



Aplicación de modelos de aprendizaje supervisados para la prevención sobre fallos de maquinaria

Angel Jian Pan Celestino, Kevin Raul Guillen Bravo,
Jorge Luis Roca Becerra

Fecha de recepción: 4 de febrero, 2023

Fecha de aprobación: 30 de marzo, 2023

Fecha de publicación: 1 de abril, 2023

Como citar: Pan, A., Guillen K. & Roca J. (2023). Aplicación de modelos de aprendizaje supervisados para la prevención sobre fallos de maquinaria. *UCV HACER*, 12(2), 9-17. <https://doi.org/10.18050/revucvhacer.v12n2a1>

Derechos de reproducción: Este es un artículo en acceso abierto distribuido bajo la licencia CC



UCV 
HACER

<http://revistas.ucv.edu.pe/index.php/ucv-hacer/index>

Aplicación de modelos de aprendizaje supervisados para la prevención sobre fallos de maquinaria

Application of supervised learning models for the prevention of machinery failures

Angel Jian Pan Celestino¹
Kevin Raul Guillen Bravo²
Jorge Luis Roca Becerra³

Resumen

Este artículo trata sobre la detección temprana de fallas en maquinarias en la Gerencia de Operaciones. Se discute la utilidad de la detección de fallas y cómo el aprendizaje automático, una técnica de inteligencia artificial, puede ser utilizado para analizar datos de monitoreo de la máquina y detectar patrones y señales que indican una posible falla en el futuro. El objetivo de la investigación es validar los modelos de Machine Learning para la predicción de fallas en maquinarias. Se describen algunos de los algoritmos de aprendizaje automático más comunes, como Support Vector Machine (SVM), Random Forest, CatBoost y XGBoost. Se realiza una comparación de métricas en los modelos para revisar cuál de ellos puede predecir de mejor manera la detección de una falla en máquinas.

Palabras clave: mantenimiento predictivo, aprendizaje automático, aprendizaje supervisado

Abstract

This article deals with the early detection of machinery failures in Operations Management. The utility of fault detection is discussed as how machine learning, an artificial intelligence technique, can be used to analyze machine monitoring data and detect patterns and signals that indicate a possible failure in the future. The objective of the research is to validate the Machine Learning models for the prediction of failures in machinery. Some of the most common machine learning algorithms are described, such as Support Vector Machine (SVM), Random Forest, CatBoost, and XGBoost. A comparison of metrics is made in the models to review which of them can better predict the detection of a failure in machines.

Keywords: predictive maintenance, machine learning, supervised learning

¹ Universidad Nacional Mayor de San Marcos (Perú). ORCID: <https://orcid.org/0000-0002-9805-4925>

² Universidad Nacional Mayor de San Marcos (Perú). ORCID: <https://orcid.org/0000-0001-5946-8925>

³ Universidad Nacional Mayor de San Marcos (Perú). correo. jroca@unmsm.edu.pe ORCID: <https://orcid.org/0000-0002-7408-7212>

INTRODUCCIÓN

En Gerencia de Operaciones, la detección de fallos en maquinarias es útil porque nos permitirá saber cuándo se le debe otorgar mantenimiento o un cambio total con anticipación, a fin de prever cualquier contingencia que pueda afectar la productividad y los costos de los procesos productivos. El uso de técnicas o modelos de Machine Learning, específicamente el aprendizaje automático ha tomado mayor protagonismo en los últimos años debido a su potencial para solucionar problemas complejos y la toma de decisiones dentro de una organización.

Según Huddleston y Brown (2018), el aprendizaje automático se basa en algoritmos matemáticos y estadísticos que sirven para analizar grandes cantidades de datos y entender ciertos patrones que se pueden dar, permitiendo aprender a la computadora de forma autónoma y mejorar su rendimiento en el tiempo. Detectar fallas tempranas de máquinas con estos modelos pueden ser mediante el monitoreo de los datos que brinda la máquina para detectar señales que nos puedan indicar una falla futura.

Según Susto et al. (2015), existen tres tipos de mantenimiento: correctivo, preventivo y predictivo. En el mantenimiento correctivo la falla ya ha ocurrido, por lo que se realizan acciones para solucionar el problema existente. El mantenimiento preventivo es el que se realiza de manera planificada y periódica, donde el plan es prevenir los posibles fallos. Por último, con el mantenimiento predictivo se usan técnicas de monitoreo y análisis que sirven de indicadores para predecir una posible falla en las máquinas.

El uso de estas técnicas permite detectar averías o fallos en etapas tempranas y reduce la frecuencia de mantenimientos como se podría realizar en los procesos preventivos. Además, para maquinarias de alta importancia, donde los repuestos puedan ser difíciles de conseguirse en caso de fallas, el mantenimiento predictivo ayudará a prever los posibles altos costos, minimizando los tiempos de

inactividad operaria y mejorando el rendimiento efectivo. Así también, cualquier tipo de estrategia de mantenimiento puede ayudar a minimizar los riesgos y las condiciones del equipo (Çınar et. al, 2020).

El objetivo de esta investigación es validar los modelos de Machine Learning para la predicción de fallas en maquinarias que serán útiles para que el gerente de Operaciones pueda tomar decisiones al respecto con los indicadores obtenidos, realizando el mantenimiento correspondiente.

Machine Learning

Machine Learning no es solo un problema de base de datos; también es una parte de inteligencia artificial. Para ser inteligente, un sistema que está en proceso de cambio de ambiente debe tener la capacidad de aprender. Si el sistema puede aprender y adaptarse a tales cambios, el diseñador del sistema no necesita prever y proporcionar soluciones para todas las situaciones posibles (Alpaydin, 2010).

Aprendizaje supervisado

Según Alpaydin (2010), el aprendizaje supervisado es una técnica de aprendizaje automático en la que se entrena un modelo en datos etiquetados, con el objetivo de predecir la etiqueta correspondiente para nuevos datos.

Support Vector Machine (SVM)

SVM funciona correlacionando datos a un espacio de características de grandes dimensiones de forma que los puntos de datos se puedan categorizar, incluso si los datos no se puedan separar linealmente de otro modo. Se detecta un separador entre las categorías y los datos se transforman de forma que el separador se puede extraer como un hiperplano. Tras ello, las características de los nuevos datos se pueden utilizar para predecir el grupo al que pertenece el nuevo registro (IBM, 2021)

Random Forest

Random Forest es un método de aprendizaje automático basado en la técnica de ensamblaje de árboles de decisión. Consiste en construir muchos árboles de decisión de manera aleatoria y promediar sus resultados para mejorar la precisión y estabilidad del modelo (Breiman, 2001).

CatBoost

Para Prokhorenkova et al. (2018), CatBoost es un algoritmo de aprendizaje automático que combina la eficacia de la técnica de gradient boosting con la capacidad de tratar variables categóricas sin codificación previa.

XGBoost

XGBoost (eXtreme Gradient Boosting) es un algoritmo de aprendizaje automático de tipo ensamblado que se utiliza comúnmente para problemas de clasificación y regresión. Se basa en la idea de construir un modelo complejo a partir de una secuencia de modelos más simples y se ha demostrado ser uno de los algoritmos de aprendizaje automático más efectivos en la actualidad (Chen, 2016).

Optimización de hiperparámetros

Gerón (2017) menciona que la optimización o ajuste de hiperparámetros se refiere al proceso de elección de los valores óptimos en un modelo de machine learning para que este tenga un mejor rendimiento predictivo y de aprendizaje. Estos valores se asignan antes de la ejecución del entrenamiento para así calcular y otorgar los mejores parámetros al modelo.

Métricas de modelos de machine learning

Las métricas tienen un factor muy importante a la hora de evaluar los modelos. Una primera es la matriz de confusión, que conlleva a sacar otras más y que contiene filas y columnas que representan a la clase predicha y real (Borja et. al, 2020)

La matriz de confusión es una herramienta que sirve para medir el grado de precisión de los modelos. Se encuentran valores verdaderos negativos (TN), falsos positivos (FP), falsos negativos (FN) y verdaderos positivos (TP).

Figura 1.

Matriz de confusión

Matriz de confusión		Resultado de predicción	
		Negativo	Positivo
Valor real	Negativo	TN	FP
	Positivo	FN	TP

A partir de estos resultados se obtienen las métricas siguientes:

- Exactitud: Mide a los elementos predichos o clasificados correctamente respecto al total de datos.

$$\text{accuracy} = \frac{\text{TN} + \text{TP}}{\text{TN} + \text{FP} + \text{FN} + \text{TP}}$$

- Sensibilidad: Muestra la cantidad de verdaderos positivos respecto a todos los valores positivos reales.

$$\text{recall} = \frac{\text{TP}}{\text{FN} + \text{TP}}$$

- Precisión: Calcula el porcentaje de predicciones positivas hechas correctamente.

$$\text{precision} = \frac{\text{TP}}{\text{FP} + \text{TP}}$$

- Puntuación F1: Combinación de métricas de sensibilidad y precisión.

$$F1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

- Especificidad: Porcentaje de casos negativos detectados.

$$\text{specificity} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

METODOLOGÍA

La presente investigación se realizó aplicando los modelos de aprendizaje automático dentro del machine learning. Se plantea evaluar y comparar los métodos muy conocidos y aplicados dentro de la Ciencia de Datos. Estos son SVM, Random Forest, XGBClassifier y CatBoost con librerías que pueden implementarse en el lenguaje de programación Python con la herramienta de Jupyter Notebook.

La data obtenida ha sido extraída del repositorio Centro de Aprendizaje Automático y Sistemas Inteligentes (UCI - 2020), un sitio web donde se pueden usar datasets sintéticos o basados en la realidad para la aplicación de modelos de machine learning.

A su vez, este conjunto de datos provienen de Stephan Matzka, donde detalla lo siguiente:

Este documento presenta y proporciona un conjunto de datos de mantenimiento predictivo realista, pero sintético, para uso en este documento y por la comunidad. Se describen un modelo explicativo y una interfaz explicativa, se entrenan con el conjunto de datos y se evalúa y compara su rendimiento explicativo (2020).

El dataset contiene 10000 filas y los siguientes campos:

- UDI: el identificador único y de conteo.
- Product ID: el identificador del producto, comenzando por la letra L, M o H.
- Type: Contiene las letras L, M y H en referencia al anterior campo.

- Air temperature [K]: generada usando un proceso de caminata aleatoria luego normalizada a una desviación estándar de 2 K alrededor de 300 K.

- Process temperature [K]: generada utilizando un proceso de caminata aleatoria normalizado a una desviación estándar de 1 K, sumada a la temperatura del aire más 10 K.

- Rotational speed [rpm]: calculada a partir de una potencia de 2860 W, superpuesta con un ruido normalmente distribuido.

- Torque [Nm]: los valores de par se distribuyen normalmente alrededor de 40 Nm con un $\dot{I}f = 10$ Nm y sin valores negativos.

- Tool wear [min]: Las variantes de calidad H/M/L añaden 5/3/2 minutos de desgaste de herramienta a la herramienta usada en el proceso.

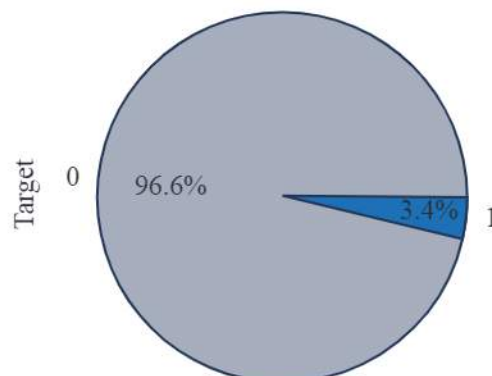
- Target: indica si la máquina ha fallado en este punto de datos en particular para cualquiera de los siguientes modos de falla es verdadera.

- Failure Type: indica el tipo de falla que se dio.

Adicionalmente, se entiende que, dentro una empresa que opera con máquinas, por lo general mantiene un mayor porcentaje de no fallos. Es el caso de este conjunto de datos, donde la distribución es desproporcionada para las intenciones de aplicación en este trabajo.

Figura 2.

Proporción de casos de fallo y no fallo de máquinas



El desequilibrio de datos se da cuando se tiene en la muestra muchos datos de una clase y pocos de otra, por lo que el problema debería ser solucionado con un resampling, ya sea un undersampling u oversampling (López, 2021). No obstante, dentro de la librería de imblearn hay formas de codificar la cantidad de datos para realizar un submuestreo o sobremuestreo. En esta ocasión, para no perder los datos importantes de los casos de no fallos, se procede a realizar el segundo proceso.

RESULTADOS

Se presentan los resultados obtenidos de los modelos implementados a fin de realizar una comparación de métricas entre ellos.

Support Vector Machine (SVM)

Figura 3.

Matriz de confusión - SVM

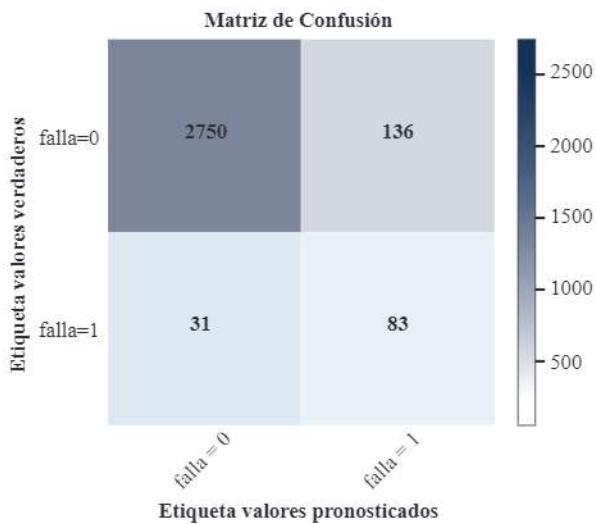


Figura 4.

Matriz de confusión - Random Forest

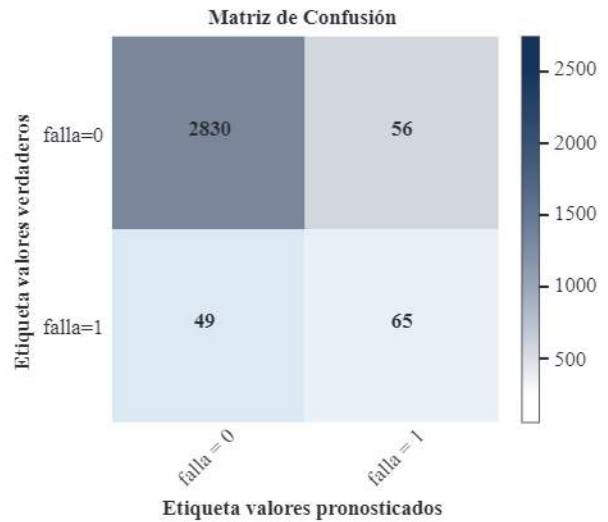


Figura 5.

Matriz de confusión - CatBoost

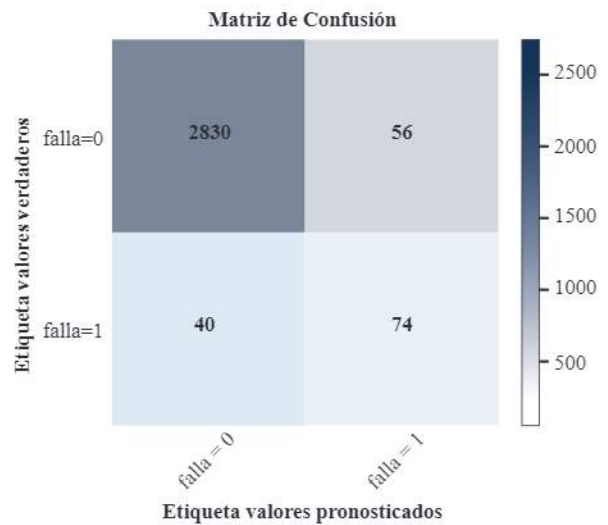
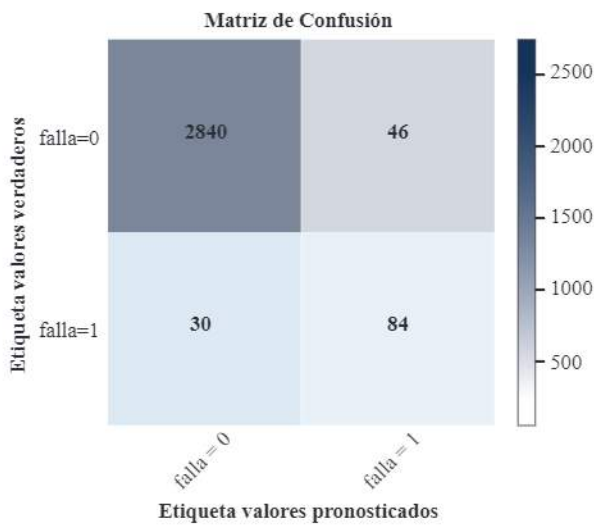
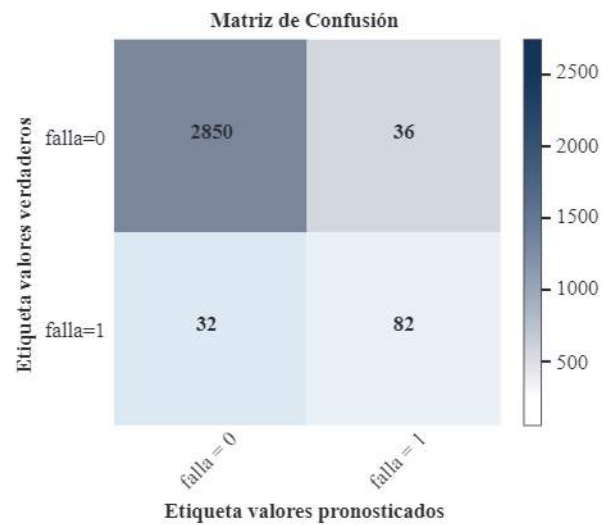


Figura 6.*Matriz de confusión – XGBoost***Figura 7.***Matriz de confusión - XGBoost Calibrado***Tabla 1.***Métricas y resultados de los modelos de Machine Learning utilizados*

Modelo	precision_score	recall_score	accuracy_score	f1_score
SVM	0.47	0.46	0.96	0.46
Random Forest	0.54	0.57	0.96	0.55
CatBoost	0.57	0.65	0.96	0.61
XGBoost	0.64	0.73	0.97	0.69
XGBoost calibrado	0.70	0.72	0.98	0.71

DISCUSIÓN DE RESULTADOS

El uso de sensores o diversas técnicas de extracción y de análisis de datos en tiempo real de las máquinas registrado a elementos computacionales servirá no solo para predecir si fallará o no, sino también para mejorar el modelo de predicción con mayor cantidad de información relevante en la detección temprana de anomalías. Importante también con la extensión de detección de tipo de fallas, que permitirá al equipo y área de Operaciones tener una mejor noción y diagnóstico sobre la situación y estado de las máquinas para tomar decisiones al respecto, ya que se sabe que los costos

en etapas tardías y compra de nuevos repuestos pueden ser elevados a comparación de realizar la correcta prevención o predicción para el mantenimiento de los distintos tipos de máquinas relevantes en la gestión y operación de una empresa.

Como se puede notar en la tabla 1, el modelo de Support Vector Machine no logra sacar los mejores resultados a nivel de predicción general. No obstante, cabe recalcar que el número de máquinas predichas que tendrán un fallo es el más alto con 83 aciertos.

Por otro lado, con la implementación del Random Forest se notó una mejora en las métricas, reduciendo sus aciertos a nivel de máquinas con falla, pero aumentando la cantidad de máquinas sin falla correctamente predichas.

Los modelos que mejores resultados entregaron fueron los de CatBoost y XGBoost, ambos por métodos de clasificación.

Por último, se realizó una optimización de hiperparámetros del XGBoost classifier, donde se calibraron e iterar 5 veces por cada validación para obtener mejores resultados con precision_score de 70%, recall_score de 72%, f1_score de 71% y un accuracy_score de 98% que valida un gran porcentaje de predicción con respecto a toda la data de prueba.

Estos resultados se pueden extender a la mejora de los modelos con mayor cantidad de información histórica que se va recopilando a lo largo del tiempo en tiempo real sobre el estado de las máquinas. En adición, se puede también aplicar otros tipos de métodos e incluso investigar más campos de la prospección como el uso del Deep Learning y redes neuronales.

CONCLUSIONES

- La actual investigación plantea la aplicación del aprendizaje supervisado, dentro del Machine Learning, para la predicción de fallos o anomalías en maquinarias mediante el uso de datos históricos y modelos para la gestión del mantenimiento predictivo dentro de la Gerencia de Operaciones.
- Se realizó un benchmarking o comparativa de los modelos utilizados, siendo el SVM el que mayor cantidad de fallas efectivas predijo, pero teniendo en cuenta que no se obtuvo el mejor resultado a nivel de precisión general. Los modelos con mejor rendimiento fueron el CatBoost y XGBoost, que redujeron los falsos positivos y falsos negativos.

- A la hora de optimizar los hiperparámetros del XGBoost, se notó una mejoría en la predicción, llegando a predecir de mejor manera las máquinas que fallarán (82) y las que no (2850). Sin embargo, como todo modelo no puede ser perfecto, aún se siguieron presentando 36 falsos positivos y 32 falsos negativos.

- Se puede notar que estos modelos de aprendizaje supervisado, que reciben datos etiquetados de entrada, son útiles y pueden dar información relevante y en tiempo real sobre si una máquina tiene una probabilidad de fallar o no.

- Para futuras investigaciones se puede ampliar la cantidad y el tipo de modelos a utilizar dentro del Machine Learning e incluso explorar áreas del Deep Learning como las redes neuronales. Además, servirá para la predicción no tan solo de fallas en maquinarias, sino también centrándose en el tipo de falla que brindará mucho mayor información al área de Operaciones para tomar una decisión más concreta y eficaz.

REFERENCIAS

Alpaydin, E. (2010). *Introduction to machine learning* (2nd ed.). Cambridge, MA: MIT Press.

Borja Robalino, R., Monleón Getino, A., & Rodellar, J. (2020). Estandarización de métricas de rendimiento para clasificadores Machine y Deep Learning. *Revista Ibérica de Sistemas e Tecnologías de Informação*, 30, 184-196.

Breiman, L. (2001) Random Forests. *Machine Learning*, 45, 5-32. <http://dx.doi.org/10.1023/A:1010933404324>

Centro de Aprendizaje Automático y Sistemas Inteligentes (2020). *Machine Learning Repository*. <https://archive.ics.uci.edu/ml/index.php>

- Chen, T. (2016). Xgboost: *A scalable tree boosting system*. In Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining (pp. 785-794). ACM.
<https://doi.org/10.1145/2939672.2939785>
- Çımar, Z. M., Abdussalam Nuhu, A., Zeeshan, Q., Korhan, O., Asmael, M., & Safaei, B. (2020). Machine Learning in Predictive Maintenance towards Sustainable Smart Manufacturing in Industry 4.0. (19), 8211. MDPI AG.
<http://dx.doi.org/10.3390/su12198211>
- Géron, A. (2017). *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*. O'Reilly Media, Inc.
- Huddleston, S. H., & Brown, G. G. (2018). *Machine learning*. In *Informatics Body of Knowledge* (1st ed., Vol. 38).
<https://doi.org/10.1002/9781119505914.ch7>
- IBM. (2021). *Support Vector Machines (SVM)*. IBM Developer. <https://developer.ibm.com/technologies/machine-learning/tutorials/support-vector-machines-svm>
- López Camuñas, J. M. (2021). *Predictive maintenance using deep learning* (Bachelor's thesis). Universitat de Barcelona. <https://hdl.handle.net/2445/182392>
- Matzka, S. (2020). Explainable Artificial Intelligence for Predictive Maintenance Applications. In *2020 Third International Conference on Artificial Intelligence for Industries (AI4I)* (pp. 69-74). IEEE.
<https://doi.org/10.1109/AI4I49448.2020.00023>
- Prokhorenkova, L., Gusev, G., Vorobev, A., & Gulin, A. (2018). *CatBoost: gradient boosting with categorical features support*. <https://arxiv.org/abs/1706.09516>
- Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., & - Beghi, A. (2015). Machine Learning for Predictive Maintenance: A Multiple Classifiers Approach. *IEEE Transactions on Industrial Informatics*, 11(3), 812-820. <https://ieeexplore.ieee.org/document/6879441>