ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

Identificación del origen de fallas en el sistema de distribución eléctrica del Ecuador mediante la aplicación de técnicas de machine learning

Proyecto Integrador

Previo la obtención del Título de:

Ingeniero en Electricidad

Presentado por:
Walter Josue Burgos Díaz

Guayaquil - Ecuador
Año: 2023

Dedicatoria

Este trabajo está dedicado a mis abuelos, a mis padres y a mis hermanos, quienes siempre me han apoyado en cada uno de los retos que he enfrentado. También a todos los profesores que con sabiduría supieron aconsejarme para seguir creciendo personal y profesionalmente.

Walter Josue Burgos Díaz

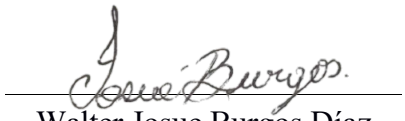
Agradecimientos

Quiero agradecer en primer lugar a mis abuelos Marcia Posso y Walter Díaz por ser los pilares en mi vida personal, académica y profesional. A mis padres Katty Díaz y Julio Burgos por siempre brindarme su apoyo y compañía. A mis amigos Dayana Camacho, Jhon Mosquera, Hernán Ullon, Tamara Saavedra, Darwin Murillo, Karla Tovar y María Leonor quienes supieron brindarme su apoyo en cada momento, además de abrirme las puerta de su hogar acogiéndome como un miembro mas de su familia. A los Ingenieros Efrén Herrera y Holger Cevallos por las enseñanzas y consejos brindados tanto académica y personalmente. A Karen Velásquez por todo el apoyo brindado para que pueda culminar mi trabajo. Finalmente a mi tutor el Ingeniero Luis Ugarte por la confianza depositada en mí y mi trabajo.

Walter Josue Burgos Díaz

Declaración Expresa

"Los derechos de titularidad y explotación, me corresponde conforme al reglamento de propiedad intelectual de la institución; Walter Josue Burgos Díaz doy mi consentimiento para que la ESPOL realice la comunicación pública de la obra por cualquier medio con el fin de promover la consulta, difusión y uso público de la producción intelectual"



Walter Josue Burgos Díaz

Evaluadores

Firmado electrónicamente por:
ANGEL ANDRES
RECALDE LINO

Ph.D. Ángel Recalde

PROFESOR DE LA MATERIA

Firmado electrónicamente por:
LUIS FERNANDO
UGARTE VEGA

Ph.D. Luis Ugarte

PROFESOR TUTOR

Resumen

El presente trabajo tiene como finalidad identificar el origen de fallas en el sistema de distribución eléctrico del Ecuador mediante la aplicación de técnicas de machine learning, con el objetivo clasificar las fallas cuyas causas son desconocidas, registradas en la base de datos de la empresa distribuidora CNEL EP. Las técnicas de machine learning permiten mejorar de manera significativa la precisión de identificación del origen de las fallas, para de esta manera mejorar la eficiencia y calidad del suministro eléctrico. En el desarrollo de este trabajo se aplicó la técnica KDD (Knowledge Discovery in Databases), los algoritmos de clasificación que se aplicaron fueron Support vector machine (SVM), Random Forest y K-nearest neighbors (KNN). El algoritmo de Random Forest demostró una precisión del 98.96% luego de la validación de los datos entrenados, mientras que el algoritmo de K-nearest neighbors demostró una precisión del 67.92% y el algoritmo de Support vector machine demostró una precisión de 59.25%. El algoritmo cuyas métricas de medición identifican y manejan de precisa las clases es Random Forest, mientras que el algoritmo de Support vector machine no clasifica de manera correcta las clases, siendo el algoritmo con menor efectividad al momento de clasificar una falla.

Palabras Clave: Distribución Eléctrica, Machine Learning, Algoritmos, Clasificación.

Abstract

The purpose of this study is to identify the origin of faults in Ecuador's electric distribution system through the application of machine learning techniques, with the goal of classifying faults whose causes are unknown, recorded in the database of the distribution company CNEL EP. Machine learning techniques significantly improve the precision in identifying the origin of faults, thereby enhancing the efficiency and quality of the electrical supply. In the development of this work, data provided by the company were processed, which were filtered according to the relevant information that could be obtained from them. For data extraction, the KDD (Knowledge Discovery in Databases) technique was applied, and the classification algorithms used were Support Vector Machine (SVM), Random Forest, and K-Nearest Neighbors (kNN). The Random Forest algorithm demonstrated a precision of 98.96% after the validation of the trained data, while the K-Nearest Neighbors algorithm showed a precision of 67.92%, and the Support Vector Machine algorithm had a precision of 59.25%. The algorithm whose measurement metrics accurately identify and handle classes is Random Forest, whereas the Support Vector Machine algorithm does not correctly classify the classes, being the least effective algorithm when classifying a fault.

Keywords: *Electrical Distribution, Machine Learning, Algorithms, Classification.*

Índice General

Abreviaturas

ESPOL	Escuela Superior Politécnica del Litoral
SIN	Sistema Nacional Interconectado
CNEL EP	Corporación Nacional de Electricidad del Ecuador
SVM	Support vector machine
KNN	K-Nearest Neighbors
KDD	Knowledge Discovery in Databases
ENSI	Energía No Suministrada en el Intervalo
PCA	Principal Component Analysis

Simbología

min	minuto
h	hora
kWh	kilovatio-hora

Índice de figuras

Índice de tablas

Capítulo 1

1.1 Introducción

La calidad de la energía eléctrica abarca varios aspectos fundamentales, como la forma de onda, la continuidad del servicio y la satisfacción del cliente. La detección de fallas eléctricas está directamente vinculada a garantizar la continuidad en el suministro eléctrico [1]. Algunas fallas pueden detectarse y solucionarse fácilmente, mientras que otras, como las fallas en las líneas de distribución, pueden originarse de objetos externos o mal funcionamiento del equipo. Estas fallas son inevitables y no pueden prevenirse de manera sencilla [2]. Adicionalmente, la identificación errónea de un punto de fallo podría retardar la recuperación de la red, resultando en mayores pérdidas económicas y la insatisfacción de los clientes [3] .

Hasta la actualidad se han utilizado distintos métodos para la detección fallas eléctricas en el sistema de distribución, como los mencionados [4] [5] [6] en que se encuentran basados en el cálculo de impedancias, el estudio de las ondas viajeras y métodos inteligentes respectivamente, siendo este último método el enfoque principal de este trabajo.

Las técnicas de Machine Learning avanzan de manera prometedoras, incluyendo su enfoque para la identificación de fallas en las redes de distribución eléctrica, gracias a su flexibilidad y eficacia [7]. En [8], se enfocan en incorporar la extracción de características al proceso de aprendizaje ya que les brinda una ventaja clave la cual sería la ausencia de la necesidad de parámetros de línea. En el primer paso del procedimiento, se determina el tipo de falla para luego establecer su ubicación. Las principales limitaciones de este método incluyen su complejidad y su dependencia de la señal de corriente para su funcionamiento. Los medidores inteligentes de alimentadores pueden utilizarse en las redes de distribución, jugando un papel útil en la identificación de fallas.

Los algoritmos de machine learning incorporan un enfoque innovador al entrenamiento de máquinas. Ya que no dependen de una programación explicita, sino que permiten a la computadora adquirir inteligencia por sí misma. Esta capacidad de aprendizaje autónomo capacita a la computadora para tomar decisiones y realizar predicciones sin intervención humana directa. La eficiencia y rendimiento de estos algoritmos dependen netamente de la proporción de la muestra del conjunto de datos, ya que esto permite un mejor aprendizaje y entrenamiento de este. Este proceso dinámico de mejora continua hace que los algoritmos de machine learning sean herramientas eficaces en diversos campos, desde la medicina hasta las finanzas, entre otros campos [9].

Estas técnicas se han extendido a diversas áreas de investigación en ingeniería, incluyendo química, eléctrica, civil, industrial, mecánica automotriz y procesamiento de imágenes. Un claro ejemplo de aplicación de estos algoritmos es, la detección de fallas en el sistema de distribución eléctrica [10]

1.2 Descripción del problema

Anualmente en el Ecuador se registra un aproximado de 15500 MWh de energía no suministrada de los cuales el 58% corresponden a fallas eléctricas y de estas fallas el 66% se producen en el sistema de distribución [11].

Las fallas eléctricas que ocurren en los sistemas de distribución se dividen en fallas técnicas y no técnicas. Las fallas técnicas son causadas por los efectos físicos de la electricidad sobre los componentes y equipos del sistema (subestaciones, redes de media tensión, transformadores, redes secundarias, luminarias, conectores y medidores) y dependen de las características y topología de la red de distribución. La fallas no técnicas por motivos administrativos y comerciales, tales como: errores y mala gestión de facturación; equipos de medición en mal estado o dañados; y en caso de fraude por conexiones ilegales de usuarios [12].

Estas fallas tienen impacto directo en la confiabilidad del suministro eléctrico. Un gran porcentaje de estas fallas técnicas no logran ser identificadas representando un obstáculo frecuente debido a la insuficiencia de datos. Esta limitación impide realizar diagnósticos efectivos, afectando así la eficiencia operativa y la estabilidad del sistema.

1.3 Justificación del problema

En el sistema de distribución eléctrica se identifican importantes riesgos ambientales, donde los aspectos que producen dichos riesgos son la afectación a la fauna por posible colisiones, electrocuciones de aves con los cables de distribución, generación de radiación no ionizante que proviene de los flujos eléctricos y magnéticos. Esto desemboca en problemas en la comunidad como por ejemplo el impacto del ruido, incluso impactos en los animales, plantas y paisajes del área [13].

Entre las medidas propuestas para abordar los riesgos más importantes en este sector se encuentra estudiar la viabilidad de automatizar el mantenimiento de la red de distribución, mediante el uso de tecnologías avanzadas como machine learning, esto puede mejorar las actividades de mantenimiento, reducir la necesidad de intervención humana en áreas de difícil acceso y reducir los riesgos para los trabajadores [13].

Implementar estas alternativas contribuirán de manera significativa al sistema de distribución mediante un análisis profundo que facilite la prevención de interrupciones futuras, mejorando así la gestión, la confiabilidad y la eficiencia del suministro eléctrico en el país.

1.4 Objetivos

1.4.1 Objetivo General

Aplicar técnicas de Machine Learning para la identificación precisa de las causas de fallas en la red de distribución eléctrica del Ecuador, con la finalidad de mejorar la confiabilidad y eficiencia del sistema de distribución de energía eléctrica.

1.4.2 Objetivos Específicos

- Implementar algoritmos de clasificación para evaluar su capacidad de manejo de las distintas clases de una base de datos.
- Comparar las métricas de medición estándar de los algoritmos implementados con la finalidad de seleccionar el de mayor precisión.
- Demostrar la importancia que tiene la aplicación de técnicas de machine learning en el sector eléctrico para mejorar los aspectos importantes del sistema de distribución eléctrica.

1.5 Marco teórico

1.5.1 Variables multiclase

Las variables multiclase hacen alusión a la clasificación de datos cuando una variable puede pertenecer a más de dos categorías distintas, en donde elegir el enfoque adecuado para resolver la problemática es algo que estará ligado a varios factores, como la naturaleza de los datos o las características específicas del problema abordado [14].

El concepto de variable multiclase es de gran relevancia en el campo del aprendizaje automático, esto permite que abordar situaciones complejas, adaptarse a contextos cambiantes y brindar una mayor información y contexto sobre los elementos de la base de datos [15].

Algunas de las técnicas que emplean las variables multiclase son [15]:

- Label encoding
- One-Hot Encoding
- SoftMax Activation

- Regresión Logística Multiclase
- Máquinas De Soporte Vectorial Multiclase (SVM)
- Redes Neurales Multicapa

1.5.2 Principal Component Analysis

El análisis de componentes principales se emplea en estadística para disminuir el tamaño de un conjunto de datos, pero conservando toda la información relevante posible.

Ayuda a convertir un conjunto de variables que se relacionan entre sí, a un conjunto de variables las cuales no están relacionadas, a las cuales conocemos como componentes principales [16].

Desde 1901 hasta la actualidad, el PCA ha estado en constante cambio, llegando a ser una técnica que se torna crucial en varias disciplinas, como la reducción de la dimensionalidad de los datos, la visualización o el preprocesamiento. [17]

El corazón del PCA consiste en la simplificación del grado de dificultad relacionado a los datos, al convertir las variables iniciales del banco de datos, a un grupo de variables no relacionadas, la cual adquiere el nombre de componentes principales. [18] Los componentes del grupo de variables no relacionadas se ordenan de tal manera que el primero de ellos capta la máxima varianza posible los datos, y así sucesivamente con los demás componentes, dándonos así la cantidad máxima de información sobre el banco de datos, pero disminuyendo su dimensión, lo cual facilidad en gran manera la interpretación de los datos para su análisis. [19]

En cuanto a las aplicaciones del método, tenemos las siguientes [20]:

- Reducción de la dimensionalidad: mejora la eficiencia computacional al comprimir bancos de datos reteniendo únicamente la información principal.
- Eliminación de multicolinealidad: si las variables originales están muy relacionadas entre sí, se puede reducir la multicolinealidad mediante la conversión de los datos iniciales en un grupo de dados no relacionados.
- Visualización de datos: acelera el proceso de visualización de patrones o tendencias.

1.5.3 One Hot Encoding

Es una técnica empleada para el análisis de datos y automatización de aprendizaje, cuyo concepto se basa en la representación de categorías o el etiquetado de categorías de forma numérica. Se asigna un vector de carácter binario y único, a cada categoría que se tenga en los datos a analizar, este vector tiene la particularidad que todos sus elementos son ceros, a excepción de uno, el cual estará en la posición que corresponde a la categoría que se está representando con dicho vector [21].

El funcionamiento de One Hot Encoding es implementar una columna para cada una de las categorías que exista en la muestra de datos en la que se aplica, y les da valor de 1 o 0 si la categoría se encuentra presente o no, dándole un valor de 0 cuando no está presente. De esta manera, no es necesario que la columna contenga los datos en forma matricial [22].

1.5.4 Varianza explicada

Es la proporción de variabilidad total que tenemos en una nube de datos, la cual se representa mediante cierto conjunto de componentes. En PCA, la varianza explicada es un indicativo de cuanta información se mantuvo luego de retener un determinado número de componentes principales [23].

1.5.5 Support Vector Machine

Es un algoritmo de aprendizaje, el cual puede ser empleado en la clasificación de nubes de datos, donde busca un hiperplano capaz de dividir los elementos de la base de datos en diversas clases o realizar una predicción de valores continuos [24].

1.5.6 Knowledge Discovery in Databases

Es un algoritmo de aprendizaje, empleado en tareas enfocadas a la clasificación o regresión e datos, este algoritmo predice un nuevo punto de datos empleando los valores de las instancias de su alrededor, analizando las vecindades en el espacio de características [25].

Una vez que se tiene el elemento que se quiere clasificar, se analizan las etiquetas que tiene en sus vecindades en el espacio de características, pudiendo de esta manera evaluar a donde pertenece dicho elemento simplemente revisando sus etiquetas más cercanas, tomando como base la suposición de que los elementos en sus cercanías tienen a contar con similares etiquetas. [26]

1.5.7 Sistemas desbalanceados

Un sistema se considera desbalanceado cuando en una base de datos se tienen clases que no están representadas de una manera equivalente, esto puede traer complicaciones en el aprendizaje automático [27].

Los puntos clave en estos sistemas están dados por [28]:

- Clasificación desbalanceada
- Sesgo en el modelo
- Precisión engañosa
- Remuestreo.

Capítulo 2

2. Metodología

En este proyecto se adoptó la metodología de KDD para el desarrollo del algoritmo de Machine Learning enfocado en la clasificación precisa de fallas no clasificadas en sistemas de distribución eléctrica. Esta metodología consta de varios pasos, diseñados para encontrar información relevante en grandes conjuntos de datos. Su proceso es netamente iterativo y requiere la participación en la toma de decisiones, adaptándose de manera dinámica a medida que este avanza, cada paso contribuye al objetivo final de mejorar la clasificación y el entendimiento de las fallas del sistema [29].

A continuación se detallan los pasos de la metodología KDD que se siguieron en este trabajo:

2.1 Selección de Datos

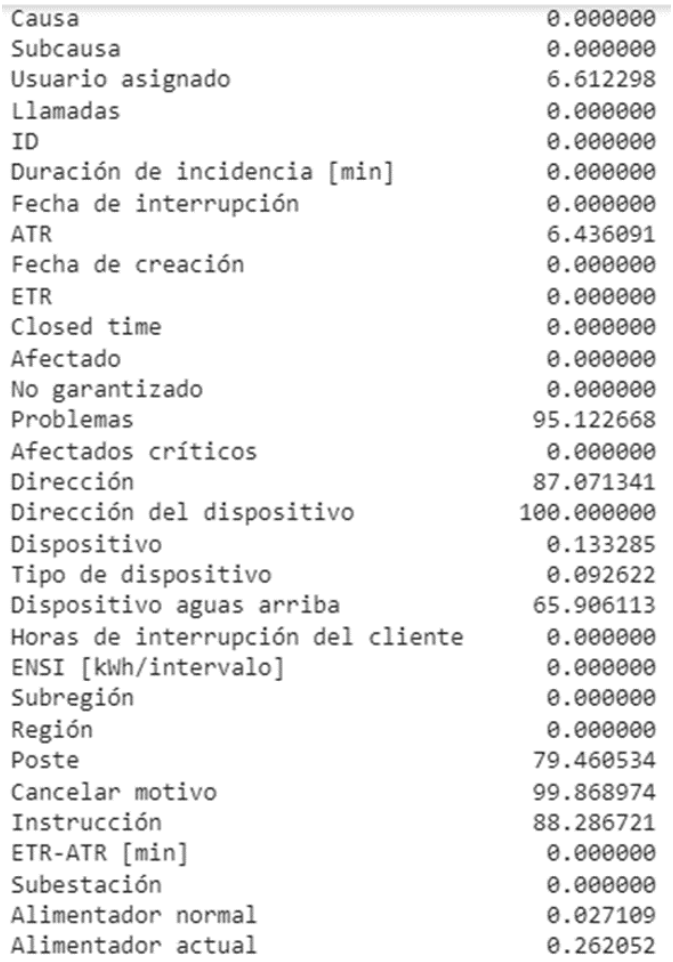
Se recopiló un conjunto de datos de registros históricos de fallas del sistema de distribución eléctrica entre los años 2020 y 2022 de la empresa CNEL EP, en donde se registra cada incidente detallando 36 parámetros en total. Cada registro contiene parámetros con información de cada falla entre los principales se encuentran los siguientes:

- Prioridad: indica cuál es el nivel de importancia asignado a cada falla reportada de 1 al 5.
- Tipo: describe el tipo de falla producida, de las que hay 4 tipos registradas.
- Causa: detalla la causa principal de la falla reportada, dentro de este parámetro se encuentran causas no clasificadas, las cuales se busca clasificar mediante el algoritmo de Machine Learning.
- Subcausa: detalla de información relevante del motivo principal de la falla registrada.

En esta fase de la metodología se realizaron tareas de limpieza y preparación de datos, lo cual permitió una selección adecuada de los parámetros que se usaron para el desarrollo del algoritmo.

Primero se realizó una revisión de la información de cada parámetro registrado en la base de datos, así se descartaron los parámetros cuyos valores no proporcionaron información relevante sobre la falla ocurrida, también la mayoría cuyos datos se encontraban vacíos.

Figura 1
Parámetros de la base de datos



Causa	0.000000
Subcausa	0.000000
Usuario asignado	6.612298
Llamadas	0.000000
ID	0.000000
Duración de incidencia [min]	0.000000
Fecha de interrupción	0.000000
ATR	6.436091
Fecha de creación	0.000000
ETR	0.000000
Closed time	0.000000
Afectado	0.000000
No garantizado	0.000000
Problemas	95.122668
Afectados críticos	0.000000
Dirección	87.071341
Dirección del dispositivo	100.000000
Dispositivo	0.133285
Tipo de dispositivo	0.092622
Dispositivo aguas arriba	65.906113
Horas de interrupción del cliente	0.000000
ENSI [kWh/intervalo]	0.000000
Subregión	0.000000
Región	0.000000
Poste	79.460534
Cancelar motivo	99.868974
Instrucción	88.286721
ETR-ATR [min]	0.000000
Subestación	0.000000
Alimentador normal	0.027109
Alimentador actual	0.262052

Nota. La figura muestra el porcentaje de valores vacíos de cada parámetro de la base de datos.

Se procedió a descartar los parámetros cuya información no es relevante para la identificación de fallas debido a la elevada cantidad de datos únicos que poseen en sus columnas (datos no numéricos), estos parámetros fueron:

- Confirmado
- Estado
- Cuadrillas
- Usuario asignado
- Llamadas
- ID
- Fecha de interrupción
- Fecha de creación
- Closed time

- ATR
- ETR (Debido a que existe un parámetro con la diferencia de tiempo con ATR)
- Tipo de dispositivo
- No garantizado
- Subestación
- Alimentador normal
- Alimentador actual

Una vez realizada una selección de preliminar de los parámetros para el estudio, los resultados fueron los siguientes:

Figura 2
Parámetros luego de la depuración

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 44266 entries, 0 to 44265
Data columns (total 13 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   Prioridad                       44266 non-null  int64
 1   Tipo                            44266 non-null  object
 2   Causa                           44266 non-null  object
 3   Subcausa                        44266 non-null  object
 4   Duración de incidencia [min]    44266 non-null  int64
 5   Afectado                        44266 non-null  int64
 6   Problemas                       2159 non-null   object
 7   Afectados críticos              44266 non-null  int64
 8   Horas de interrupción del cliente  44266 non-null  float64
 9   ENSI [kWh/intervalo]            44266 non-null  float64
 10  Subregión                       44266 non-null  object
 11  Región                          44266 non-null  object
 12  ETR-ATR [min]                   44266 non-null  int64
dtypes: float64(2), int64(5), object(6)
memory usage: 4.4+ MB
```

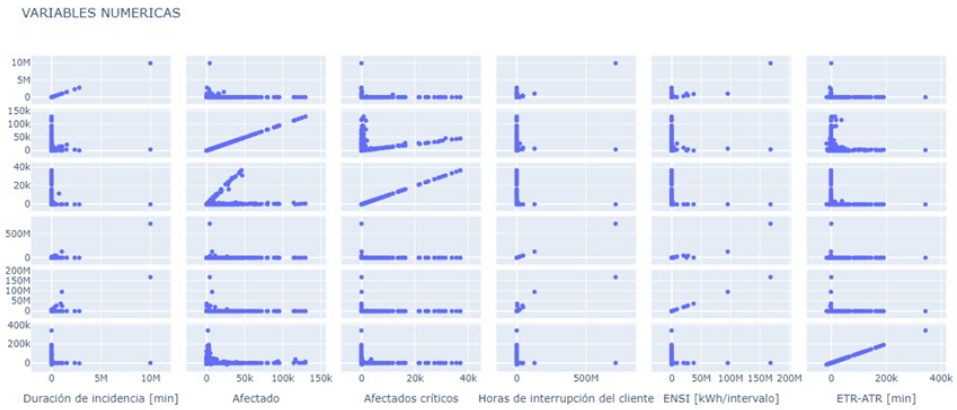
Nota. La figura muestra los parámetros cuyos datos se encuentran completos en la base de datos.

2.2 Preprocesamiento de Datos

Se realizó una revisión detallada para corregir valores faltantes y resolver inconsistencias en la categorización.

Los datos luego de este tratamiento se observaban de esta manera:

Figura 3
Dispersión de las variables numéricas



Nota. La figura muestra como se encuentran dispersos los valores de las variables numéricas.

Figura 4
Cantidad de datos totales

	Duración de incidencia [min]	Afectado	Afectados críticos	Horas de interrupción del cliente	ENSI [kWh/intervalo]	ETR-ATR [min]
0	1	4231	78	70.52	29.626140	89
1	2	2897	6	96.57	47.402070	89
2	26	4667	123	155.57	39.798980	64
3	131	33	0	72.05	40.736150	241
4	161	19	0	50.98	11.792090	205
...	...	...	...	...	...	...
44261	1	1582	13	26.37	31.440000	89
44262	19	1	0	0.05	0.004057	0
44263	0	4789	0	0.00	0.000000	90
44264	0	7360	61	0.00	0.000000	90
44265	1	1	0	0.02	0.005192	150

44266 rows × 6 columns

Nota. La figura muestra cuantos datos se encuentran registrados en la base de datos.

La cantidad de datos obtenidos son 44266, es decir la cantidad de fallas registradas, pero como se puede observar la distribución de los datos tuvo una buena dispersión para continuar con el proceso. Debido a esto se estandarizaron formatos y se normalizaron las variables numéricas para evitar sesgos en el análisis. Para esto se aplicaron métodos para poder eliminar los valores atípicos o outliers, lo cual permitió que se tenga una mejor distribución de los datos como se puede observar a continuación:

Figura 5
Dispersión de las variables numéricas elimando outliers

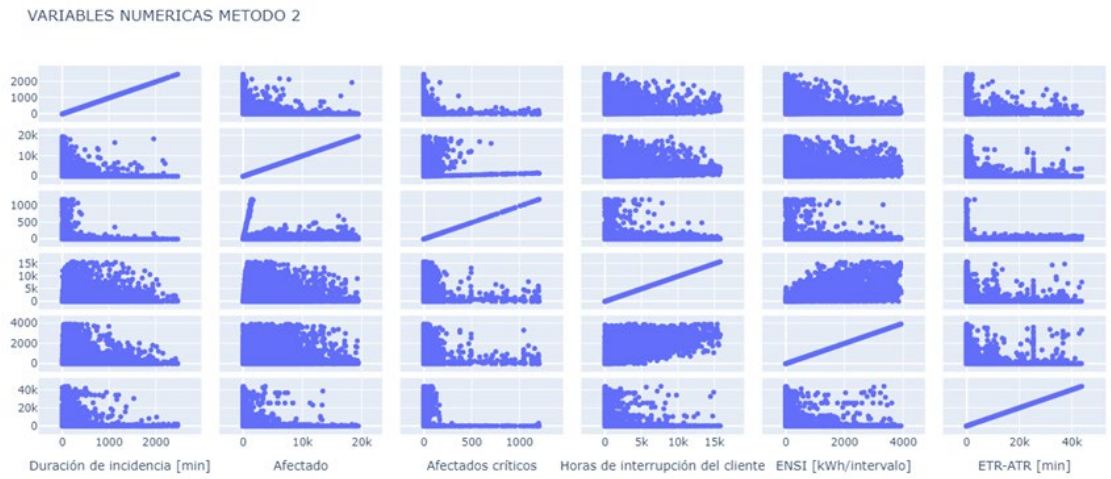


Figura 6
Cantidad de datos totales eliminando outliers

	Duración de incidencia [min]	Afectado	Afectados críticos	Horas de interrupción del cliente	ENSI [kWh/intervalo]	ETR-ATR [min]
0	1	4231	78	70.52	29.626140	89
1	2	2897	6	96.57	47.402070	89
2	26	4667	123	155.57	39.798980	64
3	131	33	0	72.05	40.736150	241
4	161	19	0	50.98	11.792090	205
...	...	...	...	...	...	...
44261	1	1582	13	26.37	31.440000	89
44262	19	1	0	0.05	0.004057	0
44263	0	4789	0	0.00	0.000000	90
44264	0	7360	61	0.00	0.000000	90
44265	1	1	0	0.02	0.005192	150

41491 rows × 6 columns

2.3 Transformación de Datos

En esta etapa de la metodología se realizó la normalización de datos debido a los diferentes tipos de datos que se tenía en los variables escogidas para entrenamiento del algoritmo. Se dividieron los datos en dos partes, la primera parte contenía la información de todas las fallas cuyas causas si estaban identificadas, mientras que la segunda parte se apartó la información de las fallas cuyas causas no estaban identificadas. En la normalización de los datos se aplicó la técnica de One hot encoding para transformar la variable de estudio causa en tipo numérica y de esta manera adaptarla al tipo de datos que requieren los parámetros ajustables de los algoritmos de machine learning.

Figura 7
Resultado de la normalización de los datos mediante la técnica One hot encoding

	Causa_1	Causa_2	Causa_3	Causa_4	Causa_5	Causa_6	Causa_7	Causa_8	\
0	1	0	0	0	0	0	0	0	
3	0	1	0	0	0	0	0	0	
6	1	0	0	0	0	0	0	0	
12	0	0	1	0	0	0	0	0	
14	1	0	0	0	0	0	0	0	
...	...	...	...	...	...	...	...	...	
40921	0	0	0	1	0	0	0	0	
40922	1	0	0	0	0	0	0	0	
40925	0	1	0	0	0	0	0	0	
40927	1	0	0	0	0	0	0	0	
40928	0	1	0	0	0	0	0	0	

	Causa_9	Causa_10	Causa_11
0	0	0	0
3	0	0	0
6	0	0	0
12	0	0	0
14	0	0	0
...	...	...	...
40921	0	0	0
40922	0	0	0
40925	0	0	0
40927	0	0	0
40928	0	0	0

[34557 rows x 11 columns]

Una vez terminada la normalización de los datos se procedió a graficar la varianza acumulada de cada clase resultante del One hot encoding con la finalidad de obtener el valor ideal de para reducir la dimensionalidad de los datos. Esta reducción de los datos se llevó a cabo mediante la técnica de PCA, cabe recalcar que esta técnica se aplica para ambas parte de los datos seccionados.

Figura 8
Varianza Explicada por los Componentes Principales

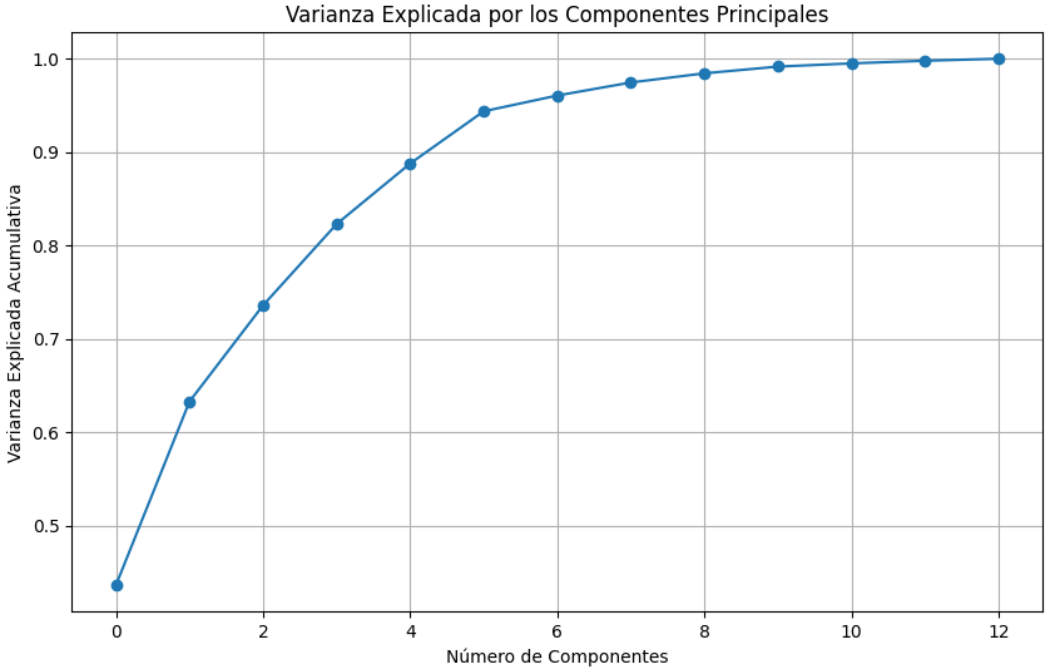


Figura 9
Resultado de la reducción de dimensionalidad mediante PCA

	PC1	PC2	PC3	PC4	PC5	PC6	PC7
0	-0.973156	0.467055	-0.028675	-0.293636	0.288277	-0.212727	-0.057079
1	0.299504	-0.478302	0.146017	-0.198825	0.135791	-0.073855	-0.031276
2	-0.759161	-0.160146	-0.046341	0.270152	-0.189226	-0.098616	-0.088749
3	-0.100925	-0.314368	0.524247	0.100675	0.005431	-0.458035	-0.123762
4	0.122349	1.017231	0.508703	-0.227995	-0.000730	-0.566781	0.564937
...	...	...	...	...	...	...	...
34552	0.304674	-0.483241	0.145776	-0.197983	0.147729	-0.074936	-0.023802
34553	0.304325	-0.483059	0.145935	-0.197800	0.147395	-0.075287	-0.023645
34554	-0.083668	-0.320562	0.513600	0.086805	0.012398	-0.434414	-0.157608
34555	-0.519571	1.169652	-0.128545	-0.101408	-0.208455	-0.232739	-0.012302
34556	0.170595	-0.424123	0.267545	-0.104936	0.090252	-0.193716	-0.087833

34557 rows × 7 columns

Nota. La figura muestra como se redujo la dimensionalidad de las 11 variables creadas por la técnica One Hot encoding a 7 mediante la técnica de PCA.

2.4 Minería de Datos

Una vez terminado el proceso de transformación de datos se continuó con la etapa de minería de datos donde se ajustaron los parámetros estándar que se necesitaba para el entrenamiento de un algoritmo de machine learning. Como se mencionó en la etapa anterior los datos fueron seccionados dos partes, la primera con información conocida la cual en esta etapa se dividió en una proporción de 80-20, donde el 80% fueron usados para el entrenamiento del algoritmo, el 20% se lo utilizó para validación del algoritmo y finalmente los datos de las fallas no identificadas fueron utilizados para la prueba final.

Para la etapa del entrenamiento se consideró tres algoritmos, los cuales fueron Random Forest, SVM y KNN.

2.5 Evaluación e Interpretación

Una vez terminado el entrenamiento de los algoritmos seleccionados, se procedió con la evaluación e interpretación de los resultados obtenidos, se realizó una comparación de los resultados de los algoritmos para posteriormente pueda ser aplicado a la identificación de fallas en el sistema de distribución eléctrico.

Capítulo 3

3. Resultados y Análisis

Para el análisis del rendimiento de los algoritmos puestos a prueba en este trabajo, se tomaron en cuenta las métricas de medición tales como precisión, recall y f1-score, las cuales permitieron observar la efectividad de cada uno de los algoritmos.

3.1 Algoritmo SVM

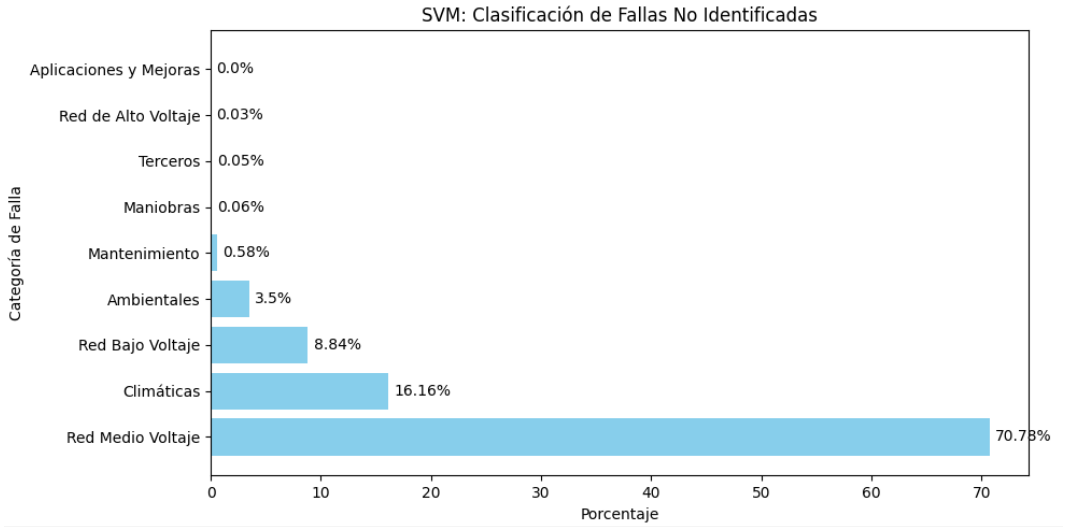
La principal característica que se observó fue el tiempo que le tomó llevar a cabo el entrenamiento, ya que fue una cantidad de tiempo considerable en comparación de los tiempo de entrenamiento de los otros algoritmos, las métricas de medición obtenidas se presentan a continuación en la siguiente tabla:

Tabla 1
Resultado de las principales métricas de medición del algoritmo SVM con datos conocidos

Causas	precisión	recall	f1-score	Support
Ambientales	35,15%	56,71%	43,40%	991
Aplicaciones y Mejoras	0,00%	0,00%	0,00%	70
Baja Frecuencia	0,00%	0,00%	0,00%	21
Climáticas	13,40%	1,49%	2,68%	872
Maniobras	0,00%	0,00%	0,00%	61
Mantenimiento	73,05%	100,00%	84,42%	374
Red Bajo Voltaje	90,32%	75,24%	82,09%	1.761
Red Medio Voltaje	56,28%	80,70%	66,31%	2.254
Red de Alto Voltaje	0,00%	0,00%	0,00%	119
Terceros	60,00%	0,85%	1,67%	355
Transmisor	0,00%	0,00%	0,00%	34
accuracy	59,26%	59,26%	59,26%	59,26%
macro avg	29,84%	28,64%	25,51%	6.912
weighted avg	55,13%	59,26%	53,75%	6.912

Como se observa en la Tabla 1 el algoritmo de SVM obtuvo una precisión del 59.26%, este algoritmo demostró tener limitaciones al momento de manejar las características de las causas teniendo una precisión del 0% en muchas de las causas registradas, pero se observó que en la causa "Red Bajo Voltaje" obtuvo una precisión alta, lo que significa que el algoritmo es bueno identificando dicha clase.

Figura 10
Clasificación de Fallas No Identificadas por el Algoritmo SVM



Como se mostró en la Figura 10 el 70.78% de las fallas no clasificadas el algoritmo les asignó la causa "Red Medio Voltaje lo cual no es una clasificación confiable debido a que la precisión para clasificar esa causa es del 56.26%, mientras que la causa con la que mas dificultades tiene para clasificar es "Aplicación y Mejoras" debido a que no logra clasificar dichas causas.

3.2 Algoritmo KNN

El tiempo de entrenamiento de este algoritmo fue considerablemente menor al tiempo de entrenamiento de SVM, las métricas de medición obtenidas se presentan a continuación en la siguiente tabla:

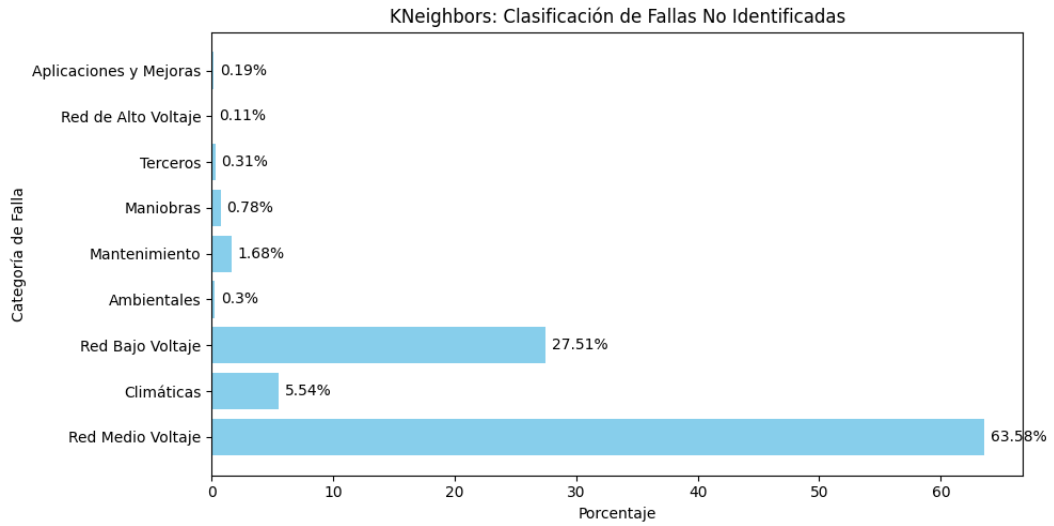
Tabla 2
Resultado de las principales métricas de medición del algoritmo KNN con datos conocidos

Causas	precisión	recall	f1-score	Support
Ambientales	67,88%	35,62%	46,72%	991

Aplicaciones y Mejoras	60,00%	47,14%	52,80%	70
Baja Frecuencia	100,00%	4,76%	9,09%	21
Climáticas	67%	41,06%	50,89%	872
Maniobras	79,63%	70,49%	74,78%	61
Mantenimiento	89,03%	91,18%	90,09%	374
Red Bajo Voltaje	89,44%	84,67%	86,99%	1.761
Red Medio Voltaje	55,85%	89,80%	68,87%	2.254
Red de Alto Voltaje	66,67%	5,04%	9,38%	119
Terceros	71,43%	11,27%	19,46%	355
Transmisor	62,50%	14,71%	23,81%	34
accuracy	67,93%	67,93%	67,93%	67,93%
macro avg	73,58%	45,07%	48,44%	6.912
weighted avg	70,73%	67,93%	65,11%	6.912

Las métricas de medición de este algoritmo mostraron una mejor capacidad para identificar las causas, se demostró una mejor adaptación a la complejidad presentada en los valores de las variables, obtuvo una precisión del 100% en la causa “Baja Frecuencia” la cual tuvo una muestra de datos pequeña sin embargo el algoritmo fue capaz de clasificarla de manera correcta todas las ocasiones que se presentó una falla de “Baja Frecuencia”.

Figura 11
Clasificación de Fallas No Identificadas por el Algoritmo KNN



En este algoritmo al igual que el SVM se mantiene la tendencia asignarle la clasificación de "Red Medio Voltaje" a pesar de no tener una alta precisión clasificando esas causas, aunque el resultado obtenido no fue seguro, el algoritmo KNN resultó mas preciso que el algoritmo SVM, también se demostró que se tiene una mejor distribución de causas no clasificas pero no son del todo confiable debido a la precisión que mostraron con los datos de entrenamiento.

3.3 Algoritmo Random Forest

El tiempo de entrenamiento de este algoritmo fue considerablemente menor al tiempo de entrenamiento de los algoritmos SVM y KNN, las métricas de medición obtenidas se presentan a continuación en la siguiente tabla:

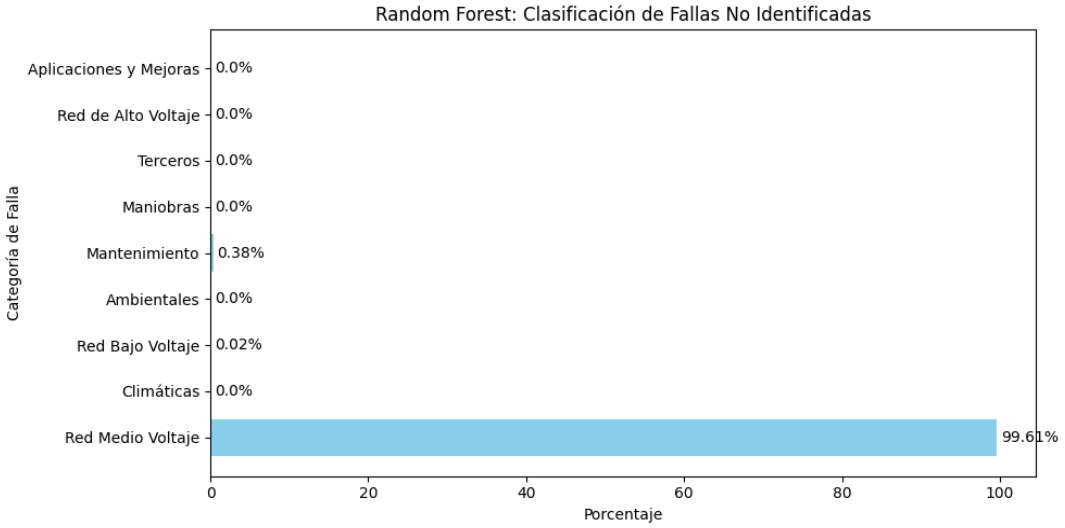
Tabla 3
Resultado de las principales métricas de medición del algoritmo Random Forest con datos conocidos

Causas	precisión	recall	f1-score	Support
Ambientales	98,89%	99,09%	98,99%	991
Aplicaciones y Mejoras	97,22%	100,00%	98,59%	70
Baja Frecuencia	100,00%	100,00%	100,00%	21
climáticas	98,73%	98,39%	98,56%	872
Maniobras	96,83%	100,00%	98,39%	61

Mantenimiento	100,00%	98,93%	99,46%	374
Red Bajo Voltaje	99,94%	99,83%	99,89%	1.761
Red Medio Voltaje	98,93%	98,62%	98,78%	2.254
Red de Alto Voltaje	96,69%	98,32%	97,50%	119
Terceros	95,32%	97,46%	96,38%	355
Transmisor	100,00%	100,00%	100,00%	34
accuracy	98,96%	98,96%	98,96%	98,96%
macro avg	98,41%	99,15%	98,78%	6.912
weighted avg	98,97%	98,96%	98,96%	6.912

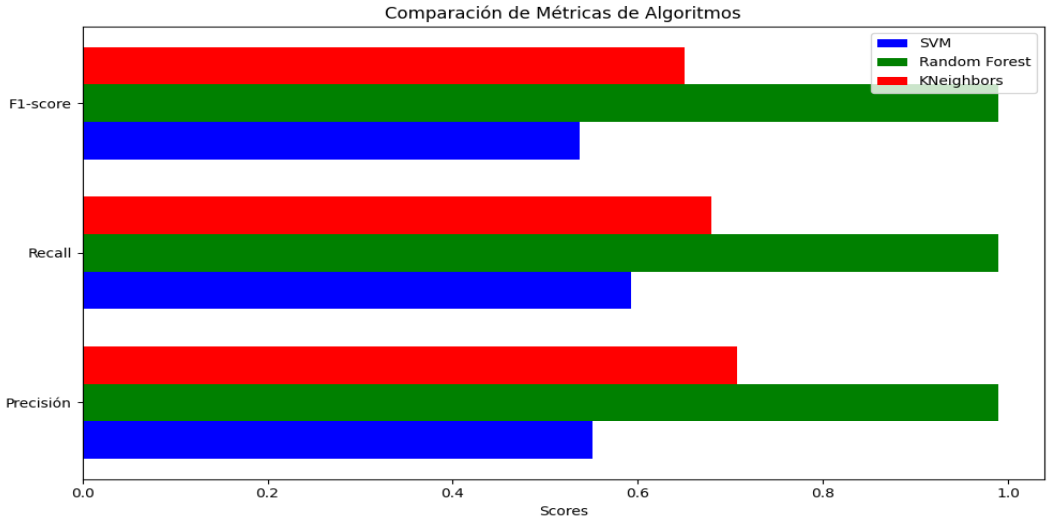
Las métricas de medición del algoritmo Random Forest mostraron resultados muy satisfactorios, con un 98,96% de precisión de manera general y una alta precisión en el manejo de cada una de las causas, lo cual lo convirtió en el algoritmo mas confiable de los puestos a prueba en este trabajo. También demuestra una perfecta clasificación de causas cuyas muestras son pequeñas como por ejemplo la causa “Transmisor”.

Figura 12
Clasificación de Fallas No Identificadas por el Algoritmo Random Forest



Se comprueba que mantiene la tendencia de los algoritmos anteriores debido a que la mayor cantidad de fallas no identificadas las clasifica como “Red de Medio Voltaje”, con la diferencia en que su precisión para clasificar esta falla es del 98,93% lo cual demuestra que el algoritmo es altamente confiable para clasificar fallas en la red de distribución eléctrica.

Figura 13
Comparación de las métricas de medición de los algoritmos puestos a prueba



Se comprobó la superioridad del algoritmo de Random Forest al momento de clasificar causas desconocidas, ya que tuvo una mayor precisión en todas las métricas de medición utilizadas para esta comparación.

Capítulo 4

4. Conclusiones y Recomendaciones

4.1 Conclusiones

Se demostró la importancia que tiene la implementación de técnicas de machine learning en el procesamiento y tratamiento de datos, ya que es una herramienta de mucha eficiencia que logra adaptarse a la complejidad de los datos.

Se demostró que el algoritmo de Random Forest tuvo una gran capacidad al momento de manejar las causas esto debido a la precisión del 98.96% que obtuvo durante el entrenamiento, lo cual lo convierte en un algoritmo muy confiable al momento de clasificar las causas desconocidas que presentó la base de datos. También obtuvo un alto porcentaje en las demás métricas de medición que se utilizaron para comparar los resultados obtenidos.

Durante el entrenamiento se demostró la importancia de ajustar los parámetros del algoritmo, ya que estos tienen una gran influencia en los resultados obtenidos, siendo la precisión la métrica de medición más afectada.

4.2 Recomendaciones

Ser cuidadosos durante el preprocesamiento de datos, debido a que por la complejidad de estos un error en dicha etapa puede comprometer los resultados del algoritmo.

Realizar el ajuste de los parámetros del algoritmo teniendo en cuenta la cantidad de variable y el tipo de estas, puesto que esto puede implicar trabajar con datos desbalanceados afectando directamente la calidad del modelo a entrenar.

Explorar las aplicaciones de las técnicas de machine learning en el sector eléctrico, ya que esta rama de la inteligencia artificial tiene el potencial de mejorar varios aspectos como la eficiencia y fiabilidad del sistema de distribución eléctrica.

Se recomienda la implementación del algoritmo de clasificación de Random Forest ya que demostró altos porcentajes de precisión al momento de clasificar fallas eléctricas, dicha implementación mejoraría la calidad del suministro eléctrico, además de reducir las pérdidas de energía en el sistema de distribución eléctrica.

5. Bibliografía

[1] J. M. H. V. Germán Morales, «Método de localización de fallas en sistemas de distribución basado en gráficas de reactancia,» 2004.

[2] M.-T. A. S. A. Mohammad Reza Shadi, «A real-time hierarchical framework for fault detection, classification, and location in power systems using PMUs data and deep learning,» *International Journal of Electrical Power & Energy Systems,* vol. 134, pp. 107-399, 2022.

[3] D. P. Antonio Parejo, «Monitoring and Fault Location Sensor Network for Underground Distribution Lines,» 2019.

[4] R. D. A. K. Hamid Mirheka, «A novel fault location methodology for smart distribution network,» 2020.

[5] X. Y. Y. W. Jian Qiao, «A multi-terminal traveling wave fault location method for active distribution network based on residual clustering,» vol. 131, 2021.

[6] M. D. Alireza Forouzesh, «Support Vector Machine Based Fault Location Identification in Microgrids Using Interharmonic Injection,» 2021.

[7] C. O.-H. W. J. MArín Quintero, «Toward an adaptive protection scheme in active distribution networks: Intelligent approach fault detector».

[8] J. T. Yong-gang zhang, «A novel displacement prediction method using gated recurrent unit model with time series analysis in the Erdaohe landslide,» 20021.

[9] U. B. R. Tilottama Goswami, «Predictive Model for Classification of Power System Faults using Machine Learning,» 2019.

[10] J.-M. F. O. A. Ndeye Gueye Lo, «Review of Machine Learning Approaches In Fault Diagnosis applied to IoT Systems,» 2019.

[11] C. N. d. Electricidad, «Informa Anual 2022,» CENACE, Quito, 2022.

[12] ARCERNNR, «Estadística Anual y Multianual del Sector Eléctrico Ecuatoriano 2020,» Quito, 2021.

[13] A. D. B. P. D. ECUADOR, «GUÍA DE DISTRIBUCIÓN DE ENERGÍA ELÉCTRICA,» Quito, 2021.

[14] P. M. Michel Gendreau, Transportation and Network Analysis: Current Trends, 2002.

[15] E. B. G. V. Margherita Grandini, Metrics for Multi-Class Classification: an Overview, 2020.

[16] D. A. J. K. M. S. Pedro R. Peres, «How many principal components? stopping rules for determining the number of non-trivial axes revisited,» *Computational Statistics & Data Analysis,* 2005.

[17] A. Grané, «Análisis de componentes principales,» Madrid, 2002.

[18] J. M. N. Céspedes, «Análisis de componentes principales y análisis de regresión para datos categóricos,» *Revista de Matemática: Teoría y Aplicaciones,* 2010.

[19] J. A. D. M. L. Villarroel, «Aplicación del Análisis de Componentes Principales en el Desarrollo de Productos,» 2003.

[20] P. L. R. Carlos Lozares Colina, «EL ANÁLISIS DE COMPONENTES PRINCIPALES: APLICACIÓN AL ANÁLISIS DE DATOS SECUNDARIOS,» 2000.

[21] M. B. J. G. S. E. Pau Rodríguez, «Beyond one-hot encoding: Lower dimensional target embedding,» *Image and Vision Computing,* 2018.

[22] J. Shallcrass Susinos, «Aplicación de técnicas de Machine Learning al estudio del cáncer de endometrio,» 22 Septiembre 2020. [En línea]. Available: https://repositorio.unican.es/xmlui/bitstream/handle/10902/20783/Shallcrass%20Susinos%20Juan.pdf?sequence=1&isAllowed=y.

[23] E. Bologna, «Estimación por intervalo del tamaño del efecto expresado como proporción de varianza explicada,» 2014.

[24] E. J. C. Suárez, «Tutorial sobre Máquinas de Vectores Soporte,» 2013.

[25] K. L. Oren Anava, «k*-Nearest Neighbors: From Global to Local,» *Advances in Neural Information Processing Systems ,* 2016.

[26] X. L. M. Z. X. Z. Shichao Zhang, «Learning k for kNN Classification,» 2017.

[27] I. P. A. V. L. M. Davel Borges Vasconcellos, «Compensación de potencia reactiva en sistemas desbalanceados utilizando algoritmos genéticos,» *Ingeniare. Revista chilena de ingeniería,* 2012.

[28] O. G. Carlos Julio Zapata, «VALORACIÓN DE CONFIABILIDAD DE SISTEMAS DE DISTRIBUCIÓN,» *Scientia et Technica,* 2006.

[29] O. Nigro, D. Xodo, G. Corti y D. Terren, «KDD (Knowledge Discovery in Databases): un proceso centrado en el usuario,» *Red de Universidades con Carreras en Informática (RedUNCI),* pp. 53-58, 2004.