



# EVALUACIÓN DE ALGORITMOS DE INTELIGENCIA ARTIFICIAL APLICADOS AL MANTENIMIENTO PREDICTIVO.

## (Mayo de 2022)

B. F. Falla y D. A. Ortega

I. INTRODUCCIÓN

**Resumen -** Cualquier tipo de mantenimiento es la acción más importante que se realiza en la factoría, el preventivo, correctivo y predictivo son los tipos clásicos, los dos primeros son los más implementados casi en todas las empresas, con un grado menor de implantación, pero no menos importe está el mantenimiento predictivo, este tiene un alto costo debido a que se realiza con equipos de alto costo y con personal calificado.

En la última década en empresas grandes ha tenido un auge bastante importante el mantenimiento predictivo, se ha apoyado en herramientas tecnológicas. Por otro lado, la inteligencia artificial también está apoyando diversos procesos en diferentes áreas y el mantenimiento no podía ser la excepción. En este trabajo se evalúa dos algoritmos de inteligencia artificial sobre un conjunto de datos provenientes de sensores que se ubican en las máquinas.

El conjunto de datos usado en este trabajo esta desbalanceado por lo que es importante buscar una estrategia de balance, para resolver este problema se usa algoritmos de vecinos cercanos para generar una data que no afecte de manera negativa en el entrenamiento de la red neuronal. Se consigue un recall de 0.86 y una precisión alrededor de 0.95.

**Índice de Términos – Algoritmos de inteligencia artificial, redes neuronales, mantenimiento predictivo.**

1

---

<sup>1</sup> Documento recibido el 20 de mayo de 2022. Este artículo fue apoyado del trabajo de grado titulado EVALUACION DE ALGORITMOS DE INTELIGENCIA ARTIFICIAL APLICADO AL MANTENIMIENTO PREDICTIVO

B. F. Falla, Facultad de ingeniería programa de electrónica, AUNAR, Pasto – Colombia (correspondiente al autor - Cel: 3145390751; e-mail: pinpiff@gmail.com ).

D. A. Ortega, Facultad de ingeniería programa de electrónica, AUNAR, Pasto – Colombia, (correspondiente al autor - Cel: 3176023711 ; e-mail: diilmerortega@gmail.com).

La industria busca transformar las materias primas en productos que satisfacen las necesidades de personas, en el proceso de transformación de la materia se ve involucrados diversos dispositivos mecánicos, electrónicos y otros. Estos equipos por estar sometidos a diversas variables físicas y por su construcción misma tienden a presentar fallas, estas obligan a detener el proceso causando pérdidas monetarias e incluso comprometiendo la vida de operarios, para mitigar este efecto los fabricantes recomiendan realizar mantenimiento con regularidad.

En su mayoría, los fabricantes establecen la frecuencia y protocolos para realizar el mantenimiento preventivo con el cual se logra disminuir en un alto porcentaje las fallas repentinas de la maquinaria. A pesar de cumplir lo estipulado en la documentación técnica del equipo este llega a fallar es aquí donde entra a operar el mantenimiento correctivo que su objetivo es poner en funcionamiento el dispositivo para continuar con el proceso.

Con el mantenimiento preventivo y correctivo se evita y corrige fallas, pero deja del lado el pronóstico hacia un futuro cercano o inmediato de la , es decir, se desconoce el tiempo o ciclo aproximado que una parte de la maquinaria fallará o también cuando la falla está en la etapa inicial, como una posible respuesta a estos acontecimientos surge el mantenimiento predictivo que se encarga de evaluar el estado de las partes, y así tener un posible ciclo de falla lo que se puede traducir a una “fecha”.

Realizar mantenimiento predictivo es complejo debido al análisis de diversas variables, además se debe contar con personal altamente calificado. Por otro lado, el avance tecnológico especialmente en la sensorica ha facilitado la obtención de información sobre el estado de los equipos sin detener la marcha para medir cierta variable física con un alto grado de precisión, pero también a complejizado su análisis

debido a la gran cantidad de datos que se suministrados a lo largo del tiempo.

Como un apoyo para el análisis de grandes volúmenes de datos suministrados por los sensores sobre el estado de la máquina, en este trabajo se entrena dos algoritmos de inteligencia artificial para detectar si un dispositivo (maquina) está operando con normalidad o por el contrario presenta falla. Se usa el dataset publicado en el repositorio UCI, la metodología es cuasi experimental debido a que no hay grupos de control, pero si presenta las características de una investigación experimental.

## II. PROCEDIMIENTO DEL DESARROLLO DEL TRABAJO

El mantenimiento predictivo tiene como objetivo evitar las paradas innecesarias de los equipos, esto se realiza evaluando a través de diversas técnicas el estado de los componentes de una máquina con el fin determinar la vida útil en el mejor de los casos. Idealmente se busca que las maquinas operen dentro de sus parámetros establecidos, pero si por algún motivo los valores exceden los límites normales esta debe detener su marcha, para que personal calificado realice una inspección donde se detecte y se corrija el desperfecto, con esto se garantiza que la maquina no sufra daños graves y por ende minimiza costo operación.

En el escenario que las variables físicas estén por fuera del rango de operación se podría pensar que la acción de control emitida por el controlador corrija este desperfecto, en el mejor de los casos esto se suele corregir, pero también puede pasar que esta acción de control empeore la situación debido a que está forzando a una respuesta.

Generalmente las fallas inician por anomalías que son imperceptibles al operario las cuales van creciendo con el tiempo hasta desencadenar en rupturas o en inoperatividad de las maquinas llevando a paradas de los procesos y disminuyendo utilidad a las empresas.

### A. Proceso a realizar

#### Análisis de los datos.

En este trabajo evalúa redes neuronales en el contexto de mantenimiento predictivo, se inicia con un *dataset* en el cual relaciona datos reales de mantenimiento predictivo encontrados en la industria, el archivo se denomina *ai4i2020.csv* y está disponible en <https://archive.ics.uci.edu/ml/datasets/AI4I+2020+Predictive+Maintenance+Dataset>.

En esta investigación se aborda el mantenimiento predictivo y algoritmos *random forest* y redes neuronales. Los algoritmos

se entrenan utilizando el 70% de los datos disponibles en el *dataset*, la implementación se realiza haciendo uso del lenguaje de programación Python, y finalmente el 30% restante de los datos se usa para evaluar los algoritmos implementados con el fin de evaluar dichos algoritmos.

#### Análisis de los datos Redes neuronales y random forest.

Las redes neuronales tienen una gran acogida en los últimos años se involucran en diferentes áreas del conocimiento la medicina, sector energético (Hurtado 2011), robótica (Aguilar 2011) y muchas otras aplicaciones. Este tipo de técnicas no son realmente nuevas, pero en los últimos años están teniendo un auge bastante considerable. A nivel comercial ya están disponibles diversas soluciones.

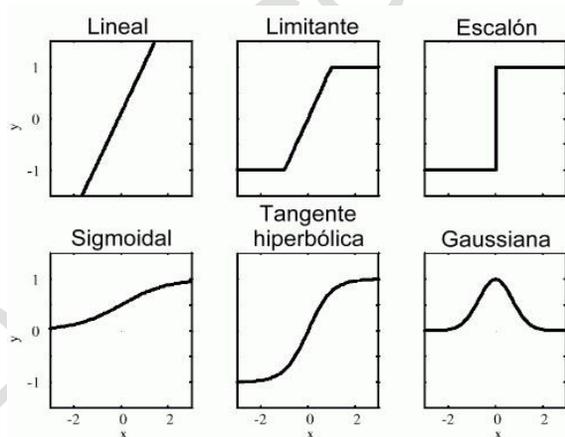


Fig. 1. Función de activación

Las redes neuronales tienen una en su salida una función que se denomina activación esta cobra real importancia en el momento de entrenamiento, esta permite que el algoritmo de convergencia encuentre el mínimo global y converja. Hay varias funciones de cómo se puede observar en la Fig 1.

Por otro lado, los bosques aleatorios son bastante utilizados en diversas aplicaciones, la literatura está bien nutrida en el uso, esto es debido a la facilidad de entrenamiento, Se basan en arboles de decisión.

En este trabajo se sigue una metodología en cascada, es decir, se cumple en fases en un orden secuencial donde el resultado de la anterior alimenta a la siguiente, por lo tanto, se debe garantizar que cada una de ellas este bien ejecutada y los resultados sean consistentes.

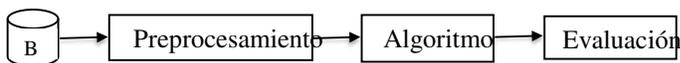


Fig. 2. Etapas del desarrollo de la investigación

En la Figura 2 se presenta las etapas que se siguen para el desarrollo de este trabajo, inicia con BD que es la base de datos, para este caso es el *dataset*, en el preprocesamiento se realiza el análisis de los datos donde se garantiza que son numéricos y están normalizados, posteriormente se entrena los algoritmos siguiendo las técnicas propuestas en la literatura vigente, finalmente se evalúa la implementación haciendo uso de los datos.

Id	Nombre Columna	Comentario
1	UID	Es el identificador de las filas
2	Producto ID	Relaciona el numero de la serie de la maquina
3	Type	Relaciona la calidad de la máquina, se indica tres posibles valores que son: L para bajo, M medio y H alto.
4	Air temperatura [k]	Está asociada a la temperatura que indica el proceso su unidad de medida es el kelvin
5	Rotation speed [rpm]	Indica la velocidad de rotación medida en revoluciones por minuto.
6	Torque [Nm]	Es información del par generado para realizar el movimiento, su unidad de medida es el Newton-metro
7	Tool wear failure (TWF)	Relaciona si el sistema ha fallado u opera con normalidad, se indica dos clases que son: Falla: indicada con el número 1. Normal: Simbolizada con el número 0.

El *dataset* es una matriz con 10 columnas que identifican las características de los datos, y 10000 columnas que son el histórico del estado del sistema mecánico, en la Tabla 2 se describe cada una de las columnas.

	A	Pr	R	T	Too				
	ir	o			I	failur	TW	H	PW
	T	T	[rpm	[N	wea	e	F	DF	F
	[K]	[K]	]	m]	r				
Promedio	3	31	1538.	39.	107.	0.039	0.00	0	0.00
	00	0	77	98	95		46	.011	95

Desviación	2	1.4	179.2	9.9	63.6	0.18	0.06	0	0.09	0.043
		8	8		5		7	.1	7	
Min	2	30	1168	3.8	0	0	0	0	0	0
	95.3	5.7								
25%	2	30	1423	33.	53	0	0	0	0	0
	98.3	8.8		2						
50%	3	31	1503	40.	108	0	0	0	0	0
	00.1	0.1		1						
75%	3	31	1612	46.	162	0	0	0	0	0
	01.5	1.1		8						
Max	3	31	2886	76.	253	1	1	1	1	1
	04.5	3.8		6						

Fig. 3. Estadística dataset.

La fig. 3 describe las estadísticas del conjunto de datos a evaluar, las primeras cinco columnas representan las variables físicas medidas a través de instrumentos, las cinco restantes relacionan la falla y cuál es el sistema que falla. Las filas muestran cómo están distribuidos los datos, se inicia por el promedio, continuando con la desviación estándar, valor mínimo y los cuartiles hasta terminar con el valor máximo.

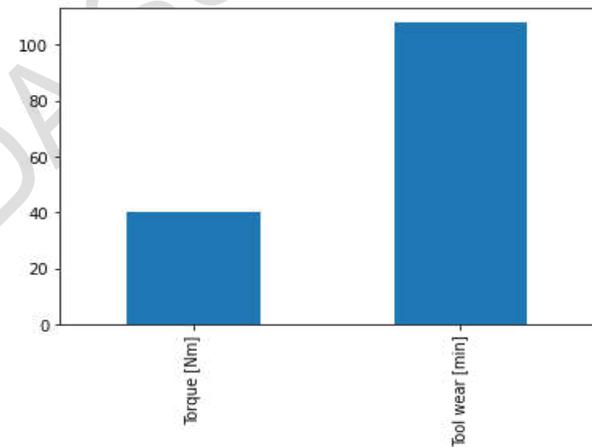


Fig. 4. Promedio de características de dataset torque y desgaste de la herramienta.

La Fig. 4 muestra el valor promedio del torque y desgaste de la herramienta, se puede observar que el valor para torque es 40 Nm y el desgaste mínimo es aproximadamente 100 unidades.

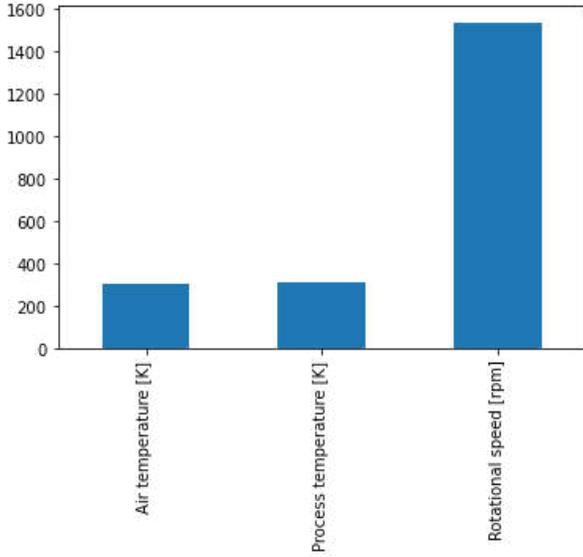


Fig. 5. Promedio de características de dataset velocidad de rotación, temperatura del aire, proceso.

La Fig. 5 presenta el promedio de las variables velocidad de rotación, temperatura del aire y temperatura del proceso; las unidades de medida son kelvin para temperatura y revoluciones por minuto para rotación, también se infiere que la temperatura del proceso y del aire están por encima de los 300 k y la velocidad de rotación tiene un valor aproximado de 1500 rpm.

Machine failure representa la falla, TWF es falla por desgaste de la herramienta, HDF es fallo de disipación de calor, PWF es fallo por potencia, OSF es fallo por desgaste de la herramienta y exceso de par, RNF es falla independiente.

Las Fig. 7 a 10, muestra la distribución de cada una de las condiciones de falla, es evidente que los datos se encuentran desbalanceados, en el mejor de los casos la condición de falla está cerca al 1.8%.

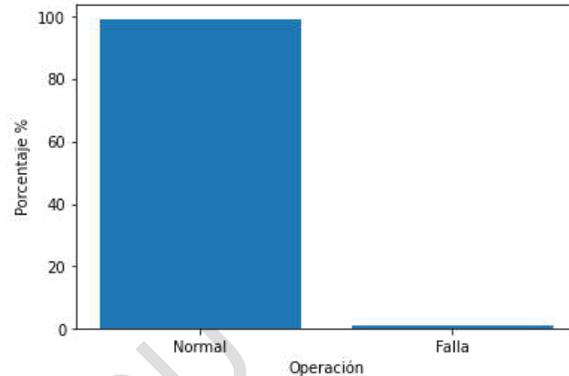


Fig. 7. Condición de falla por desgaste de la herramienta.

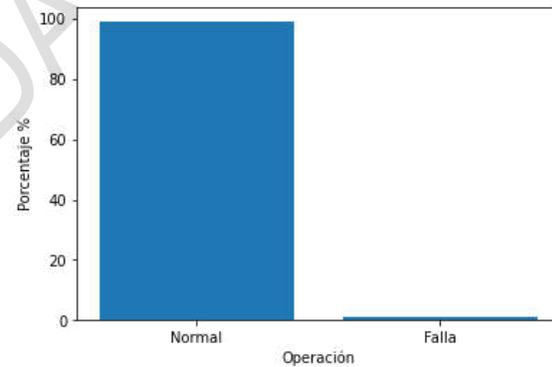


Fig. 8. Condición de falla por disipación de calor.

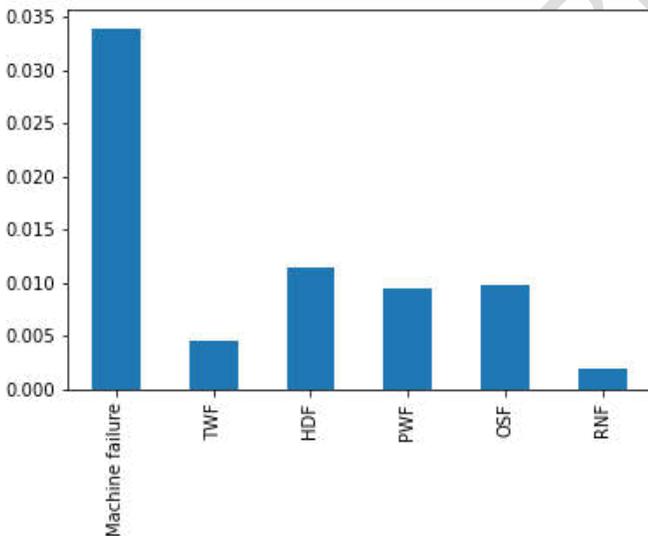


Fig. 6. Promedio de características de dataset condiciones de falla.

La Fig 6 muestra las diferentes condiciones de falla, la falla de maquina agrupa todas las posibles fallas como se puede ver tiene el promedio más alto, en general, es la suma de los valores de fallas específicas.

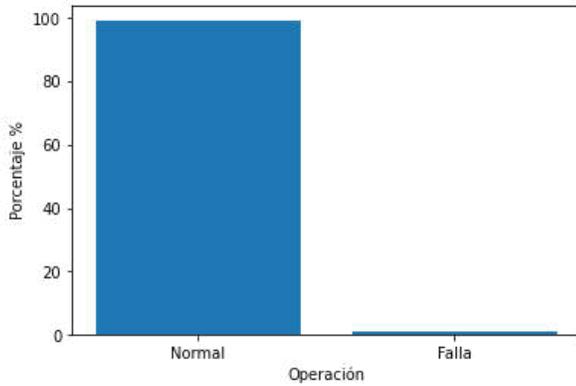


Fig. 9. Condición de falla por potencia.

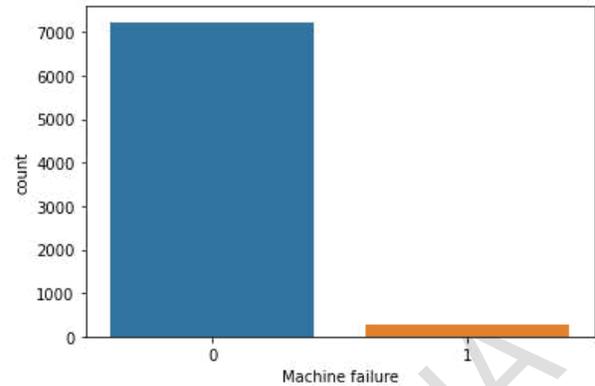


Fig. 11. Distribución de los datos para entrenamiento.

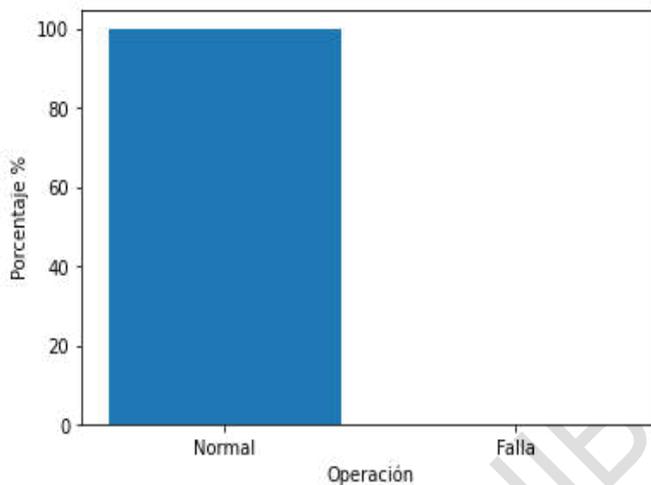


Fig. 10. Condición de falla por exceso de par y desgaste de la herramienta.

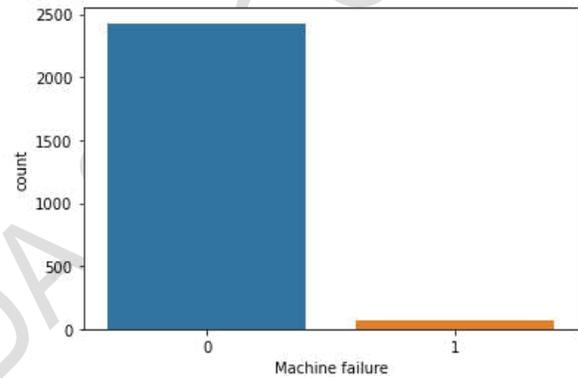


Fig. 11. Distribución de los datos para evaluación.

El Dataset que se usa en este trabajo está desbalanceado, es decir, el 97% pertenece a una clase y el restante 3% a otra clase, cabe mencionar que las clases se definen como operación normal de la máquina o falla se representa con un cero y uno respectivamente.

Dentro del procesamiento de datos se trabaja la segmentación de la información, se debe dividir un grupo de datos en dos, el primero de ellos sirve para el entrenamiento y el segundo es para la validación y test, generalmente se usa 70 % y 30 % para entrenamiento y validación respectivamente.

En las Figuras 11 y 12, se presenta la columna de condición de falla que está representada por un cero y un uno, en la imagen izquierda se observa la distribución de datos para test, que corresponde al 30% de la data total que corresponde aproximadamente a 2500 de la clase 0 y menor a 200 valores para la clase 1; por el contrario, los datos de entrenamiento son mucho más grandes.

#### Ajuste de dataset

Como es evidente los datos tienen un desbalance bastante grande, hay diversas técnicas para enfrentar este problema, por ejemplo, random oversampling esta técnica es duplicar valores de la clase menos favorecida, la principal desventaja es la posibilidad de sobreentrenamiento; random undersampling es un método que extrae información aleatoria de toda la data, generalmente se deja una relación de 1:1, pero es aceptable una relación de 1:3.

SMOTE oversampling es una técnica de sobre muestreo de minorías sintéticas genera datos a partir de la tendencia del



grupo menos favorecido, es decir, realiza un vecindario y genera grupos dentro de las fronteras.

Antes de elegir la técnica de inteligencia artificial que resolverá este trabajo, es necesario plantear el problema que trae consigo el conjunto de datos, inicialmente se tiene información proveniente de sensores que identifican variables físicas de la máquina, por otro lado, en un ciclo en específico se presenta la falla.

Analizando la condición de fallo se ve que existe una columna con información sobre el ciclo en el que se presenta, centrando el foco aquí se determina que es un sistema de dos clase, operación normal y condición de fallo. Ahora observando las columnas contiguas a la condición de fallo se observa que hay una especificación del fallo, es decir que se puede determinar que la maquina fallo y por qué fallo.

Es evidente que es un problema de clasificación, ahora también se puede analizar desde el punto de vista biclase donde se determina que ocurrió el evento o esté operando normal. También se analiza desde un sistema multiclase donde las clases están determinadas por la condición de falla de sistema.

Con base en la literatura consultada y definido el sistema como un clasificar, se procede a evaluar el tipo de entrenamiento que se usa, por ser un clasificador se elige evaluar una red neuronal, así como también implementar el algoritmo de random forest esta decisión se basa en el desbalance del dataset dado que este algoritmo es menos propenso al sobreajuste.

#### *Entrenamiento, evaluación de red neuronal y random forest*

En el desarrollo de este trabajo se asume que se tiene dos dataset para entrenamiento, el primero un conjunto de datos totalmente desbalanceado con relación 97:1 aproximadamente, y el segundo un nuevo conjunto de datos que ya está balanceado con una relación 1:1, el balance se realiza a través de generar datos sintéticos basado en la tendencia de los datos actuales.

En este trabajo se usa la librería sklearn, se deja los parámetros por defecto, salvo la inicialización de aleatoriedad con un valor de 42, esto con el fin de particionar los datos en grupos y facilitar su entrenamiento.

Para el caso de las redes neuronales se elige una red neuronal con 10 capas oculta y cada capa tiene diez neuronas, el máximo número de iteraciones es de 500, el solved usado es el propuesto por Kingma et al. Es un estimador estocástico, el parámetro de penalización está en una milésima, la generación par los datos iniciales está en 21, los demás parámetros se dejan por defecto. Con el fin de evaluar los resultados obtenidos dentro de los dos algoritmos en los dos datasets se mantendrán los parámetros intactos.

#### *Datos desbalanceados*

Algoritmo	Accuracy	Recall
RandomForest	0.98	0.54
Redes Neuronales	0.98	0.39

Tabla. 1. Accuracy y recall con datos desbalanceados.

Después de crear la red neuronal y el clasificador random forest, se procede a evaluar los datos obtenidos usando los datos de segmentados para test, en la Tabla 1 se puede apreciar los resultados de accuracy y recall para los dos algoritmos implementados. Los dos algoritmos tienen un alto porcentaje en la clasificación de verdaderos positivos, por el contrario, tienen un bajo porcentaje en la clasificación de verdaderos negativos, siendo la red neuronal la que presenta un menor porcentaje.

#### *Matriz de confusion*

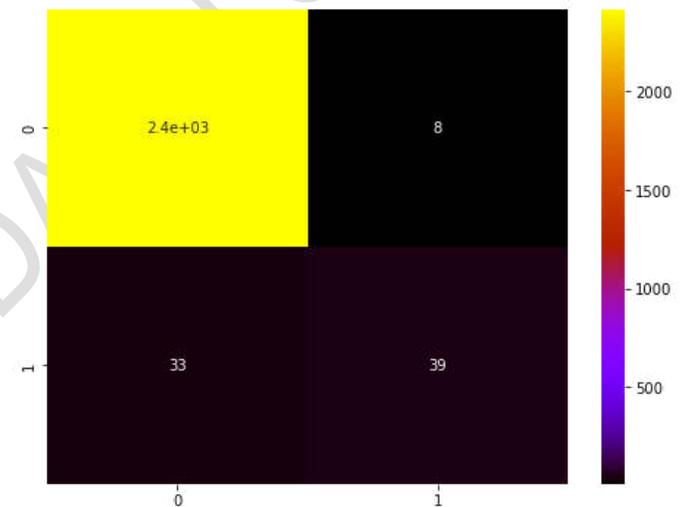


Fig. 12. Matriz de confusión random forest datos desbalanceados.

La Fig 12 muestra la matriz de confusión generada después de evaluar el algoritmo random forest, En la parte superior de la imagen se puede observar que 8 elementos están mal clasificados, es decir, el algoritmo los clasifica como fallo de la maquina cuando estos no pertenecen a este grupo. Por otro lado, la parte inferior de la imagen muestra la clasificación negativa, se observa que 33 elementos están mal clasificados.

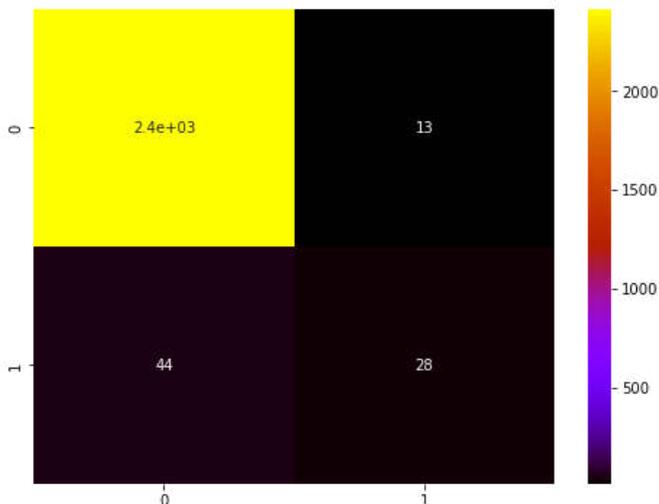


Fig. 13. Matriz de confusión redes neuronales datos desbalanceados.

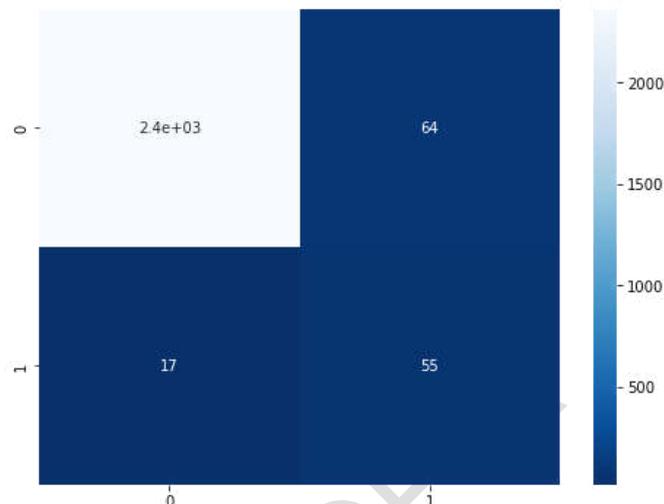


Fig. 14. Matriz de confusión random forest.

La Fig 12 muestra la matriz de confusión generada por la red neuronal después de evaluar los datos particionados para test. En la parte inferior de la imagen se muestra los casos negativos que se asemejan al fallo de la máquina, la red neuronal clasifica 44 fallos de maquina como si esta estuviera operando normalmente por lo que se denota un error de clasificación. La parte superior de la imagen muestra solo 13 casos mal clasificados.

En la Fig 14 se observa la matriz de confusión, se evidencia que 2400 datos están bien clasificados frente a 64 que su clasificación es errónea. Por otro lado, se observa que 55 elementos están bien clasificados frente a 17 que presentan error. Esta grafica es el resultado de la evaluación del algoritmo random forest.

Los resultados que se obtuvieron con los algoritmos son consecuentes con la teoría de inteligencia artificial, donde plantea que en datos desbalanceados los algoritmos no pueden generalizar los datos de entrada, los algoritmos entrenados son muy buenos clasificando cuando la maquina funciona bien, pero presenta deficiencias al momento de evaluar anomalías.

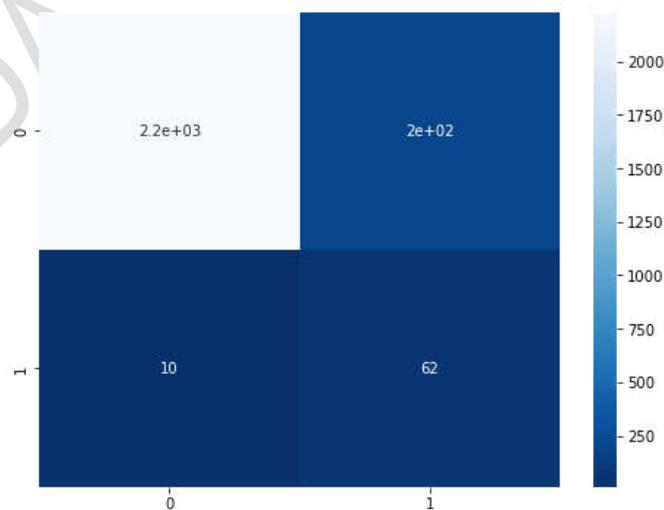


Fig. 15. Matriz de confusión red neuronal.

*Conjunto de datos balanceados*

Algoritmo	Accuracy	Recall
RandomForest	0.97	0.76
Redes Neuronales	0.92	0.86

Tabla2. Resultados de algoritmos acuracy y recall.

Observando la Tabla 2 es evidente que la clasificación de los casos negativos está por encima de un 70%, siendo las redes neuronales más precisas llagando al 86 %, pero tienen una precisión del 92 % levemente menor que la que tiene random forest, en general para este trabajo se considera resultados aceptables.

Centrando la vista en la clasificación de los verdaderos negativos generada por la red neuronal se ve que 62 están bien clasificados y 10 presentan una mala clasificación, por otro lado, un alto número de verdaderos positivos son clasificados como se observa en la Fig 15.

*Conjunto de datos balanceados*

El *dataset* permite hacer análisis multiclase, es decir, el algoritmo debe determinar cuál de las condiciones falla



mostradas en la Fig 3 se presenta, para ello se asigna clases a la operación normal y las condiciones de falla.

Para el desarrollo y evaluación se sigue la metodología trabajada a lo largo del documento, primero se evalúa el conjunto de datos desbalanceado y luego se hace un balance de los datos usando la técnica antes mencionada.

La Figura 16 presenta la distribución de elementos en cada una de las clases, se puede observar que la clase 0 es quien tiene mayor participación lo que es consecuente con lo mencionado anteriormente, luego sigue la clase 2 con una menor participación, la clase 3 y 4 tienen una participación similar, finalmente la clase 1 con una menor participación.

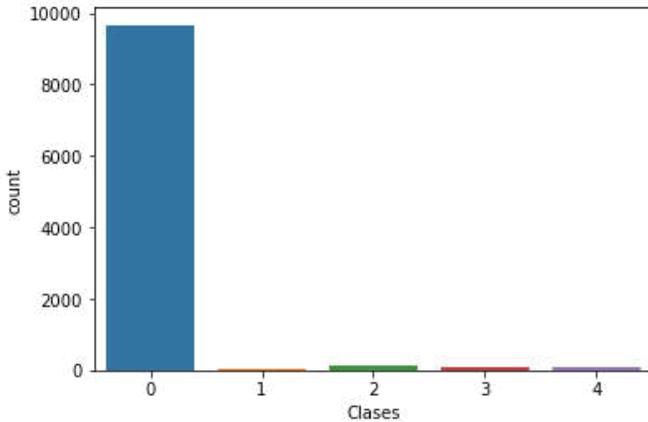


Fig. 16. Distribución de dataset en clases.

Ahora se presenta la matriz de confusión resultado del diseño de una red neuronal multiclase, se observa que la clase 1 no tiene clasificación verdadera todos los elementos están mal clasificados, la clase 3 y la clase 5 presentan una clasificación aceptable, para c4 de los 17 elementos se clasifican 15 positivamente, para c3 de 18 se clasifica bien 12 (Ver Fig 17).

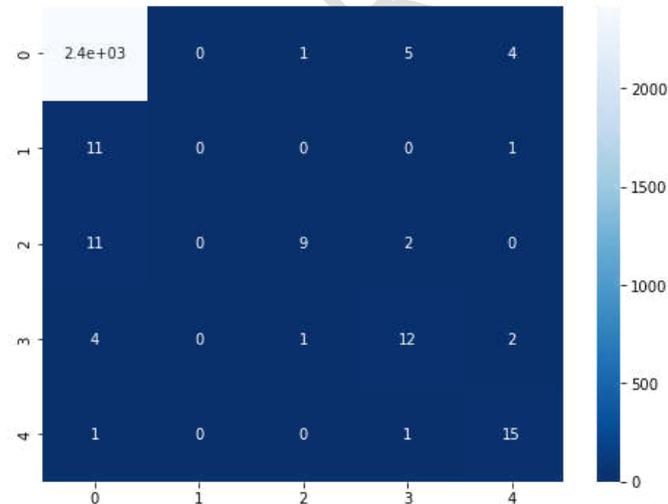


Fig. 17. Matriz de confusión para red neuronal multiclases.

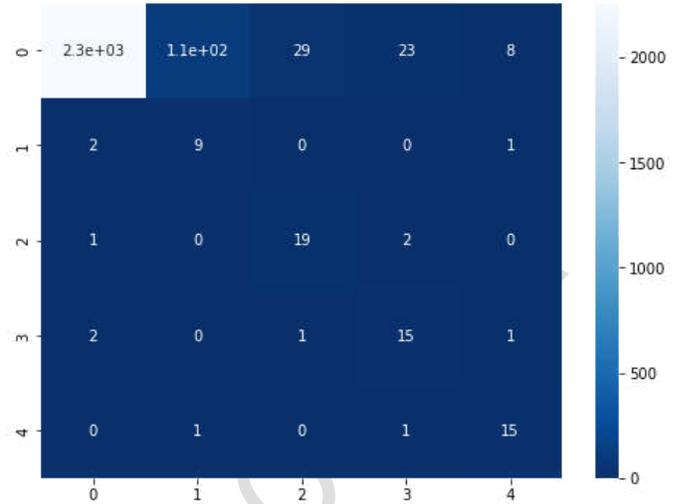


Fig. 18. Matriz de confusión datos balanceados.

La diagonal mostrada en la Fig 18 representa la clasificación positiva, a diferencia de la Figura 20, la clase 1 ahora presenta 9 elementos bien clasificados frente a 3 que se ubican fuera del grupo, la clase 2 de los 22 elementos que pertenecen a este grupo 19 son clasificados correctamente, en la clase 3 se tiene 20 elementos y 15 son positivos los demás son negativos, la clase 4 tiene 17 elementos con 15 clasificados positivamente.

Comparando las dos matrices de confusión se observa que el desbalance también afecta a un sistema multiclase, con el balance propuesto en este documento mejora la clasificación de todas las clases.

### III. CONCLUSIÓN

Las redes neuronales presentan un avance significativo en la predicción de mantenimiento predictivo, se implementaron dos técnicas (algoritmos) Random Forest y Redes Neuronales para clasificar operación normal y anormal, también se evaluó un sistema multiclase donde permite segmentar la condición de falla, dentro de las posibilidades del conjunto de datos.

Random Forest y Redes Neuronales vistos desde métricas de precisión son bastante precisas llegando a 99 %, esto quiere decir que clasificaron bien los verdaderos positivos que representan una condición normal, pero al evaluar los verdaderos negativos, estos representan condición de falla, presentan una alta deficiencia donde clasifican datos condiciones de falla como esta operara con normalidad. Esto errores se presentan cuando se hace entrenamiento con datos desbalanceados.



La técnica de SMOTE que se emplea para equilibrar o balancear el conjunto de datos presenta buenos resultados para este caso, tanto en el sistema biclase como el multiclase permitió mejorar el acierto de datos verdaderos positivos.

Las redes neuronales presentan un mejor desempeño en la clasificación tanto de verdaderos positivos como de verdaderos negativos, se obtuvieron resultados de 0.86 para recall y 0.92 para accuracy.

#### REFERENCES

- Medina Hurtado, S., Moreno Cadavid, J., & Gallego Valencia, J. P. (2011). Pronóstico de la demanda de energía eléctrica horaria en Colombia mediante redes neuronales artificiales. *Revista Facultad de Ingeniería Universidad de Antioquia*, (59), 98-107.
- Villaseñor-Aguilar, M., Ramírez-Agundis, A., Padilla-Medina, J. A., & Orozco-Mendoza, H. (2011). Control de estabilidad de un manipulador planar paralelo 3RRR utilizando redes neuronales. *Científica*, 15(3), 107-115.
- Olarte, W., Botero, M., & Cañón, B. (2010). Técnicas de mantenimiento predictivo utilizadas en la industria. *Scientia et technica*, 2(45), 223-226.
- Sánchez-Gómez, A. M. (2017). Técnicas de mantenimiento predictivo: metodología de aplicación en las organizaciones.
- Pozo Gallego, C. D. (2018). Mantenimiento predictivo en motores de inducción aplicando técnicas de inteligencia artificial y teoría de desequilibrios electromagnéticos.
- Fernández, M. C. G. (2004). Planificación y medida de la efectividad del mantenimiento predictivo aplicado a un proceso industrial basándose en el uso de técnicas de modelado de su comportamiento y de inteligencia artificial (Doctoral dissertation, Universidad Pontificia Comillas).