



UNIVERSIDAD TÉCNICA DE COTOPAXI

FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS

CARRERA DE INGENIERÍA EN ELECTRICIDAD

PROPUESTA INVESTIGACIÓN

**PRONÓSTICO DE GENERACIÓN DE UNA MINI CENTRAL
HIDROELÉCTRICA MEDIANTE APRENDIZAJE AUTOMÁTICO
UTILIZANDO SOFTWARE DE CÓDIGO ABIERTO.**

Proyecto de investigación presentado previo a la obtención del Título de
Ingeniero Eléctrico

Autores:

Benalcazar Cisneros Dylan Ariel

Tandalla Cando Jordan Leonel

Tutor:

Ing. MSc. Salazar Achig Edgar Roberto

LATACUNGA - ECUADOR

2024



DECLARACIÓN DE AUTORÍA

Nosotros **BENALCAZAR CISNEROS DYLAN ARIEL** y **TANDALLA CANDO JORDAN LEONEL** declaramos ser autores del presente proyecto de investigación: **“PRONÓSTICO DE GENERACIÓN DE UNA MINI CENTRAL HIDROELÉCTRICA MEDIANTE APRENDIZAJE AUTOMÁTICO UTILIZANDO SOFTWARE DE CÓDIGO ABIERTO**, siendo el Ing. MSc. **SALAZAR ACHIG EDGAR ROBERTO** el tutor del presente trabajo; y eximo expresamente a la Universidad Técnica de Cotopaxi y a sus representantes legales de posibles reclamos o acciones legales.

Además, certifico que las ideas, conceptos, procedimientos y resultados vertidos en el presente trabajo investigativo, son de mi exclusiva responsabilidad.

Latacunga, marzo 2024

.....
Benalcazar Cisneros Dylan Ariel

C.C. 172399837-1

.....
Tandalla Cando Jordan Leonel

C.C. 055043981-4



AVAL DEL TUTOR DE PROYECTO DE TITULACIÓN

En calidad de Tutor del Trabajo de Investigación sobre el título:

“PRONÓSTICO DE GENERACIÓN DE UNA MINI CENTRAL HIDROELÉCTRICA MEDIANTE APRENDIZAJE AUTOMÁTICO UTILIZANDO SOFTWARE DE CÓDIGO ABIERTO”, de los estudiantes: **BENALCAZAR CISNEROS DYLAN ARIEL** y **TANDALLA CANDO JORDAN LEONEL**, de la carrera de Ingeniería en Electricidad, considero que dicho Informe Investigativo cumple con los requerimientos metodológicos y aportes científico-técnicos suficientes para ser sometidos a la evaluación del Tribunal de Validación de Proyecto que el Consejo Directivo de la Facultad de Ciencias de la Ingeniería y Aplicadas de la Universidad Técnica de Cotopaxi designe, para su correspondiente estudio y calificación.

Latacunga, marzo 2024

Ing. MSc. Salazar Achig Edgar Roberto

C.C. 050284761-9



APROBACIÓN DEL TRIBUNAL DE TITULACIÓN

En calidad de Tribunal de Lectores, aprueban el presente Informe de Investigación de acuerdo a las disposiciones reglamentarias emitidas por la Universidad Técnica de Cotopaxi, y por la Facultad de Ciencias de la Ingeniería y Aplicadas; por cuanto, los postulantes: **BENALCAZAR CISNEROS DYLAN ARIEL** y **TANDALLA CANDO JORDAN LEONEL** con el título de Proyecto de titulación: “**PRONÓSTICO DE GENERACIÓN DE UNA MINI CENTRAL HIDROELÉCTRICA MEDIANTE APRENDIZAJE AUTOMÁTICO UTILIZANDO SOFTWARE DE CÓDIGO ABIERTO**” han considerado las recomendaciones emitidas oportunamente y reúne los méritos suficientes para ser sometido al acto de Sustentación de Proyecto.

Por lo antes expuesto, se autoriza realizar los empastados correspondientes, según la normativa institucional.

Latacunga, marzo 2024

Para constancia firman:

Lector 1 (Presidente)

Ing. MSc. Rueda Flores Walter Paul
CC: 050368989-5

Lector 2

Ing. MSc. León Segovia Manuel Ángel
CC: 050204135-3

Lector 3

Ing. MSc. Castillo Fiallos Jessica Nataly
CC: 060459021-6



AVAL DE ACEPTACIÓN

Mediante el presente pongo a consideración que los señores estudiantes **Benalcazar Cisneros Dylan Ariel** y **Tandalla Cando Jordan Leonel** realizaron su tesis a beneficio de la UNIVERSIDAD TÉCNICA DE COTOPAXI con el tema: “**PRONÓSTICO DE GENERACIÓN DE UNA MINI CENTRAL HIDROELÉCTRICA MEDIANTE APRENDIZAJE AUTOMÁTICO UTILIZANDO SOFTWARE DE CÓDIGO ABIERTO**”, trabajo que fue presentado y probado de manera satisfactoria.

Latacunga, marzo 2024

Ing. MSc. Salazar Achig Edgar Roberto

C.C. 050284761-9



AGRADECIMIENTO

Quiero expresar mi más profundo agradecimiento a Dios por brindarme sabiduría, valor y confianza para terminar esta etapa de mi vida. Dirijo mis palabras en un acto de gratitud especial a mis padres, quienes han depositado su confianza inquebrantable en mí, donde su constante apoyo y su fé en mis capacidades han sido los pilares fundamentales que me han permitido terminar esta etapa con éxito y abrazar con orgullo mi profesión.

A todos los docentes de la carrera de Ingeniería Eléctrica que durante toda esta etapa de formación académica me brindaron todos sus conocimientos. De la misma manera al Ing. MSc. Edgar Roberto Salazar Achig como tutor de mi proyecto quien me ha guiado con su paciencia, y sabiduría como docente.

También quiero expresar mi agradecimiento a todas las personas especiales que, de diversas maneras, me han brindado su apoyo a lo largo de este camino.

Benalcazar Dylan



Mi gratitud más profunda va dirigida a Dios, quien ha sido mi guía y fortaleza, brindándome la sabiduría necesaria para culminar con éxito este proyecto de investigación. Asimismo, deseo expresar mi profundo agradecimiento hacia mis padres, Rafael y Beatriz, quienes han sido el pilar fundamental en mi trayectoria universitaria. Su apoyo incondicional, tanto en el ámbito económico como en el moral, ha sido fundamental para mi desarrollo académico y personal. El amor y la dedicación que me han brindado han sido una fuente constante de inspiración y motivación a lo largo de este camino. A ellos les debo gran parte de este logro, y les estoy profundamente agradecido por su constante apoyo y sacrificio.

Tandalla Jordan



DEDICATORIA

Dedico el presente proyecto de investigación a mis padres que me apoyaron y contuvieron los buenos y malos momentos. Quienes me han enseñado a ser la persona que soy hoy, mis principios, mis valores, mi perseverancia y mi empeño son gracias a ustedes. Gracias por enseñarme a afrontar las dificultades durante toda esta etapa universitaria.

A mi hermano, mi ancla emocional y confidente a lo largo de este arduo trayecto. Tu apoyo inquebrantable ha sido la luz que iluminó los momentos oscuros y la celebración en los triunfos. Este proyecto lleva consigo la huella imborrable de tu aliento constante. Gracias por ser mi inagotable fuente de motivación y por ser parte fundamental de este logro. De la misma manera a una persona especial cuya presencia fue constante en mi vida y en este camino académico cuyo apoyo ha sido un respaldo crucial que ha marcado la diferencia agradeciendo la compañía y el respaldo que ha aportado de manera valiosa para poder finalizar este proyecto de investigación.

Benalcazar Dylan



Dedico este proyecto a Dios y a mis padres Rafael y Beatriz, quienes desde el principio han moldeado mi ser tanto en lo personal como en lo académico, guiándome hacia el camino del crecimiento y la honestidad. Su constante apoyo y ejemplo han sido los pilares que me han permitido culminar este trabajo de investigación y completar mi etapa universitaria.

Tandalla Jordan



ÍNDICE GENERAL

DECLARACIÓN DE AUTORÍA	ii
AVAL DEL TUTOR DE PROYECTO DE TITULACIÓN	iii
APROBACIÓN DEL TRIBUNAL DE TITULACIÓN.....	iv
AGRADECIMIENTO	vi
DEDICATORIA.....	viii
RESUMEN	xix
ABSTRACT	xx
CAPÍTULO I.....	1
1. INFORMACIÓN GENERAL	1
CAPÍTULO II.....	2
2. INTRODUCCIÓN.....	2
2.1. PROBLEMA	2
2.1.1. Situación Problémica.....	2
2.1.2. Formulación del problema.....	3
2.2. OBJETO Y CAMPO DE ACCIÓN.....	3
2.2.1. Campo de acción	3
2.3. BENEFICIARIOS	3
2.3.1. Directos.....	3
2.3.2. Indirectos	3
2.4. JUSTIFICACIÓN	4
2.5. OBJETIVOS	4
2.5.1. General.....	4
2.5.2. Específicos.....	4
2.6. REVISIÓN DE LA LITERATURA.....	5
2.7. HIPÓTESIS Y SISTEMAS DE TAREAS	8
2.7.1. Hipótesis	8
2.7.2. Sistema de tareas	8



CAPÍTULO III	10
3. MARCO TEÓRICO	10
3.1. GENERACIÓN DE ENERGÍA EN CENTRALES HIDROELÉCTRICAS	10
3.1.1. Central Hidroeléctrica.....	10
3.1.2. Descripción de la mini Central Hidroeléctrica Catazacón.....	10
3.2. APRENDIZAJE AUTOMÁTICO.....	11
3.2.1. Fundamento teórico del Aprendizaje Automático	11
3.2.2. Tipos de aprendizaje automático	12
3.2.2.1. Supervisados.....	12
3.2.2.2. No supervisados.....	12
3.2.2.3. Mediante refuerzos.	12
3.2.3. Aplicaciones de aprendizaje automático	13
3.3. TÉCNICAS DE APRENDIZAJE AUTOMÁTICO	14
3.3.1. Modelo de Regresión Lineal Simple	14
3.3.1.1. Construcción del modelo de Regresión Lineal Simple	15
3.3.1.2. Hipótesis de modelos de regresión lineal simple.....	17
3.3.2. Modelo con unidad recurrentes cerradas (GRU)	18
3.3.3. Modelo de Redes Neuronales Long Short-Term Memory (LSTM).....	19
3.3.3.1. Historia de las redes neuronales	19
3.3.3.2. Definición de la Red Neuronal	20
3.3.3.3. Aplicaciones de las Redes Neuronales (LSTM).....	20
3.3.3.4. Estructura de la Red Neuronal LSTM	21
3.3.3.5. Niveles o capas de una Red Neuronal LSTM.....	22
3.3.4. Métricas estadísticas para los Modelos de Aprendizaje Automático	22
3.3.4.1. Error Medio Cuadrático MSE.....	22
3.3.4.2. Error Medio Absoluto MAE.....	23
3.3.4.3. Coeficiente de Determinación R²	24
3.3.4.4. Error de Porcentaje Medio Absoluto MAPE	25
3.3.4.5. Errores de Truncamiento	25
3.3.4.6. Redondeo Truncado.....	26
3.4. SOFTWARE DE CÓDIGO ABIERTO	26



3.4.1. Lenguaje de Programación	26
3.4.1.1. Clasificación de los lenguajes de programación.....	26
3.4.2. Uso de software de código abierto	27
3.4.3. Ventajas del software de código abierto.....	28
3.4.4. Python.....	29
3.4.5. Características del Lenguaje Python.....	30
3.4.5.1. Python es un lenguaje interpretado y multiplataforma.	30
3.4.5.2. Tiene un lenguaje multiparadigma.	30
3.4.5.3. Lenguaje de propósito general.....	30
3.4.6. Librerías de Python.....	30
3.4.6.1. Matplotlib	30
3.4.6.2. Numpy	31
3.4.6.3. Scipy	31
3.4.6.4. Pandas.....	31
3.4.6.5. Scikit-Learn	31
3.4.6.6. keras.....	31
3.4.6.7. PyTorch	31
3.4.6.8. sklearn.metrics.....	32
3.4.6.9. Plotly Express	32
CAPÍTULO IV	33
4. METODOLOGÍA.....	33
4.1. TIPOS DE INVESTIGACIÓN.....	33
4.1.1. Investigación descriptiva	33
4.1.2. Investigación Experimental	33
4.1.3. Investigación Cuantitativa	33
4.1.4. Ubicación del Caso de Estudio de la Mini Central Hidroeléctrica Catazacón	34
4.2. TÉCNICA E INSTRUMENTOS.....	34
4.2.1. Programación.....	34
4.2.2. Simulación	35
4.3. MÉTODOLÓGÍA GENERAL	35
4.3.1. Caso de estudio en la Mini Central Hidroeléctrica Catazacón	35



4.3.2. Comparación de los Modelos de Aprendizaje Automático, Lineal, GRU, LSTM.....	36
4.3.3. Procesamiento inicial de la información	38
4.3.4. Selección de variables	38
4.3.5. Depuración de la data de la Mini Central Hidroeléctrica Catazacón	39
4.3.6. Comportamiento de la Potencia Activa	40
4.3.7. Metodología para el análisis de datos	40
4.3.7.1. Comportamiento de la Potencia en la Mini Central Hidroeléctrica Catazacón	41
4.3.8. Validación de los datos.....	42
4.3.9. Implementación en Python	43
4.3.9.1. Estructura del modelo de Regresión Lineal Simple	46
4.3.9.2. Estructura del modelo de Unidades Recurrentes Cerradas GRU	46
4.3.9.3. Estructura del Modelo de Redes Neuronales LSTM.....	47
4.3.9.4. Utilización de librerías en los Modelos de Aprendizaje Automático Lineal Simple, GRU, LSTM	48
4.3.9.5. Lectura de datos generales.....	49
4.3.9.6. Indexación de los datos depurados	50
4.3.9.7. División y normalización de datos	51
4.3.9.8. Agrupamiento de datos.....	52
4.3.9.9. Aplicación del Modelo de Regresión Lineal Simple.....	52
4.3.9.10. Aplicación del Modelo de Unidades Recurrentes Cerradas GRU.....	54
4.3.10. Aplicación del Modelo de Redes Neuronales LSTM.....	58
CAPÍTULO V.....	61
5. ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS	61
5.1. ANÁLISIS DIARIO DE RESULTADOS.....	61
5.1.1. Modelo de Regresión Lineal Simple (Diario)	61
5.1.2. Modelo de unidades recurrentes cerradas GRU (Diario)	62
5.1.3. Modelo de Redes Neuronales LSTM (Diario)	63
5.1.4. Comparación de los Modelos de Aprendizaje Automático	64
5.1.5. Métricas de evaluación de análisis diario	65
5.2. ANÁLISIS SEMANAL DE RESULTADOS	65
5.2.1. Modelo de Regresión Lineal Simple (Semanal).....	65



5.2.2. Modelo de Unidades Recurrentes Cerradas GRU (Semanal).....	67
5.2.3. Modelo de Redes Neuronales LSTM (Semanal).....	68
5.2.4. Comparación de los Modelos de Aprendizaje Automático.....	69
5.2.5. Métricas de evaluación de análisis semanal.....	69
5.3. ANÁLISIS MENSUAL DE RESULTADOS.....	70
5.3.1. Modelo de Regresión Lineal Simple (Mensual).....	70
5.3.2. Modelo de Unidades Recurrentes Cerradas GRU (Mensual).....	71
5.3.3. Modelo de Redes Neuronales LSTM (Mensual).....	72
5.3.4. Métricas de evaluación de análisis mensual.....	73
5.3.5. Comparación de los Modelos de Aprendizaje Automático.....	74
5.3.6. Justificación por el Error De Truncamiento para el Modelo GRU Caso Diario.....	75
CAPÍTULO VI.....	76
6. CONCLUSIONES Y RECOMENDACIONES.....	76
6.1. CONCLUSIONES.....	76
6.2. RECOMENDACIONES.....	77
CAPÍTULO VII.....	78
7. REFERENCIAS.....	78
8. ANEXOS.....	83



ÍNDICE DE FIGURAS

Figura 3.1 Central Hidroeléctrica [8].	10
Figura 3.2 Cuarto de Máquinas de la Central Hidroeléctrica Catazacón	11
Figura 3.3. Diagrama de los algoritmos de aprendizaje automático [11]......	13
Figura 3.4. Recta de regresión con intervalo de confianza y predicción [15].	16
Figura 3.5. Modelo de la celda GRU [16].	18
Figura 3.6. Estructura de la Red Neuronal LSTM [18]......	21
Figura 3.7. Red Neuronal LSTM con 2 Entradas y 1 Salida [17].	22
Figura 3.8. Interrelación entre Software libre y Código abierto [24]......	29
Figura 3.9. Ejecución de un programa en Python [25]......	30
Figura 3.10. Uso de Plotly en gráficos visualmente atractivos [28]......	32
Figura 4.1. Mini Central Hidroeléctrica Catazacón.....	34
Figura 4.2. Diagrama de bloque para el proceso de investigación.....	36
Figura 4.3. Diagrama de Bloques para los Modelos de Aprendizaje Automático.	37
Figura 4.4. Base de datos de los parámetros eléctricos del mes de enero del año 2018.....	38
Figura 4.5. Comportamiento de la Potencia Activa de la Central Hidroeléctrica Catazacón...40	
Figura 4.6. Comportamiento diario real de la potencia activa del día 01 de enero del 2022 ...41	
Figura 4.7. Comportamiento semanal real de la potencia activa del 01 al 07 enero del 2022 .41	
Figura 4.8. Comportamiento mensual real de la potencia activa del mes de enero 2022.....	42
Figura 4.9. Diagrama de Flujo del Modelo de Regresión Lineal Simple	43
Figura 4.10. Diagrama de Flujo del Modelo Unidades Recurrentes Cerradas GRU.....	44
Figura 4.11. Diagrama de Flujo del Modelo de Redes Neuronales LSTM.....	45
Figura 4.12. Estructura del modelo de Regresión Lineal	46
Figura 4.13. Estructura del modelo de Unidades Recurrentes Cerradas GRU.....	46
Figura 4.14. Estructura del Modelo de la Red Neuronal LSTM	47



Figura 4.15. Librerías de Python para la programación del Modelo de Regresión Lineal Simple	48
Figura 4.16 . Librerías de Python para la programación del Modelo Unidades Recurrentes Cerradas GRU.....	49
Figura 4.17. Librerías de Python para la programación del Modelo Redes Neuronales LSTM	49
Figura 4.18. Lectura de datos depurados para la ejecución de los Modelos de Aprendizaje Automático	50
Figura 4.19. Lectura de datos depurados para la ejecución de los Modelos	51
Figura 4.20. División y normalización de datos	51
Figura 4.21. Agrupamiento de datos para el entrenamiento del Modelo de Aprendizaje Automático	52
Figura 4.22. Modelo de Regresión Lineal Simple	52
Figura 4.23. Preparación de datos para predicción.....	53
Figura 4.24. Creación de tabla de resultados de potencia activa pronosticada y real para el Modelo de Regresión Lineal Simple	54
Figura 4.25. Métricas de evaluación del Modelo Regresión Lineal Simple.....	54
Figura 4.26. Modelo de Unidades Recurrentes Cerradas GRU.....	55
Figura 4.27. Preparación de datos para predicción.....	56
Figura 4.28. Configuración de las unidades y épocas del Modelo Unidades Recurrentes Cerradas GRU.....	56
Figura 4.29. Métricas de evaluación para el Modelo de Unidades Recurrentes Cerradas GRU	57
Figura 4.30. Creación de tabla de resultados de potencia activa pronosticada y real para el Modelo de Unidades Recurrentes Cerradas GRU	57
Figura 4.31. Modelo de Redes Neuronales LSTM.....	58
Figura 4.32. Preparación de datos para predicción del Modelo de Redes Neuronales LSTM .	59
Figura 4.33. Configuración de las unidades y épocas del Modelo de Redes Neuronales LSTM	59
Figura 4.34. Métricas de evaluación para el modelo de pronóstico del Modelo de Redes Neuronales LSTM	59



Figura 4.35. Creación de tabla de resultados de potencia activa pronosticada y real para el Modelo de Redes Neuronales LSTM	60
Figura 5.1. Pronóstico de la potencia activa del día 01 de enero del 2022 (Modelo de Regresión Lineal Simple)	62
Figura 5.2. Pronóstico de la potencia activa del día 01 de enero del 2022 (Modelo de Unidades Recurrentes Cerradas de GRU)	63
Figura 5.3. Pronóstico de la potencia activa del día 01 de enero del 2022 Modelo de Redes Neuronales LSTM	64
Figura 5.4. Modelos de Aprendizaje Automático de Pronostico de Potencia Activa: (a) Regresión Lineal Simple, (b) Unidades Recurrentes Cerradas GRU, (c) Redes Neuronales LSTM.....	64
Figura 5.5. Pronóstico de la potencia activa del día 01 al 07 de enero del 2022 (Modelo de Regresión Lineal Simple)	66
Figura 5.6. Pronóstico de la potencia activa del día 01 al 07 de enero del 2022 (Modelo de GRU)	67
Figura 5.7. Pronóstico de la potencia activa del día 01 al 07 de enero del 2022 (Modelo de Redes Neuronales LSTM).....	68
Figura 5.8. Modelos de Aprendizaje Automático de Pronostico de Potencia Activa: (a) Regresión Lineal Simple, (b) Unidades Recurrentes Cerradas GRU, (c) Redes Neuronales LSTM.....	69
Figura 5.9. Pronóstico de la potencia activa del mes de enero del 2022 (Modelo de Regresión Lineal Simple)	71
Figura 5.10. Pronóstico de la potencia activa del mes de enero del 2022 (Modelo de Unidades Recurrentes Cerradas GRU)	72
Figura 5.11. Pronóstico de la potencia activa del mes de enero del 2022 (Modelo LSTM)	73
Figura 5.12. Modelos de Aprendizaje Automático de Pronostico de Potencia Activa: (a) Regresión Lineal Simple, (b) Unidades Recurrentes Cerradas GRU, (c) Redes Neuronales LSTM.....	74
Figura 5.13. Recomendación del $\pm 1\%$ para el Modelo de Unidades Recurrentes Cerradas GRU Caso Diario.....	75
Figura 5.14 Recomendación del -1% para el Modelo de Unidades Recurrentes Cerradas GRU Caso Diario	75



ÍNDICE DE TABLAS

Tabla 2.1. Sistema de tareas en relación con el objetivo específico 1.....	8
Tabla 2.2. Sistema de tareas en relación con el objetivo específico 2.....	9
Tabla 2.3. Sistema de tareas en relación con el objetivo específico 3.....	9
Tabla 4.1. Comparación de los Modelos Utilizados de Predicción.....	36
Tabla 4.2. Variables seleccionadas.....	39
Tabla 4.3. Datos Depurados para los Modelos de Aprendizaje Automático.....	39
Tabla 5.1. Resultado diario del Modelo de Regresión Lineal Simple.	61
Tabla 5.2. Resultado diario del Modelo de Unidades Recurrentes Cerradas.	62
Tabla 5.3. Resultado diario del Modelo de Redes Neuronales.....	63
Tabla 5.4. Resultados de las métricas de evaluación de cada Modelo de Aprendizaje Automático (Diario)	65
Tabla 5.5. Resultado semanal del Modelo de Regresión Lineal Simple.	66
Tabla 5.6. Resultado semanal del Modelo de Unidades Recurrentes Cerradas.	67
Tabla 5.7. Resultado semanal del Modelo de Redes Neuronales.....	68
Tabla 5.8. Resultados de las métricas de evaluación de cada modelo (Semanal)	69
Tabla 5.9. Resultado mensual del Modelo de Regresión Lineal.	70
Tabla 5.10. Resultado semanal del Modelo de Unidades Recurrentes Cerradas.	71
Tabla 5.11. Resultado semanal del Modelo de Redes Neuronales.....	72
Tabla 5.12. Resultados de las métricas de evaluación de cada modelo (Mensual)	73



**UNIVERSIDAD TÉCNICA DE COTOPAXI
FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS**

**TITULO: “PRONÓSTICO DE GENERACIÓN DE UNA MINI CENTRAL
HIDROELÉCTRICA MEDIANTE APRENDIZAJE AUTOMÁTICO UTILIZANDO
SOFTWARE DE CÓDIGO ABIERTO”**

Autores:

Benalcazar Cisneros Dylan Ariel

Tandalla Cando Jordan Leonel

Tutor:

Ing. MSc. Salazar Achig Edgar Roberto

RESUMEN

En este proyecto de investigación, se presentó el desafío de prever la generación de energía en la Mini Central Hidroeléctrica Catazación mediante técnicas de aprendizaje automático y software de código abierto. El objetivo principal fue comparar modelos de pronóstico para determinar la mayor precisión en las predicciones de generación. En este proyecto de investigación se emplearon un total de 87647 datos recopilados a lo largo de cinco años históricos los cuales fueron divididos en 80% de datos para el entrenamiento del modelo y el 20% de datos para prueba. Estos datos fueron utilizados con el propósito de integrarlos en Modelos de Aprendizaje Automático, entre los que se incluyen la Regresión Lineal Simple, las Unidades Recurrentes Cerradas GRU y las Redes Neuronales LSTM los mismos que fueron implementados en el software de código abierto Python. Los resultados mostraron que la aplicación de estos modelos brindaba predicciones útiles y orientadas a decisiones informadas, contribuyendo significativamente a la planificación y gestión eficiente de los recursos energéticos. Adicionalmente, se llevó a cabo la evaluación de los resultados de predicción en diversos horizontes temporales. Se destacó que el Modelo de Unidades Recurrentes Cerradas GRU exhibió una mayor cercanía con la curva de Potencia real. Durante este proceso de evaluación, se analizaron varias métricas, dando como resultado un Error de Porcentaje Medio Absoluto (MAPE) del 1.42% en el caso diario, 1.61% en el caso semanal y 1.82% en el caso mensual.

Palabras Clave: Aprendizaje Automático, hidroeléctrica, python, predicción de generación.



COTOPAXI TECHNICAL UNIVERSITY

ENGINEERING SCIENCES AND APPLIED FACULTY

THEME: “PRONÓSTICO DE GENERACIÓN DE UNA MINI CENTRAL HIDROELÉCTRICA MEDIANTE APRENDIZAJE AUTOMÁTICO UTILIZANDO SOFTWARE DE CÓDIGO ABIERTO”

Authors:

Benalcazar Cisneros Dylan Ariel

Tandalla Cando Jordan Leonel

Tutor:

Ing. MSc. Salazar Achig Edgar Roberto

ABSTRACT

In this research Project, it was present the challenge to see the energy generator in the Catacazon Mini Hydroelectric power plant by learning technique automatic and open source software. the main objective was to compare forecast models to determine the accuracy about generation predictions. In this research Project, it was taken 87647 data collected over 5 years which were divide into 80% data in order to of model training and 20% data for testing. This datas were used for the porpuse to add learning technique automatic, between Simple Linear Regression, GRU Closed Recurrent Units and LSTM Neural Networks, which were added in the open source software Python. The results were showed that the applied of these models give useful predictions and focus to inform decision, helping significantly to the planification and efficient gestion about energy sources. Besides, it was taken the evaluation of the prediction results in differents temporal horizon. It was focused in the GRU Closed Recurrent Units Model showing a great close to the real Power curve. During this evaluation proces, it was analized several metrics as result a mistake of Absolute Percentage (MAPE) of 1.42% daily case, 1.61% weekly case and 1.82% monthly case.

Keywords: learning automatic, hydroelectric, python, generation forecast.

UNIVERSIDAD TÉCNICA DE COTOPAXI
FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS
CARRERA DE INGENIERÍA EN ELECTRICIDAD

Autores:

Benalcazar Cisneros Dylan Ariel
Tandalla Cando Jordan Leonel

AVAL DE TRADUCCIÓN - Profesional Externo

Tandalla Chicaiza Elver Rolando con cédula de identidad número: **0503643991** Licenciado en: **Ciencias de la Educación Mención Inglés** con número de registro de la SENESCYT: 1020-2016-1736825 ; **CERTIFICO** haber revisado y aprobado la traducción al idioma Inglés del resumen del trabajo de investigación con el título: **“PRONÓSTICO DE GENERACIÓN DE UNA MINI CENTRAL HIDROELÉCTRICA MEDIANTE APRENDIZAJE AUTOMÁTICO UTILIZANDO SOFTWARE DE CÓDIGO ABIERTO”** de los estudiantes: **Benalcazar Cisneros Dylan Ariel y Tandalla Cando Jordan Leonel**, egresados de la carrera de **Ingeniería en Electricidad**, perteneciente a la **Facultad de Ciencias de la Ingeniería y Aplicadas**.

En virtud de lo expuesto y para constancia de lo mismo se registra la firma respectiva.

Latacunga, marzo, 2024


.....
Tandalla Chicaiza Elver Rolando
CI: 0503643991
Registro de la SENESCYT
1020-2016-1736825

CAPÍTULO I

1.INFORMACIÓN GENERAL

Título: Pronóstico de generación de una mini central hidroeléctrica mediante aprendizaje automático utilizando software de código abierto.

Tipo de proyecto: Proyecto de Investigación.

Carrera: Carrera de Ingeniería en Electricidad.

Proyecto de investigación asociado: Desarrollo de sistema eléctricos eficientes a nivel residencial comercial e industrial.

Equipo de Trabajo:

Tutor de Titulación:

Ing. MSc. Salazar Achig Edgar Roberto

Estudiante/s:

Dylan Ariel Benalcazar Cisneros

Jordan Leonel Tandalla Cando

Área de Conocimiento: 07 Ingeniería, Industria y Construcción / 071 Ingeniería y Profesiones Afines / 0713 Electricidad y Energía.

Línea de investigación: Energías Alternativas y Renovables, Eficiencia Energética y Protección Ambiental.

Sublíneas de investigación de la Carrera:

Conversión y uso racional de la energía eléctrica.

CAPÍTULO II

2.INTRODUCCIÓN

En el presente documento se realizará el pronóstico de la generación de la central hidroeléctrica Catazación utilizando la potencia generada de cinco años que se los ha tomado como dato inicial con el fin de desarrollar modelos mediante el uso de técnicas de aprendizaje automático. Además, a medida que transcurre el tiempo, ha surgido incertidumbre en el sector eléctrico respecto a los posibles acontecimientos que podrían influir en la generación de energía eléctrica a corto, mediano y largo plazo.

Al utilizar técnicas de aprendizaje automático, como una herramienta fundamental para realizar pronósticos en la producción energética, se aprovecha el registro histórico de generación de la central hidroeléctrica Catazación. Por lo que se analizará la adaptación de las variables cambiante que afectan a los modelos empleados con el objetivo de anticipar posibles desafíos.

2.1.PROBLEMA

2.1.1.Situación Problemática

En el sector eléctrico se ha generado un gran interés en conocer la proyección de generación de energía eléctrica dada a corto, mediano y largo plazo, con el fin de obtener datos satisfactorios. Las predicciones de generación garantizarán una operación eficiente, económica y sostenible de la central, así como para satisfacer las necesidades de energía de la comunidad y la red eléctrica en general.

Existen modelos de predicción relacionados con la generación de energía eléctrica mediante técnicas de aprendizaje computacional.

Los modelos de aprendizaje computacional como Regresión Lineal Simple, Unidades Recurrentes Cerradas (GRU) y comparados con Redes Neuronales (LSTM) se utilizan en la predicción a partir de una base de datos. Sin embargo, es crucial señalar que cada uno de estos modelos presentan un ligero problema en la precisión de resultados.

Al realizar una mala predicción de la generación de energía eléctrica, se puede tener varias consecuencias negativas, como un desequilibrio entre la generación y la demanda, si la

predicción subestima la generación eléctrica, podría haber una falta de electricidad disponible para satisfacer la demanda, lo que podría dar lugar a apagones o cortes de energía. Por otro lado, si la predicción sobreestima la generación, podría existir un exceso de electricidad que no se utilice eficientemente, lo que podría resultar en costos innecesarios.

2.1.2. Formulación del problema

Determinación del modelo de aprendizaje automático para la predicción de generación de energía eléctrica en la mini central hidroeléctrica Catazacón.

2.2. OBJETO Y CAMPO DE ACCIÓN

Pronóstico de la generación de energía eléctrica en la mini central hidroeléctrica Catazacón ubicada en la Provincia de Cotopaxi.

2.2.1. Campo de acción

330000 Ciencias Tecnológicas / 3311 Tecnología de la Instrumentación / 331101 Tecnología de Automatización.

2.3. BENEFICIARIOS

2.3.1. Directos

El beneficiario directo de la investigación será la Empresa Eléctrica Provincial de Cotopaxi S.A. (ELEPCO S.A.), debido a que ayudará a tener una acertada comprensión sobre la predicción de la generación de energía eléctrica mediante modelos de aprendizaje automático.

2.3.2. Indirectos

Los beneficiarios indirectos, en este caso, son aquellos individuos, como los estudiantes y docentes de la carrera de Ingeniería en Electricidad o carreras afines de la Universidad Técnica de Cotopaxi, que participan en la ejecución de investigaciones vinculadas con la predicción de la generación de electricidad.

2.4.JUSTIFICACIÓN

En todo el país, las empresas de electricidad deben enviar la programación de energía al CENACE, los datos deben corresponder a la curva de generación prevista para el día siguiente el cual se deben de enviar hasta las 10 de la mañana. Además, para optimizar la producción, se deben desarrollar modelos de predicción de la energía eléctrica lo que permitirá un mejor ajuste de la generación y, por lo tanto, reducir el riesgo de apagón.

Debido al inconveniente de predecir la generación de electricidad, se necesitan actividades para predecir la generación de energía. Los modelos de aprendizaje automático como Regresión Lineal Simple, Unidades Recurrentes Cerradas y Redes Neuronales LSTM se desarrollan utilizando herramientas de programación accesibles como Python para desarrollar algoritmos para predecir la generación de electricidad.

En la actualidad, la empresa de distribución de energía eléctrica carece de un análisis que permita determinar su capacidad de producción energética. Por ende, resulta esencial llevar a cabo un estudio para facilitar la predicción de la cantidad de energía que la central puede suministrar al sistema en distintos momentos del día. Este análisis se basará en datos históricos de generación proporcionados por la empresa distribuidora ELEPCO S.A., permitiendo así la creación de Modelos de Aprendizaje Automático que aborden y resuelvan esta problemática identificada.

2.5.OBJETIVOS

2.5.1.General

Comparar modelos de pronóstico para la variable de generación utilizando diversas técnicas de aprendizaje automático y software de código abierto, con el fin de determinar qué modelo demuestre mayor precisión en el pronóstico de generación en la mini central hidroeléctrica Catazacón.

2.5.2. Específicos

- Revisar el estado del arte referente a las técnicas de aprendizaje automático, modelos de pronósticos y uso de software de código abierto.

- Aplicar los modelos de pronóstico utilizando los parámetros de generación eléctrica.
- Comparar los resultados de los modelos de pronóstico con los datos reales para determinar la precisión de las predicciones.

2.6. REVISIÓN DE LA LITERATURA

En el artículo científico elaborado por los autores R. Salazar, J.A. González, C.A. Hildago [1], titulado “Predicción de flujo para la generación de energía hidroeléctrica utilizando redes neuronales LSTM” que se encuentra publicado en la revista “ResearchGate” en el cual los autores con esta investigación consiguieron anticipar la producción de energía eléctrica, empleando redes neuronales artificiales y algoritmos de entrenamiento de la biblioteca Keras en Python. Para ello, se utilizaron datos históricos de los años 2010 hasta el 2019 de caudal y potencia activa. Donde los datos utilizados se dividieron y se agruparon para su posterior utilización en el proceso de entrenamiento y pruebas. El diseño de la red neuronal artificial comprendió una capa de entrada con 10 neuronas, dos capas ocultas, dos capas de congelamiento y una capa de salida. El entrenamiento de los modelos se llevó a cabo mediante una plataforma de software en Python, haciendo uso de algoritmos de entrenamiento de la biblioteca Keras. En cuanto a la validación, se empleó el optimizador RMSprop con 10 valores de retraso y 10 neuronas. Es importante destacar que, al lograr predecir la generación de energía eléctrica, se contribuye al Plan Nacional de Desarrollo "Toda una Vida" del Ecuador, con el fin de optimizar la utilización de los recursos hídricos para los niveles de producción requeridos.

En el año 2018, R.M. Aguilar, J.M. Torres, C.A. Martín [2], desarrollaron un artículo acerca del Aprendizaje Automático en el caso de Estudio en la Predicción de la Generación Eléctrica de un Parque Eólico. En este artículo otorgaron relevancia a los datos reunidos durante el proceso, empleándolos para identificar dicho proceso a través del uso de aprendizaje automático. Los autores para lograrlo analizaron las modificaciones requeridas en los datos recopilados de generación de energía eléctrica y a su vez buscaron un algoritmo que sea el más apropiado. El propósito de este artículo es establecer un modelo predictivo mediante el entrenamiento y ajuste del modelo seleccionado. Además, realizaron la programación en el entorno de Lenguaje de programación Python que cuenta con biblioteca especializadas que facilitan este tipo de proyecto de investigación de predicción.

Las técnicas de predicción de variables meteorológicas son ampliamente investigadas por los autores J.A. Segovia, J.F. Toaquiza, J.R. Llanos, D.R. Rivas [3], mencionando que desempeñan un papel fundamental en la gestión eficiente de energías renovables y tienen aplicaciones en diversas disciplinas científicas, como agricultura, salud, ingeniería y energía, entre otras. En este estudio, se ha desarrollado, implementado y comparado una serie de modelos de pronóstico para variables meteorológicas utilizando diversas técnicas de aprendizaje automático. Entre las técnicas implementadas se incluyen regresión lineal múltiple, regresión polinómica, bosques aleatorios, árboles de decisión, XGBoost y redes neuronales perceptrón multicapa (MLP). Para determinar la técnica más adecuada, se emplearon métricas de evaluación que incluyen el error cuadrático medio (RMSE), el error porcentual absoluto medio (MAPE) y el error absoluto medio (MAE).

Mediante la aplicación de redes neuronales artificiales y unidades recurrentes cerradas (GRU) el autor F.S. Bustamante [4], señala que la investigación realizada se enmarca en desarrollar dos sistemas de predicción mediante el uso de redes neuronales artificiales y GRU. Estos sistemas se aplicaron para estimar la generación eléctrica prevista en la Central Hidroeléctrica Illuchi. La metodología utilizada en esta investigación consistió en la recopilación de datos proporcionados por los operadores de ELEPCO S.A., abarcando los años de 2008 a 2020. Las variables de entrada que incluye el autor es la fecha y la energía generada, realizando diversos escenarios con distintas condiciones con el propósito de crear un modelo de Red Neuronal Recurrente que fuera efectivo. Los datos se dividieron en dos grupos: uno para el entrenamiento, que representaba el 70%, y otro para la validación, que correspondía al 30%. Para el proceso de entrenamiento, se empleó el algoritmo ADAM junto con las bibliotecas proporcionadas por Python. Además, el autor consiguió determinar que el error no disminuye constantemente a medida que aumenta el tiempo de entrenamiento. Si se tuviera acceso a una supercomputadora que permitiera la creación de múltiples modelos, resulta innegable que los resultados podrían optimizarse, lo que a su vez conduciría a una significativa reducción en el error entre los datos reales y las predicciones.

La previsión en una forma exacta de producción de energía es un proceso muy relevante para los entes que hacen la gestión de la energía donde H.A. Rayess proveniente de Turquía [5], presenta en su artículo, resultados del uso de técnicas de aprendizaje automático en el pronóstico a corto plazo de la cantidad de energía producida mediante el uso de los datos de

modelos de circulación general (GMC) de la presa Almus y la Central Hidroeléctrica en Tokat, Turquía. Donde el estudio que el realizó demuestra el uso de la Técnicas de Aprendizaje Automático utilizando los datos de manera mensual la energía hidroeléctrica pronosticada por el elemento GCM del periodo de tiempo “2018-2080”, detallándose el uso de los modelos de aprendizaje automático, como el modelo de aprendizaje profundo, arboles potenciados por gradiente, árbol de decisión. El autor representa los resultados con el modelo del gradiente sienta igual a 0,717 significando que el modelo de árboles potenciados por gradiente es el modelo más exitoso para los datos actuales. Sin embargo, recalcó que existen pequeñas diferencias entre los modelos representando ciertos direccionamientos de predicción en sentidos diferentes.

La energía en centrales hidroeléctricas se la puede producir por ciertos tipos de tecnologías con gran alto de eficiencia con el objetivo de producir energía renovable, sin embargo, las centrales hidroeléctricas en cierto modo son directamente afectadas en su caudal, temperatura y el precio de la energía eléctrica, es por esto que los autores J. Barzola, J. Gómez, M. Espinoza y W. Fajardo [6] donde afirman que los pronósticos de generación de energía deben ser confiables y precisos siendo vitales para planificación, programación y operación de los sistemas eléctricos. La investigación propuesta por los autores mantenía el objetivo de desarrollar y aplicar los modelos de aprendizaje automático, como lo es el de redes neuronales artificiales, permitiendo la predicción hidroeléctrica en el corto, mediano y largo plazo en el Ecuador, con datos históricos de generación en la central hidroeléctrica. Además, los autores plantearon dos escenarios que fueron basados en el horizonte de predicción, en otras palabras, problemas de previsión de un paso y varios pasos. Se planteó el entrenamiento y validación de más de 3000 modelos considerando un algoritmo de búsqueda de cuadrícula que es basada en hiperparámetros, presentando los resultados de un modelo unificado y diferenciado de MLP de escenarios de un paso a otras arquitecturas analizadas en escenarios.

La precisión de la previsión de carga a corto plazo puede brindar una gran ayuda a las empresas generadoras para realizar una breve planificación en cuanto a generación. Los autores W. Zhao, X. He, W. Wang y X. Li [7] mencionaron en el artículo que la previsión a corto plazo se la puede predecir con distintas técnicas de aprendizaje automático como el método de unidades recurrentes cerradas GRU haciendo énfasis en predicción de intervalos de carga utilizando datos del precio de la electricidad en tiempos reales, Además se concentraron en la predicción de

carga con los datos históricos reales de carga, la cual les reflejaron ciertos resultados experimentales en la comparación de métodos utilizados para la previsión, llegando a la conclusión que el modelo propuesto tuvo una mayor probabilidad de cobertura y un ancho de intervalo más lineal.

2.7.HIPÓTESIS Y SISTEMAS DE TAREAS

2.7.1.Hipótesis

El desarrollo de modelos de aprendizaje automático en software de código abierto permitirá determinar la predicción de la generación de la mini central hidroeléctrica Catazacón.

2.7.2.Sistema de tareas

Tabla 2.1. Sistema de tareas en relación con el objetivo específico 1.

Objetivo específico 1	Actividad (tareas)	Resultados Esperados	Técnicas, Medios e Instrumentos
Revisar el estado del arte referente a las técnicas de aprendizaje automático, modelos de pronósticos y uso de software de código abierto.	Revisión del estado del arte referente a técnicas de aprendizaje automático.	Fuentes de información de las técnicas de aprendizaje automático más recientes.	Artículos científicos relacionados a técnicas de aprendizaje automático.
	Investigación bibliográfica sobre modelos de predicción (Regresión Lineal, Unidades recurrentes cerradas GRU, Redes Neuronales LSTM).	Información sobre las opciones de software disponible para crear modelos de pronóstico.	Tesis relacionadas a la generación de centrales hidroeléctricas.
			Bibliografías.

Tabla 2.2. Sistema de tareas en relación con el objetivo específico 2.

Objetivo específico 2	Actividad (tareas)	Resultados Esperados	Técnicas, Medios e Instrumentos
Aplicar los modelos de pronóstico utilizando el parámetro de generación eléctrica.	Obtención de los datos históricos de generación de la mini hidroeléctrica.	Base de datos de generación en la mini central hidroeléctrica Catazacón.	Solicitud dirigida a ELEPCO S.A. de los datos.
	Preprocesamiento de los datos.	Base de datos fiable.	Librerías de Software Python.
	Aplicación de los modelos de predicción.	Archivo de Python.	Software Python

Tabla 2.3. Sistema de tareas en relación con el objetivo específico 3.

Objetivo específico 3	Actividad (tareas)	Resultados Esperados	Técnicas, Medios e Instrumentos
Comparar los resultados de los modelos de pronóstico con los datos reales para determinar la precisión de las predicciones.	Ejecución de la programación con datos de prueba y texteo.	Porcentaje de error de los datos obtenidos en comparación de los datos experimentales.	Datos obtenidos en el software de Python.
	Validación de los resultados obtenidos de los modelos de predicción.	Informe de resultados.	Software Python.

CAPÍTULO III

3.MARCO TEÓRICO

3.1.GENERACIÓN DE ENERGÍA EN CENTRALES HIDROELÉCTRICAS

3.1.1.Central Hidroeléctrica

Las centrales hidroeléctricas son infraestructuras que usan la fuerza del agua en movimiento para producir electricidad. Esto se logra al convertir la energía del agua en movimiento en energía eléctrica mediante turbinas conectadas a generadores. Este método es muy eficaz, ya que puede aprovechar hasta un 90% de la energía disponible. Son altamente rentables en comparación con otros tipos de centrales eléctricas. Aunque la construcción inicial puede ser costosa, una vez en funcionamiento, los costos de mantenimiento son bajos, siempre y cuando las condiciones de lluvia sean favorables [8].

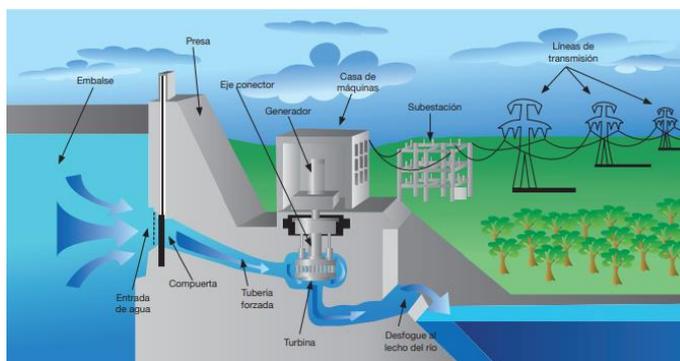


Figura 3.1 Central Hidroeléctrica [8].

3.1.2.Descripción de la mini Central Hidroeléctrica Catazacón

En el año 2009 V.J. Quispe [9] describe a la Central Hidroeléctrica de Catazacón que está situada en el cantón Pangua, dentro de la provincia de Cotopaxi, está bajo la administración de la empresa eléctrica provincial Cotopaxi ELEPCO S.A. Esta central, que comenzó a operar en 1991, se distingue por sus notables especificaciones técnicas y eléctricas.

Uno de sus puntos destacados es la presencia de dos grupos generadores que aprovechan la energía potencial del agua. Además, la central dispone de un tanque de presión, situado a una distancia de 200 metros y a una altura de 150 metros con respecto al centro de la instalación. Esta infraestructura funciona a un voltaje de 440 Voltios y tiene una capacidad de 1000 kVA,

la cual esta energía es elevada por la subestación Catazacón que está equipada con un transformador con una conexión Ynd11. El voltaje en alta es de 13.8 kV, mientras que en baja es de 0.44 kV. Desde la central, la energía eléctrica se transmite a través de una red trifásica de 13.8 kV mediante un conductor de tipo ACSR de calibre 3/0. La electricidad generada se integra en el sistema nacional interconectado, desempeñando un papel fundamental en la provisión de energía eléctrica en la región.



Figura 3.2 Cuarto de Máquinas de la Central Hidroeléctrica Catazacón

3.2. APRENDIZAJE AUTOMÁTICO

3.2.1. Fundamento teórico del Aprendizaje Automático

Para comprender el aprendizaje automático, es esencial entender primero qué implica el aprendizaje en general. Este proceso implica el desarrollo de las habilidades de una persona para llevar a cabo tareas, utilizando información disponible que permite comprender el funcionamiento de ciertas cosas.

En el ámbito de la ingeniería, el aprendizaje automático, también conocido como aprendizaje artificial, se centra en entrenar a las computadoras para que aprendan a realizar tareas específicas. Estos algoritmos se utilizan para resolver problemas y tomar decisiones basadas en la información proporcionada. Es importante mencionar que requieren una cantidad considerable de datos para funcionar de manera efectiva. El aprendizaje automático se basa en una estructura clave:

$$\text{Aprendizaje} = \text{Selección} + \text{Adaptación}$$

Este proceso se desarrolla en dos etapas: primero, el algoritmo identifica las características más relevantes, realiza comparaciones y, cuando detecta diferencias significativas, ajusta su modelo y lo adapta según el objeto analizado [10].

3.2.2. Tipos de aprendizaje automático

3.2.2.1. Supervisados

En este método, es vital ofrecer ejemplos que provean la información necesaria para facilitar el aprendizaje y alcanzar los objetivos deseados. Este enfoque es similar a aprender con la orientación de un profesor, como recibir clases que presentan ejemplos claros para comprender un tema específico [10].

3.2.2.2. No supervisados.

El método no supervisado tiene como objetivo descubrir nuevos conocimientos al identificar patrones, estructuras o relaciones presentes en los datos. Esto se logra sin depender de resultados predefinidos o esperados, permitiendo así explorar la información sin la guía de objetivos específicos [10].

3.2.2.3. Mediante refuerzos.

Este método representa prácticamente una fusión de los dos enfoques previamente mencionados. En esta técnica, el sistema se enfrenta a un problema para resolver. Su aprendizaje se lleva a cabo mediante una señal de evaluación proporcionada por un guía o el entorno, indicando si ha resuelto de manera correcta el problema planteado [10].

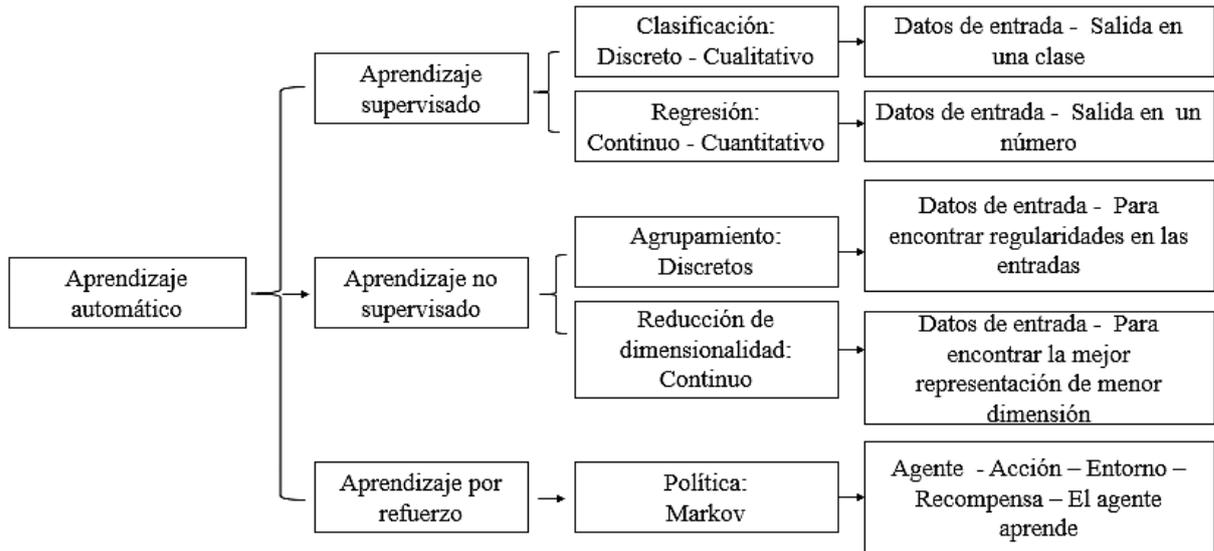


Figura 3.3. Diagrama de los algoritmos de aprendizaje automático [11].

3.2.3. Aplicaciones de aprendizaje automático

En este artículo R. Vila, M. T. Fonseca, M. R. Álvarez [12] detallan los pasos fundamentales para llevar a cabo un sólido modelo de regresión lineal múltiple con la ayuda de la potente herramienta de análisis SPSS el cual utilizado para análisis estadísticos. Para construir este modelo matemático se siguen 5 pasos, estos pasos críticos engloban desde la crucial selección de la variable dependiente que buscamos predecir hasta la cuidadosa elección de las variables independientes que funcionarán como predictores claves. Así mismo, se explora la etapa de validación de los datos para garantizar que estos cumplan con las suposiciones necesarias, seguida de una detallada interpretación del modelo resultante para evaluar la importancia estadística de las variables predictoras. Finalmente, se pone énfasis en la esencial evaluación del ajuste del modelo, determinando si este posee la capacidad necesaria para predecir de manera efectiva los valores de la variable dependiente. La cuidadosa ejecución de estos pasos resulta esencial en la construcción de un robusto modelo de regresión lineal múltiple.

Por otro lado, este artículo ejemplifica de manera clara la aplicación práctica de esta técnica en un estudio relacionado con la violencia de género, ofreciendo un valioso vistazo a cómo la regresión lineal múltiple se convierte en una herramienta poderosa y versátil en la investigación y análisis de fenómenos complejos en una diversidad de campos y disciplinas.

En el presente proyecto C. O Duarte [13] se enfoca en estimar la demanda eléctrica en hogares residenciales mediante un método de regresión propuesto, debido a las limitaciones evidenciadas en el método actual de estimación eléctrica en Ecuador. Estas limitaciones se originan en un estudio original que se basó en datos de usuarios residenciales del suroeste de los Estados Unidos, lo que introdujo variaciones significativas en términos de condiciones climáticas, cultura energética y patrones de consumo eléctrico.

Los datos utilizados provienen de CNEL Guayas-Los Ríos y se obtuvieron a través de una empresa contratista en cumplimiento de regulaciones de la Agencia de Regulación y Control de Energía y Recursos No Renovables. Estos datos se recopilan de manera aleatoria anualmente y se descargaron en formato PQ log, que posteriormente se convirtió a formato xls de Excel. La fase de análisis se centró en la selección de lecturas clave de armónicos para su examen detallado, con el fin de estimar variables temporales y calcular la potencia en un contexto específico. Los resultados se registraron en una nueva hoja de cálculo en Excel para preservar la integridad de la base de datos primaria. Luego, se importaron a Matlab, donde se aplicaron técnicas de regresión, destacando los métodos de regresión lineal y polinomial de segundo grado por su mayor precisión en la aproximación.

3.3.TÉCNICAS DE APRENDIZAJE AUTOMÁTICO

3.3.1.Modelo de Regresión Lineal Simple

En relación con el modelo de regresión lineal, este enfoque implica la incorporación de un conjunto de variables independientes, también conocidas como variables predictoras, junto con un conjunto de variables dependientes. Este modelo, basado en ecuaciones lineales, facilita la predicción de la variable de respuesta mediante las variables de entrada incorporadas. En este marco, los parámetros relacionados con las variables dependientes se ajustan en función de la medida predictiva. En esencia, el modelo de regresión se esfuerza por minimizar el error, buscando formalizar o seleccionar parámetros que reduzcan al mínimo la suma de los errores al cuadrado, lo que se asemeja a la estimación de mínimos cuadrados ordinarios [14].

3.3.1.1. Construcción del modelo de Regresión Lineal Simple

De acuerdo con la construcción del modelo, se toma en consideración factores que influyen en una variable de respuesta o predictiva dividiéndose en dos grupos, donde uno de ellos contiene la variable que puede tomar el nombre de explicativa “X” y que se supone para observar la variable de respuesta “Y”, mientras que el otro grupo puede ser tomado en cuenta para los que influyen en pequeñas magnitudes para el error, entonces:

Al suponer que se tiene dos variables “X” y “Y” estarán relacionada linealmente, donde para cada valor de “X”, la variable dependiente, siendo así “Y” una variable aleatoria.

$$Y_i = \beta_0 + \beta_1 X_i + \varepsilon \quad (3.1)$$

Donde:

Y_i es el valor de la variable predictiva.

X_i es la variable explicativa con ciertos valores conocidos.

β_0 es un parámetro desconocido, representa el valor medio de “Y” cuando “X” es cero.

β_1 es un parámetro desconocido, muestra el incremento del valor medio de “Y” cuando “X” aumenta en una unidad.

ε es un error aleatorio con media cero y varianza σ^2

La ecuación anterior es conocida como “modelo de regresión lineal simple”. La suposición de este modelo es que el valor esperado del error $E(\varepsilon) = 0$, observándose que el valor esperado de la variable “ Y_i ”, para cada valor de “ X_i ”, donde la linealidad viene dada de la siguiente manera:

$$E(Y_i|X_i) = \beta_0 + \beta_1 X_i \quad (3.2)$$

Se toma la esperanza de la ecuación 1, condicionada al valor “ X_i ”.

Como una segunda suposición, se tiene la “homocedasticidad”, la cual corresponde la varianza de la distribución de “ Y_i ” con respecto al valor de “ X_i ”, denotado de la siguiente manera:

$$Var(Y_i|X_i) = \sigma^2 \quad (3.3)$$

Por tanto, se puede decir que esta varianza corresponderá a la medida de dispersión de los resultados sobre la recta de regresión lineal, es decir con respecto a “ Y_i ”.

Como una tercera suposición, se denomina a la “Normalidad” la cual es considerada como la distribución de “ Y ” para cada valor correspondiente de “ X ” sea normal, por lo tanto:

$$Y_i \rightarrow N(\beta_0 + \beta_1 X_i, \sigma^2) \quad (3.4)$$

Como un cuarto y último supuesto, se habla de la “independencia” o covarianza, cuya suposición puede definir las observaciones de la variable de respuesta “ Y_i ” sean independientes entre sí, entonces:

$$Cov[Y_i, Y_k] = 0 \quad (3.5)$$

La pendiente de una ecuación de regresión representa el impacto de la variable predictora o pronosticada en la variable que será la respuesta. En el proceso de este método de regresión, se emplean los mínimos cuadrados, cuyos nombrados también como mínimos cuadrados de regresión, los cuales definen la línea que reduce al mínimo la suma de las distancias verticales al cuadrado desde los puntos de regresión hasta la recta.

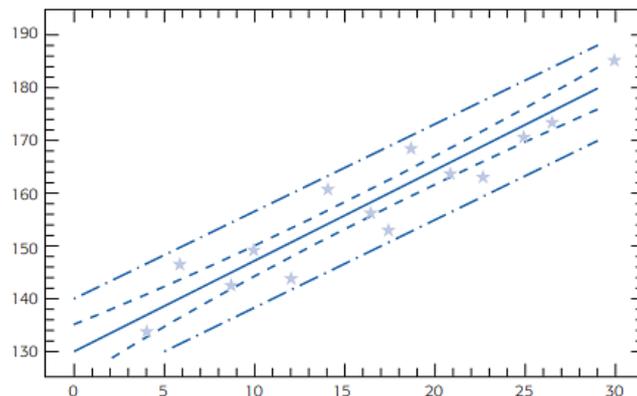


Figura 3.4. Recta de regresión con intervalo de confianza y predicción [15].

3.3.1.2. Hipótesis de modelos de regresión lineal simple

Se presenta un resumen de las hipótesis respectivas a considerar en este modelo son las siguientes:

3.3.1.2.1. Linealidad

La esperanza de la respuesta “Y” depende linealmente de la variable siguiente “X”. Donde la esperanza condicionada al valor de “X” es:

$$E(Y_i|X_i) = \beta_0 + \beta_1 X_i \quad (3.6)$$

3.3.1.2.2. Homocedasticidad

La varianza de la distribución de la variable “Y” dada una “X” es constante viene dado de la siguiente manera:

$$\text{Var}(Y_i|X_i) = \sigma^2 \quad (3.7)$$

3.3.1.2.3. Independencia

Las observaciones en esta hipótesis con respecto a “Y” son independientes entre sí, por lo que se muestra a continuación:

$$\text{Cov}[Y_i, Y_k] = 0 \quad (3.8)$$

3.3.1.2.4. Normalidad

En esta hipótesis de normalidad se realiza la distribución para la variable “Y” con respecto a cada valor de “X” sea normal, entonces:

$$Y_i \rightarrow N(\beta_0 + \beta_1 X_i, \sigma^2) \quad (3.9)$$

3.3.2. Modelo con unidades recurrentes cerradas (GRU)

La estructura de la unidad recurrente cerrada se considera una evolución del modelo LSTM, incorporando características que le proporcionan ciertas ayudas y beneficios. Esta unidad mantiene tanto una puerta de actualización como una puerta de reinicio, fusionando la ventana de olvido de LSTM con la mencionada puerta de entrada. Este enfoque facilita la creación de una puerta de actualización, al mismo tiempo que representa la combinación del estado oculto de la celda, abordando así la posible solución al problema del gradiente de fuga. También mantiene el enfoque donde, la unidad recurrente cerrada GRU elimina la puerta de salida y devuelve el vector de estado en cada instante de tiempo.

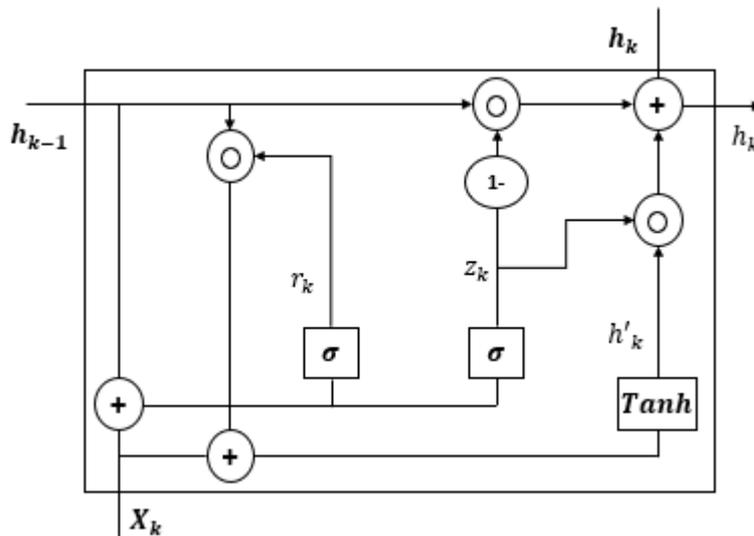


Figura 3.5. Modelo de la celda GRU [16].

En la Figura 3.5 se representa el modelo de la celda GRU donde, X_k y h_k son la entrada y salida en la unidad k . Entonces h_k es la memoria que se transmite a través de las unidades; h'_k es la memoria interna; y, z_k y r_k son las puertas de actualización y reinicio, respectivamente.

Mientras que σ_g representa a la función de activación sigmoide; $tanh$ a la tangente hiperbólica; y, el operador \odot se lo denomina la multiplicación de matrices elemento a elemento o producto Hadamard [16].

$$z_k = \sigma_g(W_z X_k + U_z h_{k-1} + b_z) \quad (3.10)$$

La puerta de actualización z_k controla cuanto del estado anterior h_{k-1} y la entrada actual x_k debe influir en h_k y h'_k .

$$r_k = \sigma_g(W_r X_k + U_r h_{k-1} + b_r) \quad (3.11)$$

La ecuación anterior representa, la puerta del reinicio r_k la cual define cuanta información de h_{k-1} y x_k debe olvidarse para h'_k .

$$h'_k = \tanh(W_h X_k + U_h (h_{k-1} \odot r_k) + b_h) \quad (3.12)$$

La memoria interna h'_k se define en base a x_k y la información que r_k dejó pasar de $h_k - 1$.

$$h_k = h_{k-1} \odot (1 - z_k) + h'_k \odot z_k \quad (3.13)$$

Se utiliza el complemento $1 - z_k$ para actualizar los valores de $h_k - 1$ debido a que z_k está cercana a 1. Por tanto z_k actualiza los valores de h'_k , el cual añade la nueva información a h_k para compensar el efecto de $(1 - z_k)$.

3.3.3. Modelo de Redes Neuronales Long Short-Term Memory (LSTM)

3.3.3.1. Historia de las redes neuronales

Las redes neuronales tienen sus raíces en el año 1936, cuando Alan Turing exploró la analogía del cerebro humano como un paradigma para el procesamiento computacional. Posteriormente, en 1949, Donald Hebb proporcionó las primeras explicaciones de los procesos de aprendizaje desde una perspectiva psicológica. En 1950, Karl Lashley agregó una nueva dimensión al descubrir que la información no se almacenaba de manera centralizada, sino que estaba distribuida en todo el sistema nervioso [17].

En 1957, Frank Rosenblatt llevó a cabo estudios pioneros en el campo con la creación y análisis del Perceptrón, la primera red neuronal formalizada. Durante la década de 1960, Bernard y Hoff desarrollaron el modelo Adaline, representando un hito al aplicar las redes neuronales en casos reales.

En 1974, Paul Werbos presentó una idea revolucionaria al desarrollar los fundamentos del algoritmo de aprendizaje conocido como retro propagación, o regresión, marcando un avance significativo en el campo de las redes neuronales.

A partir de 1986, el interés y la investigación en el campo de las redes neuronales experimentaron un renacimiento notable. Desde entonces, numerosos avances y trabajos significativos han sido desarrollados, impulsando continuamente el progreso y la aplicabilidad de estas redes en una variedad de campos y aplicaciones [17].

3.3.3.2. Definición de la Red Neuronal

Las redes neuronales son sistemas computacionales compuestos por elementos llamados neuronas, interconectadas entre sí. Cada neurona desempeña una tarea simple dentro de la red, pero cuando trabajan en conjunto, son capaces de procesar grandes volúmenes de información y tomar decisiones. Esta comunicación entre neuronas permite resolver problemas de manera eficiente. En otras palabras, las redes neuronales son sistemas informáticos inspirados en el funcionamiento del cerebro humano [17].

3.3.3.3. Aplicaciones de las Redes Neuronales (LSTM)

Las redes neuronales son herramientas útiles con una amplia gama de aplicaciones en campos comerciales, militares y más allá. Se destacan por su capacidad para realizar tareas específicas de manera eficiente, a menudo superando a otras tecnologías. La ventaja clave de las redes neuronales radica en su capacidad para procesar información de manera paralela, adaptativa y no lineal. Además, se pueden clasificar según su aplicación en áreas como asociación y clasificación, regeneración de patrones, regresión y generalización, y optimización [17].

3.3.3.3.1. Predicción de la demanda de energía

Las Redes Neuronales LSTM tienen la capacidad de prever cuánta energía eléctrica se necesitará en el futuro, basándose en datos pasados, cambios estacionales y condiciones climáticas. Esto es crucial para planificar y operar eficientemente las redes eléctricas, mejorando así la producción y distribución de energía.

3.3.3.3.2. Diagnóstico de fallas y mantenimiento predictivo

Estas redes neuronales pueden ser utilizadas para detectar problemas en equipos eléctricos, como transformadores, generadores y líneas de transmisión. Al analizar datos de sensores y registros históricos, las LSTM pueden identificar señales de alerta temprana de posibles fallas, permitiendo realizar mantenimiento preventivo y evitar cortes de energía no programados.

3.3.3.4. Estructura de la Red Neuronal LSTM

Una celda en una red neuronal recurrente LSTM es como una pequeña unidad que almacena información importante a largo plazo (representada por la letra C), produce salidas (representadas por la letra H) y toma entradas (representadas por la letra X). En esta celda, las partes en amarillo son como pequeñas redes neuronales simples que usan funciones de activación para procesar la información, mientras que los puntos en rosa realizan operaciones especiales para eliminar información innecesaria, actualizar la memoria y calcular el nuevo estado oculto [18].

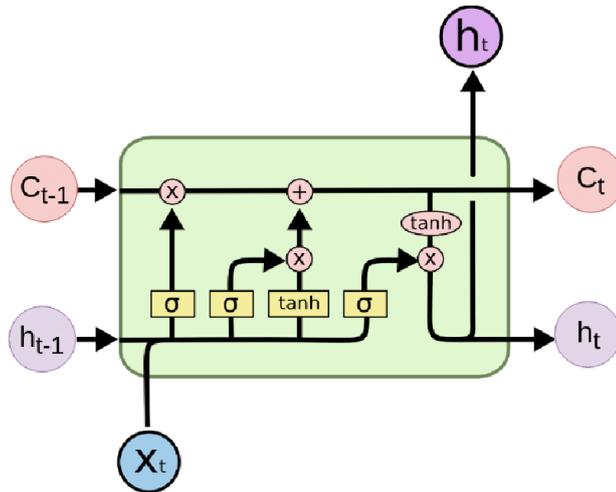


Figura 3.6. Estructura de la Red Neuronal LSTM [18].

Los valores de entrada se multiplican por los pesos que la neurona ya tiene asignados. Estos pesos, que suelen ser flexibles, determinan cuánto impacto tienen los valores de entrada. Por ejemplo, incluso un valor de entrada grande puede tener un impacto pequeño si los pesos son lo suficientemente bajos.

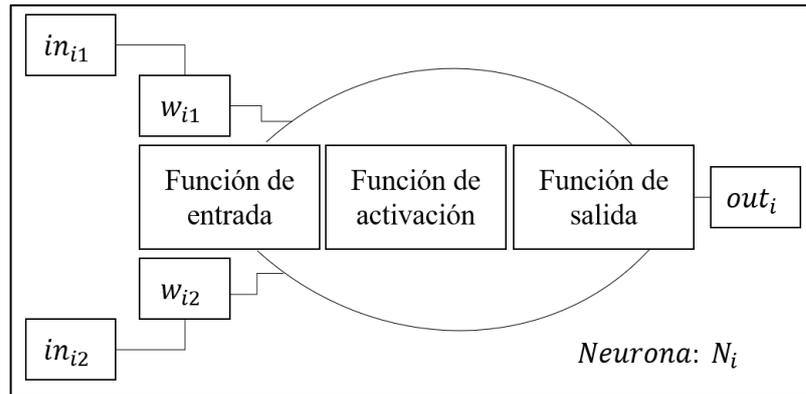


Figura 3.7. Red Neuronal LSTM con 2 Entradas y 1 Salida [17].

3.3.3.5. Niveles o capas de una Red Neuronal LSTM

La red neuronal organiza sus neuronas en niveles o capas, con un número específico en cada una. Hay tres tipos de capas según su posición en la red.

Capa de entrada: Recibe información directamente desde fuentes externas.

Capas ocultas: Internas a la red y no interactúan directamente con el entorno externo. Pueden variar en número y están conectadas de diferentes maneras, determinando las diferentes estructuras de redes neuronales.

Capa de salida: Envía información procesada por la red hacia el exterior.

3.3.4. Métricas estadísticas para los Modelos de Aprendizaje Automático

3.3.4.1. Error Medio Cuadrático MSE

Es considerada como una métrica importante comúnmente utilizada en problemas de regresión que es calculada como la media de los errores al cuadrado entre las predicciones del modelo y los valores reales.

$$mse = \frac{1}{n} \sum (y_n - \hat{y})^2 \quad (3.14)$$

Donde:

n es considerado el número de observaciones

y_n es el valor real de la observación

\hat{y} es el valor de predicción del modelo para la observación

Además, se puede decir que el error medio cuadrático busca medir la magnitud promedio de los errores cuadráticos entre las predicciones y los valores reales, por lo que, al elevar al cuadrado, esta métrica penaliza de manera más pronunciada los errores grandes en comparación con los de menos valor. Tiene como característica principal que el resultado de este error es un valor que mayor a cero, lo cual indica que entre valores más bajos de este error indica un mejor ajuste del modelo a los datos. De manera más profunda, esta métrica conlleva a evaluar la calidad de los valores de predicción [19].

3.3.4.2. Error Medio Absoluto MAE

El Error Absoluto Medio es una métrica que es utilizada en problemas de regresión lineal simple y múltiple la cual se calcula como la media de las diferencias absolutas entre las predicciones del modelo y los valores reales.

$$mae = \frac{1}{n} \sum |y_i - \hat{y}| \quad (3.15)$$

Donde:

n : es considerado el número de observaciones

y_i : es el valor real de la observación

\hat{y} : es el valor de predicción del modelo para la observación

Esta métrica de regresión proporciona una medida intuitiva de la magnitud promedio de los errores en la misma escala que la variable objetivo, a su vez, el error medio absoluto también resulta ser un número mayor a cero o valor no negativo lo que indica que será menos sensible a valores atípicos en comparación con el error medio cuadrado debido a la falta de elevación al cuadrado. De manera más específica el error medio absoluto verifica la precisión de los valores de predicción proporcionados en el modelo de regresión ya sea simple, múltiple o polinómico,

donde se busca una medida que sea más resistente a valores atípicos que proporcione una interpretación más directa de la magnitud de errores promedios [19].

3.3.4.3. Coeficiente de Determinación R^2

La métrica del R^2 es utilizada de forma estadística que proporciona información sobre la proporción de la variabilidad en la variable dependiente que es explicada por el modelo de regresión. También es llamado como coeficiente de determinación el cual evalúa la habilidad de un modelo para anticipar eventos futuros. Un rendimiento óptimo se logra con un valor de 1, indicando que la predicción coincide perfectamente con los valores reales de la variable objetivo. Es importante señalar que R^2 puede asumir valores negativos, lo que reflejaría predicciones considerablemente deficientes. Cuando la predicción se alinea con la expectativa basada en la media de los valores de la variable objetivo, el coeficiente R^2 es igual a 0. Su definición implica restar de 1 la proporción de la suma de cuadrados totales entre la suma de cuadrados de los residuos [19].

$$R^2 = 1 - \frac{\sum(y_i - \hat{y})^2}{\sum(y_i - \bar{y})^2} \quad (3.16)$$

Donde:

y_i : es el valor real de la observación

\hat{y} : es el valor de predicción del modelo para la observación

\bar{y} : es el valor de la media de los valores reales

La métrica del R^2 se interpreta como el porcentaje de variabilidad explicada por el modelo en relación con la variabilidad total, por lo que entre más alto sea este valor generalmente indica un mejor ajuste del modelo de predicción a su vez, la métrica antes mencionada proporciona información sobre la capacidad explicativa del modelo en relación con la variabilidad total de la variable dependiente. Sin embargo, debe usarse con precaución y junto con otras métricas para una evaluación más completa del rendimiento del modelo [19].

3.3.4.4. Error de Porcentaje Medio Absoluto MAPE

El MAPE, también referido como MAPD, es una métrica que evalúa la precisión de un método utilizado en la construcción de valores en series temporales en estadística. Esta métrica nos proporciona información sobre qué tan cercanos están los resultados obtenidos a la realidad. Es útil para realizar ajustes en los datos y mejorar la exactitud de los modelos estadísticos utilizados [20].

$$\text{MAPE} = \frac{100}{N} \cdot \sum_{i=1}^N \left| \frac{x_i - \hat{x}_i}{x_i} \right| \quad (3.17)$$

x_i : Observaciones actuales de las series de tiempo.

\hat{x}_i : Series de tiempo estimadas o pronosticadas.

N : Número de punto de datos no faltantes.

Cuando el MAPE tiene valores más bajos indican un mejor ajuste. Si un modelo individual no tiene los valores más bajos para las tres medidas de precisión, MAPE es generalmente la medida más recomendable. Las medidas de precisión se basan en los residuos de un período adelantado. En cada momento en el tiempo, el modelo se utiliza para predecir el valor Y correspondiente al próximo período temporal. La discrepancia entre los valores pronosticados y el valor real de Y son los residuos de un período adelantado. Por lo tanto, las medidas de precisión ofrecen una indicación de la precisión que se podría esperar al predecir un período desde el final de los datos. Por ende, no indican la precisión al predecir más de un período.

3.3.4.5. Errores de Truncamiento

Los errores de truncamiento suceden cuando tratamos de aproximar la solución exacta de un problema usando un método numérico. Por ejemplo, si intentamos calcular la función exponencial usando la serie de Taylor, necesitamos detenernos en algún punto porque no podemos considerar infinitos términos. El problema es que, al detenernos, estamos dejando fuera algunos términos de la serie, lo que introduce un error en nuestra aproximación. Este error

se llama error de truncamiento y es independiente de cómo hagamos los cálculos, solo depende del método numérico que estemos utilizando [21].

3.3.4.6.Redondeo Truncado.

El redondeo truncado consiste en truncar el resultado de una operación al número de cifras significativas que se estén utilizando. Por ejemplo, si redondeamos $\frac{7}{9}$ a 4 cifras significativas tenemos 0.7777.

3.4.SOFTWARE DE CÓDIGO ABIERTO

3.4.1.Lenguaje de Programación

A lo largo de la historia, los lenguajes de programación han sido una constante en la evolución de la tecnología. Algunos han resistido el paso del tiempo y siguen siendo ampliamente utilizados en la actualidad, mientras que otros experimentaron un auge en su momento para luego desaparecer. Estos lenguajes juegan un papel fundamental al permitir describir la estructura de la información que se quiere analizar, así como detallar las acciones necesarias para llevar a cabo tareas específicas.

3.4.1.1.Clasificación de los lenguajes de programación

3.4.1.1.1.Por su nivel de abstracción

La clasificación de los lenguajes de programación según su nivel de abstracción se divide en tres categorías principales: bajo, medio y alto. En el nivel de abstracción bajo, se hace alusión al lenguaje directamente interpretado por las máquinas, que se expresa en códigos binarios. Estos códigos resultan incomprensibles para la mayoría de las personas debido a su complejidad y estructura técnica. En el nivel de abstracción medio, se encuentran lenguajes como C++ que, si bien son más comprensibles para los humanos en comparación con el código binario, aún implican un nivel de detalle y complejidad que puede resultar desafiante para muchos.

Por otro lado, los lenguajes de alto nivel representan una categoría más cercana al entendimiento humano. Ejemplos notables incluyen Python, donde la expresión de ideas se

vuelve más intuitiva y clara, facilitando a los programadores la comunicación de conceptos complejos de manera más comprensible y menos centrada en detalles técnicos [22].

3.4.1.1.2. Por su nivel de ejecución

Los lenguajes de programación se dividen en dos categorías principales según su método de ejecución: los compilados y los interpretados. Sin embargo, algunos lenguajes pueden combinar características de ambos métodos. Los lenguajes compilados se refieren a aquellos que, antes de su ejecución, se convierten completamente a un lenguaje de máquina mediante un compilador. Este proceso traduce el código escrito por el programador a un formato binario entendible por la máquina, lo que facilita la comprensión para el sistema.

Por otro lado, los lenguajes interpretados, como su nombre lo sugiere, requieren de un intérprete para ejecutarse. Este intérprete lee y ejecuta el código línea por línea, lo que brinda flexibilidad y facilita la comprensión del programa. En este caso, la traducción ocurre durante la ejecución, lo que permite una mayor facilidad de entendimiento en comparación con los lenguajes compilados [22].

3.4.2. Uso de software de código abierto

El código abierto fue fundado en el año de 1985, por el personaje Richard M. Stallman. Entre los ejemplos más conocidos mencionamos TEX de Donald Knuth (1978) o la Berkeley Software Distribution (BSD) del Computer Systems Research Group de la Universidad de California (1977-1995) T. Gomez [23]. Se puede profundizar en la consideración de un software libre, en cuatro niveles de libertad, siendo el nivel 3 el más actualizado ha copias de versiones modificadas a terceros, lo que permite que toda la comunidad se beneficie de las mismas. Por otra parte, el código abierto, no se puede decir que es completamente libre o que se tenga acceso al código abierto y consecuentemente de aquello debe de cumplir con una normativa. Se utilizó el lenguaje de programación Python ya que es considerado como un lenguaje interpretado de propósito general siendo fácil de entender, es por esto decir que el lenguaje de programación Python se encuentran dentro del top 10 bien dicho por Stack Overflow Trends. Además, este lenguaje es dinámico siendo muy útil para crear un prototipo, y ampliamente es utilizado en proyectos de “Machine Learning”. La librería que se utiliza para

aplicar los modelos de predicción que están dentro de Machine Learning es “SciPy” la cual contiene ciertas librerías:

Numpy: esta librería permite realizar los trabajos o problemas de forma eficiente con datos en vectores y matrices.

Matplotlib: Esta librería permite realizar gráficos dinámicos.

Pandas: Esta librería ofrece herramientas con estructuras de acceso, organización y análisis de datos.

Otras de las librerías en Machine Learning en Python es “scikit-learn” de tal forma que esta brinda algoritmos para clasificación, regresión y clustering.

El concepto de software de código abierto fue presentado en 1998 por Eric Raymond y Bruce Perens. Este tipo de software se rige por una serie de criterios fundamentales, que incluyen la capacidad de ser modificado y distribuido libremente, la disponibilidad del código fuente sin restricciones y la libertad para redistribuir modificaciones, entre otros aspectos. Esencialmente, el software de código abierto se caracteriza por ser accesible para la sociedad, ofreciendo una serie de ventajas significativas [24].

3.4.3. Ventajas del software de código abierto

Acceso al código fuente: Prácticamente el código fuente es accesible para cualquier persona por lo que cualquier persona puede entender y conocer cómo funciona el software.

Flexibilidad y personalización: Los usuarios tienen la facilidad de modificar el software permitiendo así ajustarse a las necesidades que requieren.

Costo: Es menos costoso ya que no se necesita de licencias.

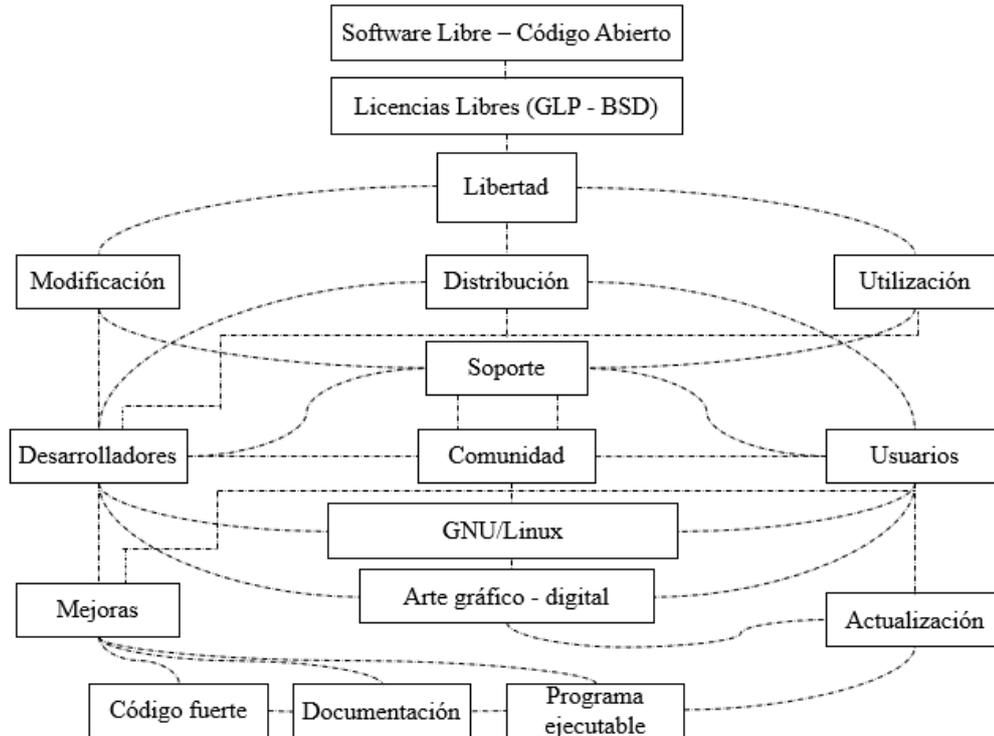


Figura 3.8. Interrelación entre Software libre y Código abierto [24].

3.4.4. Python

Python se destaca como un lenguaje de alto nivel en la programación. A diferencia de los lenguajes de bajo nivel, Python no es directamente comprensible para la máquina. En su lugar, requiere un paso intermedio: un compilador. Este compilador actúa como un traductor, convirtiendo el código escrito en Python a un formato que la máquina pueda entender y ejecutar. En el caso específico de Python, este proceso de compilación genera lo que se conoce como 'bytecode'.

El bytecode, generado a partir del código fuente de Python, es una forma intermedia de código que aún no es el lenguaje de máquina directamente ejecutable, pero tampoco es el código fuente original. En esencia, el bytecode sirve como una especie de traducción intermedia, facilitando la ejecución eficiente del código Python en una variedad de plataformas y sistemas [25].

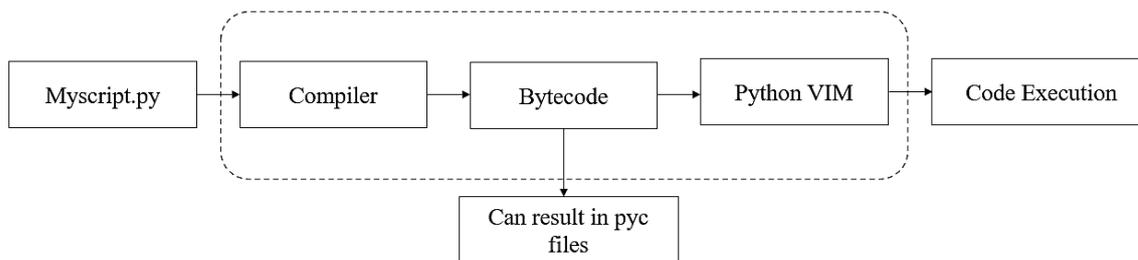


Figura 3.9. Ejecución de un programa en Python [25].

3.4.5. Características del Lenguaje Python

3.4.5.1. Python es un lenguaje interpretado y multiplataforma.

Esta particularidad implica que el código se interprete y ejecute línea por línea, lo que facilita su procesamiento para la máquina. Es por ello que Python es compatible y puede ejecutarse en distintos sistemas operativos.

3.4.5.2. Tiene un lenguaje multiparadigma.

Tiene la facilidad de combinar múltiples paradigmas entre ellos tenemos programación imperativa, declarativa, orientada a objetos entre otros, lo que permite a los usuarios combinar elementos de todos los paradigmas en un solo programa.

3.4.5.3. Lenguaje de propósito general.

En términos sencillos, un lenguaje de programación de propósito general se destaca por su versatilidad y su habilidad para integrarse fácilmente con otros lenguajes. Ejemplos claros de este tipo de lenguajes incluyen Python, Java, C++, entre otros. Estos lenguajes son reconocidos por su capacidad de adaptarse a una amplia gama de aplicaciones y entornos de desarrollo [25].

3.4.6. Librerías de Python

3.4.6.1. Matplotlib

Esta librería es ampliamente empleada para generar diversos tipos de gráficos, desde histogramas hasta espectros de potencia y series temporales, con la ventaja de requerir solo una

mínima cantidad de código. Ofrece una solución práctica y versátil para la creación de múltiples representaciones visuales [26].

3.4.6.2.Numpy

Proporciona una estructura de datos universal mediante matrices o vectores, permitiendo el almacenamiento eficiente de grandes volúmenes de información.

3.4.6.3.Scipy

Esta biblioteca se enfoca en el cálculo numérico, ofreciendo una amplia gama de herramientas y funcionalidades diseñadas para realizar operaciones matemáticas avanzadas de manera eficiente y precisa.

3.4.6.4.Pandas

Esta biblioteca se emplea para la manipulación eficiente de grandes conjuntos de datos en áreas tan diversas como finanzas, estadística e ingeniería.

3.4.6.5.Scikit-Learn

Esta biblioteca se emplea ampliamente en la clasificación y regresión, ofreciendo herramientas como máquinas de vectores de soporte, árboles de decisión, regresión lineal y polinómica, técnicas de clustering, reducción de dimensiones, y una variedad de otras aplicaciones especializadas en el análisis de datos.

3.4.6.6.keras

Esta biblioteca permite saber si una idea proporcionada va tener resultados satisfactorios.

3.4.6.7.PyTorch

Esta librería se usa para trabajar en predicción con redes neuronales [26].

3.4.6.8.sklearn.metrics

El módulo sklearn.metrics, parte de la biblioteca Scikit-learn, ofrece una amplia gama de funciones para evaluar tanto modelos de aprendizaje automático de clasificación como de regresión. Entre las métricas más comúnmente utilizadas en sklearn.metrics se incluyen la precisión, la puntuación y el error medio cuadrático, entre otras. Esta herramienta es fundamental para analizar el rendimiento de los modelos de aprendizaje automático. Al emplear las métricas adecuadas, es posible comprender con mayor profundidad el desempeño del modelo y tomar decisiones informadas sobre cómo mejorarlo [27].

3.4.6.9.Plotly Express

Plotly Express proporciona funciones para visualizar una variedad de tipos de datos. Plotly va más allá de simplemente crear gráficos atractivos; se trata de desbloquear el potencial completo de tus datos y descubrir información que de otra manera podría permanecer oculta. Funciona como un puente entre el lenguaje de los números y el lenguaje de las narrativas, permitiendo a personas y organizaciones tomar decisiones fundamentadas y efectuar cambios significativos [28].



Figura 3.10. Uso de Plotly en gráficos visualmente atractivos [28].

CAPÍTULO IV

4.METODOLOGÍA

4.1.TIPOS DE INVESTIGACIÓN

4.1.1.Investigación descriptiva

La investigación descriptiva se enfoca en analizar detalladamente un objeto o fenómeno, desglosándolo en elementos básicos y explorando sus características específicas. Esto busca entender su estructura y comportamiento. La calidad de los datos es crucial para garantizar resultados sólidos y creíbles, por lo que se requiere recolectarlos con precisión y autenticidad. La observación es un método clave en este proceso, dividido en enfoques cualitativos y cuantitativos, ambos ofreciendo perspectivas detalladas del objeto de estudio. [29].

4.1.2.Investigación Experimental

La investigación experimental tiene como objetivo identificar las relaciones de causa y efecto en un fenómeno. Cuando un investigador logra modificar una variable con certeza, se considera que la investigación experimental ha tenido éxito. Este enfoque se emplea tanto en ciencias sociales como naturales, así como en diversos campos de estudio. Es crucial destacar que no todos los fenómenos pueden ser abordados experimentalmente, por lo que existen otros métodos y enfoques, como los estudios correlacionales y descriptivos, que también desempeñan un papel importante en la investigación científica [30].

4.1.3.Investigación Cuantitativa

La investigación cuantitativa es considerada como un tipo de metodología que recoge datos de estudios principalmente científicos donde utiliza herramientas de análisis matemático y estadístico para describir, explicar y predecir fenómenos mediante datos numéricos. Este tipo de investigación utiliza la metodología estructurada, como encuestas, experimentos y análisis estadísticos, para medir variables de interés de manera objetiva, donde este enfoque busca generalizar los resultados a poblaciones más amplias y se basa en la objetividad y replicabilidad [31].

4.1.4. Ubicación del Caso de Estudio de la Mini Central Hidroeléctrica Catazacón

La Mini Central Hidroeléctrica de Catazacón, ubicada en el cantón Pangua de la provincia de Cotopaxi, representa un hito en la historia energética de la región. Inaugurada en 1991, esta infraestructura hidroeléctrica se ha convertido en un pilar fundamental para el suministro de energía eléctrica en la zona. Gestionada por la Empresa Eléctrica Cotopaxi ELEPCO S.A., la central desempeña un papel crucial en la generación y distribución de electricidad, contribuyendo significativamente al desarrollo económico y social de la comunidad local.



Figura 4.1. Mini Central Hidroeléctrica Catazacón.

4.2. TÉCNICA E INSTRUMENTOS

4.2.1. Programación

Se desarrollarán algoritmos para los modelos de pronóstico, que incluyen el modelo de regresión lineal, el modelo de unidades recurrentes cerradas y el modelo autorregresivo integrado de media móvil. Se utilizarán datos depurados almacenados en una base de datos general, que contendrá el historial de la potencia total de la central de generación. El objetivo es determinar el modelo que mejor se ajuste a los datos reales, evaluando su desempeño mediante diversas métricas, tales como el error medio absoluto (MAE), el error absoluto medio porcentual (MAPE), el error medio cuadrático (MSE) y el coeficiente de determinación R^2 .

4.2.2.Simulación

A través del entorno de programación “colab”, se llevará a cabo la simulación que facilitará la configuración de parámetros asociados a diversos modelos de predicción. Se utilizarán datos históricos correspondientes a cinco años de la potencia total de la mini central hidroeléctrica Catazacón con el fin de realizar pronósticos específicos para fechas determinadas.

4.3.MÉTODOLOGÍA GENERAL

4.3.1.Caso de estudio en la Mini Central Hidroeléctrica Catazacón

Para los métodos de pronóstico mediante aprendizaje automático de la variable de Potencia Activa generada por la Mini Central Hidroeléctrica Catazacón, se empleó el lenguaje de programación Python. Los datos históricos fueron obtenidos de la Empresa Eléctrica Cotopaxi ELEPCO S.A., recopilados en una base de datos. Para este estudio, se consideraron datos desde el 1 de enero de 2018 hasta el 31 de diciembre de 2022, con un intervalo de muestreo de 30 minutos, lo que resultó en un total de 87647 puntos de datos de Potencia Activa generada. Del conjunto total, el 80% se destinó al entrenamiento del modelo, mientras que el 20% se reservó para la prueba. La validación se llevó a cabo en tres niveles: un día específico (1 de enero del 2022), una semana completa (1-7 de enero del 2022) y un mes completo (enero del 2022).

En cuanto a los métodos empleados para el pronóstico de la variable de potencia activa de generación, se exploraron tres enfoques: Regresión Lineal Simple, Unidades Recurrentes Cerradas (GRU) y Modelo de Redes Neuronales (LSTM). El modelo de Redes Neuronales (LSTM) en este estudio se emplea exclusivamente con el propósito de compararlo con los modelos de Regresión Lineal Simple y en Unidades Recurrentes Cerradas GRU.

Estos modelos serán evaluados con el propósito de determinar cuál se ajustaba mejor a la realidad. Además, se aplican diversas métricas de evaluación, tales como el Error Medio Cuadrático (MSE), el Error de Porcentaje Medio Absoluto (MAPE) y el Error Medio Absoluto (MAE), en todo el rango de validación de los modelos. Para garantizar la adecuación del algoritmo de pronóstico, se utilizará la métrica de coeficiente de determinación R^2 .



Figura 4.2. Diagrama de bloque para el proceso de investigación.

En el contexto de este proyecto de investigación, se emplea la metodología delineada en el diagrama de bloques presentado en la Figura 4.2, se lleva a cabo una revisión exhaustiva de diversas fuentes bibliográficas que contribuyan al conocimiento sobre técnicas de aprendizaje automático y temáticas asociadas.

4.3.2. Comparación de los Modelos de Aprendizaje Automático, Lineal, GRU, LSTM

Tabla 4.1. Comparación de los Modelos Utilizados de Predicción.

Modelo	Ventajas	Desventajas	Utilización
LINEAL	Simplicidad y facilidad de interpretación	Limitación para modelar relaciones no lineales complejas	Interpretación simple, relaciones lineales aproximadas, poco hiperparámetros
GRU	Manejo de Secuencias Temporales, Memoria de Corto y Largo Plazo, Parámetros Ajustables	Sensibilidad a la Elección de Hiperparámetros, Complejidad de Entrenamiento, Interpretación Compleja	Capturar patrones temporales y dependencias a largo plazo en series temporales

Modelo	Ventajas	Desventajas	Utilización
LSTM	Maneja secuencias de longitud variable	Problemas con la desaparición/explotación del gradiente.	Secuencias cortas y datos de secuencia.

En la figura adjunta se presenta el diagrama de bloques que ilustra el proceso de implementación de los Modelos de Aprendizaje Automático. Estos modelos tienen como objetivo principal pronosticar la Potencia Activa generada por la Mini Central Hidroeléctrica Catazacón.

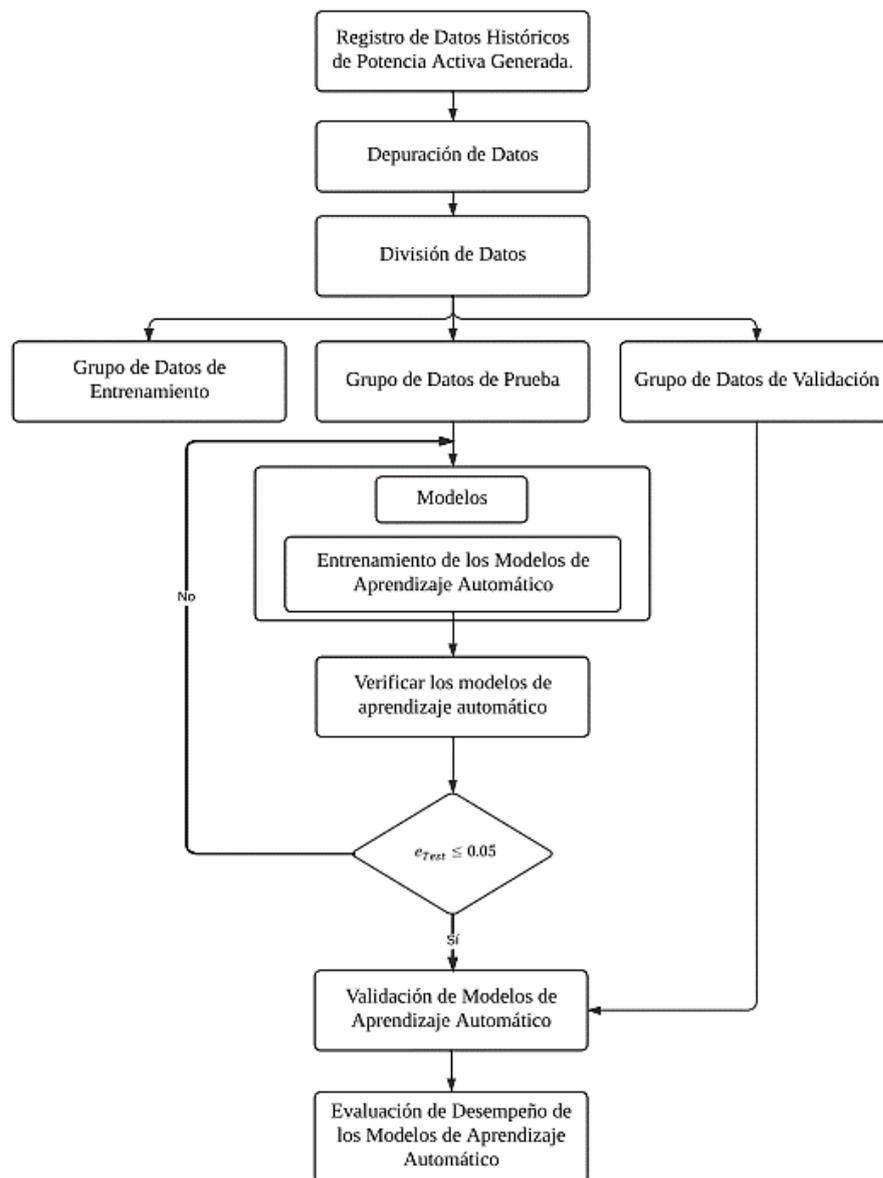


Figura 4.3. Diagrama de Bloques para los Modelos de Aprendizaje Automático.

4.3.3. Procesamiento inicial de la información

Se presenta un modelo de la base de datos históricos de la Mini Central Hidroeléctrica Catazacón, cuya información corresponde al mes de enero del año 2018, recalcando que los datos para la elaboración de la data final son de cinco años, es decir desde el 2018 hasta el 2022. Los datos están registrados en intervalos de 30 minutos a lo largo de las 24 horas del día, correspondientes a una fecha específica. Incluyen parámetros eléctricos de cada unidad de generación, así como la potencia activa total de la Central Hidroeléctrica.

EMPRESA ELECTRICA PROVINCIAL COTOPAXI S.A.																												
REPORTE DIARIO DE GENERACION																												
CENTRAL HIDROELECTRICA CATAZACON																												
																									FECHA:		Lunes	
																									1/1/2018			
GENERADOR No.1													GENERADOR No.2													TOTAL		
HORAS	I (A)	P (KW)	E (KWH)	E (KWH)	I (A)	T (C)	COSE Φ	Q (KVAR)	I (A)	P (KW)	E (KWH)	E (KWH)	I (A)	T (C)	COSE Φ	Q (KVAR)	V (V)	E (KWH)	PT (KW)	QT (KVAR)								
0:00	400	141.504	5.274.526,0	0	53	0,90	68.534	540	279.488	5.551.321,0	279.488	50	0,90	136.362	440	420.592	420.992	203.896	203.896									
1:00	400	141.504	5.274.727,0	141.504	53	0,90	68.874	540	285.824	5.552.124,0	285.824	50	0,90	138.431	440	428.032	428.032	207.305	207.305									
2:00	400	142.208	5.274.929,0	142.208	53	0,90	69.215	540	287.232	5.552.532,0	287.232	50	0,90	139.113	440	430.144	430.144	208.328	208.328									
3:00	400	142.208	5.275.132,0	142.208	53	0,90	69.215	540	287.232	5.552.824,0	287.232	50	0,90	140.818	440	432.960	432.960	209.692	209.692									
4:00	400	142.208	5.275.334,0	142.208	53	0,90	68.874	540	284.416	5.553.349,0	284.416	50	0,90	137.749	440	426.624	426.624	206.623	206.623									
5:00	400	142.208	5.275.536,0	142.208	53	0,90	68.874	540	287.936	5.553.939,0	287.936	50	0,90	139.454	440	430.144	430.144	208.328	208.328									
6:00	400	142.208	5.275.738,0	142.208	53	0,90	68.874	540	287.936	5.553.758,0	287.936	50	0,90	139.454	440	430.144	430.144	208.328	208.328									
7:00	400	104.192	5.275.886,0	104.192	53	0,90	50.462	540	127.424	5.553.939,0	127.424	50	0,90	61.714	440	231.616	231.616	112.177	112.177									
8:00	400	100.872	5.276.029,0	100.872	53	0,90	48.758	520	152.768	5.554.156,0	152.768	50	0,90	73.969	440	253.440	253.440	122.747	122.747									
9:00	400	109.824	5.276.185,0	109.824	53	0,90	53.190	520	144.320	5.554.381,0	144.320	50	0,90	89.897	440	254.144	254.144	123.088	123.088									
10:00	400	109.824	5.276.341,0	109.824	53	0,90	53.190	520	144.320	5.554.606,0	144.320	50	0,88	77.896	440	254.144	254.144	131.096	131.096									
11:00	400	83.072	5.276.497,0	83.072	53	0,90	40.234	520	219.648	5.554.831,0	219.648	50	0,88	118.553	440	302.720	302.720	158.787	158.787									
12:00	400	83.072	5.276.653,0	83.072	53	0,90	40.234	520	219.648	5.555.056,0	219.648	50	0,88	118.553	440	302.720	302.720	158.787	158.787									
13:00	400	147.136	5.276.809,0	147.136	53	0,90	71.261	520	219.648	5.555.281,0	219.648	50	0,88	118.553	440	302.720	302.720	158.787	158.787									
14:00	400	147.136	5.276.965,0	147.136	53	0,90	71.261	520	219.648	5.555.506,0	219.648	50	0,88	118.553	440	302.720	302.720	158.787	158.787									
15:00	400	127.424	5.277.121,0	127.424	53	0,90	61.714	520	97.856	5.555.731,0	97.856	50	0,88	52.817	440	225.280	225.280	114.531	114.531									
16:00	400	127.424	5.277.277,0	127.424	53	0,90	61.714	520	97.856	5.555.956,0	97.856	50	0,88	52.817	440	225.280	225.280	114.531	114.531									
17:00	400	113.344	5.277.433,0	113.344	53	0,90	54.895	520	292.160	5.556.181,0	292.160	50	0,88	157.691	440	405.504	405.504	212.596	212.596									
18:00	400	113.344	5.277.589,0	113.344	53	0,90	54.895	520	292.160	5.556.406,0	292.160	50	0,88	157.691	440	405.504	405.504	212.596	212.596									
19:00	400	166.848	5.277.745,0	166.848	53	0,90	80.808	520	271.744	5.556.631,0	271.744	50	0,88	146.672	440	438.592	438.592	227.480	227.480									
20:00	400	166.848	5.277.901,0	166.848	53	0,90	80.808	520	271.744	5.556.856,0	271.744	50	0,88	146.672	440	438.592	438.592	227.480	227.480									
21:00	400	107.712	5.278.057,0	107.712	53	0,90	52.167	520	252.736	5.557.081,0	252.736	50	0,88	136.412	440	360.448	360.448	188.580	188.580									

Figura 4.4. Base de datos de los parámetros eléctricos del mes de enero del año 2018.

4.3.4. Selección de variables

Después de completar el procesamiento de la información, se identificaron las variables críticas, es decir, los datos que aportan información relevante para las fases subsiguientes de entrenamiento y prueba en los modelos de aprendizaje automático. En este contexto, se optó por seleccionar la variable de tiempo y la potencia activa, ya que el objetivo principal de la investigación es pronosticar la potencia activa basándose en datos históricos. Aunque existen otras variables que podrían contribuir al pronóstico de la potencia activa, esta investigación se centra exclusivamente en la variable de potencia activa total de la Mini Central Hidroeléctrica Catazacón.

La Tabla 4.2, muestra las variables seleccionadas que se incluirán posteriormente en la base de datos depurada.

Tabla 4.2. Variables seleccionadas

Variables	Unidad de medida
Año, mes, día, hora	Tiempo [h]
Potencia Total	kW

4.3.5. Depuración de la data de la Mini Central Hidroeléctrica Catazacón

Este proceso resulta fundamental para crear una base de datos actualizada que se utilizará en las fases posteriores del desarrollo y evaluación del sistema. La disponibilidad de datos depurados y confiables son importantes para alcanzar resultados precisos y llevar a cabo una validación efectiva.

Es evidente realizar una depuración de los datos presentes en la Figura 4.4. Se ha elaborado un archivo de Excel, como se muestra en la Tabla 4.3, que contiene información detallada, incluyendo el año, mes, día, fecha, hora y Potencia Activa. Estos datos abarcan el periodo desde enero de 2018 hasta diciembre de 2022. Este conjunto de datos se utilizará como punto de partida en la creación de los modelos de aprendizaje automático en Python.

Tabla 4.3. Datos Depurados para los Modelos de Aprendizaje Automático.

AÑO	MES	DIA	FECHA	HORA	POTENCIA [kW]
2018	1	lunes	1	0:30	420,992
2018	1	lunes	1	1:00	420,992
2018	1	lunes	1	1:30	428,032
2018	1	lunes	1	2:00	428,032
2018	1	lunes	1	2:30	430,144
...
2018	1	lunes	1	21:30	392,128
2018	1	lunes	1	22:00	392,128
2018	1	lunes	1	22:30	377,344
2018	1	lunes	1	23:00	377,344
2018	1	lunes	1	23:30	382,272

4.3.6. Comportamiento de la Potencia Activa

En la Figura 4.5 se puede observar el comportamiento de la Potencia Activa Total de la Mini Central Hidroeléctrica Catazacón en la cual se destacan dos eventos considerables donde el primer evento empezó el 30 de Noviembre del 2018 donde la central Catazacón se encontró paralizada debido a trabajos que se realizan en bocatoma, desviación del río para el arreglo de las compuertas; una vez terminados los trabajos el 10 de Enero del 2019 empezó a operar, por lo que este primer evento si se considera para el estudio ya que es un evento normal de mantenimiento que se dio en la Mini Central Hidroeléctrica Catazacón. Mientras que el segundo evento se registró en el 2022 donde ocurrió una interrupción total en la Central, originada por un incidente ocurrido en febrero de dicho año. En este evento, la central hidroeléctrica Catazacón experimentó una detención en su generación debido a la ruptura de los canales que la alimentaban, consecuencia directa del desbordamiento del río Angamarca. Lo cual se atribuye a un incidente natural, el cual no puede ser evaluado en el contexto de la aplicación de los modelos predictivos debido a su naturaleza impredecible que puede ocurrir en cualquier momento.

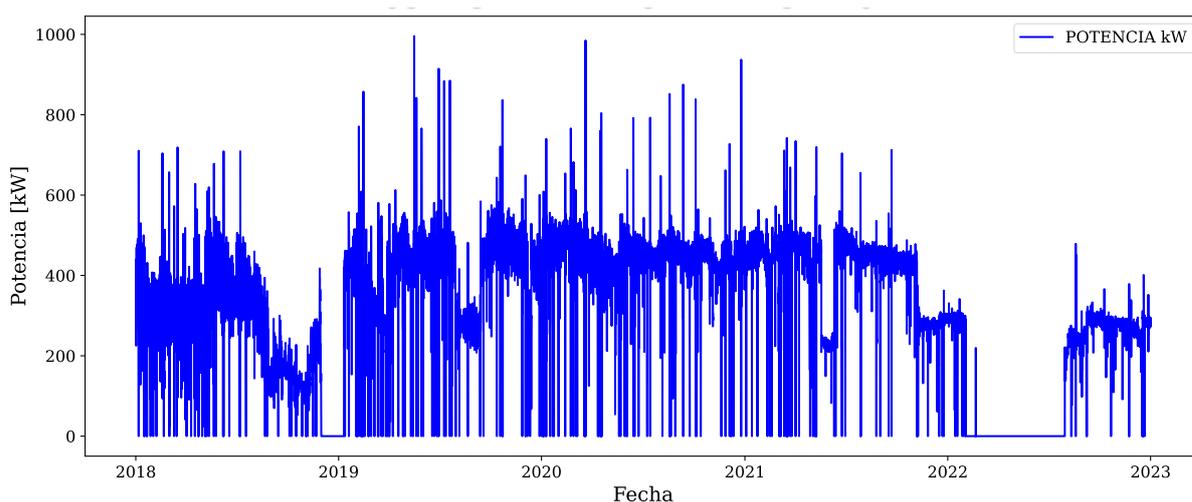


Figura 4.5. Comportamiento de la Potencia Activa de la Central Hidroeléctrica Catazacón.

4.3.7. Metodología para el análisis de datos

Para la construcción de la base de datos definitiva, se tuvieron en cuenta diversos parámetros esenciales, entre los cuales se incluyen el año, mes, día, hora y potencia activa total. Es importante destacar que la potencia total representa la suma de las dos unidades de generación.

Se recopilarán datos de un amplio período de 5 años, abarcando desde el año 2018 hasta el año 2022, lo que permitirá obtener una visión completa y detallada de los patrones y tendencias en la producción de energía de la Mini Central Hidroeléctrica Catazacón.

4.3.7.1. Comportamiento de la Potencia en la Mini Central Hidroeléctrica Catazacón

Se lleva a cabo el análisis diario de datos, como se ilustra en la Figura 4.6, permitiendo visualizar el comportamiento de la potencia total. Es relevante destacar que los datos se registran cada 30 minutos a lo largo del día 01 de enero del 2022.

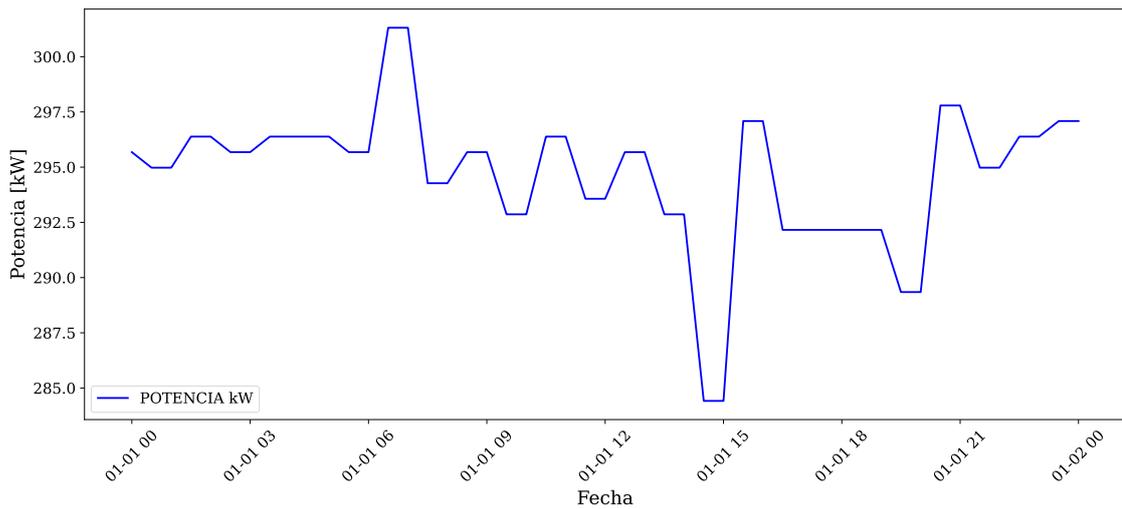


Figura 4.6. Comportamiento diario real de la potencia activa del día 01 de enero del 2022

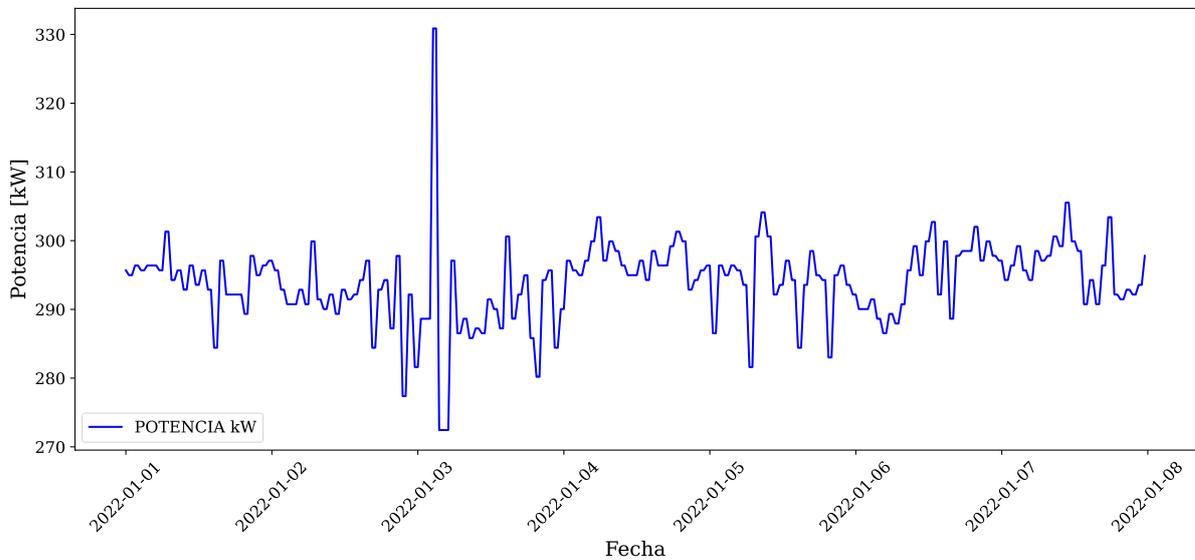


Figura 4.7. Comportamiento semanal real de la potencia activa del 01 al 07 enero del 2022

El análisis de datos de manera semanal se muestra en la Figura 4.7, proporciona una visión más completa de las variaciones y tendencias en la generación de potencia. Este enfoque detallado permite identificar patrones específicos y evaluar la estabilidad del sistema de generación en diferentes momentos de la primera semana de enero del 2022.

Mensualmente, se pretende analizar y comparar la potencia total activa recibida durante el año 2022. Estas representaciones gráficas permitirán identificar claramente los períodos con niveles más altos y más bajos de potencia total, como se ilustra en la Figura 4.8.

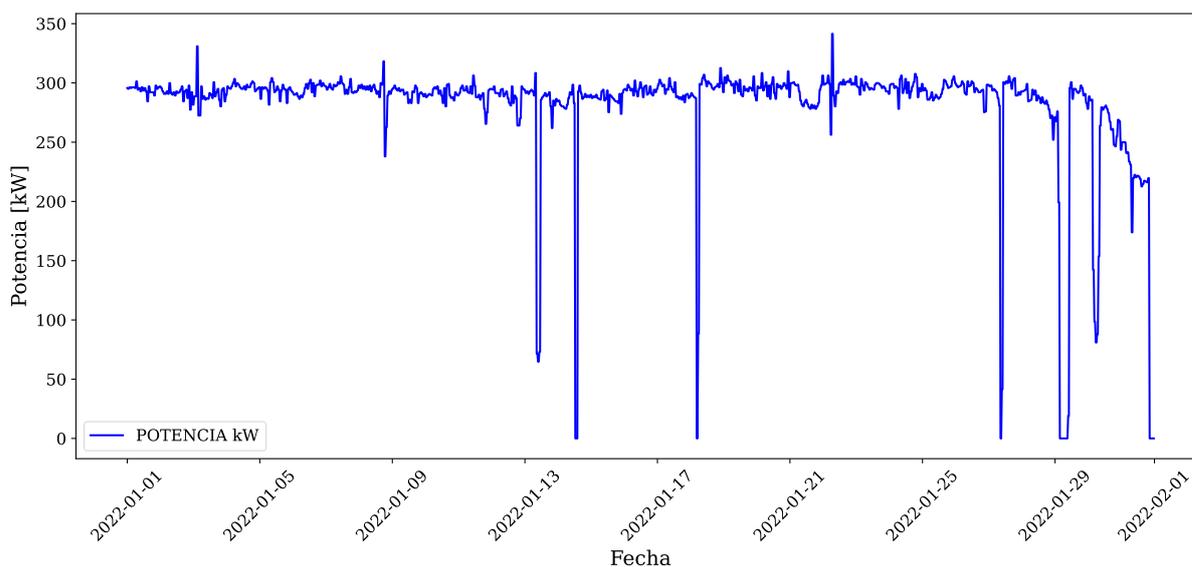


Figura 4.8. Comportamiento mensual real de la potencia activa del mes de enero 2022

4.3.8. Validación de los datos

El periodo de datos que se va a utilizar en los modelos de Aprendizaje Automático comprende desde el año 2018 hasta el 2022, los mismos que serán clasificados en datos de prueba y datos de entrenamiento, ya que esta práctica es esencial para evaluar la capacidad de generalización del modelo, por tanto, se implementa una división estratégica que consiste en asignar el 80% de los datos al conjunto de entrenamiento y el 20% restante al conjunto de prueba.

El conjunto de entrenamiento, compuesto por el 80%, se utiliza para enseñar al modelo patrones y tendencias presentes en los datos. Mientras que, para el conjunto de prueba comprende el 20% de datos el cual se emplea para evaluar la capacidad del modelo de predecir con precisión datos no vistos durante el entrenamiento.

4.3.9.Implementación en Python

Para la implementación de los modelos de aprendizaje en el software Python, debe seguir cada modelo de diagrama de flujo correspondiente, así como se muestra en la Figura 4.9 el cual representa la programación del Modelo de Regresión Lineal Simple, la Figura 4.10 representa el programa para el Modelo de Unidades Recurrentes Cerradas GRU, mientras que en la Figura 4.11 representa el diagrama de flujo para el Modelo de Redes Neuronales LSTM.

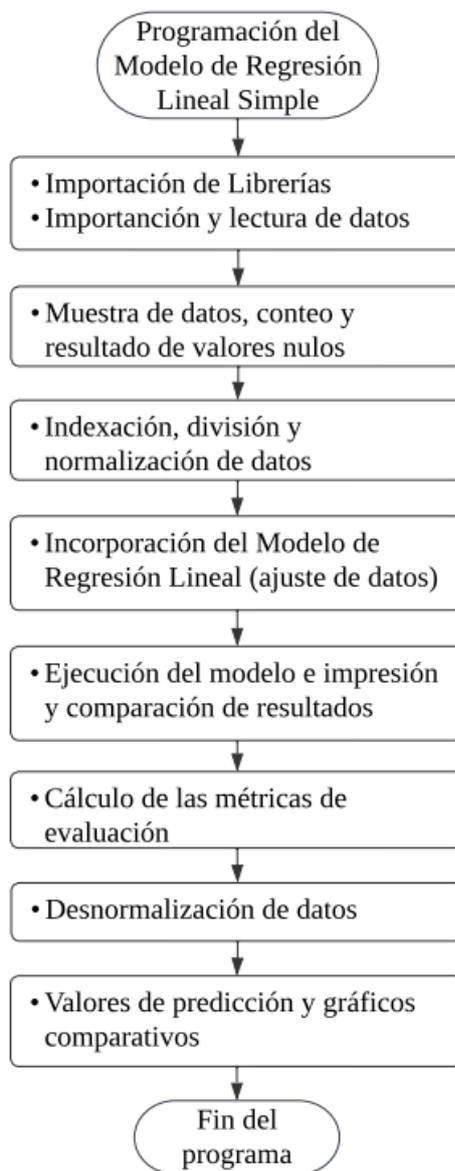


Figura 4.9. Diagrama de Flujo del Modelo de Regresión Lineal Simple

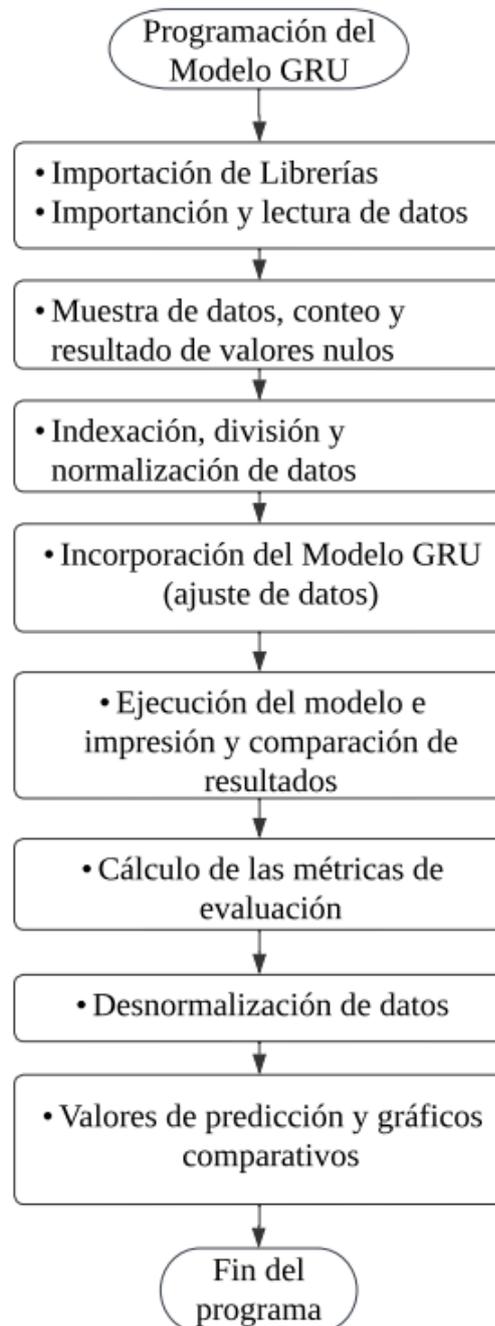


Figura 4.10. Diagrama de Flujo del Modelo Unidades Recurrentes Cerradas GRU

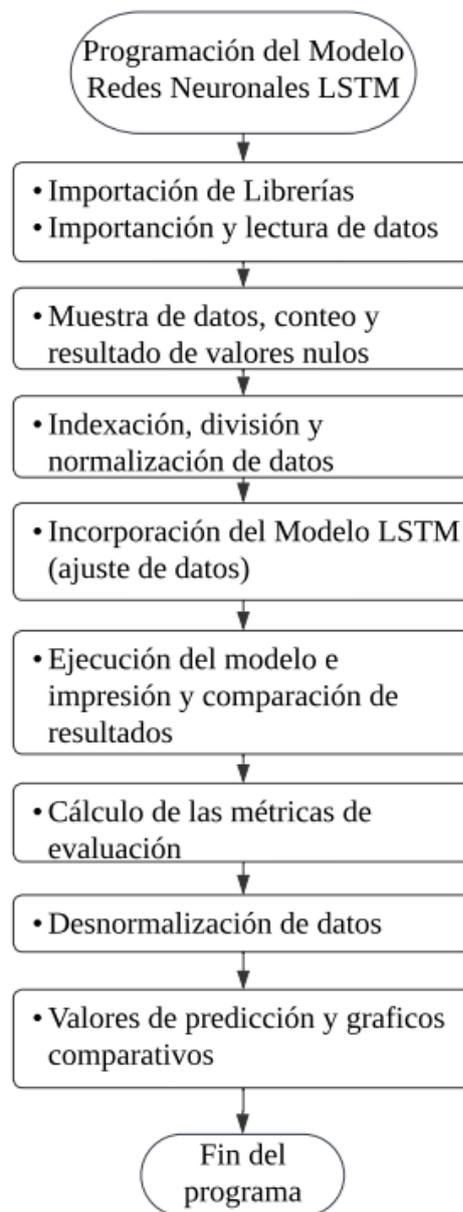


Figura 4.11. Diagrama de Flujo del Modelo de Redes Neuronales LSTM

4.3.9.1. Estructura del modelo de Regresión Lineal Simple

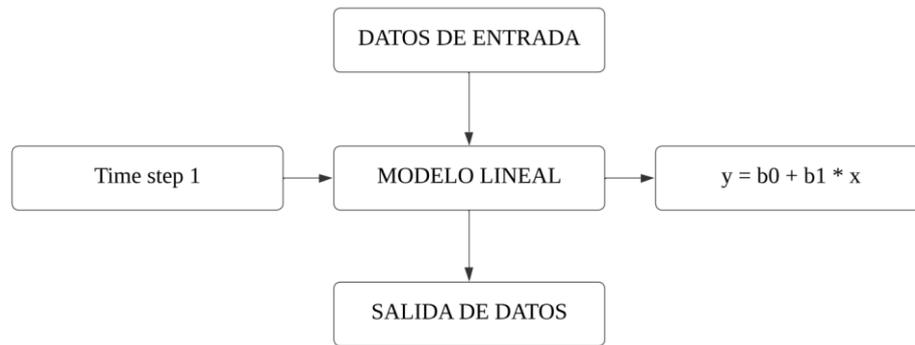


Figura 4.12. Estructura del modelo de Regresión Lineal

El modelo utiliza datos históricos de series temporales para entrenar un modelo de Regresión Lineal Simple que predice la Potencia Activa futura en función de valores pasados. Los parámetros del modelo (intercepto y coeficiente) se ajustan durante el entrenamiento para minimizar la diferencia entre las predicciones y los valores reales. Este enfoque proporciona una herramienta simple pero efectiva para realizar pronósticos en contextos de series temporales.

4.3.9.2. Estructura del modelo de Unidades Recurrentes Cerradas GRU

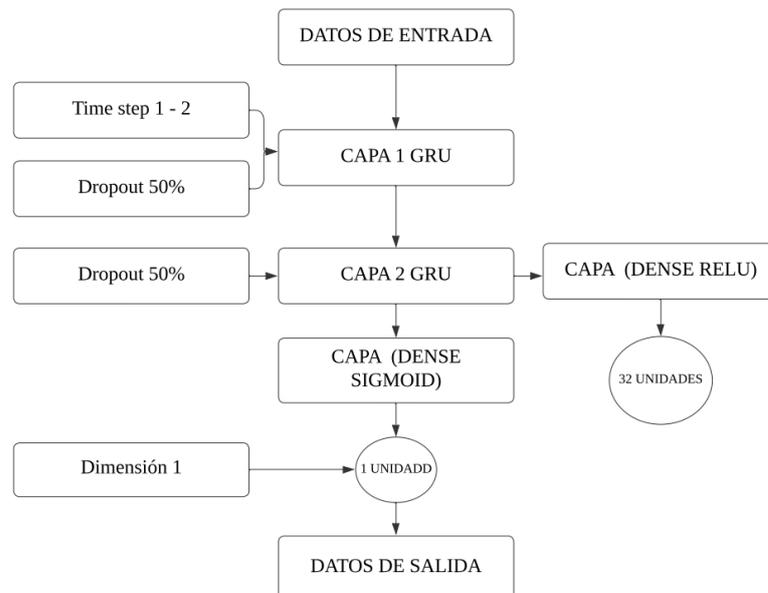


Figura 4.13. Estructura del modelo de Unidades Recurrentes Cerradas GRU

En la Figura 4.13, el Modelo de Unidades Recurrentes Cerradas (GRU) en el código presenta seis capas principales. Comienza con una capa GRU configurada para generar secuencias completas, lo que permite conexiones con capas recurrentes adicionales. Luego, se emplea una capa Dropout para regularizar la red, desactivando el 50% de las unidades y prevenir el sobreajuste. A continuación, se introduce otra capa GRU para procesar la salida de la capa anterior. Posteriormente, se utiliza una capa Dense con activación ReLU para capturar características no lineales más complejas. Se añade otra capa Dropout para mitigar el sobreajuste, seguida por la capa de salida con una sola unidad y activación sigmoid, apropiada para problemas de regresión binaria. Este diseño secuencial permite capturar patrones temporales y realizar predicciones con eficacia en la tarea específica.

4.3.9.3. Estructura del Modelo de Redes Neuronales LSTM

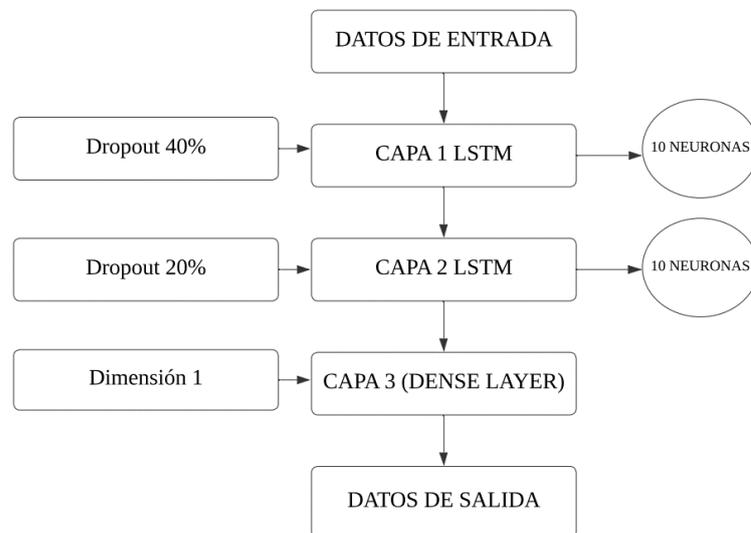


Figura 4.14. Estructura del Modelo de la Red Neuronal LSTM

En la Figura 4.14, la primera capa LSTM conserva la dimensión temporal de los datos y aplica un dropout del 40% para evitar el sobreajuste. La segunda capa LSTM, con el mismo número de neuronas, devuelve solo el último resultado de la secuencia y se aplica un dropout del 20%. Finalmente, se utiliza una capa dense para reducir la salida a una sola dimensión, ideal para la predicción de una variable de Potencia Activa. Este enfoque estructurado permite un procesamiento efectivo de la información en la red neuronal.

4.3.9.4. Utilización de librerías en los Modelos de Aprendizaje Automático Lineal Simple, GRU, LSTM

Se importaron diversas librerías utilizadas en la programación para el entrenamiento de cada modelo de pronóstico, así como muestra en la Figura 4.15 donde se representan las librerías a utilizar para la programación del Modelo Lineal Simple las cuales son: pandas, numpy, scikit-learn, matplotlib y sklearn.metrics. Con la ayuda de última librería mencionada, se puede determinar el Error Medio Cuadrático, el Error Medio Absoluto, el Error De Porcentaje Medio Absoluto y el Coeficiente de Determinación, que posteriormente se puede evaluar al modelo.

Por otro lado, se muestra en la Figura 4.16, la representación de las librerías a utilizar para la programación del Modelo de Unidades Recurrentes Cerradas (GRU) las cuales son: pandas, numpy, scikit-learn, matplotlib y así también como lo es la librería keras, la cual es una biblioteca de nivel superior en el ámbito de modelado que proporciona elementos predefinidos para la construcción de modelos de aprendizaje profundo. Keras simplifica significativamente el proceso de desarrollo al ofrecer componentes predefinidos y abstractos que permiten a los desarrolladores concentrarse en la arquitectura del modelo en lugar de lidiar con detalles técnicos y operaciones de bajo nivel.

En la Figura 4.17, se presentan las bibliotecas específicas que desempeñarán un papel fundamental en la implementación del Modelo Redes Neuronales LSTM. Estas bibliotecas son esenciales para aprovechar al máximo las capacidades de LSTM en el análisis de series temporales.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score, mean_absolute_error
from sklearn.preprocessing import MinMaxScaler
from datetime import datetime
```

Figura 4.15. Librerías de Python para la programación del Modelo de Regresión Lineal Simple

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
import keras
from keras.models import Sequential
from keras.layers import Dense, Activation, Dropout
from sklearn.metrics import mean_squared_error, mean_absolute_error, r2_score
import plotly.express as px
import os
from datetime import datetime

```

Figura 4.16 . Librerías de Python para la programación del Modelo Unidades Recurrentes Cerradas GRU

```

import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
import keras
from keras.models import Sequential
from keras.layers import Dense, Activation, Dropout
from sklearn.metrics import mean_squared_error

# No presentar advertencias
import warnings
# No presentar advertencias
import warnings
warnings.filterwarnings("ignore")

```

Figura 4.17. Librerías de Python para la programación del Modelo Redes Neuronales LSTM

4.3.9.5.Lectura de datos generales

En cuanto a la adquisición de información global, se lleva a cabo la transferencia de datos desde Excel a Python, abarcando un período que va desde el año 2018 hasta el 2022. Estos datos se emplearán en la programación, como se ilustra en la Figura 4.18. La base de datos importada comprende detalles específicos, incluyendo año, mes, día y potencia activa total. Estos elementos fueron previamente seleccionados en Excel y posteriormente importados a Python para su utilización.

```
# Importar datos
data = pd.read_excel('RNNMODEL.xlsx', sheet_name='PRUEBA1')
print(data)
```

	AÑO	MES	DIA	FECHA	HORA	POTENCIA
0	2018	1	lunes	1	00:30:00	420.992
1	2018	1	lunes	1	01:00:00	420.992
2	2018	1	lunes	1	01:30:00	428.032
3	2018	1	lunes	1	02:00:00	428.032
4	2018	1	lunes	1	02:30:00	430.144
...
87642	2022	12	sábado	31	21:30:00	288.640
87643	2022	12	sábado	31	22:00:00	288.640
87644	2022	12	sábado	31	22:30:00	285.824
87645	2022	12	sábado	31	23:00:00	285.824
87646	2022	12	sábado	31	23:30:00	287.936

[87647 rows x 6 columns]

Figura 4.18. Lectura de datos depurados para la ejecución de los Modelos de Aprendizaje Automático

4.3.9.6. Indexación de los datos depurados

La indexación de datos es una técnica utilizada para combinar información de distintas unidades temporales, como año, mes, día y hora, en un solo registro coherente: la fecha. Esta práctica se lleva a cabo con el propósito de optimizar la organización y el acceso a los datos almacenados. Al unificar estos datos en un formato uniforme, se facilita la búsqueda y la recuperación de información de manera más eficiente, lo que resulta especialmente útil en bases de datos grandes o sistemas que manejan gran cantidad de registros temporales. En la Figura 4.19 se representa la programación para realizar la indexación de los registros la cual se almacenará en un “DataFrame” con el fin de tener datos ordenados en una tabla de dos columnas.

```

# Convertir fechas

datos_concatenado = pd.DataFrame()
fechas_dt = []
PT_dt = []
for registro in data.values:
    fecha = "{}-{}-{} {}".format(registro[0], registro[1], registro[3], registro[4])
    fecha_dt = datetime.strptime(fecha, '%Y-%m-%d %H:%M:%S')
    fechas_dt.append(fecha_dt)
    PT_dt.append(registro[5])
datos_concatenado['DIA'] = fechas_dt
datos_concatenado['PT'] = PT_dt

#print(datos_concatenado)
# Establecer el índice como fechas
datos_concatenado.set_index('DIA', drop=True, inplace=True)
datos_concatenado.sort_index(inplace=True)

```

Figura 4.19. Lectura de datos depurados para la ejecución de los Modelos

4.3.9.7. División y normalización de datos

En la Figura 4.20, se representa la codificación para la división de datos, abarcando tanto el conjunto de prueba como el de entrenamiento. Se ha optado por una distribución de "80-20", lo que significa que el 80% del conjunto total de datos se asignará para llevar a cabo la fase de entrenamiento del modelo, mientras que el 20% restante se reservará específicamente para la evaluación del rendimiento del modelo. Este enfoque estratégico en la división de datos garantiza un equilibrio efectivo entre el proceso de aprendizaje y la evaluación, contribuyendo así a un desarrollo y evaluación robustos del modelo. Mientras que, en la normalización de datos, consiste en evitar diferencias en la escala de los datos proporcionados, de tal manera que podría afectar a la convergencia y estabilidad del modelo. Entonces al aplicar la normalización de datos, se consigue transformar los datos en un rango específico de 0 a 1.

```

# División de datos
division = int(0.2 * datos_concatenado.shape[0])
datos_test = datos_concatenado[-division:]
datos_train = datos_concatenado[:-division]

# Normalización de datos
sc = MinMaxScaler(feature_range=(0, 1))
datos_train_scaled = sc.fit_transform(datos_train)

```

Figura 4.20. División y normalización de datos

4.3.9.8. Agrupamiento de datos

El agrupamiento está relacionado con el preprocesamiento de datos para un Modelo de Aprendizaje Automático de datos consiste en la crear conjuntos de entrenamiento “X_train” y “Y_train” los cuales ayudan al modelo capturar patrones y características temporales para realizar predicciones precisas, así como se muestra en Figura 4.21.

```
# Entrenamiento del modelo
time_step = 1
X_train = []
Y_train = []
m = len(datos_train_scaled)

for i in range(time_step, m):
    X_train.append(datos_train_scaled[i - time_step:i, 0])
    Y_train.append(datos_train_scaled[i, 0])

X_train, Y_train = np.array(X_train), np.array(Y_train)
X_train = X_train.reshape(-1, time_step)
```

Figura 4.21. Agrupamiento de datos para el entrenamiento del Modelo de Aprendizaje Automático

4.3.9.9. Aplicación del Modelo de Regresión Lineal Simple

En la Figura 4.22, se muestra la programación del Modelo de Regresión Lineal Simple para realizar pronóstico, el cual se utiliza el comando “LinearRegression” que está dentro de la librería “scikit-learn” cuyo objetivo es predecir la Potencia Activa dado una fecha de inicio de predicción y una fecha final de predicción. La elección de este enfoque de Regresión Lineal Simple demuestra la intención de capturar y modelar las relaciones lineales subyacentes en los datos para realizar pronósticos precisos en el periodo especificado.

```
# Modelo Lineal
modelo = LinearRegression()
modelo.fit(X_train, Y_train)

# Datos de prueba
x_test = datos_test.values
x_test_scaled = sc.transform(x_test)
```

Figura 4.22. Modelo de Regresión Lineal Simple

4.3.9.9.1. Preparación de datos para predicción del Modelo de Regresión Lineal Simple

En la Figura 4.23, se refleja el código que se encarga de preparar los datos y hacer predicciones usando un modelo previamente entrenado de modo que crea un conjunto de características llamado `X_predicción` a partir de los datos de predicción que han sido escalados y remuestreados, usando un intervalo de tiempo definido (`time_step`). Luego, se ajusta el formato de `X_predicción` para que sea el adecuado. Después, utiliza el modelo entrenado para hacer predicciones, obteniendo valores predichos en una escala general (`y_prediccion_scaled`). Finalmente, revierte la escala de los valores predichos de vuelta a la escala original, lo que nos da las predicciones finales (`y_predicción`) en su escala original.

```
# Organizar datos de prueba
X_test = []
for i in range(time_step, len(x_test_scaled)):
    X_test.append(x_test_scaled[i - time_step:i, 0])

X_test = np.array(X_test)
X_test = X_test.reshape((X_test.shape[0], X_test.shape[1]))

# Realizar predicciones
prediccion = modelo.predict(X_test)
prediccion1 = sc.inverse_transform(prediccion.reshape(-1, 1))
```

Figura 4.23. Preparación de datos para predicción

4.3.9.9.2. Visualización para comparación de resultados del Modelo de Regresión Lineal Simple

Mediante la utilización del comando "DataFrame", se genera una tabla destinada a la recopilación de resultados comparativos. En esta tabla, se almacenan tanto los datos de predicción correspondientes a cada 30 minutos como los datos reales de Potencia Activa. Este enfoque posibilita la realización de comparaciones detalladas entre las predicciones realizadas y los valores reales observados. La creación de esta estructura tabular facilita la evaluación de la precisión del modelo, permitiendo una visualización efectiva de las diferencias y similitudes entre las predicciones y la realidad en cada punto horario, como se muestra en la Figura 4.24.

```

# DATAFRAME para los resultados
Preal_prediccion = pd.DataFrame({

    'Proyeccion_PT': y_prediccion.flatten()[:len(datos_prediccion_resampled)],
    'PotenciaReal': datos_prediccion_resampled.values.flatten()

}, index=datos_prediccion_resampled.index)

print(Preal_prediccion)
# Guardar el DataFrame en un archivo Excel
Preal_prediccion.to_excel('resultados_prediccion_MES_ML.xlsx', index=True)

```

Figura 4.24. Creación de tabla de resultados de potencia activa pronosticada y real para el Modelo de Regresión Lineal Simple

La Figura 4.25, presenta el código utilizado para calcular diversas métricas que evaluarán el desempeño del modelo de Regresión Lineal Simple. Este código se encarga de calcular el Error Medio Cuadrático, el Coeficiente de Determinación, el Error Medio Absoluto y el Error de Porcentaje Medio Absoluto.

```

# METRICAS DE EVALUACION DEL MODELO
mse = mean_squared_error(datos_prediccion_resampled.values.flatten(), y_prediccion.flatten())
print(f'Mean Squared Error (mse): {mse}')
r2 = r2_score(datos_prediccion_resampled.values.flatten(), y_prediccion.flatten())
print(f'R^2 Score: {r2}')
mae = mean_absolute_error(datos_prediccion_resampled.values.flatten(), y_prediccion.flatten())
print(f'Mean Absolute Error (mae): {mae}')
mape = np.mean(np.abs((datos_prediccion_resampled.values.flatten() - y_prediccion.flatten()) / datos_prediccion_resampled.values.flatten()) * 100)
#mape = np.mean(np.abs((datos_prediccion_resampled.values.flatten() - y_prediccion.flatten()) / (datos_prediccion_resampled.values.flatten() + 1e-8)))

print(f'Mean Absolute Percentage Error (mape): {mape}%')

```

Figura 4.25. Métricas de evaluación del Modelo Regresión Lineal Simple

4.3.9.10. Aplicación del Modelo de Unidades Recurrentes Cerradas GRU

En la Figura 4.26, se muestra la programación del Modelo de Unidades Recurrentes Cerradas para realizar pronóstico, el cual se crea una función “basic_GRU” el cual realiza y entrena un modelo de red neuronal recurrente con unidades de tipo GRU, utilizando la biblioteca Keras. En el modelo se implementó dos capas GRU, capas de Dropout para regularización, y una capa de salida con activación sigmoide para realizar tareas de regresión. El modelo se compila con un optimizador, se utiliza la función de pérdida de Error Medio Cuadrático (MSE), y se registra la métrica de Error Medio Absoluto (MAE). Luego, se entrena el modelo con datos de entrada y salida específicos durante un número definido de épocas. La función devuelve el modelo entrenado y el historial del entrenamiento.

```

from keras.layers import Dense, Activation, Dropout

def basic_GRU(optimizador, unidades, epocas):

    input_dim = (X_train.shape[1],X_train.shape[2])
    output_dim = 1
    num_unidades = unidades

    model = Sequential()

    model.add(GRU(units=num_unidades, return_sequences = True, input_shape=(input_dim)))
    # model.add(GRU(units=256, recurrent_dropout = 0.2) #, return_sequences = True
    model.add(Dropout(0.5))
    model.add(GRU(units=num_unidades, input_shape=(input_dim)))
    #model.add(Dense(128))
    model.add(Dense(32, activation='relu'))
    # model.add(Dense(32))
    model.add(Dropout(0.5))
    model.add(Dense(1,activation='sigmoid'))
    model.compile(optimizer= optimizador, loss='mse', metrics = ['mean_absolute_error'])
    history = model.fit(X_train, Y_train, epochs = epocas, batch_size = 128)
    return model, history

```

Figura 4.26. Modelo de Unidades Recurrentes Cerradas GRU

4.3.9.10.1. Preparación de datos para predicción del Modelo Unidades Recurrentes Cerradas GRU

La preparación de datos para la aplicación del Modelo de Unidades Recurrentes Cerradas (GRU) es un paso esencial en el desarrollo de sistemas de pronóstico de series temporales. En primer lugar, se realiza la normalización de los datos para ajustarlos a una escala que facilite el entrenamiento del modelo. Este proceso suele involucrar técnicas como Min-Max Scaling para garantizar que los datos estén en un rango adecuado. Posteriormente, se estructuran los datos en secuencias temporales, creando ventanas de entrada y salida que permiten a las GRU aprender patrones temporales complejos. La selección de la longitud de las secuencias temporales puede ser crucial y depende de la naturaleza de la serie temporal. Además, se lleva a cabo la división del conjunto de datos en conjuntos de entrenamiento y prueba para evaluar la capacidad del modelo en datos no vistos. Este enfoque sistemático en la preparación de datos es crucial para optimizar el rendimiento de las GRU al modelar dependencias temporales en series temporales, facilitando la creación de pronósticos precisos y generalizables.

```

def predDataTest(modelo):
    x_test = data_test.values
    x_test = sc.transform(x_test)
    ## x_test
    X_test = []
    for i in range(time_step, len(x_test)):
        X_test.append(x_test[i-time_step: i, 0])
    X_test= np.array(X_test)
    X_test = np.reshape(X_test, (X_test.shape[0], X_test.shape[1], 1))
    X_test.shape
    prediccion = modelo.predict(X_test)
    prediccion = sc.inverse_transform(prediccion)
    return prediccion

```

Figura 4.27. Preparación de datos para predicción

El modelo se configura mediante la construcción de una red neuronal secuencial con una capa GRU y una capa de Dropout para mejorar la generalización del modelo. El optimizador “adam” se elige para entrenar la red, y se realiza el entrenamiento durante 10 épocas. Además, se destaca que, según los resultados obtenidos, el optimizador “rmsprop” ha demostrado proporcionar los mejores resultados en términos de rendimiento del modelo. Finalmente, el modelo entrenado se guarda en un archivo “model.h5”, lo que permite su reutilización sin necesidad de volver a entrenar en futuras instancias, facilitando así la exploración y evaluación del rendimiento del modelo en diversas aplicaciones

```

#!pip install tensorflow
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import GRU, Dropout

#rmsprop LOS MEJORES RESULTADOS
model, history = basic_GRU(optimizador = 'adam', unidades = 10, epocas = 10)
#Guardamos el modelo para no tener que entrenar otra vez para probar
model.save('MODELOS GRU/model.h5')

```

Figura 4.28. Configuración de las unidades y épocas del Modelo Unidades Recurrentes Cerradas GRU.

```

# Calcular el Mean Absolute Error (MAE)
mae = mean_absolute_error(semana_estudio_test, semana_estudio_pred)
print(f'MAE: {mae}')

# Calcular el Mean Absolute Percentage Error (MAPE)
mape = np.mean(np.abs((semana_estudio_test - semana_estudio_pred) / semana_estudio_test)) * 100
print(f'MAPE: {mape:.2f}%')
# Calcular el Mean Squared Error (MSE)
mse = mean_squared_error(semana_estudio_test, semana_estudio_pred)
print(f'MSE: {mse}')

# Calcular el coeficiente de determinación (R^2)
r2 = r2_score(semana_estudio_test, semana_estudio_pred)
print(f'R^2: {r2}')

```

Figura 4.29. Métricas de evaluación para el Modelo de Unidades Recurrentes Cerradas GRU

4.3.9.10.2. Visualización para comparación de resultados del Modelo de Unidades Recurrentes Cerradas GRU

Mediante la aplicación del comando "DataFrame", se crea una tabla con el propósito de compilar resultados comparativos. En esta tabla, se registran tanto las predicciones correspondientes a cada hora específica como los datos reales de Potencia Activa. Este enfoque habilita la realización de análisis detallados entre las predicciones del modelo y los valores reales observados. La construcción de esta estructura tabular simplifica la evaluación de la precisión del modelo, permitiendo una representación visual efectiva de las discrepancias y similitudes entre las predicciones y la realidad en cada punto horario, tal como se ilustra en la Figura 4.30.

```

#Union de datos
#LSTM nos predice el dato time_step + 1
data_test_slic = data_test[1:].copy()
data_test_pred = pd.concat([data_test_slic,
                             pd.DataFrame(prediccion,
                                           columns=['POTENCIA_PRED'],
                                           index = data_test_slic.index)],
                             axis =1)
# print(data_test_slic.shape)
# print(prediccion.shape)
# print(data_test_slic)
# print(prediccion)
# Guardar los resultados en un archivo Excel
data_test_pred.to_excel('resultados_prediccion.xlsx', index=True)
data_test_pred.head(100)

```

Figura 4.30. Creación de tabla de resultados de potencia activa pronosticada y real para el Modelo de Unidades Recurrentes Cerradas GRU

4.3.10. Aplicación del Modelo de Redes Neuronales LSTM

En la Figura 4.31, se utiliza la combinación de TensorFlow y Keras, dos bibliotecas populares para el desarrollo de modelos de aprendizaje profundo. En este ejemplo, se presenta la creación de un modelo secuencial que utiliza capas LSTM (Long Short-Term Memory), conocidas por su capacidad para capturar dependencias temporales en series temporales. El modelo se compone de capas LSTM apiladas seguidas de una capa densa para la salida. Después de compilar el modelo con un optimizador y una función de pérdida, se procede al entrenamiento utilizando datos de series temporales.

```
#IMPLEMENTACIÓN DEL MODELO
def NN_LSTM(optimizador, neuronas, epocas):
    dim_entrada = (X_train.shape[1],X_train.shape[2])
    dim_salida = 1
    num_neuronas = neuronas

    model = Sequential()

    model.add(LSTM(units=num_neuronas, return_sequences = True, input_shape=dim_entrada))
    model.add(Dropout(0.4))
    model.add(LSTM(units=num_neuronas))
    model.add(Dropout(0.2))
    model.add(Dense(units = dim_salida))
    # model.add(activation = 'linear')
    model.compile(optimizer= optimizador, loss='mse', metrics = ['mean_absolute_error']) #mean_squared_error

    #Entrenamiento
    history = model.fit(X_train, Y_train, epochs = epocas, batch_size = 128)
    return model, history
```

Figura 4.31. Modelo de Redes Neuronales LSTM

4.3.10.1.1. Preparación de datos para predicción del Modelo de Redes Neuronales LSTM

En la preparación de datos para la aplicación de un modelo de Redes Neuronales, como LSTM para series temporales, es esencial realizar pasos clave. Primero, se suele normalizar los datos para que estén en una escala adecuada, facilitando el proceso de entrenamiento. Luego, se estructuran los datos en ventanas temporales, creando secuencias de entrada y salida para entrenar la red de manera secuencial. A partir de esto, se divide el conjunto de datos en conjuntos de entrenamiento y prueba para evaluar el rendimiento del modelo en datos no vistos. Estas prácticas son fundamentales para mejorar la capacidad del modelo para capturar patrones temporales en series temporales específicas.

```

def predDataTest(modelo):
    x_test = data_test.values
    x_test = sc.transform(x_test)
    ## x_test
    X_test = []
    for i in range(time_step, len(x_test)):
        X_test.append(x_test[i-time_step: i, 0])
    X_test= np.array(X_test)
    X_test = np.reshape(X_test, (X_test.shape[0], X_test.shape[1], 1))
    X_test.shape
    prediccion = modelo.predict(X_test)
    prediccion = sc.inverse_transform(prediccion)
    return prediccion

```

Figura 4.32. Preparación de datos para predicción del Modelo de Redes Neuronales LSTM

En la Figura 4.33, se realiza la implementación de una red neuronal utilizando el algoritmo de optimización “rmsprop”, que ha demostrado ofrecer los mejores resultados en términos de rendimiento del modelo. La red neuronal está configurada con una capa LSTM (Long Short-Term Memory) que consta de 10 neuronas, y se lleva a cabo el proceso de entrenamiento a lo largo de 10 épocas. Posteriormente, con el objetivo de evitar la necesidad de repetir el proceso de entrenamiento al probar el modelo en instancias futuras

```

#rmsprop LOS MEJORES RESULTADOS
model, history = NN_LSTM(optimizador = 'rmsprop', neuronas = 10, epocas = 10)
#Guardamos el modelo para no tener que entrenar otra vez para probar
model.save('modelos/model.h5')

```

Figura 4.33. Configuración de las unidades y épocas del Modelo de Redes Neuronales LSTM

```

# Calcular el Mean Absolute Error (MAE)
mae = mean_absolute_error(semana_estudio_test, semana_estudio_pred)
print(f'MAE: {mae}')

# Calcular el Mean Absolute Percentage Error (MAPE)
mape = np.mean(np.abs((semana_estudio_test - semana_estudio_pred) / semana_estudio_test)) * 100
print(f'MAPE: {mape:.2f}%')

# Calcular el Mean Squared Error (MSE)
mse = mean_squared_error(semana_estudio_test, semana_estudio_pred)
print(f'MSE: {mse}')

# Calcular el coeficiente de determinación (R^2)
r2 = r2_score(semana_estudio_test, semana_estudio_pred)
print(f'R^2: {r2}')

```

Figura 4.34. Métricas de evaluación para el modelo de pronóstico del Modelo de Redes Neuronales LSTM

4.3.10.1.2. Visualización para comparación de resultados del Modelo de Redes Neuronales LSTM

Mediante la aplicación del comando "DataFrame", se genera una tabla diseñada para recopilar resultados comparativos. En esta tabla, se almacenan tanto los resultados de predicción correspondientes a cada hora específica como los datos reales de Potencia Activa. Este enfoque permite realizar comparaciones detalladas entre las predicciones del modelo y los valores reales observados. La construcción de esta estructura tabular simplifica la evaluación de la precisión del modelo, posibilitando una visualización efectiva de las discrepancias y similitudes entre las predicciones y la realidad en cada punto horario.

Este proceso de tabulación y comparación se relaciona con el uso de redes neuronales en la medida en que la red, al realizar predicciones, genera resultados que pueden ser organizados en una estructura tabular similar. Esto facilita la evaluación sistemática de la precisión de la red al comparar sus predicciones con los datos reales almacenados en la tabla, permitiendo una comprensión detallada de su desempeño en diferentes momentos temporales.

```
#Union de datos
#LSTM nos predice el dato time_step + 1
data_test_slic = data_test[1:].copy()
data_test_pred = pd.concat([data_test_slic,
                            pd.DataFrame(prediccion,
                                          columns=['POTENCIA_PRED'],
                                          index = data_test_slic.index)],
                            axis =1)
# print(data_test_slic.shape)
# print(prediccion.shape)
# print(data_test_slic)
# print(prediccion)
# Guardar los resultados en un archivo Excel
data_test_pred.to_excel('resultados_prediccionRNN.xlsx', index=True)
data_test_pred.head(100)
```

Figura 4.35. Creación de tabla de resultados de potencia activa pronosticada y real para el Modelo de Redes Neuronales LSTM

CAPÍTULO V

5. ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS

5.1. ANÁLISIS DIARIO DE RESULTADOS

Para llevar a cabo la predicción diaria, se realizó un análisis detallado de los datos correspondientes del año 2022. En este proceso, se realizó un pronóstico de la Potencia Activa, centrándose específicamente en el día 01 de enero de dicho año.

5.1.1. Modelo de Regresión Lineal Simple (Diario)

En la Tabla 5.1. presenta los resultados correspondientes al caso diario de Potencia Activa, tanto los valores reales como las predicciones.

Tabla 5.1. Resultado diario del Modelo de Regresión Lineal Simple.

FECHA	POTENCIA PRE. [kW]	POTENCIA REAL [kW]	ERROR [%]
01/01/2022 0:00	299,77	295,68	1,38
01/01/2022 0:30	299,10	294,98	1,40
01/01/2022 1:00	299,10	294,98	1,40
01/01/2022 1:30	300,43	296,38	1,37
01/01/2022 2:00	300,43	296,38	1,37
...
01/01/2022 21:30	299,10	294,98	1,40
01/01/2022 22:00	299,10	294,98	1,40
01/01/2022 22:30	300,43	296,38	1,37
01/01/2022 23:00	300,43	296,38	1,37
01/01/2022 23:30	301,10	297,09	1,35

La Figura 5.1, muestra el comportamiento tanto de la potencia real como de la potencia predicha para el día 01 de enero de 2022. Este análisis proporciona una comparación visual entre los datos reales observados y las predicciones realizadas por el modelo de Regresión Lineal.

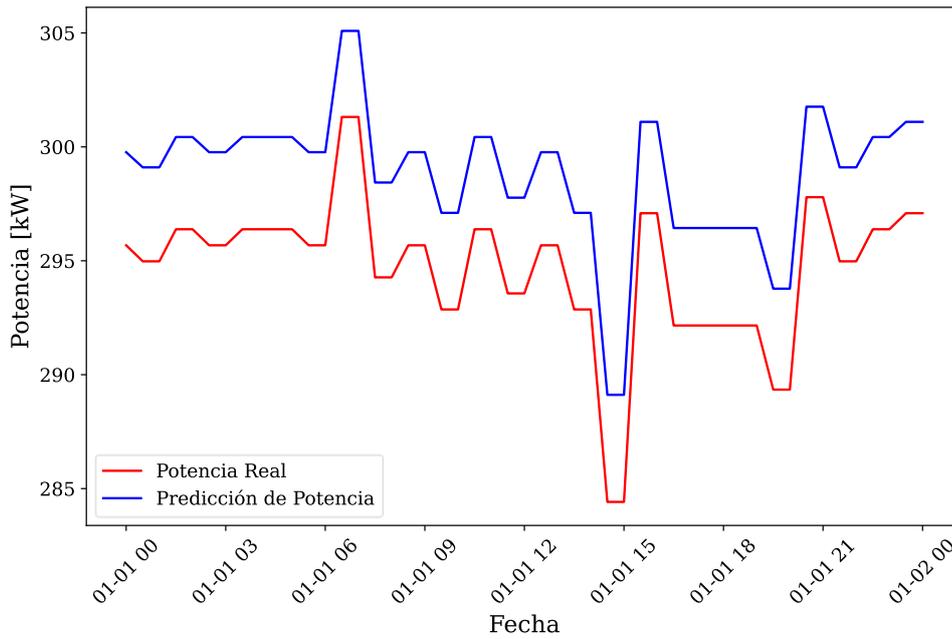


Figura 5.1. Pronóstico de la potencia activa del día 01 de enero del 2022 (Modelo de Regresión Lineal Simple)

5.1.2. Modelo de unidades recurrentes cerradas GRU (Diario)

En la Tabla 5.2, presenta los resultados correspondientes al caso diario de Potencia Activa, tanto los valores reales como las predicciones.

Tabla 5.2. Resultado diario del Modelo de Unidades Recurrentes Cerradas.

FECHA	POTENCIA REAL [kW]	POTENCIA PRE. [kW]	ERROR [%]
01/01/2022 0:00	295,68	297,10	0,48
01/01/2022 0:30	294,98	297,10	0,72
01/01/2022 1:00	294,98	296,21	0,42
01/01/2022 1:30	296,38	296,21	0,06
01/01/2022 2:00	296,38	297,99	0,54
...
01/01/2022 21:30	294,98	299,77	1,62
01/01/2022 22:00	294,98	296,21	0,42
01/01/2022 22:30	296,38	296,21	0,06
01/01/2022 23:00	296,38	297,99	0,54
01/01/2022 23:30	297,09	297,99	0,30

La Figura 5.2, muestra el comportamiento tanto de la potencia real como de la potencia predicha para el día 01 de enero de 2022. Este análisis proporciona una comparación visual entre los datos reales observados y las predicciones realizadas por el modelo de Unidades Recurrentes Cerradas.

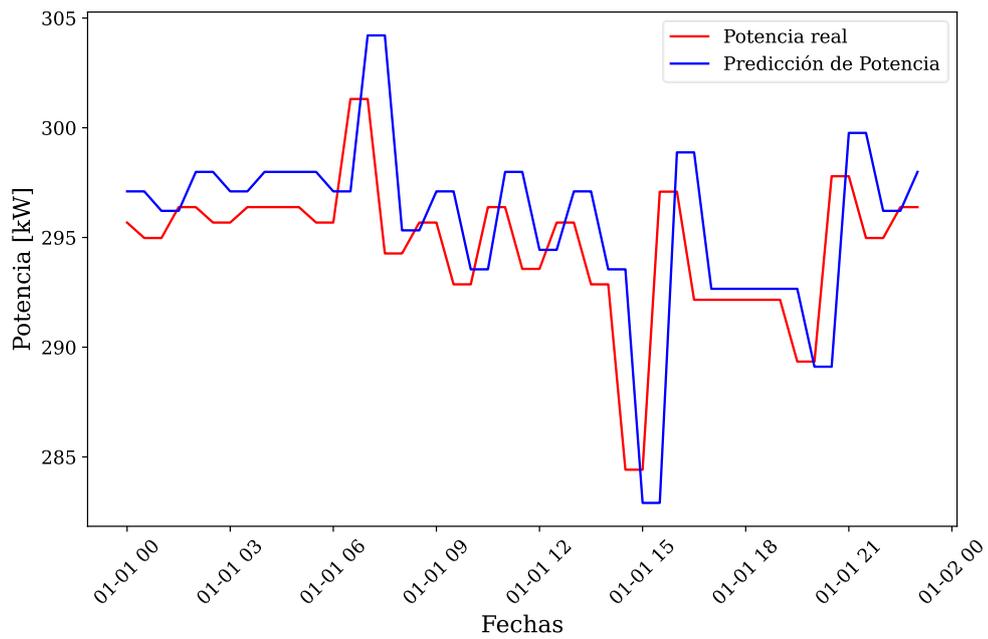


Figura 5.2. Pronóstico de la potencia activa del día 01 de enero del 2022 (Modelo de Unidades Recurrentes Cerradas de GRU)

5.1.3. Modelo de Redes Neuronales LSTM (Diario)

En la Tabla 5.3, presenta los resultados correspondientes al caso diario de Potencia Activa, tanto los valores reales como las predicciones.

Tabla 5.3. Resultado diario del Modelo de Redes Neuronales.

FECHA	POTENCIA REAL [kW]	POTENCIA PRE. [kW]	ERROR [%]
01/01/2022 0:00	295,68	299,46	1,28
01/01/2022 0:30	294,98	299,46	1,52
01/01/2022 1:00	294,98	298,84	1,31
01/01/2022 1:30	296,38	298,84	0,83
01/01/2022 2:00	296,38	300,09	1,25
...
01/01/2022 21:30	294,98	298,81	1,30
01/01/2022 22:00	294,98	296,40	0,48
01/01/2022 22:30	296,38	296,40	0,01
01/01/2022 23:00	296,38	297,61	0,41
01/01/2022 23:30	297,09	297,61	0,17

La Figura 5.3, muestra el comportamiento tanto de la potencia real como de la potencia predicha para el día 01 de enero de 2022. Este análisis proporciona una comparación visual entre los datos reales observados y las predicciones realizadas por el modelo de Redes Neuronales.

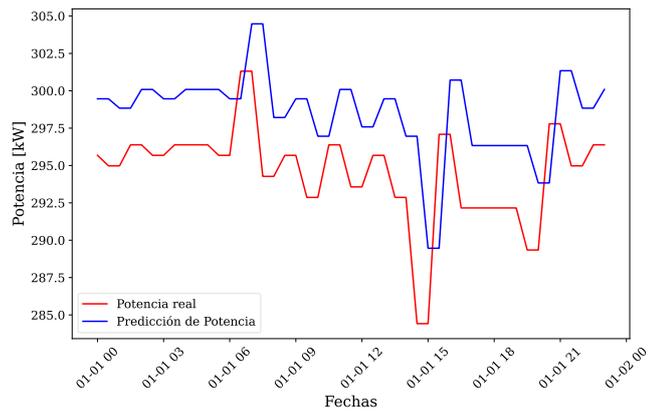


Figura 5.3. Pronóstico de la potencia activa del día 01 de enero del 2022 Modelo de Redes Neuronales LSTM

5.1.4. Comparación de los Modelos de Aprendizaje Automático

En la Figura 5.4, se presentan los resultados del pronóstico de Potencia Activa y Potencia Activa Real utilizando los Modelos de Aprendizaje Automático en el caso de estudio correspondiente al 01 de enero de 2022.

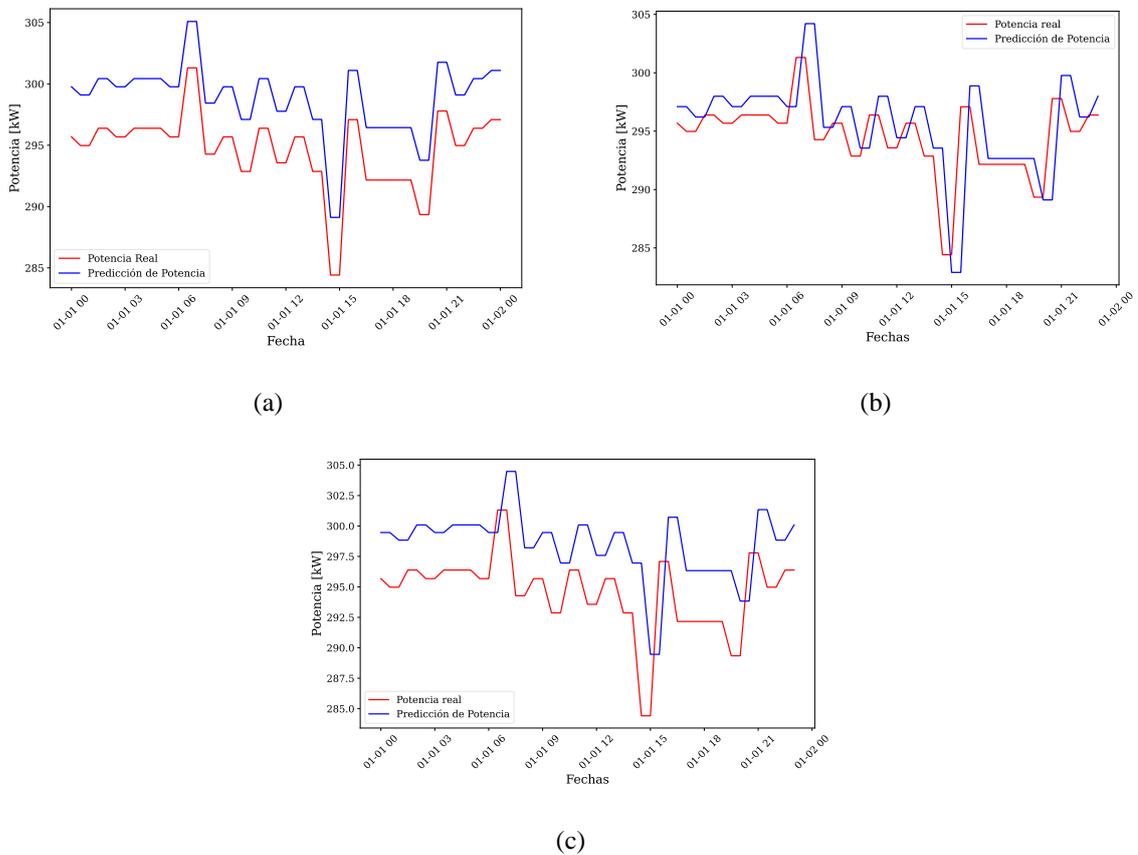


Figura 5.4. Modelos de Aprendizaje Automático de Pronostico de Potencia Activa: (a) Regresión Lineal Simple, (b) Unidades Recurrentes Cerradas GRU, (c) Redes Neuronales LSTM

5.1.5.Métricas de evaluación de análisis diario

En la Tabla 5.4 se presentan las métricas de evaluación para el caso diario de los tres modelos utilizados. Se observa claramente que el modelo de Unidades Recurrentes Cerradas (GRU) muestra un mejor ajuste y rendimiento, ya que sus métricas son inferiores en comparación con las de los otros modelos aplicados. Por ejemplo, en el Error Medio Absoluto (MAE) obtiene un valor de 2.24, mientras que en el Error Medio Cuadrático (MSE) alcanza un valor de 13.75. El MSE al penalizar más a los errores grandes que el MAE, resulta ser una métrica más confiable para verificar el ajuste del modelo. Además, el Error de Porcentaje Medio Absoluto (MAPE) tiene un valor del 0.76%.

Tabla 5.4. Resultados de las métricas de evaluación de cada Modelo de Aprendizaje Automático (Diario)

Técnica	Coefficiente de Determinación R^2	Error medio absoluto (MAE)	Error de Porcentaje Medio Absoluto (MAPE) [%]	Error Medio Cuadrático (MSE)
Regresión Lineal Simple	0.71	4.14	1.40	17.17
Unidades Recurrentes Cerradas	0.92	2.24	0.76	13.75
Redes Neuronales LSTM	0.81	4.17	1.42	21.67

5.2.ANÁLISIS SEMANAL DE RESULTADOS

Para realizar la predicción semanal, se verifica un análisis detallado de los datos correspondientes al año 2022. En este proceso, se realiza un pronóstico de la Potencia Activa, centrándose específicamente en el día 01 al 7 de enero del mismo año.

5.2.1.Modelo de Regresión Lineal Simple (Semanal)

En la Tabla 5.5, presenta los resultados correspondientes al caso semanal de Potencia Activa, tanto los valores reales como las predicciones.

Tabla 5.5. Resultado semanal del Modelo de Regresión Lineal Simple.

FECHA	POTENCIA PRE. [kW]	POTENCIA REAL [kW]	ERROR [%]
01/01/2022 0:00	299,77	295,68	1,38
01/01/2022 0:30	299,10	294,98	1,40
01/01/2022 1:00	299,10	294,98	1,40
01/01/2022 1:30	300,43	296,38	1,37
01/01/2022 2:00	300,43	296,38	1,37
...
07/01/2022 21:30	296,44	292,16	1,46
07/01/2022 22:00	296,44	292,16	1,46
07/01/2022 22:30	297,77	293,57	1,43
07/01/2022 23:00	297,77	293,57	1,43
07/01/2022 23:30	301,76	297,79	1,33

La Figura 5.5, muestra el comportamiento tanto de la potencia real como de la potencia predicha desde el día 01 al 07 de enero de 2022. Este análisis proporciona una comparación visual entre los datos reales observados y las predicciones realizadas por el modelo de Regresión Lineal Simple.

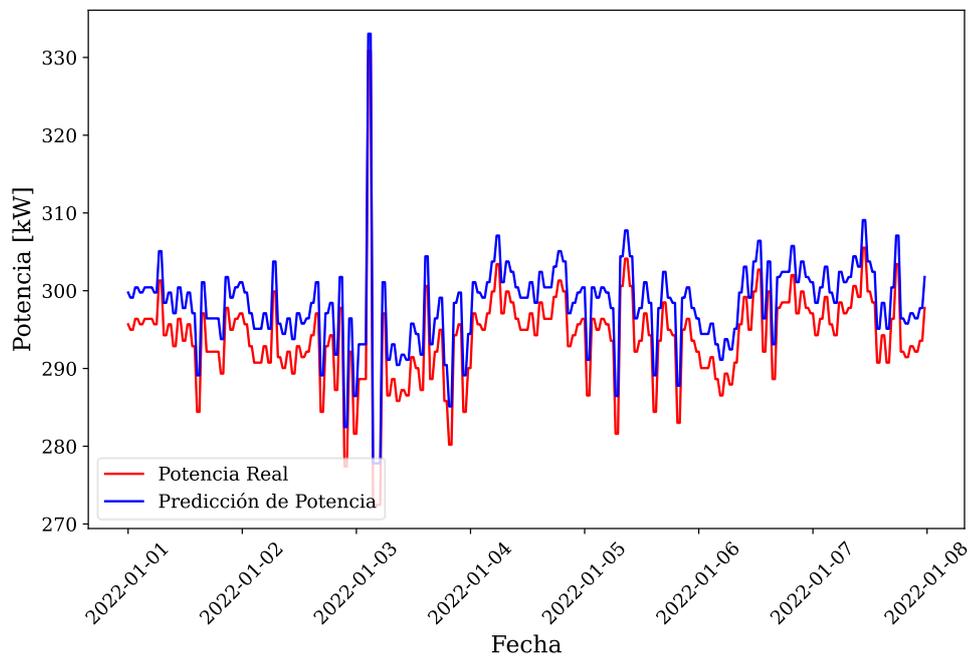


Figura 5.5. Pronóstico de la potencia activa del día 01 al 07 de enero del 2022 (Modelo de Regresión Lineal Simple)

5.2.2. Modelo de Unidades Recurrentes Cerradas GRU (Semanal)

En la Tabla 5.6, presenta los resultados correspondientes al caso semanal de Potencia Activa, tanto los valores reales como las predicciones.

Tabla 5.6. Resultado semanal del Modelo de Unidades Recurrentes Cerradas.

FECHA	POTENCIA REAL [kW]	POTENCIA PRE. [kW]	ERROR [%]
01/01/2022 0:00	295,68	296,61	0,31
01/01/2022 0:30	294,98	296,61	0,55
01/01/2022 1:00	294,98	295,70	0,25
01/01/2022 1:30	296,38	295,70	0,23
01/01/2022 2:00	296,38	297,51	0,38
...
07/01/2022 21:30	292,16	293,55	0,48
07/01/2022 22:00	292,16	292,66	0,17
07/01/2022 22:30	293,57	292,66	0,31
07/01/2022 23:00	293,57	294,44	0,30
07/01/2022 23:30	297,79	294,44	1,13

La Figura 5.6, muestra el comportamiento tanto de la potencia real como de la potencia predicha desde el día 01 al 07 de enero de 2022. Este análisis proporciona una comparación visual entre los datos reales observados y las predicciones realizadas por el modelo de Unidades Recurrentes Cerradas.

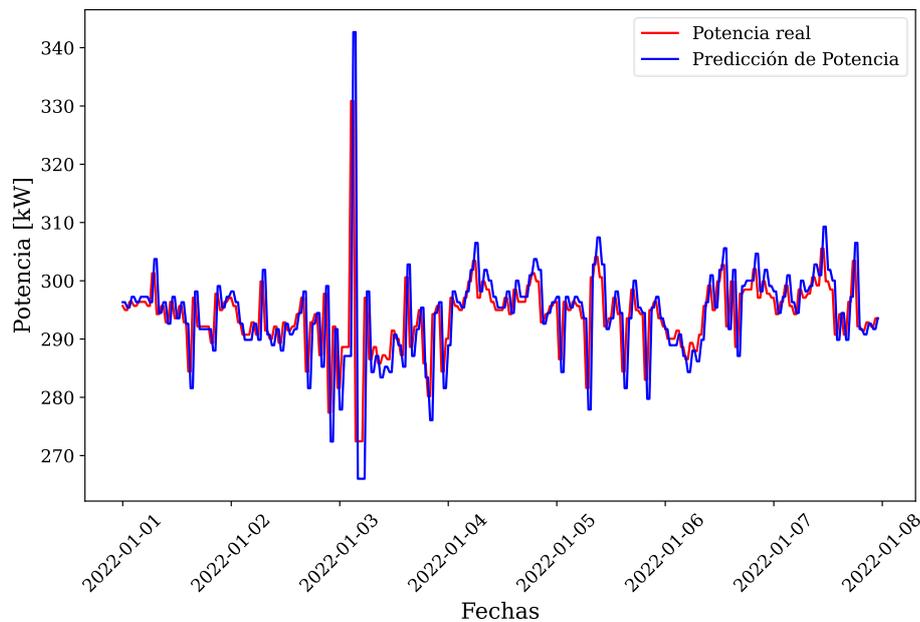


Figura 5.6. Pronóstico de la potencia activa del día 01 al 07 de enero del 2022 (Modelo de GRU)

5.2.3. Modelo de Redes Neuronales LSTM (Semanal)

En la Tabla 5.7, presenta los resultados correspondientes al caso semanal de Potencia Activa, tanto los valores reales como las predicciones.

Tabla 5.7. Resultado semanal del Modelo de Redes Neuronales.

FECHA	POTENCIA REAL [kW]	POTENCIA PRE. [kW]	ERROR [%]
01/01/2022 0:00	295,68	299,46	1,28
01/01/2022 0:30	294,98	299,46	1,52
01/01/2022 1:00	294,98	298,84	1,31
01/01/2022 1:30	296,38	298,84	0,83
01/01/2022 2:00	296,38	300,09	1,25
...
07/01/2022 21:30	292,16	296,96	1,64
07/01/2022 22:00	292,16	296,33	1,43
07/01/2022 22:30	293,57	296,33	0,94
07/01/2022 23:00	293,57	297,59	1,37
07/01/2022 23:30	297,79	297,59	0,07

La Figura 5.7; **Error! No se encuentra el origen de la referencia.**, muestra el comportamiento tanto de la potencia real como de la potencia predicha desde el día 01 al 07 de enero de 2022. Este análisis proporciona una comparación visual entre los datos reales observados y las predicciones realizadas por el modelo de Redes Neuronales.

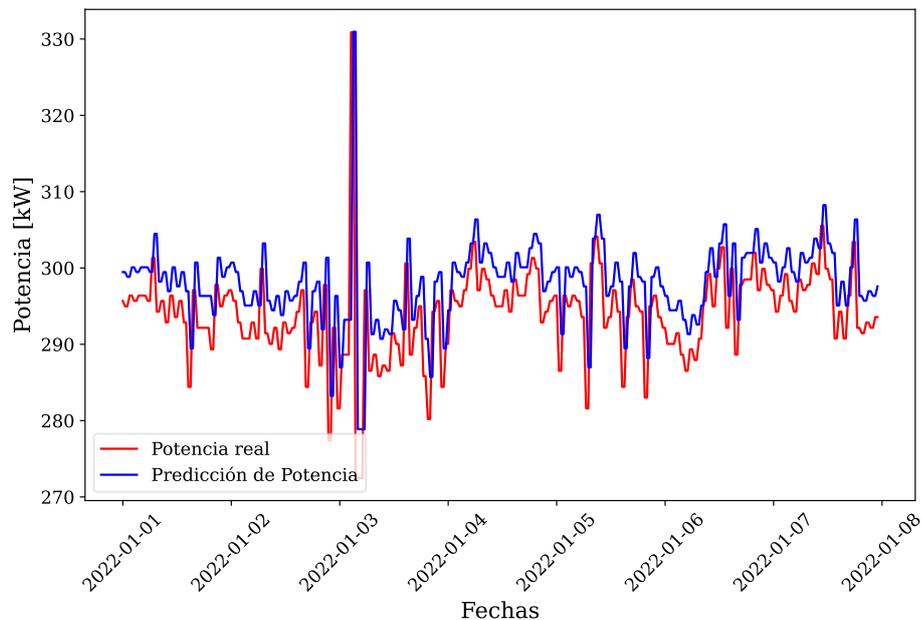


Figura 5.7. Pronóstico de la potencia activa del día 01 al 07 de enero del 2022 (Modelo de Redes Neuronales LSTM)

5.2.4. Comparación de los Modelos de Aprendizaje Automático

En la Figura 5.8, se presentan los resultados del pronóstico de Potencia Activa y Potencia Activa Real utilizando los Modelos de Aprendizaje Automático en el caso de estudio correspondiente al 01 de enero de 2022 hasta el 07 de enero de 2022.

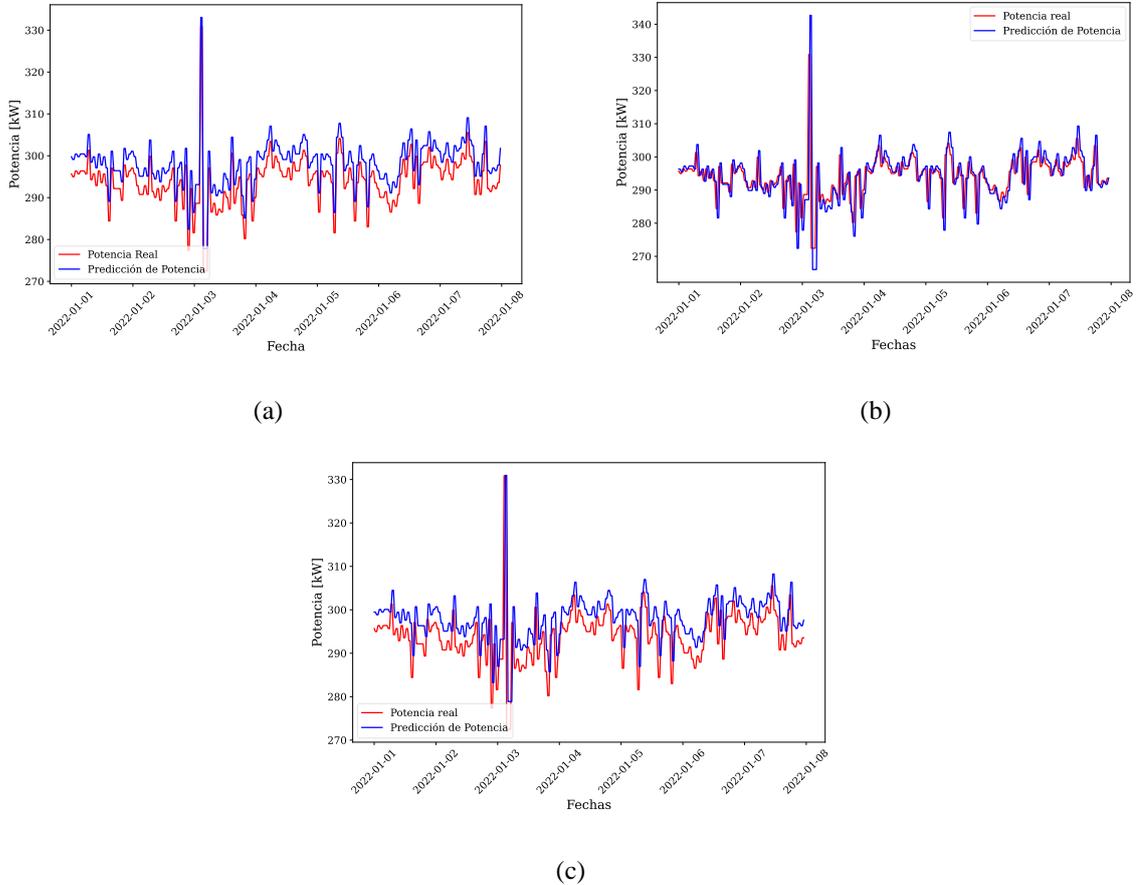


Figura 5.8. Modelos de Aprendizaje Automático de Pronóstico de Potencia Activa: (a) Regresión Lineal Simple, (b) Unidades Recurrentes Cerradas GRU, (c) Redes Neuronales LSTM

5.2.5. Métricas de evaluación de análisis semanal

Tabla 5.8. Resultados de las métricas de evaluación de cada modelo (Semanal)

Técnica	Coefficiente de Determinación R^2	Error medio absoluto (MAE)	Error de Porcentaje Medio Absoluto (MAPE) [%]	Error Medio Cuadrático (MSE)
Regresión Lineal Simple	0.52	4.17	1.42	47.54
Unidades Recurrentes Cerradas	0.94	3.44	1.18	43.77
Redes Neuronales LSTM	0.87	4.71	1.61	46.13

En la Tabla 5.8 se presentan las métricas de evaluación para el caso semanal de los tres modelos utilizados. Concretamente, el MAE del modelo GRU se sitúa en 3.44, señalando una precisión significativa en la predicción semanal de Potencia Activa. Este valor difiere con el Error Medio Cuadrático (MSE), que alcanza 43.77, indicando que, aunque existe cierta dispersión cuadrática de los errores, el MAE proporciona una medida más robusta de la magnitud de los errores absolutos. Adicionalmente, el MAPE del modelo GRU se establece en un mínimo del 1.18%, subrayando la capacidad del modelo para prever con precisión y minimizar los errores relativos en el contexto del escenario semanal. Estas métricas respaldan la afirmación de que el modelo GRU demuestra un mejor desempeño en comparación con los otros modelos evaluados.

5.3. ANÁLISIS MENSUAL DE RESULTADOS

Para realizar la predicción de un mes, se verificó un análisis detallado de los datos correspondientes al año 2022. En este proceso, se realiza un pronóstico de la Potencia Activa, centrándose específicamente de todo el mes de enero del día 1 al 31 del mismo año.

5.3.1. Modelo de Regresión Lineal Simple (Mensual)

En la Tabla 5.9, presenta los resultados correspondientes al caso mensual de Potencia Activa, tanto los valores reales como las predicciones.

Tabla 5.9. Resultado mensual del Modelo de Regresión Lineal.

FECHA	POTENCIA PRE. [kW]	POTENCIA REAL [kW]	ERROR [%]
01/01/2022 0:00	299,77	295,68	1,38
01/01/2022 0:30	299,10	294,98	1,40
01/01/2022 1:00	299,10	294,98	1,40
01/01/2022 1:30	300,43	296,38	1,37
01/01/2022 2:00	300,43	296,38	1,37
...
31/01/2022 18:00	225,22	216,83	3,87
31/01/2022 18:30	224,55	216,13	3,90
31/01/2022 19:00	224,55	216,13	3,90
31/01/2022 19:30	227,88	219,65	3,75
31/01/2022 20:00	227,88	219,65	3,75

La Figura 5.9, muestra el comportamiento tanto de la potencia real como de la potencia predicha de todo el mes de enero de 2022. Este análisis proporciona una comparación visual entre los datos reales observados y las predicciones realizadas por el modelo de Regresión Lineal Simple.

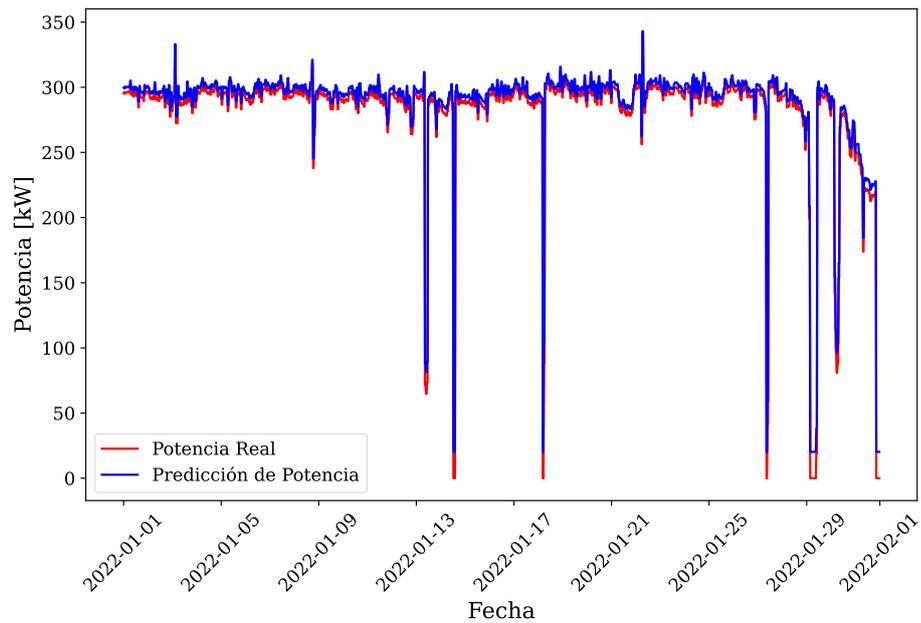


Figura 5.9. Pronóstico de la potencia activa del mes de enero del 2022 (Modelo de Regresión Lineal Simple)

5.3.2. Modelo de Unidades Recurrentes Cerradas GRU (Mensual)

En la Tabla 5.10, presenta los resultados correspondientes al caso mensual de Potencia Activa, tanto los valores reales como las predicciones.

Tabla 5.10. Resultado semanal del Modelo de Unidades Recurrentes Cerradas.

FECHA	POTENCIA REAL [kW]	POTENCIA PRE. [kW]	ERROR [%]
01/01/2022 0:00	295,68	296,61	0,31
01/01/2022 0:30	294,98	296,61	0,55
01/01/2022 1:00	294,98	295,70	0,25
01/01/2022 1:30	296,38	295,70	0,23
01/01/2022 2:00	296,38	297,51	0,38
...
31/01/2022 18:00	216,83	215,62	0,56
31/01/2022 18:30	216,13	214,62	0,70
31/01/2022 19:00	216,13	215,81	0,15
31/01/2022 19:30	219,65	218,81	0,38
31/01/2022 20:00	219,65	218,88	0,35

La Figura 5.10, muestra el comportamiento tanto de la potencia real como de la potencia predicha de todo el mes de enero de 2022. Este análisis proporciona una comparación visual entre los datos reales observados y las predicciones realizadas por el modelo de Unidades Recurrentes Cerradas.

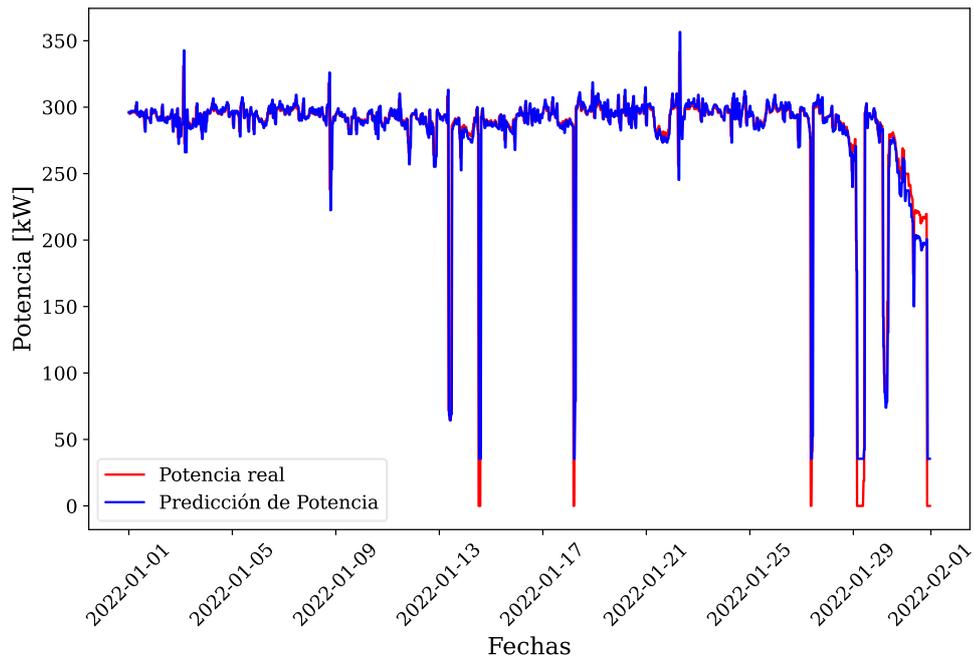


Figura 5.10. Pronóstico de la potencia activa del mes de enero del 2022 (Modelo de Unidades Recurrentes Cerradas GRU)

5.3.3. Modelo de Redes Neuronales LSTM (Mensual)

En la Tabla 5.11, presenta los resultados correspondientes al caso mensual de Potencia Activa, tanto los valores reales como las predicciones.

Tabla 5.11. Resultado semanal del Modelo de Redes Neuronales.

FECHA	POTENCIA REAL [kW]	POTENCIA PRE. [kW]	ERROR [%]
01/01/2022 0:00	295,68	299,46	1,28
01/01/2022 0:30	294,98	299,46	1,52
01/01/2022 1:00	294,98	298,84	1,31
01/01/2022 1:30	296,38	298,84	0,83
01/01/2022 2:00	296,38	300,09	1,25
...
31/01/2022 18:00	216,83	219,33	1,15
31/01/2022 18:30	216,13	218,33	1,02
31/01/2022 19:00	216,13	218,73	1,20
31/01/2022 19:30	219,65	221,73	0,95
31/01/2022 20:00	219,65	215,76	1,77

La Figura 5.11, muestra el comportamiento tanto de la potencia real como de la potencia predicha de todo el mes de enero de 2022. Este análisis proporciona una comparación visual entre los datos reales observados y las predicciones realizadas por el modelo de Redes Neuronales (LSTM).

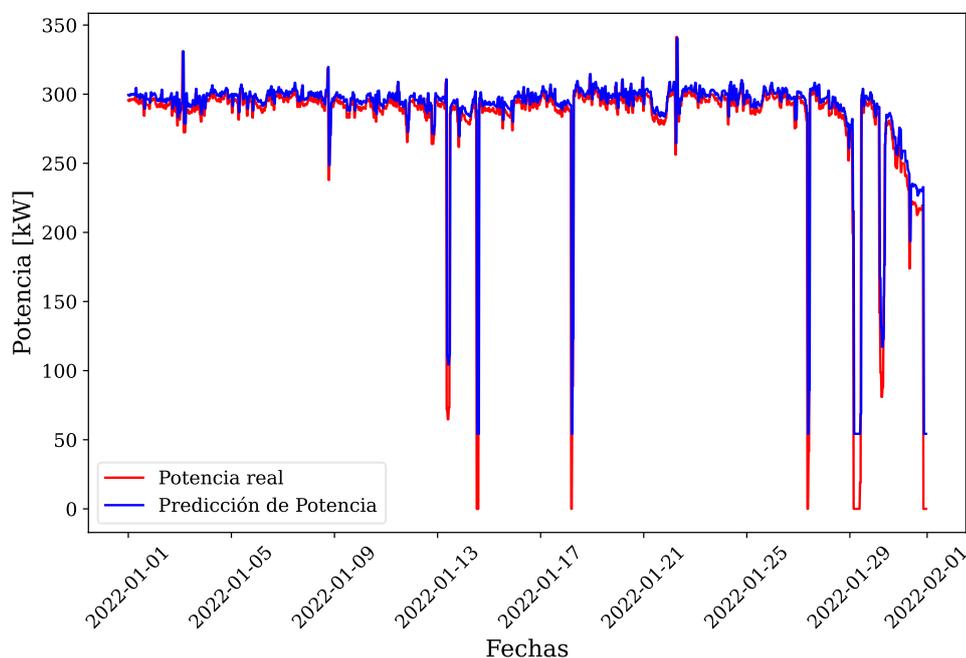


Figura 5.11. Pronóstico de la potencia activa del mes de enero del 2022 (Modelo LSTM)

5.3.4. Métricas de evaluación de análisis mensual

Tabla 5.12. Resultados de las métricas de evaluación de cada modelo (Mensual)

Técnica	Coefficiente de Determinación R^2	Error medio absoluto (MAE)	Error de Porcentaje Medio Absoluto (MAPE) [%]	Error Medio Cuadrático (MSE)
Regresión Lineal Simple	0.46	8.28	1.87	574.71
Unidades Recurrentes Cerradas	0.81	4.84	1.42	558.29
Redes Neuronales LSTM	0.76	6.87	1.82	560.36

En Tabla 5.12 se presentan las métricas de evaluación correspondientes al caso semanal de los tres modelos empleados para la predicción. Específicamente, el modelo de Unidades Recurrentes Cerradas (GRU) destaca con un MAE de 4.84, señalando una notable precisión en la estimación de la variable considerada. Este valor contrasta con el Error Medio Cuadrático

(MSE) de 558.29, indicando una dispersión cuadrática de los errores que, aunque presente, no invalida la solidez del MAE como medida más robusta de la magnitud de los errores absolutos. Adicionalmente, el Error de Porcentaje Medio Absoluto (MAPE) para el modelo GRU se encuentra en 1.42.

5.3.5. Comparación de los Modelos de Aprendizaje Automático

En la Figura 5.12, se presentan los resultados del pronóstico de Potencia Activa y Potencia Activa Real utilizando los Modelos de Aprendizaje Automático en el caso de estudio correspondiente al mes de enero de 2022.

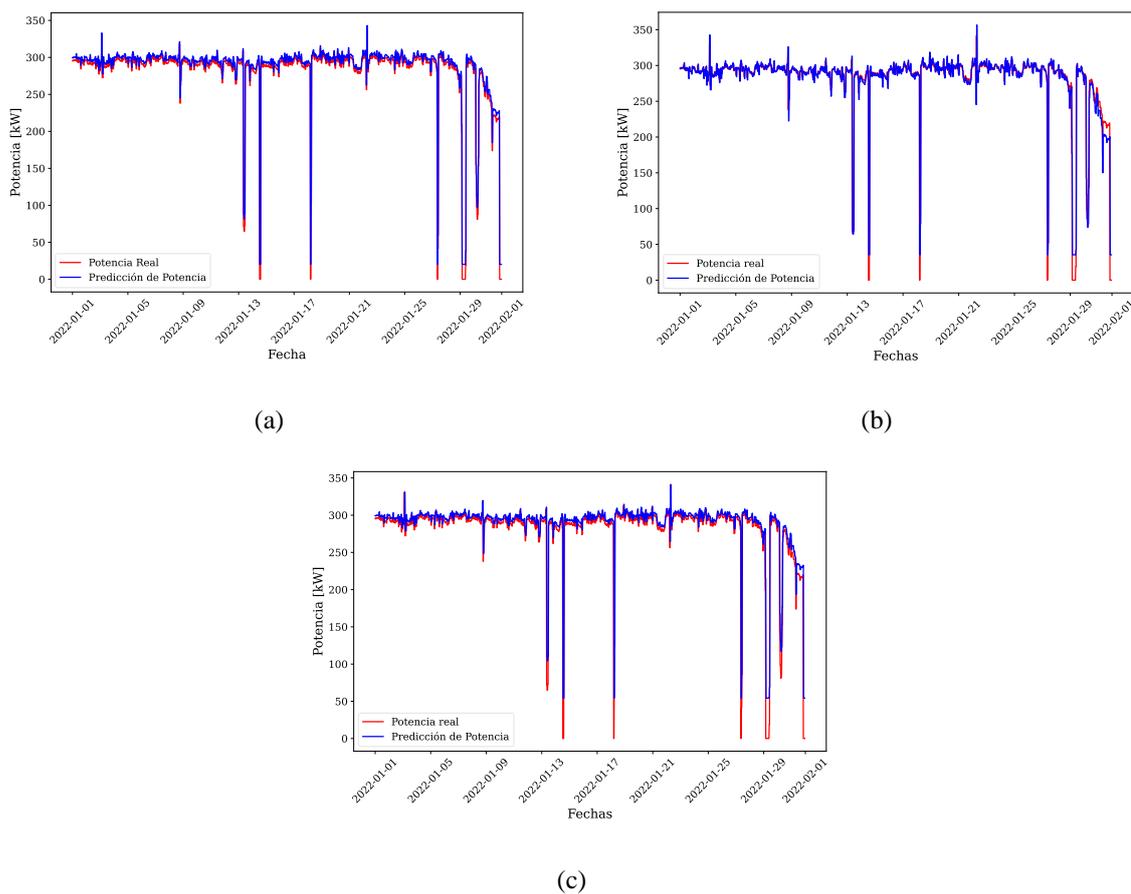


Figura 5.12. Modelos de Aprendizaje Automático de Pronostico de Potencia Activa: (a) Regresión Lineal Simple, (b) Unidades Recurrentes Cerradas GRU, (c) Redes Neuronales LSTM

5.3.6. Justificación por el Error De Truncamiento para el Modelo GRU Caso Diario

Mediante el análisis de resultados de los diferentes Modelos de Aprendizaje Automático aplicados dentro de los horizontes de tiempo (diario, semanal y mensual), se identificó que las curvas de predicción se encuentran por encima de las curvas de potencia real, esto se debe a que los resultados de predicción se aproximan causando así un error de truncamiento haciendo que los resultados se eleven. A causa de esto, se recomienda aplicar el $\pm 1\%$ a los resultados de predicción de potencia activa total, con el fin de tener un rango de potencia de predicción. Por esta razón se recomienda elegir la curva del -1% , ya que esta curva está por debajo de la curva real, así como se muestra en la Figura 5.14.

Esta sugerencia asegura que las predicciones se mantendrán consistentemente por debajo de los valores reales, lo que favorece la igualdad entre la generación y la demanda. Esto, a su vez, garantiza la estabilidad continua del sistema.

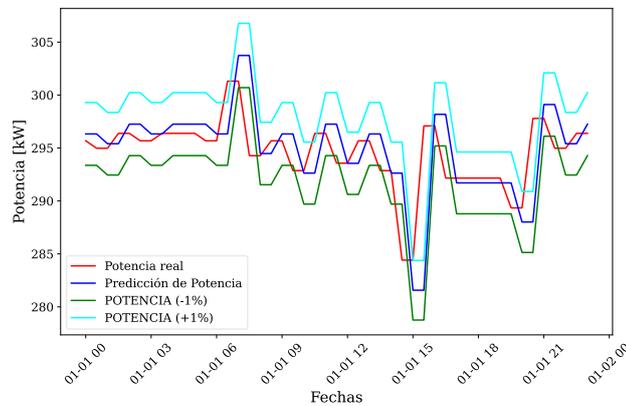


Figura 5.13. Recomendación del $\pm 1\%$ para el Modelo de Unidades Recurrentes Cerradas GRU Caso Diario

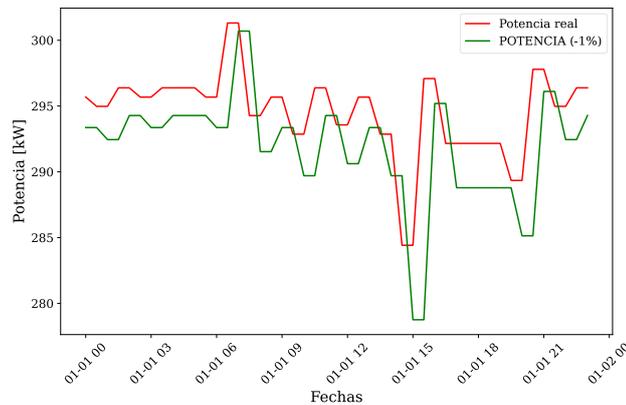


Figura 5.14 Recomendación del -1% para el Modelo de Unidades Recurrentes Cerradas GRU Caso Diario

CAPÍTULO VI

6.CONCLUSIONES Y RECOMENDACIONES

6.1.CONCLUSIONES

- La aplicación de modelos de pronóstico basados en los parámetros de generación eléctrica resultó en un proceso analítico robusto y eficiente. La integración de estos modelos permitió anticipar con precisión la generación eléctrica, contribuyendo significativamente a la planificación y gestión de recursos. La aplicación práctica de los modelos demostró su capacidad para adaptarse a la complejidad de los datos de generación eléctrica, proporcionando predicciones útiles y orientadas a la toma de decisiones informada.
- Después de completar detalladamente todas las fases de evaluación en diferentes horizontes de tiempo (diario, semanal y mensual), se ha concluido que el Modelo de Unidades Recurrentes Cerradas (GRU) muestra el ajuste más destacado a los datos de Potencia Activa Real. Se han obtenido valores significativos en la métrica de evaluación del Error de Porcentaje Medio Absoluto (MAPE), registrando un 0.76% en el caso diario, 1.18% en el caso semanal y 1.42% en el caso mensual. Por otro lado, el Modelo de Redes Neuronales LSTM muestra un ajuste superior al Modelo de Regresión Lineal Simple, con resultados de MAPE del 1.42% en el caso diario, 1.61% en el caso semanal y 1.82% en el caso mensual. Finalmente, para el Modelo de Regresión Lineal Simple, se determinó que los valores de MAPE son superiores a los dos Modelos de Aprendizaje Automático (Unidades Recurrentes Cerradas GRU y Redes Neuronales LSTM) en los diferentes horizontes de tiempo.
- La flexibilidad y la amplia variedad de bibliotecas especializadas en aprendizaje automático, como Scikit-Learn y TensorFlow, hacen de Python una herramienta integral para desarrollar y evaluar modelos de pronóstico de manera eficiente como el Modelo de Regresión Lineal Simple, Modelo de Unidades Recurrentes Cerradas (GRU) y el Modelo de Redes Neuronales LSTM. Además, la comunidad del lenguaje Python y su robusta infraestructura de desarrollo proporcionan un entorno favorable para la experimentación, ajuste y mejora continua de estos modelos, permitiendo adaptar eficazmente las soluciones a distintos contextos y conjuntos de datos.

- La investigación realizada ha demostrado que los Modelos de Aprendizaje Automático brindan una aproximación estadística del comportamiento de la Potencia Activa, lo que resulta altamente ventajoso para la planificación y operación de la Mini Central Hidroeléctrica Catazación. El resultado de este estudio destaca la importancia de estas técnicas en el ámbito energético, en particular para el departamento de generación y operación de la empresa eléctrica ELEPCO S.A. De esta manera, se demuestra la urgencia de que este departamento envíe al Operador Nacional de Electricidad CENACE un reporte detallado al día siguiente sobre la generación eléctrica de la central. Este flujo de información permite una gestión eficiente y oportuna de la generación eléctrica, lo que contribuye a la estabilidad y eficacia general del sistema eléctrico.

6.2.RECOMENDACIONES

- Se recomienda que, al aplicar los Modelos de Aprendizaje Automático, se considere la disponibilidad de una mayor cantidad de datos históricos, ya que esto conllevará a una mayor precisión en la predicción de la curva de potencia activa real, por lo que el aumento de la cantidad de datos históricos contribuye significativamente a mejorar la capacidad predictiva de los modelos, lo que resulta en pronósticos más precisos y confiables.
- Se sugiere explorar otros modelos de Aprendizaje Automático, como la Regresión Lineal Polinómica, Árbol de Decisión y Random Forest, con el propósito de identificar cuál ofrece una estimación más precisa de la curva real, en comparación con los modelos de Aprendizaje Automático empleados en esta investigación.
- Se puede considerar la implementación de los modelos de aprendizaje automático en otros tipos de software, tales como Matlab, Orange y KNIME, ya que estas plataformas ofrecen una amplia gama de herramientas y funcionalidades para el desarrollo y la evaluación de modelos predictivos, que podría permitir una mayor flexibilidad y adaptabilidad a las necesidades específicas de la investigación.

CAPÍTULO VII

7.REFERENCIAS

- [1] J. G. C. R. Salazar, «Predicción de flujo para la generación de energía hidroeléctrica utilizando redes neuronales LMST,» Octubre 2023. [En línea]. Available: https://www.researchgate.net/publication/374895488_Flow_Prediction_for_Hydropower_Generation_using_LMST_Neural_Networks. [Último acceso: 2 Diciembre 2023].
- [2] R. M. Aguilar, J. M. Torres y C. A. Martín, «Aprendizaje Automático en la Identificación de Sistemas. Un Caso de Estudio en la Predicción de la Generación Eléctrica de un Parque Eólico,» 2018. [En línea]. Available: <https://doi.org/10.4995/riai.2018.9421>. [Último acceso: 28 10 2023].
- [3] J. Segovia, J. F. Toaquiza, J. Llanos y D. Rivas, «Meteorological Variables Forecasting System Using Machine Learning and Open-Source Software,» Electronics, Febrero 2023. [En línea]. Available: <https://www.mdpi.com/2079-9292/12/4/1007>. [Último acceso: 30 10 2023].
- [4] F. Bustamante, «Unidades Recurrentes Cerradas (GRU) vs Redes Neuronales Artificiales en la predicción de la generación eléctrica de la Central Hidroeléctrica Illuchi,» 2023. [En línea]. Available: <https://repositorio.utc.edu.ec/jspui/bitstream/27000/10774/1/MUTC-001708.pdf>. [Último acceso: 31 10 2023].
- [5] H. Reyes y A. Ulke, «Forecasting the hydroelectric power generation of GCMs using machine learning techniques and deep learning (Almus Dam, Turkey),» Octubre 2020. [En línea]. Available: https://hrcak.srce.hr/ojs/index.php/geofizika/article/view/16932/volume38_1.

- [6] B. J y J. Gomez, «Hydropower production prediction using artificial neural networks: an Ecuadorian application case,» 2021. [En línea]. Available: <https://doi.org/10.1007/s00521-021-06746-5>. [Último acceso: 02 10 2023].
- [7] Z. W, W. Wen y X. Li, «A Novel Method of GRU Interval Prediction of Short-term Electric Load Considering Real-time Electricity Price,» Agosto 2023. [En línea]. Available: <https://ieeexplore.ieee.org/abstract/document/10263127>. [Último acceso: 3 11 2023].
- [8] W. R. S. Erazo, «Conceptos Básicos de Centrales Hidroeléctricas,» Julio 2018. [En línea]. Available: <file:///C:/Users/Lenovo/Downloads/Cap61.pdf>. [Último acceso: 15 Noviembre 2023].
- [9] Q. V, «ESTUDIO Y PLANIFICACION DEL SISTEMA DE DSITRIBUCION ELECTRICA DEL CANTON LA MANA JURISDICCION DE LA EMPRESA ELECTRICA PROVINCIAL DE COTOPAXI (ELEPCO S. A.) UTILIZANDO EL PROGRAMA NEPLAN,» 2009. [En línea]. Available: <https://bibdigital.epn.edu.ec/bitstream/15000/1565/1/CD-2241.pdf>. [Último acceso: 15 11 2023].
- [10] E. A. J. B. L. B. U. C. R. G. J. G. B. L. M. M. M. S. A. Moreno, «Aprendizaje Automático,» [En línea]. Available: <https://upcommons.upc.edu/bitstream/handle/2099.3/36157/9788483019962.pdf>. [Último acceso: 16 Noviembre 2023].
- [11] G. J. N. Benito, «Predicción de series temporales mediante técnicas de aprendizaje automático. Automatización del proceso,» 2023. [En línea]. Available: https://idus.us.es/bitstream/handle/11441/149376/TFG4769_Nieto%20Benito.pdf?sequence=1&isAllowed=y. [Último acceso: 16 Noviembre 2023].
- [12] R. V. Baños, M. Torrado y M. Reguant, «Análisis de regresión lineal múltiple con SPSS: un ejemplo práctico,» 2018. [En línea]. Available:

<https://revistes.ub.edu/index.php/REIRE/article/view/reire2019.12.222704/28913>.
[Último acceso: 3 11 2023].

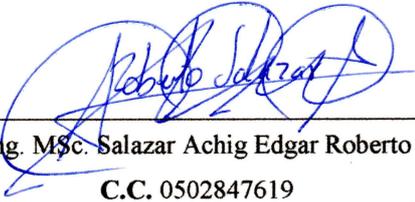
- [13] C. Duarte, «ESTIMACION DE LA DEMANDA ELECTRICA EN USUARIOS DE BAJA TENSION UTILIZANDO EL METODO DE REGRESION,» [En línea]. Available: <https://dspace.ups.edu.ec/bitstream/123456789/25832/1/UPS-GT004532.pdf>.
- [14] C. Espino, «Análisis predictivo: técnicas y modelos utilizados y aplicaciones del mismo-herramientas Open Source que permiten su uso,» 2017. [En línea]. Available: <https://openaccess.uoc.edu/bitstream/10609/59565/6/caresptimTFG0117mem%C3%B2ria.pdf>. [Último acceso: 07 11 2023].
- [15] H. Gutierrez y R. Salazar, «Análisis y diseño de experimentos,» 2008. [En línea]. Available: https://gc.scalahed.com/recursos/files/r161r/w19537w/analisis_y_diseno_experimentos.pdf. [Último acceso: 25 11 2023].
- [16] A. Manasses, «Análisis del dolor crónico en pacientes adultos,» 2021. [En línea]. Available: <https://repositorio.ucsp.edu.pe/bitstreams/66e1c112-ab05-4377-a2b7-157e72ba61d5/download>. [Último acceso: 29 11 2023].
- [17] D. J. Matich, «Redes Neuronales: Conceptos Básicos y Aplicaciones,» Marzo 2001. [En línea]. Available: https://www.frro.utn.edu.ar/repositorio/catedras/quimica/5_anio/orientadora1/monografias/matich-redesneuronales.pdf. [Último acceso: 13 Febrero 2024].
- [18] M. L. Molina, «Sistema Explicable de Reconocimiento de Emociones Usando una Red Neuronal Tipo LSTM en la Base de Datos EMOMatchSpanishDB,» Junio 2023. [En línea]. Available: https://oa.upm.es/75139/1/TFG_MARTIN_LOPEZ_MOLINA.pdf. [Último acceso: 15 Diciembre 2023].

- [19] M. S. E. |. I. Chaos, «Interactivechaos.com,» 2023. [En línea]. Available: <https://interactivechaos.com/es/manual/tutorial-de-machine-learning/mean-squared-error>. [Último acceso: 03 12 2023].
- [20] Mohamad, «MAPE - Error medio de porcentaje absoluto,» 26 Octubre 2016. [En línea]. Available: <https://support.numxl.com/hc/es/articles/215959443-MAPE-Error-medio-de-porcentaje-absoluto>. [Último acceso: 31 Enero 2024].
- [21] Á. N. Menéndez, «Error en la solución numérica de ecuaciones diferenciales,» Junio 2010. [En línea]. Available: https://www.argentina.gob.ar/sites/default/files/ina-errores_en_la_solucion_numerica_ecuaciones_diferenciales.pdf. [Último acceso: 29 Diciembre 2023].
- [22] A. T. Herivia, «Los lenguajes de programación,» 2020. [En línea]. Available: https://www.academia.edu/43620885/Los_lenguajes_de_programaci%C3%B3n. [Último acceso: 20 Noviembre 2023].
- [23] T.Gomez, «Software libre, software de código abierto, licencias,» Septiembre 2015. [En línea]. Available: https://igm.univ-mlv.fr/~teresa/logicielsLIGM/documents/Internacional/2015septTGD_FS_OSS_Lic_Proc_esp.pdf. [Último acceso: 04 11 2023].
- [24] E. H. G. Oscar Arriola Nabarrete, «Uso de software de código abierto: Una alternativa para la gestión integral de la biblioteca,» 2011. [En línea]. Available: <https://core.ac.uk/download/pdf/11887625.pdf>. [Último acceso: 21 Noviembre 2023].
- [25] S. D. Quintero, «Aprende python,» 4 Diciembre 2023. [En línea]. Available: https://aprendepython.es/_downloads/907b5202c1466977a8d6bd3a2641453f/aprende-python.pdf. [Último acceso: 6 Diciembre 2023].
- [26] A. Flores, «Librerías de Python que te ayudarán a transformar el mundo digital sin complicarte,» 5 Noviembre 2021. [En línea]. Available:

- <https://www.crehana.com/blog/transformacion-digital/librerias-python/>. [Último acceso: 22 Noviembre 2023].
- [27] Omar, «El módulo sklearn.metrics.,» 15 Agosto 2023. [En línea]. Available: <https://medium.com/@bouimouass.o/the-sklearn-metrics-c568e0abcf03>. [Último acceso: 2 Febrero 2024].
- [28] B. Tychiey, «Python Plotly Express,» 21 Marzo 2023. [En línea]. Available: <https://www.datacamp.com/tutorial/python-plotly-express-tutorial#rdl>. [Último acceso: 2 Febrero 2024].
- [29] A. V. N. M. Gladys Guevara, «Metodologías de investigación educativa (descriptivas, experimentales, participativas, y de investigación-acción),» 1 Julio 2020. [En línea]. Available: <https://www.recimundo.com/index.php/es/article/view/860>. [Último acceso: 5 Diciembre 2023].
- [30] I. Sollitto, «Tesis y Masters Argentina,» 16 02 2022. [En línea]. Available: <https://tesisymasters.com.ar/investigacion-experimental-definicion/>. [Último acceso: 15 12 2023].
- [31] Qualtrics, «Gestion de la Experiencia,» 06 2020. [En línea]. Available: <https://www.qualtrics.com/es/gestion-de-la-experiencia/investigacion/investigacion-cuantitativa/>. [Último acceso: 09 12 2023].

8. ANEXOS

Anexo A: Informe antiplagio Proyecto de Titulación

Facultad:	Ciencias de la Ingeniería y Aplicadas
Carrera:	Ingeniería en Electricidad
Nombre del docente evaluador que emite el informe:	Ing. MSc. Salazar Achig Edgar Roberto
Documento evaluado:	Propuesta Investigativa presentada previo a la obtención del Título de Ingeniero Eléctrico
Programa de similitud utilizado:	Sistema COMPILATION
Porcentaje de Similitud según el programa utilizado	5%
Observaciones: Calificación de originalidad atendiendo a los siguientes criterios: <ul style="list-style-type: none"> • El documento cumple criterios de originalidad, sin observaciones. • El documento cumple criterios de originalidad, sin observaciones. • El documento cumple criterios de originalidad, sin observaciones. 	-x- --- ---
Fecha de realización del informe:	18/02/2024
Captura de pantalla del documento analizado:	
 <hr/> Ing. MSc. Salazar Achig Edgar Roberto C.C. 0502847619	

Anexo B: Resultados de Pronóstico de Generación con Regresión Lineal Simple (Diario)

MODELO DE REGRESIÓN LINEAL SIMPLE			
FECHA	POTENCIA PRE. [kW]	POTENCIA REAL [kW]	ERROR [%]
01/01/2022 0:00	299,77	295,68	1,38
01/01/2022 0:30	299,10	294,98	1,40
01/01/2022 1:00	299,10	294,98	1,40
01/01/2022 1:30	300,43	296,38	1,37
01/01/2022 2:00	300,43	296,38	1,37
01/01/2022 2:30	299,77	295,68	1,38
01/01/2022 3:00	299,77	295,68	1,38
01/01/2022 3:30	300,43	296,38	1,37
01/01/2022 4:00	300,43	296,38	1,37
01/01/2022 4:30	300,43	296,38	1,37
01/01/2022 5:00	300,43	296,38	1,37
01/01/2022 5:30	299,77	295,68	1,38
01/01/2022 6:00	299,77	295,68	1,38
01/01/2022 6:30	305,09	301,31	1,25
01/01/2022 7:00	305,09	301,31	1,25
01/01/2022 7:30	298,43	294,27	1,41
01/01/2022 8:00	298,43	294,27	1,41
01/01/2022 8:30	299,77	295,68	1,38
01/01/2022 9:00	299,77	295,68	1,38
01/01/2022 9:30	297,10	292,86	1,45
01/01/2022 10:00	297,10	292,86	1,45
01/01/2022 10:30	300,43	296,38	1,37
01/01/2022 11:00	300,43	296,38	1,37
01/01/2022 11:30	297,77	293,57	1,43
01/01/2022 12:00	297,77	293,57	1,43
01/01/2022 12:30	299,77	295,68	1,38
01/01/2022 13:00	299,77	295,68	1,38
01/01/2022 13:30	297,10	292,86	1,45
01/01/2022 14:00	297,10	292,86	1,45
01/01/2022 14:30	289,12	284,42	1,65
01/01/2022 15:00	289,12	284,42	1,65
01/01/2022 15:30	301,10	297,09	1,35
01/01/2022 16:00	301,10	297,09	1,35
01/01/2022 16:30	296,44	292,16	1,46
01/01/2022 17:00	296,44	292,16	1,46
01/01/2022 17:30	296,44	292,16	1,46
01/01/2022 18:00	296,44	292,16	1,46
01/01/2022 18:30	296,44	292,16	1,46
01/01/2022 19:00	296,44	292,16	1,46
01/01/2022 19:30	293,78	289,34	1,53
01/01/2022 20:00	293,78	289,34	1,53
01/01/2022 20:30	301,76	297,79	1,33
01/01/2022 21:00	301,76	297,79	1,33
01/01/2022 21:30	299,10	294,98	1,40
01/01/2022 22:00	299,10	294,98	1,40
01/01/2022 22:30	300,43	296,38	1,37
01/01/2022 23:00	300,43	296,38	1,37
01/01/2022 23:30	301,10	297,09	1,35

Anexo C: Resultados de Pronóstico de Generación con GRU (Diario)

MODELO DE UNIDADES RECURRENTE CERRADAS			
FECHA	POTENCIA PRE. [kW]	POTENCIA REAL [kW]	ERROR [%]
01/01/2022 0:00	295,68	297,10	0,48
01/01/2022 0:30	294,98	297,10	0,72
01/01/2022 1:00	294,98	296,21	0,42
01/01/2022 1:30	296,38	296,21	0,06
01/01/2022 2:00	296,38	297,99	0,54
01/01/2022 2:30	295,68	297,99	0,78
01/01/2022 3:00	295,68	297,10	0,48
01/01/2022 3:30	296,38	297,10	0,24
01/01/2022 4:00	296,38	297,99	0,54
01/01/2022 4:30	296,38	297,99	0,54
01/01/2022 5:00	296,38	297,99	0,54
01/01/2022 5:30	295,68	297,99	0,78
01/01/2022 6:00	295,68	297,10	0,48
01/01/2022 6:30	301,31	297,10	1,42
01/01/2022 7:00	301,31	304,21	0,95
01/01/2022 7:30	294,27	304,21	3,27
01/01/2022 8:00	294,27	295,32	0,36
01/01/2022 8:30	295,68	295,32	0,12
01/01/2022 9:00	295,68	297,10	0,48
01/01/2022 9:30	292,86	297,10	1,43
01/01/2022 10:00	292,86	293,55	0,23
01/01/2022 10:30	296,38	293,55	0,97
01/01/2022 11:00	296,38	297,99	0,54
01/01/2022 11:30	293,57	297,99	1,48
01/01/2022 12:00	293,57	294,44	0,30
01/01/2022 12:30	295,68	294,44	0,42
01/01/2022 13:00	295,68	297,10	0,48
01/01/2022 13:30	292,86	297,10	1,43
01/01/2022 14:00	292,86	293,55	0,23
01/01/2022 14:30	284,42	293,55	3,11
01/01/2022 15:00	284,42	282,90	0,53
01/01/2022 15:30	297,09	282,90	5,01
01/01/2022 16:00	297,09	298,88	0,60
01/01/2022 16:30	292,16	298,88	2,25
01/01/2022 17:00	292,16	292,66	0,17
01/01/2022 17:30	292,16	292,66	0,17
01/01/2022 18:00	292,16	292,66	0,17
01/01/2022 18:30	292,16	292,66	0,17
01/01/2022 19:00	292,16	292,66	0,17
01/01/2022 19:30	289,34	292,66	1,13
01/01/2022 20:00	289,34	289,11	0,08
01/01/2022 20:30	297,79	289,11	3,00
01/01/2022 21:00	297,79	299,77	0,66
01/01/2022 21:30	294,98	299,77	1,60
01/01/2022 22:00	294,98	296,21	0,42
01/01/2022 22:30	296,38	296,21	0,06
01/01/2022 23:00	296,38	297,99	0,54
01/01/2022 23:30	297,09	297,99	0,30

Anexo D: Resultados de Pronóstico de Generación con Redes Neuronales LSTM (Diario)

MODELO DE REDES NEURONALES LSTM			
FECHA	POTENCIA PRE. [kW]	POTENCIA REAL [kW]	ERROR [%]
01/01/2022 0:00	295,68	299,46	1,26
01/01/2022 0:30	294,98	299,46	1,50
01/01/2022 1:00	294,98	298,84	1,29
01/01/2022 1:30	296,38	298,84	0,82
01/01/2022 2:00	296,38	300,09	1,23
01/01/2022 2:30	295,68	300,09	1,47
01/01/2022 3:00	295,68	299,46	1,26
01/01/2022 3:30	296,38	299,46	1,03
01/01/2022 4:00	296,38	300,09	1,23
01/01/2022 4:30	296,38	300,09	1,23
01/01/2022 5:00	296,38	300,09	1,23
01/01/2022 5:30	295,68	300,09	1,47
01/01/2022 6:00	295,68	299,46	1,26
01/01/2022 6:30	301,31	299,46	0,62
01/01/2022 7:00	301,31	304,48	1,04
01/01/2022 7:30	294,27	304,48	3,35
01/01/2022 8:00	294,27	298,21	1,32
01/01/2022 8:30	295,68	298,21	0,85
01/01/2022 9:00	295,68	299,46	1,26
01/01/2022 9:30	292,86	299,46	2,20
01/01/2022 10:00	292,86	296,96	1,38
01/01/2022 10:30	296,38	296,96	0,19
01/01/2022 11:00	296,38	300,09	1,23
01/01/2022 11:30	293,57	300,09	2,17
01/01/2022 12:00	293,57	297,59	1,35
01/01/2022 12:30	295,68	297,59	0,64
01/01/2022 13:00	295,68	299,46	1,26
01/01/2022 13:30	292,86	299,46	2,20
01/01/2022 14:00	292,86	296,96	1,38
01/01/2022 14:30	284,42	296,96	4,22
01/01/2022 15:00	284,42	289,46	1,74
01/01/2022 15:30	297,09	289,46	2,63
01/01/2022 16:00	297,09	300,72	1,21
01/01/2022 16:30	292,16	300,72	2,85
01/01/2022 17:00	292,16	296,33	1,41
01/01/2022 17:30	292,16	296,33	1,41
01/01/2022 18:00	292,16	296,33	1,41
01/01/2022 18:30	292,16	296,33	1,41
01/01/2022 19:00	292,16	296,33	1,41
01/01/2022 19:30	289,34	296,33	2,36
01/01/2022 20:00	289,34	293,83	1,53
01/01/2022 20:30	297,79	293,83	1,35
01/01/2022 21:00	297,79	301,34	1,18
01/01/2022 21:30	294,98	301,34	2,11
01/01/2022 22:00	294,98	298,84	1,29
01/01/2022 22:30	296,38	298,84	0,82
01/01/2022 23:00	296,38	300,09	1,23
01/01/2022 23:30	297,09	300,09	1,00

Anexo E: Solicitud de Información sobre la Generación de la Mini Central Hidroeléctrica Catazacón



Ing. Diego Moscoso
Presidente Ejecutivo

Asunto: Solicitud de Información sobre Generación de la Central Hidroeléctrica Catazacón.

Mi nombre es Jordan Tandalla con CI: 0550439814 estudiante de la Universidad Técnica de Cotopaxi. Me dirijo a usted en calidad de estudiante para solicitar información relevante sobre la generación de energía en la Central Hidroeléctrica Catazacón para la ejecución de mi proyecto de titulación.

Estoy llevando a cabo el proyecto de Titulación titulado “PRONÓSTICO DE GENERACIÓN DE UNA MINI CENTRAL HIDROELÉCTRICA MEDIANTE APRENDIZAJE AUTOMÁTICO UTILIZANDO SOFTWARE DE CÓDIGO ABIERTO.” En este sentido me encuentro en la necesidad de obtener datos específicos sobre la generación de energía eléctrica en la mencionada central hidroeléctrica. Mi solicitud se enfoca en obtener información detallada sobre la capacidad de generación abarcando el periodo comprendido desde el año 2010 hasta el 2022. Este conjunto de datos será fundamental para el análisis y la validación de los modelos propuestos en mi investigación.

Agradecería mucho si pudiera proporcionarnos la información solicitada. Además, si hay algún procedimiento específico que debo seguir para acceder a estos datos, le agradecería que me lo indique.

Quedo a su disposición para cualquier aclaración adicional que pueda requerir y agradecemos de antemano su atención a esta solicitud. Espero con interés recibir la información solicitada y agradezco su colaboración en este asunto.

Atentamente:

Jordan Leonel Tandalla Cando
0550439814
0984340088
Jordan.tandalla9814@utc.edu.ec

RECIBIDO
HORA 12:13
17 NOV 2023

3

Ing. Diego Moscoso

Presidente Ejecutivo

Asunto: Solicitud de Información sobre Generación de la Central Hidroeléctrica Catazación.

Mi nombre es Jordan Tandalla con CI: 0550439814 estudiante de la Universidad Técnica de Cotopaxi. Me dirijo a usted en calidad de estudiante para solicitar información relevante sobre la generación de energía en la Central Hidroeléctrica Catazación para la ejecución de mi proyecto de titulación.

Estoy llevando a cabo el proyecto de Titulación titulado "PRONÓSTICO DE GENERACIÓN DE UNA MINI CENTRAL HIDROELÉCTRICA MEDIANTE APRENDIZAJE AUTOMÁTICO UTILIZANDO SOFTWARE DE CÓDIGO ABIERTO." En este sentido me encuentro en la necesidad de obtener datos específicos sobre la generación de energía eléctrica en la mencionada central hidroeléctrica. Mi solicitud se enfoca en obtener información detallada sobre la capacidad de generación abarcando el periodo comprendido desde el año 2010 hasta el 2022. Este conjunto de datos será fundamental para el análisis y la validación de los modelos propuestos en mi investigación.

Agradecería mucho si pudiera proporcionarnos la información solicitada. Además, si hay algún procedimiento específico que debo seguir para acceder a estos datos, le agradecería que me lo indique.

Quedo a su disposición para cualquier aclaración adicional que pueda requerir y agradecemos de antemano su atención a esta solicitud. Espero con interés recibir la información solicitada y agradezco su colaboración en este asunto.

Atentamente:



Jordan Leonel Tandalla Cando

0550439814

0984340088

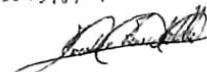
Jordan.tandalla9814@utc.edu.ec

RECIBIDO

HORA 12:13

27 NOV 2023


Tandalla Cando Jordan Leonel
0550439814


Berislav Cosic Oylan Ariel
1723908374