento adquirido hasta ese momento, seleccionando la acción que la red principal determina como óptima. Un desafío del RL es encontrar el balance adecuado entre exploración y explotación.

Para ello, existen múltiples estrategias como el Epsilon-Greedy. En este método, el agente escogerá una acción al azar con una probabilidad  $\epsilon$  y una acción dada por la red principal con una probabilidad  $1 - \epsilon$ . El valor de  $\epsilon$  se define según el balance que se quiera entre exploración y explotación. Además, se puede implementar una

estrategia de decaimiento de  $\epsilon$  (epsilon decay) que reduce el valor de  $\epsilon$  gradualmente a medida que la red neuronal aprende. Esto implica que al inicio del entrenamiento haya más exploración, pero a medida que la red va adquiriendo conocimientos se favorezca la explotación ya que se espera que la red haya aprendido una política óptima.

En resumen, en el contexto de la Doble DQN, se utilizan estrategias como el Epsilon-Greedy y el decaimiento de  $\epsilon$  para encontrar un equilibrio entre exploración y explotación, mejorando así el rendimiento del agente en el aprendizaje.

#### 4.2.4 ACTUALIZACIÓN DE LOS Q-VALORES EN LA DOBLE DQN

Al igual que con las redes neuronales, el aprendizaje en la Doble DQN implica la actualización de los pesos de la red mediante el método de backpropagation. En la sección anterior, se han explorado en detalle los conceptos de backpropagation y optimización en redes neuronales. En el contexto de la Doble DQN, la red neuronal también pasa por un proceso de aprendizaje mediante el cual se ajusta para mejorar la aproximación de los Q-valores. Este proceso de aprendizaje se basa en una función de pérdida específica para la DDQN que se detalla a continuación:

$$L(\theta) = MSE = \frac{1}{N} \sum (y_{true} - y_{pred})^2$$
$$y_{pred} = Q_{policy}(s_t, a_t)$$
$$y_{true} = r_t + \gamma * \max_{a_{t+1}} Q_{target}(s_{t+1}, a_{t+1})$$

*Ecuación 3 - Función de pérdida de la DDQN*

En la Ecuación 3, la  $y_{pred}$  es el Q-valor predicho por la red principal para el estado y acción actual. Por otro lado, la  $y_{true}$  se calcula utilizando la red objetivo y se define como la suma de la recompensa actual y el factor de descuento ( $\gamma$ ) multiplicado por el máximo Q-valor para la acción siguiente utilizando la red objetivo. Finalmente, se calcula el error cuadrático medio entre ambos valores para tener la función de pérdida de la DDQN.

Por lo tanto, la red principal se utiliza para seleccionar las acciones, mientras que la red objetivo se utiliza para evaluar los Q-valores de esas acciones. La red objetivo se actualiza

más lentamente y se mantiene fija durante cierto período de tiempo antes de ser actualizada. Al utilizar una red objetivo separada, la DDQN reduce la sobreestimación de los Q-valores. Esto ocurre en el Q-learning tradicional donde se puede pensar que las acciones son mejores de lo que realmente son, lo cual lleva a decisiones subóptimas. En la DDQN, la red principal no influye directamente en la estimación por lo que ayuda a obtener una estimación más precisa y equilibrada de los Q-valores.

El factor de descuento en la ecuación representa la importancia de las recompensas futuras en comparación con las recompensas actuales. Un valor de  $\gamma$  cercano a 1 significa que se les da más importancia a las recompensas a largo plazo, mientras que un valor más cercano a 0 enfatiza más las recompensas inmediatas.

Para concluir, la DDQN aprovecha una red objetivo y una función de pérdida especializada para actualizar los Q-valores de manera más precisa y evitar la sobreestimación. Esto promueve la toma de mejores decisiones por parte del agente y, por lo tanto, mejora su desempeño en el aprendizaje por refuerzo.

#### **4.2.5 REPLAY MEMORY EN LA DOBLE DQN**

En esta sección, se explorará el papel de la replay memory en el entrenamiento de la DDQN, pero antes es importante comprender los dos bucles que conforman el proceso de entrenamiento.

En el ámbito de los videojuegos, el primer bucle se refiere a las rondas dentro de una partida individual del juego, donde el agente interactúa con el entorno y toma decisiones en cada paso del tiempo. Este bucle a nivel de episodio permite al agente acumular experiencia y aprender de sus interacciones con el entorno. El segundo bucle, el bucle de iteración global, representaría cada partida del juego que se realiza. Durante este bucle, la actualización de pesos de la red neuronal puede ocurrir múltiples veces dentro de una misma partida, permitiendo una adaptación más continua del agente, o al final de cada partida, consolidando los aprendizajes obtenidos y teniendo una mayor estabilidad.



La replay memory es un mecanismo que almacena y gestiona las transiciones de estado-acción-recompensa-estado a lo largo de múltiples episodios. Esta experiencia acumulada se utiliza para actualizar los pesos de la red neuronal y mejorar el rendimiento del agente.

Según los requerimientos específicos, se puede reiniciar la memory replay al final de cada partida o mantenerla a lo largo de múltiples partidas. Reiniciarla al final de cada partida permite que el agente se base en datos nuevos y relevantes para cada nuevo juego, evitando que la experiencia acumulada tenga un impacto excesivo en el entrenamiento actual. Sin embargo, también se puede optar por mantener la replay memory intacta a lo largo de múltiples partidas para aprovechar la experiencia acumulada y permitir un aprendizaje continuo.

En conclusión, los bucles de entrenamiento y la replay memory desempeñan un papel crucial en el proceso de entrenamiento de la DDQN. Los bucles de episodios permiten al agente acumular experiencia en el entorno, mientras que el bucle global, con la ayuda de la replay memory, utiliza esa experiencia para actualizar los pesos de la red neuronal y mejorar el rendimiento del agente.

### **4.3 RESUMEN DEL CAPÍTULO**

En el campo de la detección de anomalías en maquinaria rotativa, tanto las CNN como las DDQN ofrecen ventajas y limitaciones distintas.

Las CNN de una dimensión destacan por su capacidad para procesar señales y extraer características relevantes complejas, lo que las hace efectivas en la clasificación y reconocimiento de patrones en datos secuenciales. Esto les permite identificar diferentes tipos de fallos y determinar el estado saludable de la maquinaria. Sin embargo, las CNN requieren un volumen considerable de datos etiquetados para un entrenamiento efectivo y pueden sufrir de sobreajuste en situaciones de datos limitados.

Por otro lado, las DDQN presentan la ventaja de aprender directamente de la interacción con el entorno, lo que resulta beneficioso cuando se dispone de pocos datos etiquetados. Su

capacidad para adaptarse a entornos dinámicos y secuenciales les permite ajustarse a cambios en el comportamiento de la maquinaria rotativa. Además, las DDQN pueden aprovechar la experiencia acumulada a lo largo del tiempo, mejorando el rendimiento en casos donde los datos históricos son relevantes para la detección de anomalías. No obstante, las DDQN necesitan un tiempo de entrenamiento prolongado y una configuración precisa de los hiperparámetros, además de presentar posibles desafíos en la convergencia debido a la naturaleza del aprendizaje por refuerzo.

En conclusión, tanto las CNN como las DDQN presentan fortalezas y limitaciones en el contexto de la detección de anomalías en maquinaria rotativa. En el siguiente capítulo, se explorará un caso de aplicación específico donde se analizará cómo estas técnicas se adaptan a los desafíos de detección de anomalías.

## Capítulo 5. CASO DE APLICACIÓN

En este capítulo, se presenta en detalle el caso de aplicación desarrollado, que tiene como objetivo la detección de anomalías en maquinaria rotativa utilizando técnicas de Deep Learning. El presente estudio aborda el procesamiento de datos, el desarrollo de modelos y el análisis de resultados obtenidos. Para visualizar de manera concisa el flujo de trabajo seguido en el proyecto, se ha incluido la Figura 15, que muestra el diagrama de flujo desde la adquisición de los datos en bruto hasta el análisis de los resultados. A través de esta figura, se pueden observar de manera visual las etapas clave del proceso y su interconexión.

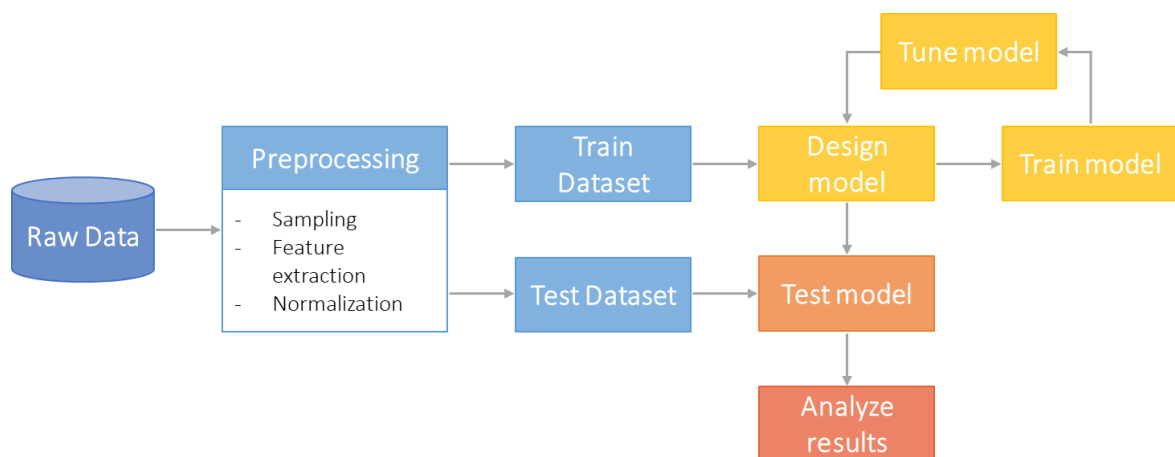
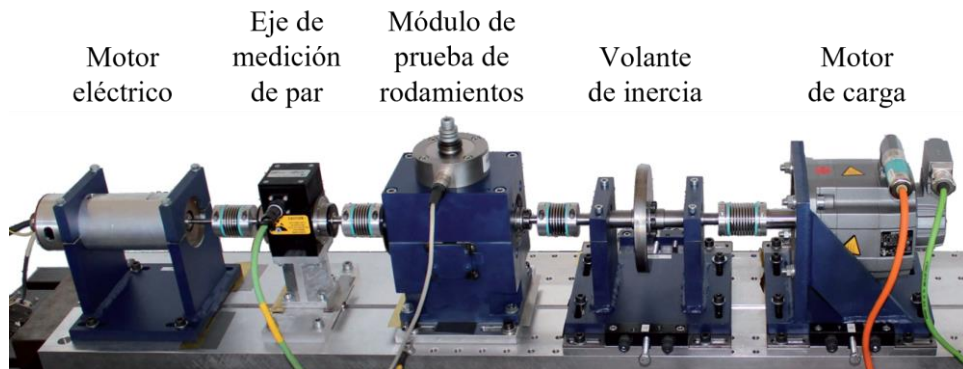


Figura 15 - Diagrama de flujo del desarrollo del proyecto

### 5.1 DESCRIPCIÓN Y PROCESAMIENTO DE DATOS

En este proyecto, se ha utilizado un conjunto de datos público con el propósito de entrenar y validar los modelos de DL para la detección de anomalías en maquinaria rotativa. Esto ha permitido comparar los resultados obtenidos con estudios similares. Los datos utilizados en este trabajo provienen de un estudio realizado en la Universidad de Paderborn [7]. El experimento consistió en usar motores eléctricos para medir dos señales principales: la vibración y la corriente de motor a una frecuencia de 64kHz. Además, se midieron otras

variables como la velocidad, el torque, la temperatura y la carga radial. En la Figura 16 se muestra el sistema de pruebas modular usado donde el elemento central es el módulo de prueba de rodamientos.



*Figura 16 - Sistema de pruebas modular del Paderborn dataset [7]*

El objetivo fue usar los datos para detectar fallos en la maquinaria debido a daños en el rodamiento por lo que se realizaron 32 experimentos, cada uno atribuyéndose a una de las 3 clases del estado de la maquinaria: sana (Healthy), dañada en el anillo interior (IR) o dañada en el anillo exterior (OR). Algunos de estos daños se provocaron de manera artificial y otros son reales. La Tabla 5 muestra los tres distintos métodos que se usaron para provocar los daños artificiales: mecanizado por descarga eléctrica (EDM), taladrado y grabado eléctrico manual. Se puede apreciar en las imágenes cómo los dos primeros métodos provocan daños muy precisos por lo que son fácilmente reproducibles, pero se acercan menos a la realidad. En cambio, el tercer tipo de daño se asemeja mucho a un daño provocado por el funcionamiento real de la máquina. Los fallos reales se formaron mediante pruebas de vida útil acelerada.

Tabla 5 - Daños artificiales en el rodamiento

		
Sharp trench by EDM - KA01	Drilling - KA09	Artificial pitting by electric engraver - KA03

Además, cada experimento se realizó bajo 4 configuraciones en las que cambiaban distintos parámetros como se muestra en la Tabla 6 y se tomaron 20 muestras de 4 segundos para cada uno. Por lo tanto, la base de datos está compuesta por 2560 ficheros '.mat', cada uno con 4 segundos de datos de las señales.

Tabla 6 - Parámetros de operación bajo distintas configuraciones

Velocidad de rotación [rpm]	Par de carga [Nm]	Fuerza radial [N]	Nombre de la configuración
1500	0.7	1000	N15_M07_F10
900	0.7	1000	N09_M07_F10
1500	0.1	1000	N15_M01_F10
1500	0.7	400	N15_M07_F04

Para la realización de este trabajo se ha tomado la decisión de utilizar datos de todos los experimentos con el objetivo de diseñar un modelo que pueda aplicarse de manera generalizada a cualquier tipo de fallo. Sin embargo, se opta por no usar todas las mediciones ya que esto supondría una carga computacional excesiva. Por lo tanto, se toma una muestra representativa del conjunto de datos, garantizando al mismo tiempo el equilibrio de las clases de fallo.

Por otro lado, se emplean los datos de la señal de vibración ya que se espera que estos proporcionen un mayor rendimiento en los modelos. Se toman muestras de 1024 puntos que se transforman al espectro de la frecuencia usando la transformada rápida de Fourier (Fast Fourier Transform, FFT) para ser utilizados como entrada en los modelos. Esto se debe a que, como se puede observar en la Figura 17, las características que diferencian las distintas clases se ven con mayor claridad en el dominio de la frecuencia. La señal en el dominio del tiempo puede parecer ruido y resulta complicado extraer información de manera visual, mientras que en el dominio de la frecuencia se pueden identificar ciertos patrones.

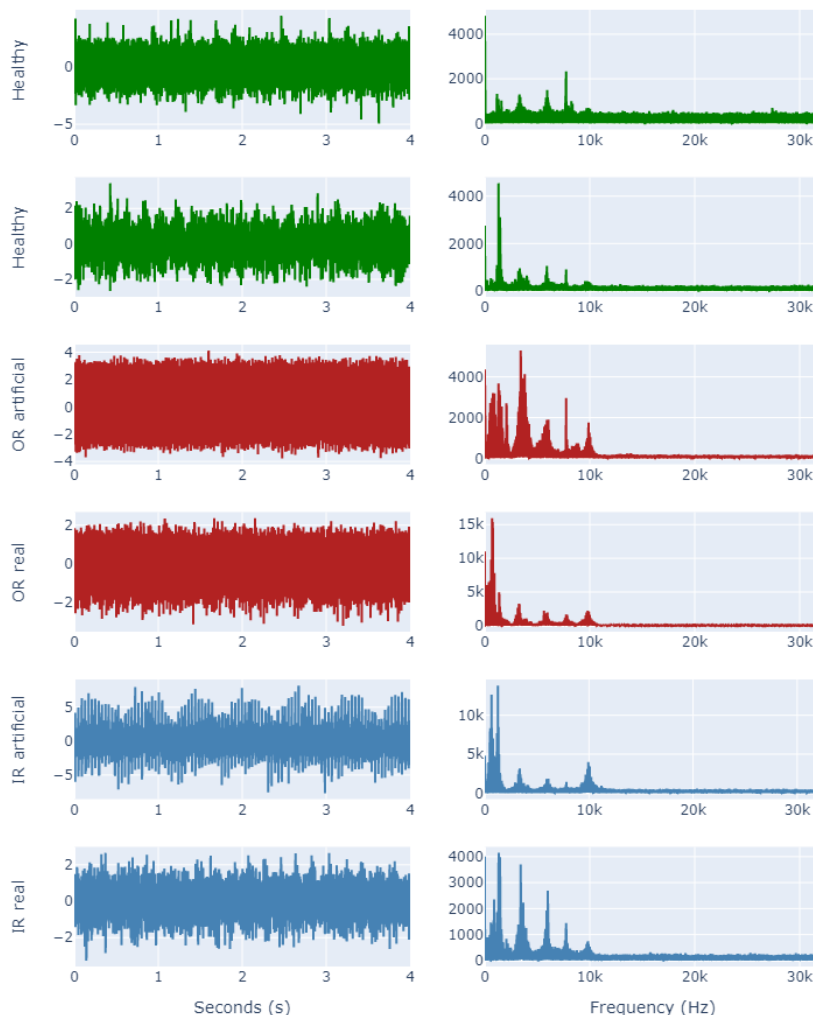


Figura 17 - Visualización de las señales en el dominio del tiempo y de la frecuencia

Además de utilizar los datos de la señal de vibración, también se ha añadido la velocidad de rotación como dato de entrada en los modelos. La velocidad de rotación es una variable que influye en las frecuencias de la señal así que incluirla permite diseñar un modelo capaz de funcionar bajo diferentes configuraciones.

Inicialmente, se realizó una prueba utilizando datos de una única configuración, y al evaluar el rendimiento del modelo, se observó que éste era muy bueno para dicha configuración, así como para otras dos configuraciones adicionales (aquellas en las que se modificaban la fuerza radial y el par de carga). Parecía que estos parámetros no afectaban significativamente el rendimiento del modelo. Sin embargo, al probar el modelo con datos correspondientes a la configuración que cambia la velocidad de rotación, se observó una disminución significativa en el rendimiento del modelo.

Por lo tanto, esta observación respalda el uso de la variable de velocidad de rotación como entrada en los modelos. Al hacerlo, se logrará diseñar un modelo que pueda aplicarse a cualquier configuración, lo cual es especialmente relevante en entornos reales, como una central eléctrica, donde es común contar con maquinarias que operan a distintas velocidades.

Por último, para terminar la fase del preproceso de los datos, se realiza una separación en conjuntos de 70% entrenamiento, 15% validación y 15% prueba. Posteriormente, se normaliza tanto la señal de vibración como la variable de velocidad de rotación. De esta manera, se obtiene una base de datos procesada que ya puede ser utilizada para el entrenamiento y la evaluación de los modelos de detección de anomalías.

## **5.2 *MODELOS DESARROLLADOS***

En esta sección, se describen en detalle los modelos de Deep Learning desarrollados para la detección de anomalías en maquinaria rotativa. Estos modelos han sido diseñados con el fin de aprovechar las capacidades del aprendizaje profundo y mejorar la capacidad de detección de fallos en este tipo de sistemas.

En el capítulo anterior, se ha proporcionado una explicación teórica de los modelos CNN y DDQN, y se ha realizado una breve análisis de sus fortalezas y limitaciones. La elección de las CNN se basa en su capacidad para procesar señales y extraer características relevantes en datos secuenciales. En comparación con otros modelos de Deep Learning, como las redes neuronales recurrentes (RNN), las cuales son más adecuadas cuando se requiere considerar el orden y relación temporal de los datos, en el caso de la detección de anomalías en maquinaria rotativa, el enfoque se centra principalmente en la extracción de características relevantes de la señal, en lugar de capturar relaciones temporales complejas.

En cuanto a la elección de usar técnicas de aprendizaje por refuerzo, el principal beneficio reside en su capacidad para aprender directamente de la interacción con el entorno. Además, se decide utilizar el algoritmo DDQN ya que estos modelos son capaces de adaptarse a entornos dinámicos y secuenciales, así como de aprovechar la experiencia acumulada a lo largo del tiempo, lo cual resulta especialmente relevante en el contexto de la detección de anomalías en maquinaria rotativa.

A lo largo de esta sección, se discutirá la arquitectura utilizada en cada modelo, así como los procedimientos de entrenamiento implementados. El objetivo principal es proporcionar una comprensión completa de los modelos desarrollados y su potencial para la detección precisa de anomalías en maquinaria rotativa.

### **5.2.1 MODELO DE REDES NEURONALES CONVOLUCIONALES**

El primer modelo diseñado es una red neuronal convolucional ya que éstas son capaces de procesar datos secuenciales de manera efectiva aprendiendo características complejas. Durante el proceso de diseño y entrenamiento del modelo para la detección de anomalías en maquinaria rotativa, se llevaron a cabo diferentes pruebas y se exploraron diversas arquitecturas de redes neuronales convolucionales (CNN). Estas pruebas incluyeron variaciones en los parámetros, añadir y modificar capas, así como la consideración de distintas técnicas de mejora del rendimiento. En esta sección, se presentará detalladamente el proceso seguido. Además, se explicará en profundidad la arquitectura final seleccionada y justificada en base a los resultados obtenidos.



En un inicio, se utilizaron datos de entrada de una sola configuración, en concreto la ‘N15\_M07\_F10’ cuyas características se pueden ver en la Tabla 6, leyendo datos de múltiples experimentos y procesando las señales de vibración como se ha detallado anteriormente. En este primer diseño se construyó una red convolucional con una sola capa de convolución 1D que constaba de 4 filtros de tamaño 3. A continuación, se aplicó una capa de max pooling, seguida de dos capas completamente conectadas. La primera capa tenía una función de activación ReLU y un tamaño de 100, mientras que la segunda utilizaba una función de activación softmax y tenía un tamaño de 3, representando así la distribución de probabilidad de pertenecer a cada clase de fallo. Esta red se entrenó durante 10 episodios logrando un accuracy del 99.26%. Este resultado es altamente satisfactorio ya que el modelo comete muy pocos errores de clasificación y es efectivo en la detección de fallos en las máquinas. Sin embargo, se observó que el tiempo de entrenamiento era de 2 minutos y 40 segundos, por lo que se quiso explorar el efecto de reducir el tamaño de los datos de entrada en los resultados y la eficiencia del modelo.

Por lo tanto, se probó el mismo modelo reduciendo los datos de entrada en un tercio. Esto implicó utilizar datos de menos experimentos y, como consecuencia, incluir menos tipos de fallos. Se observó que el tiempo de entrenamiento se redujo significativamente a 34 segundos, sin afectar en gran medida al rendimiento de la red ya que el accuracy con los datos de test fue 99.24%, valor muy similar al anterior. Además, al analizar la matriz de confusión se apreció que la red solo tenía dificultades para distinguir entre los daños en el anillo interior y máquinas saludables, es decir que todos los daños en el anillo exterior se clasificaban correctamente. Esto era comprensible, dado que los fallos en el anillo interior resultan más complicados de detectar.

A raíz de estos resultados, se decidió probar a reducir el conjunto de datos aún más para evaluar el rendimiento del modelo. En esta ocasión, se utilizaron datos de solo dos experimentos por clase (en el caso de los fallos, uno artificial y otro real), reduciendo los datos a una vigésima parte del conjunto inicial. Al igual que antes, esto significaba perder información de fallos provocados por distintos métodos. La primera observación que se tuvo fue que la red convolucional con una capa no era capaz de detectar patrones para clasificar

correctamente los datos de tamaño reducido. Como solución, se añadió una capa adicional de convolución con 8 filtros de tamaño 3 a la red anterior. Con esta arquitectura se redujo el tiempo de entrenamiento a 8 segundos y se obtuvo un accuracy del 97.85%, valor inferior a los anteriores, pero aun considerado satisfactorio. Al igual que en la segunda prueba, solo se encontraron fallos de clasificación entre la clase de fallo en el anillo interior y la clase saludable. Además, se comprobó que el 80% de estos errores se atribuían a fallos provocados artificialmente. Esto es comprensible, ya que los fallos generados de forma artificial pueden tener frecuencias que difieren de las de los fallos reales, dificultando así su distinción.

Al haber reducido el tamaño del conjunto de datos usados para el entrenamiento, se decidió realizar una prueba para evaluar el último modelo propuesto con datos de otros experimentos para comprobar si era generalizable. Especialmente, se evaluó la red con datos de experimentos con otras configuraciones de la Tabla 6. Al realizar estas inferencias, se observó que las variables de fuerza radial y par de carga no alteraban los resultados de manera significativa ya que el accuracy obtenido fue de 96.27% y 97.47% respectivamente. Sin embargo, al realizar la clasificación con datos de experimentos con otra velocidad de rotación se observó que el accuracy disminuyó a un 62.25%.

Estas pruebas y sus resultados demostraron la necesidad de incluir la velocidad de rotación como dato de entrada ya que demuestra ser una variable relevante para la correcta detección de fallos en maquinaria rotativa. Además, se comprobó cómo el rendimiento de la red se puede ver afectado al añadir más capas de convolución.

Por lo tanto, los datos de entrada a la arquitectura final son la señal de vibración en el espectro de la frecuencia y la velocidad de rotación de la maquinaria en revoluciones por minuto. Además, en el modelo final se incluyeron datos de todos los experimentos para tener una aproximación más robusta al problema de detección de anomalías en maquinaria. La Figura 18 muestra un esquema de la arquitectura final de la red diseñada y, a continuación, se entrará en detalle en los aspectos más relevantes de la estructura.

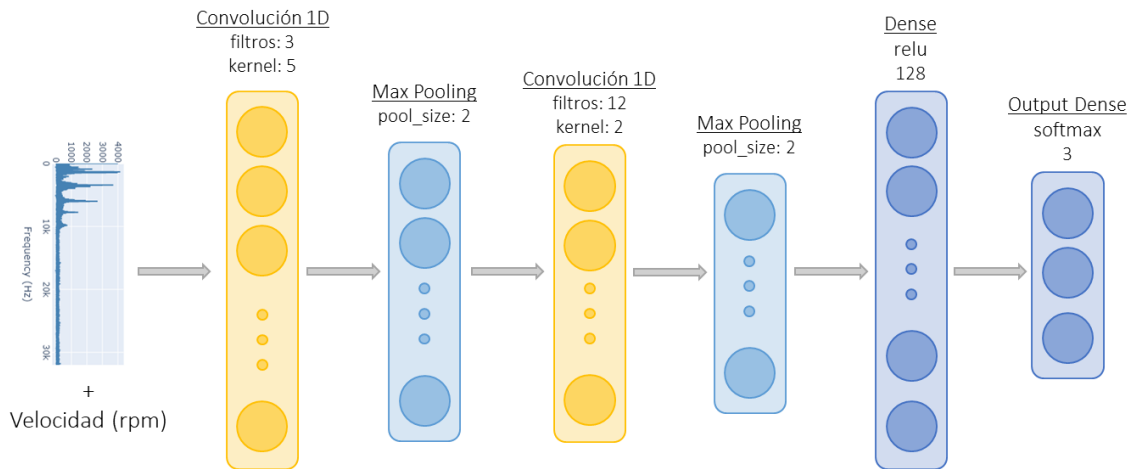


Figura 18 - Esquema del modelo CNN

La red contiene dos capas de convolución en las cuales, a medida que se avanza en la red, el número de filtros aumenta y el tamaño de la kernel disminuye. La idea detrás de esta estrategia es que la capa inicial de convolución capture características más generales y de alto nivel, como tendencias, patrones básicos o cambios globales en la vibración. Al tener menos filtros y una mayor ventana de convolución, esta capa puede enfocarse en extraer características más amplias y ayudar a identificar características fundamentales de la señal.

A medida que se avanza en la red, se aumenta el número de filtros y se reduce el tamaño de la ventana de convolución. Esto permite que la segunda capa de convolución se especialice en capturar detalles más finos y específicos de la señal de vibración, como picos, oscilaciones o patrones locales.

A cada capa de convolución la sigue una capa de max pooling con el objetivo de reducir el número de parámetros, y por tanto la carga computacional, manteniendo las características relevantes identificadas.

A la salida de la CNN se tienen unas capas completamente conectadas para extraer patrones más complejos, terminando con una capa densa con 3 neuronas, una para cada clase. La red tendrá a la salida la distribución de probabilidad de que los datos de entrada pertenezca a cada clase de fallo. Finalmente, se realiza la predicción seleccionando la clase que tenga el valor máximo a la salida para cada muestra de datos.

Para el entrenamiento del modelo CNN, se realizaron 20 epochs o iteraciones completas a través de los datos de entrenamiento, y se registró un tiempo de entrenamiento de 4 minutos. Durante este proceso, se registraron las métricas de exactitud (accuracy) y pérdida (loss) en cada iteración para evaluar el rendimiento del modelo a medida que aprende. La Figura 19 muestra los gráficos de la evolución de estas métricas con los datos de entrenamiento durante el aprendizaje de la red.

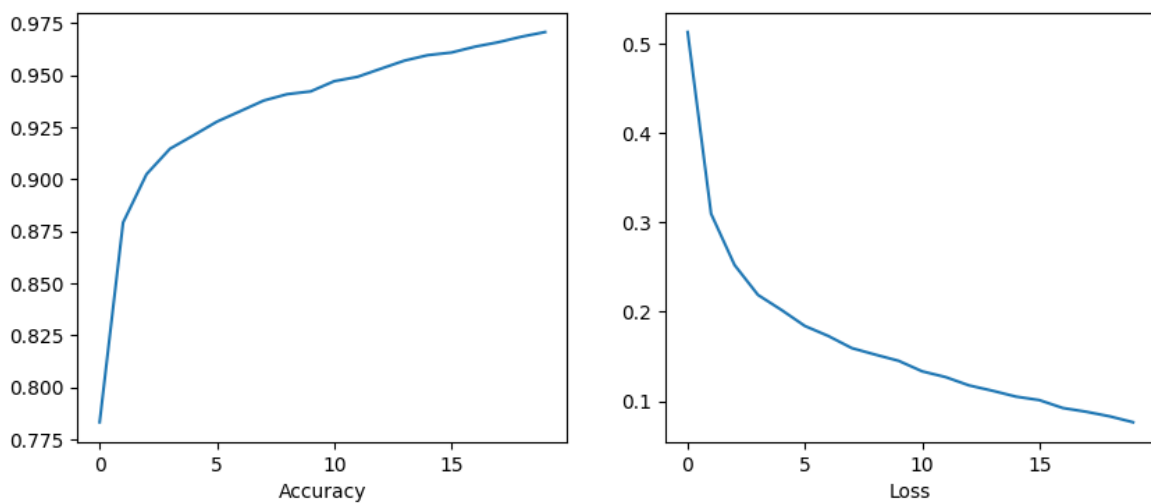


Figura 19 - Aprendizaje del modelo CNN

Al analizar la gráfica del accuracy, se observa una tendencia ascendente a medida que avanzan los epochs. Esto indica que el modelo está mejorando su capacidad para clasificar correctamente las muestras de entrenamiento. Al finalizar las 20 epochs, el accuracy del entrenamiento alcanza aproximadamente el 97%, lo cual es un valor bastante alto.

Por otro lado, al examinar la gráfica de pérdida, se observa una tendencia descendente. Las pérdidas disminuyen gradualmente a medida que el modelo aprende a hacer predicciones más precisas. Al finalizar las 20 epochs, las pérdidas alcanzan valores muy bajos, lo cual indica que el modelo está realizando buenas predicciones y ajustándose bien a los datos de entrenamiento.

Estos resultados son positivos, ya que muestran que el modelo ha aprendido de manera efectiva y ha logrado una alta precisión en la clasificación de las muestras de entrenamiento.

Sin embargo, es importante tener en cuenta que estos resultados se basan en el conjunto de datos de entrenamiento y es necesario evaluar el rendimiento del modelo en un conjunto de datos de prueba independiente para obtener una evaluación más completa de su capacidad para generalizar y realizar predicciones precisas en nuevos datos.

Tabla 7 - Resumen comparativo de los modelos CNN

Muestra de datos	Velocidad dato de entrada	Episodios	Batch size	Factor de aprendizaje	Primera capa Conv1D	Segunda capa Conv2D	Capas Densas	Accuracy en test	F1-score en test
Muchos experimentos por clase, 1 configuración	No	10	32	0.001	4 filtros tamaño 3	-	Dense(100, relu) Dense(3, softmax)	99.26%	Healthy: 98.92% OR fault: 99.89% IR fault: 98.94%
Pocos experimentos por clase, 1 configuración	No	10	32	0.001	4 filtros tamaño 3	-	Dense(100, relu) Dense(3, softmax)	99.24%	Healthy: 98.87% OR fault: 100% IR fault: 98.84%
Solo 2 experimentos por clase, 1 configuración	No	10	32	0.001	4 filtros tamaño 3	8 filtros tamaño 3	Dense(100, relu) Dense(3, softmax)	97.85%	Healthy: 96.93% OR fault: 100% IR fault: 96.69%
Todos los experimentos, todas las configuraciones	Sí	20	32	0.001	3 filtros tamaño 5	12 filtros tamaño 2	Dense(128, relu) Dense(3, softmax)	<b>93.91%</b>	Healthy: 94.47% OR fault: 95.62% IR fault: 91.51%

En esta etapa de evaluación el modelo final alcanzó un accuracy del 93.91% en los datos de prueba. En la Tabla 7 se presenta un resumen con los casos probados más relevantes, incluyendo sus arquitecturas y los resultados que obtienen. Al comparar los resultados del modelo final con las pruebas anteriores, se observa una disminución en la exactitud y en la métrica F1-score, que representa el equilibrio entre precisión y recall.

Sin embargo, es importante tener en cuenta que en las pruebas anteriores se obtuvo un accuracy de 62% cuando se evaluaron datos no vistos que pertenecían a diferentes configuraciones. En contraste, en esta evaluación más reciente se incluyeron datos de todos

los experimentos, lo que aumenta la variabilidad y la capacidad del modelo de generalizar. Por lo tanto, aunque el accuracy pueda ser inferior, este resultado es más representativo de la capacidad del modelo para clasificar diversos tipos de errores en condiciones más realistas. En la sección de resultados se realizará un análisis exhaustivo y se compararán los resultados detalladamente para obtener una visión completa de la efectividad del modelo.

### **5.2.2 MODELO DE DOBLE DQN**

El siguiente modelo que se diseñó fue una red de aprendizaje por refuerzo, en concreto una doble DQN. En el capítulo anterior se han detallado los conceptos básicos del RL, donde un agente interactúa con un entorno. El entorno tiene distintos estados y el agente le responde con acciones. Dependiendo de si la acción ha sido buena o no, se le dará una recompensa.

En este contexto, se ha adaptado el aprendizaje por refuerzo al problema de detección de fallos en maquinaria. La arquitectura de la red se muestra en la Figura 20. En este caso de aplicación, los estados corresponden a los datos de entrada, es decir, la señal de vibración en el dominio de la frecuencia y la variable de velocidad de rotación. El agente toma acciones que corresponden a la clase de fallo que se predice. Si la predicción del agente es correcta, se le otorga una recompensa positiva (+1) y si no, una negativa (-1).

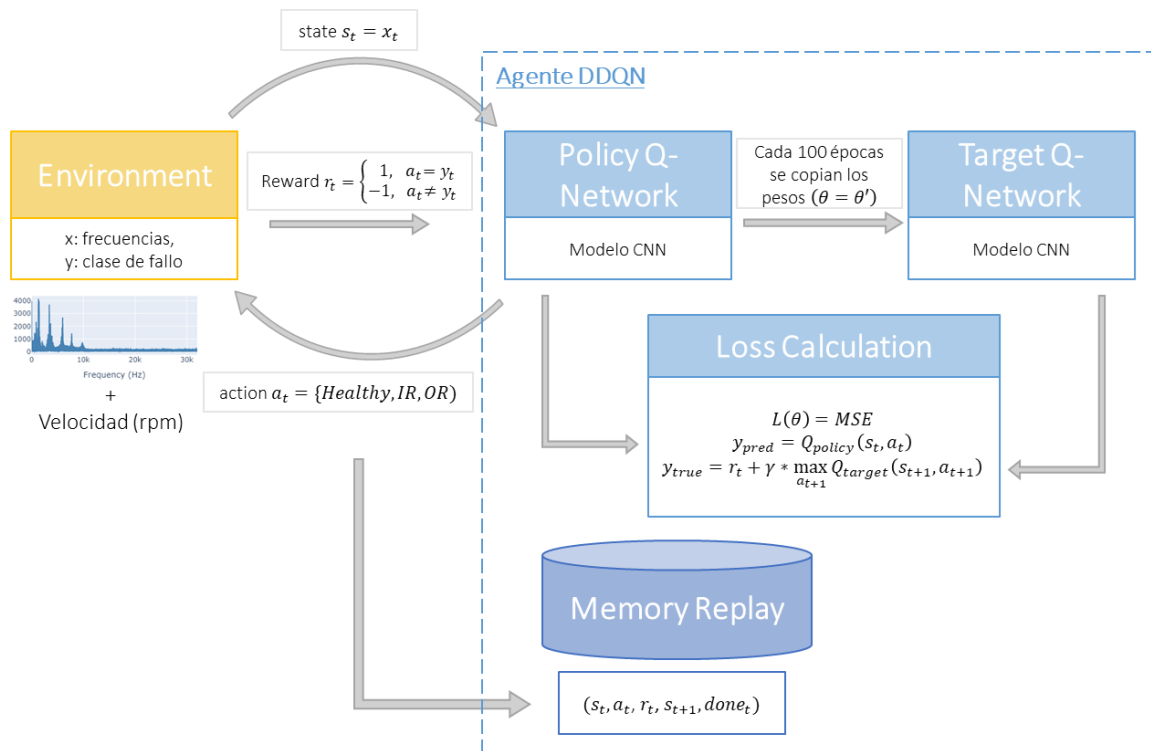


Figura 20 - Esquema del modelo DDQN

Como se explicó anteriormente, la característica particular de las DDQN es que el agente está compuesto por 2 redes, una principal y otra objetivo. Esta configuración evita que el agente se quede atascado en un bucle infinito de acciones subóptimas ya que la red objetivo ayuda a guiar al agente hacia acciones que pueden ser mejores en el futuro. En la arquitectura diseñada, la red objetivo actualizará sus pesos cada 'C' épocas al sincronizarse con la red principal, donde 'C' corresponde al periodo de copia.

Se ha optado por una estrategia en la cual se reinicia la memoria del agente al comenzar cada episodio y se actualizan los Q-valores al final de cada episodio. Esta elección se basa en las siguientes razones.

En primer lugar, reiniciar la memoria al iniciar cada partida permite al agente comenzar desde un estado limpio, sin ninguna influencia de episodios anteriores. Esto es especialmente importante en el contexto de detección de fallos en maquinaria, donde cada partida representa una instancia independiente de medición. Al reiniciar la memoria, se asegura que

el agente no conserve información no relevante o sesgos de episodios anteriores, lo que puede mejorar su capacidad para aprender y adaptarse a nuevas situaciones.

En segundo lugar, la actualización de los Q-valores al final de cada episodio permite al agente aprender de forma más efectiva a medida que se acumula información a lo largo de cada partida. Al actualizar los Q-valores al final de cada episodio, se proporciona una retroalimentación inmediata al agente sobre la calidad de las acciones tomadas durante ese episodio específico. Esto permite que el agente ajuste sus estrategias de toma de decisiones de manera más rápida y precisa, mejorando así su rendimiento en futuros episodios.

Además, es importante mencionar que la arquitectura escogida para las redes principal y objetivo es la del modelo CNN expuesto en la sección anterior. De esta manera se pretende aprovechar las capacidades que ofrecía el modelo de aprendizaje por refuerzo en la detección de anomalías. También, al igual que en la red convolucional, la salida del modelo DDQN es la distribución de probabilidad para cada clase de fallo y para realizar la clasificación se escoge la que tenga una mayor probabilidad.

Para explicar el proceso de entrenamiento se puede utilizar una analogía con videojuegos. En este trabajo, el modelo fue entrenado a lo largo de 900 episodios, que se refiere al número de veces que se juega el videojuego. Cada episodio consiste en 64 rondas, que representa el número de iteraciones dentro de cada partida.

Durante el proceso de afinamiento del modelo DDQN, se llevaron a cabo diversas pruebas para ajustar los parámetros y optimizar su rendimiento. Inicialmente, se realizó una prueba preliminar utilizando una muestra reducida de los datos, donde se excluyó la velocidad de rotación como entrada, solo se cogieron datos de una configuración y no se incluyeron todos los experimentos. Los parámetros clave, como el factor de aprendizaje, el factor de descuento y el período de copia, se ajustaron cuidadosamente. El factor de aprendizaje se estableció en 0.001, que es un valor típico utilizado en muchos algoritmos de aprendizaje automático. Para el factor de descuento ( $\gamma$ ) se probó en un rango de 0 a 1. En el repaso teórico se explicó cómo un valor de  $\gamma$  cercano a 1 significa que se les da más importancia a las recompensas a largo plazo, mientras que un valor más cercano a 0 enfatiza más las



recompensas inmediatas. Para este primer modelo, se observó que valores de  $\gamma$  altos afectaban negativamente al rendimiento, por lo que se dejó en un 0.001. El periodo de copia se configuró en 10, que significa que cada 10 episodios los pesos de la red objetivo se sincronizan con los de la red principal. En cuanto a las estrategias de exploración y la explotación, se adoptó una estrategia Epsilon-Greedy con decaimiento de épsilon. Se llevaron a cabo pruebas utilizando dos enfoques diferentes para definir la variable épsilon. Uno de ellos consistía en un decaimiento lineal, mientras que el otro se basaba en un decaimiento exponencial. Tras evaluar el rendimiento de ambos enfoques, se determinó que el decaimiento exponencial mostraba mejores resultados, proporcionando una transición más suave y adaptable de la exploración a la explotación a lo largo del entrenamiento. Por lo tanto, se decidió utilizar el algoritmo de decaimiento exponencial para ajustar el valor de épsilon a lo largo de los episodios de entrenamiento. Para ello, se utilizó la Ecuación 4.

$$\varepsilon = \varepsilon_{min} + (\varepsilon_0 - \varepsilon_{min}) \times e^{-\varepsilon_{decay} \times episode\_count}$$

*Ecuación 4 - Decaimiento exponencial de épsilon en la estrategia Epsilon-Greedy para el modelo DDQN*

Donde  $\varepsilon$  es el valor actual de épsilon,  $\varepsilon_{min}$  es el valor mínimo de épsilon,  $\varepsilon_0$  es el valor inicial de épsilon,  $\varepsilon_{decay}$  es el factor de decaimiento de épsilon y  $episode\_count$  es el número de episodios de entrenamiento hasta el momento. Esta fórmula permite ajustar de manera efectiva el valor de épsilon a medida que se avanza en el entrenamiento del modelo. En este caso, se establecieron los siguientes valores:  $\varepsilon_{min} = 0.01$ ,  $\varepsilon_0 = 0.4$  y  $\varepsilon_{decay} = 0.3$ .

Este primer modelo se entrenó a lo largo de 450 episodios con 64 rondas cada uno. Se alcanzó un accuracy con datos de test de 95.63%, siguiendo un comportamiento similar a los primeros modelos CNN ya que únicamente presenta dificultades al distinguir entre daños en el anillo interior y maquinaria saludable. Además, al igual que ocurría en las CNN, estos errores se deben principalmente a los fallos de la máquina provocados artificialmente. Debido a que este primer modelo utiliza datos de entrenamiento muy reducidos, se comprobó el rendimiento en datos de test con distintas configuraciones de la Tabla 6. Las métricas de accuracy son las siguientes al modificar las siguientes variables: la fuerza radial un 95.68%, el par de carga un 95.32% y la velocidad de rotación un 68.7. Los resultados obtenidos

muestran una mayor estabilidad en comparación con el modelo de red convolucional, ya que el rendimiento no se ve tan afectado al realizar pruebas con diferentes configuraciones. Sin embargo, se observa un deterioro evidente al evaluar datos correspondientes a velocidades de rotación distintas. Esto justifica la inclusión de la velocidad de rotación como una variable de entrada en el modelo DDQN.

Para continuar con el proceso de optimización, las pruebas adicionales se llevaron a cabo incluyendo datos de todas los experimentos, abarcando diversas configuraciones y considerando la velocidad de rotación como una variable adicional de entrada. Durante este proceso de optimización, especialmente se realizaron ajustes en dos parámetros clave del modelo con el objetivo de mejorar su capacidad predictiva y su capacidad de generalización: el periodo de copia y el factor de descuento.

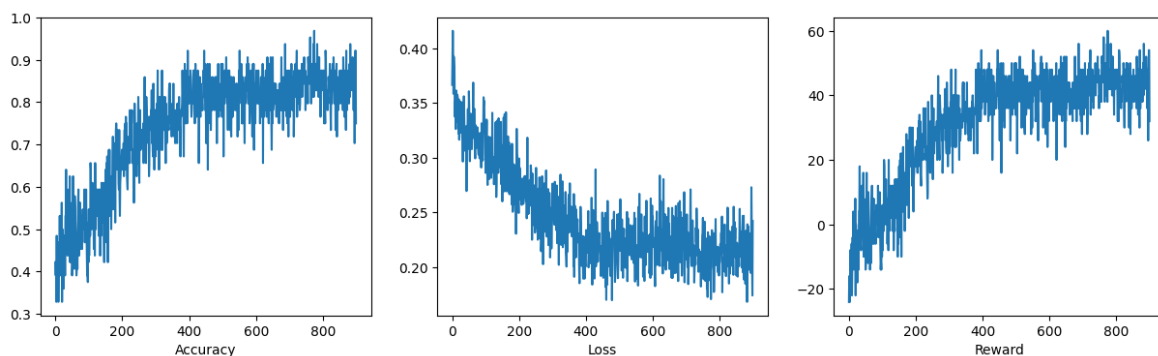
El periodo de copia de los pesos a la red objetivo influye en la estabilidad y convergencia del modelo. En el modelo inicial se había establecido en 10, pero se decidió aumentar este valor facilitar una transferencia de conocimientos más sólida y gradual de la red principal a la red objetivo. Se realizaron pruebas con un periodo de copia de 50 y de 100, pero se optó por el último ya que el entrenamiento del modelo presentaba mejoras al converger más rápido y de manera más estable.

Además, se realizaron modificaciones en el factor de descuento, el cual determina la importancia relativa de las recompensas futuras en relación con las recompensas inmediatas. Se probaron diferentes valores para el factor de descuento, evaluando su impacto en el desempeño del modelo. En el modelo inicial los resultados indicaban que un factor de descuento cercano a 0 era la elección óptima. Sin embargo, al realizar pruebas con el conjunto de datos de entrenamiento completo se obtuvo que al utilizar un factor de descuento de 0.01 el accuracy bajaba a un 78.91%, mientras que al utilizar un factor de descuento de 0.9 el accuracy llegaba al 82.53%. A raíz de estos resultados, se decidió establecer en 0.7 ya que el rendimiento mejoraba.

Las gráficas proporcionadas en la Figura 21 muestran el progreso del aprendizaje del modelo definitivo durante el entrenamiento. En la gráfica de accuracy se puede observar cómo el

rendimiento del modelo va aumentando a medida que aprende. Al final del entrenamiento, el modelo logra alcanzar un nivel de exactitud con datos de entrenamiento que oscila entre el 0.8 y el 0.9, lo cual es considerado un resultado satisfactorio.

Por otro lado, la última gráfica representa la recompensa acumulada. Dado que cada episodio consta de 64 rondas, la recompensa máxima posible sería 64. Aunque el modelo no alcanza el puntaje máximo, se observa que se mantiene alrededor de 50, lo cual indica un desempeño bastante bueno.



*Figura 21 - Aprendizaje del modelo DDQN*

En resumen, el modelo de aprendizaje por refuerzo fue entrenado a través de múltiples episodios y rondas, y se pudo apreciar un progreso en términos de accuracy, pérdida y recompensa acumulada. Estos resultados respaldan la efectividad del enfoque de aprendizaje por refuerzo en la detección de anomalías en maquinaria rotativa.

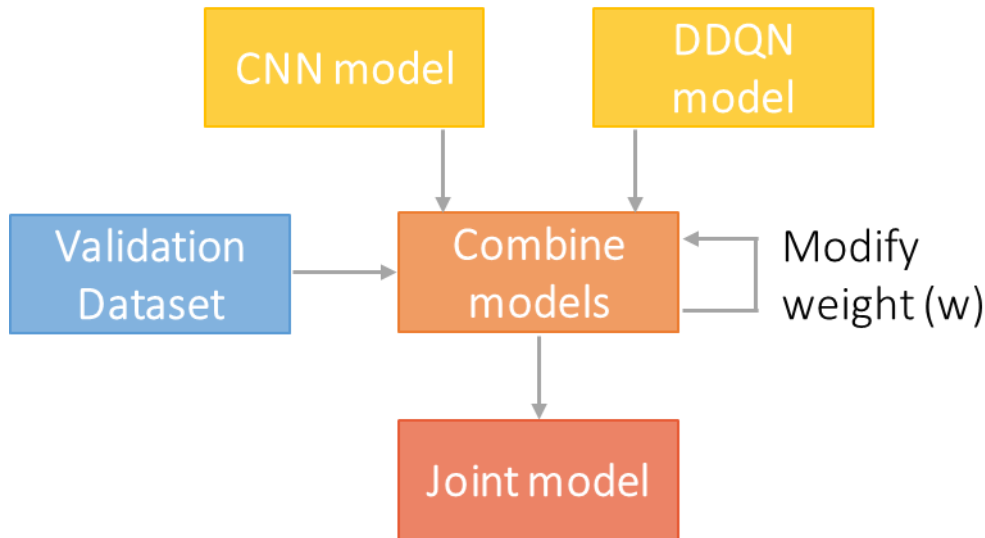
Una vez completado el entrenamiento, se procedió a evaluar el rendimiento del modelo utilizando el conjunto de datos de prueba, obteniendo un accuracy del 84.81%. Este valor supera a los obtenidos en las pruebas anteriores con diferentes configuraciones de parámetros. Para una mejor visualización y comparación de los resultados, se presenta en la Tabla 8 un resumen que muestra los parámetros modificados en cada prueba relevante, junto con el correspondiente accuracy alcanzado en el conjunto de datos de prueba. En la sección de análisis de resultados al final del capítulo, se realizará un análisis detallado y exhaustivo que compara diferentes métricas obtenidas al evaluar los modelos diseñados con los datos de prueba.

Tabla 8 - Resumen comparativo de los modelos DDQN

<i>Muestra de datos</i>	<i>Velocidad dato de entrada</i>	<i>Episodios</i>	<i>Iteraciones por episodio</i>	<i>Factor de aprendizaje</i>	<i>Factor de descuento</i>	<i>Periodo de copia</i>	<i>Accuracy en test</i>
Pequeña y poca variedad	No	450	64	0.001	0.001	10	95,63% (68,7% con otra velocidad)
Incluye todos los experimentos	Sí	900	64	0.001	0.001	50	71,35%
Incluye todos los experimentos	Sí	900	64	0.001	0.01	100	78,91%
Incluye todos los experimentos	Sí	900	64	0.001	0.9	100	82,53%
Incluye todos los experimentos	Sí	900	64	0.001	0.7	100	<b>84,81%</b>

### 5.2.3 MODELO CONJUNTO

Finalmente, se ha desarrollado un modelo conjunto que combina las salidas de los modelos previamente entrenados, con el objetivo de fortalecer la fiabilidad en la clasificación de anomalías. Para lograr esto, se asigna un peso específico a cada modelo, y a través de pruebas con datos de validación, se determina la combinación óptima de pesos. Tras exhaustivas iteraciones, se llega a la conclusión de asignar un peso de 0.7 al modelo convolucional y un peso de 0.3 a la DDQN. En la Figura 22 se muestra un esquema del proceso seguido para el desarrollo del modelo conjunto mediante el ajuste de los pesos. Esta combinación ponderada busca aprovechar las fortalezas individuales de cada modelo y mejorar la precisión general del sistema de detección de anomalías.



*Figura 22 - Proceso de diseño del modelo conjunto*

Además, al evaluar el modelo conjunto con los datos de prueba, se obtuvo un accuracy del 94.03%, lo que indica un desempeño prometedor en la clasificación de anomalías. En la siguiente sección de análisis de resultados, se realizará una comparativa exhaustiva con otros modelos, se explorarán diferentes métricas y se profundizará en el rendimiento y las limitaciones del sistema de detección de anomalías desarrollado.

### **5.3 ANÁLISIS DE RESULTADOS**

Este capítulo se divide en dos secciones principales. En primer lugar, se realizará una comparación exhaustiva de los tres modelos desarrollados: la red convolucional (CNN), el modelo de aprendizaje por refuerzo (DDQN) y el modelo conjunto. Se evaluarán diferentes métricas para determinar las fortalezas y debilidades de cada uno. A continuación, se procederá a un análisis adicional en el que se compararán los resultados obtenidos con datos de referencia de artículos previos. Esta estructura permitirá una evaluación integral de los modelos y establecerá una base sólida para evaluar su rendimiento en relación con el estado del arte.

### 5.3.1 MODELO CONJUNTO

En el análisis de resultados, se realizaron diversas métricas y evaluaciones para los modelos individuales (CNN y DDQN) y el modelo conjunto. Los resultados mostraron mejoras en varios aspectos al utilizar el modelo conjunto en comparación con los modelos individuales.

En términos de accuracy, como se puede observar en la Tabla 9, el modelo CNN supera al DDQN, pero al construir el modelo conjunto se consigue combinar y aprovechar el conocimiento adquirido por ambos modelos de manera eficiente. Ya podíamos ver en la asignación de pesos del modelo conjunto que el modelo CNN iba a tener un mejor rendimiento ya que se le daba más importancia, pero la DDQN consigue detectar patrones que la CNN no, por lo que al juntarlos se consigue un modelo más robusto.

Tabla 9 - Accuracy de los modelos: análisis comparativo

	<i>CNN</i>	<i>DDQN</i>	<i>Conjunto</i>
<i>Accuracy</i>	0.9391	0.8481	<b>0.9403</b>

Sin embargo, es importante evaluar otras métricas para poder tener una visión global del rendimiento de los modelos. En las siguientes tablas se observan los resultados obtenidos para la precisión (Tabla 10), el recall (Tabla 11) y el F1-score (Tabla 12), donde el modelo conjunto obtiene mejores resultados en la mayoría de los casos.

En el análisis del recall se puede apreciar cómo el modelo conjunto disminuye el recall de la clase healthy con respecto al modelo CNN. Esto significa que está fallando más al categorizar maquinaria sana como defectuosa. No obstante, es importante destacar que, en este caso de aplicación, se sigue un enfoque que se centra en maximizar el recall para las clases de fallos (IR y OR). Esto se debe a que, en el caso de detección de fallos en maquinaria, detectar la mayor cantidad posible de daños es crucial. Es preferible tener más falsos positivos a tener falsos negativos ya que esto significaría tener maquinaria dañada sin detectar, lo que podría causar pérdidas económicas significativas. El modelo conjunto demuestra mejores resultados en cuanto al recall para las clases de fallos, en comparación

con los modelos individuales y, por lo tanto, para una empresa es preferible ya que el coste de revisar una maquinaria saludable es menor que el coste de reparación añadido al no evitar que la maquinaria se desgaste aún más.

Además, se evalúa el F1-score, que es una métrica que combina la precisión y el recall. En todas las clases de fallos, se observa una mejora en el F1-score al utilizar el modelo conjunto en comparación con los modelos individuales. Esto indica que el modelo conjunto logra un mejor equilibrio entre la precisión y el recall, lo que resulta en un rendimiento general mejorado.

Tabla 10 - Precisión de los modelos: análisis comparativo

	<i>CNN</i>	<i>DDQN</i>	<i>Conjunto</i>
<i>Healthy</i>	0.9309	0.85	<b>0.9373</b>
<i>OR fault</i>	<b>0.9657</b>	0.8471	0.9615
<i>IR fault</i>	0.9201	0.8468	<b>0.9209</b>

Tabla 11 - Recall de los modelos: análisis comparativo

	<i>CNN</i>	<i>DDQN</i>	<i>Conjunto</i>
<i>Healthy</i>	<b>0.9589</b>	0.9347	0.9558
<i>OR fault</i>	0.9469	0.8156	<b>0.9514</b>
<i>IR fault</i>	0.9101	0.7919	<b>0.9122</b>

Tabla 12 - F1-score de los modelos: análisis comparativo

	<i>CNN</i>	<i>DDQN</i>	<i>Conjunto</i>
<i>Healthy</i>	0.9447	0.8903	<b>0.9465</b>
<i>OR fault</i>	0.9562	0.831	<b>0.9564</b>
<i>IR fault</i>	0.9151	0.8185	<b>0.9165</b>

En cuanto a las matrices de confusión, se muestran los resultados para los tres modelos (Figura 23, Figura 24 y Figura 25). Se puede observar que el modelo conjunto reduce los errores de clasificación para las clases de fallos, lo cual se alinea con los resultados de las métricas mencionadas anteriormente. Sin embargo, también se observa que el modelo conjunto tiene un ligero aumento en los falsos positivos ya que la clase de estado saludable (healthy) tiene menos aciertos. Esto se relaciona con la estrategia mencionada anteriormente de maximizar el recall para las clases de fallos.

Según estudios previos en el campo, se ha observado que algunos investigadores tienden a focalizarse en los daños en el anillo exterior, a pesar de que los daños en el anillo interior son más propensos a ocurrir, como se explica en [7]. Además, detectar los fallos en el anillo interior resulta más desafiante, ya que la señal se ve afectada por una modulación debido a la rotación de dicho anillo. En consonancia con estas observaciones, en las matrices de confusión se observa que el caso más común de error se produce cuando un fallo en el anillo interior es confundido con el estado de maquinaria saludable. Estos hallazgos respaldan la importancia de abordar tanto los fallos en el anillo interior como en el anillo exterior en el sistema de detección de anomalías. Aunque se reconocen las dificultades inherentes a la detección de fallos en el anillo interior, la metodología de este trabajo logra obtener buenos resultados en esta tarea desafiante.



CNN Model

		Predicted Label		
		Healthy	OR fault	IR fault
True Label	Healthy	3406 (32.3%)	16 (0.2%)	130 (1.2%)
	OR fault	53 (0.5%)	3409 (32.3%)	138 (1.3%)
	IR fault	200 (1.9%)	105 (1.0%)	3088 (29.3%)

Figura 23 - Matriz de confusión modelo CNN

DDQN Model

		Predicted Label		
		Healthy	OR fault	IR fault
True Label	Healthy	3320 (31.5%)	194 (1.8%)	38 (0.4%)
	OR fault	216 (2.0%)	2936 (27.8%)	448 (4.2%)
	IR fault	370 (3.5%)	336 (3.2%)	2687 (25.5%)

Figura 24 - Matriz de confusión modelo DDQN

Joint Model		Predicted Label		
		Healthy	OR fault	IR fault
True Label	Healthy	3395 (32.2%)	31 (0.3%)	126 (1.2%)
	OR fault	35 (0.3%)	3425 (32.5%)	140 (1.3%)
	IR fault	192 (1.8%)	106 (1.0%)	3095 (29.4%)

Figura 25 - Matriz de confusión modelo conjunto

En relación a las curvas ROC, se obtienen resultados satisfactorios para los tres modelos (Figura 26, Figura 27 y Figura 28). La curva ROC representa la relación entre la tasa de verdaderos positivos y la tasa de falsos positivos a medida que se varía el umbral de clasificación. Si bien es cierto que la curva ROC del modelo conjunto mostró un ligero deterioro en comparación con la red convolucional, esto se compensa con las mejoras en otras métricas mencionadas previamente, como el recall y el F1-score.

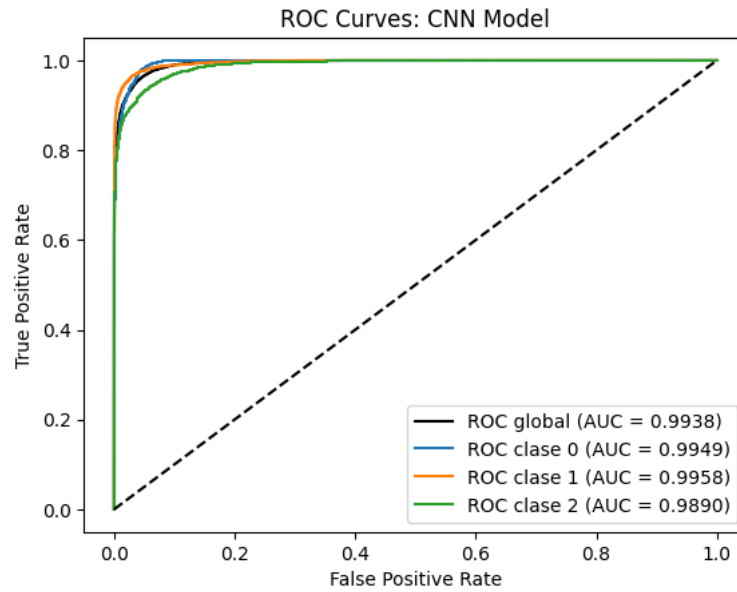


Figura 26 - Curvas ROC modelo CNN

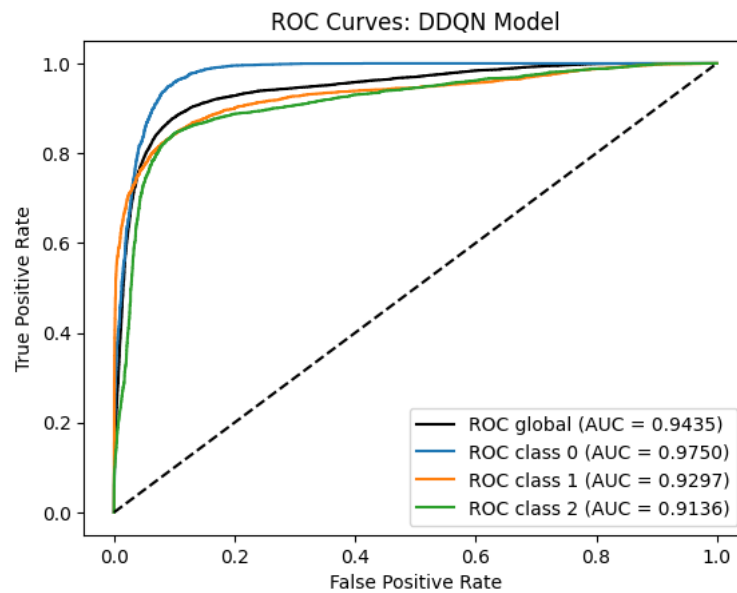


Figura 27 - Curvas ROC modelo DDQN

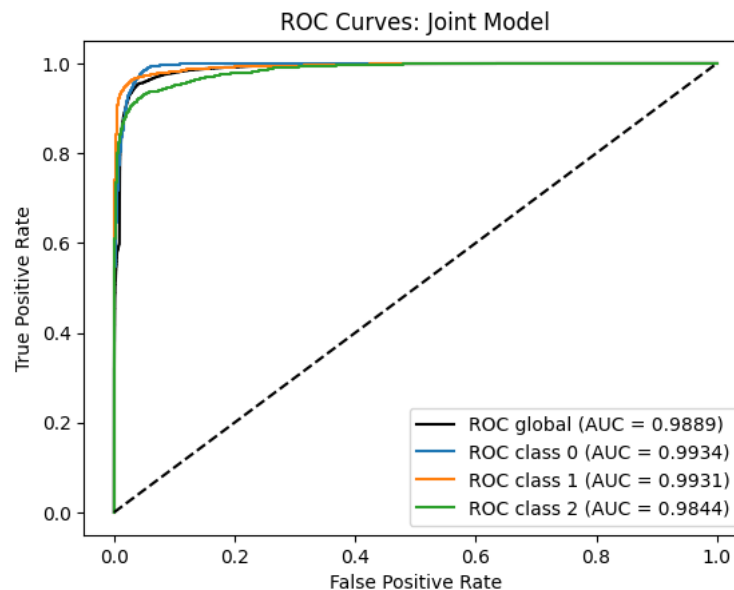


Figura 28 - Curvas ROC modelo conjunto

En base a los resultados obtenidos, se puede concluir que el enfoque de modelo conjunto, que combina la red convolucional y la red de aprendizaje por refuerzo, ha demostrado ser efectivo en la detección de anomalías en maquinaria rotativa. Se observa que el modelo conjunto logra mejorar el rendimiento en términos de métricas clave, como el accuracy y el recall, en comparación con los modelos individuales.

Es importante destacar que, si bien la DDQN requiere un tiempo de entrenamiento más prolongado en comparación con la CNN, se espera que este enfoque mantenga un rendimiento sólido incluso cuando se dispone de menos datos de entrenamiento. Por otro lado, la CNN puede experimentar un empeoramiento en su rendimiento con una cantidad limitada de datos de entrenamiento.

Al combinar ambos modelos en el enfoque conjunto, se obtienen sinergias aprovechando las fortalezas de cada uno. Por un lado, la CNN captura características relevantes en la señal de vibración, mientras que la DDQN aporta capacidades de toma de decisiones más sofisticadas. Esta combinación nos permite detectar de manera efectiva los diferentes tipos de fallos, incluidos aquellos que son más propensos a ocurrir y que pueden resultar más difíciles de identificar.

### **5.3.2 ANÁLISIS COMPARATIVO CON DATOS DE REFERENCIA**

En esta sección, se llevará a cabo un análisis comparativo de los resultados obtenidos en este trabajo en relación con otros trabajos de referencia. Para ello, se enfoca en dos fuentes clave: el estudio del cual se han extraído los datos utilizados [7] y el artículo de referencia que ha influido en el enfoque de este trabajo [4].

En el artículo [7], se llevaron a cabo diversas pruebas utilizando varios modelos, como árboles de clasificación y regresión (CART), Random Forest (RF), Boosted Trees (BT), redes neuronales (NN) sin optimización de parámetros, SVM, máquinas de aprendizaje extremo (ELM) y k-NN. Además, se exploró un modelo conjunto (Ensemble) que combinaba todos los algoritmos probados con el objetivo de mejorar el rendimiento. En su análisis, se evaluaron estos modelos cambiando el conjunto de datos de entrada, centrándose en las señales de vibración y de corriente del motor (MCS) para determinar cuál de ellas era más eficiente en la detección de anomalías en maquinaria rotativa. Además, para todos los experimentos realizados utilizan datos de la primera configuración de la Tabla 6 ya que observan que es la que mejores resultados proporciona.

En la primera prueba que realizan utilizan datos de fallos provocados artificialmente para el entrenamiento y datos de fallos reales para el test. En la Tabla 13, se muestran los resultados del accuracy de los modelos. La primera observación que se hace es que al utilizar las señales de vibración como entrada, se obtenía un rendimiento superior en comparación con las señales de corriente del motor. Además, el modelo conjunto (Ensemble) demostró ser el mejor para los datos de vibración, aunque se observó que el rendimiento era relativamente bajo, con un accuracy del 75%, mientras que el modelo SVM resultó en el mejor accuracy con un 60.9% para los datos de corriente del motor. Estos resultados son coherentes, ya que durante el entrenamiento se utilizaron datos de daños artificiales que pueden no ser extrapolables a casos reales.

Tabla 13 - Rendimiento de los modelos entrenados con datos de daños artificiales y testeados con datos de daños reales

Algorithm	Classification Accuracy [%]	
	MCS	Vibration Signals
CART	26.8	65.9
RF	45.0	64.1
BT	38.6	62.3
NN	45.5	65.5
SVM-PSO	<b>60.9</b>	65.5
ELM	45.5	65.9
kNN	45.5	63.2
Ensemble	45.9	<b>75.0</b>

En la siguiente prueba, se utilizaron únicamente datos reales para el entrenamiento y el testeo, aplicando además una técnica de validación cruzada de 5 pliegues (5-fold cross-validation). En esta configuración, los resultados mejoraron significativamente como se muestra en la Tabla 14, alcanzando un accuracy del 93.3% para la señal de corriente del motor en el modelo conjunto (Ensemble), y del 98.3% para los modelos CART, RF y Ensemble en la señal de vibración.

Tabla 14 - Rendimiento de los modelos entrenados y testeados con datos reales utilizando 5-fold cross-validation

Algorithm	Classification Accuracy [%]	
	MCS	Vibration Signals
CART	66.7	<b>98.3</b>
RF	83.3	<b>98.3</b>
BT	81.7	83.3
NN	65.8	44.2
SVM-PSO	56.7	75.8
ELM	69.2	60.8
kNN	68.3	62.5
Ensemble	<b>93.3</b>	<b>98.3</b>

Por último, se llevó a cabo otra prueba similar a la anterior, pero incluyendo datos de experimentos con múltiples fallos en el anillo interior y exterior de las máquinas, asignando

una clase del daño dominante para cada caso. Se observó que los resultados fueron similares a los anteriores, aunque ligeramente inferiores (Tabla 15), lo que indica que el modelo enfrenta dificultades para distinguir cuando se tienen datos de máquinas con múltiples fallos. En esta configuración, se obtuvo un accuracy del 86.3% para la señal de corriente del motor en el modelo conjunto (Ensemble), y del 91.3% para los modelos CART, RF y Ensemble en la señal de vibración.

*Tabla 15 - Rendimiento de los modelos entrenados y testeados con datos reales utilizando 5-fold cross-validation incluyendo experimentos con múltiples fallos*

Algorithm	Classification Accuracy [%]	
	MCS	Vibration Signals
CART	86.3	<b>91.3</b>
RF	68.1	<b>91.3</b>
BT	63.7	79.4
NN	66.3	33.2
SVM-PSO	80.6	70.8
ELM	81.9	71.9
kNN	61.3	70.0
Ensemble	<b>86.3</b>	<b>91.3</b>

La importancia de utilizar la señal de vibración como entrada se destaca en este artículo, ya que ha demostrado mejorar el rendimiento de los modelos, tal como se ha observado en este proyecto. Asimismo, se resalta la utilidad de emplear un modelo conjunto (ensemble) para aprovechar las fortalezas de diversos algoritmos. En comparación con los resultados obtenidos en este trabajo, es importante tener en cuenta algunas diferencias en el enfoque.

En primer lugar, los modelos desarrollados en este proyecto han utilizado tanto datos de daños artificiales como datos reales para el entrenamiento y la evaluación. Además, se han empleado datos de las cuatro configuraciones disponibles, a diferencia de los modelos analizados en el artículo que solo utilizaron una configuración. El modelo conjunto diseñado en este trabajo ha logrado un accuracy del 94.03%, como se ha presentado en la sección anterior, superando en valor a la mayoría de los modelos descritos en el artículo, excepto aquellos que lograron un mejor rendimiento al ser entrenados y evaluados con datos reales

(Tabla 14). Cabe destacar que el modelo desarrollado en este proyecto ofrece una mayor versatilidad al ser adaptable a diferentes velocidades de rotación.

No obstante, es importante tener en cuenta que no se han utilizado datos de maquinaria con múltiples fallos en este proyecto, lo cual constituye una ventaja de los resultados presentados en la Tabla 15 del artículo. Por lo tanto, los resultados inferiores obtenidos en comparación pueden justificarse por la mayor robustez de los modelos del artículo en ese aspecto. La inclusión de datos de máquinas con múltiples fallos sería una mejora potencial a considerar para adaptar los modelos propuestos.

En resumen, los modelos de Deep Learning diseñados en este trabajo superan en rendimiento a los modelos expuestos en el artículo de la Universidad de Paderborn, demostrando resultados más sólidos y prometedores. Sin embargo, se reconoce la ventaja de los modelos del artículo al incluir datos de maquinaria con múltiples fallos. En consecuencia, se sugiere como posible mejora futura la incorporación de dichos datos para aumentar la robustez de los modelos propuestos.

En relación al artículo de referencia [4], a pesar de utilizar conjuntos de datos diferentes, se puede realizar un análisis comparativo de los resultados ya que se basa en datos de señales de vibración para la detección de fallos en maquinaria rotativa. En dicho artículo se emplearon dos conjuntos de datos: el conjunto de datos ‘A’ de fallos en rodamientos de la Universidad Case Western Reserve (CWRU) y el conjunto de datos ‘B’ de fallos en cajas de cambios de la Universidad de Connecticut (UOC). A pesar de las diferencias en los datos utilizados, este análisis comparativo permitirá obtener una perspectiva más amplia y contrastar los resultados obtenidos en este trabajo con los hallazgos previos.

En este artículo se compara el rendimiento de modelos de DL como las redes neuronales convolucionales (CNN) y las unidades recurrentes cerradas (GRU), y modelos de DRL que combinan el aprendizaje por refuerzo (RL) con el Deep Learning, como las CNN-RL y GRU-RL, con un modelo de SVM. En la Tabla 16 se pueden observar los resultados del accuracy en la evaluación de los modelos diseñados, destacando la mejora en los modelos de DRL que consiguen cerca del 100% en accuracy con ambos conjuntos de datos. Además,



este artículo elabora un análisis detallado del efecto de reducir el conjunto de entrenamiento en el rendimiento de los modelos. La Figura 29 muestra la evolución del accuracy al variar el tamaño de los datos de entrenamiento. Se puede observar cómo los modelos de DRL son estables manteniendo un accuracy satisfactorio, mientras que la CNN se ve más afectada por la reducción de datos.

Tabla 16 - Rendimiento de los modelos mostrando la media del accuracy  $\pm$  la desviación típica de 10 experimentos

Model	Dataset	
	A (Bearing)	B (Gear)
SVM	89.47 $\pm$ 1.38	90.27 $\pm$ 1.00
CNN	98.73 $\pm$ 1.81	95.31 $\pm$ 1.62
GRU	99.90 $\pm$ 0.20	99.11 $\pm$ 0.65
CNN-RL	99.95 $\pm$ 0.15	99.61 $\pm$ 0.50
GRU-RL	99.98 $\pm$ 0.05	99.95 $\pm$ 0.11

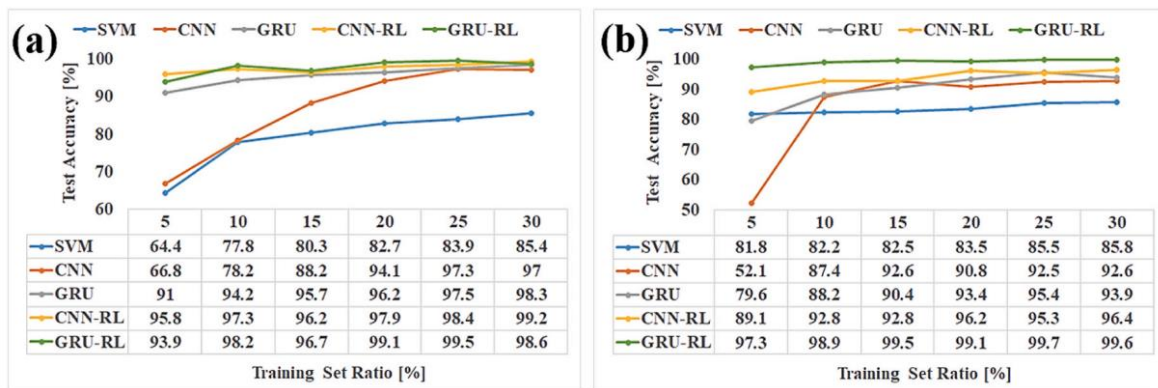


Figura 29 - Rendimiento de los modelos en escenarios de pocas muestras de datos

Comparando estos resultados con los obtenidos en este proyecto, se evidencia una mejora significativa. Esto podría atribuirse al uso de diferentes conjuntos de datos en el entrenamiento y la evaluación de los modelos, o también a la inclusión de datos de diversas configuraciones de velocidad de rotación, fuerza radial y par de carga en los modelos diseñados en este trabajo, a diferencia del artículo de referencia que se basa en una única configuración de datos.

En resumen, al comparar los resultados obtenidos con los estudios mencionados en el estado de la cuestión, se pueden extraer tres conclusiones significativas. En primer lugar, se evidencia claramente que el uso de técnicas de DL conduce a un rendimiento superior en comparación con enfoques de ML tradicionales. El estudio [7] que emplea métodos convencionales no logra alcanzar métricas tan satisfactorias como las obtenidas en este trabajo.

En segundo lugar, aunque los resultados obtenidos son altos, son ligeramente inferiores a los informados en otros estudios, como se observa en [4] donde se alcanza un accuracy cercano al 100%. Esta diferencia se debe a que se ha priorizado la construcción de un modelo robusto capaz de detectar diferentes tipos de fallos, originados por diversos mecanismos y bajo diversas configuraciones, aspectos que no fueron considerados en la literatura revisada.

En tercer lugar, los enfoques de DL y DRL demuestran un potencial significativo en sistemas de detección de anomalías como el presente. El enfoque de modelo conjunto brinda una solución robusta y confiable para la detección de anomalías en maquinaria rotativa, superando las limitaciones individuales de los modelos, y aprovechando las fortalezas del aprendizaje profundo y el aprendizaje por refuerzo. Esto permite obtener una mayor precisión y confiabilidad en la clasificación de fallos, lo que contribuye a la mejora de la eficiencia y la reducción de los costos asociados a las averías en la maquinaria rotativa.

## **Capítulo 6. CONCLUSIONES Y TRABAJOS FUTUROS**

En este trabajo se han elaborado diversos modelos para la detección de anomalías en maquinaria rotativa a partir de datos de vibración de la misma. Los datos experimentales se obtuvieron mediante la operación de un sistema de pruebas modular diseñado en la Universidad de Paderborn, el cual consta de un motor eléctrico, un eje de medición de torque, un módulo de prueba de rodamientos, un volante y un motor de carga. Además, se variaron la velocidad de rotación del sistema, la fuerza radial sobre el rodamiento de prueba y el par de carga en el tren de transmisión, garantizando así la robustez de los métodos de monitoreo de fallos en diferentes condiciones operativas.

La aplicación de estos modelos muestra resultados prometedores acerca del potencial de las técnicas de aprendizaje profundo (DL) y aprendizaje profundo reforzado (DRL) en esta área. Demuestran la capacidad de las técnicas de aprendizaje profundo para analizar y clasificar eficazmente los patrones de vibración, así como para detectar fallos en la maquinaria rotativa. Asimismo, se destaca el potencial del aprendizaje profundo reforzado, que permite mejorar la precisión y confiabilidad de los modelos al incorporar técnicas de refuerzo y toma de decisiones.

El modelo basado en Convolutional Neural Networks (CNN) ha demostrado ser eficaz en el procesamiento de datos de vibración y la extracción de características relevantes para la detección de fallos. Su capacidad para identificar patrones complejos en los datos ha contribuido a un rendimiento satisfactorio, con un accuracy del 93.91%. Esta red neuronal convolucional tiene la ventaja de aprender características automáticamente, lo que la hace adecuada para datos de alta dimensionalidad. Sin embargo, es importante tener en cuenta que puede requerir una gran cantidad de datos de entrenamiento para alcanzar su máximo potencial.

La CNN se enfoca en el procesamiento y extracción automática de características relevantes de los datos de vibración, lo que la hace adecuada para detectar patrones complejos. Por otro

lado, el modelo basado en la Doble DQN ha utilizado el aprendizaje por refuerzo para mejorar la toma de decisiones y el control en entornos dinámicos. Esta aproximación ha permitido capturar relaciones entre acciones y estados, lo que ha llevado a una clasificación de fallos con un accuracy del 84.81%. La ventaja principal de la DDQN radica en su capacidad para capturar relaciones entre acciones y estados a través de la interacción con el entorno. Sin embargo, es importante tener en cuenta que el entrenamiento de este modelo puede ser más lento debido a la naturaleza del aprendizaje por refuerzo.

Una de las principales novedades de este trabajo radica en la combinación de diferentes modelos aprovechando tanto el aprendizaje profundo como el aprendizaje profundo reforzado en el modelo conjunto. Esta estrategia ha demostrado ser eficaz para mejorar la capacidad de detección y clasificación de fallos en la maquinaria rotativa, superando a modelos individuales. La combinación de la CNN y la DDQN en el modelo conjunto ha permitido aprovechar las fortalezas de cada uno, obteniendo un modelo más robusto y preciso con un accuracy del 94%.

Además, otro aspecto destacable es la inclusión de la velocidad como dato de entrada en los modelos. Esto ha permitido que los modelos sean adaptables a cualquier estado de funcionamiento de la maquinaria, lo que resulta crucial para su aplicación en entornos industriales reales. Al considerar tanto las vibraciones como la velocidad, se logra una detección más precisa y se pueden identificar patrones de fallos específicos relacionados con la velocidad de rotación.

Los beneficios que ofrecen estos modelos son de gran relevancia para las empresas que dependen de maquinaria rotativa. La detección temprana de anomalías y fallos puede prevenir daños mayores, reducir costos de mantenimiento y evitar pérdidas económicas significativas. Los modelos propuestos proporcionan una herramienta eficiente y confiable para el monitoreo y el mantenimiento predictivo, permitiendo a las empresas tomar decisiones más informadas y optimizar sus procesos productivos.

En cuanto a los objetivos del proyecto, el diseño y desarrollo de los modelos de DL y DRL ha permitido evaluar su rendimiento de manera exhaustiva, tanto entre sí como en

comparación con la literatura existente. Los resultados obtenidos respaldan la efectividad y el potencial de estos enfoques en la detección de anomalías en maquinaria rotativa.

En términos de trabajos futuros, se sugiere explorar la posibilidad de abordar el problema de detección de anomalías como un problema de regresión, con el objetivo de predecir la intensidad de la vibración como un indicador continuo de la gravedad del fallo. Esto proporcionaría una información más detallada y permitiría una evaluación más precisa del estado de la maquinaria.

Además, el uso de “transfer learning” es otra línea de investigación interesante a considerar como se detalla en el artículo [8]. Aprovecharse de modelos pre-entrenados en áreas similares puede acelerar el proceso de entrenamiento y mejorar la generalización de los modelos, especialmente cuando los datos de entrenamiento son limitados.

En resumen, este proyecto ha logrado conclusiones sólidas y ha cumplido con los objetivos establecidos. Los modelos desarrollados, basados en técnicas de DL y DRL, han demostrado su eficacia en la detección de anomalías en maquinaria rotativa, superando en muchos aspectos a los modelos tradicionales. Estas contribuciones son relevantes tanto desde el punto de vista académico como para las empresas que buscan mejorar la eficiencia y la seguridad de sus sistemas de monitorización de maquinaria rotativa. Los trabajos futuros propuestos ofrecen nuevas oportunidades para seguir avanzando en este campo y maximizar el potencial de estos modelos en entornos industriales reales.

## Capítulo 7. BIBLIOGRAFÍA

- [1] S. Kim, D. An and J.-H. Choi, “Diagnostics 101: A Tutorial for Fault Diagnostics of Rolling Element Bearing Using Envelope Analysis in MATLAB,” *Applied Sciences*, vol. 10, no. 20, p. 7302, Oct. 2020.
- [2] J. van den Hoogen, S. Bloemheugel and M. Atzmueller, “Classifying Multivariate Signals in Rolling Bearing Fault Detection Using Adaptive Wide-Kernel CNNs,” *Applied Sciences*, vol. 11, no. 23, p. 11429, Dic. 2021.
- [3] SKF, “Conceptos básicos de rodamientos - Componentes y materiales,” SKF, [Online]. Available: <https://www.skf.com/ar/products/rolling-bearings/principles-of-rolling-bearing-selection/general-bearing-knowledge/bearing-basics/components-and-materials>. [Accessed May. 2023].
- [4] G. Qian and J. Liu, “Development of deep reinforcement learning-based fault diagnosis method,” *Progress in Nuclear Energy*, vol. 152, Sep. 2022.
- [5] B. Peng, Y. Bi, B. Xue, M. Zhang and S. Wan, “A Survey on Fault Diagnosis of Rolling Bearings,” *Algorithms*, vol. 15, no. 10, p. 347, Sep. 2022.
- [6] U. I. Inyang, I. Petrunin and I. Jennions, “Diagnosis of Multiple Faults in Rotating Machinery Using Ensemble Learning,” *Sensors*, vol. 23, no. 2, p. 1005, Ene. 2023.
- [7] C. Lessmeier, J. K. Kimotho, D. Zimmer and W. Sextro, “Condition Monitoring of Bearing Damage in Electromechanical Drive Systems by Using Motor Current Signals of Electric Motors: A Benchmark Data Set for Data-Driven Classification,” *European Conference of the Prognostics and Health Management Society*, 2016.

- [8] M. Hakim, A. A. Borhana Omran, A. Najah Ahmed, M. Al-Waily and A. Abdellatif, “A systematic review of rolling bearing fault diagnoses based on deep learning and transfer learning: Taxonomy, overview, application, open challenges, weaknesses and recommendations,” *Ain Shams Engineering Journal*, vol. 14, no. 4, p. 101945, 2023.
- [9] J. Peng, E. Jury, P. Dönnés y C. Ciurtin, «Machine Learning Techniques for Personalised Medicine Approaches in Immune-Mediated Chronic Inflammatory Diseases: Applications and Challenges,» *Frontiers in Pharmacology*, vol. 12, p. Sep., 2021.
- [10] Z. Elhamraoui, “Introduction to convolutional neural network,” Medium - Analytics Vidhya, May. 2020. [Online]. Available: <https://medium.com/analytics-vidhya/introduction-to-convolutional-neural-network-6942c189a723>.
- [11] S. Pokhrel, “Beginners Guide to Convolutional Neural Networks,” Towards Data Science, Sep. 2019. [Online]. Available: <https://towardsdatascience.com/beginners-guide-to-understanding-convolutional-neural-networks-ae9ed58bb17d>.
- [12] M. Chaudhary, “Activation Functions: Sigmoid, Tanh, ReLU, Leaky ReLU, Softmax,” Medium, Ago. 2020. [Online]. Available: <https://medium.com/@cmukesh8688/activation-functions-sigmoid-tanh-relu-leaky-relu-softmax-50d3778dcea5>.
- [13] S. Bhatt, “Reinforcement Learning 101,” Towards Data Science, Mar. 2018. [Online]. Available: <https://towardsdatascience.com/reinforcement-learning-101-e24b50e1d292>.
- [14] ODS, “Objetivos de Desarrollo Sostenible,” Naciones Unidas, 2015. [Online]. Available: <https://www.un.org/sustainabledevelopment/es/objetivos-de-desarrollo-sostenible/>.

# ANEXO I: ALINEACIÓN DEL PROYECTO CON LOS ODS

Los Objetivos de Desarrollo Sostenible (ODS) [14] son una iniciativa de las Naciones Unidas que se establecieron en 2015 con el propósito de abordar los desafíos globales y promover un desarrollo sostenible en todo el mundo. Los ODS están compuestos por 17 objetivos relacionados entre sí, abarcando diversos temas desde la erradicación de la pobreza hasta la lucha contra el cambio climático. En la Figura 30 se pueden observar los 17 objetivos que se definen.



Figura 30 - Objetivos de Desarrollo Sostenible

En el contexto de este proyecto, se identifican varios ODS con los que se alinea y, a continuación, se detalla cómo la aplicación de técnicas de Deep Learning para la detección de anomalías en maquinaria es capaz de contribuir al desarrollo sostenible:



- **ODS 7: Energía asequible y no contaminante.** Con este ODS se busca garantizar el acceso a una energía asequible, fiable, sostenible y moderna para todos. Una de las metas de este objetivo es mejorar la eficiencia energética. El desarrollo de modelos de detección de anomalías en este proyecto tiene el potencial de prevenir fallos de maquinaria en centrales eléctricas y reducir el consumo energético innecesario. Si estas técnicas de monitoreo se llevan a cabo de manera automática y fiable utilizando técnicas de Deep Learning, podría haber menos fallos no detectados y, por lo tanto, menos costes de mantenimiento. Esto, a su vez, podría llevar a una reducción en el costo de la energía. Además, al mejorar la eficiencia energética, se contribuye a una producción más responsable y a una disminución de la emisión de gases contaminantes. En resumen, el proyecto contribuye a mejorar la eficiencia energética en la industria y, por lo tanto, apoyar el logro del ODS 7.
- **ODS 9: Industria, innovación e infraestructura.** Se centra en mejorar la infraestructura, fomentar la innovación y promover una industrialización sostenible. Este proyecto puede contribuir a la mejora de la eficiencia y la seguridad en los sistemas industriales, lo que a su vez puede ayudar a alcanzar este objetivo. Al desarrollar modelos de detección de anomalías en maquinaria, se promueve la innovación en el monitoreo y mantenimiento de equipos industriales, lo cual es fundamental para la mejora de la infraestructura industrial. La implementación de técnicas avanzadas de Deep Learning permite un monitoreo automático y fiable de los equipos, facilitando la adopción de un enfoque proactivo en el mantenimiento. Además, al mejorar la seguridad de los sistemas industriales, se minimizan los riesgos para los trabajadores y se promueve un entorno laboral más seguro. La optimización de la infraestructura y la adopción de tecnologías innovadoras fomentan la competitividad de las industrias, impulsando el crecimiento económico y la creación de empleo sostenible.
- **ODS 12: Producción y consumo responsables.** Tiene como objetivo promover un consumo y producción sostenibles, lo cual implica la utilización eficiente de los recursos y la reducción de los residuos y la contaminación. La detección de anomalías en las máquinas rotativas puede contribuir a reducir los residuos y el

consumo excesivo de recursos al prevenir fallos y aumentar la vida útil de los equipos. Esto se alinea con el propósito de promover una producción más responsable y sostenible, en línea con el ODS 12.

En resumen, este proyecto se alinea con múltiples ODS de las Naciones Unidas. A través de la implementación de modelos de detección de anomalías para maquinaria rotativa, se busca mejorar la eficiencia energética, promover la innovación y la industrialización sostenible, y fomentar una producción y consumo más responsables. Estas acciones contribuirán al logro de los ODS 7, 9 y 12, y a la búsqueda de un desarrollo más sostenible y equitativo.

## ANEXO II: GLOSARIO DE TÉRMINOS

En este apartado, se presenta un glosario de términos clave utilizados en el contexto de este trabajo. El objetivo principal es proporcionar una referencia rápida y concisa de los conceptos fundamentales relacionados con los temas abordados en el trabajo. Este glosario tiene como propósito facilitar la comprensión del contenido y promover una mayor claridad en la exposición de los conceptos fundamentales. A continuación, se presenta una lista de términos relevantes acompañados de sus respectivas definiciones, lo que permitirá a los lectores familiarizarse con el vocabulario técnico y los conceptos necesarios para una comprensión más profunda del trabajo realizado.

- **ML (Machine Learning):** Aprendizaje automático. Campo dentro de la inteligencia artificial enfocado en enseñar a las máquinas a aprender de los datos y mejorar con la experiencia.
- **DL (Deep Learning):** Aprendizaje profundo. Enfoque de aprendizaje automático que utiliza redes neuronales profundas.
- **CNN (Convolutional Neural Networks):** Redes neuronales convolucionales. Arquitectura de redes neuronales que utiliza operaciones de convolución para procesar datos con una estructura espacial, como imágenes o señales.
- **RL (Reinforcement Learning):** Aprendizaje por refuerzo. Enfoque de aprendizaje automático en el que un agente aprende a través de la interacción con un entorno mediante la toma de acciones y la recepción de recompensas.
- **DRL (Deep Reinforcement Learning):** Aprendizaje por refuerzo profundo. Técnica de aprendizaje por refuerzo que hace uso de redes neuronales profundas.
- **SVM (Support Vector Machine):** Máquina de vectores de soporte. Algoritmo de aprendizaje supervisado basado en encontrar un hiperplano óptimo que separe los datos en diferentes clases.
- **Random Forest:** Algoritmo de aprendizaje automático que combina múltiples árboles de decisión para realizar tareas de clasificación o regresión.

- **K-NN (K-Nearest Neighbours):** K-vecinos más cercanos. Algoritmo utilizado para la clasificación y regresión basado en la idea de que los ejemplos similares se agrupan juntos.
- **RNN (Recurrent Neural Networks):** Redes neuronales recurrentes. Tipo de redes neuronales capaces de procesar datos secuenciales.
- **LSTM (Long Short Term Memory):** Variante de las redes neuronales recurrentes que pueden aprender y recordar patrones a largo plazo en secuencias de datos.
- **GRU (Gated Recurrent Units):** Unidades recurrentes con compuertas. Alternativa a las LSTM para modelar dependencias a largo plazo en redes neuronales recurrentes.
- **CNN-RL:** Aprendizaje por refuerzo con redes neuronales convolucionales.
- **GRU-RL:** Aprendizaje por refuerzo con unidades recurrentes con compuertas.
- **DDQN (Double Deep Q-Network):** Mejora del algoritmo de aprendizaje por refuerzo conocido como Q-learning, que utiliza dos redes neuronales para estimar y actualizar los valores de acción en un entorno de aprendizaje por refuerzo. Esta técnica ayuda a mejorar la estabilidad y el rendimiento del aprendizaje.
- **CART (Classification and Regression Trees):** Árboles de clasificación y regresión. Algoritmo que construye un árbol de decisiones para realizar tareas de clasificación y regresión. Cada nodo del árbol representa una característica y cada rama representa una decisión basada en esa característica.
- **Boosted Trees:** Árboles mejorados. Algoritmo que combina múltiples árboles de decisión débiles para formar un modelo fuerte. Cada árbol se construye secuencialmente, corrigiendo los errores cometidos por los árboles anteriores.
- **Extreme Learning Machine (ELM):** Máquina de aprendizaje extremo. Algoritmo de aprendizaje automático que utiliza una sola capa oculta de neuronas para realizar tareas de clasificación y regresión. La capa oculta se inicializa de manera aleatoria y los pesos se ajustan mediante una solución analítica.

## ANEXO III: REPOSITORIO DE CÓDIGO

### 7.1 INTRODUCCIÓN AL REPOSITORIO

Se ha facilitado un repositorio en GitHub que contiene el código realizado en este trabajo en el siguiente enlace: [https://github.com/carlotala/TFG\\_GITT\\_project\\_code](https://github.com/carlotala/TFG_GITT_project_code).

El propósito del repositorio es proporcionar acceso al código fuente necesario para replicar y extender el trabajo realizado en la realización de este TFG. Además, facilita el acceso a los datos utilizados y a los modelos entrenados para su uso en la detección de anomalías en maquinaria rotativa.

### 7.2 ESTRUCTURA DEL REPOSITORIO

El repositorio incluye un archivo “README” que proporciona una explicación detallada sobre cada uno de los archivos que incluye, así como instrucciones paso a paso sobre cómo utilizarlos. Además, se encontrará el enlace para descargar los datos necesarios para ejecutar el código de este trabajo. A continuación, se hace una breve explicación de los principales archivos de código fuente que se encuentran en el repositorio:

- **"load\_data.ipynb"**: Este archivo de notebook de Jupyter tiene la función de cargar y preprocesar los datos para su uso en los modelos de clasificación. Lee los datos del directorio ‘/data/raw’ y almacena los ficheros procesados en el directorio ‘/data/processed’.
- **"CNN\_Model.ipynb"**: En este notebook se implementa y entrena el modelo de Red Neuronal Convolutiva (CNN) para la clasificación de fallos en maquinaria, y se proporciona un análisis de los resultados.
- **"RL\_CNN\_Model.ipynb"**: Este notebook implementa y entrena el modelo de Doble DQN para la clasificación, y proporciona un análisis de los resultados.

- **"join\_models.ipynb"**: Aquí se combinan las predicciones de los modelos CNN y DDQN para crear un modelo de clasificación más robusto, y se ofrece un análisis de los resultados del modelo resultante.

Asimismo, se incluye un archivo "requirements.txt" que enumera todas las dependencias necesarias para ejecutar el proyecto. Es esencial asegurarse de tener estas dependencias instaladas en el entorno de desarrollo antes de ejecutar el código. También se proporciona una carpeta con los modelos entrenados disponibles como ficheros '.h5'.

Para utilizar este código, se recomienda clonar el repositorio en la máquina local y abrir los notebooks de Jupyter en un entorno Python con las bibliotecas requeridas instaladas. En cada archivo se encuentran las instrucciones para cargar los datos, entrenar los modelos y analizar los resultados.