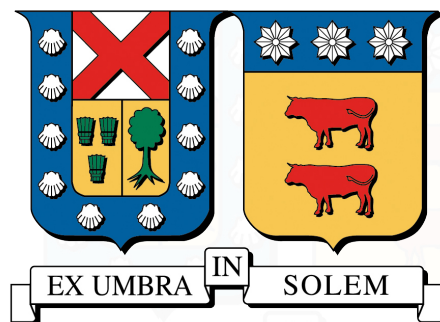


UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
DEPARTAMENTO DE INDUSTRIAS
VALPARAÍSO - CHILE



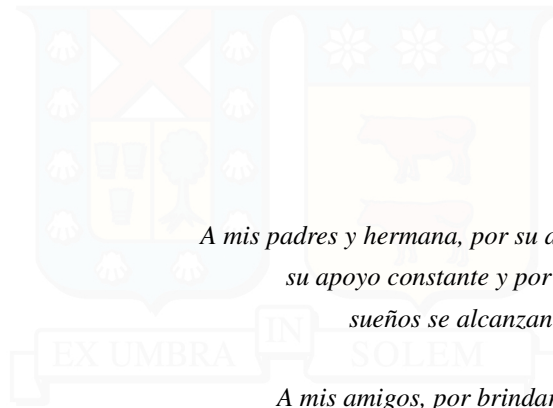
**IDENTIFICACIÓN DE DEPENDENCIA E INTERDEPENDENCIA ENTRE
COMPONENTES Y TIPOS DE FALLAS EN UNA LÍNEA DE CHANCADO
DE UNA EMPRESA MINERA DE CHILE**

VALENTINA IGNACIA FLORES CASTRO

TESIS PARA OPTAR AL GRADO DE MAGÍSTER EN CIENCIAS DE LA INGENIERÍA
INDUSTRIAL Y AL TÍTULO DE INGENIERA CIVIL INDUSTRIAL

PROFESOR GUÍA : Dr. PABLO A. VIVEROS GUNCKEL - UTFSM
PROFESOR CORREFERENTE : Dr. Rodrigo Mena Bustos - UTFSM

ENERO 2025



*A mis padres y hermana, por su amor incondicional,
su apoyo constante y por enseñarme que los
sueños se alcanzan con perseverancia.*

*A mis amigos, por brindarme ánimo, energía
y compañía en todo momento.*

*A mi profesor guía, por su paciencia, apoyo
y motivación durante todo este trabajo.*

*Y a mis abuelos que me acompañan siempre,
siendo una luz constante en mi camino.*

***Con gratitud y cariño, dedico esta
investigación a todos ustedes.***

RESUMEN EJECUTIVO

En el presente informe se estudia la forma y grado de dependencia entre los componentes y los distintos tipos de fallas que existen dentro de un sistema compuesto por un chancador y una correa transportadora de cobre de una empresa minera ubicada en el norte de Chile.

La motivación de este trabajo es lograr predecir, las diversas fallas que se pueden producir en el sistema (según la data histórica), gracias a un modelo que contemple las relaciones existentes entre los componentes numéricos y los tipos de fallas. Esto ayudará a mejorar la precisión y la eficacia de los sistemas que alertan sobre irregularidades en el proceso, además de optimizar el mantenimiento preventivo de los equipos en estudio y también entregar información importante que permita tomar las mejores decisiones.

Luego de realizar la investigación bibliográfica se llega a la conclusión que para el objetivo que se tiene, se debe utilizar un modelo de Random Forest (RF) o de Redes Neuronales Artificiales (ANN), en conjunto a un modelo de Ecuaciones Estructurales (SEM). Al principio se prueba con ambos modelos de algoritmos de aprendizaje automático y mediante los siguientes indicadores de desempeño: matriz de confusión, accuracy, precisión, recall y F1-score, se observa cual de estos presenta mejores índices, para así, implementar los resultados del modelo como la hipótesis del modelo SEM.

Una vez realizada la codificación de lo anterior, es crucial validar el ajuste y la robustez, en donde se utilizan indicadores tales como chi-cuadrado, índice de ajuste comparativo, de bondad de ajuste, la raíz del error cuadrático medio de aproximación y muchos otros.

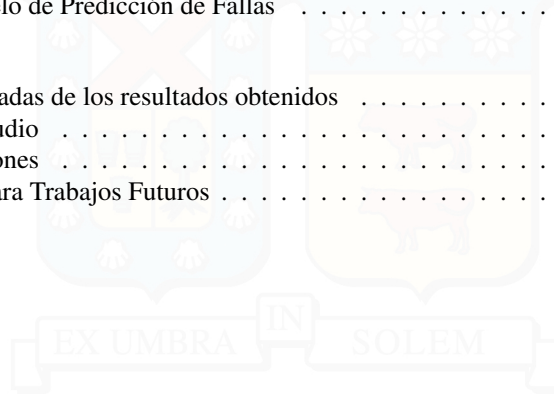
Como resultado, se obtiene que el modelo de algoritmos de aprendizaje de ANN presenta muy buenos resultados, exhibiendo un accuracy de 0.9926 y con relaciones significativas entre las variables según el modelo SEM, mientras que, por otro lado, el modelo de RF muestra un claro sobreajuste en el producto final, no logrando entregar un resultado real para medir en el modelo SEM.

La hipótesis del modelo con mejores resultados se incluye en el entrenamiento de un modelo de predicción de fallas, también desarrollado con ANN, en donde se observa que los resultados de las predicciones de fallas de este, versus un modelo de predicción que no considera las relaciones de dependencia, presenta mejores resultados y con mayor anticipación frente a una posible falla.

Índice de Contenidos

1. Introducción	1
2. Objetivos	2
2.1. Objetivo General	2
2.2. Objetivos específicos	2
3. Marco Teórico	3
3.1. Dependencia e Interdependencia	3
3.1.1. Introducción al Concepto de Dependencia e Interdependencia	3
3.1.2. Representación de las Relaciones de Dependencia e Interdependencia	4
3.1.3. Categorización de la Dependencia e Interdependencia	4
3.1.4. Modelos Teóricos	5
3.1.4.1. Tablas de Dependencias	6
3.1.4.2. Tablas de Interacción	7
3.1.4.3. Enfoque Basado en Datos	7
3.1.5. Modelo de Ecuaciones Estructurales (SEM)	7
3.2. Machine Learning	8
3.2.1. Introducción a Machine Learning	8
3.2.2. Tipos de Aprendizajes en Machine Learning	8
3.2.3. Aplicaciones	9
3.2.4. Tipos de Aprendizaje Útiles para la Clasificación	9
3.3. Digital Twin	10
3.3.1. Introducción al Digital Twin	10
3.4. Tipos de Fallas en Sistemas de Chancado Minero	11
3.4.1. Fallas Mecánicas	11
3.4.2. Fallas Eléctricas	11
3.4.3. Fallas Térmicas	12
3.4.4. Prácticas Incorrectas	12
3.4.5. Propagación de Fallas	12
3.5. Modelos de Predicción de Fallas	13
4. Metodología	14
4.1. Estructura de la Investigación	14
4.2. Marco de Trabajo Metodológico	14
4.3. Descripción del Sistema en Estudio	14
4.4. Limpieza y Procesamiento de Datos	16
4.5. Análisis y Evaluación de Datos	17
4.6. Modelos de Dependencia a Implementar	19
4.6.1. Modelo de Random Forest	19
4.6.2. Modelo de Redes Neuronales Artificiales	21
4.6.3. Modelo de Ecuaciones Estructurales	22
4.7. Indicadores de Desempeño y Validación de los Resultados	23
4.8. Modelo de Predicción de Fallas	28
5. Resultados	30
5.1. Datos de Entrada a los Modelos de Machine Learning	30
5.2. Resultados Modelo Random Forest	30
5.2.1. Resultados SEM para Hipótesis del Modelo de Random Forest	33

5.3. Resultados Modelo Redes Neuronales	39
5.3.1. Resultados SEM Para Hipótesis del Modelo de Redes Neuronales	42
5.4. Resultados Modelo Predicción de Fallas	48
6. Discusión y Alcance	49
6.1. Evaluación del Modelo de Random Forest en conjunto de SEM	49
6.2. Evaluación del modelo de Redes Neuronales en conjunto de SEM	51
6.3. Evaluación del modelo de Predicción de Fallas	52
7. Conclusiones	54
7.1. Observaciones emanadas de los resultados obtenidos	54
7.2. Limitaciones del estudio	54
7.3. Alcances y Aplicaciones	55
7.4. Recomendaciones para Trabajos Futuros	56
8. Anexos	58
Bibliografía	59



Índice de Tablas

4.1. Clasificación de Tipos de Detención en Operacionales o Falla	18
4.2. Variable Binaria del Modelo de Predicción de Fallas	28
4.3. Métricas a Evaluar para el Modelo de Predicción de Fallas	29
5.1. Variables Numéricas incorporadas en los modelos	30
5.2. Resultados Modelo Random Forest	31
5.3. Resultados Variables Más Influyentes Para Cada Evento	32
5.4. Resultados SEM del modelo Random Forest	34
5.5. Resultados SEM del modelo Random Forest	36
5.6. Resultados Modelo Redes Neuronales	39
5.7. Resultados Variables Más Influyentes Para Cada Evento	41
5.8. Resultados SEM del modelo Redes Neuronales	43
5.9. Resultados SEM del modelo Redes Neuronales	45
5.10. Resultado Modelo Predicción de Fallas Con y Sin Dependencias	48
8.1. Variables Numéricas, Significado y Abreviatura	58

Índice de Figuras

3.1. Grafo dirigido de los componentes de un sistema	4
4.1. Línea de Chancado Minero	14
4.2. Sistema Integrado en Estudio	15
4.3. Comportamiento Temporal Variables AD	17
4.4. Frecuencia de los eventos para el Chancador N°5	19
5.1. Matriz de Confusión del modelo de Random Forest	32
5.2. Comportamiento Temporal Variables Más Influyentes	33
5.3. Relaciones entre Variables Dependientes e Independientes	39
5.4. Matriz de Confusión del modelo de Redes Neuronales	40
5.5. Comportamiento Temporal Variables Más Influyentes	41
5.6. Relaciones entre Variables Dependientes e Independientes	47

1 | Introducción

En el contexto actual de los avances tecnológicos, la creación y gestión de sistemas complejos se ha vuelto fundamental en diversas áreas, desde la industria manufacturera hasta la ingeniería de software. Una de estas es la industria minera en Chile, la cual no se ha quedado atrás en este proceso y poco a poco ha ido adquiriendo actualizaciones en torno a sus formas de operar.

En este sentido es que se ha generado la necesidad de estudiar y comprender en profundidad como se puede diagnosticar una dependencia o interdependencia entre los componentes de un sistema y los diferentes tipos de fallas que pueden ocurrir dentro de este, en donde, para el caso de estudio, el sistema está compuesto por una línea de chancado minero, un Sistema de Control Distribuido (DCS), conectado a los equipos mediante sensores, y un Digital Twin que recrea digitalmente la información que recibe del DCS, los cuales alimentan una base de mantención con todas las irregularidades que se presentan.

Identificar y cuantificar estas relaciones no solo permitirá mejorar la calidad de los sistemas de alerta temprana, sino también proporcionará insights valiosos para optimizar la operación y el mantenimiento preventivo de los equipos pertenecientes al sistema, analizando cuanto influye conocer las relaciones de dependencia existentes entre las variables, en la mejora de resultados de los sistemas de predicciones de falla.

Finalmente, a medida que la complejidad de los sistemas digitales sigue aumentando, el estudio de estos se vuelve crucial para abordar desafíos como la interoperabilidad, la escalabilidad y la seguridad. Además, comprender cómo los diferentes elementos de un sistema interactúan entre sí no solo mejora la eficiencia operativa, sino que también reduce los riesgos asociados con posibles fallos o mal funcionamiento.

A continuación, se presenta el estudio de dicho problema, la metodología a utilizar, los resultados obtenidos, las discusiones y alcances que se pueden observar a partir de la hipótesis elaborada.

2 | Objetivos

2.1. Objetivo General

El objetivo principal es explorar, investigar y cuantificar las relaciones de dependencia e interdependencia entre los componentes de un sistema compuesto de una línea de chancado minera, y los diversos tipos de fallas que pueden desarrollarse en un equipo de la línea. Esto con el fin de implementar las relaciones descubiertas en el entrenamiento de un modelo de predicción de fallas, verificando si la precisión, rapidez y eficacia mejoran, en comparación a un modelo de predicción de fallas que no considera estas relaciones. Por otro lado, los objetivos específicos son los siguientes:

2.2. Objetivos específicos

- Lograr una visión amplia sobre los tipos de modelos de dependencia e interdependencia, mediante el estudio exhaustivo de la literatura sobre este tema, para aplicar los modelos correctos en el estudio.
- Identificar un modelo de aprendizaje automático, mediante la investigación de la literatura, que sirva para la ejecución del modelo de dependencias.
- Realizar una revisión y limpieza de los datos históricos del sistema, a través de lenguaje de programación Python, para que se puedan utilizar en el modelo de aprendizaje automático escogido.
- Ejecutar el modelo, logrando obtener la hipótesis de las relaciones con sus diferentes grados de importancia entre los componentes y cada tipo de falla del sistema.
- Validar los modelos mediante indicadores de desempeño, para así determinar la veracidad del estudio.
- Implementar la hipótesis obtenida en un modelo de predicción de fallas, el cual también se desarrolle en base a algoritmos de aprendizaje, logrando obtener resultados que sirvan para comparar el rendimiento de indicadores de este, en comparación a un modelo que no considere las relaciones de dependencia.
- Discutir sobre la ejecución y los resultados obtenidos, determinar en qué circunstancias se puede aplicar el modelo y analizar la complejidad computacional del problema.
- Proponer alternativas para la mejora de los procedimientos utilizados y ver cuáles son las temáticas de investigaciones futuras.

3 | Marco Teórico

3.1. Dependencia e Interdependencia

3.1.1. Introducción al Concepto de Dependencia e Interdependencia

En este trabajo, lo que se analiza son las dependencias entre componentes de un sistema y los diversos tipos de fallas, y cómo éstas se pueden identificar y cuantificar para determinar si realmente existe o no una relación y si es así, en qué grado. Evaluar la dependencia entre componentes de un sistema es fundamental para comprender cómo actúan entre sí, ya que estas pueden introducir una vulnerabilidad adicional y además provocar fallas en cascada. Es por esto que comprender las interdependencias y evaluar el impacto es fundamental para mitigar las consecuencias adversas [1]. Otros estudios afirman que las interrupciones en infraestructuras a menudo se transmiten a otras dependientes y posiblemente regresen a donde se originó [2].

En condiciones normales las interdependencias por lo general mejoran la confiabilidad y eficacia de los servicios de infraestructura [1]. Pero, por otro lado, las interconexiones entre infraestructura se hacen más evidente cuando se generan fallas. [3]. Estas interconexiones se nombran de diversas formas en otros artículos, ya que por un lado, las dependencias representan relaciones con una única dirección, en donde las ocurrencias del primer activo influyen en la funcionalidad del segundo y por el contrario las ocurrencias del segundo no afectan al primero. Y luego se tienen las interdependencias, en donde lo que le ocurra al primer activo se verá reflejado en el segundo y viceversa.

Dentro de la literatura algunos autores aseguran que para comprender cómo una contingencia puede debilitar y potencialmente interrumpir la operación segura de un sistema interconectado, es crucial modelar las relaciones establecidas a través de las conexiones entre los múltiples componentes de las infraestructuras involucradas [4]. Otros comentan que explorar las dependencias entre variables puede proporcionar información crucial para la toma de decisiones, la predicción de eventos futuros y la comprensión de la estructura subyacente de un sistema [5]. También se han destacado las dificultades intrínsecas en el modelado de las interdependencias entre infraestructuras críticas a través de estudios empíricos. Por ejemplo, se ha observado que los eventos que desencadenan cascadas son más frecuentes de lo que se cree y que las dependencias suelen ser más direccionales de lo que se piensa comúnmente [4]. Otro punto es que la detección de dependencias entre factores puede revelar conexiones no evidentes a simple vista, permitiendo una visión más profunda y completa de la interacción entre diferentes componentes de un sistema [5].

3.1.2. Representación de las Relaciones de Dependencia e Interdependencia

Las relaciones de dependencia e interdependencia entre componentes de un sistema pueden representarse utilizando matrices de adyacencia para grafos.

Una matriz de adyacencia es una matriz cuadrada utilizada para representar un grafo. Los elementos de la matriz indican si pares de vértices (nodos) están adyacentes o no en el grafo. Por un lado, si el componente i depende del componente j , el elemento (i, j) de la matriz de adyacencia $\mathbf{A}$ es 1. Lo que matemáticamente se representa como $A_{i,j} = 1$ si $i \rightarrow j$, de lo contrario, $A_{i,j} = 0$. Por otro lado, si el componente i y el componente j son interdependientes, tanto $A_{i,j}$ como $A_{j,i}$ son 1. Matemáticamente queda expresado como $A_{i,j} = 1$ y $A_{j,i} = 1$ si $i \leftrightarrow j$.

A modo de ejemplo, se puede ver el grafo de la Figura 3.1 con los componentes A, B, C, D y E , en donde las relaciones entre A y B , B y C , D y C son de dependencia, y por otro lado, la relación de A y E es de interdependencia.

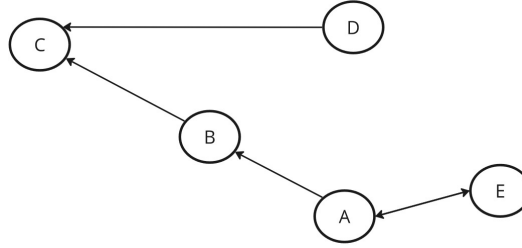


Figura 3.1: Grafo dirigido de los componentes de un sistema

El grafo se puede expresar con una matriz de adyacencia, la cual como propiedad tiene que siempre será cuadrada (de tamaño $n \times n$, donde n es el número de nodos). Además, la diagonal principal ($A_{i,i}$) generalmente es 0, ya que no se consideran bucles en este contexto. Por último, en el caso de interdependencia, los elementos fuera de la diagonal son simétricos ($A_{i,j} = A_{j,i}$).

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

3.1.3. Categorización de la Dependencia e Interdependencia

Varios estudios a través de los años han estado en la búsqueda de la clasificación de las dependencias e interdependencias. En el 2023 se publicó un artículo [3] de que hace una recopilación de toda la información existente más relevante sobre esto, en donde se puede comprender un poco mejor la descomposición de la interdependencia.

Según la investigación, las interdependencias se clasifican en dos dimensiones ortogonales: ontología y epistemología. La ontología por su lado describe el cuándo y cómo aparecen las interdependencias, clasificándolas en: episódica (interdependencias que aparecen sólo bajo ciertas condiciones o en determinadas situaciones) y crónicas (interdependencias que siempre existen). Por otro lado, la epistemología clasifica las interdependencias en cuatro etapas de modelado, las cuales son: peligro y exposición, política y control, operación y rendimiento, y deterioro y recuperación.

- **Peligro y Exposición:** En esta etapa, se desarrolla un modelo para simular los impactos de un evento peligroso, como un desastre natural, en la red de infraestructura. Se analizan los posibles peligros y la exposición de los sistemas a estos eventos.
- **Política y Control:** Se establece un modelo de políticas y control que captura las consideraciones económicas relacionadas con la planificación de la resiliencia. Se definen las estrategias y medidas de control para mitigar los impactos de los eventos peligrosos.
- **Operación y Rendimiento:** Aquí se describe la operación y el rendimiento del sistema para cada realización de desastre en la segunda etapa. Se evalúa cómo funcionan los sistemas de infraestructura durante y después de un evento peligroso.
- **Deterioro y Recuperación:** Esta etapa suele abordar la recuperación de los sistemas de infraestructura después de un evento peligroso. Se analiza cómo los sistemas se deterioran durante un desastre y cómo se pueden recuperar y restaurar para volver a la operación normal.

Lo anterior finalmente muestra como se describe el cuando y el cómo aparecen las interdependencias por un lado, y por otro, clasifica estas en cuatro partes de modelado.

Luego tenemos otras dimensiones de clasificación para las dependencias de infraestructura crítica [6], en donde el criterio de división son los tipos de conexiones entre las infraestructuras, en donde los tipos son: físico, cibernético (virtual), geográfico y lógico, en donde la conexión puede ser estático (de forma continua, sin importar el estado del sistema) o dinámico (cuando surgen circunstancias específicas). Luego se tiene que evaluar el tipo de error que evoca el enlace, el cual puede ser en cascada, de causa común o creciente y luego el rango que alcanza, el cual puede ser primario, sistémico, sectoriales o intersectorial. A partir de esto, se evalúan las características de los acoplamientos y reacciones a las interdependencias, en donde en primer lugar se ve el orden de la dependencia, ósea, si es lineal o no. Después se observa la intensidad de la dependencia y la respuesta de la infraestructura. Y finalmente se clasifican según los factores que configuran el entorno de operación, como lo pueden ser por ejemplo el ámbito social político, la salud, el cuidado de la seguridad, lo económico, etc.

3.1.4. Modelos Teóricos

A continuación, se mostrarán los modelos de dependencia e interdependencia más relevantes de acuerdo con el estudio de la literatura actual, en donde estos se pueden clasificar principalmente en tres macro categorías, [1] las cuales son:

- **Tablas de dependencia:** Basadas en encuestas principalmente, representan la interdependencia de manera cualitativa y cuantitativa.

- **Reglas de interacción:** Las cuales sigue reglas predefinidas de interacción.
- **Enfoque basado en datos:** Se alimenta de una base de datos y puede representar interdependencias complejas a nivel de componentes. Precisamente en esta categoría existe una baja madurez de desarrollo.

Ahora, cada una engloba diferentes formas de modelar la dependencia o interdependencia y para saber cual es la que mejor se puede adaptar a los datos que se desean estudiar, es muy recomendable hacer una investigación de sobre lo que trata cada uno de estos modelos.

En primer lugar, las tablas de dependencia la componen cinco modelos, los cuales son las tablas de dependencia cualitativas, las tablas de dependencia basadas en encuestas, las tablas de correlación, la teoría de grafos y la teoría de la economía.

Luego para las tablas de interacción, también la componen cinco modelos, los cuales son en primer lugar las simulaciones de eventos discretos, luego los modelos basados en agentes, los sistemas dinámicos, las redes bayesianas y finalmente los modelos de oferta y demanda agregada.

Por último, para el enfoque basado en datos se tiene el análisis de series temporales (ARIMA), sus derivadas y los algoritmos de aprendizaje de Machine Learning.

Luego de esta categorización de los modelos, se realiza una breve descripción de cada uno.

3.1.4.1. Tablas de Dependencias

- **Tablas de dependencia cualitativas:** Describen la fuerza o la naturaleza de la dependencia de manera cualitativa. Estas indican de manera cualitativa el grado de dependencia de un componente sobre otro. Son adecuadas para evaluaciones preliminares debido a factores intuitivos, es fácil de implementar y con un costo computacional bajo.
- **Tablas de dependencia basada en encuestas:** Primero se realiza una o varias preguntas y se relaciona esta a cada variable en investigación y luego los entrevistados dan su opinión sobre la percepción de la variable según la pregunta. Es una representación simple y debe tener una muestra grande para que la opinión sea representativa.
- **Tablas de correlación:** Es un método estadístico robusto para calcular las correlaciones entre variables. Estas varían entre 1 y -1, en donde mientras más cercano a uno o menos uno, significa que existe una fuerte relación lineal (positiva o negativa respectivamente), en cambio mientras más cercano a 0, se interpreta relación débil o inexistente. La más usada es la correlación de Pearson, pero también existen otras como la de Spearman y la de Kendall.
- **Teoría de grafos:** Estas se utilizan en la gestión de proyectos y planificación de tareas, en donde es esencial comprender el orden en que se deben realizar las actividades. Se utilizan grafos dirigidos por lo general, en donde los nodos representan las variables o componentes, y las aristas indican las dependencias entre los componentes.
- **Teoría de la economía:** Representan relaciones entre diferentes variables de índole económicas. Por ejemplo, muestran como las distintas industrias dependen unas de otras para satisfacer la

demanda del cliente.

3.1.4.2. Tablas de Interacción

- **Simulación de eventos discretos:** Representan eventos que ocurren en momentos discretos y tienen condiciones iniciales, periodos de tiempo, actualización de variables y registro de resultados e indicadores como rendimiento, tiempos y eficiencia. Evalúa el desempeño en condiciones dinámicas.
- **Modelos basados en agentes:** Son simulaciones en donde los agentes del modelo toman decisiones y se relacionan con reglas predefinidas. Está compuesto de agentes, el entorno y las reglas de interacción. Se utiliza para estudiar sistemas complejos y emergentes que resultan de las interacciones locales entre agentes y también considera condiciones iniciales, movimiento e interacciones.
- **Sistemas dinámicos:** Son modelos matemáticos que describen como las variables cambian a lo largo del tiempo. Se necesitan datos precisos, la complejidad computacional es alta y se deben elegir los parámetros correctos. Eso sí, son limitantes en predicciones de largo plazo.
- **Redes Bayesianas:** Estas redes escriben probabilísticamente la ocurrencia de un evento, dado el conocimiento de otros eventos en la red. Son modelos gráficos que representan las relaciones de dependencias entre variables y aquí se deben definir nodos, que representan a las variables y establecer relaciones dirigidas entre estas según la secuencia de padre a hijo.
- **Modelos de oferta y demanda agregada:** Refleja la interdependencia entre distintos componentes de una economía. Se basa siempre en variables económicas como el PIB, Consumo de Bienes y Servicios, Inversión, Gasto público, Exportaciones, Importaciones, entre las cuales se establecen relaciones de interdependencia representadas en ecuaciones matemáticas.

3.1.4.3. Enfoque Basado en Datos

- **Análisis de series temporales:** Se utilizan para modelar y prever datos que exhiben patrones temporales, tendencias y estacionalidad. Por lo general tienen tres pasos, análisis exploratorio, se escoge y se ajusta el modelo a los datos, y finalmente se realiza el diagnóstico. Las técnicas principales para hacer el análisis son Hot Winters y ARIMA.
- **Algoritmos de aprendizaje:** Son herramientas que pueden aprender patrones a partir de datos y realizar tareas específicas sin ser programados explícitamente para esas tareas, las cuales son adecuadas para problemas complejos y dinámicos. Estos pueden manejar grandes volúmenes de datos y extraer patrones significativos.

3.1.5. Modelo de Ecuaciones Estructurales (SEM)

Existen técnicas de análisis de datos multivariantes de primera generación, como las regresiones múltiples y análisis de varianza, los cuales son aplicados en numerosos campos científicos. Estas se

ocupan para investigaciones empíricas de correlaciones propuestas entre variables relevantes para el estudio. Sin embargo, estos vienen con ciertas limitaciones, incluyendo la asunción de una estructura de modelo simplista y el requisito de que las variables sean observables y medidas con precisión [7].

El modelado de la ecuación estructural (SEM), una técnica de segunda generación, la cual aborda las limitaciones mencionadas. SEM permite modelar las interacciones complejas entre múltiples variables y explica errores de medición, mejorando así la precisión de los constructos teóricos [8].

Técnicas de aprendizaje automático como Artificial Neural Network (ANN) y Random Forest (RF) de Machine Learning son de mucha utilidad para encontrar relaciones entre variables [9] y entregar una buena hipótesis que sirva como input para el modelo SEM.

3.2. Machine Learning

3.2.1. Introducción a Machine Learning

El Machine Learning (ML), o aprendizaje automático, es una disciplina dentro de la inteligencia artificial que se centra en el desarrollo de algoritmos y modelos que permiten a las máquinas aprender y mejorar a partir de la experiencia, sin necesidad de ser programadas explícitamente [10]. A lo largo de las últimas décadas, ML ha experimentado un crecimiento exponencial, transformando diversos sectores como la medicina, las finanzas, el comercio y la industria del entretenimiento. La capacidad de ML para identificar patrones complejos y realizar predicciones precisas a partir de grandes volúmenes de datos ha abierto nuevas posibilidades y aplicaciones que antes eran inimaginables [11].

Machine Learning se refiere a la creación y uso de algoritmos y modelos estadísticos que los sistemas informáticos emplean para realizar tareas específicas sin recibir instrucciones detalladas. En lugar de seguir reglas preestablecidas, estos sistemas aprenden de los datos, identificando patrones y tomando decisiones basadas en ejemplos anteriores [12]. La principal ventaja de ML es su capacidad para mejorar automáticamente a partir de la experiencia acumulada, lo que lo diferencia notablemente de los métodos tradicionales de programación [10].

3.2.2. Tipos de Aprendizajes en Machine Learning

En este ámbito, existen tres principales tipos de aprendizaje: supervisado, no supervisado y por refuerzo [12]. El aprendizaje supervisado implica el uso de un conjunto de datos etiquetados, donde cada entrada está asociada con una salida conocida, permitiendo al modelo aprender a hacer predicciones o clasificaciones basadas en estos ejemplos. Este tipo de aprendizaje es ampliamente utilizado en tareas como la clasificación de correos electrónicos en categorías de spam o no spam, así como en la predicción de precios de viviendas mediante la regresión lineal [11].

El aprendizaje no supervisado, por otro lado, trabaja con datos no etiquetados y se centra en descubrir patrones ocultos o estructuras subyacentes dentro de los datos. Este enfoque es útil

en aplicaciones como la segmentación de clientes en marketing, donde se utilizan algoritmos de clasificación como K-means para agrupar clientes con comportamientos similares, o en la reducción de dimensionalidad mediante el análisis de componentes principales (PCA) [12].

El aprendizaje por refuerzo es una técnica donde un agente aprende a tomar decisiones interactuando con un entorno [13]. A través de un proceso de prueba y error, el agente recibe recompensas o penalizaciones en función de sus acciones, ajustando su estrategia para maximizar la recompensa acumulada. Este tipo de aprendizaje ha sido fundamental en el desarrollo de tecnologías avanzadas como los vehículos autónomos y los sistemas de juegos complejos, como AlphaGo de Google [12].

3.2.3. Aplicaciones

Las aplicaciones de Machine Learning son vastas y diversas, abarcando múltiples sectores [10]. En el campo de la salud, ML se utiliza para el diagnóstico de enfermedades, el análisis de imágenes médicas y la predicción de brotes de enfermedades [11]. En finanzas, se emplea para la detección de fraudes, el análisis de riesgos crediticios y la implementación de algoritmos de trading automatizado.

El marketing ha visto un gran impacto de ML en la segmentación de clientes, la predicción de ventas y el análisis de sentimientos en redes sociales. En el ámbito del transporte, ML optimiza rutas, mejora el mantenimiento predictivo y es fundamental en el desarrollo de vehículos autónomos. Además, en la industria del entretenimiento, ML impulsa sistemas de recomendación en plataformas de streaming y analiza las preferencias de los usuarios para personalizar experiencias [12].

3.2.4. Tipos de Aprendizaje Útiles para la Clasificación

Dentro del aprendizaje supervisado, que es especialmente útil para problemas de clasificación, existen varios algoritmos y técnicas destacables [10]. La regresión logística es una de las técnicas más básicas y utiliza una función logística para modelar la probabilidad de que una instancia pertenezca a una clase particular. Las máquinas de vectores de soporte (SVM) son otro método poderoso que busca encontrar el hiperplano óptimo que separe las diferentes clases en un espacio de características [11].

Los árboles de decisión y los bosques aleatorios son técnicas que utilizan estructuras de árbol para dividir el conjunto de datos en subconjuntos homogéneos [10]. Mientras que los árboles de decisión pueden ser propensos al sobreajuste, los bosques aleatorios combinan múltiples árboles para mejorar la precisión y controlar el sobreajuste. Las redes neuronales, inspiradas en la estructura del cerebro humano, son capaces de reconocer patrones extremadamente complejos en los datos y son fundamentales en aplicaciones avanzadas como el reconocimiento de imágenes y el procesamiento del lenguaje natural [12]. Por último, el algoritmo de K-Nearest Neighbors (K-NN) clasifica una instancia basándose en la mayoría de votos de sus vecinos más cercanos, siendo útil en tareas como la clasificación de textos y el reconocimiento de patrones [11].

Un ejemplo de la utilización de estos métodos se da en la investigación de Abu Bakar, et, al., 2024, en donde se busca encontrar el modelo de aprendizaje automático que mejor prediga la naturaleza magnética de las composiciones de óxido, mediante la evaluación de cinco modelos de ML, los cuales incluyen la Regresión Logística (LR), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF) y Artificial Neural Network (ANN). Según este estudio, el RF supera a otros modelos en tres métricas de evaluación diferenciadas, con precisión 89.53 %, precisión 92,354 %, F1 91.343 %, mientras que ANN supera en la métrica Recall teniendo una puntuación 91,43 % [14].

3.3. Digital Twin

3.3.1. Introducción al Digital Twin

El concepto de Digital Twin, o gemelo digital, ha emergido como una innovación disruptiva en la era de la Industria 4.0. Este término, acuñado por primera vez por Michael Grieves en 2002, [15] ha ganado una relevancia significativa en diversos sectores industriales y de ingeniería, transformando la forma en que se diseñan, operan y mantienen productos y sistemas complejos.

El origen del Digital Twin se remonta a los primeros años del siglo XXI, donde Michael Grieves, entonces profesor de la Universidad de Michigan, propuso la idea de crear una réplica virtual de un producto físico [16]. Inicialmente concebido para optimizar el ciclo de vida de productos manufacturados, el concepto ha evolucionado para abarcar una variedad de aplicaciones en diferentes industrias, incluyendo la fabricación, la salud, la logística, la energía y la gestión de infraestructuras.

Hoy en día, múltiples artículos han definido y discutido el concepto de Digital Twin [17][18][19][20][21]. En Akanmu et al., (2014) señalaron que la naturaleza del Digital Twin es conectar un artefacto físico con su contraparte digital [22]. En Tao y Cols., (2019) propusieron la idea de que “un Digital Twin debe constar de cinco partes: parte física, parte virtual, conexiones, datos y servicios”. Específicamente, la parte física es la base de la parte virtual y se refleja en la parte virtual; los datos y la información podrían ser transferidos por la parte de conexión; la parte de servicios de DT proporciona la simulación, la toma de decisiones, el seguimiento y el control del objeto físico. Finalmente, los datos impulsan la parte de servicios para mejorar todo el sistema[23].

En consecuencia, un Digital Twin es una representación digital en tiempo real de un objeto, proceso o sistema físico. Esta representación virtual se actualiza continuamente mediante la integración de datos de sensores, simulaciones y análisis, lo que permite monitorear, predecir y optimizar el rendimiento del gemelo físico en tiempo real. Además, el Digital Twin puede abarcar tanto aspectos físicos como funcionales, lo que proporciona una comprensión holística y detallada de su contraparte real.

En Ye et al., (2023) se realiza una investigación para reducir la ocurrencia de accidentes de construcción y evitar víctimas y pérdidas de propiedad durante la construcción del túnel, y para esto se propone implementar un sistema de alerta temprana en la gestión de seguridad del proyecto de ingeniería subterránea [24]. En el túnel se instalaron sensores, los cuales recopilan información y con

esto se alimenta al Digital Twin. Gracias a la implementación, ocurrieron varias alertas que evitaron desastres, pero la más relevante fue un derrumbe en la cara principal del túnel, en donde gracias a la alerta temprana y al plan de emergencia, en donde se evacuó al personal, no existieron víctimas fatales.

Es por esto que el empleo de Digital Twin conlleva una serie de beneficios significativos en múltiples ámbitos, ya que permiten realizar un seguimiento detallado del comportamiento y el rendimiento de los sistemas físicos, lo que facilita la identificación de áreas de mejora y la optimización de procesos. Además, mediante el análisis de datos en tiempo real, los Digital Twin posibilitan la predicción de fallos y la implementación de acciones de mantenimiento proactivo, lo que reduce los tiempos de inactividad y los costos asociados. Por último, la capacidad de visualizar y analizar datos en tiempo real proporciona a los tomadores de decisiones información valiosa para la toma de decisiones estratégicas y operativas, aumentando la eficiencia y la eficacia en la gestión empresarial.

3.4. Tipos de Fallas en Sistemas de Chancado Minero

Con un enfoque en la operación de una planta de chancado minero, diversos tipos de fallas pueden ocurrir, cada una con sus propias causas y consecuencias. Identificar y comprender estos tipos de fallas es crucial para mejorar la eficiencia operativa, reducir tiempos de inactividad y minimizar costos de mantenimiento.

A continuación se presentan los principales tipos de fallas que pueden ocurrir en estos sistemas. Se debe tener presente que en la realidad existen diversos tipos de fallas dependiendo de múltiples factores, por ejemplo, de la ubicación del sistema.

3.4.1. Fallas Mecánicas

Las fallas mecánicas son comunes en los sistemas de chancado y pueden incluir roturas de componentes, desgaste excesivo, y problemas de alineación. Estas fallas son a menudo causadas por factores como la fatiga de los materiales, sobrecargas, y falta de lubricación adecuada. Por ejemplo, la rotura de un eje puede detener completamente la operación de un chancador [25].

Estas fallas pueden propagarse a través de conexiones físicas y mecánicas entre componentes. Por ejemplo, una falla en un rodamiento puede causar vibraciones excesivas que, a su vez, pueden dañar otros rodamientos o estructuras cercanas [26].

3.4.2. Fallas Eléctricas

Las fallas eléctricas abarcan desde cortocircuitos y fallas en el aislamiento hasta problemas con los sistemas de control y motores eléctricos. Estas fallas pueden ser provocadas por fluctuaciones en el suministro de energía, sobrecargas, o fallos en los componentes electrónicos.

Se pueden propagar rápidamente estas fallas a través de la red de suministro eléctrico. Un cortocircuito en un motor puede causar sobrecargas en los sistemas de control eléctrico, afectando otros motores y equipos conectados.

3.4.3. Fallas Térmicas

Las fallas térmicas resultan del sobrecalentamiento de componentes críticos debido a insuficiente ventilación, acumulación de polvo, o fallos en los sistemas de refrigeración. Un motor que se sobrecalienta puede llevar a una parada de emergencia del sistema para prevenir daños mayores.

El sobrecalentamiento de un componente puede causar un aumento de la temperatura en los componentes adyacentes, llevando a fallas térmicas en cadena. La acumulación de calor es especialmente problemática en ambientes confinados sin adecuada ventilación [26].

3.4.4. Prácticas Incorrectas

La falta de lubricación o el uso de lubricantes inapropiados pueden causar un aumento de la fricción y el desgaste de los componentes, llevando a fallas prematuras. La contaminación del lubricante también puede desempeñar un papel crucial en la degradación de los componentes.

Errores humanos y prácticas operativas incorrectas pueden desencadenar fallas en el sistema. Esto incluye el mal manejo de equipos, operación fuera de los parámetros recomendados, y falta de mantenimiento regular.

3.4.5. Propagación de Fallas

La propagación de fallas puede tener consecuencias significativas en términos de costos, seguridad y eficiencia operativa. Puede resultar en paradas no planificadas, tiempo de inactividad prolongado y la necesidad de reparaciones extensivas. Además, puede poner en riesgo la seguridad del personal.

Para lo anterior, existen diversas formas de mitigar el impacto de la propagación de fallas en sistemas, como por ejemplo, el uso de tecnologías de monitoreo en tiempo real, como sensores de vibración y temperatura, los cuales puede ayudar a detectar fallas incipientes antes de que se propaguen [26]. Otra técnica es diseñar sistemas con redundancia y mecanismos de aislamiento, ya que estos puede limitar la propagación de fallas. Implementar análisis de fiabilidad, como el Análisis de Modos y Efectos de Falla (FMEA), puede ayudar a identificar puntos críticos donde las fallas son más probables. Además de esto, es importante siempre asegurar que el personal esté bien capacitado y siga procedimientos operativos estándar puede reducir la incidencia de fallas operacionales y minimizar su propagación.

3.5. Modelos de Predicción de Fallas

Recientemente, se han llevado a cabo diversos estudios sobre modelos de predicción de fallas en el sector minero. En esta industria, los tiempos de inactividad no planificados causados por fallas en el equipo son extremadamente costosos, ya que pueden paralizar por completo la producción y provocar pérdidas financieras significativas. La maquinaria utilizada en minería es compleja y opera en condiciones adversas, lo que incrementa la probabilidad de fallas inesperadas. La utilización de modelos avanzados para predecir fallas permite anticiparse a estos problemas, optimizando el mantenimiento preventivo y reduciendo costos operativos.

En el trabajo de Mirzehi Kalateh Kazemi et al. (2023), se aborda el uso de técnicas metaheurísticas basadas en el modelo de Extreme Gradient Boosting (XGB) para predecir el tiempo hasta la falla de la maquinaria en minería [27]. Szmigiel et al. (2024) se enfocan en mejorar la predicción de la estabilidad de los espacios de extracción en ingeniería minera mediante modelos de redes neuronales artificiales (ANN), logrando mejoras significativas en esta área. Los autores concluyeron que la configuración óptima de la ANN proporcionó mayor precisión en las predicciones en comparación con los métodos convencionales [28]. En el estudio de Rafezi y Hassani (2023), se aplicó un modelo de red neuronal para la predicción de fallas y el monitoreo del desgaste de las brocas de perforación. Este enfoque permite modelar relaciones no lineales complejas entre variables de entrada, como la vibración y la tasa de penetración, mejorando así la precisión en la predicción del desgaste de las brocas [29].

El uso de las ANN ha mostrado resultados positivos en varios estudios relacionados con la predicción de fallas en minería. La investigación de Mirzehi Kalateh Kazemi et al. (2023), Szmigiel et al. (2024) y Rafezi y Hassani (2023) subraya el potencial de estos modelos para anticipar problemas y optimizar el mantenimiento de la maquinaria, lo que se traduce en una reducción de costos operativos y en una mejora de la eficiencia del proceso productivo. Dada la eficacia demostrada de las ANN, se sugiere evaluar si un modelo de ANN que considere las dependencias entre componentes superará en rendimiento a un modelo que no las incluya. Esta comparación es esencial, ya que podría ofrecer nuevas perspectivas sobre cómo maximizar la precisión de las predicciones y minimizar los tiempos de inactividad en la industria minera.

4 | Metodología

4.1. Estructura de la Investigación

Este trabajo de tesis se organiza en capítulos independientes. El capítulo 1 contiene el resumen del estudio, el capítulo 2 muestra la introducción a la investigación, el capítulo 3 determina el objetivo general del estudio y los específicos. El 4 enseña el marco teórico, dejando plasmado el estudio realizado y sentando las bases para el capítulo 5, el cual presenta la metodología que expresa la estructura del escrito y los pasos a seguir. El capítulo 6 muestra los resultados obtenidos, el 7 resume las discusiones y el alcance del estudio. Finalmente se presentan en el capítulo 8 las conclusiones, además de sugerir posibles direcciones para trabajos futuros.

4.2. Marco de Trabajo Metodológico

A continuación, se exponen los puntos esenciales de la metodología desarrollada, la cual se divide en 5 secciones principales: la descripción del sistema de estudio, el análisis y procesamiento de los datos, los modelos de dependencia a implementar según lo estudiado, los indicadores de desempeño por utilizar para la validación de los datos, y finalmente, el desarrollo del modelo de predicción de fallas.

4.3. Descripción del Sistema en Estudio

El sistema en estudio es una línea de chancado de una minera de cobre ubicada en el Norte de Chile, la cual está compuesta principalmente por el Chancador N°5, una correa transportadora CV-1C, un alimentador o feeder FE002 y otra correa transportadora denominada CV-2C, tal como se muestra en la Figura 4.1.

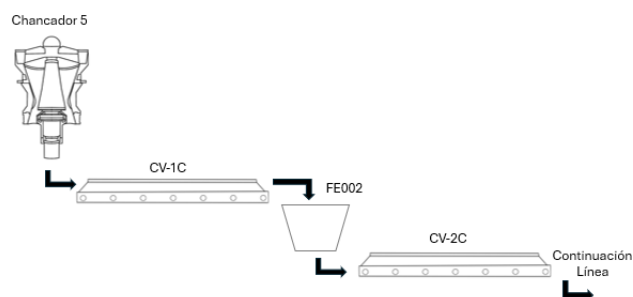


Figura 4.1: Línea de Chancado Minero

La línea de chancado es una etapa crucial en la minería ya que esta tiene como objetivo principal reducir el tamaño del mineral extraído para facilitar su procesamiento en etapas posteriores. Aguas arribas de la etapa del chancado se encuentra la extracción y el transporte del mineral. En la extracción se tienen dos métodos principales de explotación: minería a cielo abierto y minería subterránea, en donde la elección del método depende de la ubicación y la profundidad del depósito de cobre. Por otro lado, aguas abajo se tiene la concentración, refinación, transporte y comercialización.

Los equipos de la Figura 4.1 están conectados mediante sensores a un Sistema de Control Distribuido (DCS), el cual es un sistema de control automatizado que consta de elementos de control distribuidos de forma geográfica en la planta o área de control, obteniendo así, toda la información necesaria que puede ser utilizada para controlar los procesos industriales y lograr aumentar su seguridad, rentabilidad y fiabilidad.

Este a su vez está conectado a un Digital Twin, el cual replica a tiempo real todo el proceso mediante los indicadores que va recibiendo del DCS. Cuando el Digital Twin recibe información que no es común, significa que algo está ocurriendo en alguna parte del sistema y por ende, esta información se dirige a una base de mantención, tal como se puede ver en la Figura 4.2.

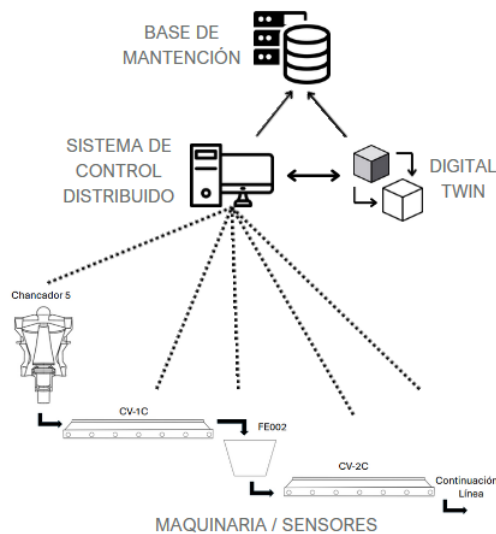


Figura 4.2: Sistema Integrado en Estudio

En la base de mantención, se puede observar que sólo se tiene información del Chancador N°5 y de la correa transportadora CV-2C, pero existen muchos más datos para el Chancador N°5 (31.834 datos versus 9.613 datos) y para el estudio de dependencia sólo usaremos un equipo, ya que así se pueden establecer las fallas que ocurren en este equipo dada toda la recopilación de las variables numéricas, por lo que solamente se utiliza la información de este último equipo para lo que sigue del estudio.

4.4. Limpieza y Procesamiento de Datos

Para comenzar con el análisis de los datos, lo primero que se realiza es la unificación de los datos de la línea con los de la base de mantención, en donde se tiene que los datos son controlados cada un minuto.

La base unificada presenta 547.953 datos históricos desde el primero de agosto de 2020 a las 00:00:00 horas, hasta el 30 de agosto de 2021 a las 23:59:00 horas, con datos cada un minuto, como se mencionó anteriormente. Dentro de las variables existentes se tiene que una es variable de tiempo, 36 son de carácter numérico y 6 variables son cualitativas, teniendo una cantidad total de 43 variables agrupadas por columnas.

Al realizar una observación de los datos, estos se filtraron para que solamente quede información respecto de las detenciones del Chancador N°5 y además, se tomaron las precauciones para que las detenciones no fueran por colación del operador o cambio de turno, quedando así 31.834 registros válidos.

A continuación de lo anterior, para que en la etapa de la codificación del modelo no existan problemas, se eliminan las columnas nulas ya que no contienen datos relevantes para el estudio, se filtran variables numéricas que presentan los mismos datos unas de otras y luego se realiza una normalización y estandarización de las variables categóricas y numéricas respectivamente.

Dentro de las variables numéricas, lo que se hizo específicamente fue, en una primera instancia, observar estos datos y ver la descripción que tenían cada una, las cuales se presentan en la Tabla 8.1 en la sección de Anexos.

Con esto, se tiene que las variables HSRS son las eliminadas porque todas estas columnas no contienen información. Luego, dada la descripción, se observa que las variables desde la AD6A hasta la AD6H tienen el mismo significado, por ende, lo que se realiza es un análisis de comportamiento de estas en relación al tiempo, en donde los resultados se aprecian en la Figura 4.3.

Desde el comportamiento, se ve que la variable AD73 es diferente al resto, además de que su descripción también lo es, por ende, esta queda dentro del análisis. Luego, la variable AD6E presenta un comportamiento distinto, variando sus registros en un rango de 0,04, lo que en comparación a la variación de las demás, es muy bajo, por lo que esta variable también se incluye en el estudio.

Finalmente, todas las demás variables AD presentan un comportamiento casi idéntico, variando un poco en las escalas y exceptuando por algunos valores atípicos, por lo cual, en el código de Python se deja la primera variable como la representante de este conjunto de 7 que son muy parecidas, que en este caso es la variable AD6A.

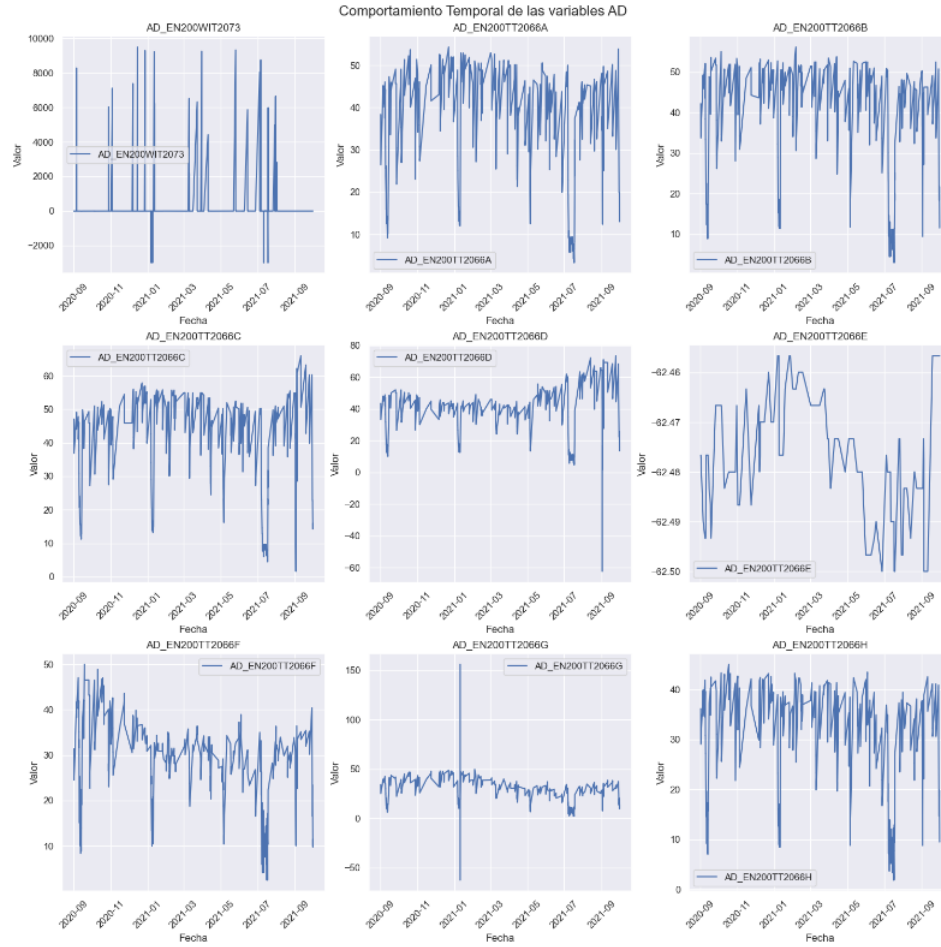


Figura 4.3: Comportamiento Temporal Variables AD

4.5. Análisis y Evaluación de Datos

En esta etapa se evalúan los datos de entrada para los modelos de Machine Learning, en donde, lo que se observa primero son las 6 variables categóricas, las cuales son: Equipo, Motivo, Turno, Descripción, Línea y Clasificación. Dentro de la variable Descripción, se encuentran los diferentes tipos de eventos que pueden ocurrir dentro del sistema, dividiéndose en detenciones operacionales y en fallas, los cuales son el objetivo de estudio.

La Tabla 4.1 presenta todos los tipos de eventos. En esta, se puede apreciar que de los 21 tipos de detenciones, 10 son diferentes tipos de fallas y los otros 11 son las detenciones operacionales.

Evento	Tipo de Detención
Detención Operacional	Mtto. programado
	Detencion no programadas aguas abajo
	Detencion no programadas aguas arriba
	Reserva de alimentación prioritaria
	Extension mantencion programada
	Tronadura
	Alto stock
	Detector de metales
	Sensor nivel
Falla	Atollo
	Tilt switch
	Operacional no programada
	Fallas mecanicas
	Corte correa electrico
	Falla no clasificada
	Pull cord
	Fallas electricas
	Desalineamiento de correa
	Corte correa mecanico
	Fallas de servicios de terceros
	Material extraño

Tabla 4.1: Clasificación de Tipos de Detención en Operacionales o Falla

De todos estos eventos, sólo 16 afectan al Chancador N°5. Estos se trabajan y se verifica la frecuencia de cada uno para ver cuántos registros se tienen, en donde se observa que 87,5 % de los datos pertenecen a mantenimiento programado tal como se aprecia en la Figura 4.4a Por otro lado, se analiza la frecuencia acumulada hasta el 98 % y se observa que 6 eventos son los que acumulan este porcentaje.

Dentro de los varios experimentos que se realizaron, en un comienzo se observó que al introducir estos 6 eventos a los modelos de Machine Learning, para los eventos que son detenciones operacionales, las variables más influyentes no tenían sentido, lo que se debe a que las detenciones operacionales son planificadas, por ende, no existe algún tipo de relación con las variables numéricas del estudio.

Es por lo anterior, que se decide utilizar para la investigación los eventos más frecuentes que no fueran planificados, en otras palabras, que sean fallas, obteniendo así como resultado las Fallas Mecánicas y las Detenciones Operacionales No Planificadas.

En la Figura 4.4b se muestra la frecuencia de los tipos de fallas escogidos para continuar con el estudio.

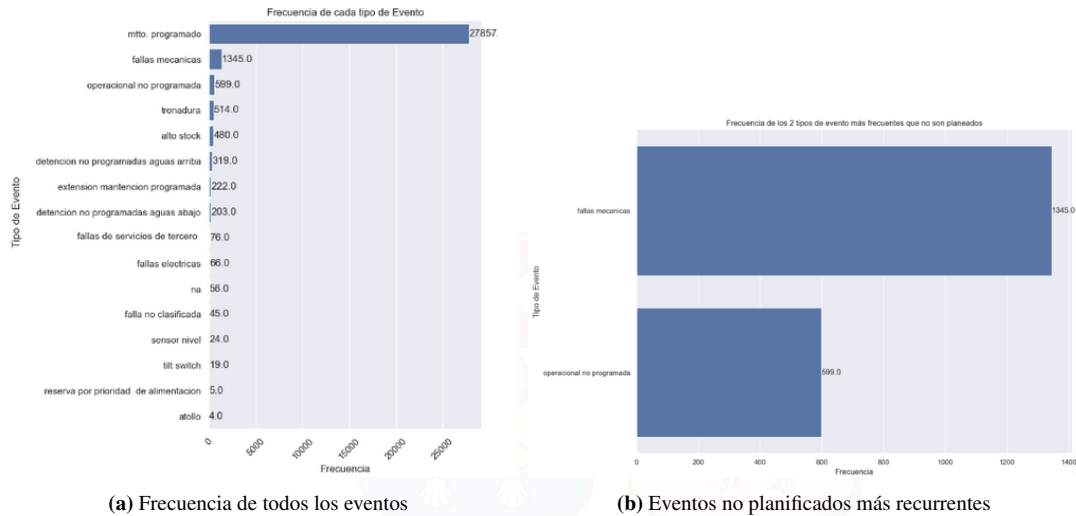


Figura 4.4: Frecuencia de los eventos para el Chancador N°5

Finalmente a los modelos de ML ingresan 31.834 filas de datos, 27 variables numéricas y 2 variables categóricas.

4.6. Modelos de Dependencia a Implementar

De la investigación se puede observar que los modelos de ML mayormente utilizados para realizar un análisis de dependencia e interdependencia entre componentes de un sistema son la Regresión Logística (LR), Support Vector Machine (SVM), Decision Tree (DT), RF y ANN. Dado estos modelos, según la literatura y sus resultados, se tiene que el modelo de RF y las ANN son los que mejores indicadores tienen en términos de clasificación [14].

Es por esto, que para el desarrollo de la investigación, y la obtención de una hipótesis sobre cómo son las relaciones entre los componentes numéricos del sistema y los distintos tipos de falla que pueden existir, se van a implementar los dos modelos nombrados anteriormente, en donde, el que tenga mejores indicadores será el que continúe y se implemente como hipótesis de un modelo de análisis estadístico multivariado, el cual puede interpretar las dependencias e interdependencia entre variables.

4.6.1. Modelo de Random Forest

Como se mencionó en el capítulo anterior, RF es un algoritmo de aprendizaje automático que se utiliza principalmente para tareas de clasificación y regresión. Este método aprovecha la variabilidad de los árboles de decisión individuales para reducir la varianza total y, por lo tanto, mejorar la precisión y la robustez del modelo.

Una de las razones de escoger este modelo es por que debido a que combina los resultados de múltiples árboles de decisión, Random Forest tiende a no sobre ajustarse a los datos de entrenamiento,

lo cual es una ventaja significativa en comparación con un solo árbol de decisión. Otro punto es que puede manejar datos faltantes en los predictores y mantiene una buena precisión incluso cuando hay valores perdidos. Además, proporciona una medida de la importancia de las variables, lo que es útil para entender las contribuciones de diferentes características en la predicción, lo que es de mucha utilidad para esta investigación [30].

Por otro lado, se debe comentar que los modelos de RF también presentan ciertas desventajas como, por ejemplo, que puede ser más complejo y difícil de interpretar que un solo árbol de decisión, y además, entrenar y hacer predicciones con un Random Forest puede ser computacionalmente intensivo, especialmente con un gran número de árboles y características, o en otras palabras, es costoso computacionalmente.

Para aplicarlo en el conjunto de datos, lo primero y esencial es preparar los datos, lo cual implica limpiar el conjunto de datos, tal como se presentó en la sección anterior. Luego, se dividen los datos en conjuntos de entrenamiento y prueba, en una proporción de 70/30 respectivamente, para evaluar el funcionamiento del modelo de manera justa.

Para el entrenamiento del modelo se utiliza el algoritmo de RF con una cantidad significativa de árboles, en este caso se incorporan 500, para asegurar la estabilidad del modelo. Luego, durante el entrenamiento, se seleccionan subconjuntos de características y datos para construir cada árbol mediante la técnica de Bagging [10].

Matemáticamente el modelo se representa de la siguiente manera:

$$\hat{f}(x) = \frac{1}{B} \sum_{b=1}^B T_b(x)$$

Donde B es el número de árboles, T_b es el b -ésimo árbol y $\hat{f}(x)$ es la predicción agregada [10].

A continuación, la precisión del modelo se evalúa utilizando métricas como la matriz de confusión, accuracy, precisión, recall, y el F1-score. En donde, además se genera una matriz de confusión para observar el desempeño del modelo en la clasificación de cada tipo de falla. Sobre estos indicadores de desempeño se profundiza más en la siguiente sección.

Luego, para realizar la validación del modelo, se implementa una validación cruzada k-fold para evaluar la estabilidad y generalización del modelo. Y, por otro lado, se prueba el modelo en datos recientes no vistos (los datos de prueba) para asegurar que puede detectar fallas de manera efectiva en un entorno real.

Finalmente, se interpreta la importancia de las variables con SHAP (SHapley Additive ex-Planations), la cual es una técnica utilizada para interpretar y explicar los resultados de modelos de aprendizaje automático. Específicamente, ayuda a desglosar las predicciones del modelo en contribuciones individuales de las características de entrada, permitiendo identificar qué variables han tenido mayor impacto en una predicción determinada.

SHAP está basado en la teoría de valores de Shapley, que fue desarrollada por Lloyd Shapley en

el contexto de la teoría de juegos. En este marco, el valor de Shapley asigna una contribución justa a cada jugador (en este caso, las variables o características) en función de su participación en un resultado determinado[31]. Actualmente, es ampliamente utilizado para entender el funcionamiento de modelos de caja negra, especialmente aquellos basados en técnicas avanzadas como árboles de decisión, redes neuronales o modelos de boosting.

En el caso de estudio, lo que se realiza es iterar sobre cada evento no planificado descrito anteriormente, para identificar las cinco variables más influyentes en la predicción de dicho evento.

4.6.2. Modelo de Redes Neuronales Artificiales

Las ANN (por sus siglas en inglés) son un conjunto de algoritmos inspirados en el funcionamiento del cerebro humano. Se componen de capas de neuronas (o nodos), que están conectadas entre sí por "pesos". Estos pesos se ajustan durante el entrenamiento para minimizar el error de predicción. Las ANN son capaces de modelar relaciones complejas y no lineales entre las variables de entrada y salida, lo que las hace adecuadas para tareas de clasificación y regresión.

Al igual que con el modelo de RF, el primer paso es cargar los datos ya procesados anteriormente. En consiguiente, se define la arquitectura de la ANN mediante programación. Para este propósito, se debe proporcionar una interfaz fácil de usar para construir y entrenar redes neuronales profundas.

Para los datos de entrada al modelo, se asignan como variables categóricas los dos eventos no planificados más frecuentes, y se dividen los datos en conjuntos de entrenamiento (70 %) y prueba (30 %). Luego, verifica si hay una GPU disponible para acelerar los cálculos, si no la hay, utiliza la CPU. Esto debido a que el modelo ANN es muy costoso computacionalmente.

Convierte los datos a tensores de PyTorch para entrenar el modelo y a continuación se crea una red con tres capas lineales, logrando definir la arquitectura de la ANN. La primera capa tiene tantas entradas como variables numéricas, y la última tantas salidas como clases de falla.

Se define una tasa de aprendizaje de 0.001 y un número de 500 épocas para el entrenamiento del modelo. Se utilizan estas ya que generalmente en modelos con una gran base de datos, estos valores son los más utilizados y los que mejor se ajustan a la red.

Ahora si se comienza a entrenar el modelo, durante 500 épocas o iteraciones, en donde, en cada una de estas se predicen las etiquetas, se calcula la pérdida, se ajustan los pesos mediante retropropagación y se actualizan los gradientes.

Luego, se evalúa la precisión del modelo utilizando diversas métricas como la matriz de confusión, accuracy, precisión, recall, y el F1-score, al igual que en el modelo de Random Forest. Esto proporciona una visión general del rendimiento del modelo en la clasificación de los diferentes tipos de fallas.

Para terminar, se obtienen las cinco variables más importantes para cada tipo de falla o evento no planificado, de la misma manera que en el modelo anterior. Esta parte es muy importante ya que

es la que entregará la hipótesis de las relaciones de dependencia entre las variables numéricas y los eventos dentro del sistema.

4.6.3. Modelo de Ecuaciones Estructurales

Luego de codificar y obtener los resultados de los modelos de algoritmos de aprendizaje automático, se escoge el que presente los mejores indicadores de desempeño, considerando además el tiempo en que demoran en correr los algoritmos, para evaluar posteriormente si el modelo se puede implementar en tiempo real y en un futuro lograr predecir fallas en el sistema.

Los resultados de este modelo serán de utilidad para implementarlos como la hipótesis de las relaciones de dependencia e interdependencia en el Modelo de Ecuaciones Estructurales (SEM), ya que el Random Forest o las Redes Neuronales Artificiales entregarán las relaciones entre los componentes numéricos, como por ejemplo la potencia del motor, temperaturas, velocidades, etc. y los diferentes tipos de fallas.

En la etapa de definición, se identifican las variables endógenas (dependientes) y exógenas (independientes) y se especifica la estructura de las relaciones entre ellas.

Las variables endógenas en este caso serán los tipos de fallas del sistema, mientras que las variables exógenas serán los componentes numéricos del sistema. Las relaciones entre estas variables se representan mediante ecuaciones estructurales que se pueden formular como:

$$Y_i = \beta_{0i} + \sum_{j=1}^P \beta_{ij} X_j + \epsilon_i$$

Donde Y_i es la variable dependiente (tipo de falla), X_j son las variables independientes (componentes numéricos), β_{ij} son los coeficientes de regresión, y ϵ_i es el término de error.

En la etapa de especificación, se detallan las relaciones entre las variables. Esto implica la definición de las variables latentes, así como las relaciones de causalidad o correlación entre ellas (entregadas por el modelo de Machine Learning). En SEM, el modelo se divide en dos partes principales: el modelo de medición y el modelo estructural.

- **Modelo de Medición:** Especifica cómo las variables latentes se relacionan con las variables observadas. Por ejemplo, una variable latente como "falla mecánica" puede estar medida por variables observadas como temperatura y potencia del motor.

$$X_i = \lambda_{i1} \eta_1 + \lambda_{i2} \eta_2 + \zeta_i$$

Donde X_i son las variables observadas, η_j son las variables latentes, λ_{ij} son las cargas factoriales, y ζ_i es el error de medición.

- **Modelo Estructural:** Define las relaciones entre las variables latentes. Esto incluye las relaciones

de causalidad entre las variables latentes y sus efectos sobre las variables observadas.

$$\eta_i = \Gamma\eta_j + \Lambda X + \zeta$$

Donde η_i son las variables latentes, Γ es la matriz de coeficientes de las relaciones entre las variables latentes, Λ es la matriz de coeficientes de las relaciones entre variables latentes y observadas, y ζ es el término de error estructural.

Una vez que el modelo ha sido especificado, el siguiente paso es estimar los parámetros utilizando los datos disponibles. Para ello, se emplean técnicas de estimación como la Máxima Verosimilitud (ML) o la Estimación de Mínimos Cuadrados (OLS).

La estimación de los parámetros se realiza para minimizar la diferencia entre la covarianza observada y la covarianza esperada bajo el modelo propuesto. La función objetivo se puede formular como:

$$\text{Minimizar } F(\Theta) = \|\mathbf{S}_{\text{obs}} - \mathbf{S}_{\text{model}}(\Theta)\|^2$$

Donde $\mathbf{S}_{\text{obs}}$ es la matriz de covarianza observada, $\mathbf{S}_{\text{model}}(\Theta)$ es la matriz de covarianza modelada en función de los parámetros Θ , y $\|\cdot\|$ es la norma de Frobenius.

Luego de obtener los resultados, la evaluación del modelo es crucial para validar su ajuste y robustez. Se utilizan varios índices para evaluar el ajuste del modelo, tales como chi-cuadrado, Índice de Ajuste Comparativo (CFI) y Índice de Bondad de Ajuste (GFI) y por último la Raíz del Error Cuadrático Medio de Aproximación.

Finalmente, se interpretan los parámetros del modelo para entender las relaciones causales entre las variables y validar la consistencia de los resultados. Esto implica examinar los coeficientes de regresión, las cargas factoriales y los errores, y además, considerar la robustez del modelo.

4.7. Indicadores de Desempeño y Validación de los Resultados

Como se vio en la sección anterior, para evaluar la precisión del modelo de Random Forest y el modelo de Redes Neuronales Artificiales, se utilizan diversos indicadores de desempeño, como lo son la matriz de confusión, accuracy, precisión, recall, y el F1-score. A continuación se detallan cada uno de estos indicadores verificando su importante función dentro del estudio.

1. **Matriz de Confusión:** La matriz de confusión es una herramienta que permite visualizar el rendimiento de un algoritmo de clasificación. Se trata de una tabla que compara las predicciones del modelo con las verdaderas etiquetas de clase. Cada fila de la matriz representa las instancias de una clase predicha, mientras que cada columna representa las instancias de una clase real (o viceversa) [32]. A continuación se presentan las 4 clasificaciones posibles.

- **True Positive (TP):** Número de instancias correctamente clasificadas como positivas.

- **True Negative (TN):** Número de instancias correctamente clasificadas como negativas.
 - **False Positive (FP):** Número de instancias incorrectamente clasificadas como positivas.
 - **False Negative (FN):** Número de instancias incorrectamente clasificadas como negativas.
2. **Accuracy:** La accuracy o exactitud mide la proporción de predicciones correctas (tanto verdaderos positivos como verdaderos negativos) en relación con el total de predicciones realizadas. Es una medida general del rendimiento del modelo.

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

3. **Precisión:** La precisión (también llamada valor predictivo positivo) es la proporción de verdaderos positivos entre el total de instancias clasificadas como positivas. Mide la exactitud de las predicciones positivas del modelo.

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

4. **Recall:** El recall (o sensibilidad) mide la proporción de verdaderos positivos en relación con el total de casos positivos reales. Indica cuán bien el modelo es capaz de identificar los casos positivos.

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

5. **F1-Score:** Es la media armónica de la precisión y el recall. Proporciona una medida equilibrada del rendimiento del modelo, especialmente útil cuando las clases están desbalanceadas.

$$\text{F1-Score} = 2 \times \frac{\text{Precisión} \times \text{Recall}}{\text{Precisión} + \text{Recall}}$$

Para validar el ajuste y robustez del Modelo SEM, se utilizan varios índices. Además, estos ayudan a determinar si el modelo especificado es una representación adecuada de las relaciones entre las variables.

1. **Grados de Libertad:** Son un concepto estadístico que se refiere al número de valores en un cálculo que son libres de variar. Los grados de libertad dependen del tipo de prueba estadística que se realicen y están relacionados con el tamaño de la muestra y las restricciones impuestas por la hipótesis nula.

Un ejemplo de aplicación es en la prueba de chi-cuadrado, aquí los grados de libertad son importantes para determinar la distribución que debe usarse para comparar el valor del estadístico chi-cuadrado y calcular el valor p. Para la prueba de independencia de este, los grados de libertad se calculan como:

$$df = (r - 1) \times (c - 1)$$

Donde r es el número de filas en la tabla y c es el número de columnas en la tabla.

2. **Chi-Cuadrado (χ^2):** Es una medida estadística que se utiliza para evaluar la relación entre

variables categóricas o la bondad de ajuste de un modelo. Existen varias pruebas basadas en el chi-cuadrado, como la prueba de independencia y la prueba de bondad de ajuste. La fórmula general para calcular el valor de chi-cuadrado es:

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

Donde, O_i es el valor observado para la categoría i , E_i es el valor esperado para la categoría i , y finalmente, la sumatoria se hace sobre todas las categorías.

La distribución chi-cuadrado se utiliza para obtener un valor p y determinar la significancia estadística del resultado de la prueba.

3. **P-Value:** Es una medida estadística que indica la probabilidad de obtener un resultado al menos tan extremo como el observado, bajo la suposición de que la hipótesis nula es cierta. Se utiliza para determinar si los resultados de un experimento o estudio son estadísticamente significativos.

La hipótesis nula, es la que indica que no hay un efecto o relación significativa en los datos. Generalmente, esta hipótesis asume que cualquier patrón observado en los datos es solo producto del azar.

El valor p muestra cuán probable es observar los datos si la hipótesis nula fuera verdadera. Cuanto más pequeño sea el valor p , más evidencia se tiene en contra de la hipótesis nula. Matemáticamente, el nivel de significancia es el umbral que se usa para decidir si se rechaza o no la hipótesis nula. Un valor comúnmente usado es $\alpha = 0.05$, donde:

Si $p \leq \alpha$, se rechaza la hipótesis nula.

Si $p > \alpha$, no se rechaza la hipótesis nula.

4. **Comparative Fit Index (CFI):** Indica cuán bien se ajusta un modelo teórico a los datos reales, en comparación con un modelo donde no se especifican relaciones entre las variables (modelo nulo). Un valor alto de CFI sugiere que el modelo teórico representa adecuadamente las relaciones observadas en los datos. La fórmula básica del CFI es la siguiente:

$$CFI = 1 - \frac{\chi^2_{\text{modelo}} - df_{\text{modelo}}}{\chi^2_{\text{nulo}} - df_{\text{nulo}}}$$

Donde χ^2_{modelo} es el valor chi-cuadrado del modelo propuesto, df_{modelo} son los grados de libertad del modelo propuesto, χ^2_{nulo} es el valor chi-cuadrado del modelo nulo (sin relaciones entre variables) y por último, df_{nulo} son los grados de libertad del modelo nulo.

El CFI ajusta los valores del chi-cuadrado en función de los grados de libertad de cada modelo, lo que lo hace menos sensible al tamaño de la muestra que otras medidas de ajuste.

Valores cercanos a 1, indican un buen ajuste del modelo. Generalmente, se considera que un valor de $CFI \leq 0.95$ sugiere un excelente ajuste, y valores $CFI \geq 0.90$ indican un ajuste adecuado.

Finalmente, a diferencia de otras medidas de ajuste como el valor chi-cuadrado, el CFI es menos afectado por el tamaño de la muestra. Esto lo hace útil cuando se trabaja con muestras grandes, donde el chi-cuadrado tiende a rechazar modelos aceptables.

5. **Tucker-Lewis Index (TLI):** Compara el ajuste de un modelo propuesto con un modelo nulo (que asume que no hay relaciones entre las variables), pero penaliza los modelos más complejos que tienen más parámetros. Este ajuste por complejidad hace que el TLI favorezca los modelos más simples cuando otros factores son iguales. La fórmula para calcular el TLI es:

$$TLI = \frac{\frac{\chi^2_{nulo}}{df_{nulo}} - \frac{\chi^2_{modelo}}{df_{modelo}}}{\frac{\chi^2_{nulo}}{df_{nulo}} - 1}$$

Donde valores cercanos a 1, indican un buen ajuste del modelo. En general, un valor de $TLI \geq 0.95$ se considera excelente, y valores entre 0.90 y 0.95 indican un ajuste aceptable.

6. **Root Mean Square Error of Approximation (RMSEA):** Estima la discrepancia por grado de libertad entre el modelo especificado y el modelo "perfecto" (un modelo en el que la covarianza predicha por el modelo coincide exactamente con la covarianza observada en los datos). Básicamente, mide el error de aproximación del modelo ajustado a los datos.

Un valor bajo de RMSEA indica un buen ajuste, mientras que un valor alto indica un mal ajuste. La fórmula para calcular el RMSEA es:

$$RMSEA = \sqrt{\frac{\chi^2_{modelo} - df_{modelo}}{df_{modelo}(N - 1)}}$$

Donde, χ^2_{modelo} es el valor chi-cuadrado del modelo propuesto, df_{modelo} son los grados de libertad del modelo y por último, N es el tamaño de la muestra.

El RMSEA se interpreta con ciertos umbrales, los cuales son: si $RMSEA < 0.05$ indica un ajuste excelente entre el modelo y los datos. Se considera un modelo bien ajustado. Si $0.05 \leq RMSEA \leq 0.08$ indica un ajuste aceptable. Por último, si $RMSEA > 0.08$ sugiere un ajuste mediocre o pobre.

El RMSEA tiene en cuenta el número de parámetros en el modelo y penaliza los modelos más complejos. Esto ayuda a evitar el sobreajuste, que ocurre cuando un modelo demasiado complicado ajusta bien los datos, pero tiene poca capacidad de generalización. Además, es una medida robusta que puede aplicarse incluso en muestras grandes, donde otros indicadores tienden a volverse demasiado estrictos.

7. **Standardized Root Mean Square Residual (SRMR):** Indica la discrepancia promedio entre las correlaciones observadas y las predichas en un modelo ajustado. Específicamente, se basa en los residuos estandarizados, lo que lo convierte en una métrica intuitiva para cuantificar el error promedio entre el modelo y los datos.

Un SRMR bajo sugiere que el modelo ajusta bien los datos, mientras que un SRMR alto indica que hay una discrepancia significativa entre el modelo y los datos observados. La fórmula del

SRMR es:

$$SRMR = \sqrt{\frac{1}{p(p+1)/2} \sum_{i \neq j} (r_{ij} - \hat{r}_{ij})^2}$$

Donde r_{ij} son las correlaciones observadas entre los indicadores i y j , $\hat{r}_{ij}$ son las correlaciones predichas por el modelo entre i y j , por último, p es el número de variables en el modelo.

Como interpretación se tiene que cuando el $SRMR \leq 0.08$, se considera un buen ajuste. Un valor cercano a 0 indica una mínima discrepancia entre los datos observados y el modelo. Por otro lado, cuando el $0.08 < SRMR < 0.10$, puede ser aceptable, pero el ajuste no es tan bueno como cuando el SRMR es menor a 0.08.

A diferencia de otras medidas de ajuste, el SRMR es menos sensible al tamaño de la muestra, lo que lo hace adecuado para comparaciones entre modelos que usan diferentes tamaños de muestra.

8. **Matriz de regresiones:** Es una representación tabular que muestra las relaciones de dependencia entre varias variables en un modelo estadístico, generalmente en el contexto de un modelo de ecuaciones estructurales (SEM). Estas relaciones se expresan en forma de ecuaciones de regresión, donde una o más variables dependientes (endógenas) se predicen o explican en función de una o más variables independientes (exógenas).

En la matriz, las filas y columnas representan variables, y las celdas contienen los coeficientes de regresión que cuantifican el efecto de una variable sobre otra.

Los coeficientes en la matriz indican la fuerza y la dirección de las relaciones entre las variables. Si un coeficiente es positivo, indica que un aumento en la variable independiente provoca un aumento en la variable dependiente, mientras que un coeficiente negativo indica una relación inversa.

9. **Matriz de covarianzas:** Es una tabla que muestra las covarianzas entre todas las posibles combinaciones de variables en un conjunto de datos. La covarianza es una medida estadística que indica la relación de variación conjunta entre dos variables, es decir, cómo varía una variable cuando la otra cambia.

Por otro lado, la covarianza entre dos variables X e Y mide la dirección en la que ambas variables tienden a moverse juntas. Si X e Y tienden a aumentar o disminuir simultáneamente, la covarianza será positiva. Si una aumenta mientras la otra disminuye, la covarianza será negativa. Si no hay una relación lineal clara entre las dos variables, la covarianza será cercana a cero.

10. **Matriz de varianzas:** Es una matriz diagonal en la que los elementos no diagonales son cero, y cada elemento en la diagonal representa la varianza de una variable individual. La varianza mide la dispersión de los valores de una variable con respecto a su media. Cuantifica cuánto varían o se alejan los valores de la media de la variable.

Mientras que la matriz de covarianzas incluye tanto varianzas como covarianzas (relaciones entre variables), la matriz de varianzas solo incluye las varianzas, es decir, la dispersión de cada variable individual con respecto a su media.

4.8. Modelo de Predicción de Fallas

El objetivo de incorporar un modelo de predicción de fallas que tenga en cuenta las relaciones de dependencia previamente identificadas es mejorar la capacidad predictiva en comparación con un modelo que no las considera. Esto se logra al integrar un entendimiento más profundo de cómo las variables se interrelacionan. Las relaciones de dependencia, que pueden ser capturadas a través de modelos SEM, permiten analizar cómo una o más variables influyen, ya sea directa o indirectamente, en el comportamiento de las fallas.

Para el modelo de predicción, se genera una variable binaria que indica la predisposición a la falla, así como su ausencia, de acuerdo a lo que se muestra en la Tabla 4.2, y se fundamenta en patrones históricos.

Variables de Estado	Codificación
Sin Predisposición a la Falla	0
Con Predisposición a la Falla	1

Tabla 4.2: Variable Binaria del Modelo de Predicción de Fallas

El código también implementa una ventana de tiempo mediante promedios móviles sobre las variables numéricas. Este enfoque captura la evolución de las variables a lo largo del tiempo, lo cual es fundamental para un análisis de series temporales y para identificar tendencias o fluctuaciones que pueden preceder a las fallas. En estudios de mantenimiento predictivo, este tipo de enfoque temporal permite que el modelo aprenda de cambios graduales y su impacto en el rendimiento del equipo [33].

Posteriormente, se integran las variables más influyentes para cada tipo de falla, seleccionadas según el modelo de análisis de dependencia con los mejores resultados. Este paso es crucial, ya que puede mejorar significativamente tanto el rendimiento como la interpretabilidad del modelo.

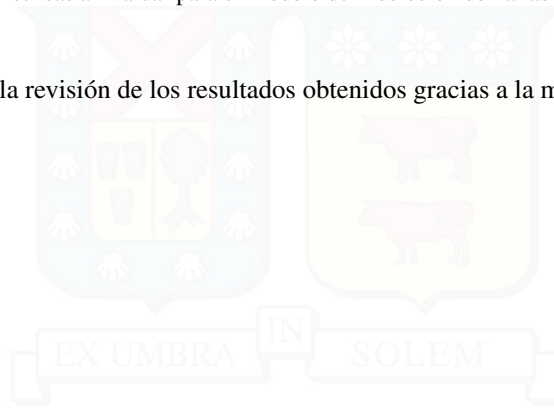
Los datos se dividen en una proporción de 70/30 y se realiza una estandarización posterior para asegurar que las variables numéricas tengan una escala uniforme, lo cual es esencial para que los modelos de redes neuronales converjan de manera eficiente durante el entrenamiento.

El modelo de predicción es una ANN secuencial que cuenta con varias capas densas y funciones de activación, lo que facilita la captura de relaciones no lineales entre las características y la variable objetivo. Para los modelos que consideran dependencias y para los que no, se evalúan distintas métricas que reflejan diversas combinaciones, tal como se muestra en la Tabla 4.3.

Ventana de Tiempo	Capas	Neuronas	Función	Tasa de Aprendizaje	Umbral
15	1	32	Tanh	0.0001	0.3
30	2	128	Sigmoid	0.00001	0.5
60	3				

Tabla 4.3: Métricas a Evaluar para el Modelo de Predicción de Fallas

Con esto, se da paso a la revisión de los resultados obtenidos gracias a la metodología explicada en este capítulo.



5 | Resultados

5.1. Datos de Entrada a los Modelos de Machine Learning

Se entrenan los modelos con las variables categóricas denominadas fallas mecánicas y operacionales no programadas. Además las variables numéricas se observan en la Tabla 5.1. Estos son los datos finales de entrada, los cuales, tal como se presentó en el capítulo anterior, fueron filtrados cuidadosamente.

Variable	Descripción	Variable	Descripción
AD73	Carga de la correa en tph del DCS	CVmp	Amperios Motor 2 DT
EN65	% de Velocidad de Correa DCS	CVst	Resistencia Diseñada de la Correa
SSM1	Amperios Motor 1 DCS	CVst	Resistencia Actual de la Correa
SSM2	Amperios Motor 2 DCS	CVor	Desviacion de la resistencia
AD6A	Temperature Transmite	CVed	% de Velocidad de Correa DT
AD6E	Temperature Transmite 2	CVor	Error de los amperios Motor 2
MTOT	Factor de Potencia Motor 1	CVOR	Error de los amperios Motor 1
MTLH	Potencia del Motor 1	CVue	Torque DT
MTOT	Factor de Potencia Motor 2	CVns	Toneladas totales DT
MTLH	Potencia del Motor 2	CVon	% De utilizacion de voltaje
CVph	Carga de la correa en tph del DT	CVon	% De utilizacion de Potencia
CVph	Error de carga de la correa en tph DT	CVHr	Performance Toneladas por Hora
CVer	Potencia DT (Dos motores)	CVft	Performance Toneladas por Turno
CVmp	Amperios Motor 1 DT		

Tabla 5.1: Variables Numéricas incorporadas en los modelos

La Tabla 5.1 muestra la abreviación de las variables y su descripción, para poder entender que significan cada una de estas, quedando finalmente 27 variables numéricas, además de las 2 variables categóricas, las cuales serán analizadas por los modelos de algoritmos de aprendizaje.

Los resultados de cada uno de los modelos se muestran en la siguiente sección.

5.2. Resultados Modelo Random Forest

Dado el entrenamiento de los datos con el modelo de Random Forest, se obtuvieron los datos presentados en la Tabla 5.2.

Accuracy:	0.9984			
Reporte de Clasificación:				
	precision	recall	f1-score	support
Fallas Mecánicas	1.00	1.00	1.00	372
Operacional no Programada	1.00	0.98	0.99	176
Otros	1.00	1.00	1.00	9003
accuracy			1.00	9551
macro avg	1.00	0.99	1.00	9551
weighted avg	1.00	1.00	1.00	9551

Tabla 5.2: Resultados Modelo Random Forest

En donde se observa que el accuracy es 0.9984, siendo este muy cercano a 1, lo cual es muy bueno, ya que quiere decir que el modelo clasificó casi a la perfección los eventos. Como aclaración, se tiene que el evento 'otros' es parte de los resultados ya que el modelo igualmente se entrena con las filas de datos que no son ni fallas mecánicas ni operacionales no planificadas, puesto que, si estos datos son eliminados, se pueden generar múltiples problemas.

Algunos de estos problemas pueden ser, por ejemplo, que al eliminar una gran cantidad de datos (ya que los datos que son fallas mecánicas y detenciones operacionales no programadas acumulan sólo 1944 de los 31834 que hay en total), el conjunto de entrenamiento se vuelve más pequeño, lo cual, puede afectar en la capacidad del modelo para generalizar, ya que tiene menos ejemplos para aprender. Otro problema es que, si se eliminan clases de fallas enteras, el modelo se vuelve altamente sesgado y limita su capacidad para manejar datos nuevos que puedan incluir tipos de fallas diferentes. Además, los otros eventos también contienen información que puede ser muy útil (como la información del comportamiento de las variables numéricas).

Es por lo anterior que, agrupar el resto de los eventos en la categoría 'otros' permite conservar todos los datos sin perder información relevante.

Se puede ver además que la precisión de las clases es perfecta en todas, el recall que es quien mide la proporción de verdaderos positivos, en relación con el total de casos positivos reales, es casi perfecto en todas, al igual que F1-score.

A continuación, se presenta en la Figura 5.1 la Matriz de Confusión del modelo, en donde se puede apreciar que el modelo tuvo errores en el tipo de evento operacional no programada, clasificándolo 4 veces en el tipo de evento otros, lo cual es muy poco considerando que clasificaron 172 eventos correctamente. Por otro lado, fallas mecánicas tuvo 372 clasificados correctamente y ninguna equivocación.

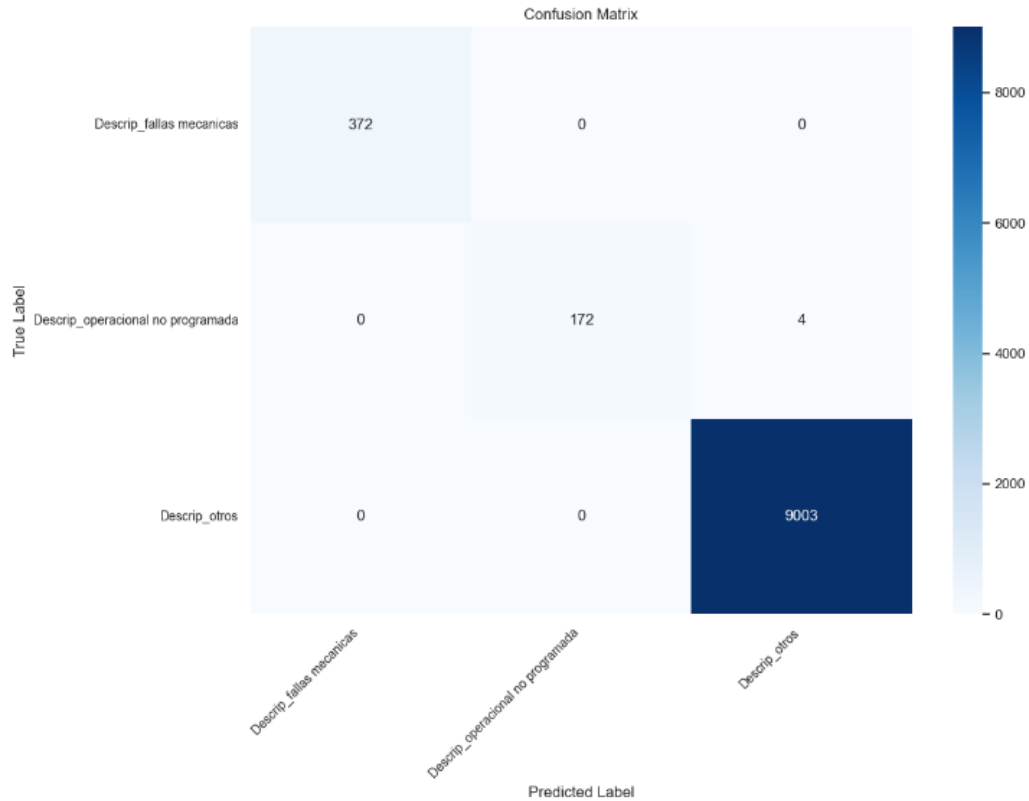


Figura 5.1: Matriz de Confusión del modelo de Random Forest

Luego, la técnica SHAP ayuda a obtener las variables numéricas más importantes para cada tipo de evento no planificado en estudio. A continuación se exponen las relaciones entre las variables en la Tabla 5.3.

Evento	Variables más Influyentes
Fallas Mecánicas	Desviación de la resistencia + Error de los amperios Motor 1 + Toneladas totales DT + Performance Toneladas por Turno + Temperature Transmite
Operacional no Programada	Desviación de la resistencia + Error de los amperios Motor 1 + Toneladas totales DT + Performance Toneladas por Turno + Temperature Transmite

Tabla 5.3: Resultados Variables Más Influyentes Para Cada Evento

Desde ya, se puede notar que las variables más influyentes para cada evento no planificado con mayor frecuencia, son las mismas para ambas, lo cual no se esperaba.

Lo anterior puede deberse a un sobreajuste del modelo (overfitting), ya que el modelo puede estar aprendiendo patrones específicos de los datos de entrenamiento, no generalizando bien. Otro punto puede ser que exista una insuficiencia de datos, puesto que si los datos de ambos eventos comparten características o variables numéricas similares, el modelo podría ver las mismas variables como importantes.

Es por esto que, para ver si existe una verdadera relación entre los eventos y las variables numéricas, primeramente, se realiza un gráfico de comportamiento temporal considerando todo el periodo de obtención de datos, el cual es desde el primero de agosto de 2020 a las 00:00:00 horas, hasta el 30 de agosto de 2021 a las 23:59:00 horas, con datos cada un minuto.

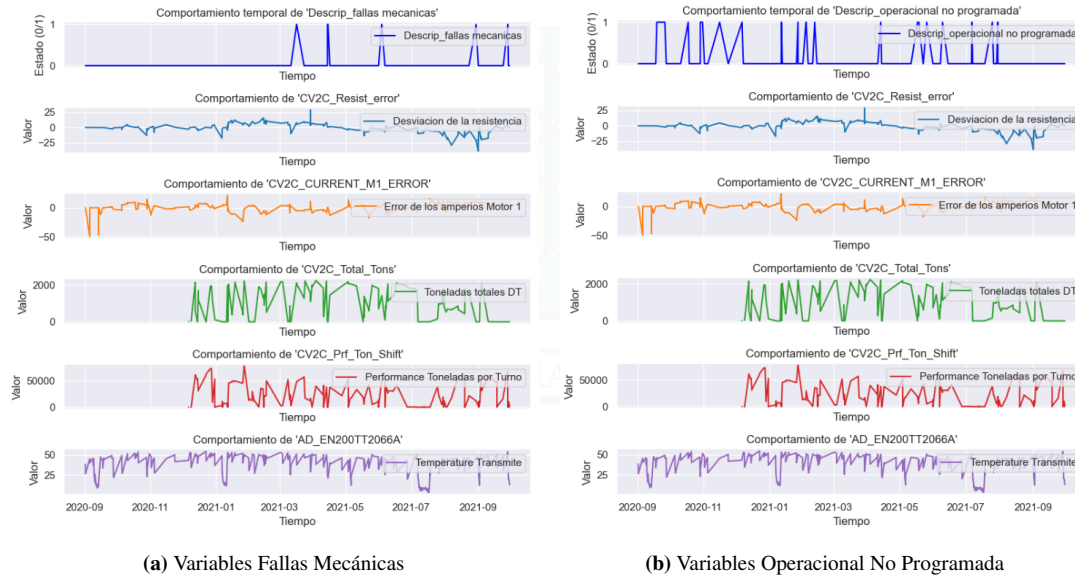


Figura 5.2: Comportamiento Temporal Variables Más Influyentes

De la Figura 5.2 se puede ver que para las fallas mecánicas en 5.2a la desviación de la resistencia de la correa transportadora CV2C tiene un comportamiento que puede asociarse a la activación de las fallas mecánicas, en relación a los picks que se pueden ver. Las demás variables presentan mucho ruido y no se ve claramente una relación en comportamiento para cuando se activan las fallas mecánicas.

Ahora, si se observa la Figura 5.2b, se puede apreciar una leve similitud en el comportamiento del error de los amperios del motor 1 de la correa transportadora, por lo que se pensaría que esta si puede tener relación con las detenciones operacionales no programadas del chancador N°5, pero las demás variables numéricas, no presentan un comportamiento que se pueda ver claramente que existe una dependencia entre estas y la falla.

Dado que no es posible ver una relación de dependencia clara con todas las variables numéricas en cada tipo de evento no planificado, se da paso a la revisión de los resultados del Modelo SEM para la hipótesis resultante del modelo de Random Forest que se muestra en la Tabla 5.3.

5.2.1. Resultados SEM para Hipótesis del Modelo de Random Forest

Los resultados obtenidos del modelo SEM para la hipótesis del modelo de Random Forest se presentan a continuación en la Tabla 5.4 y en la Tabla 5.5.

Como entrada se tiene que las variables dependientes son: fallas mecánicas y operacional no programada. Por otro lado, estas variables dependen de (o son explicadas por) cinco variables indepen-

dientes: desviación de la resistencia, error de los amperios del motor 1 de la correa transportadora, las toneladas totales, la performance de toneladas por turno y la transmisión de temperatura.

El modelo SEM se ajusta correctamente, en donde el algoritmo converge exitosamente luego de 13 iteraciones, encontrando los parámetros óptimos.

Estimator	ML
Optimization method	NLMINB
Number of model parameters	13
	Used Total
Number of observations	24292 31834
Model Test User Model:	
Test statistic	0.000
Degrees of freedom	0
Model Test Baseline Model:	
Test statistic	2.776
Degrees of freedom	11
P-value	0.000
User Model versus Baseline Model:	
Comparative Fit Index (CFI)	1.000
Tucker-Lewis Index (TLI)	1.000
Loglikelihood and Information Criteria:	
Loglikelihood user model (H0)	18.579
Loglikelihood unrestricted model (H1)	18.579
Akaike (AIC)	-37.132
Bayesian (BIC)	-37.026
Sample-size adjusted Bayesian (SABIC)	-37.068
Root Mean Square Error of Approximation:	
RMSEA	0.000
90 Percent confidence interval - lower	0.000
90 Percent confidence interval - upper	0.000
P-value H ₀ : RMSEA ≤ 0.050	NA
P-value H ₀ : RMSEA ≥ 0.080	NA
Standardized Root Mean Square Residual:	
SRMR	0.000
Parameter Estimates:	
Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Tabla 5.4: Resultados SEM del modelo Random Forest

Se analizarán paso a paso los resultados. En una primera instancia, se tiene que el método de estimación utilizado es el de Máxima Verosimilitud (ML), un estimador común en SEM cuando los datos son continuos y siguen una distribución normal. Se observa que el método de optimización que ocupa es el NLMINB, un algoritmo utilizado para encontrar los mejores parámetros. A su vez, el número estimado de parámetros es 13 y que, de un total de 31.834 datos, se utilizaron 24.292.

Por otro lado, se ve que el estadístico de prueba del modelo de usuario es 0.000, lo que indica que el modelo ajusta perfectamente los datos, pero dado que los grados de libertad son 0.000 también, no hay suficiente información para hacer una evaluación estadística de ajuste.

Que los grados de libertad sean 0.000, significa que el modelo está completamente saturado, en otras palabras, que se ajusta a todos los datos posibles y no queda ningún grado de libertad para evaluar el ajuste.

Luego se tienen los resultados del modelo base. Aquí el estadístico de prueba es 2775 con 11 grados de libertad y un valor p de 0.000. Este es un modelo nulo que asume que las variables no están correlacionadas entre sí. Comparar este valor con el modelo de usuario ayuda a evaluar si este es mejor que un modelo donde no hay relaciones entre las variables.

El CFI, que muestra la comparación entre el modelo propuesto por el usuario o investigador y el modelo base, indica un ajuste perfecto (1.000), lo cual es poco común en modelos reales y sugiere que el modelo puede estar sobre ajustado. Lo mismo ocurre con TLI.

A continuación, se presentan los criterios de información. Loglikelihood muestra el valor de verosimilitud del modelo. Un valor mayor indica un mejor ajuste, pero como los valores entre H0 y H1 son idénticos (18.578), esto indica que el modelo no tiene errores residuales. A su vez, Akaike Information Criterion (AIC) presenta un valor de -37.131. Un valor menor es mejor, lo que indica que el modelo es bueno para ajustarse a los datos sin sobre ajustar. El Bayesian Information Criterion (BIC) es similar, pero este penaliza más a modelos más complejos. Por último, el Sample-size adjusted BIC (SABIC), el cual ajusta el BIC teniendo en cuenta el tamaño de la muestra.

Luego se obtiene que el valor de RMSEA es 0.000, lo que indica un ajuste perfecto. Cabe recordar que RMSEA mide qué tan bien el modelo se ajusta a la población. Valores cercanos a 0 son deseables, pero igualmente llama la atención que este sea exactamente 0.000. Lo mismo ocurre con el SRMR, lo que sugiere que no hay diferencias entre los valores observados y los predichos, indicando un ajuste perfecto.

Ahora, se muestra en la Tabla 5.5 la continuación de los resultados del modelo SEM para la hipótesis de las relaciones entre variables, entregadas por el modelo de Random Forest.

Regressions:						
	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
fallas mecanicas ~						
Cvor	0.027	0.002	18.033	0.000	0.027	0.133
CVOR3	-0.010	0.002	-5.838	0.000	-0.010	-0.045
CVns	-0.014	0.002	-8.141	0.000	-0.014	-0.061
CVft	-0.013	0.002	-6.424	0.000	-0.013	-0.056
AD6A	0.030	0.002	14.598	0.000	0.030	0.134
operacional no programada ~						
Cvor	-0.023	0.001	-27.299	0.000	-0.023	-0.198
CVOR3	-0.018	0.001	-20.074	0.000	-0.018	-0.151
CVns	0.013	0.001	13.564	0.000	0.013	0.101
CVft	-0.018	0.001	-16.533	0.000	-0.018	-0.141
AD6A	0.024	0.001	21.889	0.000	0.024	0.198
Covariances:						
	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
fallas mecanicas ~~						
operacional no programada	-0.001	0.000	-5.794	0.000	-0.001	-0.037
Variances:						
	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
fallas mecanicas	0.050	0.000	110.209	0.000	0.050	0.960
operacional no programada	0.015	0.000	110.209	0.000	0.015	0.930
R-Square:						
	Estimate					
fallas mecanicas	0.040					
operacional no programada	0.070					

Tabla 5.5: Resultados SEM del modelo Random Forest

Aquí se muestra la segunda parte de los resultados. En la matriz de regresiones se presentan las relaciones entre las variables predictoras, que en este caso son las numéricas, y las variables dependientes, que en el caso de estudio son los dos tipos de falla.

Los resultados que se encuentran en la columna denominada Estimate, representan el coeficiente estimado de la relación entre las variables dependientes e independientes, en otras palabras, cuánto cambia la variable dependiente por cada unidad de cambio en la variable independiente, manteniendo todo lo demás constante.

Lo que resalta de lo anterior es que, fallas mecánicas en conjunto con la desviación de la resistencia (Cvor) tienen un coeficiente estimado de 0.027, lo que significa que, por cada unidad adicional de desviación de la resistencia, se espera que las fallas mecánicas aumenten en 0.027 unidades. Por otro lado, un valor negativo, como lo es el error de los amperios del motor 1 (CVOR3), indica una relación inversa, por ende, por cada unidad que aumenta el error de los amperios del motor 1, se espera que fallas mecánicas disminuya en 0.010 unidades.

También se tiene que, el error estándar (Std.Err) mide la precisión de la estimación del coeficiente, en otras palabras, cuanto menor sea el error estándar, más confiable será la estimación del coeficiente. En este caso, se aprecia que todas las variables presentan un error estándar bajo, lo que sugiere que la estimación de los coeficientes es precisa.

En la siguiente columna se muestra el valor z (z-value), el cual se calcula dividiendo el Estimate por el error estándar. Es una medida que indica cuántos errores estándar se aleja el coeficiente de 0. Sirve para probar si el coeficiente es significativamente diferente de 0. Como ejemplo, se observa que la desviación de la resistencia (Cvor) y la transmisión de temperatura (AD6A) tienen un valor z muy alto, lo que indica una relación significativa entre cada una de estas variables por separado, con las fallas mecánicas.

Luego, se tiene el valor p, el cual indica la probabilidad de obtener un coeficiente como este, o más extremo, si la relación verdadera entre las variables es 0. Un valor p menor a 0.05 generalmente se considera estadísticamente significativo. En este modelo, todos los $P(>|z|)$ son 0.000, lo que indica que todas las relaciones son altamente significativas.

El coeficiente estandarizado en la escala latente (Std.lv) es la versión estandarizada del coeficiente original, pero en la escala latente, que está relacionada con la estimación de la variable dependiente en su forma original. Permite comparar variables con diferentes escalas. En este caso, todos los valores son iguales a los estimados originalmente, ya que, la escala de las variables no ha sido ajustada.

Finalmente, se presenta el coeficiente estandarizado completamente (Std.all), el cual ajusta tanto la variable independiente como la dependiente a la misma escala (generalmente la desviación estándar). Se utiliza para comparar directamente el impacto de diferentes variables, ya que elimina las diferencias de escala. Para la desviación de la resistencia (Cvor), el valor estandarizado completamente es 0.133, lo que indica que Cvor tiene un impacto más significativo en fallas mecánicas que otras variables como el error de los amperios en motor 1 (CVOR3), cuyo coeficiente estandarizado es -0.045.

Los coeficientes estandarizados permiten interpretar la magnitud del efecto en términos de desviaciones estándar. Por ejemplo, un coeficiente de 0.133 indica que un aumento de 1 desviación estándar en Cvor incrementa fallas mecánicas en 0.133 desviaciones estándar.

Si se observa el impacto de las variables numéricas en operacional no programada, se ve que los valores estandarizados completamente son elevados, tanto positivos como negativos. Además de que tienen alta significancia, puesto que todos presentan un valor p extremadamente bajo (0.000).

Por otra parte, si se sigue con el análisis de las tablas, se presenta la de covarianza entre las fallas mecánicas y la operacional no programada.

Primeramente, el valor de Estimate de -0.001 significa que hay una ligera covarianza negativa entre fallas mecánicas y operacional no programada, lo que implica que, cuando una aumenta, la otra tiende a disminuir ligeramente. El error estándar de la covarianza mide la precisión de la estimación de la esta. Aquí, es 0.000, lo que sugiere que la estimación es muy precisa.

El valor z de -5.794 indica cuántos errores estándar se aleja la covarianza estimada de 0. Un valor tan alto en magnitud indica que la covarianza es estadísticamente significativa. Además, El valor p de 0.000 muestra que la covarianza es estadísticamente significativa.

El valor de la covarianza estandarizada en la escala latente es -0.001, lo que sugiere que, en la escala original, la covarianza negativa sigue siendo pequeña. También, el valor de -0.037 es la covarianza estandarizada completamente, lo que facilita comparar la magnitud de la covarianza en relación con otras variables. Aquí, el valor sugiere que la covarianza negativa entre fallas mecánicas y operacional no programada es pequeña.

A continuación, se encuentra la matriz de varianzas, la cual indica la cantidad de variabilidad dentro de una variable latente individual, es decir, qué tan dispersos están los valores de esa variable con respecto a su media.

Para fallas mecánicas, la varianza es 0.050, lo que indica una mayor variabilidad que en operacional no programada, cuya varianza es 0.015. Esto significa que fallas mecánicas tiene una dispersión más amplia de valores en comparación con operacional no programada.

Por otro lado, el error estándar es 0.000 para ambas varianzas, lo que indica una alta precisión en la estimación de la varianza. El valor z de 110.209 para ambas varianzas indica que la varianza es significativamente diferente de 0, lo cual es esperado para estas variables. El valor p de 0.000 muestra que las varianzas son estadísticamente significativas.

Por último, el valor de la varianza completamente estandarizada para fallas mecánicas es de 0.960, lo que significa que el 96 % de la varianza observada está explicada por el modelo. Para operacional no programada, el 0.930 indica que el 93 % de la varianza está explicada.

Finalmente, para los resultados del r-cuadrado, se tiene que el modelo explica el 4 % de la variabilidad en la ocurrencia de fallas mecánicas. Esto sugiere que el modelo ha sido ajustado de manera que capta un poco más de los factores que influyen en las fallas mecánicas, aunque aún queda una gran parte de la varianza sin explicar.

En el caso de operacional no programada, el modelo explica el 7 % de la variabilidad en las detenciones operacionales no programadas. Lo que indica que el modelo ha logrado capturar más factores que afectan a este tipo de eventos, pero aun así, faltan más variables que puedan explicar este evento en el modelo.

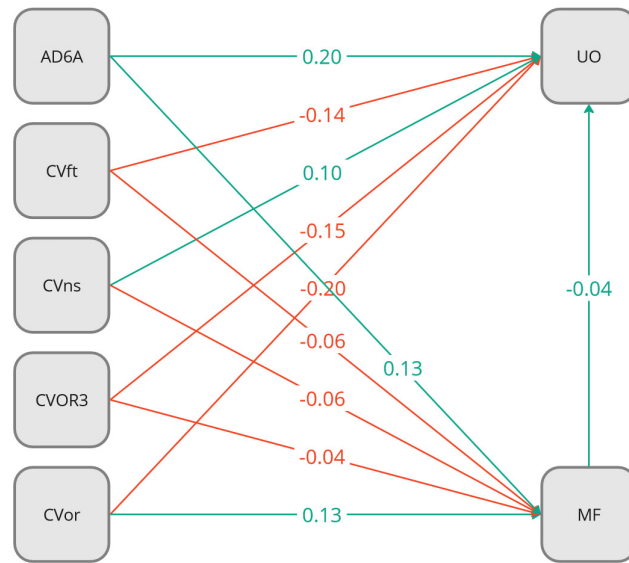


Figura 5.3: Relaciones entre Variables Dependientes e Independientes

En la Figura 5.3 se muestran las relaciones entre las variables numéricas y los dos tipos de fallas o eventos no planificados. Se logra observar que las relaciones de dependencia son significativas, tanto positiva, como negativamente.

5.3. Resultados Modelo Redes Neuronales

Dado el entrenamiento de los datos con el modelo de Redes Neuronales, se obtuvieron los datos presentados en la Tabla 5.6.

Loss Final:	0.0363			
Accuracy:	0.9926			
Precision:	0.9925			
Recall:	0.9926			
F1-score:	0.9920			
Classification Report:	precision	recall	f1-score	support
Fallas Mecánicas	1.00	0.97	0.99	372
Operacional no Programada	0.98	0.66	0.79	176
Otras	0.99	1.00	1.00	9003
accuracy			0.99	9551
macro avg	0.99	0.88	0.93	9551
weighted avg	0.99	0.99	0.99	9551

Tabla 5.6: Resultados Modelo Redes Neuronales

En donde se observa que el accuracy es 0.9926, siendo este muy cercano a 1, lo cual es muy bueno, ya que quiere decir que el modelo clasificó casi a la perfección los eventos. Como aclaración, al igual que para el modelo de Random Forest, se tiene que el evento 'otros' es parte de los resultados ya que el modelo igualmente se entrena con las filas de datos que no son ni fallas mecánicas ni operacionales no planificadas, puesto que, si estos datos son eliminados, se pueden generar múltiples problemas.

Se puede ver además que la precisión de las clases, el recall y el F1-score también son casi perfectos en general, teniendo en cuenta que la precisión para las fallas mecánicas es perfecta y el recall para las detenciones operacionales no planificadas es la más baja, presentando un 0.66, el cual mide la proporción de verdaderos positivos, en relación con el total de casos positivos reales.

A continuación, se presenta en la Figura 5.4 la Matriz de Confusión del modelo, en donde se puede apreciar que este tuvo errores en el tipo de evento operacional no programada, clasificándolo 59 veces en el tipo de evento otros y 117 veces clasificándolo correctamente. Por otro lado, fallas mecánicas tuvo 10 equivocaciones y 362 veces clasificó correctamente.

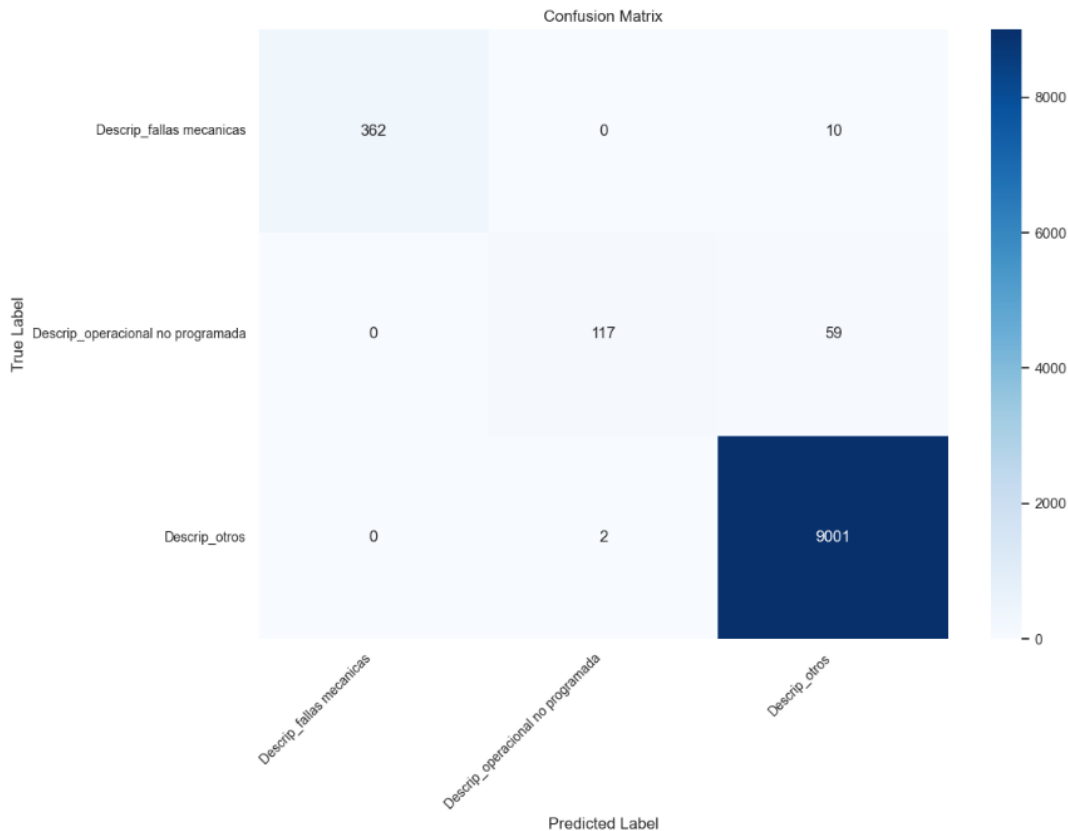


Figura 5.4: Matriz de Confusión del modelo de Redes Neuronales

Luego, la técnica SHAP ayuda a obtener las variables numéricas más importantes para cada tipo de evento no planificado de interés. A continuación se presentan las relaciones entre las variables dependientes e independientes en la tabla 5.7.

Eventos	Variables más Influyentes
Fallas Mecánicas	Error de los amperios Motor 2 + Error de los amperios Motor 1 + Temperature Transmite + Amperios Motor 1 DT + % De utilización de voltaje
Operacional no Programada	Temperature Transmite + Error de los amperios Motor 1 + Error de los amperios Motor 2 + Performance Toneladas por Turno + Temperature Transmite 2

Tabla 5.7: Resultados Variables Más Influyentes Para Cada Evento

Se observa que, a diferencia del resultado del modelo de Random Forest, las variables más influyentes para cada tipo de evento no planificado son diferentes.

Así mismo, para poder ver graficamente si existe una relación verdadera entre las variables, en la Figura 5.5 se expone el comportamiento temporal considerando todo el periodo de obtención de datos.

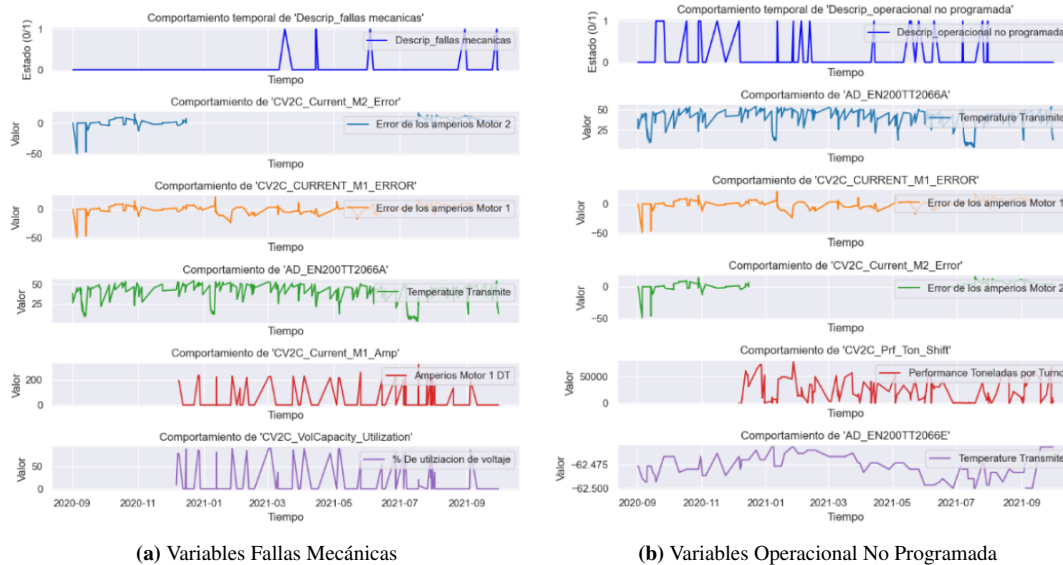


Figura 5.5: Comportamiento Temporal Variables Más Influyentes

De la Figura 5.5 se puede ver que para las fallas mecánicas en 5.5a es difícil distinguir algún comportamiento tan marcado que se asemeje, pero aún así, se puede observar que luego de que comience la actividad en los amperios del motor 1 y en el % de utilización de voltaje, comienzan a activarse las fallas mecánicas.

Por otro lado, en la Figura 5.5b se observa una alta frecuencia de activación, en donde la temperatura en transmisión presenta un comportamiento similar y a su vez, el error de los amperios Motor 1 y error de los amperios Motor 2, también pueden presentar una actividad parecida.

Dado que nuevamente no es posible ver tan claramente las relaciones de dependencia entre las variables numéricas y los tipos de eventos no planificados, se da paso a la revisión de los resultados del

Modelo SEM para la hipótesis resultante del modelo de Redes Neuronales que se muestra en la Tabla 5.7.

5.3.1. Resultados SEM Para Hipótesis del Modelo de Redes Neuronales

Los resultados obtenidos del modelo SEM para la hipótesis del modelo de Redes Neuronales se presentan en la Tabla 5.8 y la Tabla 5.9.

Las entradas en términos de variables dependientes son las fallas mecánicas y operacional no programada. Por otro lado, como variables independientes se presentan 7 distintas, las cuales son: error de los amperios del motor 2, error de los amperios del motor 1, la transmisión de temperatura, amperios motor 1 DT, % de utilización de voltaje, performance de toneladas por turno y por último, la transmisión de temperatura 2.

El modelo SEM se ajusta correctamente a la hipótesis entregada por el modelo de Redes Neuronales, en donde el algoritmo converge exitosamente luego de 44 iteraciones, encontrando los parámetros óptimos.

Estimator	ML
Optimization method	NLMINB
Number of model parameters	13
Number of observations	31834
Model Test User Model:	
Test statistic	173
Degrees of freedom	4
P-value (Chi-square)	0.000
Model Test Baseline Model:	
Test statistic	2.797
Degrees of freedom	15
P-value	0.000
User Model versus Baseline Model:	
Comparative Fit Index (CFI)	0.939
Tucker-Lewis Index (TLI)	0.772
Loglikelihood and Information Criteria:	
Loglikelihood user model (H0)	25.563
Loglikelihood unrestricted model (H1)	25.649
Akaike (AIC)	-51.099
Bayesian (BIC)	-50.991
Sample-size adjusted Bayesian (SABIC)	-51.032
Root Mean Square Error of Approximation:	
RMSEA	0.036
90 Percent confidence interval - lower	0.032
90 Percent confidence interval - upper	0.041
P-value H ₀ : RMSEA ≤ 0.050	1.000
P-value H ₀ : RMSEA ≥ 0.080	0.000
Standardized Root Mean Square Residual:	
SRMR	0.012
Parameter Estimates:	
Standard errors	Standard
Information	Expected
Information saturated (h1) model	Structured

Tabla 5.8: Resultados SEM del modelo Redes Neuronales

De la Tabla 5.8, se tiene que el método de Máxima Verosimilitud (ML) es el utilizado como estimador para los datos. El método de optimización que se ocupa es el NLMINB, algoritmo utilizado para encontrar los mejores parámetros. A su vez, el número estimado de parámetros es 13, de un total de 31.834 datos utilizados, lo que proporciona una base sólida para obtener resultados confiables.

El valor del estadístico de prueba del modelo propuesto es 173, el cual es un valor Chi-cuadrado

utilizado para evaluar la bondad de ajuste del modelo. Valores cercanos a 0 indican un buen ajuste, pero este valor elevado sugiere que el ajuste del modelo es mejorable. El modelo tiene 4 grados de libertad, y el valor p es 0.000, lo que indica que el modelo tiene un ajuste significativo, pero sugiere que la hipótesis nula (cuando el modelo es perfecto) puede ser rechazada. Aunque un valor p bajo indica significancia, no garantiza que el modelo sea perfecto.

El estadístico de prueba del modelo de referencia (el que se compara contra el modelo propuesto) es de 2.796. Un valor elevado indica que este modelo es un mal ajuste en comparación con el modelo propuesto. Tiene 15 grados de libertad, y el valor p es 0.000, lo que sugiere que el modelo de referencia es significativamente peor que el modelo propuesto.

El Comparative Fit Index (CFI) es de 0.939. Cabe recordar que el CFI compara el modelo propuesto con el modelo de referencia. Un valor mayor a 0.90 indica un buen ajuste, en este caso, 0.939 sugiere que el modelo ajusta muy bien los datos.

Por otro lado, el Tucker-Lewis Index (TLI) es de 0.772. El TLI es similar al CFI, pero penaliza modelos más complejos. Un valor por debajo de 0.90, como 0.772, indica que el modelo no ajusta perfectamente los datos, pero se esperaría que el TLI fuera menor al CFI, ya que este es un modelo más complejo.

Luego, el AIC es de -51.099, un criterio que penaliza por el número de parámetros en el modelo. Valores más bajos indican un mejor ajuste. El BIC es de -50.990, similar al AIC, pero penaliza más estrictamente los modelos con muchos parámetros. Aquí, un valor más bajo también indica un mejor ajuste. Luego, el SABIC es de -51.031, en donde nuevamente un valor más bajo sugiere un mejor ajuste.

Todo lo anterior es de utilidad para comparar resultados con otros modelos, siendo el modelo con valores más bajos, el que indique mejores ajustes.

A continuación, se tiene que el RMSEA es 0.036. Este índice mide la aproximación del modelo a los datos reales, y valores menores a 0.05 indican un buen ajuste. Un RMSEA de 0.036 indica un muy buen ajuste del modelo.

Por último, el SRMR es de 0.012, un valor menor a 0.08 indica un buen ajuste. En este caso, un SRMR de 0.012 indica que el modelo ajusta casi perfectamente a los datos.

Regressions:						
	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
fallas mecanicas ~						
CVor2	0.014	0.001	9.903	0.000	0.014	0.071
CVOR3	-0.025	0.002	-16.623	0.000	-0.025	-0.125
AD6A	0.019	0.001	16.091	0.000	0.019	0.096
CVmp	-0.009	0.006	-1.589	0.112	-0.009	-0.045
CVon	-0.012	0.006	-2.097	0.036	-0.012	-0.060
operacional no programada ~						
AD6A	0.034	0.001	34.843	0.000	0.034	0.249
CVOR3	-0.013	0.001	-12.473	0.000	-0.013	-0.095
CVor2	0.013	0.001	13.796	0.000	0.013	0.099
CVft	-0.020	0.001	-20.577	0.000	-0.020	-0.146
AD6E	-0.010	0.001	-13.216	0.000	-0.010	-0.075
Covariances:						
	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
fallas mecanicas ~~						
operacional no programada	-0.002	0.000	-10.326	0.000	-0.002	-0.058
Variances:						
	Estimate	Std.Err	z-value	P(> z)	Std.lv	Std.all
fallas mecanicas	0.039	0.000	126.163	0.000	0.039	0.971
operacional no programada	0.018	0.000	126.163	0.000	0.018	0.951
R-Square:						
	Estimate					
fallas mecanicas	0.029					
operacional no programada	0.049					

Tabla 5.9: Resultados SEM del modelo Redes Neuronales

Con relación a la Tabla 5.9, se logra observar que para la matriz de regresión, las variables independientes asociadas a las fallas mecánicas que tienen una relación positiva son Cvor2 y AD6A, mientras que para operacional no programada, las mismas variables son las que tienen una relación positiva. Las otras 3 variables que faltan para cada una tienen una relación inversa con la variable dependiente.

Para el error estándar, el cual mide la precisión de la estimación del coeficiente, se tiene que todos los valores son muy bajos, por ende, se interpreta que todas las estimaciones de los coeficientes

son precisas.

El valor z , mientras más alejado del cero esté, más significativa será la relación entre las variables dependientes e independientes. En este caso, se puede apreciar que para las fallas mecánicas, las variables CVor2, CVOR3 y AD6A tienen una relación significativa. En cambio, las variables CVmp y CVon no. Por otro lado, las variables independientes asociadas a operacional no programada todas presentan valores alejados del cero, por ende, se puede interpretar que todas estas relaciones son significativas de manera independiente con la variable categórica.

Con el valor p , se comprueba que las únicas dos variables que no tienen relaciones significativas son CVmp y CVon. Por otra parte, en la siguiente columna, se puede observar que los coeficientes estandarizados en la escala latente tienen el mismo valor que los valores originales en la columna Estimate. Esto es porque la escala de variables no ha sido ajustada, por lo que no tendrían por qué ser distintos estos valores.

Finalmente, los coeficientes estandarizados, quienes permiten interpretar la magnitud del efecto en términos de desviaciones estándar, muestran que el valor para AD6A es de 0.096, lo que indica que esta variable tiene un impacto más significativo en fallas mecánicas que las otras variables. Para operacional no programada, se tiene que la variable que tiene un impacto más significativo que las demás es CVor2, con un valor de 0.099.

A continuación, se presenta la matriz de covarianzas, en donde se observa que el valor de -0.002 del indicador Estimate, significa que hay una ligera covarianza negativa entre fallas mecánicas y operacional no programada, lo que implica que, cuando una aumenta, la otra tiende a disminuir ligeramente.

El error estándar de la covarianza mide la precisión de la estimación de la covarianza. En este caso es 0.000, lo que sugiere que la estimación es muy precisa.

El valor z de -10.326, indica cuántos errores estándar se aleja la covarianza estimada de 0. Un valor tan alto en magnitud indica que la covarianza es estadísticamente significativa. Luego, el valor p viene a reafirmar que la covarianza es estadísticamente significativa ya que este valor es menor a 0.050.

La covarianza estandarizada en la escala latente tiene un valor de -0.002, lo que sugiere que, en la escala original, la covarianza negativa sigue siendo pequeña.

Por último, el valor de -0.058 es la covarianza estandarizada completamente, lo que facilita comparar la magnitud de la covarianza en relación con otras variables. Aquí, el valor sugiere que la covarianza negativa entre fallas mecánicas y operacional no programada es pequeña.

Ahora se procede a observar los resultados de la matriz de varianzas, por un lado, para fallas mecánicas, la varianza es de 0.039, lo que indica una mayor variabilidad que en operacional no programada, cuya varianza es de 0.018. Esto significa que fallas mecánicas tiene una dispersión más amplia de valores en comparación con operacional no programada.

El error estándar es 0.000 para ambas varianzas, lo que indica una alta precisión en la estimación de la varianza.

El valor z de 126.163 para ambas varianzas indica que la varianza es significativamente diferente de 0, lo cual es esperado para estas variables. Además, el valor p de 0.000 muestra que las varianzas son estadísticamente significativas.

Por otro lado, las fallas mecánicas tienen una varianza estandarizada de 0.039, que corresponde al valor en la escala original, y operacional no programada tiene una varianza de 0.018, también sin cambios en la escala.

Por último, para fallas mecánicas, el valor de 0.971 significa que el 97.1 % de la varianza observada está explicada por el modelo. Mientras que para operacional no programada, el 0.951 indica que el 95.1 % de la varianza está explicada.

Finalmente, se observan los resultados del r-cuadrado, en donde para fallas mecánicas se obtiene un 0.029. Esto significa que el modelo explica aproximadamente el 2.9 % de la varianza en la ocurrencia de fallas mecánicas. Es un valor bajo, lo que sugiere que las variables incluidas en el modelo no están capturando la mayor parte de los factores que afectan a las fallas mecánicas.

Por otro lado, para operacional no programada se obtuvo un 0.049, en otras palabras, el modelo explica el 4.9 % de la varianza en la variable operacional no programada. Al igual que en el caso de las fallas mecánicas, este es un valor bajo, lo que indica que el modelo no está capturando muchos de los factores que influyen en las detenciones operacionales no programadas.

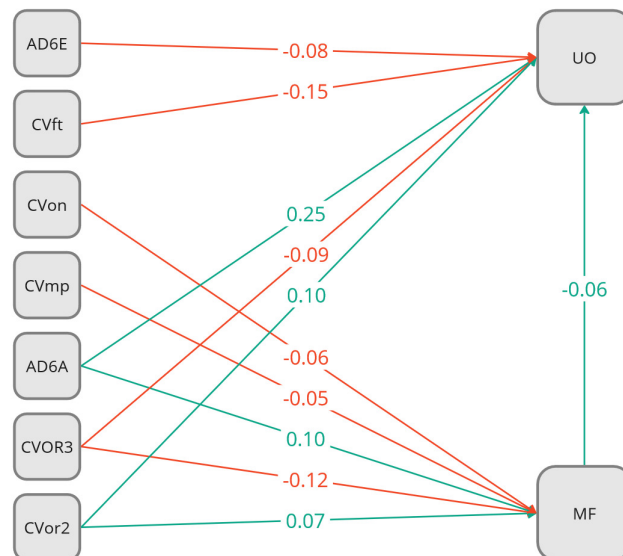


Figura 5.6: Relaciones entre Variables Dependientes e Independientes

En la Figura 5.6 se pueden ver las relaciones entre las 7 variables independientes y las 2 variables dependientes que se encuentran en el lado derecho de la imagen. Existen relaciones de dependencia que son significativas, como AD6A con operacional no programada, y otras no tan significativas como, por ejemplo, CVon con fallas mecánicas.

5.4. Resultados Modelo Predicción de Fallas

Finalmente, a partir de los resultados obtenidos, se integran las relaciones derivadas del modelo ANN-SEM como insumos para el modelo de predicción de fallas. Este modelo, también basado en una ANN, es entrenado bajo las condiciones establecidas y luego se realiza una comparación con un modelo de predicción de fallas que no incluye las relaciones de dependencia, empleando las mismas métricas de evaluación para asegurar consistencia. Los rendimientos de ambos enfoques se muestran en la Tabla 5.10, lo cual permite observar el impacto de incluir las dependencias identificadas.

V.T.	Capas	Neuronas	Función	T.A.	Umbral	Sin Dependencias		Con Dependencias	
						Accuracy	Recall	Accuracy	Recall
15	3	128	Tanh	0.0001	0.5	0.7473	0.7009	0.8622	0.4685
30	1	128	Sigmoid	0.0001	0.3	0.6523	0.5326	0.8970	0.4647
60	2	32	Tanh	0.00001	0.3	0.7873	0.3983	0.9186	0.5126

Tabla 5.10: Resultado Modelo Predicción de Fallas Con y Sin Dependencias

A partir de lo obtenido, se concluye que el modelo que implementa una ventana temporal (V.T) de 60 minutos y considera las dependencias en su fase de entrenamiento es el más sólido en términos de precisión, alcanzando un accuracy de 91.86 %. Este modelo no solo destaca en precisión, sino que además muestra un equilibrio adecuado entre accuracy y recall, con este último en un 51.26 %, lo cual refleja su capacidad para detectar fallas en una mayor proporción de los casos. Aunque los otros modelos que también consideran las dependencias ofrecen un accuracy superior frente a aquellos que no las incluyen, estos tienden a ser más conservadores, limitándose a predecir con mayor certeza solo ciertos eventos sin lograr identificar la totalidad de las fallas. Esta característica sugiere que, si bien la inclusión de dependencias refuerza el rendimiento del modelo, su capacidad para capturar un espectro más amplio de fallas mejora notablemente al optimizar el balance entre accuracy y recall en el modelo de 60 minutos. Cabe destacar que en la Tabla 5.10 T.A. es la tasa de aprendizaje.

El modelo que incorpora dependencias y opera con una ventana temporal de 60 minutos destaca como el más eficiente para predicciones a largo plazo, mostrando indicadores que avalan su rendimiento superior. Este modelo no solo logra capturar una mayor proporción de fallas reales, sino que también asegura un alto grado de certeza en sus predicciones de falla. Al mantener una precisión notable en escenarios a mayor plazo, se convierte en una herramienta confiable tanto para la identificación temprana de problemas como para la optimización de acciones preventivas.

6 | Discusión y Alcance

En esta sección, se explorará con mayor detalle los resultados obtenidos tanto del modelo de Random Forest como del modelo de Redes Neuronales, en conjunto cada uno con el desempeño obtenido del modelo SEM aplicados a los tipos de eventos no planificados dentro de la línea de chancado que tienen más frecuencia, los cuales son las fallas mecánicas y las detenciones operacionales no programadas.

Ambos modelos han permitido desentrañar relaciones importantes entre diversas variables independientes y estos dos tipos de fallas, aportando información clave para comprender mejor los factores que influyen en estos eventos. Sin embargo, los resultados también revelan limitaciones en la capacidad de los modelos para capturar completamente la complejidad inherente a los eventos no planificados, lo que abre una discusión más profunda sobre la validez, utilidad y áreas de mejora de cada enfoque.

También se analizará con mayor profundidad lo obtenido por el modelo de Predicción de Fallas, el cual integra las relaciones identificadas en el modelo ANN-SEM con el fin de mejorar la precisión en la detección temprana de fallas no planificadas. Este análisis permitirá evaluar en qué medida la incorporación de las dependencias entre variables contribuye a capturar eventos críticos de forma más precisa, especialmente en comparación con modelos que no consideran estas interrelaciones.

6.1. Evaluación del Modelo de Random Forest en conjunto de SEM

Uno de los primeros puntos a destacar es la capacidad explicativa del modelo Random Forest para ambos tipos de fallas. Los resultados muestran que el modelo fue capaz de explicar el 3 % de la variabilidad en las fallas mecánicas y el 7 % en las detenciones operacionales no programadas. Aunque estos porcentajes son relativamente bajos, es importante reconocer que las fallas mecánicas y las detenciones operacionales no planificadas pueden estar influenciadas por múltiples factores externos que no han sido incluidos en el conjunto de datos o el modelo. Esto podría sugerir la necesidad de incorporar más variables que potencialmente capturen mejor los factores clave que inciden en estos eventos.

La baja capacidad explicativa del modelo para las fallas mecánicas puede reflejar que los datos utilizados no están abarcando las complejidades asociadas con este tipo de eventos. Es probable que las fallas mecánicas estén sujetas a una mayor cantidad de factores externos o no observables en el presente análisis, tales como condiciones operativas extremas, mantenimiento inadecuado o incluso errores humanos, que no se encuentran reflejados en las variables numéricas seleccionadas.

En el caso de las detenciones operacionales no programadas, aunque el modelo explicó un porcentaje mayor de la variabilidad, es evidente que aún queda una gran parte de la variabilidad sin capturar. Esto podría indicar la necesidad de profundizar en las causas subyacentes de estos eventos. Factores como cambios repentinos en las condiciones de operación, eventos climáticos adversos o fallas en los sistemas de monitoreo podrían estar contribuyendo a la variabilidad observada y no han sido considerados adecuadamente en el modelo actual.

Por otro lado, la Figura 5.3 resalta algunas relaciones significativas entre las variables numéricas y los dos tipos de fallas o eventos no planificados más frecuentes. Las relaciones identificadas pueden interpretarse como indicativas de cómo ciertas variables operativas influyen directamente en la ocurrencia de fallas mecánicas y detenciones operacionales no planificadas. Es importante destacar que la naturaleza de las relaciones es tanto positiva como negativa, lo que subraya la complejidad del sistema y la necesidad de un análisis más detallado para desentrañar estas interacciones.

Un aspecto interesante de estas relaciones es que algunas variables pueden tener un impacto opuesto dependiendo del tipo de falla. Esto podría sugerir que ciertos factores operativos pueden ayudar a mitigar un tipo de evento, mientras que exacerban otro, lo que representa un desafío importante para los sistemas de gestión y predicción de fallas. Se recomienda realizar una evaluación más profunda para comprender cómo estas variables pueden ser gestionadas de manera efectiva para reducir la ocurrencia de fallas en ambos contextos.

A pesar de todo, se observó que el modelo de Random Forest entregó las mismas variables más influyentes para ambos tipos de fallas, lo que es bastante sospechoso. Algunas posibles causas de esto pueden ser que los tipos de eventos no estén bien balanceados, como lo fue en este caso, ya que, existen 1345 instancias para las fallas mecánicas, y 599 para las detenciones operacionales no programadas, por ende, más del doble de registros, lo cual puede tender a realizar un sobre ajuste a las características de la clase mayoritaria, reflejando así una falta de sensibilidad a las diferencias entre los dos tipos de fallas.

Otra posible causa es que las variables predictoras estén altamente correlacionadas entre sí, por lo que el modelo podría estar seleccionando las mismas variables para ambos eventos debido a esa colinealidad, en lugar de reflejar diferencias genuinas en los factores que influyen en las fallas mecánicas y las detenciones no programadas.

Además de lo anterior, se tiene que en el modelo SEM se muestran varios indicadores “perfectos” como, por ejemplo, el CFI o TLI de 1.000. Esto también podría ser sospechoso y sugiere que el modelo podría estar sobre ajustado. En teoría, un CFI o TLI de 1 indica un ajuste perfecto entre el modelo y los datos, lo cual es extremadamente raro en la práctica. Algunas de las posibles causas pueden ser que el modelo esté demasiado ajustado a los datos específicos que tienes, lo que significa que está capturando tanto el patrón real como el ruido o las particularidades de ese conjunto de datos. Esto puede resultar en un ajuste perfecto, pero no generalizable a nuevos datos.

Otro caso es si el modelo SEM ha sido mal especificado, es decir, las relaciones entre las variables no están adecuadamente representadas, obteniendo indicadores perfectos debido a una mala estructuración. Esto podría significar que se está forzando un ajuste en lugar de obtener un modelo que

verdaderamente refleje las relaciones en los datos.

Ambos resultados podrían apuntar a que los modelos, aunque ofrecen resultados llamativos, no están proporcionando una visión completa y precisa de las dinámicas reales que afectan a los eventos de fallas no planificadas. En estos casos, los indicadores pueden parecer perfectos, pero el modelo no estaría capturando la complejidad real del fenómeno.

6.2. Evaluación del modelo de Redes Neuronales en conjunto de SEM

Para el modelo de Redes Neuronales, se obtuvieron variables independientes diferentes para ambos tipos de eventos no planificados, lo cual difiere del modelo de Random Forest y es bueno para una primera instancia.

La discusión ahora se centra en los resultados obtenidos a partir del modelo de Ecuaciones Estructurales (SEM), el cual es diseñado para corroborar la hipótesis del modelo de Redes Neuronales. Los resultados sugieren que el modelo SEM tuvo un ajuste razonable, aunque no perfecto, con base en los indicadores clave de ajuste, tales como el Comparative Fit Index (CFI) y el Tucker-Lewis Index (TLI).

El valor del CFI de 0.939 sugiere un muy buen ajuste general del modelo, lo cual indica que el modelo propuesto logra capturar una parte importante de las relaciones entre las variables observadas. Sin embargo, el TLI de 0.772 muestra que el modelo SEM aún puede ser optimizado. Este índice penaliza la complejidad del modelo y, al ser inferior a 0.90, sugiere que el modelo actual, aunque adecuado, podría ser simplificado o ajustado para mejorar su capacidad explicativa sin perder precisión en la estimación de las relaciones.

Un elemento importante para destacar en el ajuste del modelo SEM es el RMSEA, cuyo valor de 0.036 indica un muy buen ajuste de los datos reales al modelo. Este resultado es significativo, ya que sugiere que, a pesar de las posibles limitaciones observadas en los otros indicadores, el modelo es capaz de aproximarse de manera efectiva a las relaciones subyacentes entre las variables independientes y dependientes. La combinación de un RMSEA bajo y un SRMR de 0.012 refuerza la idea de que el modelo ajusta adecuadamente los datos, lo que otorga confianza en la validez de los resultados.

Los resultados de la matriz de regresión en la Tabla 5.9 muestran que las variables de error de los amperios Motor 2 (CVor2) y la temperatura de transmisión (AD6A) tienen una relación positiva significativa con las fallas mecánicas y las detenciones operacionales no programadas. Esto sugiere que estas dos variables, que representan factores operativos clave, juegan un papel importante en la ocurrencia de ambos tipos de fallas. La consistencia en la dirección de estas relaciones refuerza la idea de que estas variables deberían ser monitoreadas de cerca para mejorar la capacidad de predicción y control de los eventos no planificados.

En contraste, otras variables como amperios Motor 1 DT (CVmp) y % de utilización de voltaje (CVon) no mostraron una relación significativa con las fallas mecánicas, lo que podría indicar que

estas variables no son factores determinantes para este tipo de eventos o que su impacto es moderado por otros factores no incluidos en el modelo. La falta de significancia estadística para estas variables sugiere que su inclusión en futuros modelos podría ser reevaluada o, alternativamente, que se exploren interacciones o transformaciones de estas variables para obtener una mejor comprensión de su rol en la predicción de fallas.

Por otro lado, las relaciones significativas observadas entre las variables independientes y las detenciones operacionales no programadas son coherentes con las expectativas de que variables como AD6A y CVor2 desempeñan un papel importante en estos eventos. Dado que estas variables tienen una relación positiva con los eventos operacionales, su monitoreo constante podría proporcionar señales tempranas de problemas potenciales, permitiendo a los operadores tomar medidas preventivas antes de que se produzcan fallas.

Al analizar los coeficientes estandarizados, se observa que la variable AD6A tiene el mayor impacto en las fallas mecánicas, mientras que CVor2 tiene un impacto similar en las detenciones operacionales no programadas. Estos hallazgos son consistentes con las relaciones previamente observadas y sugieren que estas variables pueden ser consideradas como indicadores clave de desempeño en la prevención y gestión de fallas.

El análisis de los coeficientes estandarizados también muestra que las otras variables incluidas en el modelo tienen un impacto menor, lo que sugiere que estas variables podrían tener un efecto más moderado o indirecto en la ocurrencia de los eventos no planificados. Sin embargo, no es correcto descartar su relevancia, ya que podrían actuar en combinación con otras variables o tener un papel en la aparición de fallas en contextos específicos.

Los resultados de la matriz de covarianza mostraron una ligera covarianza negativa entre fallas mecánicas y detenciones operacionales no programadas. Esto es un hallazgo interesante que sugiere que, en algunos casos, la ocurrencia de un tipo de falla puede estar asociada con una reducción en la probabilidad de que ocurra el otro tipo de evento. Aunque esta covarianza es pequeña, su significancia estadística indica que puede haber un cierto grado de competencia entre estos dos tipos de fallas en términos de recursos o condiciones operativas que influyen en su aparición.

Es importante tener en cuenta que esta covarianza negativa no significa necesariamente que los dos tipos de eventos estén completamente desvinculados, sino que podría haber factores externos que afectan a ambos eventos de manera diferente. Por ejemplo, una mejora en las condiciones mecánicas de los equipos podría reducir la probabilidad de fallas mecánicas, mientras que podría aumentar la presión sobre los sistemas operacionales, lo que resultaría en un incremento en las detenciones operacionales no programadas.

6.3. Evaluación del modelo de Predicción de Fallas

Al incorporar las dependencias generadas por el modelo ANN-SEM en un segundo modelo de predicción basado también en ANN, se observó una mejora significativa en la robustez de las predicciones de fallas, lo que resalta el valor de considerar las interrelaciones entre variables. Este

modelo de ANN, configurado con una ventana de tiempo de 60 minutos antes de una posible falla, 2 capas ocultas y 32 neuronas en cada capa, alcanzó un notable accuracy de 91.86 % y un recall de 51.26 %. Estos resultados son superiores en comparación con las simulaciones en las que no se incluyeron las dependencias, lo que subraya la importancia de modelar relaciones estructurales en el proceso predictivo.

El intervalo de 60 minutos antes de un evento de falla, además, ofrece una ventaja práctica considerable, ya que proporciona un margen de tiempo razonable para implementar acciones preventivas o correctivas. Esto permite que los equipos de mantenimiento y operación actúen de forma proactiva, minimizando el impacto en la línea de producción y potenciando la capacidad de respuesta operativa. La configuración del modelo no solo optimiza la precisión de las predicciones, sino que también equilibra la sensibilidad hacia las fallas, manteniendo un balance entre alta precisión y capacidad de detección efectiva.

Este análisis pone en evidencia que la combinación de tener en cuenta las relaciones de dependencia y la capacidad de aprendizaje de los modelos de ML, puede resultar en un sistema de predicción más adaptado a las dinámicas complejas y no lineales de los eventos no planificados. Se considera, por tanto, que la incorporación de estas dependencias permite no solo una mejora en las métricas predictivas, sino también una capacidad de aprendizaje que captura mejor la naturaleza de los eventos críticos recurrentes, haciendo del modelo una herramienta más confiable para la gestión y planificación del mantenimiento..

7 | Conclusiones

Las conclusiones clave de este estudio están vinculadas tanto a las ventajas evidenciadas a partir de los resultados obtenidos y el análisis realizado, como a los conocimientos adquiridos y las limitaciones inherentes del modelo utilizado y sus aplicaciones. Estos hallazgos sientan las bases para investigaciones futuras, especialmente dirigidas a incorporar el modelo propuesto en la simulación de eventos de fallas no planificadas, lo que podría mejorar aún más su capacidad predictiva en contextos industriales.

7.1. Observaciones emanadas de los resultados obtenidos

En esta investigación, los resultados obtenidos resaltan el papel de las ANN para la predicción de eventos no planificados, particularmente en fallas mecánicas y detenciones operacionales no programadas dentro del sistema de chancado minero. A diferencia del modelo de RF, que mostró tendencias de sobreajuste debido a la naturaleza desequilibrada de los datos, las ANN demostraron una mayor capacidad para identificar las relaciones complejas entre variables y capturar de manera más precisa los patrones asociados a las fallas. Esto permitió no solo una predicción más precisa, sino también una comprensión más detallada de los factores críticos que preceden a estos eventos.

La integración de dependencias identificadas mediante el modelo SEM en el modelo de Predicción de Fallas fortaleció la estructura predictiva, logrando mejorar la precisión y el recall de las fallas detectadas. Con una ventana temporal de 60 minutos antes del evento de falla y una configuración de 2 capas y 32 neuronas, este modelo específico de ANN alcanzó un accuracy de 91.86 % y un recall de 51.26 %, cifras superiores a las de los modelos que no incluían estas relaciones de dependencia. El análisis mediante SEM también fue crucial para confirmar la generalización de las dependencias y mitigar posibles sesgos en los datos de entrenamiento, especialmente dado el desequilibrio de clases en el conjunto de datos, donde los eventos de falla representaban solo una pequeña proporción frente a la totalidad de los registros.

7.2. Limitaciones del estudio

Entre las limitaciones destacadas, el desequilibrio en la base de datos resultó ser un factor que afectó significativamente la generalización del modelo RF y, en menor medida, de las ANN. Este desequilibrio puede llevar al modelo a centrarse en la clase mayoritaria, reduciendo así su eficacia en la detección de eventos menos frecuentes.

De un total de 31.834 observaciones, solo 1.944 representan eventos de fallas mecánicas o detenciones operacionales no planificadas. Este marcado desequilibrio en el conjunto de datos puede

tener un impacto significativo en la capacidad de los modelos para generalizar los resultados.

Cuando la mayoría de los datos corresponden a eventos que no son de interés, los modelos tienden a priorizar la clase mayoritaria, lo que afecta su desempeño en la identificación y predicción de los eventos minoritarios, en este caso, los tipos de eventos no planificados con más frecuencia. Este tipo de sesgo puede llevar a un modelo que sea preciso en términos generales pero ineficaz para predecir los eventos más importantes en este estudio, como en el caso del Random Forest. Además, la escasez relativa de datos sobre fallas mecánicas y detenciones operacionales no planificadas podría reducir la capacidad de los modelos para captar la complejidad y diversidad de estos eventos.

Es por lo anterior que se recalca que para mejorar el rendimiento en futuras investigaciones, es esencial contar con una recolección de datos más equilibrada, que permita al modelo aprender de manera más representativa sobre eventos de falla.

7.3. Alcances y Aplicaciones

Los resultados obtenidos a partir del modelo de ANN ofrecen una base sólida para profundizar en la comprensión de las causas más probables detrás de los tipos de eventos no planificados en el ámbito industrial. La precisión de este modelo en la identificación de patrones ocultos permite desentrañar qué factores son los más críticos a la hora de prever estos eventos disruptivos. Esta capacidad predictiva es particularmente valiosa para la toma de decisiones en operaciones industriales, donde las fallas no planificadas suelen tener un impacto significativo tanto en los costos como en la eficiencia operativa.

Al conocer con mayor exactitud qué variables o indicadores deben ser monitoreados de forma más rigurosa, las empresas pueden optimizar sus recursos de manera más efectiva y enfocarse en estrategias preventivas más focalizadas. Esto no solo contribuye a reducir la frecuencia de estas fallas, sino también a minimizar sus efectos adversos, mejorando la continuidad y confiabilidad de los procesos operativos. Además, la aplicabilidad de las Redes Neuronales en este contexto no se limita únicamente a la predicción de fallas, sino que también facilita la detección de patrones que, de otra manera, pasarían desapercibidos con técnicas tradicionales. Su capacidad para trabajar con datos no lineales y su flexibilidad para adaptarse a distintos tipos de variables las convierte en una herramienta muy poderosa en el análisis de sistemas complejos, donde la relación entre las variables no siempre es evidente o lineal.

Por otro lado, la incorporación de técnicas interpretativas como SHAP (SHapley Additive exPlanations) complementa el análisis de las Redes Neuronales y otros modelos de aprendizaje automático, ya que, estas técnicas permiten descomponer y visualizar la contribución específica de cada variable dentro del modelo, proporcionando una forma de entender cómo y por qué el modelo llega a ciertas predicciones, lo que es especialmente relevante cuando se trabaja con algoritmos que son percibidos como "cajas negras". Esto aporta una capa adicional de transparencia, permitiendo no solo interpretar mejor los resultados, sino también validar si las relaciones identificadas por el modelo tienen un fundamento lógico y práctico.

En estudios industriales, donde las fallas pueden ser influenciadas por múltiples factores, esta capacidad de visualizar la interacción entre variables es invaluable para los equipos de mantenimiento y gestión de activos, ya que permite mejorar las decisiones estratégicas sobre cómo abordar el mantenimiento preventivo y predictivo de manera más eficiente.

Finalmente, al complementar estos análisis con modelos SEM, se verifica la significancia estadística y relevancia teórica de las relaciones entre variables dependientes e independientes. Esto proporciona un marco más robusto para la toma de decisiones, ya que permite verificar si las correlaciones y dependencias identificadas no solo son estadísticamente válidas, sino que también tienen sentido dentro del contexto operativo de la industria. Por ejemplo, en el ámbito del mantenimiento preventivo, estos modelos pueden ayudar a priorizar las intervenciones en las áreas más críticas, basándose en un análisis riguroso de los datos y en una comprensión profunda de las relaciones causales que impulsan las fallas.

7.4. Recomendaciones para Trabajos Futuros

Uno de los aspectos clave para mejorar los resultados en investigaciones futuras es la ampliación y optimización de la recolección de datos, específicamente en lo que respecta a las fallas mecánicas y detenciones operacionales no programadas. El desequilibrio observado en el conjunto de datos utilizado en este estudio, donde solo una pequeña fracción de los datos representa eventos de interés, puede influir negativamente en la capacidad de los modelos de aprendizaje automático para generalizar correctamente. Los algoritmos, al estar expuestos a un número significativamente mayor de casos sin fallas que de casos con fallas, pueden inclinarse hacia la predicción de no fallas, lo que reduce su efectividad en identificar y predecir estos eventos críticos.

Por lo tanto, se recomienda encarecidamente que futuros estudios busquen obtener un mayor volumen de datos relacionado con estos eventos no planificados. Una estrategia efectiva sería implementar un sistema de recolección de datos más riguroso en campo, que capture una mayor variedad de situaciones en las que ocurren fallas mecánicas y detenciones operacionales no programadas. Esto no solo ayudaría a balancear el conjunto de datos, sino que también permitiría a los modelos de aprendizaje aprender patrones más detallados y específicos asociados a las fallas. Además, contar con una base de datos más grande y equilibrada proporcionaría un marco más sólido para realizar análisis estadísticos y aplicar técnicas de validación más robustas, lo que, a su vez, mejoraría la capacidad de los modelos para generalizar a nuevos conjuntos de datos.

Por otro lado, otra observación importante radica en la estabilidad del intervalo de 60 minutos antes de una falla obtenido en el modelo de Predicción de Fallas, ya que proporciona un margen práctico para implementar acciones preventivas y mitigar impactos operacionales. Este intervalo, combinado con la precisión alcanzada por el modelo, convierte a la ANN en una herramienta eficaz para la detección temprana, destacando su aplicabilidad en escenarios industriales.

Un enfoque prometedor que se desprende de esta investigación es la posible implementación de un sistema de asesoramiento en tiempo real, que funcione como un recomendador dinámico para

los operadores. Este sistema podría anticipar situaciones de riesgo, ofreciendo alertas y recomendaciones prácticas cuando se detecte una alta probabilidad de falla, basadas en los modelos predictivos desarrollados. Dado el alto nivel de precisión demostrado por las ANN en la identificación de fallas, un recomendador de esta naturaleza optimizaría significativamente la gestión operativa en distintas industrias, permitiendo intervenciones preventivas más efectivas y minimizando tiempos de inactividad no planificados.

Este estudio también pone de relieve el valor de transformar extensas bases de datos industriales en información procesable para la toma de decisiones en tiempo real, proporcionando una base sólida para continuar el perfeccionamiento de estos modelos en futuras investigaciones. El análisis profundo de las relaciones entre variables permitió estructurar un modelo predictivo sólido y confiable para anticipar fallas. La precisión lograda por las ANN refuerza su utilidad como una herramienta esencial para comprender y prever eventos de falla en entornos industriales complejos, permitiendo a las organizaciones gestionar de manera más eficaz sus activos críticos.

8 | Anexos

Variable	Descripción	Abreviación
DateTime	Fecha	DT
AD_EN200WIT2073	Carga de la correa en tph del DCS	AD73
EN200SI2065	% de Velocidad de Correa DCS	EN65
HS_EN200CV2C_RS	Estado de funcionamiento de correa CV2C	HSRS
HS_EN200CV2CM3_RS	Información Motor3	HSRS
HS_EN200CV2CM4_RS	Información Motor4	HSRS
SS001_EN200IICV2CM1	Amperios Motor 1 DCS	SSM1
SS001_EN200IICV2CM2	Amperios Motor 2 DCS	SSM2
AD_EN200TT2066A	Temperature Transmite	AD6A
AD_EN200TT2066B	Temperature Transmite	AD6B
AD_EN200TT2066C	Temperature Transmite	AD6C
AD_EN200TT2066D	Temperature Transmite	AD6D
AD_EN200TT2066E	Temperature Transmite 2	AD6E
AD_EN200TT2066F	Temperature Transmite	AD6F
AD_EN200TT2066G	Temperature Transmite	AD6G
AD_EN200TT2066H	Temperature Transmite	AD6H
MTLCV2CM1_FACTPOT	Factor de Potencia Motor 1	MTOT
MTLCV2CM1_POTREALH	Potencia del Motor 1	MTLH
MTLCV2CM2_FACTPOT	Factor de Potencia Motor 2	MTOT
MTLCV2CM2_POTREALH	Potencia del Motor 2	MTLH
CV2C_Load_tph	Carga de la correa en tph del DT	CVph
CV2C_Load_Cal_Error_tph	Error de la carga de la correa en tph DT	CVph
CV2C_Power	Potencia DT (Dos motores)	CVVer
CV2C_Current_M1_Amp	Amperios Motor 1 DT	CVmp
CV2C_Current_M2_Amp	Amperios Motor 2 DT	CVmp
CV2C_Design_Resist	Resistencia Diseñada de la Correa	CVst
CV2C_Actual_Resist	Resistencia Actual de la Correa	CVst
CV2C_Resist_error	Desviacion de la resistencia	CVor
CV2C_Speed	% de Velocidad de Correa DT	CVed
CV2C_Current_M2_Error	Error de los amperios Motor 2	CVor
CV2C_CURRENT_M1_ERROR	Error de los amperios Motor 1	CVOR
CV2C_Torque	Torque DT	CVue
CV2C_Total_Tons	Toneladas totales DT	CVns
CV2C_VolCapacity_Utilization	% De utilizacion de voltaje	CVon
CV2C_PowerCapacity_Utilization	% De utilizacion de Potencia	CVon
CV2C_Prft_Ton_Hr	Performance Toneladas por Hora	CVHr
CV2C_Prft_Ton_Shift	Performance Toneladas por Turno	CVft

Tabla 8.1: Variables Numéricas, Significado y Abreviatura

Bibliografía

- [1] W. Sun, P. Bocchini, and B. D. Davison. Overview of interdependency models of critical infrastructure for resilience assessment. *Natural Hazards Review.*, (2022). 3.1.1, 3.1.4
- [2] C. Huang, C. Sun, and X. Liu. Cascading failures in interdependent infrastructure systems: A review. *Safety Science.*, (2013). 3.1.1
- [3] R. L. Moglen, J. Barth, S. Gupta, E. Kawai, K. Klise, and B. D. Leibowicz. A nexus approach to infrastructure resilience planning under uncertainty. *Reliability Engineering & System Safety*, 2023. 3.1.1, 3.1.3
- [4] E. Zio and G. Sansavini. Modeling interdependent network systems for identifying cascade-safe operating margins. *IEEE Transactions on Reliability*, (2011). 3.1.1
- [5] H. Huang, Y. K. Tang, X. Shi, and X. L. Mao. Dependency-aware neural topic model. *Information Processing & Management*, (2024). 3.1.1
- [6] Wróbel R. Dependencies of elements recognized as critical infrastructure of the state. *Transportation Research Procedia*, (2019). 3.1.3
- [7] M. Haenlein and A. M. Kaplan. A beginner's guide to partial least squares analysis. *Understanding Statistics*, (2004). 3.1.5
- [8] D. A. Cole and K. J. Preacher. Manifest variable path analysis: Potentially serious and misleading consequences due to uncorrected measurement error. *Psychological Methods*, (2014). 3.1.5
- [9] A. S. Albahri, O. S. Albahri, K. I. Mohammed, A. A. Zaidan, B. B. Zaidan, A. H. Alamoodi, and M. Alaa. Systematic review of artificial intelligence techniques in the prediction and classification of respiratory diseases. *PeerJ*, (2021). 3.1.5
- [10] T. Hastie, R. Tibshirani, and J. Friedman. The elements of statistical learning: Data mining, inference, and prediction. *Springer Series in Statistics.*, (2009). 3.2.1, 3.2.3, 3.2.4, 4.6.1
- [11] C. M. Bishop. Pattern recognition and machine learning. *Springer.*, (2006). 3.2.1, 3.2.2, 3.2.3, 3.2.4
- [12] I. Goodfellow, Y. Bengio, and A. Courville. Deep learning. *MIT Press.*, (2016). 3.2.1, 3.2.2, 3.2.3, 3.2.4
- [13] R. S. Sutton and A. G. Barto. Reinforcement learning: An introduction. *MIT Press.*, (2018). 3.2.2
- [14] A. B. Siddique, N. Ali, M. Hamraz, S. U. Khan, and S. A. Khattak. Prediction of magnetic nature of oxide compositions by using machine learning models. *Computational Condensed Matter*, (2024). 3.2.4, 4.6
- [15] E. Zio and L. Miqueles. Digital twins as a security risk? *Risk Analysis.*, (2024). 3.3.1
- [16] M. Grieves. *Digital twin: Manufacturing excellence through virtual factory replication*. University of Michigan., (2004). 3.3.1
- [17] E. J. Tuegel, A. R. Ingraffea, T. G. Eason, and S. M. Spottswood. Reengineering aircraft structural life prediction using a digital twin. *International Journal of Aerospace Engineering*, (2011). 3.3.1
- [18] Y. Pan and L. Zhang. Roles of artificial intelligence in construction engineering and management: A critical review and future trends. *Automation in Construction*, (2021). 3.3.1

- [19] C. Boje, A. Guerriero, S. Kubicki, and Y. Rezgui. Towards a semantic construction digital twin: Directions for future research. *Automation in Construction*, (2020). 3.3.1
- [20] B. Schleich, Anwer N., L. Mathieu, and S. Wartzack. Shaping the digital twin for design and production engineering. *CIRP Annals*, (2017). 3.3.1
- [21] L. Hou, X. Wang, and J. Wang. Digital twin-oriented service supply chain architecture: An initial exploration. *Journal of Manufacturing Systems*, (2020). 3.3.1
- [22] A. Akanmu, C. J. Anumba, and J. I. Messner. Scenarios for cyber-physical systems integration in construction. *Journal of Information Technology in Construction*, (2014). 3.3.1
- [23] F. Tao, H. Zhang, A. Liu, and A. Y. C. Nee. Digital twin in industry: State of the art. *IEEE Transactions on Industrial Informatics*, (2019). 3.3.1
- [24] Z. Ye, Y. Ye, C. Zhang, Z. Zhang, Li W., X. Wang, and L. Wang. A digital twin approach for tunnel construction safety early warning and management. *Computers in Industry*, (2023). 3.3.1
- [25] H. P. Bloch. *Practical Machinery Management for Process Plants: Volume 3, Third Edition*. Elsevier., (2006). 3.4.1
- [26] R. K. Mobley. *An Introduction to Predictive Maintenance*. Butterworth-Heinemann., (2002). 3.4.1, 3.4.3, 3.4.5
- [27] M. Mirzehi Kalateh Kazemi, Z. Nabavi, M. RezaKhah, and A Masoudi. Application of xgb-based metaheuristic techniques for prediction time-to-failure of mining machinery. *Systems and Soft Computing*, (2023). 3.5
- [28] A. Szmigiel, D. B. Apel, J. Wang, Y. Pu, L. Wojtecki, and Y. Pourrahimian. Enhancing open stope stability prediction in mining engineering: Optimal configuration of an artificial neural network model. *Journal of Industrial Safety*, (2024). 3.5
- [29] H. Rafezi and F. Hassani. Drill bit wear monitoring and failure prediction for mining automation. *International Journal of Mining Science and Technology*, (2023). 3.5
- [30] A. Liaw and M. Wiener. Classification and regression by random forest. *R News*, (2002). 4.6.1
- [31] S. M. Lundberg and S. I. Lee. A unified approach to interpreting model predictions. *In Proceedings of the 31st International Conference on Neural Information Processing Systems*, (2017). 4.6.1
- [32] D. M. Powers. Evaluation: From precision, recall and f-measure to roc, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, (2011). 1
- [33] C. A. Zonta, T. and da Costa, R. da Rosa Righi, M. J. de Lima, E. S. da Trindade, and G. P. Li. Predictive maintenance in the industry 4.0: A systematic literature review. *Computers & Industrial Engineering*, (2020). 4.8