



**UNIVERSIDAD TÉCNICA DE COTOPAXI**  
**FACULTAD DE CIENCIAS DE LA INGENIERÍA Y APLICADAS**  
**INGENIERÍA ELÉCTRICA EN SISTEMAS ELÉCTRICOS DE POTENCIA**

**PROYECTO DE INVESTIGACIÓN**

**PREDICCIÓN DE LA GENERACIÓN DE ENERGÍA ELÉCTRICA  
PRODUCIDA EN UNA CENTRAL HIDROELÉCTRICA UTILIZANDO  
REDES NEURONALES ARTIFICIALES**

Proyecto de Titulación presentado previo a la obtención del título de Ingeniero Eléctrico  
en Sistemas Eléctricos de Potencia

**Autor/es**

González Piedra Joel André

Hidalgo Barrera Christopher Alexander

**Tutor Académico**

Ing. MSc. Salazar Achig Edgar Roberto

**LATACUNGA - ECUADOR**

**Marzo - 2022**



## DECLARACIÓN DE AUTORÍA

Nosotros **González Piedra Joel André e Hidalgo Barrera Christopher Alexander** declaramos ser autores del presente proyecto de titulación: **“PREDICCIÓN DE LA GENERACIÓN DE ENERGÍA ELÉCTRICA PRODUCIDA EN UNA CENTRAL HIDROELÉCTRICA UTILIZANDO REDES NEURONALES ARTIFICIALES”** siendo el **Ing. MSc. Salazar Achig Edgar Roberto** tutor del presente trabajo; y eximimos expresamente a la Universidad Técnica de Cotopaxi y a sus representantes legales de posibles reclamos o acciones legales.

Además, certificamos que las ideas, conceptos, procedimientos y resultados vertidos en el presente trabajo de titulación, son de nuestra exclusiva responsabilidad.

.....  
González Piedra Joel André  
C.C. 1600488256

.....  
Hidalgo Barrera Christopher Alexander  
C.C. 1600839490



## AVAL DEL TUTOR DE PROYECTO DE TITULACIÓN

En calidad de Tutor del Trabajo de Investigación sobre el título: “ **PREDICCIÓN DE LA GENERACIÓN DE ENERGÍA ELÉCTRICA PRODUCIDA EN UNA CENTRAL HIDROELÉCTRICA UTILIZANDO REDES NEURONALES ARTIFICIALES** ”, de **González Piedra Joel André e Hidalgo Barrera Christopher Alexander**, de la carrera de **Ingeniería Eléctrica**, 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 del 2022

Ing. MSc. Salazar Achig Edgar Roberto

CC. 0502847619



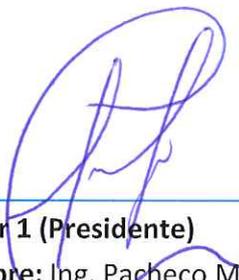
### APROBACIÓN DEL TRIBUNAL DE TITULACIÓN

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, el o los postulantes: **González Piedra Joel Andre e Hidalgo Barrera Christopher Alexander** con el título de Proyecto de titulación: **PREDICCIÓN DE LA GENERACIÓN DE ENERGÍA ELÉCTRICA PRODUCIDA EN UNA CENTRAL HIDROELÉCTRICA UTILIZANDO REDES NEURONALES ARTIFICIALES** 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 del 2022

Para constancia firman:

  
\_\_\_\_\_  
**Lector 1 (Presidente)**  
**Nombre:** Ing. Pacheco Mena Carlos Francisco  
**CC:** 0503072902

  
\_\_\_\_\_  
**Lector 2**  
**Nombre:** Ing. Barbosa Galarza Efren Jose  
**CC:** 0501420723

  
\_\_\_\_\_  
**Lector 3**  
**Nombre:** Ing. Quinatoa Caiza Carlos Ivan  
**CC:** 0503287864

## **AGRADECIMIENTO**

*Agradezco a Dios por bendecirnos la vida, por guiarnos a lo largo de nuestra existencia, ser el apoyo y fortaleza en aquellos momentos de dificultad y de debilidad.*

*A mis padres Rubén González y Carmita Piedra y hermano Bryan González, por su amor, trabajo y sacrificio en todos estos años, gracias a ustedes hemos logrado llegar hasta aquí y convertirnos en lo que somos. Ha sido el orgullo y el privilegio de ser su hijo, son los mejores padres.*

*A mi tutor Ing. MSc. Roberto Salazar por la ayuda incondicional que siempre me ha prestado, así como su completa confianza y el haber sabido guiar mi trabajo en los momentos más delicados.*

*A mis compañeros Christopher y Mauricio que con sus conocimientos hicieron excelentes aportes a este proyecto de grado.*

*A mi enamorada Jessica Ordoñez, por entenderme en todo momento, gracias en cada instante que fue un apoyo incondicional, por impulsarme que siga cada día adelante con mis estudios y sobre todo por tu paciencia y amor incondicional.*

*Joel André González Piedra*

## **AGRADECIMIENTO**

*Le agradezco a Dios por guiarme y estar a mi lado a lo largo de mi carrera, por ser mi luz y fortaleza en momentos de debilidad y por darme una vida colmada de aprendizajes, experiencias y mucha prosperidad.*

*Le doy gracias a mi padre Wilson, a mi madre Jaqueline y a mi hermana Sabine por apoyarme en todo momento, por los valores que me han enseñado, y por concederme una educación excelente en el transcurso de mi vida, gracias a ellos por cada día confiar y creer en mí y en mis expectativas. A mi querida madre, mi gran inspiración para ser mejor cada día, ha sido mi ejemplo a seguir, su ánimo, sus enseñanzas y amor diario me guiaron durante todo este camino; gracias a mi padre por siempre desear y anhelar siempre lo mejor para mi vida, gracias por cada consejo y por cada una de sus palabras que me guiaron durante mi vida.*

*A mi tutor Ing. MSc. Roberto Salazar por la ayuda incondicional que siempre me ha prestado, así como su completa confianza y el haber sabido guiar mi trabajo en los momentos más delicados.*

*A mis compañeros André y Mauricio que con sus conocimientos hicieron valiosos aportes a este proyecto de grado.*

*A ti Andrea, por ser alguien importante en mi vida, por haberme apoyado en los días buenos y malos, brindarme mucha felicidad y sobre todo por tu paciencia y cariño incondicional.*

*Christopher Alexander Hidalgo Barrera*

## **DEDICATORIA**

*Dedico este trabajo ante todo a Dios, por permitir haber llegado hasta este momento tan importante en mi carrera profesional*

*También con todo mi amor y cariño a mis padres y hermano, ya que no solo les debo mis estudios sino la vida entera ustedes son y serán mi fortaleza.*

*En especial a mi enamorada Jessica Ordoñez, por su comprensión, por toda la ayuda y el apoyo que me ha brindado durante el tiempo que hemos compartido juntos, y aún después.*

*A mis amigos Alex, Bryan, Christopher, David, Dennis, Fabian, Fausto, Jordín, Juan, Kevin, Mauricio, Paúl, por sus cuidados, su paciencia y apoyo constante. Por haberme ayudado y animado en los momentos más duros.*

*Joel André González Piedra*

## **DEDICATORIA**

*Mi tesis la dedico principalmente a mis padres quienes fueron los que me brindaron su apoyo en días buenos y malos, me alentaron a seguir adelante y a luchar por mis metas es por ellos que estoy aquí.*

*A mis amigos André, David, Fabian, Jordin, Juan, Kevin, Mauricio, los grandes “Conejos Crew” por sus cuidados, su paciencia y apoyo constante. Por haberme ayudado y animado en los momentos más duros. En especial Alex, por su comprensión, por toda la ayuda y el apoyo que me ha brindado durante el tiempo que hemos compartido juntos, y aún después.*

*Christopher Alexander Hidalgo Barrera*

## ÍNDICE GENERAL

DECLARACIÓN DE AUTORÍA .....	i
AVAL DEL TUTOR DEL PROYECTO DE INVESTIGACIÓN.....	ii
APROBACIÓN DEL TRIBUNAL DE TITULACIÓN .....	iii
AGRADECIMIENTO.....	iv
DEDICATORIA .....	vi
ÍNDICE GENERAL.....	viii
ÍNDICE DE FIGURAS.....	xii
ÍNDICE DE TABLAS .....	xiv
RESUMEN.....	xv
ABSTRACT .....	xvi
AVAL DE TRADUCCIÓN.....	xvii
1. INFORMACIÓN GENERAL .....	1
2. INTRODUCCIÓN .....	2
2.1. EL PROBLEMA .....	3
2.1.1. Situación Problemática.....	3
2.1.2. Formulación del problema .....	3
2.2. OBJETO Y CAMPO DE ACCIÓN .....	3
2.2.1. Objeto de investigación.....	3
2.2.2. El campo de acción para el proyecto.....	3
2.3. BENEFICIARIOS .....	4
2.3.1. Beneficiarios directos.....	4
2.3.2. Beneficiarios indirectos.....	4
2.4. JUSTIFICACIÓN .....	4
2.5. HIPÓTESIS.....	4
2.6. OBJETIVOS .....	5
2.6.1. General .....	5
2.6.2. Específicos .....	5
2.7. SISTEMA DE TAREAS .....	5
3. FUNDAMENTACIÓN TEÓRICA.....	8
3.1. ANTECEDENTES.....	8
3.2. CENTRAL HIDROELÉCTRICA ILLUCHI 1 .....	9
3.3. LA CENTRAL ILLUCHI 1 ES UNA CENTRAL “DE DERIVACIÓN” .....	11
3.4. RELACIÓN CAUDAL Y ENERGÍA PRODUCIDA .....	12
3.4.1. Cálculo del caudal .....	13
3.5. ELEMENTOS ESENCIALES DE UNA CENTRAL HIDROELÉCTRICA .....	13
3.5.1. Área de captación.....	14

3.5.2. Embalse.....	14
3.5.3. Presa.....	14
3.5.4. Aliviaderos .....	14
3.5.5. Tanques de compensación.....	14
3.5.6. Geometría del tanque mediante fotogrametría .....	15
3.6. PROCESO DE MEDICIÓN DE LOS TANQUES MEDIANTE FOTOGRAMETRÍA .....	17
3.7. ALGORITMOS QUE SE USAN PARA PREDECIR LA ENERGÍA ELÉCTRICA SEGÚN DATOS DE ENTRADA Y SALIDA.....	19
3.7.1. Algoritmos de entrenamiento .....	19
3.7.2. Redes neuronales artificiales .....	19
3.7.3. Ventajas de las redes neuronales artificiales .....	20
3.7.4. Desventajas de las redes neuronales artificiales.....	20
3.7.5. Elementos de una red neuronal artificial.....	20
3.7.6. Red neuronal LSTM (Long Short-Term Memory) .....	25
3.7.7. Importancia del pronóstico de la red neuronal artificial.....	26
3.7.8. Función de activación de la red neuronal artificial .....	26
3.8. IDENTIFICAR EL MACHINE LEARNING Y EL DEEP LEARNING .....	27
3.9. MODELO MATEMÁTICO DE UNA RED NEURONAL ARTIFICIAL.....	27
3.9.1. Fase de aprendizaje o entrenamiento de la red neuronal artificial .....	28
3.9.2. Aprendizaje por corrección de error.....	28
3.9.3. Aprendizaje por refuerzo.....	29
3.9.4. Aprendizaje no supervisado .....	29
3.10. MÉTODOS PARA PREDECIR LA ENERGÍA ELÉCTRICA .....	30
3.10.1. Métodos de Tendencia simple.....	30
3.10.2. Método de Suavizamiento exponencial y promedios móviles .....	30
3.10.3. Método ARIMA .....	31
3.10.4. Método Redes Neuronales Artificiales.....	32
3.10.5. Método de Redes neuronales aplicadas a series temporales .....	35
3.11. FACTORES QUE AFECTAN LOS PRONÓSTICOS DE GENERACIÓN DE ENERGÍA .....	36
3.12. ARQUITECTURA DE REDES NEURONALES .....	36
3.12.1. Arquitectura de una FANN (Feedforward Artificial Neuronal Network).....	36
3.13. ALGORITMO DE APRENDIZAJE PARA REDES NEURONALES ARTIFICIALES ..	37
3.14. REDES NEURONALES PROBABILÍSTICAS.....	40
3.15. SOFTWARES PARA EL DESARROLLO DE REDES NEURONALES ARTIFICIALES ...	41
3.16. LENGUAJE DE PROGRAMACIÓN PYTHON.....	41
3.16.1. Librerías necesarias para el uso del software de Python .....	42
3.16.2. Paquetes especiales para el desarrollo de redes neuronales .....	43
3.17. OPTIMIZADORES DE TENSORFLOW.....	44

3.18. PROBLEMAS DERIVADOS DEL ENTRENAMIENTO DE LA RED NEURONAL ...	44
3.18.1. El problema de la convergencia prematura .....	44
3.18.2. El problema de la permutación.....	45
3.18.3. Underfitting .....	46
3.18.4. Overfitting .....	46
3.18.5. Normalfitting.....	47
3.19. COMPORTAMIENTO DE LA COMPUTADORA AL ENTRENAR LA RED NEURONAL.....	47
3.19.1. Características del computador utilizado .....	47
3.19.2. Rendimiento del CPU.....	48
4. MATERIALES Y MÉTODOS .....	50
4.1. TIPO DE INVESTIGACIÓN.....	50
4.1.1. Investigación descriptiva.....	50
4.1.2. Investigación diagnóstica .....	50
4.1.3. Investigación aplicada .....	50
4.2. MÉTODO DE INVESTIGACIÓN .....	50
4.2.1. Metodología de la investigación bibliográfica. ....	50
4.2.2. Inductivo – deductivo.....	51
4.3. TÉCNICAS E INSTRUMENTOS .....	51
4.3.1. Observación de la red neuronal.....	51
4.3.2. Simulación de la red neuronal .....	51
4.3.3. Programación de la red neuronal.....	51
4.4. DIAGRAMA DE FLUJO PARA EL DESARROLLO DE LA RED NEURONAL .....	52
5. ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS.....	55
5.1. COMPRENDER EL COMPORTAMIENTO DE LA GENERACIÓN EN LA CENTRAL HIDROELÉCTRICA ILLUCHI 1 .....	55
5.2. ANÁLISIS PARA LA CORRELACIÓN DE LAS VARIABLES .....	55
5.3. PREPROCESAMIENTO DE LA INFORMACIÓN .....	57
5.3.1. Adquisición y almacenamiento de la información .....	57
5.4. PREPROCESAMIENTO DE LOS DATOS.....	57
5.4.1. Análisis de la base de datos históricos de los generadores.....	58
5.5. SELECCIÓN DE VARIABLES DE ENTRADA.....	61
5.6. VALIDACIÓN DE LOS DATOS HISTÓRICOS .....	62
5.7. IMPLEMENTACIÓN EN PYTHON .....	62
5.7.1. Librerías utilizadas .....	62
5.7.2. Procesamiento de datos .....	62
5.7.3. Implementación de red LSTM .....	65
5.7.4. Entrenamiento de la red neuronal.....	66

5.7.5. Verificación de la red neuronal artificial.....	66
5.8. VALIDAR LA RED NEURONAL ARTIFICIAL .....	67
5.9. ALMACENAMIENTO DE LOS PARÁMETROS DE LA RED NEURONAL ARTIFICIAL.....	68
5.10. PREDICCIÓN DE LA GENERACIÓN ELÉCTRICA .....	69
5.10.1. Comparación de los algoritmos de entrenamiento .....	69
5.10.2. Comparación con el número de neuronas .....	70
5.10.3. Comparación con el número de épocas.....	71
5.11. ANÁLISIS DEL COMPORTAMIENTO DE LA BASE DE DATOS REAL .....	71
5.12. ANÁLISIS DEL COMPORTAMIENTO DEL CAUDAL PRONOSTICADO .....	75
5.13. COMPARACIÓN DE DATOS REALES Y PRONÓSTICO DE LA RED NEURONAL	78
5.14. DESARROLLO DE LA INTERFAZ GRÁFICA .....	83
6. CONCLUSIONES .....	90
7. RECOMENDACIONES .....	91
8. BIBLIOGRAFÍA.....	92
9. ANEXOS.....	94

## ÍNDICE DE FIGURAS

Figura 1: Ubicación de la casa de máquinas central hidroeléctrica Illuchi 1 .....	9
Figura 2: Ubicación del dique de la central hidroeléctrica Illuchi 1 .....	10
Figura 3: Partes principales de la central De Derivación .....	12
Figura 4: Diagrama de flujo de una central hidroeléctrica típica .....	13
Figura 5: Tanque de compensación a nivel del suelo .....	15
Figura 6: Fotogrametría terrestre.....	16
Figura 7: Fotogrametría aérea .....	17
Figura 8: Repositorio del lado derecho .....	17
Figura 9: Repositorio del lado izquierdo.....	18
Figura 10: Repositorio del tanque central .....	18
Figura 11: Neurona Biológica .....	21
Figura 12: Relación entre la neuronal artificial y biológica .....	22
Figura 13: Estructura de una neurona artificial y una biológica .....	23
Figura 14: Comparación entre una red artificial y biológica.....	23
Figura 15: Capas de la red neuronal.....	24
Figura 16: Funciones de activación .....	26
Figura 17: Machine Learning y el Deep Learning .....	27
Figura 18: Unidades de la red neuronal artificial.....	27
Figura 19: Estructura de modelo supervisado .....	28
Figura 20: Estructura de aprendizaje por refuerzo .....	29
Figura 21: Estructura aprendizaje no supervisado .....	29
Figura 22: Ejemplo del método de tendencia simple .....	30
Figura 23: Ejemplo de método de suavizamiento exponencial simple.....	31
Figura 24: Ejemplo del método de ARIMA.....	32
Figura 25: Ejemplo de método de redes neuronales artificiales .....	34
Figura 26: Estructura de la red neuronal .....	35
Figura 27: Modelo de una red neuronal artificial feedforward .....	36
Figura 28: Softwares para el desarrollo de redes neuronales artificiales .....	41
Figura 29: Optimizadores de tensorflow.....	44
Figura 30: El problema de la permutación .....	45
Figura 31: Equilibrio de aprendizaje en underfitting .....	46
Figura 32: Equilibrio de aprendizaje en overfitting .....	47
Figura 33: Equilibrio de aprendizaje en normalfitting .....	47
Figura 34: Rendimiento del CPU con epoch de 3 .....	48
Figura 35: Rendimiento del CPU con epoch de 10.....	48
Figura 36: Rendimiento del CPU con epoch de 50.....	49
Figura 37: Diagrama de flujo del desarrollo de la RNA .....	52
Figura 38: Consumo de la energía mensual del 2015 al 2019.....	55
Figura 39: Diagrama de calor de relación de las variables.....	56
Figura 40: Correlación de las variables .....	56
Figura 41: Base de datos de los Generadores Illuchi 1 .....	57
Figura 42: Diagrama del preprocesamiento de datos históricos .....	57
Figura 43: Datos históricos preprocesados.....	58
Figura 44: Datos históricos preprocesados incluido los generadores.....	58
Figura 45: Ecuación del generador 1 .....	59
Figura 46: Ecuación del generador 2.....	60

Figura 47: Ecuación del generador 3 .....	60
Figura 48: Ecuación del generador 4 .....	61
Figura 49: Librerías de Python.....	62
Figura 50: Llamado de datos preprocesados .....	63
Figura 51: Verificación de valores nulos .....	63
Figura 52: Valor correcto de la fecha y el caudal .....	64
Figura 53: División de los datos .....	64
Figura 54: Normalización de los datos .....	64
Figura 55: Agrupamiento de datos .....	65
Figura 56: Reshape de los datos.....	65
Figura 57: Implementación de red LSTM .....	65
Figura 58: Estructura de la red neuronal artificial .....	66
Figura 59: Optimizadores seleccionados para el entrenamiento de la RNA .....	66
Figura 60: Entrenamiento con el optimizador RMSPROP .....	67
Figura 61: Comparación del caudal real con el pronosticado .....	68
Figura 62: Modelos guardados en formato h5 .....	68
Figura 63: Curva de los diferentes algoritmos de entrenamiento .....	69
Figura 64: Curva de los diferentes números de neuronas .....	70
Figura 65: Curva de los diferentes números de épocas .....	71
Figura 66: Comportamiento del caudal real por día .....	72
Figura 67: Curva del caudal diaria en horas .....	74
Figura 68: Comportamiento del caudal precedido por día .....	75
Figura 69: Curva del caudal diaria en horas .....	77
Figura 70: Curva del caudal real vs el pronosticado .....	78
Figura 71: Gráfica de error entre el caudal real y el pronosticado .....	79
Figura 72: Intervalos de caudal semanal real.....	79
Figura 73: Intervalos de caudal semanal pronosticado .....	80
Figura 74: Curva del caudal real vs el pronosticado diario .....	80
Figura 75: Gráfica de error entre el caudal real y el pronosticado .....	81
Figura 76: Intervalos de caudal diario real.....	82
Figura 77: Intervalos de caudal diario pronosticado.....	82
Figura 78: Datos históricos reales .....	83
Figura 79: Conversión de potencia a caudal .....	84
Figura 80: Gráfica del caudal real .....	85
Figura 81: Entrenamiento de la red .....	86
Figura 82: Gráfica del caudal pronosticado .....	87
Figura 83: Tabla de resultados de la predicción.....	88
Figura 84: Comparación del volumen del tanque con el volumen pronosticado .....	89

## ÍNDICE DE TABLAS

Tabla 1: Sistema de tareas en relación con el objetivo específico 1 .....	5
Tabla 2: Sistema de tareas en relación con el objetivo específico 2 .....	6
Tabla 3: Sistema de tareas en relación con el objetivo específico 3 .....	6
Tabla 4: Sistema de tareas en relación con el objetivo específico 4 .....	7
Tabla 5: Coordenadas del dique de captación y de las casas de máquinas .....	9
Tabla 6: Equipos de la central hidroeléctrica Illuchi 1 .....	11
Tabla 7: Datos Área de Interés del lado derecho.....	17
Tabla 8: Datos Área de Interés Tanque del lado izquierdo .....	18
Tabla 9: Datos Área de Interés Tanque Central .....	18
Tabla 10: Factores que influyen en el comportamiento de la generación eléctrica.....	36
Tabla 11: Potencia y Caudal de los 4 generadores .....	59
Tabla 12: Variables de entrada de los datos seleccionados .....	61
Tabla 13: Porcentaje de error entre el caudal real y pronosticado .....	67
Tabla 14: Error absoluto de los algoritmos de entrenamiento.....	69
Tabla 15: Error absolute con el número de neuronas .....	70
Tabla 16: Error absoluto con el número de épocas .....	71
Tabla 17: Análisis descriptiva con datos reales del comportamiento del caudal .....	73
Tabla 18: Análisis descriptiva con datos reales del comportamiento del caudal .....	74
Tabla 19: Análisis descriptiva con datos pronosticados del comportamiento del caudal .....	76
Tabla 20: Análisis descriptiva con datos pronosticados del comportamiento del caudal .....	77
Tabla 21: Error promedio entre el caudal real y el pronosticado semanal .....	78
Tabla 22: Error promedio entre el caudal real y el pronosticado diario .....	81



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

**Título: “PREDICCIÓN DE LA GENERACIÓN DE ENERGÍA ELÉCTRICA PRODUCIDA EN UNA CENTRAL HIDROELÉCTRICA UTILIZANDO REDES NEURONALES ARTIFICIALES”**

**Autor/es:**

González Piedra Joel André

Hidalgo Barrera Christopher Alexander

### **RESUMEN**

La aplicación de las redes neuronales artificiales se ejecutó debido a la existencia de pérdidas económicas en los sistemas de distribución, provocado por el inconveniente que existe en la predicción de la generación de energía eléctrica. La presente investigación tiene como objetivo predecir la generación de energía eléctrica en la central hidroeléctrica Illuchi 1, utilizando redes neuronales artificiales mediante algoritmos de entrenamiento de la librería Keras de Python. Para esto se utilizó datos históricos (2 015 – 2 019) de caudal y potencia activa de la central hidroeléctrica Illuchi 1, proporcionados por ELEPCO SA, los cuales se dividieron, transformaron y agruparon para su posterior entrenamiento y prueba. La red neuronal artificial se estructuró por una capa de entrada con 10 neuronas, dos capas ocultas, dos capas de congelamiento y una capa de salida, para el entrenamiento de los modelos se desarrolló una plataforma de software de Python, con el uso de algoritmos de entrenamiento de la librería Keras y para la validación se utilizó el optimizador RMSPROP con 10 números de retrasos y 10 neuronas. Cabe mencionar que al predecir energía eléctrica se contribuye al Plan Nacional de Desarrollo “Toda una Vida” para la optimización de la utilización de los recursos naturales.

**Palabras clave**— Red neuronal, predicción, energía eléctrica, algoritmos, keras.



**TECHNICAL UNIVERSITY OF COTOPAXI  
FACULTY OF ENGINEERING AND APPLIED SCIENCES**

**Title: "PREDICTION OF ELECTRICAL ENERGY GENERATION PRODUCED IN A HYDROELECTRIC POWER PLANT USING ARTIFICIAL NEURAL NETWORKS"**

**Author/s:**

González Piedra Joel André

Hidalgo Barrera Christopher Alexander

**ABSTRACT**

The application of artificial neural networks was executed due to the existence of economic losses in the distribution systems, caused by the inconvenience that exists in the prediction of the generation of electrical energy. The objective of this research is to predict the generation of electrical energy in the Illuchi 1 hydroelectric plant, using artificial neural networks through training algorithms from the Keras Python library. For this, historical data (2015 - 2019) of the flow and active power of the Illuchi 1 hydroelectric plant were obtained, provided by ELEPCO SA, which were divided, transformed and grouped for subsequent training and testing. The artificial neural network was structured by an input layer with 10 neurons, two hidden layers, two freezing layers and an output layer, for the training of the models a Python software platform was started, with the use of algorithms of training of the Keras library and for validation the optimizer RMSPROP was achieved with 10 numbers of delays and 10 neurons. It should be mentioned that when predicting electrical energy, it contributes to the National Development Plan "Toda una Vida" for the optimization of the use of natural resources.

**Keywords**— Neural network, prediction, electric energy, algorithms, keras.

## *AVAL DE TRADUCCIÓN*

En calidad de Docente del Idioma Inglés del Centro de Idiomas de la Universidad Técnica de Cotopaxi; en forma legal CERTIFICO que:

La traducción del resumen al idioma Inglés del artículo científico cuyo título versa: **“PREDICCIÓN DE LA GENERACIÓN DE ENERGÍA ELÉCTRICA PRODUCIDA EN UNA CENTRAL HIDROELÉCTRICA UTILIZANDO REDES NEURONALES ARTIFICIALES.”** presentado por: **González Piedra Joel André e Hidalgo Barrera Christopher Alexander**, estudiantes de la Carrera de **Ingeniería Eléctrica** perteneciente a la **Facultad de Ciencias de la Ingeniería y Aplicadas** lo realizaron bajo mi supervisión y cumple con una correcta estructura gramatical del Idioma.

Es todo cuanto puedo certificar en honor a la verdad y autorizo a los peticionarios hacer uso del presente aval para los fines académicos legales.

Latacunga, 21 marzo del 2022

Atentamente,



CENTRO  
DE IDIOMAS

Mg. Marco Paúl Beltrán Semblantes

**DOCENTE CENTRO DE IDIOMAS-UTC**  
**CI: 0502666514**

## 1. INFORMACIÓN GENERAL

**Título:** Predicción de la generación de energía eléctrica producida en una central hidroeléctrica utilizando redes neuronales artificiales.

**Fecha de inicio:** Octubre 2021

**Fecha de finalización:** Marzo 2022

**Lugar de ejecución:** Central Hidroeléctrica Illuchi 1 Provincia de Cotopaxi

**Facultad que auspicia:** Facultad de Ciencias de la Ingeniería y Aplicadas.

**Carrera que auspicia:** Carrera de Ingeniería Eléctrica.

**Proyecto macro asociado:** Desarrollo de sistemas para el abastecimiento y uso del servicio de energía eléctrica a nivel local, regional o nacional.

### **Equipo de trabajo**

Tutor: Ing. MSc. Salazar Achig Edgar Roberto

Estudiantes: González Piedra Joel André

Hidalgo Barrera Christopher Alexander

Grupo de investigación: Sistemas Eléctricos de Potencia

**Tutor de titulación:** Ing. MSc. Salazar Achig Edgar Roberto

**Estudiantes:** González Piedra Joel André

Hidalgo Barrera Christopher Alexander

**Área de conocimiento:** 07 Ingeniería, Industria y Construcción / 071 Ingeniería y Profesionales Afines / 0713 Electricidad y Energía.

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

**Sublínea de investigación:** Explotación y Diseño en Sistemas Eléctricos de Potencia.

## 2. INTRODUCCIÓN

Los errores en la predicción de la energía eléctrica pueden afectar de distintas formas al sector eléctrico como en el precio de operación, flujo de la carga, análisis de seguridad, planes de mantenimiento preventivo. Por lo cual el pronóstico de generación de energía es una función importante para el funcionamiento fiable y económico del sistema.

El proyecto investigativo plantea llevar a cabo la predicción de la energía eléctrica, aplicando redes neuronales artificiales, con datos históricos reales los cuales pertenecen a la central hidroeléctrica Illuchi 1 ubicada en la provincia de Cotopaxi.

Continuando con el desarrollo de una herramienta computacional, que basa la relación implícita del caudal mediante el uso de RNA utilizando un algoritmo computacional, permitiendo realizar un análisis del comportamiento de la energía producida por la central.

A continuación, en el punto dos se revisará el problema, la situación problemática, formulación del problema, el objeto y el campo de acción, beneficiarios, justificación, hipótesis, objetivos, sistema de tareas. En el punto tres se verá la fundamentación teórica. En el punto cuatro se verán los materiales y métodos, mientras que en el punto cinco se realizará un análisis y una discusión de resultados, terminando con el punto seis y siete con las conclusiones y recomendaciones.

## **2.1. EL PROBLEMA**

### **2.1.1. Situación Problémica**

Para realizar predicciones usualmente se usan simulaciones en el dominio del tiempo más criterios apropiados, que tienen las ventajas de un cálculo preciso. Sin embargo, el modelo contiene ecuaciones algebraicas diferenciales no lineales, que son complicadas y lleva mucho tiempo calcularlas, lo cual es difícil de cumplir con los requisitos del cálculo en línea. Los algoritmos de inteligencia artificial pueden establecer la relación del caudal con la producción de energía mediante el mapeo entre la entrada y salida de datos a través del aprendizaje y la velocidad de cálculo es rápida, obteniendo así una predicción de cuál será el caudal necesario para la producción de energía en una central hidroeléctrica.

Al existir el problema de predicción de energía eléctrica se producen pérdidas económicas en el sistema de distribución, por lo que se decide aplicar las redes neuronales artificiales como modelo de aproximación de funciones. Para poder lograr que la red neuronal pueda realizar estas funciones debe ser entrenada mediante el proceso de pruebas y errores, logrando un modelo que brinda resultados apropiados.

### **2.1.2. Formulación del problema**

La necesidad de estimar la producción de energía en la central hidroeléctrica permitirá la aplicación de redes neuronales artificiales que prediga el comportamiento por hora del caudal para la generación.

## **2.2. OBJETO Y CAMPO DE ACCIÓN**

### **2.2.1. Objeto de investigación**

Predicción de la generación de energía eléctrica en una central hidroeléctrica ubicada en la Provincia de Cotopaxi.

### **2.2.2. El campo de acción para el proyecto**

Aplicación del software PYTHON utilizando redes neuronales artificiales para la predicción de la generación de energía eléctrica de forma óptima.

330000 Ciencias Tecnológicas / 3306 Ingeniería y Tecnología Eléctricas / 3306.09 Transmisión y Distribución.

## **2.3. BENEFICIARIOS**

### **2.3.1. Beneficiarios directos**

Los beneficiarios directos del presente proyecto de investigación son, la empresa de distribución Illuchi 1, debido a que esto permitirá predecir la producción de energía eléctrica mejorando la confiabilidad del sistema y autores del presente trabajo, dado que esta investigación permitirá alcanzar el título universitario.

### **2.3.2. Beneficiarios indirectos**

Los consumidores de la energía eléctrica aportada por la hidroeléctrica Illuchi I.

## **2.4. JUSTIFICACIÓN**

En el país las empresas de electricidad deben enviar al CENACE la programación de energía, los datos enviados corresponden a la curva de generación prevista para el siguiente día, debe ser enviada hasta las 10 de la mañana, esta obligación de optimizar la producción hace necesario que se desarrollen modelos de predicción de la energía eléctrica precisos, lo que da lugar a un mejor ajuste de la producción y, como consecuencia, a una reducción del riesgo de que se produzca un apagón en la central.

Debido a la inconveniencia que existe en la predicción de la energía eléctrica, es necesario realizar actividades para optimizar la generación de energía. Teniendo en cuenta esta razón, se realiza el desarrollo de redes neuronales con el uso de herramientas de programación accesibles (como el software PYTHON) necesarias para desarrollar algoritmos para resolver el pronóstico de la energía eléctrica.

Actualmente, la empresa de distribución de energía eléctrica no cuenta con un análisis predictivo para conocer cuál es la cantidad de energía que puede producir a partir de su caudal, por lo tanto, es necesario este estudio para ayudar a predecir cuanta energía la central puede inyectar al sistema en las diferentes horas del día. Para lo cual se utilizará los datos históricos y mediciones de campo permitiendo desarrollar redes neuronales artificiales dando solución a este tipo de problemática encontrada.

## **2.5. HIPÓTESIS**

¿El desarrollo de redes neuronales artificiales en Python permitirá la predicción de la generación de energía eléctrica a partir del caudal?

## 2.6. OBJETIVOS

### 2.6.1. General

Predecir la generación de energía eléctrica producida en la central hidroeléctrica Illuchi 1 utilizando redes neuronales artificiales.

### 2.6.2. Específicos

- Revisar el estado del arte referente a parámetros de interés para el entrenamiento y testeo de la red neuronal en la producción de energía eléctrica.
- Construir la red neuronal artificial utilizando los parámetros de producción de energía.
- Entrenar la red neuronal para determinar la producción de energía eléctrica en la central hidroeléctrica Illuchi 1.
- Validar la red neuronal artificial.

## 2.7. SISTEMA DE TAREAS

Tabla 1: Sistema de tareas en relación con el objetivo específico 1

<b>Objetivo específico 1</b>	<b>Actividad (tareas)</b>	<b>Resultados esperados</b>	<b>Técnicas, medio e instrumentos</b>
Revisar el estado del arte referente a parámetros de interés para el entrenamiento y testeo de la red neuronal en la producción de energía eléctrica.	Revisar el estado del arte referente a redes neuronales	Fuentes de información sobre redes neuronales	Libros, papers, páginas web
	Revisar el estado del arte sobre las variables para el entrenamiento de la red neuronal	Tipos de variables a utilizar para el entrenamiento de la red neuronal	Libros, papers, páginas web
	Redacción	Archivo que contiene toda la información encontrada para su posterior revisión	Formato otorgado por el tutor

Tabla 2: Sistema de tareas en relación con el objetivo específico 2

<b>Objetivo específico 2</b>	<b>Actividad (tareas)</b>	<b>Resultados esperados</b>	<b>Técnicas, medio e instrumentos</b>
Construir la red neuronal artificial utilizando los parámetros de producción de energía.	Revisión de los históricos y obtención de los datos de campo	Determinación de la capacidad de almacenamiento de los tanques e información sobre la construcción de la red neuronal	Libros, papers, páginas web y tanques de almacenamiento de la central
	Aplicación de los históricos en la red neuronal	Pasos para construir la red neuronal	Datos y variables obtenidos
	Construcción de la red neuronal en el software Python	Archivo ejecutable en formato .ipynb para el software de Python	Software Python

Tabla 3: Sistema de tareas en relación con el objetivo específico 3

<b>Objetivo específico 3</b>	<b>Actividad (tareas)</b>	<b>Resultados esperados</b>	<b>Técnicas, medio e instrumentos</b>
Entrenar la red neuronal para determinar la producción de energía eléctrica en una central hidroeléctrica.	Aplicar los datos de entrada	Dimensionamiento de la red neuronal	Datos y variables obtenidos
	Aplicar los datos de salida	Producción de energía a partir del caudal	Datos y variables obtenidos
	Realizar pruebas de aprendizaje	Mejor resultado de predicción	Intercambio de datos y variables

Tabla 4: Sistema de tareas en relación con el objetivo específico 4

<b>Objetivo específico 4</b>	<b>Actividad (tareas)</b>	<b>Resultados esperados</b>	<b>Técnicas, medio e instrumentos</b>
Validar la red neuronal artificial.	Ejecutar una prueba piloto de la red neuronal	Muestra de posibles fallas en los datos	Datos y variables obtenidos
	Limpiar los datos recopilados	Mejores resultados en los datos	Algoritmos de programación
	Validar datos	Red neuronal terminada	Datos y variables obtenidos

### **3. FUNDAMENTACIÓN TEÓRICA**

#### **3.1. ANTECEDENTES**

Para decidir entre uno u otros modelos para el entrenamiento de las redes neuronales se procedió a revisar documentos científicos los cuales se muestran a continuación.

Según el autor [1] Modelo de predicción de la producción de energía de la Central Hidroeléctrica Coca Codo Sinclair, basado en técnicas de aprendizaje computacional. Previa a la obtención del título en Gestión de Sistemas de Información e Inteligencia de Negocios en la Universidad de las Fuerzas Armadas. “Los investigadores indican que la determinación de la cantidad de energía que se puede producir en la central hidroeléctrica a través de la predicción del caudal que provee los recursos hídricos a la central. Para lograr este objetivo, se utilizan registros de datos históricos del caudal promedio. Los datos históricos se estructuraron mediante una serie temporal y sirvieron como base para la generación y entrenamiento de los modelos construidos usando Redes Neuronales.”

Según el autor [2] Pronóstico de producción hidroeléctrica del Ecuador usando redes neuronales. Previa a la obtención del título de Ingeniero en Sistemas Computacionales en la Universidad de Guayaquil “La predicción de la energía hidroeléctrica tiene un rol muy importante en las estrategias de planificación del sector eléctrico ecuatoriano. Un pronóstico preciso y fiable de la producción de energía eléctrica logrará satisfacer la creciente demanda de energía eléctrica en Ecuador, por lo mencionado, se requiere que los pronósticos de producción energética sean lo más precisos y fiables, En Ecuador, estudios de pronósticos energéticos considerando técnicas bio-inspiradas son muy escasos.”

Se evidencia que la predicción de la energía hidroeléctrica tiene un rol muy importante en las estrategias de planificación del sector eléctrico ecuatoriano. Un pronóstico preciso y fiable de la producción de energía eléctrica logrará satisfacer la creciente demanda de energía eléctrica en Ecuador [2].

Según el autor [3] Redes neuronales artificiales y sus aplicaciones. Carrera de Ingeniería de Sistemas y Autónomos en la Escuela Superior de Ingeniería Eléctrica de Bilbao “La implementación de las nuevas técnicas y herramientas computacionales están marcado un nuevo paradigma de creación, hoy en día, las aplicaciones de estas técnicas computacionales modernas, van desde las cadenas de producción de la industria alimentaria, hasta la generación eléctrica y medición de la predicción de la generación, debido a las nuevas necesidades de la

sociedad moderna, se han desarrollado diversidad de técnicas y algoritmos informáticos como el desarrollo de la inteligencia artificial o redes neuronales.”

Particularmente los algoritmos de inteligencia artificial, más concretamente, las redes neuronales artificiales (RNA), en los últimos años han demostrado un gran desarrollo en su evolución, y un gran impacto en muchas áreas del conocimiento [3].

### 3.2. CENTRAL HIDROELÉCTRICA ILLUCHI 1

En la figura 1 y 2 se muestra la central que se encuentra ubicada en la ciudad de Latacunga, provincia de Cotopaxi. Desde el punto de vista hidro energético el área del proyecto tiene dos zonas con las condiciones hidrológicas, climatológicas y fisiográficas. El área de estudio está ubicada en las coordenadas planas UTM zona 17, descritas en la tabla 5:

Tabla 5: Coordenadas del dique de captación y de las casas de máquinas [4]

	<b>Este</b>	<b>Norte</b>
Captación del dique	782495,55	9899185,49
Casa de máquinas central Illuchi 1	774997,58	9898281,86

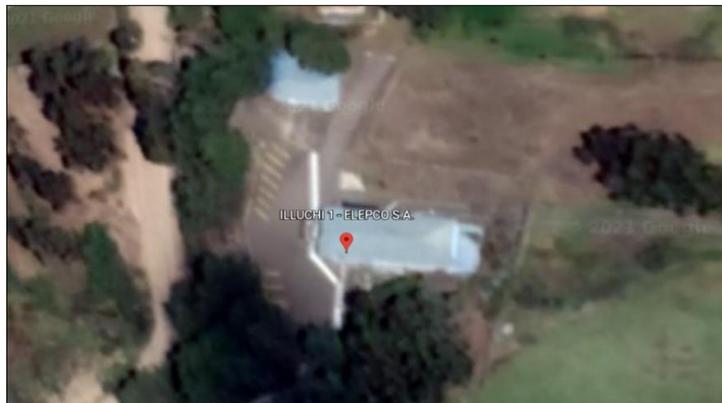


Figura 1: Ubicación de la casa de máquinas central hidroeléctrica Illuchi 1 [4]



Figura 2: Ubicación del dique de la central hidroeléctrica Illuchi 1 [4]

La central hidroeléctrica Illuchi 1, tiene 72 años de operación en forma continua, se encuentra operando desde el año 1 951.

La central hidroeléctrica Illuchi 1 es una central hidroeléctrica que consta de cuatro turbinas tipo Pelton de la marca Bell que impulsan cuatro generadores de la marca BBC que funcionan a 2 400 voltios. La potencia total instalada es de 5,244 kVA, dividida en dos grupos de 872 kVA a 1.200 rpm cada uno y dos grupos de 1 750 kVA a 900 rpm.

La subestación de elevación consta de tres transformadores 23/2,4 kV, 1 750 kVA operando en paralelo e interconectados al sistema de ELEPCO a través de líneas trifásicas de 23 kV con El Calvario S/E [4].

La central hidroeléctrica Illuchi 1 abastece la demanda eléctrica a la subestación El Calvario, que provee a los siguientes sectores:

- Oriental
- Latacunga Sur
- Latacunga Centro Sur
- Industrial Sur

En la central hidroeléctrica Illuchi 1 existe varios equipos los cuales se mientras en la tabla 6.

Tabla 6: Equipos de la central hidroeléctrica Illuchi 1 [4]

Equipo	Magnitud	Unidad	Parámetro
Generadores 1 y 2	3250	kVA	Potencia aparente
	0,8	-	Factor de potencia
	60	Hz	Frecuencia
	720	rpm	Velocidad
	10	-	Polos
	2400±5%	V	Voltaje
	781,8	A	Corriente
Generador 3	1,51*	p.u.	Impedancia ( $X_d$ )
	1,057*	p.u.	Impedancia ( $X_d$ )
Generador 4	1,69*	p.u.	Impedancia ( $X_d$ )
	1,183*	p.u.	Impedancia ( $X_d$ )
Turbina 1 y 2	2675	kW	Potencia activa
	950	l/s	Caudal
	720/1320	rpm	Velocidad
Transformador principal	6500	kVA	Potencia aparente
	2400/13800	V	Voltaje
	8,43	%	Impedancia
	±2 de 2,5%	%	Taps en Alto Voltaje

### 3.3. LA CENTRAL ILLUCHI 1 ES UNA CENTRAL “DE DERIVACIÓN”

En este tipo de proyectos, se acumula una gran cantidad de líquido “aguas arriba” de la turbina mediante la construcción de una o más represas que forman un lago artificial.

El depósito permite regular la cantidad de agua que pasa por la turbina. La cantidad que puede pasar por la turbina depende del volumen de interceptación.

La Central Illuchi 1 consta de estas características porque cuenta con un embalse desde donde se conduce el agua a la casa de máquinas, donde se mueven las turbinas y se genera la electricidad.

De acuerdo con la clasificación comúnmente utilizada en nuestro país, la central Illuchi 1 es una central en cascada ya que forma parte de un grupo de centrales que aprovechan el agua de las centrales aguas arriba. La central Illuchi 1 es la primera central del sistema en cascada como el ejemplo de la figura 3 [5].

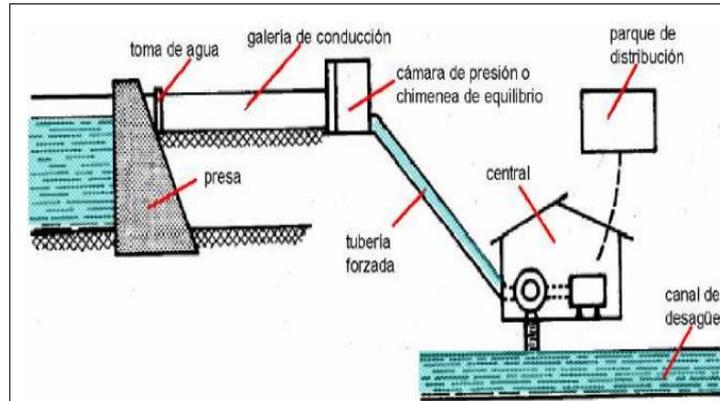


Figura 3: Partes principales de la central De Derivación [5]

El presente trabajo busca realizar una predicción de energía eléctrica utilizando el caudal de una central hidroeléctrica por medio de redes neuronales artificiales.

Se ha realizado una revisión de tecnología utilizadas en la predicción de la energía eléctrica, la tecnología de interés son las redes neuronales, las cuales establecen la relación del caudal con la producción de energía mediante el mapeo entre la entrada y salida de datos a través del aprendizaje [5].

### 3.4. RELACIÓN CAUDAL Y ENERGÍA PRODUCIDA

En una central hidroeléctrica, la energía del agua se utiliza para impulsar una turbina, que a su vez impulsa un generador para producir electricidad. El agua de lluvia que cae sobre la superficie de la Tierra tiene energía potencial relacionada al océano al que fluye. Esta energía se convierte en trabajo de eje y el agua fluye a lo largo de una distancia vertical considerable.

Por lo tanto, la energía hidroeléctrica es una fuente de energía renovable naturalmente disponible proporcionada por (3.1):

$$P = g\rho QH \quad (3.1)$$

En donde P es la potencia hidráulica en Watts, g es  $9,81 \text{ m/s}^2$  (aceleración de la gravedad), p es la densidad del agua,  $1\,000 \text{ kg/m}^3$ , Q es el caudal o descarga,  $\text{m}^3/\text{s}$  y H es la caída o altura [6].

Aquí P es la potencia hidráulica en Watts, g es  $9,81 \text{ m/s}^2$  (la aceleración debida a la gravedad), p es la densidad del agua,  $1\,000 \text{ kg/m}^3$ , Q es el caudal o descarga,  $\text{m}^3/\text{s}$  y H es la altura de caída del agua o altura, m.

La energía eléctrica producida en kWh se puede escribir en la forma de la ecuación (3.2):

$$W = 9,81 \times 1000 \times Q \times H \times \eta \times t \quad (3.2)$$

$$= 9,81QH \text{ nt KWh}$$

donde  $t$  es el tiempo de funcionamiento en horas (8 760 horas/año) y  $\eta$  es la eficiencia del grupo turbina-generador, que varía entre 0,5 y 0,9. Por tanto, la potencia producida depende de la cantidad de agua ( $Q$ ) y de la altura ( $H$ ). La importancia de la energía hidráulica sólo es superada por la energía térmica. Casi el 20 % de la energía total del mundo proviene de centrales hidroeléctricas.

### 3.4.1. Cálculo del caudal

La dinámica de fluidos define al caudal como la cantidad de fluido que avanza en una unidad de tiempo. También es conocido como caudal volumétrico o índice de flujo fluido [6].

El cálculo de caudal de agua viene expresado por la ecuación (3.3) de continuidad:

$$Q = V * S \quad (3.3)$$

Donde es:

- $Q$  es el caudal ( $\text{m}^3/\text{s}$ )
- $V$  es la velocidad ( $\text{m}/\text{s}$ )
- $S$  es la sección de la tubería ( $\text{m}^2$ )

### 3.5. ELEMENTOS ESENCIALES DE UNA CENTRAL HIDROELÉCTRICA

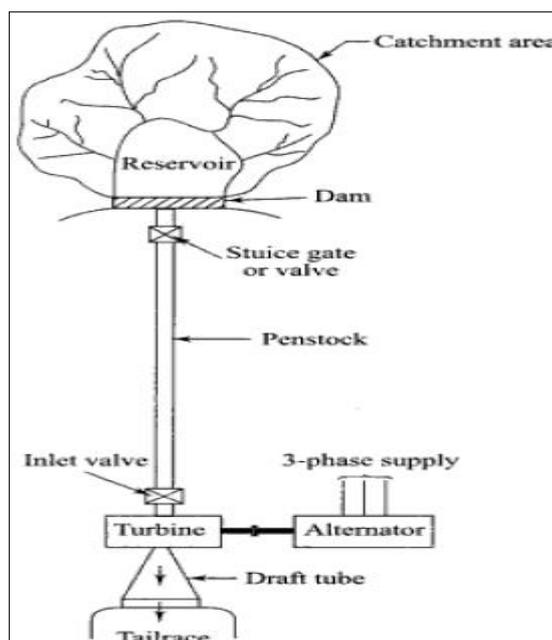


Figura 4: Diagrama de flujo de una central hidroeléctrica típica [7]

Los elementos esenciales de la figura 4 son los siguientes.

1. Área de captación
2. Embalse
3. Presa
4. Aliviaderos
5. Conductos
6. Tanques de compensación
7. Tubos de tiro
8. Casa de máquinas
9. Patio de distribución para transmisión de energía

### **3.5.1. Área de captación**

Toda el área detrás de la presa que desemboca en un arroyo o río a través del cual se ha construido la presa se denomina área de captación. Las características de la cuenca incluyen su tamaño, forma, superficie, orientación, altitud, topografía y geología. Cuanto mayor es la cuenca, más empinada es la pendiente, mayor es la altitud y mayor es la escorrentía total de agua [7].

### **3.5.2. Embalse**

El embalse básicamente es el depósito de agua propiamente, de acuerdo a las especificaciones y circunstancias puede ser necesario la remoción de grandes cantidades de tierra o la inundación de áreas extensas. El uso posterior en tiempos de escasez es fundamental para el uso eficiente de los recursos hídricos [7].

### **3.5.3. Presa**

La presa o represa hidráulica es la estructura cuya función principal es la de retener y almacenar grandes cantidades de agua, las represas están diseñadas para soportar altas presiones y se pueden construir de cemento, concreto armado o materiales que sean capaces de cumplir con la tarea de almacenar agua [7].

### **3.5.4. Aliviaderos**

Cuando el nivel del agua en la cuenca del embalse aumenta, la estabilidad de la estructura de la presa está en peligro. Para aliviar el depósito de este exceso de agua, se proporciona una estructura en el cuerpo de una presa o cerca de ella. Esta estructura de protección se llama aliviadero. Proporciona estabilidad estructural a la presa en condiciones de inundaciones sin elevar el nivel del embalse [7].

### **3.5.5. Tanques de compensación**

Un tanque de compensación es un pequeño depósito en el que el nivel del agua sube o baja para reducir los cambios de presión para que no se transmitan al conducto cerrado. Si la casa de

máquinas está ubicada a poca distancia de la obra maestra, los tanques de compensación no son necesarios. Por lo tanto, para plantas de escorrentía y esquemas de altura media, no se necesita un tanque de compensación. Los tanques de compensación son necesarios para plantas de gran altura donde el agua se lleva a la casa de máquinas a través de túneles y compuertas. En la figura 5 se muestra una disposición típica, donde el tanque de compensación es un tubo vertical conectado a la compuerta. Sin desbordamiento de agua [7].

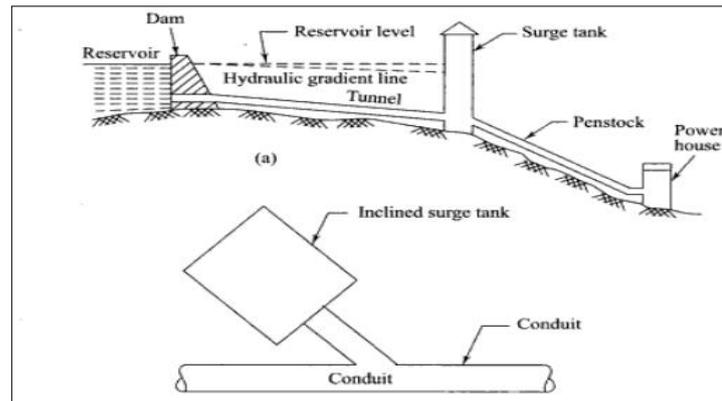


Figura 5: Tanque de compensación a nivel del suelo [7]

### 3.5.6. Geometría del tanque mediante fotogrametría

El concepto de fotogrametría se refiere a “medición sobre fotos”, palabra que proviene de los vocablos “fotograma” y “metrón”, que corresponden a los siguientes significados:

- Fotograma “phos” o “photós”: luz “gramma”: trazado, dibujo
- Metrón: medir

La Sociedad Americana de Fotogrametría y Teledetección (ASPRS) define a la fotogrametría como "Es el arte, ciencia y tecnología para la obtención de medidas fiables de objetos físicos y su entorno, a través de grabación, medida e interpretación de imágenes y patrones de energía electromagnética radiante y otros fenómenos" [8].

La fotogrametría se define como la unión de técnicas para obtener medidas confiables de elementos específicos mediante el uso de fotografías aéreas, ya sea una o varias, con la finalidad de definir atributos geométricos como posición, tamaño y forma de los mismos. De una manera resumida se puede decir que esta técnica consiste en hacer uso de fotografías desde el punto de vista métrico con el objetivo de obtener mapas y planos de alta precisión [8].

Algunas de las ventajas de trabajar con fotogrametría son las siguientes:

- Se logra obtener información objetiva con representaciones actualizadas de las zonas de interés.
- Genera resultados homogéneos de planimetría y altimetría de alta precisión.
- Para levantar la información se utilizan equipos relativamente económicos.
- Facilidad y rapidez de obtención de resultados.
- Permite realizar levantamientos de regiones de difícil acceso.

#### a. Fotogrametría terrestre

La figura 6 corresponde al uso de fotografías que fueron tomadas horizontalmente y verticales al terreno, se pueden realizar mediciones sobre estas, pero presentan una dificultad en cuanto al manejo de la escala. Su aplicación se puede dar para estudios arqueológicos y arquitectónicos [9].

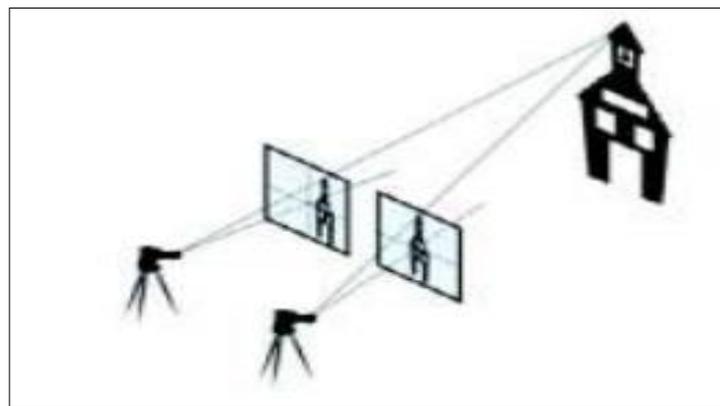


Figura 6: Fotogrametría terrestre [9]

#### b. Fotogrametría aérea

Se obtiene a partir de vehículos aéreos los cuales cuentan con una cámara que captura imágenes perpendiculares a la superficie terrestre. A partir de esta se puede obtener representaciones del terreno en 2 y 3 dimensiones, además permiten cubrir grandes superficies de terreno en un menor tiempo como se observa en la figura 7 [9].

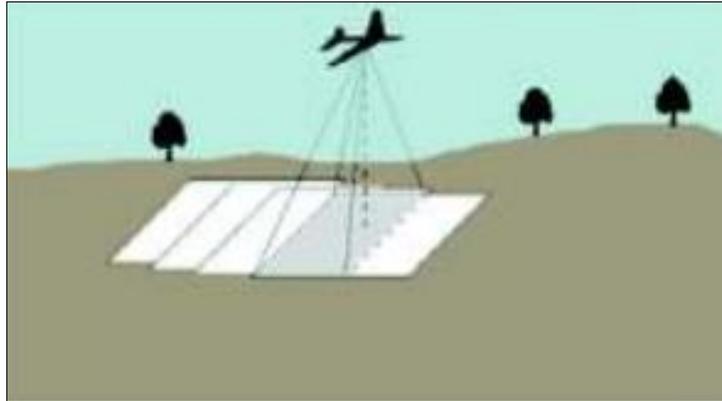


Figura 7: Fotogrametría aérea [9]

### 3.6. PROCESO DE MEDICIÓN DE LOS TANQUES MEDIANTE FOTOGRAMETRÍA

En la figura 8 se observa el repositorio del lado derecho del sector, se adjunta ortomosaico del área de interés con excedente a las orillas en formato Tiff (OrtomosaicoE2TIFF), y en formato JPG (OrtomosaicoE2JPG), georreferenciados, se adjunta el contorno del área de interés sin excedente en orillas en formato DXF (E2DXF), para uso en software CAD, además en la tabla 7 se observa el valor del perímetro y área del repositorio [9].

Tabla 7: Datos Área de Interés del lado derecho [autores]

Datos Área de Interés del lado derecho	
Perímetro	811,460 m
Área	2899,4 m <sup>2</sup>

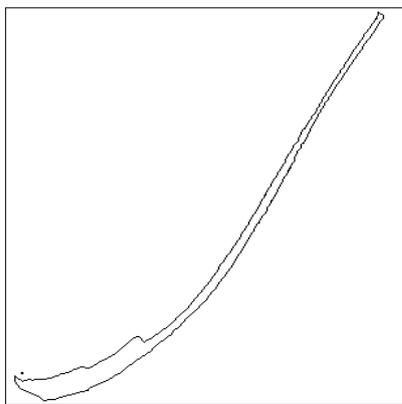


Figura 8: Repositorio del lado derecho [autores]

En la figura 9 se observa repositorio tanque grande del lado izquierdo del sector 2, se adjunta ortomosaico del área de interés con excedente a las orillas en formato Tiff

(OrtomosaicoE3TIFF), y en formato JPG (OrtomosaicoE3JPG), georreferenciados, se adjunta el contorno del área de interés sin excedente en orillas en formato DXF (E3DXF), para uso en software CAD además en la tabla 8 se observa el valor del perímetro y área del repositorio [9].

Tabla 8: Datos Área de Interés Tanque del lado izquierdo [autores]

<b>Datos Área de Interés Tanque del lado izquierdo</b>	
Perímetro	300.589 m
Área	2741.5 m <sup>2</sup>

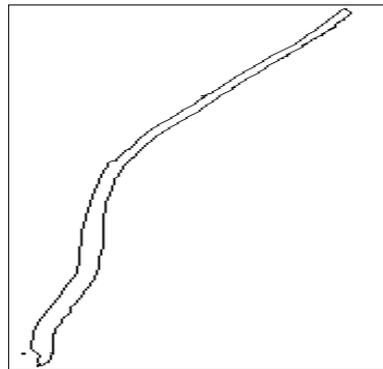


Figura 9: Repositorio del lado izquierdo [autores]

En la figura 10 se observa repositorio tanque central con bocatomas del sector, se adjunta ortomosaico del área de interés con excedente a las orillas en formato Tiff (TanqueTIFF), y en formato JPG (TanqueJPG), georreferenciados, se adjunta el contorno del área de interés sin excedente en orillas en formato DXF (TanqueDXF), para uso en software CAD además en la tabla 9 se observa el valor del perímetro y área del repositorio [9].

Tabla 9: Datos Área de Interés Tanque Central [autores]

<b>Datos Área de Interés Tanque Central</b>	
Perímetro	52.268 m
Área	67.442 m <sup>2</sup>

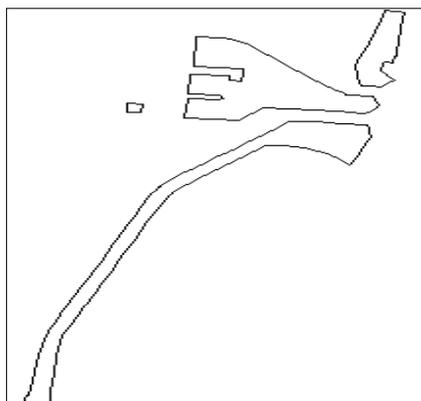


Figura 10: Repositorio del tanque central [autores]

### **3.7. ALGORITMOS QUE SE USAN PARA PREDECIR LA ENERGÍA ELÉCTRICA SEGÚN DATOS DE ENTRADA Y SALIDA**

#### **3.7.1. Algoritmos de entrenamiento**

El entrenamiento de una red neuronal consiste en la determinación de los pesos sinápticos de cada una de las neuronas.

Se emplean múltiples métodos para determinar los pesos. El método más conocido es el algoritmo backpropagation, sin embargo, también existen otros métodos que determinan el valor de los pesos sinápticos minimizando la suma de cuadrados del error (mse) de predicción en los datos de entrenamiento y que también son perfectamente válidos.

El método backpropagation o método del gradiente, es un método clásico de optimización, en el cual se calcula el gradiente mediante un proceso de propagación del error de atrás hacia adelante, es decir, el error se propaga desde las neuronas de salida hasta las neuronas de entrada. Este algoritmo, varía los pesos sinápticos minimizando el error de la salida real y la salida predicha. El método de LevenbergMarquardt, que es el que se utilizará en la toolbox, es una mezcla entre el método backpropagation y el algoritmo de Gauss-Newton [9].

#### **3.7.2. Redes neuronales artificiales**

La red neuronal artificial (RNA, siglas de Artificial Neural Network en inglés) es un modelo inspirado en el sistema nervioso humano, que consiste en un conjunto de interconexiones de neuronas biológicas que permiten procesar información y obtener el resultado deseado, acorde a satisfacer las necesidades humanas. Se estima que en el cerebro humano promedio hay alrededor de 100 mil millones de neuronas biológicas, con más de 10 000 interconexiones entre neuronas.

Las redes neuronales permiten la aproximación de funciones no lineales, lo que permite modelar fenómenos complejos donde las variables no están correlacionadas. Así, con la aplicación de redes neuronales artificiales se han logrado atractivos resultados en medicina, economía, comunicaciones y otros campos del conocimiento, una de las características es la reducción lineal que domina el análisis de diferentes problemas a lo largo de la historia, si bien los supuestos simplificadores permiten desarrollar formulaciones de modelos matemáticos para explicar fenómenos de diferente naturaleza, es imposible abandonar el modelo lineal básico, que se basa precisamente en la linealidad de sus ecuaciones [9].

### 3.7.3. Ventajas de las redes neuronales artificiales

Las ventajas más importantes de las RNA son:

- a. **Aprendizaje adaptativo:** Las redes neuronales artificiales tienen la capacidad de aprender a realizar tareas basadas en el entrenamiento (datos de entrenamiento). Es decir, una RNA puede cambiar su estructura en función de los datos de entrenamiento.
- b. **Autoorganización:** Las RNA crean su propia organización o representan la información recibida durante la fase de aprendizaje.
- c. **Tolerancia a fallos:** Las RNA tienen la ventaja de preservar la funcionalidad de la red cuando la red se destruye parcialmente. No olvide que esta tolerancia puede ser mayor o menor dependiendo de la aplicación que utilice la RNA.
- d. **Modelación no lineal:** Las redes neuronales artificiales permiten la aproximación de funciones no lineales (eliminando el supuesto de reducción lineal), lo que permite modelar fenómenos complejos donde las variables no están correlacionadas.

### 3.7.4. Desventajas de las redes neuronales artificiales

- a. Los paquetes computacionales en el mercado tienen una adaptabilidad limitada al problema y algunos tienen limitaciones de tamaño de muestra.
- b. La programación en red es un poco tediosa, pero resulta ser muy exitosa.

### 3.7.5. Elementos de una red neuronal artificial

#### a. Neurona biológica

En la figura 11 se observa una neurona biológica es una célula nerviosa eléctricamente excitable e interconectada en el cerebro que procesa y transmite información a través de señales eléctricas y químicas. Estas neuronas están conectadas entre sí para formar una red neuronal en el cerebro. Una neurona se compone de tres partes: cuerpo celular, las dendritas y el axón [9].

Fue Ramón y Cajal (1888) quién descubrió la estructura celular (neurona) del sistema nervioso. Defendió la teoría de que las neuronas se interconectaban entre sí de forma paralela, y no formando un circuito cerrado como el sistema sanguíneo.

Una neurona consta de un cuerpo celular (soma) de entre 10 y 80  $\mu\text{m}$ , del que surge un denso árbol de ramificaciones (dendritas) y una fibra tubular (axón) de entre 100  $\mu\text{m}$  y un metro.

De alguna forma, una neurona es un procesador de información muy simple:

- Canal de entrada: dendritas.
- Procesador: soma.
- Canal de salida: axón.

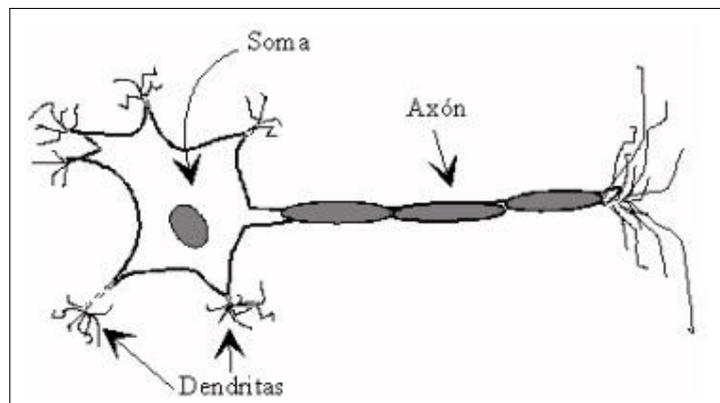


Figura 11: Neurona Biológica [9]

Una neurona cerebral puede recibir unas 10 000 entradas y enviar a su vez su salida a varios cientos de neuronas.

La conexión entre neuronas se llama sinapsis. No es una conexión física, sino que hay unos 2  $\mu\text{m}$  de separación. Son conexiones unidireccionales, en la que la transmisión de la información se hace de forma eléctrica en el interior de la neurona y de forma química entre neuronas; gracias a unas sustancias específicas llamadas neurotransmisores.

No todas las neuronas son iguales, existen muchos tipos diferentes según el número de ramificaciones de sus dendritas, la longitud del axón y otros detalles estructurales. Sin embargo, como se ha visto, todas ellas operan con los mismos principios básicos [9].

#### **b. Neurona artificial**

Son elementos computacionales simples que proporcionan una única respuesta (salida) basada en vectores de entrada del mundo exterior y/o en base a estímulos recibidos de otras neuronas.

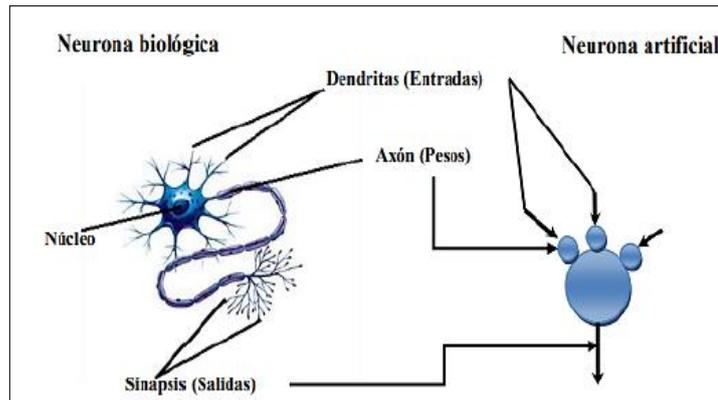


Figura 12: Relación entre la neuronal artificial y biológica [9]

En la figura 12 se observa la relación entre la neuronal artificial y biológica, las neuronas biológicas se consideran simples procesadores de información, tienen un canal de entrada de información llamado dendrita (rama) donde la información que llega por impulsos eléctricos es recolectada y propagada dentro de la neurona, esta información llega al órgano informático (el soma) que la procesa, produciendo una respuesta que se transmitirá al canal de salida (el axón) (dependiendo del nivel del impulso eléctrico, la neurona se activará o no), los axones envían esta señal a las dendritas de las neuronas vecinas, y esta conexión entre los axones y las dendritas se denomina sinapsis.

Una sinapsis es un espacio fluido (concentración de elementos ionizados) donde los axones y las dendritas contactan, pero no se fusionan, debido a la presencia de elementos ionizantes, este espacio tiene propiedades conductoras que activan o impiden el paso de pulsos eléctricos.

Las neuronas artificiales consisten en estructuras llamadas capas, una red RNA consta de un conjunto de capas, por lo tanto, la información se distribuye a lo largo de las sinapsis de la red, lo que hace que el sistema sea algo tolerante a fallas. Las redes neuronales artificiales pueden adaptar sus operaciones a una variedad de contextos modificando las conexiones entre las neuronas, lo que significa que pueden aprender de la experiencia y generalizar conceptos.

De manera similar, las redes neuronales artificiales consisten en conexiones simples entre elementos llamados nodos neuronales, células, unidades de procesamiento o elementos, están dispuestos en una estructura para formar una red que permite la ejecución y distribución de cálculos, al igual que las redes neuronales biológicas, las RNA pueden aprender, reconocer patrones en los datos y adaptarse a los cambios, que es una de las principales razones de su uso en el modelado de datos. Las redes neuronales artificiales imitan la estructura de los sistemas neuronales biológicos. La figura 13 muestra la equivalencia de las estructuras neuronales biológicas y artificiales [9].

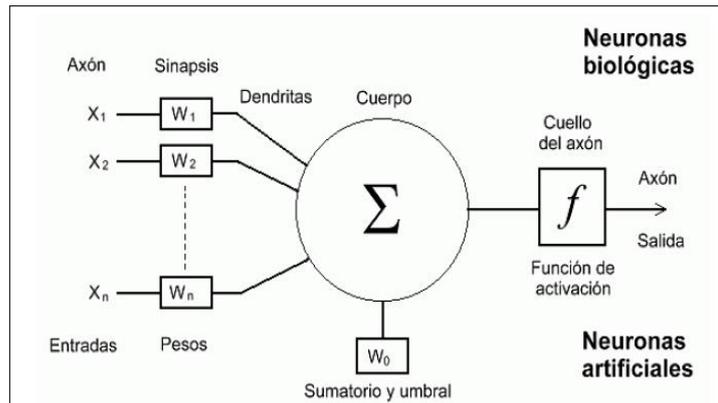


Figura 13: Estructura de una neurona artificial y una biológica [9]

Las neuronas artificiales tienen un número determinado de entradas (externas o de salida de neuronas anteriores). Cada entrada tiene un peso asignado que representa cuánto afecta la entrada a la neurona (sinapsis). Las neuronas procesan la información agregando entradas y umbrales (siempre se recomienda el umbral para mejorar el procesamiento de la señal) y luego pasan a través de una función de activación para generar la señal de salida de la neurona. La RNA funciona para adaptarse a múltiples entornos modificando las conexiones entre las neuronas [9].

### c. Comparación entre la neurona biológica y la artificial

En la figura 14 se observa un diagrama de flujo de la comparación entre la neurona artificial y biológica.

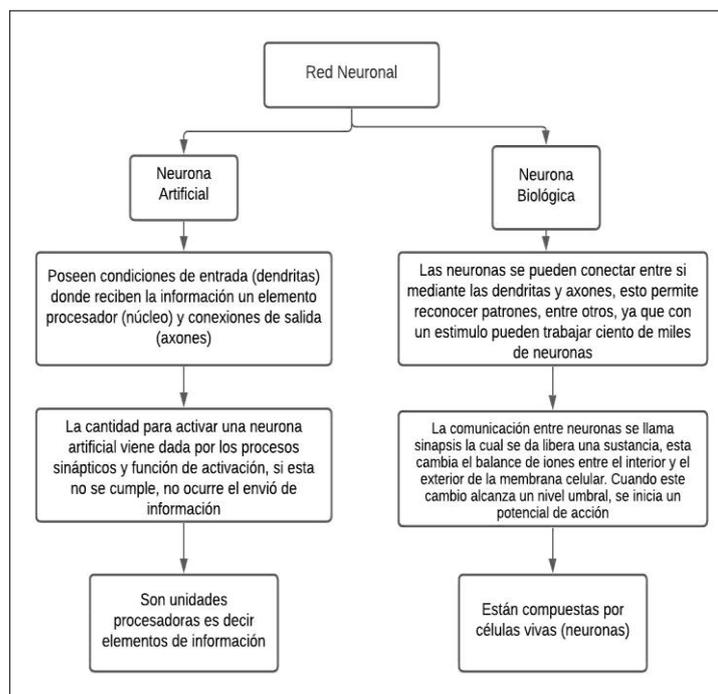


Figura 14: Comparación entre una red artificial y biológica [9]

#### **d. Topología de una red neuronal**

La topología de una red neuronal artificial se refiere a la organización y disposición de las neuronas en la red, formando capas de procesadores interconectados por sinapsis unidireccionales, a arquitectura de una red neuronal artificial depende de cuatro parámetros principales: la cantidad de capas en el sistema, la cantidad de neuronas en cada capa, el grado de conexión entre las neuronas y el tipo de conexiones de las neuronas, por su parte, la conectividad entre nodos de la red está relacionada con la forma en que se transmiten las salidas de las diferentes unidades y la forma en que se convierten en entradas a otros procesadores. Las topologías neuronales se pueden clasificar según diferentes criterios:

#### **e. Numero de capas**

Para realizar la implementación de la red neuronal se ha utiliza una toolbox de Python que permite manejar redes neuronales con gran facilidad. Sin embargo, habrá que definir una serie de parámetros, como el tipo de red neuronal, el número de capas, el número de neuronas, el método de entrenamiento.

La primera característica a completar, es definir el número de capas que van a formar la red neuronal, como se mencionó con anterioridad una red neuronal está compuesta por una capa de entrada con 10 neuronas, dos capas ocultas, dos capas de congelamiento para evitar recordar datos y una capa de salida. La mayoría de los problemas se pueden resolver con una capa intermedia y por lo tanto una única capa intermedia será la elección. La red neuronal se puede representar como se muestra en la figura 15 [9].

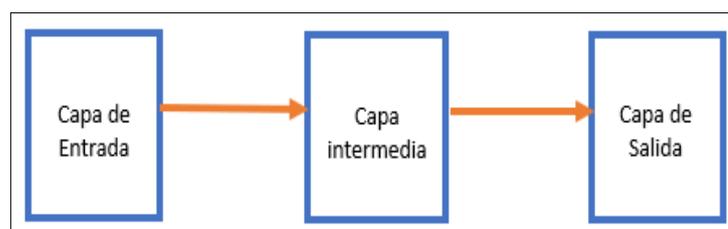


Figura 15: Capas de la red neuronal [9]

#### **➤ Capa de entrada**

El número de neuronas de la capa de entrada viene definido por el número de entradas a la red neuronal por lo tanto inicialmente se tendrán n neuronas en la capa de entrada, aunque después este número de neuronas se verá reducido debido a la eliminación de alguna de las entradas [9].

### ➤ **Capa intermedia**

No existe ningún procedimiento específico que indique de una manera clara el número de neuronas de la capa intermedia, aunque si existen dos vertientes que son bastante utilizadas para hallar el número de neuronas de la capa oculta:

- a) **Prueba y error:** Este método consiste en entrenar la red varias veces con un número de neuronas diferentes en la capa intermedia. Se seleccionará el número de neuronas con un menor error sobre los datos de validación.
- b) **Mediante reglas heurísticas:** Se realiza una optimización de los parámetros de la red mediante algoritmos genéticos. Esto presenta el inconveniente de que se necesita mucho tiempo para dar con la solución al problema. La determinación del número óptimo de neuronas de la capa oculta está estrechamente ligada con la determinación de la complejidad óptima del modelo [9].

### ➤ **Capa de Salida**

Únicamente se utilizará una neurona en la capa de salida, ya que solo se quiere conocer la potencia estimada [9].

#### **3.7.6. Red neuronal LSTM (Long Short-Term Memory)**

Las redes LSTM (Long Short-Term Memory) son un tipo especial de redes neuronales recurrentes diseñadas con celdas de memoria que mantienen su estado a largo plazo. El principal objetivo de este tipo de redes es la solución del problema de desvanecimiento del gradiente experimentado en las redes RNN.

En una red recurrente, la información se almacena de dos modos diferentes. La función de activación de una unidad neuronal es una función de la historia reciente del modelo por lo que forman una memoria a corto plazo (short-term memory).

Los pesos de los enlaces forman una memoria ya que son modificados basados en la experiencia. Los pesos de los enlaces se inician con un valor aleatorio y durante el proceso de entrenamiento se buscan los valores óptimos de los pesos que minimizan la función de pérdida. La función de pérdida se calcula mediante el cálculo de la entropía cruzada entre el resultado

calculado por la red y el resultado esperado. Sin embargo, su cambio es mucho más lento que el de las funciones de activación.

Las redes Long Short-Term Memory (LSTM) son un intento de almacenar la función de activación por un periodo de tiempo más largo. Una unidad LSTM básica está compuesta de una celda de memoria, una puerta de entrada y una puerta de salida [10].

### 3.7.7. Importancia del pronóstico de la red neuronal artificial

El objetivo principal del sistema eléctrico es brindar servicios de energía de manera continua, segura y confiable, por lo que debe adaptarse constantemente a la demanda. Sin embargo, los sistemas eléctricos no son sistemas adaptativos, y para cumplir con dichas demandas es necesario estimar los valores que requiere el sistema para poder realizar modificaciones y cumplir con las demandas del servicio. Las previsiones más precisas reducen el riesgo de tener que utilizar redespachos más caros o, por el contrario, tener que abandonar la reprogramación de grupos previamente programados [11].

### 3.7.8. Función de activación de la red neuronal artificial

Las redes neuronales artificiales siguen la regla de que la activación o el estado de una neurona se determina en función de la entrada que llega a dicha neurona. Estas funciones pueden ser lineales o no lineales. A continuación, en la figura 16, se muestra las diferentes funciones de activación más comunes:

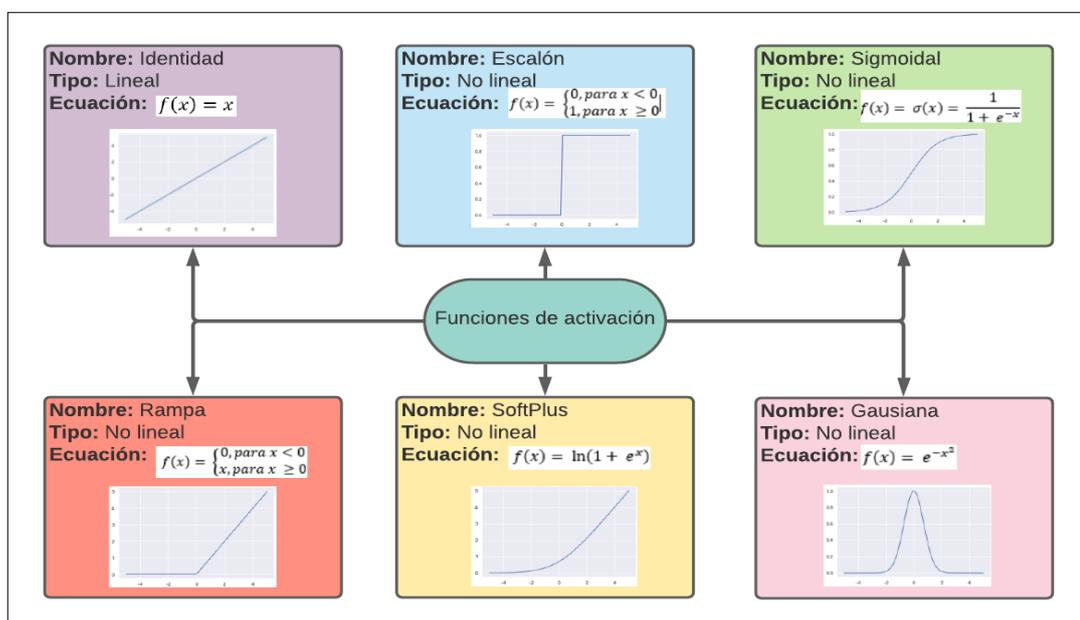


Figura 16: Funciones de activación [12]

### 3.8. IDENTIFICAR EL MACHINE LEARNING Y EL DEEP LEARNING

En la figura 17 se observa un diagrama de flujo de la comparación entre Machine Learning y el Deep Learning.

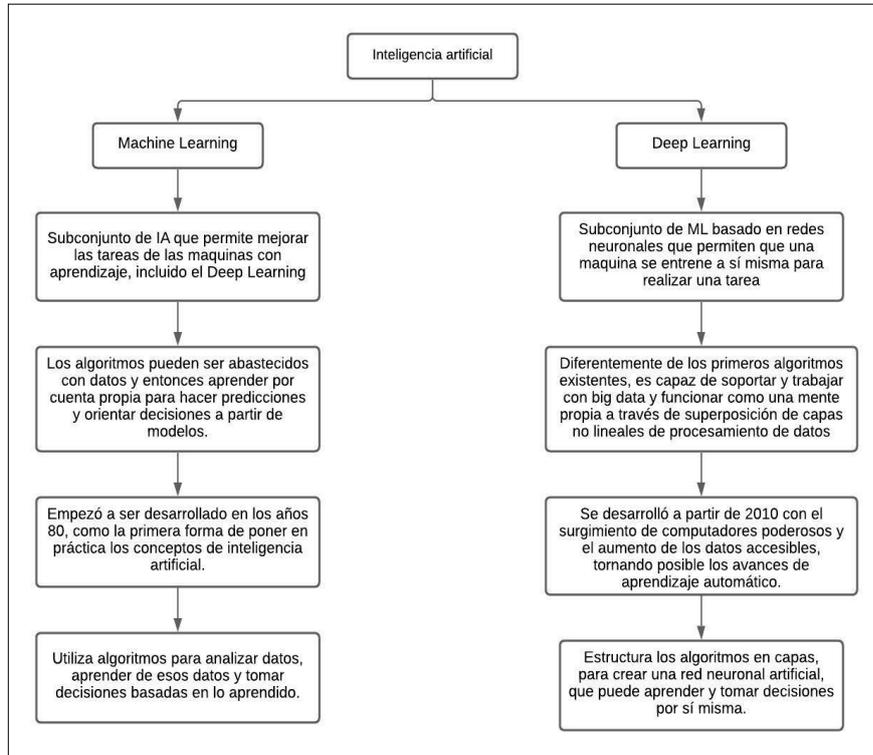


Figura 17: Machine Learning y el Deep Learning [13]

### 3.9. MODELO MATEMÁTICO DE UNA RED NEURONAL ARTIFICIAL

Conocidas como redes neuronales artificiales, consisten en una serie de procesadores básicos llamados neuronas artificiales que constituyen dispositivos informáticos simples que brindan una respuesta única basada en entradas del mundo exterior o estímulos recibidos de otras neuronas (salida), se caracterizan tres tipos de neuronas artificiales: unidades de entrada, unidades de salida y unidades ocultas, se observa en la figura 18.

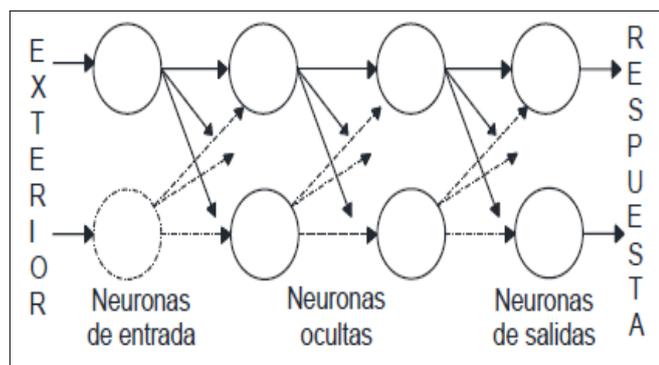


Figura 18: Unidades de la red neuronal artificial [14]

La gran variedad de modelos de redes neuronales existentes en la actualidad obliga en cierta medida a la realización de clasificaciones o taxonomías. De esta forma se pueden clasificar desde triple óptica: en función de la forma de aprendizaje en función de la arquitectura y la tercera en las aplicaciones [14].

### 3.9.1. Fase de aprendizaje o entrenamiento de la red neuronal artificial

Se parte de un modelo determinado de neurona y de una determinada arquitectura de red, estableciéndose pesos iniciales aleatorios o nulos. A partir de este modelo es necesario entrenar la red para solucionar el problema objeto de estudio. El ajuste puede ser mediante el modelado de las sinapsis (pesos) de la red, a través de una regla de aprendizaje y mediante la creación y/o destrucción de neuronas en la red.

Se caracteriza por la presencia de un agente externo (supervisor o maestro) que controla el proceso de entrenamiento, estableciendo la respuesta que debería generar la red (output del sistema) a partir de una entrada determinada. El tipo de algoritmo de aproximación permite distinguirse tres tipos de aprendizajes supervisado: por corrección de error, por refuerzo o de tipo estocástico como se observa en la figura 19 [14].

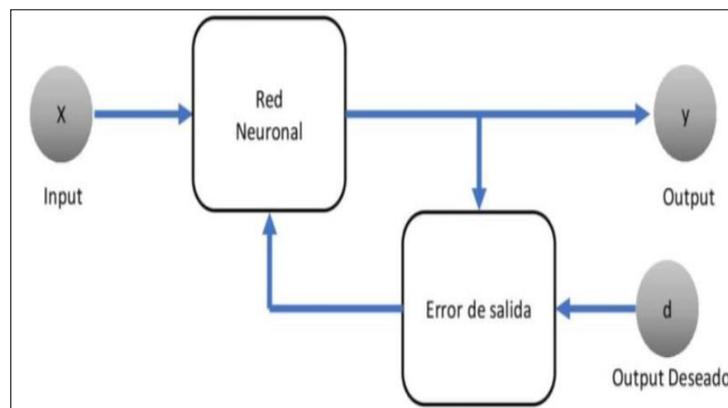


Figura 19: Estructura de modelo supervisado [14]

### 3.9.2. Aprendizaje por corrección de error

Constituye el tipo de aprendizaje supervisado más utilizado en la práctica. Su funcionamiento se basa en el ajuste de los pesos de las conexiones de la red a partir de la diferencia entre valores deseados y los obtenidos por el sistema, esto es, en función del error cometido en la salida. Una de las reglas más sencillas de aprendizaje por corrección se muestra en la ecuación (3.4):

$$\Delta W_{ij} = \alpha * x_i (t_j - y_j) \quad (3.4)$$

Donde  $\Delta W_{ij}$  la variación en el peso de la conexión entre las neuronas  $i$  y  $j$ ,  $x_i$  la  $i$ -ésima entrada a la  $j$ -ésima neurona,  $t_j$  el valor de salida deseado para la neurona  $j$ ,  $y_j$  el valor de salida obtenido en la  $j$ -ésima neurona y  $\alpha$  el factor o tasa de aprendizaje que regula su velocidad [14].

### 3.9.3. Aprendizaje por refuerzo

La tarea del supervisor se limita a indicar, mediante una señal de refuerzo (éxito = +1 o fracaso = -1), si la salida obtenida por la red se ajusta o no a la deseada  $y$ , en función de ello, se procede al ajuste de los pesos utilizando un mecanismo basado en probabilidades, también es denominado aprendizaje de “premio-castigo” como se observa en la figura 20.

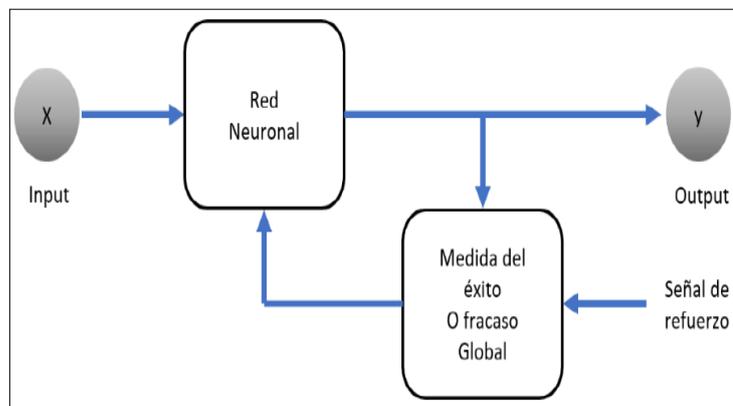


Figura 20: Estructura de aprendizaje por refuerzo [14]

### 3.9.4. Aprendizaje no supervisado

Redes con aprendizaje no supervisado no requieren información externa para ajustar los pesos de las conexiones neuronales. Se presenta a la red un conjunto de patrones sin adjuntar la respuesta deseada por lo que la red, por medio del algoritmo de aprendizaje, estima la función de densidad probabilística  $p(x)$  que describe la distribución de patrones  $x$ ,  $x \in \mathcal{R}^p$  como se observa en la figura 21.

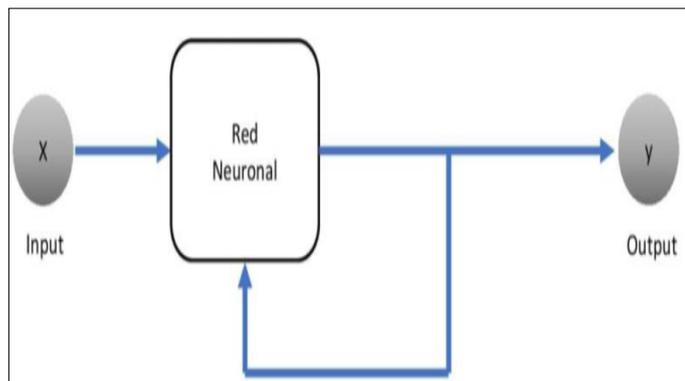


Figura 21: Estructura aprendizaje no supervisado [14]

### 3.10. MÉTODOS PARA PREDECIR LA ENERGÍA ELÉCTRICA

#### 3.10.1. Métodos de Tendencia simple

Si la serie no tiene una tendencia marcada de crecimiento o de decrecimiento, a simple vista no se puede determinar su comportamiento por lo que se recurre a la regresión simple y a partir del coeficiente de determinación  $R^2$ , se conocerá qué función es semejante al comportamiento de la serie ( $R^2 \geq 0,8$ ). También se tomará en cuenta el criterio de errores entre el valor estimado y el valor real de la serie.

El concepto de media aritmética o promedio también contempla la decisión de qué tan largo será el periodo que usaran para calcular el promedio, pues entre más largo sea, nuestro pronóstico calculado será más suavizado. Por otro lado, cuando se observa una tendencia en el caudal, un periodo más corto será mejor al entregarnos resultados con mayor oscilación, llegando a la tendencia actual.

En este sentido, un periodo más largo en el promedio simple nos da un resultado más uniforme, pero con tendencia retrasada. Con un periodo más corto, se tiene más oscilación, pero se está cerca a la tendencia [15].

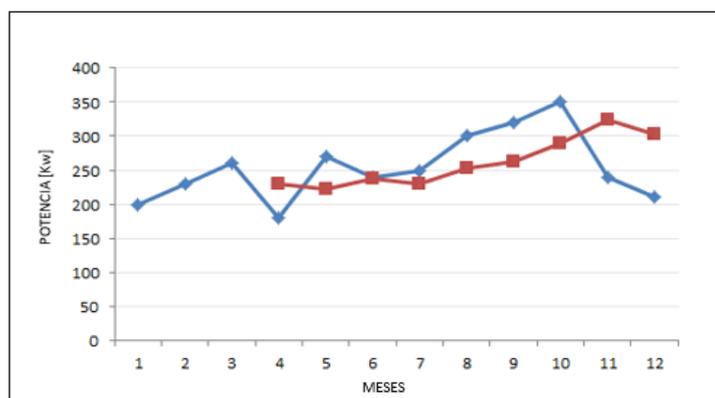


Figura 22: Ejemplo del método de tendencia simple [15]

En la figura 22 se puede observar que en el mes 3, 5 y 10 existen cambios y consumos significativos de potencia.

#### 3.10.2. Método de Suavizamiento exponencial y promedios móviles

El modelo de ajuste del método Holt y Brown no se ajustan al comportamiento de la serie mientras que el modelo simple estacional genera un comportamiento similar a la serie de datos, no obstante el Modelo Winter Aditivo es quien posee el mayor coeficiente de correlación entre los modelos ajustados, en este caso se puede afirmar el comportamiento de la serie de

generación de energía eléctrica obedece al Modelo Winter Aditivo pues el coeficiente de correlación es ni mayor a 0,8 y es el más cercano a 1.

A diferencia de los métodos de promedio móvil simple y ponderado, este método no necesita de gran volumen de datos históricos del caudal. Por ende, cada vez que se calcula el pronóstico, se remueve la observación anterior y es reemplazada por el valor de caudal más reciente, y aquí es donde radica la ventaja [15].

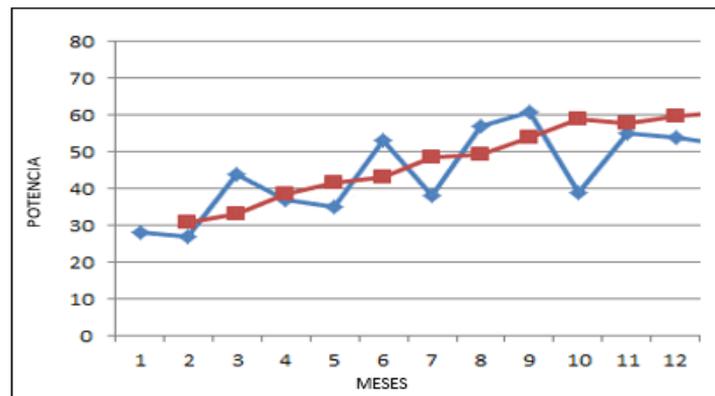


Figura 23: Ejemplo de método de suavizamiento exponencial simple [15]

En la figura 23 se puede observar que la potencia sufre cambios y consumos significativos con el método de suavizamiento en el mes 3, 6 y 9.

### 3.10.3. Método ARIMA

Es un modelo estadístico que utiliza variaciones y regresiones de datos estadísticos con el fin de encontrar patrones para una predicción hacia el futuro. Se trata de un modelo dinámico de series temporales, es decir, las estimaciones futuras vienen explicadas por los datos del pasado y no por variables independientes.

Se utiliza FAC y FACP para estimar los valores de  $p$ ,  $d$ ,  $q$ , la longitud del rezago es criterio del analista y tamaño de la muestra, algunos autores recomiendan que la longitud del rezago sea un tercio de la muestra, en este caso se tienen 40 datos así que se realizara con 10 rezagos.

Una vez obtenidos los correlogramas se procede a analizar su comportamiento y si es necesaria la diferenciación de la serie para transformarla en una serie estacionaria [15].

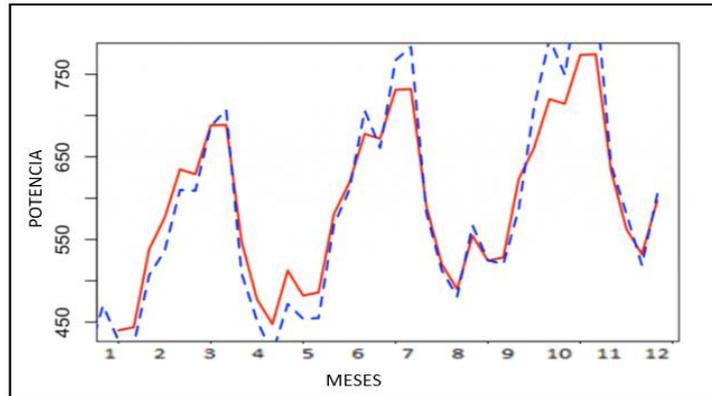


Figura 24: Ejemplo del método de ARIMA [15]

En la figura 24 se puede observar que la potencia sufre cambios y consumos periódicos y repetitivos con el método de ARIMA en el mes 3, 7 y 10.

#### 3.10.4. Método Redes Neuronales Artificiales

Están inspirados en la red neuronal biológica del cerebro humano, que intenta interactuar con objetos en el mundo real de la misma manera que los sistemas biológicos.

Una red neuronal artificial es una estructura compuesta de un número de unidades interconectadas (neuronas artificiales). Cada unidad posee una característica de entrada/salida e implementa una computación local o función. La salida de cualquier unidad está determinada por su característica de entrada/salida, su interconexión con otras unidades, y (posiblemente) de sus entradas externas [15].

##### 3.10.4.1. Selección de la Variable

- **Variable a Predecir:** Caudal mensual de energía eléctrica de una zona del país.
- **Tipo de Variable:** Cuantitativa.
- **Unidad:** GWh.

En caso de contar con una muestra más amplia y detallada del caudal se podrían establecer perfiles de consumo según hora del día, día de la semana y mes del año. Además de la posibilidad de adición de otras variables que influyan en el comportamiento del caudal como la temperatura, precio, PIB entre otras [15].

##### 3.10.4.2. Recolección de Datos

Los datos utilizados para el presente trabajo fueron obtenidos por parte de la hidroeléctrica y consisten en mediciones de la cantidad mensual generación de la energía eléctrica en la zona, además de las mediciones del tanque tomadas.

Para introducir la serie de tiempo a la red neuronal es necesario que los datos sean de fácil manipulación para el software, para ello la base de datos se convierte en una matriz, con el fin de manipulación de los mismos en forma vectorial [15].

#### **3.10.4.3. Procesamiento de Datos**

Se realiza un análisis y transformación de las variables de entrada y salida para minimizar el ruido, detectar tendencias y establecer perfiles para ayudar a la red neuronal en el aprendizaje de patrones relevantes, debido a que las redes neuronales son buscadores de patrones, la representación de los datos es crítica en el diseño de una red exitosa. Las variables de entrada son raramente ingresadas a la red en su forma original [15].

#### **3.10.4.4. Definir conjunto de entrenamiento, validación y prueba**

- a. Conjunto de Entrenamiento:** Es el conjunto de datos que utiliza la red neuronal para aprender los patrones presentes en los datos. Por lo general los datos utilizados para el entrenamiento de la red neuronal corresponden al 80 % de los datos de la serie, dichos datos se seleccionan en forma consecutiva y ordenada.
  
- b. Conjunto de Validación:** Conjunto de datos utilizados para verificar el aprendizaje de la red donde los datos empleados son los más recientes consecuentemente al último valor de la muestra. Durante el proceso de entrenamiento de la red puede ocurrir un problema llamado overfitting (se da cuando la red tan solo aprende puntos individuales en vez de generales), para evitarlo, es necesario incluir un conjunto de validación durante el proceso de aprendizaje de la red. En este caso este conjunto se obviará, debido a que se está utilizando un componente de retardo y por lo tanto el conjunto de entrenamiento se va generando en el tiempo, impidiendo de esta manera que exista un conjunto de validación priori.
  
- c. Conjunto de Prueba:** Es el conjunto de datos encargados de evaluar la capacidad de predicción y precisión de la red, corresponden al 20 % de datos de la serie, estos son los datos restantes una vez seleccionados los patrones de entrenamiento [15].

### 3.10.4.5. Función de Transferencia

El propósito de esta función es prevenir a las salidas de alcanzar valores muy elevados que puedan paralizar la red y detener el entrenamiento de la misma. Como función de transferencia se utiliza la función Sigmoidal (rango de salida entre -1 y +1), en la figura 25 se muestra un ejemplo [15].

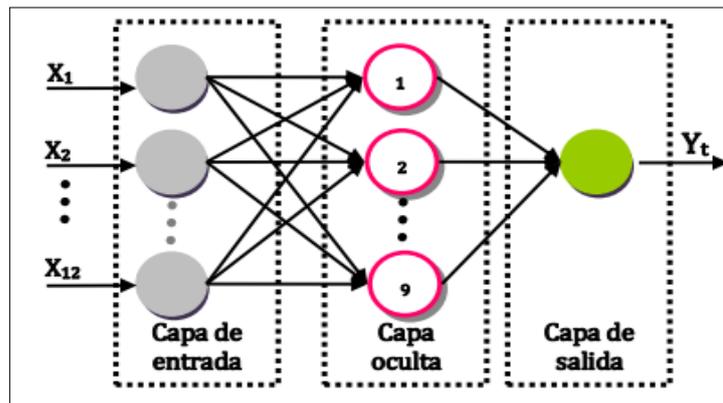


Figura 25: Ejemplo de método de redes neuronales artificiales [15]

### 3.10.4.6. Entrenamiento de la red

Entrenar una red neuronal para aprender patrones involucra el presentarle ejemplos de manera iterativa de las respuestas correctas.

El entrenamiento de la red utiliza la técnica del gradiente descendente inmerso en el algoritmo de propagación hacia adelante. El entrenamiento se detendrá básicamente cuando suceda uno de estos 3 motivos:

- El número de iteraciones excede el número de épocas establecidas.
- La función de evaluación (EMC) toma un valor por debajo de la meta establecida.
- El error medio de la función de evaluación se incrementa para un número específico de iteraciones (este último caso requiere de la existencia del conjunto de validación).

### 3.10.4.7. Número de Iteraciones

A mayor número de iteraciones en el entrenamiento producirá un menor error en el entrenamiento, pero éste a su vez no garantiza que se obtendrá el menor error con el conjunto de prueba. En nuestro caso el número máximo es 10 suficiente para conseguir un nivel de error aceptable [15].

### 3.10.4.8. Estructura de la red neuronal

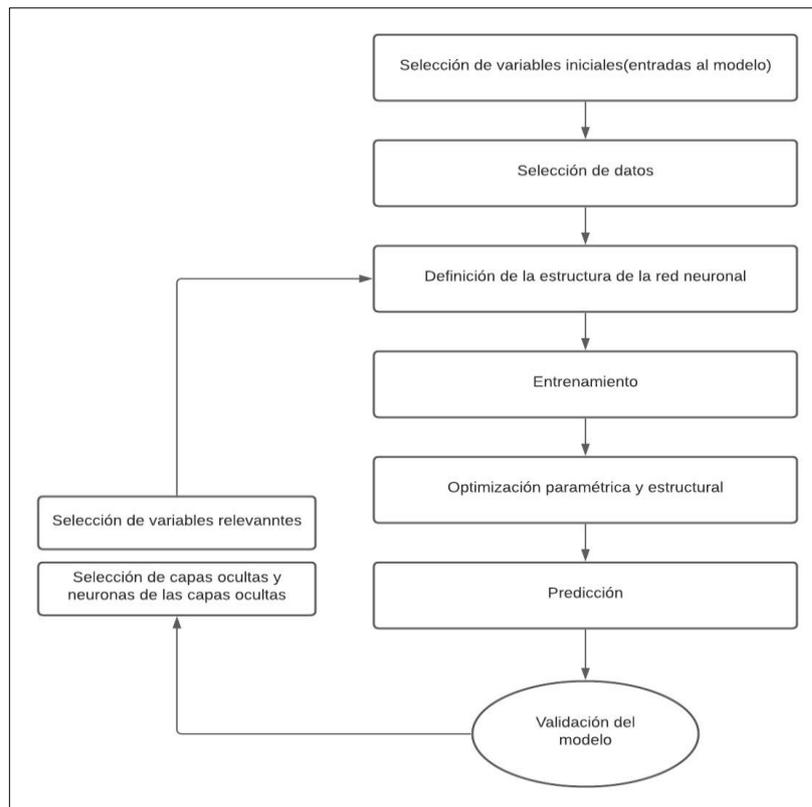


Figura 26: Estructura de la red neuronal [16]

En la figura 26 se puede apreciar el diagrama de flujo para estructurar y construir una red neuronal artificial.

### 3.10.5. Método de Redes neuronales aplicadas a series temporales

A continuación, se mencionan ejemplos actuales de la literatura donde se utilizan modelos basados en redes neuronales para la predicción de series temporales.

Entre los métodos de aprendizaje automático que se utilizan se encuentran modelos creados a partir de redes neuronales basados en capas MLP y LSTM. En la arquitectura del modelo MLP, la red neuronal cuenta con una sola capa oculta. Se definen la cantidad  $N$  de nodos de entrada entre un rango de valores de 1 a 5 mediante un proceso de validación, las entradas las últimas observaciones previas al objetivo de la predicción, mientras que el número de nodos de la capa oculta se establece en  $2N + 1$ . En dicho estudio, el modelo utilizado para implementar la red neuronal recurrente se compone de una capa oculta que contiene los nodos recurrentes, y una de salida que contiene uno o más nodos lineales. La capa de entrada consiste de tres nodos de entrada, seis unidades LSTM que forman la capa oculta y un nodo lineal en la capa de salida para todas las series del conjunto de datos [17].

### 3.11. FACTORES QUE AFECTAN LOS PRONÓSTICOS DE GENERACIÓN DE ENERGÍA

Existen una serie de factores que pueden interferir en el comportamiento de la generación de energía eléctrica, cuya magnitud variará en función del horizonte temporal de la predicción en los cuales se observan en la tabla 10.

Tabla 10: Factores que influyen en el comportamiento de la generación eléctrica [autores]

CORTO PLAZO
Tiempo
Hora del día, día de la semana
Tipo de día
Laboral, feriado
Factores climáticos
Humedad

### 3.12. ARQUITECTURA DE REDES NEURONALES

#### 3.12.1. Arquitectura de una FANN (Feedforward Artificial Neuronal Network)

Morfológicamente, las FANN consisten en un conjunto de neuronas organizadas en capas como se muestra en la figura 27.

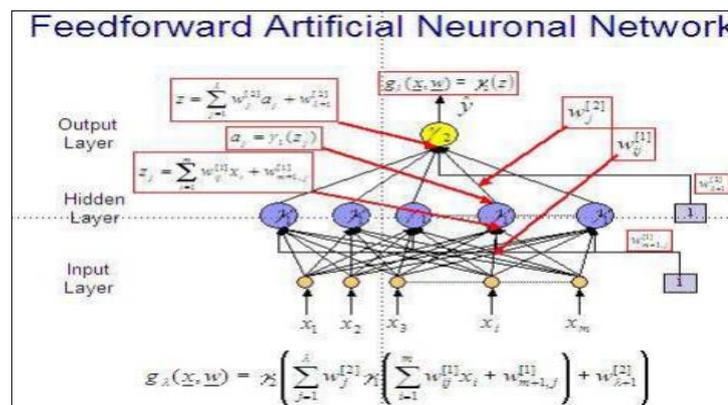


Figura 27: Modelo de una red neuronal artificial feedforward [16]

Las capas se componen en:

- Capa de entrada
- Capa oculta
- Capa de salida

Una vez definida la arquitectura que se desea utilizar en un problema particular, la red neuronal debe ajustarse a una muestra dada a través del proceso de aprendizaje. El algoritmo de aprendizaje consiste en un problema de inferencia de los parámetros libres de la red dado por las conexiones entre las neuronas.

El proceso de aprendizaje de una red neuronal feedforward se caracteriza por ser supervisado pues los parámetros de la red, conocidos como pesos, se estiman a partir de un conjunto de patrones de entrenamiento que consisten en patrones de entrada y salida.

Por lo tanto, el ajuste de la red se produce como resultado de la estimación de los parámetros basado en una muestra de tamaño  $n$ . La estimación es obtenida minimizando una función de costo, mostrado en la ecuación (3.5):

$$w^{LS} = \arg \min \{ L_n(w) : w \in W \subseteq \mathfrak{R}^d \} \quad (3.5)$$

donde  $L_n(w)$  normalmente viene dado por la función del promedio de los errores al cuadrado entre el dato estimado por la red y el dato real, mostrado en la ecuación (3.6):

$$L_n = \frac{1}{2n} \sum_{t=1}^n (y_t - g(x', w))^2 \quad (3.6)$$

El término de "backpropagation" se refiere a la forma en que el gradiente de la función de costo es calculado para redes neuronales feedforward. El aprendizaje involucra un ajuste de los pesos comparando la salida deseada con la respuesta de la red de manera que el error sea mínimo. La función que usualmente es utilizada para medir el error es la suma de los errores al cuadrado o su promedio.

Una de las características más importantes y poderosas de las redes neuronales feedforward es su capacidad como aproximadores universales, esto es, dada cualquier función derivable y acotada, existe una red neuronal con un número suficiente de neuronas escondidas que puede aproximar dicha función con el nivel de precisión deseada [16].

### **3.13. ALGORITMO DE APRENDIZAJE PARA REDES NEURONALES ARTIFICIALES**

El proceso de aprendizaje o entrenamiento de una red neuronal se entiende como la etapa en la que la red neuronal adquiere la capacidad de realizar la función específica que requiere el problema a resolver, los diferentes modelos se clasifican según el paradigma de aprendizaje básico que utilizan, los tres principales paradigmas de aprendizaje son: supervisado, no supervisado y refuerzo.

El aprendizaje supervisado es el paradigma de entrenamiento más común para realizar tareas de clasificación y aplicaciones de predicción. En este enfoque el algoritmo de aprendizaje ajusta adaptativamente la diferencia entre el output deseado y el que la red neuronal genera. Así, la red aprende a realizar automáticamente una tarea prescrita. Esto requiere que el diseñador de la red conozca de antemano la respuesta de la red en una muestra de casos que son usados para el entrenamiento.

El aprendizaje supervisado es el paradigma de entrenamiento más común para tareas de clasificación y aplicaciones de predicción, en este enfoque, el algoritmo de aprendizaje ajusta de forma adaptativa la diferencia entre la salida deseada y la salida generada por la red neuronal, por lo tanto, la red aprende a realizar tareas prescritas automáticamente, esto requiere que el diseñador de la red sepa de antemano cómo responderá la red en las muestras utilizadas para el entrenamiento.

Por el contrario, el aprendizaje no supervisado es adecuado para resolver problemas cuando el problema está bien definido pero la respuesta es desconocida, por ejemplo, la pregunta podría ser: ¿Cómo se relacionan los datos? ¿Qué datos son similares y qué constituye esta similitud? ¿Qué patrón o estructura global es la base para la recopilación de datos? por lo tanto, los cálculos realizados automáticamente por la red neuronal no supervisada finalmente revelan la organización o estructura final. Este tipo de redes neuronales se denominan redes "autoorganizadas". Estas redes se utilizan ampliamente para realizar tareas de agrupamiento y partición jerárquica de datos, que son útiles para descubrir conocimiento en grandes bases de datos.

Mediante el aprendizaje se modifican los pesos de las conexiones de red, y se dice que la red ha aprendido cuando el valor de los pesos se mantiene estable o tiene un error mínimo. Los criterios para cambiar los pesos de conexión están determinados por las reglas de aprendizaje. Algunas redes pueden aprender durante el funcionamiento normal, lo que se denomina aprendizaje ON-LINE; en cambio, si el aprendizaje requiere desconectar la red hasta el final del proceso, se denomina aprendizaje fuera de línea [16].

- a. Aprendizaje a través de la corrección de errores:** los pesos de la red se ajustan en función de la diferencia entre el valor esperado y el valor obtenido por la salida de la red, minimizando así el error resultante.

- b. Aprendizaje reforzado:** considerando éxito = +1 y fracaso = -1, si la salida obtenida en la red se ajusta a la salida deseada, la función se simplifica para indicar aumentando la señal y ajustar el mecanismo de ponderación según la probabilidad.
  
- c. Aprendizaje aleatorio:** suele hacerse según una analogía termodinámica, relacionando una red con un sólido físico con un determinado estado de energía. La energía de la red representará qué tan estable es el sistema. El estado de mínima energía corresponderá al caso en que el peso de la conexión haga que su funcionamiento sea el más adecuado al objetivo deseado. El aprendizaje implicará realizar cambios aleatorios en los valores de peso y determinar la energía de la red (por lo general, la función de energía es una función de Lyapunov). Si la energía después del cambio es menor, acepta el cambio. Por otro lado, si la energía no es menor que el cambio se aceptará de acuerdo a una cierta y predeterminada distribución de probabilidad.

Los diferentes tipos de mecanismos de aprendizaje se describen a continuación:

- a. Regla de Widrow-Hoff o LMS:** es muy importante porque es la base de la mayoría de los algoritmos de aprendizaje para una gran cantidad de redes neuronales.
  
- b. Definición de función de error:** también conocida como función de costo. Mide lo bueno o malo que es el modelo. Cuanto menor es el error, más eficiente es el modelo. Depende de pesos sinápticos, que son incógnitas a resolver durante el entrenamiento.
  
- c. Optimización de función de error:** encuentra un conjunto de pesos sinápticos que minimicen la función de error. Esta búsqueda se realiza a través de un proceso iterativo denominado descenso de gradiente.
  
- d. Reglas de retro-propagación:** Este tipo de red neuronal es adecuada para una variedad de problemas de predicción, especialmente cuando hay datos de entrada ruidosos y la señal de salida debe aproximarse mucho a los valores reales o la señal de entrada debe transformarse para obtener otras señales de salida. La red de propagación hacia atrás está diseñada como una red de propagación hacia adelante de varias capas que utiliza un modelo de aprendizaje supervisado.

### **3.14. REDES NEURONALES PROBABILÍSTICAS**

Según nuestro el problema, una red neuronal probabilística debe ser la encargada de clasificar el vector de entrada en dos clases cuyas regiones de decisión sean asintóticamente cercanas a las determinadas por el clasificador bayes óptimo. La primera capa realiza la tarea de asignar valores de entrada y consta de unidades denominadas unidades vectoriales. Cada vector de entrenamiento tiene uno de esos nodos.

En esta capa existe una unidad que se encarga de sumar la salida de la unidad vectorial de la clase correspondiente. Al sumar estas salidas, se puede "aproximar" la función de densidad estimada correspondiente a cada clase. La última capa contiene un nodo responsable de hacer cumplir las reglas de decisión.

Se puede utilizar este tipo de red para resolver problemas de clasificación. Cuando se presenta una entrada, la primera capa calcula la distancia desde el vector de entrada hasta el vector de entrenamiento de entrada y genera un vector cuyos elementos indican qué tan cerca está la entrada de entrenamiento. La segunda capa agrega estas contribuciones para que cada tipo de entrada produzca la salida de la red como un vector de probabilidad. Finalmente, una función de transferencia competidora a la salida de la segunda capa selecciona la mayor de estas probabilidades y genera 1 para esta clase y 0 para las otras clases. Cabe mencionar que, para un vector de entrenamiento, la clase se conoce de antemano [16].

### 3.15. SOFTWARES PARA EL DESARROLLO DE REDES NEURONALES ARTIFICIALES

En la figura 28 se muestra los Softwares para el desarrollo de redes neuronales artificiales.

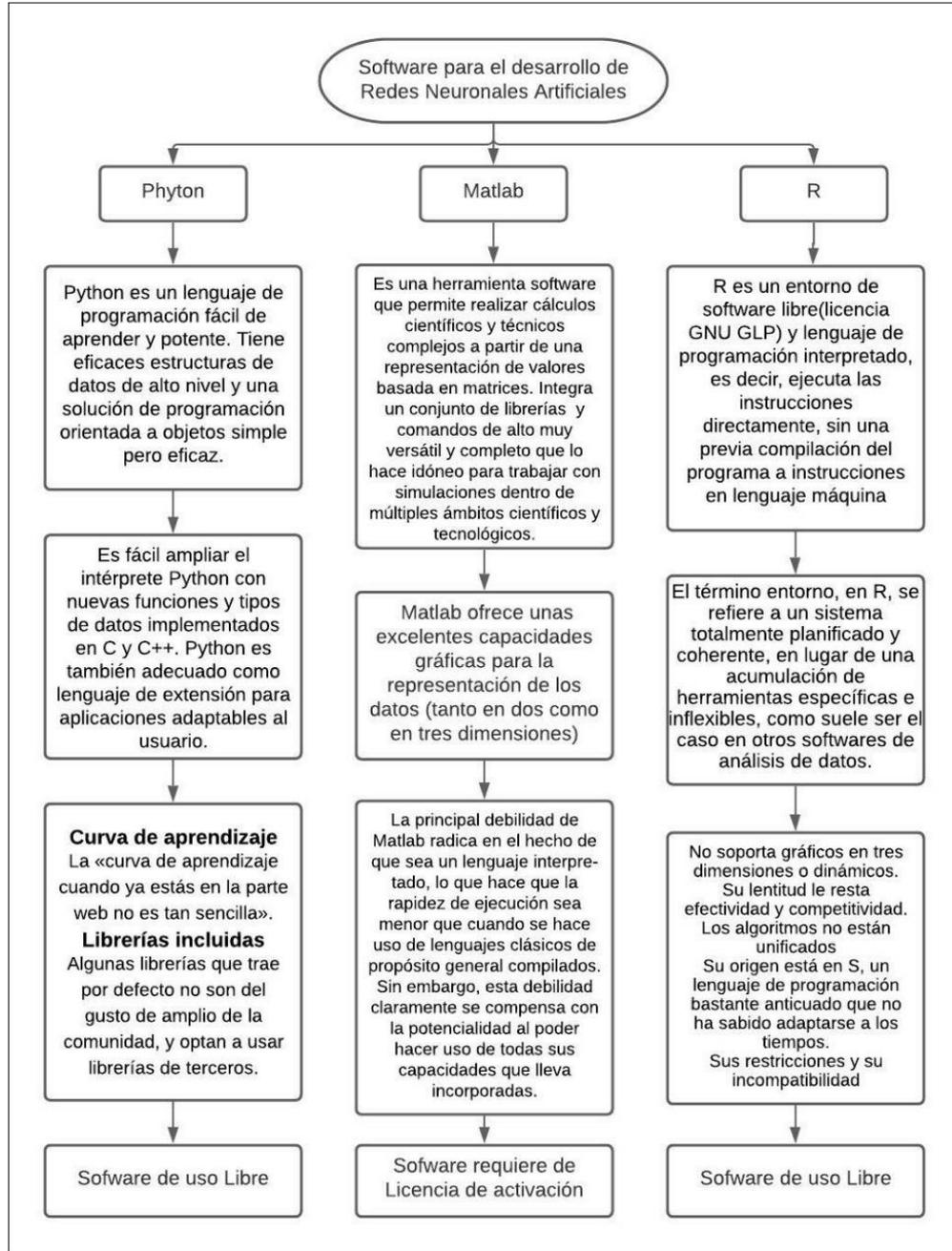


Figura 28: Softwares para el desarrollo de redes neuronales artificiales [18]

### 3.16. LENGUAJE DE PROGRAMACIÓN PYTHON

Python es un lenguaje de programación fácil de aprender y potente. Tiene eficaces estructuras de datos de alto nivel y una solución de programación orientada a objetos simple pero eficaz. La elegante sintaxis de Python, su gestión de tipos dinámica y su naturaleza interpretada hacen

de él el lenguaje ideal para guiones (scripts) y desarrollo rápido de aplicaciones, en muchas áreas y en la mayoría de las plataformas [19].

### **3.16.1. Librerías necesarias para el uso del software de Python**

- a.** Seaborn: es una librería gráfica basada en matplotlib, especializada en la visualización de datos estadísticos. Se caracteriza por ofrecer un interfaz de alto nivel para crear gráficos estadísticos visualmente atractivos e informativos.
- b.** NumPy: es una librería de Python para Machine Learning caracterizada por ser la librería de procesamiento de arrays por excelencia. Contiene una gran colección de funciones que permiten realizar cálculos matemáticos complejos sobre arrays multidimensionales. La funcionalidad que implementa permite realizar operaciones lógicas, redimensiones, búsquedas y aplicar estadísticos entre otras muchas. El núcleo de la librería se basa en los objetos ndarray, los cuales permiten encapsular arrays de n dimensiones sobre los que se pueden realizar las operaciones antes descritas de una forma muy eficiente.
- c.** Pandas: es una de las bibliotecas de Python más útiles para los cálculos de datos. Las principales estructuras de datos en pandas son series de datos unidimensionales y DataFrame para datos bidimensionales.
- d.** PyTorch: es una librería de python, desarrollada por Facebook, que permite el cálculo numérico eficiente en CPU y GPUs.
- e.** Jupyter Notebook: es una aplicación web para crear documentos que contienen código, ecuaciones, visualizaciones y texto. Puedes usar Jupyter notebooks para limpiar datos, transformarlos, realizar simulaciones numéricas, modelos estadísticos, visualizaciones de datos, machine learning y mucho más.
- f.** Anaconda: es una distribución de python para Cálculo Numérico, Análisis de Datos y Machine Learning. Contiene las librerías más usadas por los científicos de datos. Además, hace muy fácil la instalación de otras librerías que puedas necesitar [19].

### 3.16.2. Paquetes especiales para el desarrollo de redes neuronales

- a. TensorFlow: Tensorflow es una de las librerías open source más importantes de Deep Learning y ha sido creada por Google. Está formada por un ecosistema flexible de herramientas, librerías y recursos de la comunidad y ayuda a los investigadores a innovar aplicando técnicas de Machine Learning. Además, permite a los desarrolladores incorporar este tipo de tecnologías en sus aplicaciones.
- b. Keras: es una interfaz de alto nivel para trabajar con redes neuronales. La interfaz de Keras es más fácil de usar que la interfaz de TensorFlow. Esta facilidad de uso es su característica principal.
- c. Matplotlib: Hasta este punto únicamente se han introducido librerías que permiten manipular los datos y realizar algoritmos. Sin embargo, existe un procedimiento igual de importante en los flujos de trabajo y ese es el de visualización. La librería de Python para Machine Learning empleada en tareas de visualización por excelencia es Matplotlib y entre sus cualidades destacan que es open source y trabaja a bajo nivel. Es tal su importancia que otras librerías, como Plotly, se basan en ella.
- d. NumPy: es una librería de Python para Machine Learning caracterizada por ser la librería de procesamiento de arrays por excelencia. Contiene una gran colección de funciones que permiten realizar cálculos matemáticos complejos sobre arrays multidimensionales. La funcionalidad que implementa permite realizar operaciones lógicas, redimensiones, búsquedas y aplicar estadísticos entre otras muchas. El núcleo de la librería se basa en los objetos ndarray, los cuales permiten encapsular arrays de n dimensiones sobre los que se pueden realizar las operaciones antes descritas de una forma muy eficiente.
- e. Scikit-learn: una de las librerías Python más importantes enfocadas al Machine Learning es Scikit-Learn. Esta librería está formada por una gran cantidad de algoritmos de ML (clasificación, regresión, clustering, etc.) y proporciona funcionalidades básicas que facilitan el trabajo diario de los ingenieros dedicados a

este tipo de tareas. Entre sus funcionalidades se puede destacar las herramientas para preprocesar datos, las funciones para evaluar modelos y los mecanismos para ajustar los parámetros de cada modelo [19].

### 3.17. OPTIMIZADORES DE TENSORFLOW

En la Figura 29 se pueden observar los optimizadores que se encuentran inmersos en tensorflow de la librería de Keras.

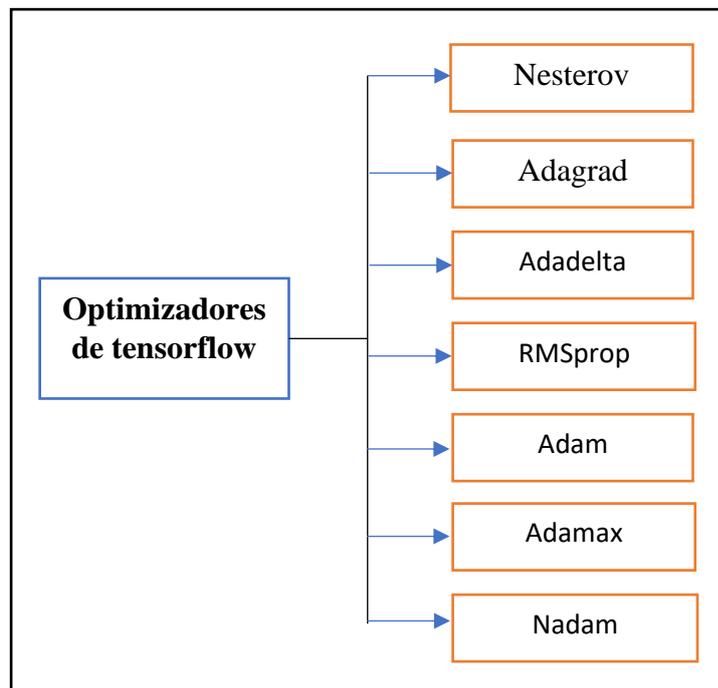


Figura 29: Optimizadores de tensorflow [19]

### 3.18. PROBLEMAS DERIVADOS DEL ENTRENAMIENTO DE LA RED NEURONAL

#### 3.18.1. El problema de la convergencia prematura

La convergencia prematura es un problema muy común en los algoritmos genéticos. Este problema se presenta cuando aparece en una generación un individuo con un nivel de adaptación muy superior al resto de la población. En este caso, el súper-individuo domina rápidamente la población, antes de que se haya podido realizar una exploración suficientemente amplia del espacio de búsqueda. Esta característica se presenta fundamentalmente cuando se hace uso de operadores de selección proporcionales, donde el súper-individuo contribuye con

gran cantidad de copias a la siguiente generación. Así, la población tiende a hacerse homogénea y el algoritmo genético se estabiliza en una solución que puede no ser lo suficientemente buena.

Para evitar este problema se estudiará el uso de operadores de selección alternativos a la selección proporcional, donde todos los individuos tengan posibilidades de pasar a la siguiente generación. También, para garantizar la heterogeneidad de la población, se hará uso de probabilidades de mutación superiores a las utilizadas normalmente en algoritmos genéticos [16].

### 3.18.2. El problema de la permutación

El problema de la permutación es un tema tratado muy frecuentemente en trabajos sobre diseño de topologías de redes neuronales mediante algoritmos genéticos. El origen de este problema se encuentra en la disposición de las neuronas ocultas de una red neuronal. Dos redes neuronales pueden ser funcionalmente equivalentes independientemente del orden de sus neuronas ocultas, como se muestra en la figura 30.

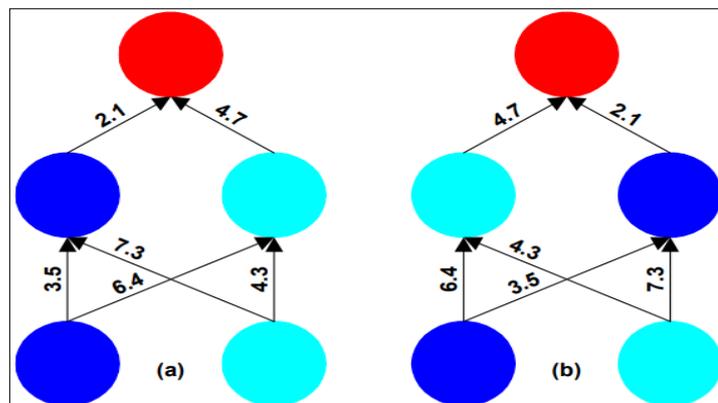


Figura 30: El problema de la permutación [18]

Dado que una red neuronal puede tener varios genotipos que la representan, la probabilidad de que el operador de cruce produzca de ellos un individuo más adaptado es baja. Por esto, varios trabajos no hacen uso de este operador, a pesar de la importancia del mismo [18].

El principal desafío en el uso de redes neuronales es que nuestros algoritmos deben funcionar bien en entradas nuevas, no observadas previamente, no solo en aquellas en las que se ha entrenado nuestro modelo. Esta capacidad de funcionar correctamente en nuevas entradas se llama generalización. Por lo general, en un modelo de red neuronal, se tiene acceso a un conjunto de entrenamiento y se puede calcular alguna métrica de error en ese conjunto, llamada

error de entrenamiento. Lo que separa el aprendizaje automático y el aprendizaje profundo de los problemas de optimización pura es que se espera que el error de generalización también sea pequeño. Este error se define como la expectativa de un nuevo error de entrada. Esta expectativa son las diferentes entradas posibles extraídas de la distribución de entradas que se espere que el sistema encuentre en la práctica. El error de generalización de un modelo generalmente se estima midiendo su desempeño en un conjunto de prueba que contiene ejemplos obtenidos por separado del conjunto de entrenamiento.

Lo que determina qué tan bien funciona un modelo de red neuronal es si puede hacer que el error de entrenamiento sea lo más pequeño posible y que la brecha entre el error de entrenamiento y el error de prueba sea lo más pequeña posible [20].

### 3.18.3. Underfitting

La figura 31 corresponde al underfitting, cuando el modelo no logra obtener un error suficientemente bajo en el conjunto de entrenamiento. El modelo podría no ser lo suficientemente ajustado. Debido a la falta de tiempo de capacitación o datos insuficientes para un modelado confiable. Este es un problema fácil de detectar [21].

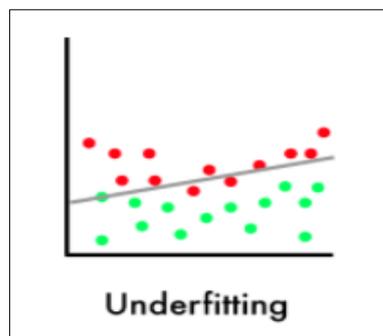


Figura 31: Equilibrio de aprendizaje en underfitting [21]

### 3.18.4. Overfitting

La figura 32 corresponde con el de overfitting (sobreajuste), que ocurre cuando la brecha entre el error de entrenamiento y el error de prueba es demasiado grande. Es posible que la previsibilidad no mejore a medida que el modelo se entrena y se itera.

Matemáticamente hablando, es posible obtener funciones adecuadas para casi cualquier conjunto de datos de entrenamiento. Sin embargo, si se quiere minimizar el error de entrenamiento, el modelo puede volverse complejo rápidamente. Este aumento en la complejidad solo dará como resultado ajustes de peor calidad y mayores costos de tiempo de ejecución durante la generalización del modelo [21].

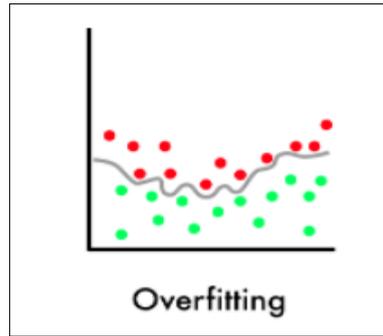


Figura 32: Equilibrio de aprendizaje en overfitting [21]

### 3.18.5. Normalfitting

Este es el punto de entrenamiento donde el modelo se adapta mejor, los errores correspondientes a la secuencia de entrenamiento y generalización deben ser lo más similares posible. Hacer que el modelo sea lo suficientemente complejo para tener un comportamiento robusto como se muestra en la figura 33 [21].



Figura 33: Equilibrio de aprendizaje en normalfitting [21]

## 3.19. COMPORTAMIENTO DE LA COMPUTADORA AL ENTRENAR LA RED NEURONAL

A continuación, se detallará el rendimiento de la computadora para entrenar la red neuronal.

### 3.19.1. Características del computador utilizado

- Windows 10
- NVIDIA GeForce GTX 1050Ti 4 GB GDDR5
- Core i7-8550U CPU
- 8GB DDR4 RAM

### 3.19.2. Rendimiento del CPU

En la figura 34, se puede observar el rendimiento de la computadora cuando se tiene un número reducido de epoch, en este caso 3, se observa que el rendimiento del CPU no aumenta ya que el entrenamiento de la red neuronal artificial no tiene mayor peso con la estructura seleccionada.

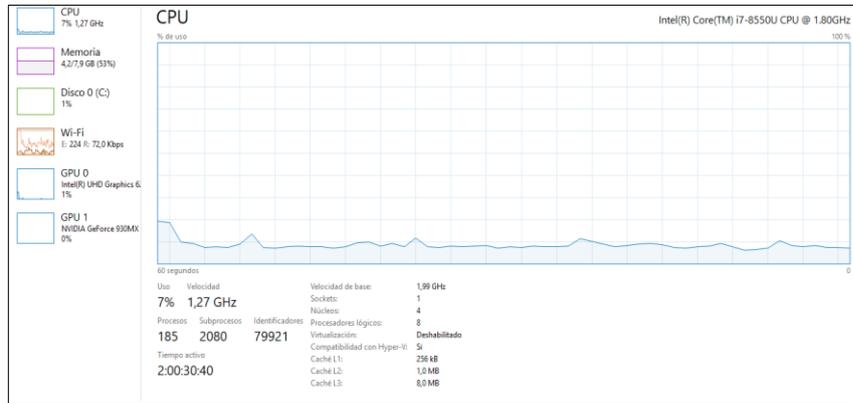


Figura 34: Rendimiento del CPU con epoch de 3 [autores]

En la figura 35, se puede observar el rendimiento de la computadora cuando se tiene un número recomendado de epoch, en este caso 10. Se observa que el rendimiento del CPU aumenta con respecto a la anterior estructura ya que el entrenamiento de la red neuronal artificial tiene mayor peso con la estructura seleccionada

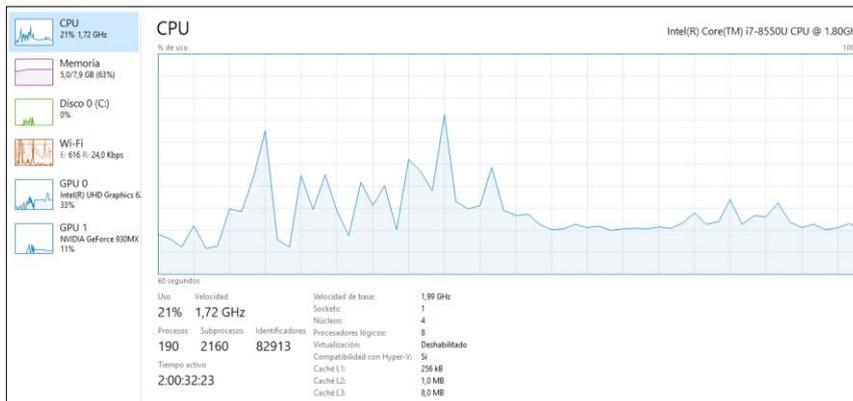


Figura 35: Rendimiento del CPU con epoch de 10 [autores]

Se utilizó la estructura de la figura 35, ya que se obtuvo resultados positivos con 10 epoch, esta estructura va acorde con los componentes del computador, obteniendo un rendimiento favorable para entrenar la red neuronal. Según el análisis efectuado los más apropiado es utilizar 10 epoch.

En la gráfica 36, se puede observar el rendimiento de la computadora cuando se tiene un número elevado de epoch, en este caso 50. Se observa que el rendimiento del CPU aumenta con respecto a la anterior estructura ya que el entrenamiento de la red neuronal artificial tiene mayor peso con la estructura seleccionada para este ejemplo se utilizó una estructura muy pesada para observar los resultados.



Figura 36: Rendimiento del CPU con epoch de 50 [autores]

El número de epoch de esta estructura ocasiona un incremento de memoria excesivo provocando así la saturación del computador.

## **4. MATERIALES Y MÉTODOS**

### **4.1. TIPO DE INVESTIGACIÓN**

Este proyecto de investigación describe la situación de los sistemas de distribución, haciendo referencia a la predicción de la generación de energía eléctrica, en la cual los generadores trabajan para satisfacer la demanda de energía, por lo que es necesario proponer alternativas que puedan predecir la generación de energía eléctrica, para obtener la curva de generación de potencia del día siguiente.

#### **4.1.1. Investigación descriptiva**

Utilizando un enfoque analítico, es decir, desglosando el objeto de aprendizaje en puntos para lograr una comprensión más técnica de hechos e ideas sobre el comportamiento de la producción de electricidad, en la central hidroeléctrica Illuchi 1 tales estudios ayudan en el análisis del comportamiento de generación de energía ayudando a obtener una base de datos histórica con valores de potencia por hora.

#### **4.1.2. Investigación diagnóstica**

Este tipo de investigación ayuda a caracterizar un problema que abarca un área específica, y a comprender la situación actual en la que se encuentra la central hidroeléctrica, identifica factores para profundizar en el estudio de la pregunta planteada, obtiene los datos obtenidos del análisis para lograr la producción eléctrica prevista solución alternativa al problema.

#### **4.1.3. Investigación aplicada**

En este punto se identificarán los conocimientos científicos adquiridos, así como el análisis y programación de los algoritmos mediante el software Python, que permitan predecir la producción de energía eléctrica con resultados cercanos a la realidad.

### **4.2. MÉTODO DE INVESTIGACIÓN**

#### **4.2.1. Metodología de la investigación bibliográfica.**

Este tipo de investigación contribuye a la concreción del análisis, el cual se basará en los antecedentes de consulta de artículos técnicos, libros, publicaciones en internet y proyectos similares que involucren pronósticos de generación eléctrica. Lo que proporciona un razonamiento teórico que facilita la votación sobre el procedimiento adecuado para la solución del problema.

#### **4.2.2. Inductivo – deductivo**

Examina todos los procedimientos, temas y subtemas utilizados en la elaboración del trabajo de investigación de manera descriptiva, mientras que el procedimiento permite una descripción detallada de las conclusiones y recomendaciones de la pregunta que se investiga. En cuanto a los métodos utilizados, se analiza cada método de abordar los pronósticos de generación a corto plazo en base a un análisis de encuestas que se han realizado en diferentes escenarios, pero con la misma consistencia, donde a su vez ayuda a implementar un algoritmo que nos permite obtener los datos necesarios para construir nuestras propias inferencias para recomendar procesos que pueden usarse para tales investigaciones.

### **4.3. TÉCNICAS E INSTRUMENTOS**

#### **4.3.1. Observación de la red neuronal**

Esta técnica es la más adecuada para la investigación, ya que la realización parcial de la red neuronal artificial corresponde al registro de los datos obtenidos durante el proceso, con resultados satisfactorios.

#### **4.3.2. Simulación de la red neuronal**

Se desarrollará utilizando el software Python que nos ayudará a determinar los parámetros de la red neuronal artificial de referencia y la base de datos histórica de la central hidroeléctrica Illuchi 1 para validar los datos con el algoritmo propuesto y determinar los límites de error existentes.

#### **4.3.3. Programación de la red neuronal**

Se llevará a cabo mediante un software Python que permitirá diseñar algoritmos basados en vectores y fórmulas matemáticas, aplicados a redes neuronales artificiales, en una base de datos de historial de potencia eléctrica, lo que permitirá identificar el porcentaje de error en la previsión de generación de energía. electricidad.

#### 4.4. DIAGRAMA DE FLUJO PARA EL DESARROLLO DE LA RED NEURONAL

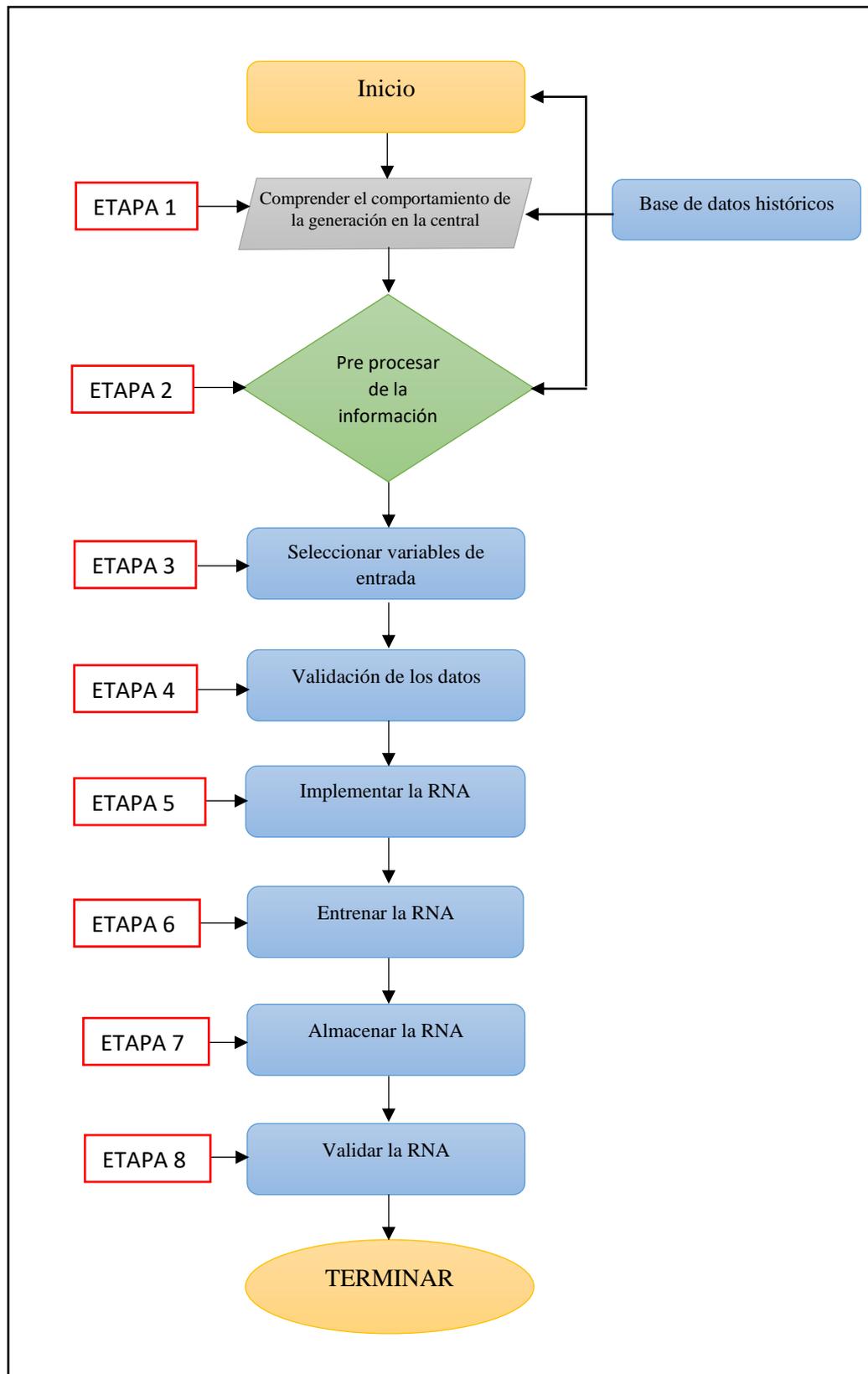


Figura 37: Diagrama de flujo del desarrollo de la RNA [autores]

A continuación, en la figura 37 Se resumen ocho etapas de predicción de energía eléctrica utilizando métodos de redes neuronales artificiales.

- **Etapa 1.** Entender el comportamiento de la energía producida se trata de encontrar toda la información (curva de generación diaria) que ayude a comprender mejor el funcionamiento del generador. Como resultado final de este paso, permitirá conocer la base de datos obtenida.
- **Etapa 2.** Preprocesamiento de datos de entrada. La red neuronal artificial aprende y hace predicciones en base a los datos históricos de las variables bajo consideración, si hay datos erróneos en estas bases de datos, los resultados pronosticados no serán los más precisos, por eso es necesario eliminar la mayoría de los datos erróneos antes de ingresar la base de datos históricos.
- **Etapa 3.** Seleccione la variable de entrada de la base de datos. Con base en el conocimiento adquirido en la primera etapa, es necesario seleccionar las variables que tienen mayor influencia en el comportamiento de generación de energía.
- **Etapa 4.** La validación de los datos consiste en garantizar que los datos se hayan sometido a una limpieza y que estos sean correctos y útiles para su posterior uso.
- **Etapa 5.** Implementación de una red neuronal artificial para pronosticar la generación de energía a través de un algoritmo de programación en Python para proporcionar resultados de pronóstico diarios.
- **Etapa 6.** Entrenamiento de la red neuronal artificial con conjunto de datos de entrenamiento (criterios de aprendizaje, número de iteraciones). En este paso se asignan valores finales, incluyendo el número de neuronas, el retraso o época y el algoritmo de entrenamiento.
- **Etapa 7.** Los parámetros de la red neuronal artificial se almacenan una vez verificados. Es necesario almacenar la red neuronal artificial para uso futuro sin repetir la etapa 6.

- **Etapa 8.** Validar la red neuronal artificial utilizando el conjunto de datos de prueba. La validación de la función de la red neuronal artificial se lleva a cabo prediciendo el valor de generación de energía y luego comparándolo con los datos reales.

## 5. ANÁLISIS Y DISCUSIÓN DE LOS RESULTADOS.

En este punto se seleccionará las variables de entrada de los datos que son: (año, mes, día, fecha, hora y caudal) desde el año 2 015 al 2 019, se procesa y valida estos datos, se define la estructura de la red neuronal artificial, así como entrenarla y almacenarla para su posterior implementación.

### 5.1. COMPRENDER EL COMPORTAMIENTO DE LA GENERACIÓN EN LA CENTRAL HIDROELÉCTRICA ILLUCHI 1

En la siguiente figura 38 se puede apreciar el consumo de energía en los 12 meses de los años en estudio que son del 2 015 al 2 019, encontrando así que los meses con mayor consumo son: enero, febrero, abril, julio, agosto, octubre, noviembre y diciembre. Los meses que tiene menor consumo son los de marzo, mayo, junio y septiembre.

En los meses nombrados con anterioridad el consumo de energía es menor, ya que son meses en los que al no existir demasiadas festividades solo trabaja un par de los cuatro generadores, cumpliendo así con lo demandado.

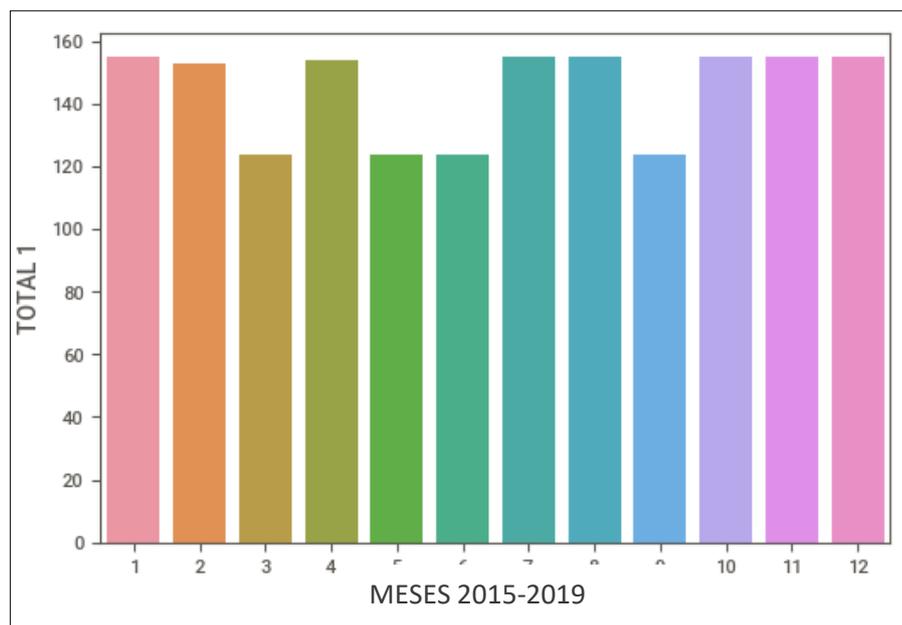


Figura 38: Consumo de la energía mensual del 2 015 al 2 019 [autores]

### 5.2. ANÁLISIS PARA LA CORRELACIÓN DE LAS VARIABLES

La correlación es una técnica usada para determinar la relación entre dos o más variables. El objetivo es conocer si alguna de las variables se relaciona entre sí, o si tienen comportamientos similares.

En la figura 39 se muestra las relaciones entre las múltiples variables, las áreas que están de color azul se encuentran en perfecta relación, además se puede apreciar que en la generación eléctrica existe una relación entre el generador 3 con el día mes y año, el generador 1 es independiente de los meses al igual que el 4, por otra parte, el generador 2 depende del año.

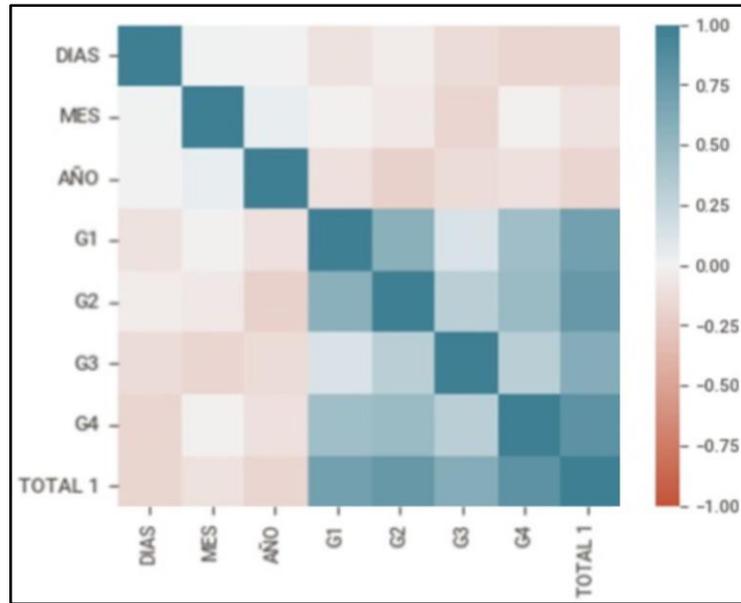


Figura 39: Diagrama de calor de relación de las variables [autores]

En la figura 40 se muestran las variables de entrada seleccionadas para el desarrollo de la red neuronal artificial, las cuales son (días, mes, año, G1, G2, G3, G4, Total) se seleccionaron dichas variables con el fin de compararlas si estas tienen un comportamiento similar quiere decir que están correlacionadas.

	DIAS	MES	AÑO	G1	G2	G3	G4	TOTAL 1
DIAS	1.000000	0.003133	-0.002040	-0.098510	-0.044317	-0.138827	-0.186510	-0.172763
MES	0.003133	1.000000	0.039124	-0.022585	-0.049879	-0.181950	-0.020962	-0.089533
AÑO	-0.002040	0.039124	1.000000	-0.111559	-0.203570	-0.137575	-0.110676	-0.186470
G1	-0.098510	-0.022585	-0.111559	1.000000	0.570815	0.119605	0.440971	0.705430
G2	-0.044317	-0.049879	-0.203570	0.570815	1.000000	0.298451	0.477529	0.770254
G3	-0.138827	-0.181950	-0.137575	0.119605	0.298451	1.000000	0.311604	0.587314
G4	-0.186510	-0.020962	-0.110676	0.440971	0.477529	0.311604	1.000000	0.826821
TOTAL 1	-0.172763	-0.089533	-0.186470	0.705430	0.770254	0.587314	0.826821	1.000000

Figura 40: Correlación de las variables [autores]

### 5.3. PREPROCESAMIENTO DE LA INFORMACIÓN

#### 5.3.1. Adquisición y almacenamiento de la información

Se muestra la base de datos históricos reales que contienen los datos de corriente, potencia y tensión de los cuatro generadores que cuenta la Hidroeléctrica Illuchi 1 en un rango del año 2 015 al año 2 019, los cuales serán necesarios para nuestra predicción.

Los datos se encuentran representados tanto por horas, días, meses y años. Tomando los datos de las 24 horas del día para tener una predicción más acertada.

Los datos están en un archivo de Excel, este es compatible con Python, así que se puede exportar dichos datos para comenzar con la programación.

4	HORAS	GENERADOR No.1					GENERADOR No.2					GENERADOR No.3					GENERADOR No.4			
		I (A)	P (KW)	E <sub>L</sub> (KV-H)	E <sub>G</sub> (KV-H)	Q (KVARS)	I (A)	P (KW)	E <sub>L</sub> (KV-H)	E <sub>G</sub> (KV-H)	Q (KVARS)	I (A)	P (KW)	E <sub>L</sub> (KV-H)	E <sub>G</sub> (KV-H)	Q (KVARS)	I (A)	P (KW)	E <sub>L</sub> (KV-H)	
6	0:00			5.691.100	0			6.783.200	0				1.895.600	0						177.600
7	0:30	90	400			57	125	600			85	270	1.100			157	300	1.300		
8	1:00	90	400	5.691.500	400	57	125	600	6.783.800	600	85	270	1.100	1.896.700	1.100	157	300	1.300	179.100	
9	1:30	90	400			57	125	600			85	270	1.100			157	300	1.300		
10	2:00	90	400	5.691.900	400	57	125	600	6.784.400	600	85	270	1.100	1.897.800	1.100	157	300	1.300	180.400	
11	2:30	90	400			57	125	600			85	270	1.100			157	300	1.300		
12	3:00	90	400	5.692.300	400	57	125	600	6.785.000	600	85	270	1.100	1.898.900	1.100	157	300	1.300	181.700	
13	3:30	90	400			57	125	600			85	270	1.100			157	300	1.300		
14	4:00	90	400	5.692.700	400	57	125	600	6.785.600	600	85	270	1.100	1.900.000	1.100	157	300	1.300	183.000	
15	4:30	90	400			57	125	600			85	270	1.100			157	300	1.300		
16	5:00	90	400	5.693.100	400	57	125	600	6.786.200	600	85	270	1.100	1.901.100	1.100	157	300	1.300	184.300	
17	5:30	90	400			57	125	600			85	270	1.200			171	300	1.400		
18	6:00	90	400	5.693.500	400	57	125	600	6.786.800	600	85	270	1.200	1.902.300	1.200	171	300	1.400	185.700	
19	6:30	90	400			57	125	600			85	270	1.200			171	300	1.400		
20	7:00	90	400	5.693.900	400	57	125	600	6.787.400	600	85	270	1.200	1.903.500	1.200	171	300	1.400	187.100	
21	7:30	90	400			57	125	600			85	270	1.200			171	300	1.400		
22	8:00	90	400	5.694.300	400	57	125	600	6.788.000	600	85	270	1.200	1.904.700	1.200	171	300	1.400	188.500	
23	8:30	90	400			57	125	600			85	280	1.200			171	300	1.400		
24	9:00	90	400	5.694.700	400	57	125	600	6.788.600	600	85	280	1.200	1.905.900	1.200	171	300	1.400	189.900	
25	9:30	90	400			57	125	600			85	280	1.200			171	300	1.400		
26	10:00	90	400	5.695.100	400	57	125	600	6.789.200	600	85	280	1.200	1.907.100	1.200	171	300	1.400	191.300	

Figura 41: Base de datos de los Generadores Illuchi 1 [autores]

### 5.4. PREPROCESAMIENTO DE LOS DATOS

En la figura 42 se observa el diagrama para el procesamiento de los datos históricos el cual es necesario para obtener una nueva base de datos para su posterior uso en la programación y testeo, ya que se requieren datos limpios para obtener resultados acertados y poder validar los mismos.

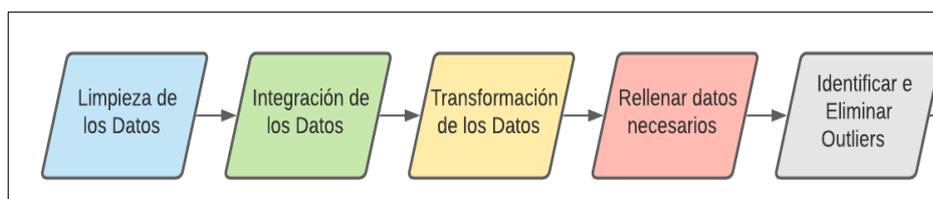


Figura 42: Diagrama del preprocesamiento de datos históricos [autores]

Es necesario realizar una depuración de los datos nombrados en la figura 41 de los cuales, se realizó un Excel que se muestra en la figura 43 y cuenta con: Año, Mes, Día, Fecha, Hora, Caudal, desde enero del 2 015 hasta diciembre del 2 019. Estos datos serán la base para la programación en Python.

	A	B	C	D	E	F
1	AÑO	MES	DIA	FECHA	HORA	Caudal
2	2015	1	jueves	1	0:00:00	1100,7567
3	2015	1	jueves	1	0:30:00	1100,7567
4	2015	1	jueves	1	1:00:00	1100,7567
5	2015	1	jueves	1	1:30:00	742,7767
6	2015	1	jueves	1	2:00:00	742,7767
7	2015	1	jueves	1	2:30:00	742,7767
8	2015	1	jueves	1	3:00:00	742,7767
9	2015	1	jueves	1	3:30:00	742,7767
10	2015	1	jueves	1	4:00:00	742,7767
11	2015	1	jueves	1	4:30:00	742,7767
12	2015	1	jueves	1	5:00:00	742,7767
13	2015	1	jueves	1	5:30:00	742,7767
14	2015	1	jueves	1	6:00:00	742,7767
15	2015	1	jueves	1	6:30:00	742,7767
16	2015	1	jueves	1	7:00:00	742,7767
17	2015	1	jueves	1	7:30:00	742,7767

Figura 43: Datos históricos preprocesados [autores]

	A	B	C	D	E	F	G	H	I	J
1	AÑO	MES	DIA	HORA	G1	G2	G3	G4	Potencial Total	Caudal Total
2	2015	1	1	0:00:00	0	500	1.000	1.200	2.700	1100,7567
3	2015	1	1	0:30:00	0	500	1.000	1.200	2.700	1100,7567
4	2015	1	1	1:00:00	0	500	700	800	2.000	742,7767
5	2015	1	1	1:30:00	0	500	700	800	2.000	742,7767
6	2015	1	1	2:00:00	0	500	700	800	2.000	742,7767
7	2015	1	1	2:30:00	0	500	700	800	2.000	742,7767
8	2015	1	1	3:00:00	0	500	700	800	2.000	742,7767
9	2015	1	1	3:30:00	0	500	700	800	2.000	742,7767
10	2015	1	1	4:00:00	0	500	700	800	2.000	742,7767
11	2015	1	1	4:30:00	0	500	700	800	2.000	742,7767
12	2015	1	1	5:00:00	0	500	700	800	2.000	742,7767
13	2015	1	1	5:30:00	0	500	700	800	2.000	742,7767
14	2015	1	1	6:00:00	0	500	700	800	2.000	742,7767
15	2015	1	1	6:30:00	0	500	700	800	2.000	742,7767
16	2015	1	1	7:00:00	0	500	700	800	2.000	742,7767
17	2015	1	1	7:30:00	0	500	700	800	2.000	742,7767

Figura 44: Datos históricos preprocesados incluido los generadores [autores]

#### 5.4.1. Análisis de la base de datos históricos de los generadores

Para los cálculos de potencia y caudal se facilitó las tablas con datos de los 4 generadores, con esos datos se obtuvo una ecuación para cada generación en la cual se reemplazó los valores de potencia, obteniendo así el caudal buscado.

En la tabla 11 se observan los valores de la relación de potencia y caudal de los 4 generadores, los cuales se utilizarán a continuación para buscar una ecuación y así poder sustituir un nivel de potencia y obtener el caudal deseado en cada uno de los generadores.

Tabla 11: Potencia y Caudal de los 4 generadores [autores]

G1		G2		G3		G4	
Potencia l (KW)	Caudal (l/s)	Potencial (KW)	Caudal (l/s)	Potencial (KW)	Caudal (l/s)	Potencial (KW)	Caudal (l/s)
164,61	75,466	140,454	74,03099995	325,109	63,137	298,794	63,113
244,054	122,95	333,441	134,6900008	620,825	190,125	648,012	192,66
366,358	196,747	415,077	183,804999	939,947	345,492	862,295	304,476
566,203	294,463	579,144	232,5499972	1250,094	518,608	1209,636	491,926
714,412	371,53	820,723	367,7410017	1274,591	553,6949998	1307,234	559,212

**a. Generador 1**

En la figura 45, se observa la curva y sus diferentes puntos de la relación entre potencia y caudal del generador 1, alcanzando así un máximo de 371,53 (l/s) de caudal y 714,412 (KW) de potencia y un mínimo de 75,466 (l/s) de caudal y 164,61 (KW) de potencia, finalmente se adquiere una ecuación para ingresar la potencia y obtener así el caudal deseado. La función lineal es ( $y = 0,534x - 7,315$ ) en donde x es la variable a remplazar.

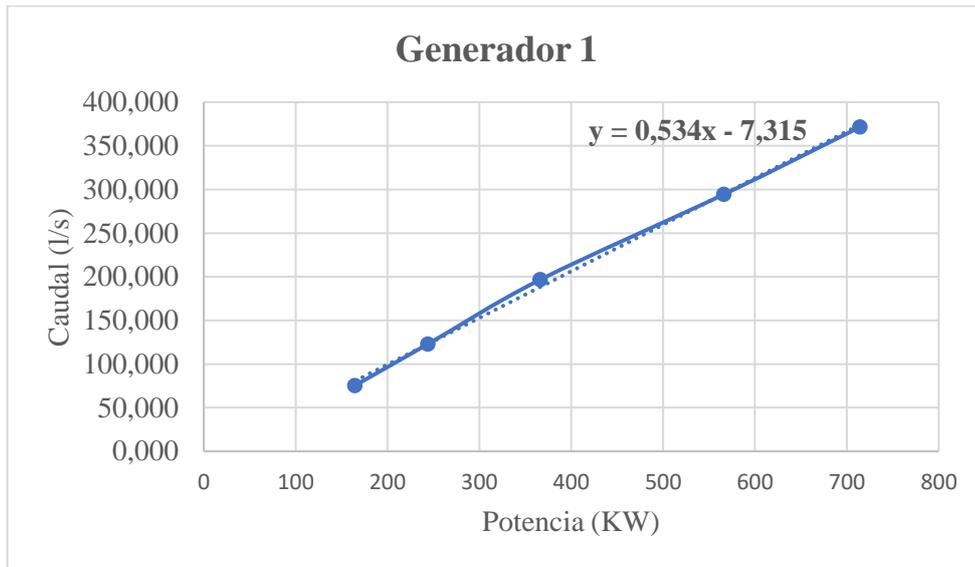


Figura 45: Ecuación del generador 1 [autores]

**b. Generador 2**

En la figura 46, se observa la curva y sus diferentes puntos de la relación entre potencia y caudal del generador 2, alcanzando así un máximo de 367,741 (l/s) de caudal y 820,723 (KW) de potencia y un mínimo de 74,030 (l/s) de caudal y 140,454 (KW) de potencia, finalmente se adquiere una ecuación para ingresar la potencia y obtener así el caudal deseado. La función lineal es ( $y = 0,4297x - 1,8817$ ) en donde x es la variable a remplazar.

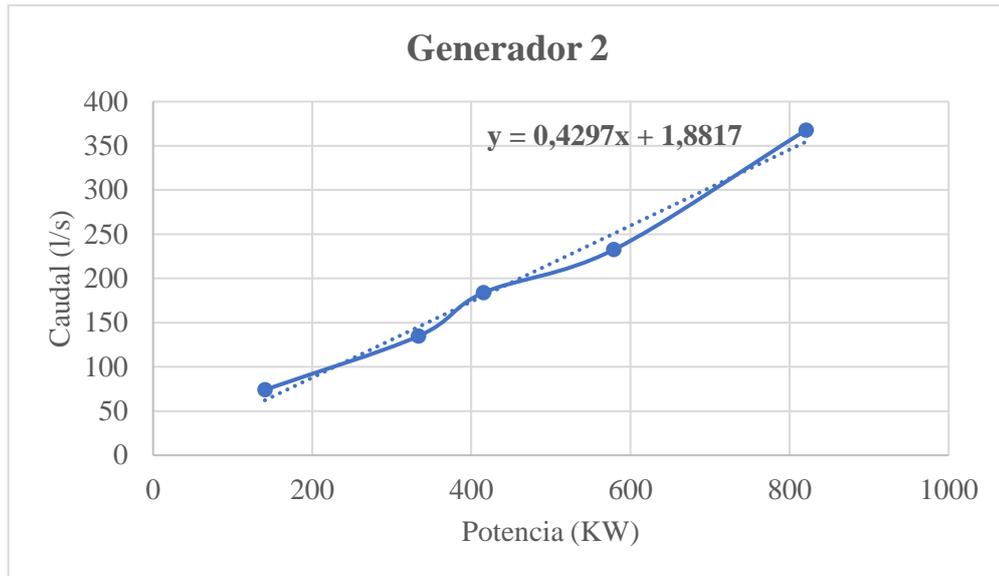


Figura 46: Ecuación del generador 2 [autores]

**c. Generador 3**

En la figura 47, se observa la curva y sus diferentes puntos de la relación entre potencia y caudal del generador 3, alcanzando así un máximo de 553,694 (l/s) de caudal y 1 274,591 (KW) de potencia y un mínimo de 63,137 (l/s) de caudal y 325,109 (KW) de potencia, finalmente se adquiere una ecuación para ingresar la potencia y obtener así el caudal deseado. La función lineal es ( $y = 0,5114x - 116,87$ ) en donde x es la variable a remplazar.

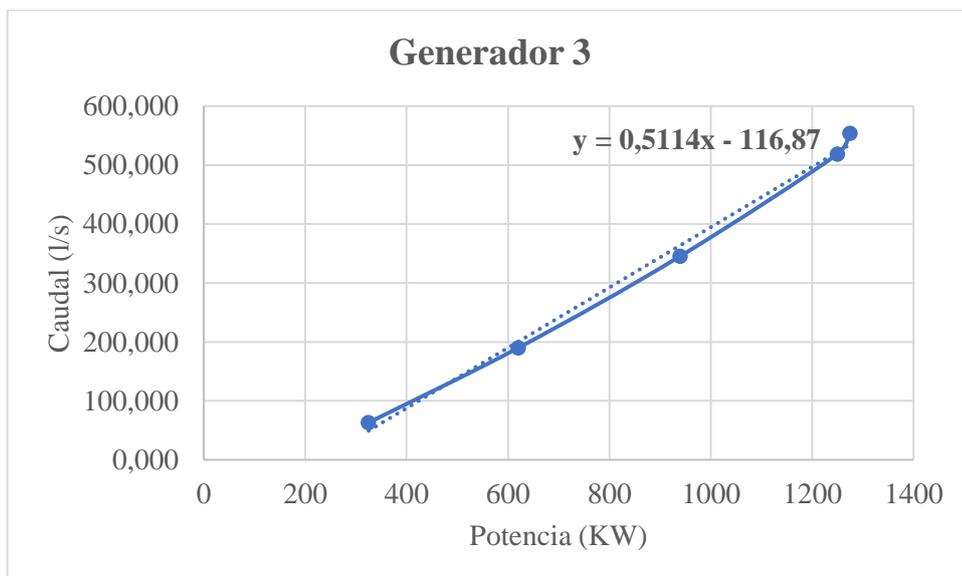


Figura 47: Ecuación del generador 3 [autores]

#### d. Generador 4

En la figura 48, se observa la curva y sus diferentes puntos de la relación entre potencia y caudal del generador 3, alcanzando así un máximo de 559,212 (l/s) de caudal y 1 307,234 (KW) de potencia y un mínimo de 63,113 (l/s) de caudal y 298,794 (KW) de potencia, finalmente se adquiere una ecuación para ingresar la potencia y obtener así el caudal deseado. La función lineal es ( $y = 0,4958x - 106,7$ ) en donde x es la variable a remplazar.

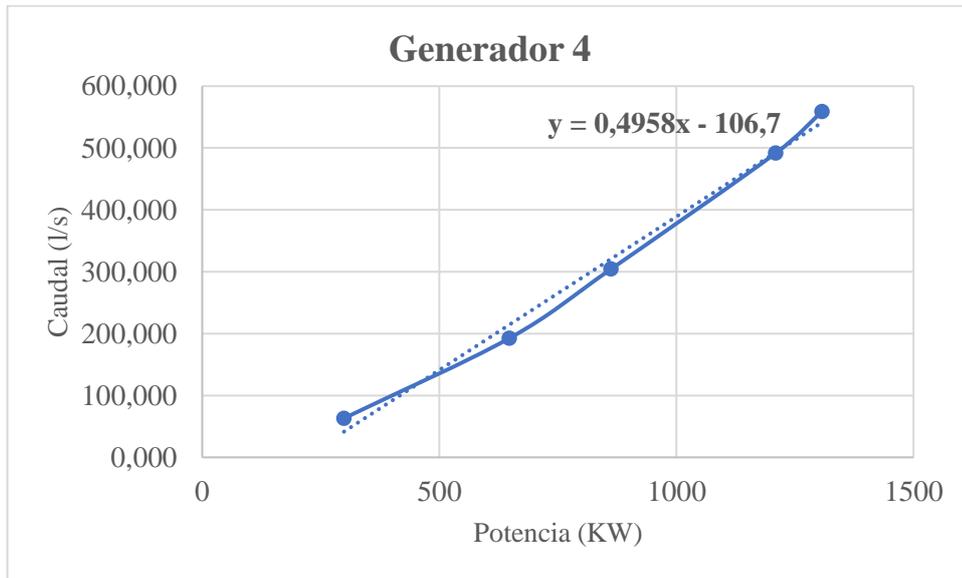


Figura 48: Ecuación del generador 4 [autores]

### 5.5. SELECCIÓN DE VARIABLES DE ENTRADA

Una vez que se preprocesaron los datos de la figura 43, se tuvo que filtrar varios datos de información para el procesamiento de datos, ya que toda la información de la figura 41 era inútil, se realizaron los cambios que se describen a continuación:

- Se identifican los indicadores importantes, los datos que entregan información a la red neuronal para el entrenamiento y las pruebas posteriores.
- Los datos que constan en la nueva matriz de información se muestran en la tabla 12.

Tabla 12: Variables de entrada de los datos seleccionados [autores]

Variables	Unidad de medición
Fecha (año, mes, día, hora)	Tiempo
Caudal	(l/s)

## 5.6. VALIDACIÓN DE LOS DATOS HISTÓRICOS

Los datos de entrenamiento se basan en los datos históricos desde el 2 015 al 2 019. Para definir el número correcto de datos para el entrenamiento se deben cumplir las siguientes características:

- a. El conjunto debe tener una dimensión apropiada para generar datos significativos.
- b. Debe ser representativo de todo el conjunto de datos.

Se dividieron los datos en un 80 % que son los datos históricos y 20 % datos de validación.

## 5.7. IMPLEMENTACIÓN EN PYTHON

### 5.7.1. Librerías utilizadas

Se importaron las siguientes librerías, que son utilizadas para la programación y entramiento de redes neuronales, además para observar todas las gráficas del proceso. Entre ellas están pandas, numpy, matplotlib, keras, tensorFlow entre otras.

Para el entrenamiento de redes neuronales artificiales, se utilizó los algoritmos de entrenamiento de la biblioteca Keras Python, una biblioteca a nivel de modelo que proporciona bloques de construcción de alto nivel para desarrollar modelos de aprendizaje profundo. No maneja operaciones de bajo nivel como productos tensoriales. En su lugar, se basa en una biblioteca especializada y bien optimizada de operaciones de tensor para hacer esto.

```
In [1]: import pandas as pd
import numpy as np
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
```

Figura 49: Librerías de Python [autores]

### 5.7.2. Procesamiento de datos

#### a. Lectura de datos

Para poder realizar la lectura de datos se hizo el llamado de la base de datos, que constan del año (2 015 – 2 019) con las variables de entrada seleccionadas que son: año, mes, día, fecha, hora y caudal, el llamado se realizó en formato de Excel.

```
In [4]: datos = pd.read_excel(io='data/datos.xlsx', sheet_name = "Hoja1")
datos

Out[4]:
```

	AÑO	MES	DIA	FECHA	HORA	Caudal
0	2015	1	1	jueves	00:00:00	1100.7567
1	2015	1	1	jueves	00:30:00	1100.7567
2	2015	1	1	jueves	01:00:00	1100.7567
3	2015	1	1	jueves	01:30:00	742.7767
4	2015	1	1	jueves	02:00:00	742.7767

Figura 50: Llamado de datos preprocesados [autores]

### b. Verificación de nulos

Se procede a verificar si la base de datos contiene valores nulos, al existir valores erróneos se causan problemas en la programación ocasionando errores que conllevan a una mala predicción, porque esta influye en el entrenamiento de la red neuronal artificial.

Como se observa en la figura 51, no existen valores nulos, entonces se puede seguir con el siguiente paso de la programación de la red neuronal artificial.

```
In [5]: datos.isnull().sum()

Out[5]: AÑO      0
        MES      0
        DIA      0
        FECHA    0
        HORA     0
        Caudal   0
        dtype: int64
```

Figura 51: Verificación de valores nulos [autores]

### c. Fecha como índice

Como se observa en la figura 52, para la indexación se tomó la hora y la fecha, creando un nuevo DataFrame con el formato de fecha y valores de caudal correctos, y así poder seguir con el siguiente paso de la programación.

```

In [6]: #Llevamos la hora y la fecha para los indices, por lo que nos interesa es el caudal
from datetime import datetime
datos_indexados = pd.DataFrame()
fechas_dt = []
caudal_dt = []
for registro in datos.values:
    fecha = "{}-{}-{} {}".format(registro[0], registro[1], registro[2], registro[4])
    fecha_dt = datetime.strptime(fecha, '%Y-%m-%d %H:%M:%S')
    fechas_dt.append(fecha_dt)
    caudal_dt.append(registro[5])
#Creamos un nueva DataFrame con el valor correcto de fecha y caudal
datos_indexados['FECHA'] = fechas_dt
datos_indexados['CAUDAL'] = caudal_dt

datos_indexados.set_index('FECHA', drop=True, inplace = True)

datos_indexados

```

Figura 52: Valor correcto de la fecha y el caudal [autores]

#### d. División y normalización de los datos

Como se observa en la figura 53, se tomó el 20 % del final de los datos, los cuales sirven para la validación de la red neuronal artificial y el 80 % restante para su entrenamiento.

Como se observa en la figura 54, se dividió los datos de entrenamiento y de test. La red es recurrente es necesaria una normalización de los datos, para así obtener una base de datos normalizada, comprimida y agrupada.

```

In [7]: #Dividir los datos de entrenamiento y de test
division = int(0.2*datos_indexados.shape[0])
fecha_fin_train = '2019-04-01 00:00:00'
datos_train = datos_indexados[:datos_indexados.index.get_loc(fecha_fin_train)[0]]
datos_test = datos_indexados[datos_indexados.index.get_loc(fecha_fin_train)[0]:]
datos_train

```

Figura 53: División de los datos [autores]

```

In [10]: from sklearn.preprocessing import MinMaxScaler
sc = MinMaxScaler(feature_range = (0,1))
datos_train_scaled = sc.fit_transform(datos_train)
print(datos_train_scaled.shape)
datos_train_scaled

```

Figura 54: Normalización de los datos [autores]

#### f. Agrupamiento de los datos

Se debe entrenar la red por bloque de datos, como se puede observar en la figura 55, se utiliza el `time_step`, para poder entrenar por bloque de datos consecutivos, además se observa que se dividió los datos en bloques de 70.

En este caso se entrenan 70 datos y se hace la predicción para el siguiente dato.

```
In [11]: #Se debe entrenar la red por bloques de datos
time_step = 70 #Se utiliza en time_step para poder entrenar por bloques de datos consecutivos
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.shape, Y_train.shape
```

Figura 55: Agrupamiento de datos [autores]

### g. Reshape de los datos

Para trabajar con redes neuronales en Keras como campo obligatorio se debe tener un array de 3 dimensiones, por lo que se aumentó una nueva dimensión como se observa en la figura 56.

```
In [12]: #Buscamos un array de 3 dimensiones
X_train = np.reshape(X_train, (X_train.shape[0], X_train.shape[1], 1))
X_train.shape, Y_train.shape
```

Figura 56: Reshape de los datos [autores]

### 5.7.3. Implementación de red LSTM

Como se observa en la figura 57 se inicia con un modelo de red secuencial para que la red neuronal inicie en vacío, y así poder añadir los siguientes modelos:

Se tiene la primera capa de entrada con una neurona, seguido de una segunda capa oculta, además se tienen dos capas de congelamiento de neuronas que congela el 40 % y 20 % de las neuronas respectivamente para evitar el overffiting. En la cuarta y última capa se tiene una neurona de salida.

```
In [13]: from keras.layers import LSTM
from sklearn.metrics import mean_squared_error

In [2]: def MODEL_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(Dropout(0.2))
model.add(Dense(units = dim_salida))
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 = 64)
return model, history
```

Figura 57: Implementación de red LSTM [autores]

En la figura 58 se realizó de manera grafica la estructura de la red neuronal artificial utilizada la cual consta de una capa de entrada con 10 neuronas una capa oculta con 10 neuronas y una

capa de salida, adicionalmente se agregaron dos capas de congelamiento para evitar la memorización de los datos.

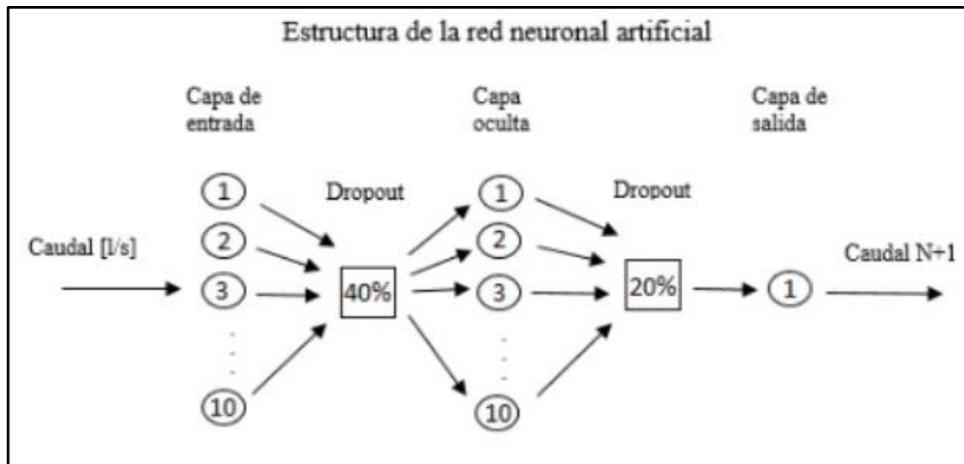


Figura 58: Estructura de la red neuronal artificial [autores]

#### 5.7.4. Entrenamiento de la red neuronal

Se utilizó el optimizador RMSPRO, con 10 neuronas y 10 épocas para el entrenamiento. Se evitó el overffiting gracias a la capa de congelamiento nombrada anteriormente, obteniéndose un normalffiting.

#### 5.7.5. Verificación de la red neuronal artificial

La validación se realiza con un conjunto de datos de prueba, y los errores absolutos se calculan con diferentes optimizadores entrenados en consecuencia para aplicarlos a redes neuronales artificiales.

Después de realizar los entrenamientos con los optimizadores que brinda la librería de Keras los optimizadores que se muestran en la figura 59 fueron los que obtuvieron mejores resultados.

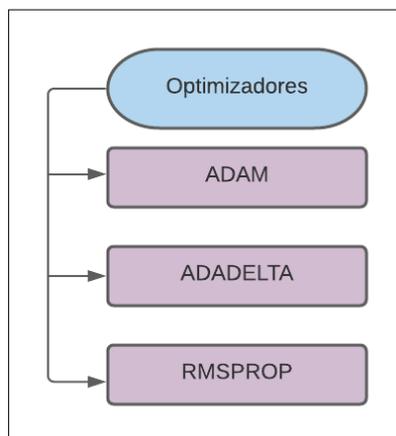


Figura 59: Optimizadores seleccionados para el entrenamiento de la RNA [autores]

A continuación, en la figura 60, se visualiza el comportamiento de ajuste normal del entrenamiento de la red neuronal, usando un modelo el cual consta con la siguiente información: número de neuronas, optimizador, épocas, el optimizador RMSPROP el que dio mejores resultados, ver tabla 14.

```
In [ ]: model, history = MODEL_LSTM(optimizador = 'rmsprop', neuronas = 10, epocas = 10)
#Guardamos el modelo para no tener que entrenar otra vez para probar
model.save('modelos/model.h5')

Epoch 1/10
1071/1071 [=====] - 72s 63ms/step - loss: 0.0139 - mean_absolute_error: 0.0842
Epoch 2/10
1071/1071 [=====] - 64s 60ms/step - loss: 0.0040 - mean_absolute_error: 0.0443
Epoch 3/10
1071/1071 [=====] - 67s 62ms/step - loss: 0.0029 - mean_absolute_error: 0.0363
Epoch 4/10
1071/1071 [=====] - 62s 58ms/step - loss: 0.0027 - mean_absolute_error: 0.0343
Epoch 5/10
1071/1071 [=====] - 63s 59ms/step - loss: 0.0026 - mean_absolute_error: 0.0336
Epoch 6/10
1071/1071 [=====] - 70s 65ms/step - loss: 0.0025 - mean_absolute_error: 0.0331
Epoch 7/10
1071/1071 [=====] - 70s 65ms/step - loss: 0.0025 - mean_absolute_error: 0.0328
Epoch 8/10
1071/1071 [=====] - 70s 65ms/step - loss: 0.0025 - mean_absolute_error: 0.0326
Epoch 9/10
1071/1071 [=====] - 61s 57ms/step - loss: 0.0024 - mean_absolute_error: 0.0324
Epoch 10/10
1071/1071 [=====] - 61s 57ms/step - loss: 0.0024 - mean_absolute_error: 0.0322
```

Figura 60: Entrenamiento con el optimizador RMSPROP [autores]

### 5.8. VALIDAR LA RED NEURONAL ARTIFICIAL

Para la validación de la red neuronal artificial es necesario analizar el porcentaje de error, se dice que la medida es buena cuando no supera el 5 %.

Como se observa en la tabla 13, se tienen los valores de entrenamiento de las 10 épocas con su respectivo error observando que en cada época el error disminuye hasta llegar a un error total de 3,22 %. Llegando a la conclusión de que la red neuronal artificial realiza predicciones acertadas, ya que el error no sobrepasa el 5 %.

Tabla 13: Porcentaje de error entre el caudal real y pronosticado [autores]

Épocas	Error	Porcentaje de error
Epoch 1/10	0,0842	8,42%
Epoch 2/10	0,0443	4,43%
Epoch 3/10	0,0363	3,63%
Epoch 4/10	0,0343	3,43%
Epoch 5/10	0,0336	3,36%
Epoch 6/10	0,0331	3,31%
Epoch 7/10	0,0328	3,28%
Epoch 8/10	0,0326	3,26%
Epoch 9/10	0,0324	3,24%
Epoch 10/10	0,0322	3,22%

En la figura 61 se observa la comparación de las curvas tanto del caudal real como del caudal pronosticado, el cual se realizó con 10 neuronas, 10 épocas y el optimizador RMSPROP, obteniendo una gran similitud en todas las horas del día pronosticado.

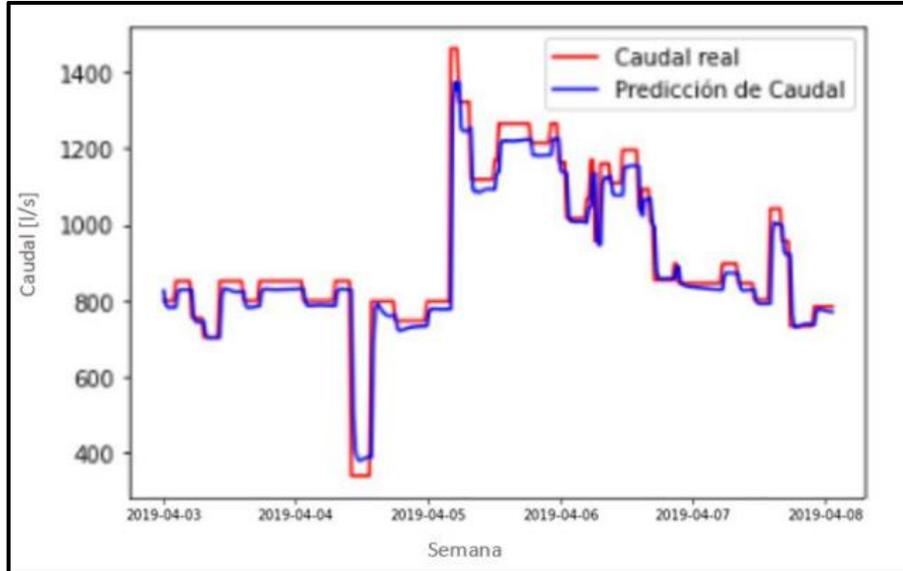


Figura 61: Comparación del caudal real con el pronosticado [autores]

## 5.9. ALMACENAMIENTO DE LOS PARÁMETROS DE LA RED NEURONAL ARTIFICIAL

Se necesita recopilar toda la información, se utilizó el formato HDF5 que es un modelo de datos, biblioteca y formato de archivo para almacenar y administrar datos.

Para no realizar de nuevo el entrenamiento de la red neuronal se procedió a guardar en formato h5 ya que este es un paquete común para manipular y usar los datos HDF5 como se observa en la figura 62.

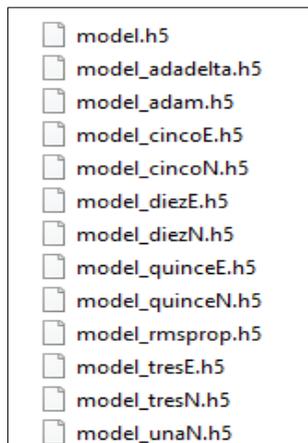


Figura 62: Modelos guardados en formato h5 [autores]

## 5.10. PREDICCIÓN DE LA GENERACIÓN ELÉCTRICA

El diseño correcto de una red neuronal artificial se desarrolla comparando las diferentes configuraciones propuestas por los diseñadores.

Se compararon diferentes configuraciones de redes neuronales artificiales para elegir los mejores resultados para el pronóstico de generación de energía, las cuales son tipos de algoritmos de entrenamiento, número de neuronas y número de épocas. Para el presente trabajo de titulación se ingresa como datos de entrada los datos de generación (caudal) de la central hidroeléctrica Illuchi 1, ELEPCO SA.

### 5.10.1. Comparación de los algoritmos de entrenamiento

A continuación, en la tabla 14, se realizó una comparativa más a fondo de los 3 algoritmos de entrenamiento seleccionados, mostrando sus porcentajes de error absoluto, además de su comparación en una gráfica para observar de manera visual cual es el que se acerca más a la curva del caudal real.

Tabla 14: Error absoluto de los algoritmos de entrenamiento [autores]

Algoritmos de entrenamiento	Error absoluto
ADAM	0,0327
ADADELTA	0,1743
RMSPROP	0,0325

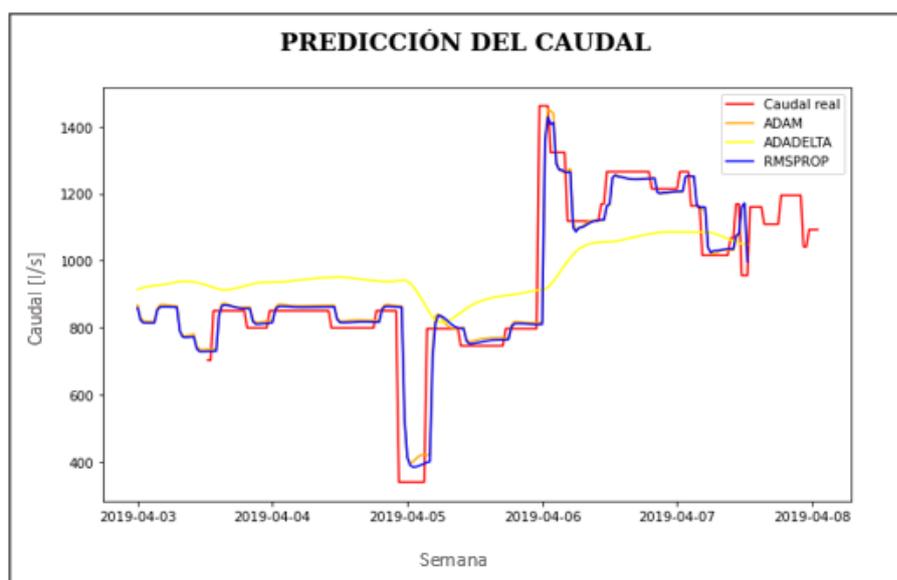


Figura 63: Curva de los diferentes algoritmos de entrenamiento [autores]

Como se observa en la figura 63 el optimizador que muestra mejores resultados es el de RMSPROP ya que la curva de este es muy similar a la del caudal real.

### 5.10.2. Comparación con el número de neuronas

A continuación, en la tabla 15, se realiza una comparativa con valores aleatorios de números de neuronas en este caso de 1,3, 5, 10 y 15, mostrando sus porcentajes de error absoluto, además de su comparación en una gráfica para observar de manera visual que valor es el que se acerca más a la curva del caudal real.

Tabla 15: Error absolute con el número de neuronas [autores]

Número de neuronas	Error absoluto
1	0,0694
3	0,0486
5	0,0409
10	0,0320
15	0,0292

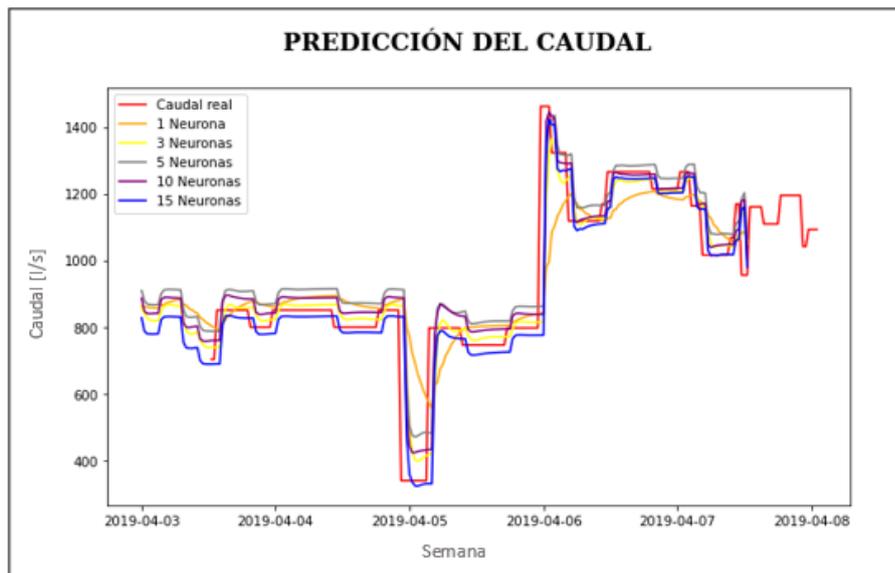


Figura 64: Curva de los diferentes números de neuronas [autores]

En la figura 64 se muestra la comparación del entrenamiento con diferente número de neuronas, la cual se decidió ocupar 10 neuronas para el entrenamiento ya que si se usa el de 15 neuronas este nos da como resultado un menor error, pero con un overfitting o un sobre entrenamiento.

### 5.10.3. Comparación con el número de épocas

A continuación, en la tabla 16, se realiza una comparativa con valores aleatorios de números de épocas en este caso de 3, 5, 10 y 15, mostrando sus porcentajes de error absoluto, además de su comparación en una gráfica para observar de manera visual que valor es el que se acerca más a la curva del caudal real.

Tabla 16: Error absoluto con el número de épocas [autores]

Número de épocas	Erro absoluto
3	0,0337
5	0,0298
10	0,0286
15	0,0281

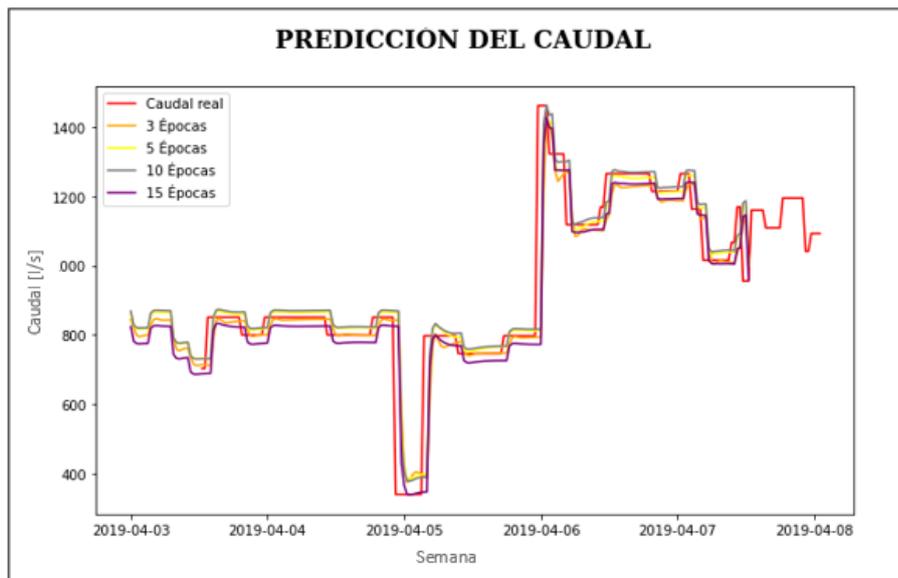


Figura 65: Curva de los diferentes números de épocas [autores]

En la figura 65 se muestra la comparación del entrenamiento con diferente número de épocas, la cual se decidió ocupar 10 épocas para el entrenamiento, aunque con 15 épocas existe un menor error, ya que 10 épocas y 10 neuronas es la combinación más óptima para la estructura de la red neuronal, así evitando un overffiting o underffiting.

## 5.11. ANÁLISIS DEL COMPORTAMIENTO DE LA BASE DE DATOS REAL

Para hacer predicciones, es necesario crear una base de datos en un nuevo Excel el cual consta de datos del 1 de enero del 2 018 al 9 de abril del 2 019.

Se debe analizar el comportamiento de los datos, estimar las técnicas a utilizar, y luego se pronostican los siguientes periodos, el tamaño de la muestra es importante para la precisión de la predicción, debido a que los datos son variables, se considera un análisis espaciotemporal para estimar su comportamiento. En la práctica, se utiliza más de un año de datos históricos y se proyecta en un marco de tiempo, es decir, una semana, para obtener predicciones confiables del comportamiento del flujo.

En la figura 66, se puede observar la curva del caudal diario, del día 3 al 9 de abril del 2019, de la Central Hidroeléctrica Illuchi 1 el cual se caracteriza por tener los intervalos de tiempo en días. Se puede apreciar que el comportamiento del caudal del día 4 de abril es menor.

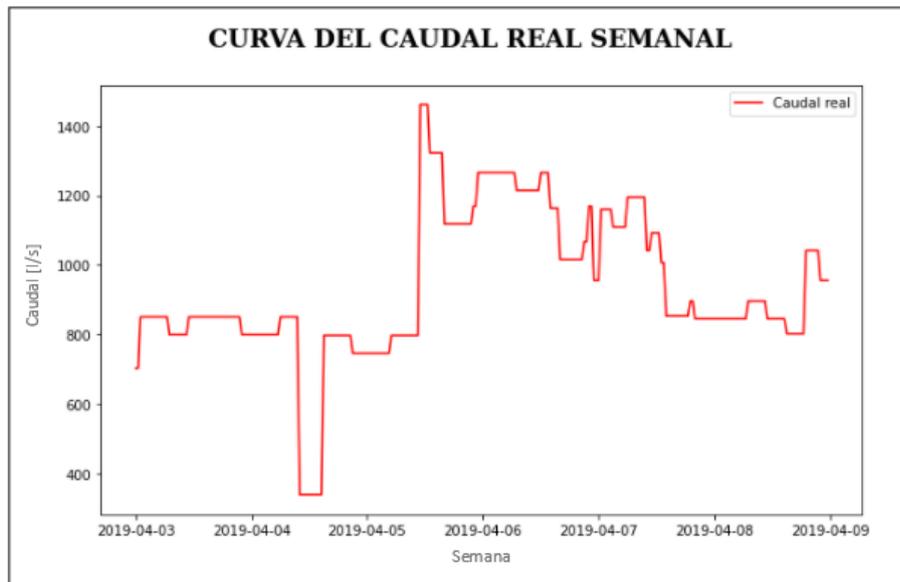


Figura 66: Comportamiento del caudal real por día [autores]

En la tabla 17, se detallan los datos de estudio, los cuales pertenecen del día 3 al 9 de abril del 2019, se consiguen valores mínimos y máximos del caudal donde se presenta un valor de rango 1122,82 (l/s).

Tabla 17: Análisis descriptiva con datos reales del comportamiento del caudal [autores]

<b>ANÁLISIS DESCRIPTIVO</b>	
MÁXIMO	1461,8667 (l/s)
MÍNIMO	339,0467 (l/s)
RANGO	1122,82 (l/s)
MEDIA	936,50058 (l/s)
MEDIANA	850,4467 (l/s)
VARIANZA	45222,8132
DESVIACIÓN ESTÁNDAR	212,28704
COEFICIENTE DE VARIACIÓN	22,66 %
LÍMITE MÁXIMO	1148,78763 (l/s)
LÍMITE MÍNIMO	724,21354 (l/s)

La media o valor medio de la tendencia central es 936,50058 (l/s), que representa el valor promedio del caudal en el intervalo de datos de análisis, la mediana es el valor que ocupa la parte central del conjunto de datos, donde el 50 % de los datos estarán por debajo y por encima del 50 % de 850,4467 (l/s), no se puede calcular el valor de la moda porque representa el valor con más repeticiones.

Una medida de dispersión muestra qué tan lejos o cerca están los datos en relación con una medida de tendencia central, donde la varianza representa la inestabilidad de la serie de datos, en este caso la media es un valor muy alto y la desviación estándar es sobre cómo estos los valores son relativos al medio (cerca o lejos), el valor es 850,4467 (l/s), es decir con el porcentaje de coeficiente de variación, la dispersión de datos es 22,66 %, y hay inconsistencia de datos.

Para visualizar mejor el comportamiento de la curva del caudal en la figura 67, se trazaron los datos diarios de abril de 2019 para el día 4, donde se observaron datos reales con diferentes comportamientos, aumentando y disminuyendo el caudal a lo largo del día.

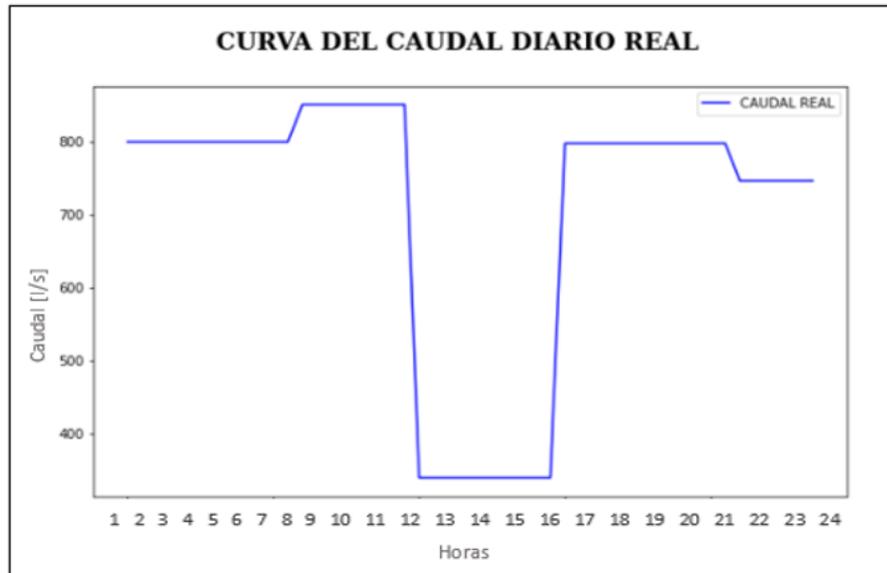


Figura 67: Curva del caudal diaria en horas [autores]

En la tabla 18, se detallan los datos de estudio, los cuales pertenecen al día 4 abril del 2019, se consiguen valores mínimos y máximos del caudal donde se presenta un valor de rango 450,9327 (l/s).

Tabla 18: Análisis descriptiva con datos reales del comportamiento del caudal [autores]

ANÁLISIS DESCRIPTIVO	
MÁXIMO	830,3599 (l/s)
MÍNIMO	379,4272 (l/s)
RANGO	450,9327 (l/s)
MEDIA	700,2163 (l/s)
MEDIANA	777,0620 (l/s)
VARIANZA	25657,2070
DESVIACIÓN ESTÁNDAR	158,5013
COEFICIENTE DE VARIACIÓN	22,63 %
LÍMITE MÁXIMO	858,7177 (l/s)
LÍMITE MÍNIMO	541,7150 (l/s)

La media o valor medio de la tendencia central es 700,2163 (l/s), que representa el valor promedio del caudal en el intervalo de datos de análisis, la mediana es el valor que ocupa la parte central del conjunto de datos, donde el 50 % de los datos estarán por debajo y por encima del 50 % de 777,0620 (l/s), no se puede calcular el valor de la moda porque representa el valor con más repeticiones.

Una medida de dispersión muestra qué tan lejos o cerca están los datos en relación con una medida de tendencia central, donde la varianza representa la inestabilidad de la serie de datos, en este caso la media es un valor muy alto y la desviación estándar es sobre cómo estos los valores son relativos al medio (cerca o lejos), el valor es 777,0620 (l/s), es decir con el porcentaje de coeficiente de variación, la dispersión de datos es 22,63 %, y hay inconsistencia de datos.

## 5.12. ANÁLISIS DEL COMPORTAMIENTO DEL CAUDAL PRONOSTICADO

En la figura 68, se puede observar la curva del caudal diario pronosticado, del día 3 al 9 de abril del 2019, de la Central Hidroeléctrica Illuchi 1 el cual se caracteriza por tener los intervalos de tiempo en días. Se puede apreciar que el comportamiento del caudal del día 4 de abril es menor.

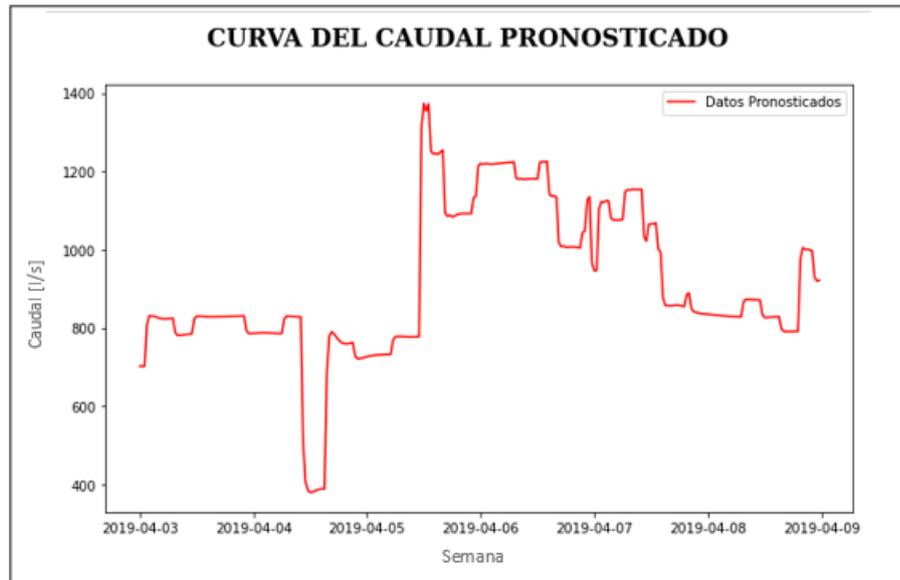


Figura 68: Comportamiento del caudal precedido por día [autores]

En la tabla 19, se detallan los datos de estudio pronosticados, los cuales pertenecen del día 3 al 9 de abril del 2019, se consiguen valores mínimos y máximos del caudal donde nos presenta un valor de rango 994,4453 (l/s).

Tabla 19: Análisis descriptiva con datos pronosticados del comportamiento del caudal [autores]

<b>ANÁLISIS DESCRIPTIVO</b>	
MÁXIMO	1373,8725 (l/s)
MÍNIMO	379,4272 (l/s)
RANGO	994,4453 (l/s)
MEDIA	913,7523 (l/s)
MEDIANA	831,8135 (l/s)
VARIANZA	37904,2421
DESVIACIÓN ESTÁNDAR	194,3518
COEFICIENTE DE VARIACIÓN	21,2696 %
LÍMITE MÁXIMO	1108,1042 (l/s)
LÍMITE MÍNIMO	719,4005 (l/s)

La media o valor medio de la tendencia central es 913,7523 (l/s), que representa el valor promedio del caudal en el intervalo de datos de análisis, la mediana es el valor que ocupa la parte central del conjunto de datos, donde el 50 % de los datos estarán por debajo y por encima del 50 % de 831,8135 (l/s), no se puede calcular el valor de la moda porque representa el valor con más repeticiones.

Una medida de dispersión muestra qué tan lejos o cerca están los datos en relación con una medida de tendencia central, donde la varianza representa la inestabilidad de la serie de datos, en este caso la media es un valor muy alto y la desviación estándar es sobre cómo estos los valores son relativos al medio (cerca o lejos), el valor es 831,8135 (l/s), es decir con el porcentaje de coeficiente de variación, la dispersión de datos es 21,26 %, y hay inconsistencia de datos.

Para visualizar mejor el comportamiento de la curva del caudal en la figura 69, se trazaron los datos diarios de abril de 2 019 para el día 4, donde se observaron los datos pronosticados con diferentes comportamientos, aumentando y disminuyendo el caudal a lo largo del día.

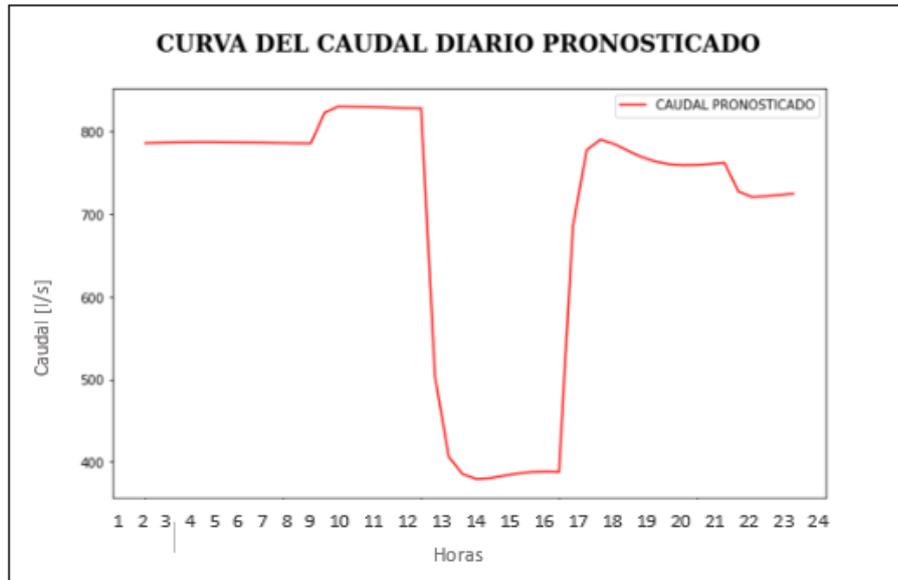


Figura 69: Curva del caudal diaria en horas [autores]

En la tabla 20, se detallan los datos de estudio, los cuales pertenecen al día 4 abril del 2019, se consigue valores mínimos y máximos del caudal pronosticado donde se presenta un valor de rango 511,4 (l/s).

Tabla 20: Análisis descriptiva con datos pronosticados del comportamiento del caudal [autores]

ANÁLISIS DESCRIPTIVO	
MÁXIMO	850,4467 (l/s)
MÍNIMO	339,0467 (l/s)
RANGO	511,4 (l/s)
MEDIA	704,70253 (l/s)
MEDIANA	797,0467 (l/s)
VARIANZA	36742,3021
DESVIACIÓN ESTÁNDAR	189,6756
COEFICIENTE DE VARIACIÓN	26,91%
LÍMITE MÁXIMO	894,3781 (l/s)
LÍMITE MÍNIMO	515,0269 (l/s)

La media o valor medio de la tendencia central es 704,70253 (l/s), que representa el valor promedio del caudal en el intervalo de datos de análisis, la mediana es el valor que ocupa la parte central del conjunto de datos, donde el 50 % de los datos estarán por debajo y por encima del 50 % de 797,0467 (l/s), no se puede calcular el valor de la moda porque representa el valor con más repeticiones.

Una medida de dispersión muestra qué tan lejos o cerca están los datos en relación con una medida de tendencia central, donde la varianza representa la inestabilidad de la serie de datos, en este caso la media es un valor muy alto y la desviación estándar es sobre cómo estos los valores son relativos al medio (cerca o lejos), el valor es 797,0467 (l/s), es decir con el porcentaje de coeficiente de variación, la dispersión de datos es 26,91 %, y hay inconsistencia de datos.

### 5.13. COMPARACIÓN DE DATOS REALES Y PRONÓSTICO DE LA RED NEURONAL

En la figura 70, se puede observar el comportamiento del caudal utilizando los datos reales del 3 al 9 de abril de 2019 y los datos del pronóstico para el mismo año. Se comportan de manera muy similar en términos de crecimiento y pérdida de caudal.

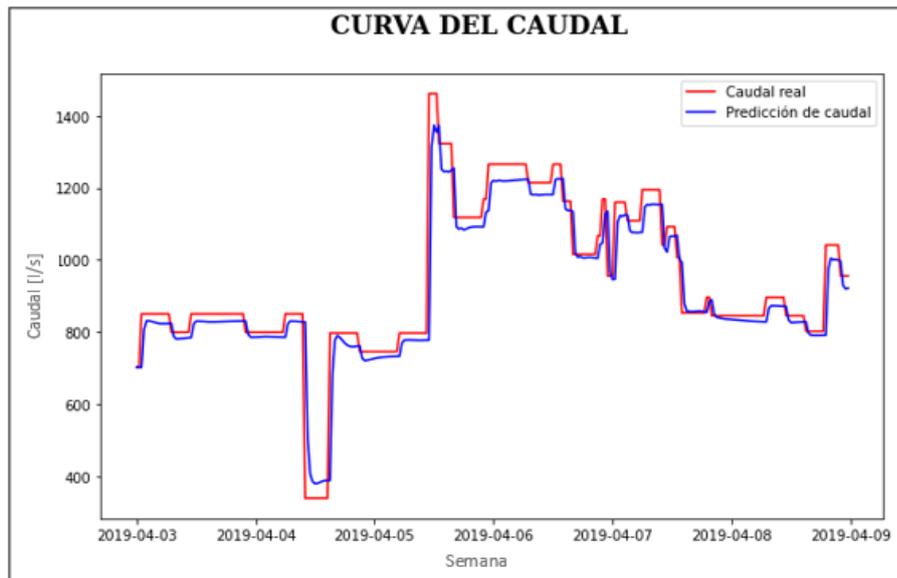


Figura 70: Curva del caudal real vs el pronosticado [autores]

El error entre los datos reales y la predicción de caudal se muestra en la figura 71. En la tabla 21 se obtiene el error medio absoluto de la comparación de los dos datos de la curva de caudal, que es 0,0322.

Tabla 21: Error promedio entre el caudal real y el pronosticado semanal [autores]

Error absoluto	Valor
Error	0,0322

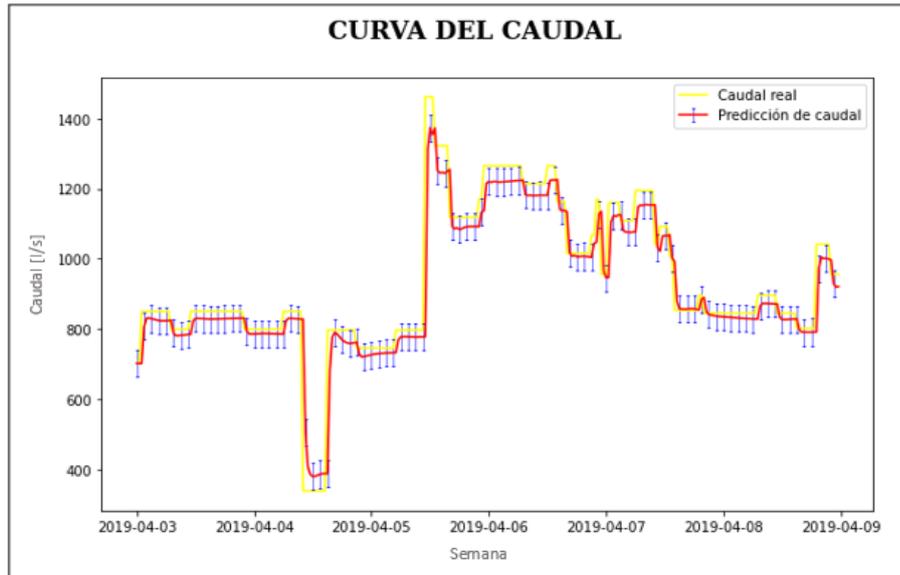


Figura 71: Gráfica de error entre el caudal real y el pronosticado [autores]

En la figura 72, se muestra un intervalo de caudal con datos reales, con el intervalo de una semana, donde se verificó el comportamiento del caudal para cada día, también se puede observar que en los primeros 3 días de la semana tenemos un caudal similar, el día 4 donde tiene un gran aumento, retomando el final de los días el mismo nivel de caudal de los días iniciales.

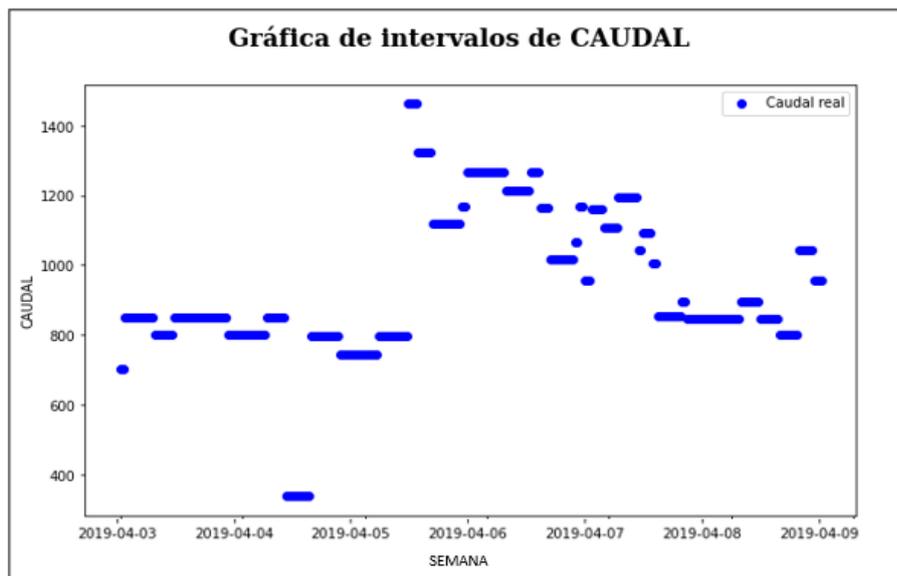


Figura 72: Intervalos de caudal semanal real [autores]

En la figura 73, se muestra un intervalo de caudal con datos pronosticados, con el intervalo de una semana, donde se verificó el comportamiento del caudal para cada día, también se puede observar que en los primeros 3 días de la semana tenemos un caudal similar, el día 4 donde

tiene un gran aumento, retomando el final de los días el mismo nivel de caudal de los días iniciales.

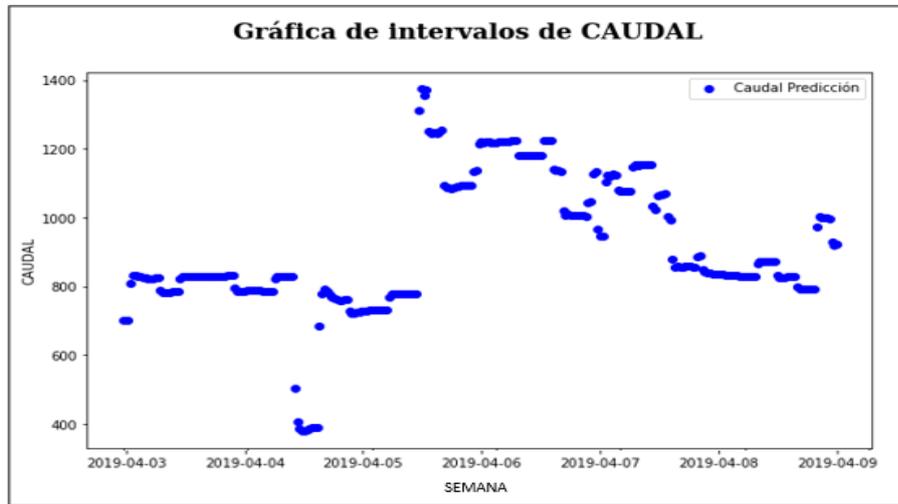


Figura 73: Intervalos de caudal semanal pronosticado [autores]

Las figuras 72 y 73 ilustran el intervalo de caudal con datos reales y pronosticados, con un intervalo de 1 semana, donde se verificó que el comportamiento del caudal es muy similar en real y pronosticado.

Para visualizar mejor el comportamiento del caudal semanal utilizando datos reales y pronosticados, la figura 74 muestra el caudal pronosticado y real, que tienden a ser similares.

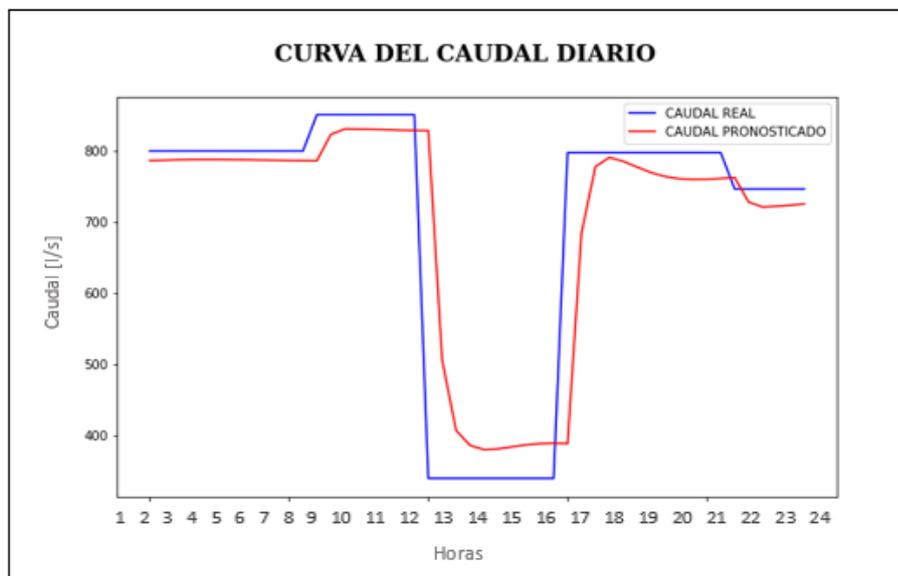


Figura 74: Curva del caudal real vs el pronosticado diario [autores]

El error entre los datos reales y la predicción de caudal se muestra en la figura 75. En la tabla 22 se obtiene el error medio absoluto de la comparación de los dos datos de la curva de caudal, que es 0,0161.

Tabla 22: Error promedio entre el caudal real y el pronosticado diario [autores]

Error medio	Valor
Error	0,0161

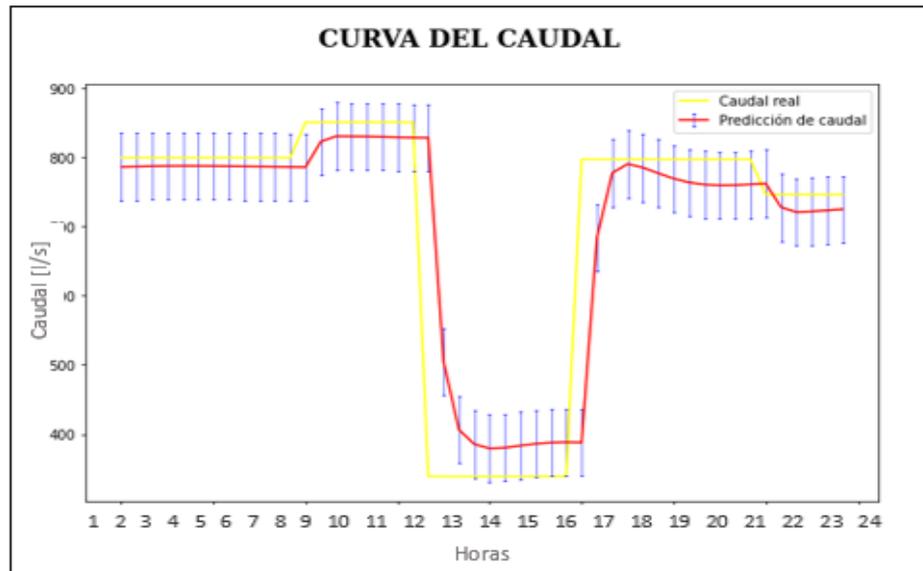


Figura 75: Gráfica de error entre el caudal real y el pronosticado [autores]

En la figura 76, se muestra un intervalo de caudal con datos reales, con el intervalo de un día, donde se verificó el comportamiento del caudal por cada una de las horas, también se puede observar que las primeras 10 horas, tenemos un caudal similar, las horas 12,13,14,15 y 16 donde tiene una gran disminución del caudal.

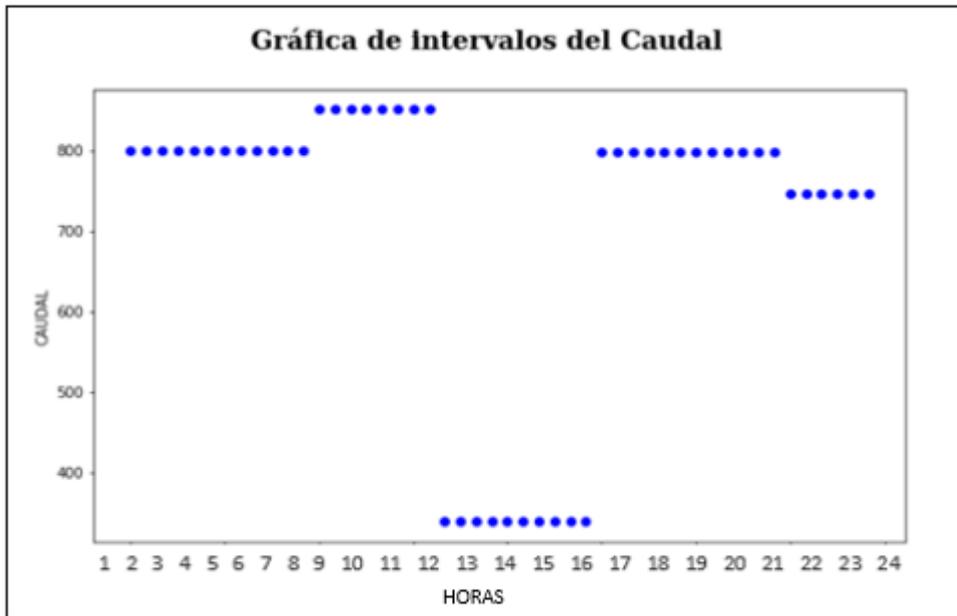


Figura 76: Intervalos de caudal diario real [autores]

En la figura 77, se muestra un intervalo de caudal con datos pronosticados, con el intervalo de un día, donde se verificó el comportamiento del caudal por cada una de las horas, también se puede observar que las primeras 10 horas, tenemos un caudal similar, las horas 12,13,14,15 y 16 donde tiene una gran disminución del caudal.

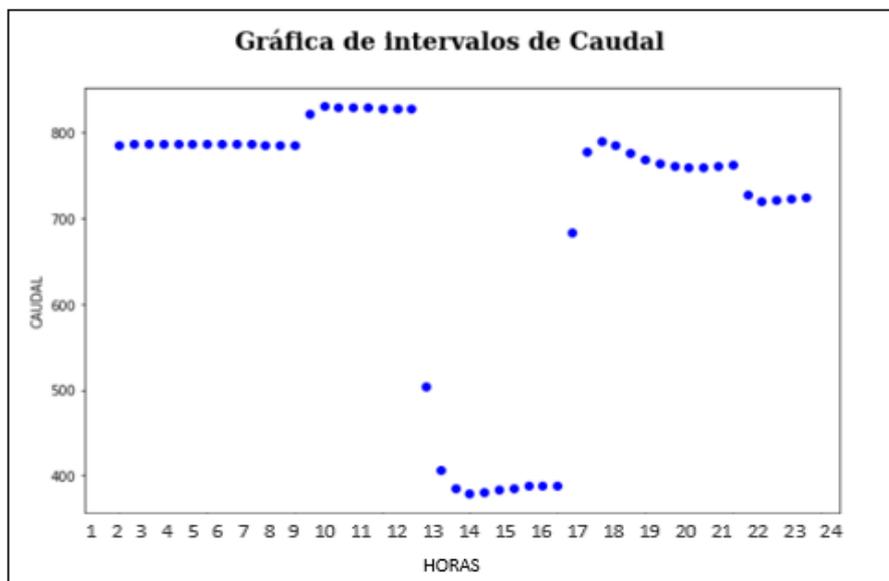


Figura 77: Intervalos de caudal diario pronosticado [autores]

Las figuras 76 y 77 ilustran el intervalo de caudal con datos reales y pronosticados, con datos del día 4 de abril, donde se verificó que el comportamiento del caudal es muy similar en real y pronosticado, teniendo los mismo altos y bajos en el día.

## 5.14. DESARROLLO DE LA INTERFAZ GRÁFICA

Se realizó una interfaz gráfica con el fin de hacer una demostración más didáctica del entrenamiento y predicción de la red neuronal. A continuación, se explicarán todos los componentes que componen la misma.

- a. Como punto inicial se tiene un visualizador de datos históricos, los cuales constan de la fecha y los 4 generadores que existe en la hidroeléctrica Illuchi 1, y además sus variables de entrada, se puede ingresar el total de los datos, como se puede observar en la figura 78 existe un cuadro en donde se inserta el año, mes, día, hora inicial y final para su posterior visualización.

ID	Año	Mes	Día	Hora	G1	G2	G3	G4
0	2015	1	1	00:00:00	0.0	500	1000.0	1200.0
1	2015	1	1	00:30:00	0.0	500	1000.0	1200.0
2	2015	1	1	01:00:00	0.0	500	700.0	800.0
3	2015	1	1	01:30:00	0.0	500	700.0	800.0
4	2015	1	1	02:00:00	0.0	500	700.0	800.0
5	2015	1	1	02:30:00	0.0	500	700.0	800.0
6	2015	1	1	03:00:00	0.0	500	700.0	800.0
7	2015	1	1	03:30:00	0.0	500	700.0	800.0
8	2015	1	1	04:00:00	0.0	500	700.0	800.0
9	2015	1	1	04:30:00	0.0	500	700.0	800.0
10	2015	1	1	05:00:00	0.0	500	700.0	800.0
11	2015	1	1	05:30:00	0.0	500	700.0	800.0
12	2015	1	1	06:00:00	0.0	500	700.0	800.0
13	2015	1	1	06:30:00	0.0	500	700.0	800.0
14	2015	1	1	07:00:00	0.0	500	700.0	800.0
15	2015	1	1	07:30:00	0.0	500	700.0	800.0
16	2015	1	1	08:00:00	0.0	500	1000.0	800.0
17	2015	1	1	08:30:00	0.0	500	1000.0	800.0

Figura 78: Datos históricos reales [autores]

- b. Siguiendo con la conversión de potencia a caudal de los datos anteriores obteniendo así el dataFrame con las variables de entradas seleccionadas (año, mes, día, hora, potencia y caudal) para iniciar con el entramiento de nuestra red neuronal como se observa en la figura 79.

UNIVERSIDAD TÉCNICA DE COTOPAXI  
INGENIERÍA ELÉCTRICA

Autor:  
- Gonzalez Piedra Joel Andre  
- Hidalgo Barrera Christopher Alexander

Datos Históricos Reales    **Conversión de Potencia a Caudal**    Entrenamiento Red Neuronal    Predicción del Caudal    Salir

Datos  
 Total  
 Parcial

Desde    Hasta  
Año    Año  
Mes    Mes  
Día    Día  
Hora    Hora

Visualizar Datos  
Graficar

ID	Año	Mes	Día	Hora	Potencial Total	Caudal Total
0	2015	1	1	00:00:00	2700	1100.7567
1	2015	1	1	00:30:00	2700	1100.7567
2	2015	1	1	01:00:00	2000	742.7766999999999
3	2015	1	1	01:30:00	2000	742.7766999999999
4	2015	1	1	02:00:00	2000	742.7766999999999
5	2015	1	1	02:30:00	2000	742.7766999999999
6	2015	1	1	03:00:00	2000	742.7766999999999
7	2015	1	1	03:30:00	2000	742.7766999999999
8	2015	1	1	04:00:00	2000	742.7766999999999
9	2015	1	1	04:30:00	2000	742.7766999999999
10	2015	1	1	05:00:00	2000	742.7766999999999
11	2015	1	1	05:30:00	2000	742.7766999999999
12	2015	1	1	06:00:00	2000	742.7766999999999
13	2015	1	1	06:30:00	2000	742.7766999999999
14	2015	1	1	07:00:00	2000	742.7766999999999
15	2015	1	1	07:30:00	2000	742.7766999999999
16	2015	1	1	08:00:00	2300	896.1967
17	2015	1	1	08:30:00	2300	896.1967

Figura 79: Conversión de potencia a caudal [autores]

- c. Como se puede observar en la figura 80, existe un botón para poder apreciar la gráfica del caudal real en cualquier rango de tiempo de nuestros años seleccionados en este caso se seleccionó el total de los datos, además se puede exportar la gráfica en formato png.

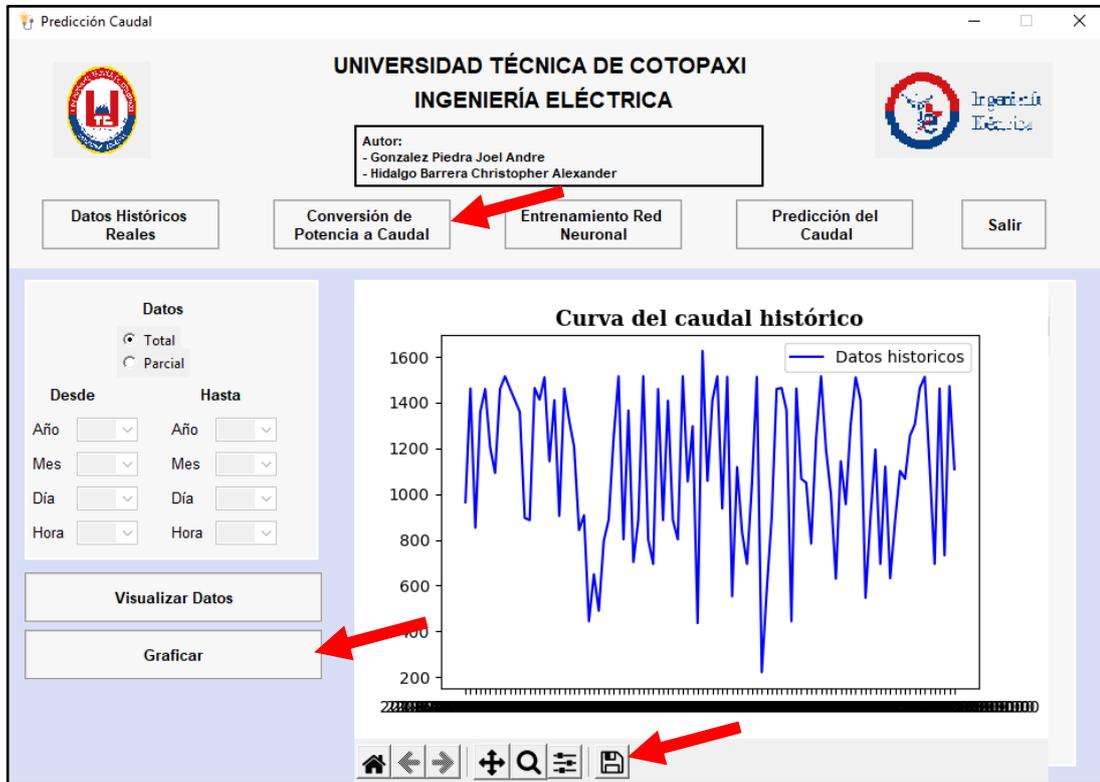


Figura 80: Gráfica del caudal real [autores]

- d. En el siguiente punto como se puede observar en la figura 81 se tiene el entrenamiento de la red neuronal artificial en la cual existe ventanas desplegadas en donde se puede modificar el optimizador a utilizar, el número de neuronas y retraso de tiempo o épocas, para su respectivo entrenamiento, también se puede observar la gráfica del error que se obtuvo después del entrenamiento y sus pérdidas.

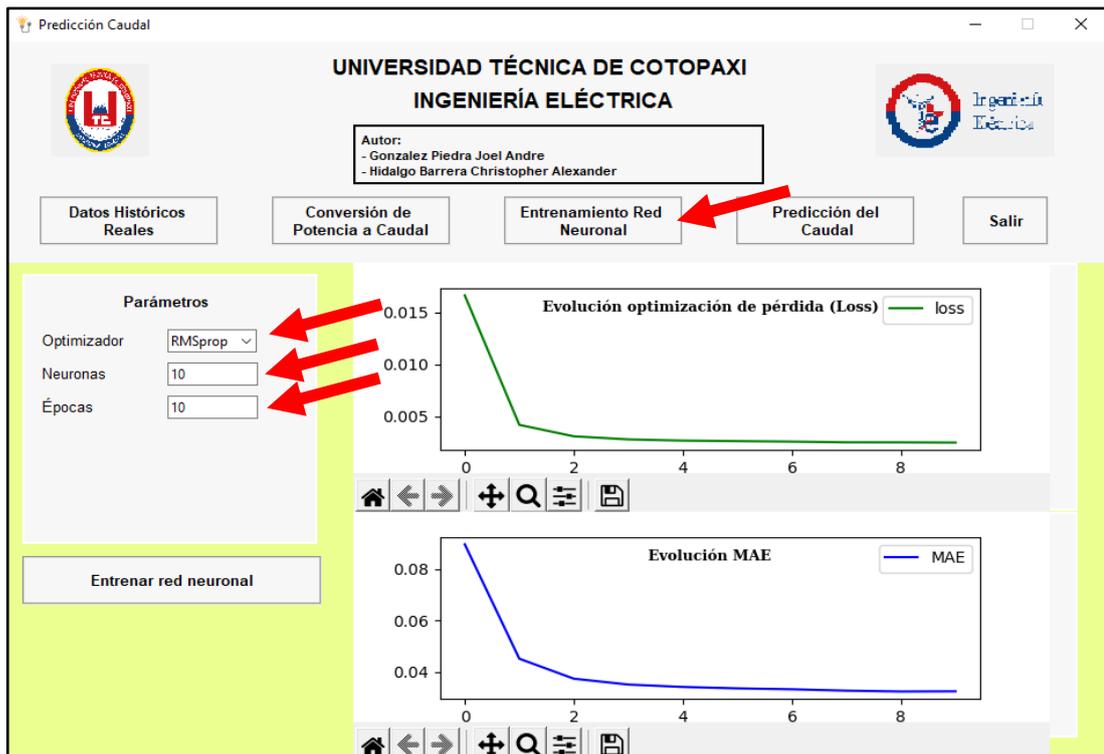


Figura 81: Entrenamiento de la red [autores]

- e. Como se puede observar en la figura 82 existen ventanas desplegables en las cuales se ingresa en año, mes y día a predecir, se obtiene una gráfica con la predicción del caudal de la fecha ingresada para la central hidroeléctrica Illuchi 1, además se puede exportar la gráfica en formato png.

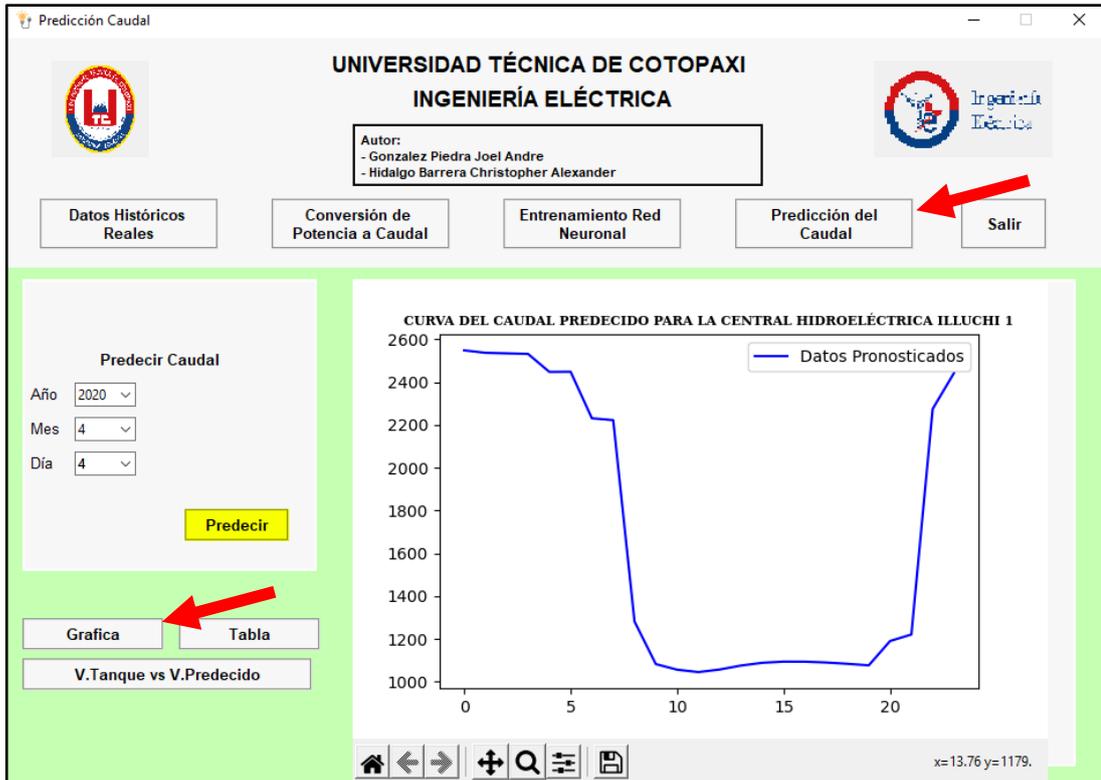


Figura 82: Gráfica del caudal pronosticado [autores]

- f. Como se puede observar en la figura 83 existe un botón para visualizar una tabla con los resultados de los datos pronosticados, además de una comparación con el caudal en  $m^3$ .

**UNIVERSIDAD TÉCNICA DE COTOPAXI**  
**INGENIERÍA ELÉCTRICA**

Autor:  
- Gonzalez Piedra Joel Andre  
- Hidalgo Barrera Christopher Alexander

Datos Históricos Reales    Conversión de Potencia a Caudal    Entrenamiento Red Neuronal    **Predicción del Caudal**    Salir

**Predecir Caudal**  
Año: 2020  
Mes: 4  
Día: 4  
Predecir

Grafica    **Tabla**    V.Tanque vs V.Predecido

Fecha	Caudal (L/s)	Caudal (m3)
2020-04-04 00:00:00	1415.894287109375	2548.609716796875
2020-04-04 01:00:00	1409.525146484375	2537.145263671875
2020-04-04 02:00:00	1408.0660400390625	2534.5188720703127
2020-04-04 03:00:00	1406.68603515625	2532.03486328125
2020-04-04 04:00:00	1360.0615234375	2448.1107421875
2020-04-04 05:00:00	1360.3773193359375	2448.6791748046876
2020-04-04 06:00:00	1239.43310546875	2230.97958984375
2020-04-04 07:00:00	1234.688232421875	2222.438818359375
2020-04-04 08:00:00	711.796630859375	1281.233935546875
2020-04-04 09:00:00	601.1829223632812	1082.1292602539063
2020-04-04 10:00:00	586.6587524414062	1055.9857543945313
2020-04-04 11:00:00	580.4141845703125	1044.7455322265625
2020-04-04 12:00:00	587.1380004882812	1056.8484008789062
2020-04-04 13:00:00	597.4249877929688	1075.3649780273438
2020-04-04 14:00:00	604.5562744140625	1088.2012939453125
2020-04-04 15:00:00	607.5899047851562	1093.6618286132812
2020-04-04 16:00:00	607.475830078125	1093.456494140625
2020-04-04 17:00:00	605.299072265625	1089.538330078125

Figura 83: Tabla de resultados de la predicción [autores]

- g. En este último punto como se muestra en la figura 84 existe un botón para comprar el volumen del tanque de la central hidroeléctrica Illuchi 1 con el caudal pronosticado en  $m^3$  (volumen), obteniendo así un mensaje en el cual se muestra las veces del volumen necesario y si este es o no apto para el día pronosticado.

The screenshot shows a web application window titled "Predicción Caudal" from the "UNIVERSIDAD TÉCNICA DE COTOPAXI INGENIERÍA ELÉCTRICA". The interface includes a navigation menu with buttons for "Datos Históricos Reales", "Conversión de Potencia a Caudal", "Entrenamiento Red Neuronal", "Predicción del Caudal", and "Salir". A red arrow points to the "Predicción del Caudal" button. The main content area is titled "Relación del Volumen para la predicción del día" and displays "Volumen tanque: 18690m<sup>3</sup>" and "Caudal: 39751.676477050794". A yellow message box states "Se necesitan 2.13 veces el volumen del tanque". Below this, there are buttons for "Grafica" and "Tabla", with a red arrow pointing to a button labeled "V.Tanque vs V.Predicado".

Figura 84: Comparación del volumen del tanque con el volumen pronosticado [autores]

## 6. CONCLUSIONES

A continuación, se presenta las conclusiones obtenidas durante el desarrollo del presente trabajo de titulación, el cual expone la predicción de la generación eléctrica mediante redes neuronales artificiales en una central hidroeléctrica.

- Al realizar el estudio investigativo del estado del arte basado en el entrenamiento y testeo de la red neuronal se determinó que existe varios algoritmos de entrenamiento los cuales proporcionan diferentes valores de error, definiendo así la estructura para obtener los mejores resultados en el entrenamiento de la red neuronal.
- A partir de los datos de la generación de energía eléctrica de la central Hidroeléctrica Illuchi 1, se realizó la limpieza y normalización de los mismos, desarrollando a continuación una herramienta computacional en Python, a través de la interfaz gráfica que permite predecir cuánto caudal diario es necesario para la generación de energía eléctrica, utilizando una red neuronal artificial que está compuesta por una capa de entrada con 10 neuronas, dos capas ocultas, dos capas de congelamiento para evitar la memorización de datos y una capa de salida. Los parámetros configurados son: el tipo de algoritmo de entrenamiento, el número de neuronas en la capa oculta, capa de congelación y número de retrasos de tiempo o épocas.
- El desarrollo del entrenamiento de la red neuronal consta con el testeo de varios optimizadores, de los cuales se tomó el de mejor resultado del entrenamiento, el optimizador RMSPROP con 10 números de retrasos y 10 neuronas. Con esta estructura se logró optimizar los tiempos de entrenamiento y además obtener un error muy bajo de 0,016, para no tener el problema de sobre entrenamiento de la red neuronal artificial se agregó dos capas de congelamiento adicional.
- Para validar la red neuronal artificial se tomó el 80 % de datos de entrenamiento y 20 % datos de validación, donde se obtuvo un error medio de 0,0322, obteniendo un acercamiento a los datos reales, llegando así a una red neuronal confiable.

## **7. RECOMENDACIONES**

- En el campo de la inteligencia artificial las redes neuronales artificiales solo ocupan una pequeña parte de esta, se sugieren otras áreas basadas en inteligencia artificial para predecir la generación de energía eléctrica, para su seguida comparación con la usada en el presente proyecto de titulación.
- Depende de la cantidad de datos utilizados para el entrenamiento del modelo de las redes neuronales artificiales se deben determinar las características de la GPU, lo que ahorra tiempo de entrenamiento.
- Se recomienda desarrollar modelos de predicción en las diferentes áreas de la ingeniería eléctrica permitiendo soluciones a problemas reales como es el caso de la presente investigación.

## 8. BIBLIOGRAFÍA

- [1] Alvarez Chiriboga, Daniel Alejandro, «Modelo de predicción de la producción de energía de la Central Hidroeléctrica Coca Codo Sinclair, basado en técnicas de aprendizaje computacional», p. 100.
- [2] J. O. Zambrano, «CARRERA: INGENIERÍA EN SISTEMAS COMPUTACIONALES FECHA DE PUBLICACIÓN: de 2019», p. 85.
- [3] X. B. Olabe, «REDES NEURONALES ARTIFICIALES Y SUS APLICACIONES», p. 80.
- [4] FABIÁN MAURICIO CALERO FREIRE, «ESTUDIO TÉCNICO-ECONÓMICO DE LA IMPLEMENTACIÓN DEL CENTRO DE CONTROL DE LAS CENTRALES HIDROELÉCTRICAS ILLUCHI 1 E ILLUHI 2 DE ELEPCO S.A )», p. 238.
- [5] J. J. E. Monar, «ESTUDIO TÉCNICO ECONÓMICO PARA EL DISEÑO Y DIMENSIONAMIENTO DE UN TRANSFORMADOR TIPO ZIGZAG PARA PUESTA A TIERRA EN LA CENTRAL DE GENERACIÓN ILLUCHI 1 DE ELEPCO», p. 104.
- [6] C. Cabrera y X. Leonardo, «Diseño de una mini central de energía hidroeléctrica en la planta de tratamiento de aguas residuales de la ciudad de Cuenca», p. 155.
- [7] N. PK, *Ingeniería de plantas de energía*. [En línea]. Disponible en: <file:///C:/Users/ANDRE%20GONZALES/Documents/Tesis%20ANDRE/Bibliografia%20de%202.2.3%20Segunda%20Parte.pdf>
- [8] J. Campaña, «ANÁLISIS DE MODELOS DIGITALES DE TERRENO PARA LA OBTENCIÓN DE UN MAPA DE DENSIDAD DE DRENAJE MEDIANTE SOFTWARES FOTOGAMÉTRICOS Y SISTEMAS DE INFORMACIÓN GEOGRÁFICA PARA IMÁGENES OBTENIDAS CON UAV, RESULTADOS APLICADOS A ESTUDIOS MORFOMÉTRICOS EN LADERAS.», p. 166.
- [9] D. M. Sotelo, «DESARROLLO CON MATLAB DE UNA RED NEURONAL PARA ESTIMAR LA DEMANDA DE ENERGÍA ELÉCTRICA», p. 109.
- [10] F. G. Hernando, «Análisis de líneas de costa con redes neuronales LSTM», p. 57.
- [11] S. M. Hurtado, J. M. Cadavid, y J. P. G. Valencia, «Forecasting of hourly electric load in Colombia using artificial neural networks», p. 10, 2011.
- [12] J. F. Capdevila, J. M. Ruiz, J. L. Salinas, y E. A. S. Perez, «Redes neuronales y algoritmos genéticos en la docencia en las escuelas de ingenieros: Un problema práctico de diseño de pantallas acústicas», vol. 3, p. 12.

- [13] S. Raschka y V. Mirjalili, *Python machine learning: machine learning and deep learning with Python, scikit-learn, and TensorFlow 2*, Third edition. Birmingham Mumbai: Packt, 2019.
- [14] J. F. Raquel Flores Lopez, «Las Redes Neuronales Artificiales», p. 135, 2008.
- [15] A. M. A. Ramírez, «MÉTODOS UTILIZADOS PARA EL PRONÓSTICO DE DEMANDA DE ENERGÍA ELÉCTRICA EN SISTEMAS DE DISTRIBUCIÓN», p. 145, 2013.
- [16] I. A. A. Fierro y D. F. Ronchetti, «Predicción de Series Temporales con Redes Neuronales», p. 65.
- [17] M. Al Mamun y K. Nagasaka, «Artificial Neural Networks Applied to Long-term Electricity Demand Forecasting,» Department of Electrical and Electronics Engineering, Tokyo, 2005.
- [18] «ENTRENAMIENTO DE REDES NEURONALES BASADO EN ALGORITMOS EVOLUTIVOS».
- [19] G. van Rossum y F. L. Drake, «Guía de aprendizaje de Python», p. 77.
- [20] C. A. Ruiz, M. S. Basualdo, y D. J. Matich, «Redes Neuronales: Conceptos Básicos y Aplicaciones.», p. 55.
- [21] J. J. M. Moreno, «Redes Neuronales Artificiales aplicadas al Análisis de Datos», p. 275.

## 9. ANEXOS



Figura A1: Tanques de la central hidroeléctrica Illuchi 1



Figura A2: Mediciones para el modelado de los tanques

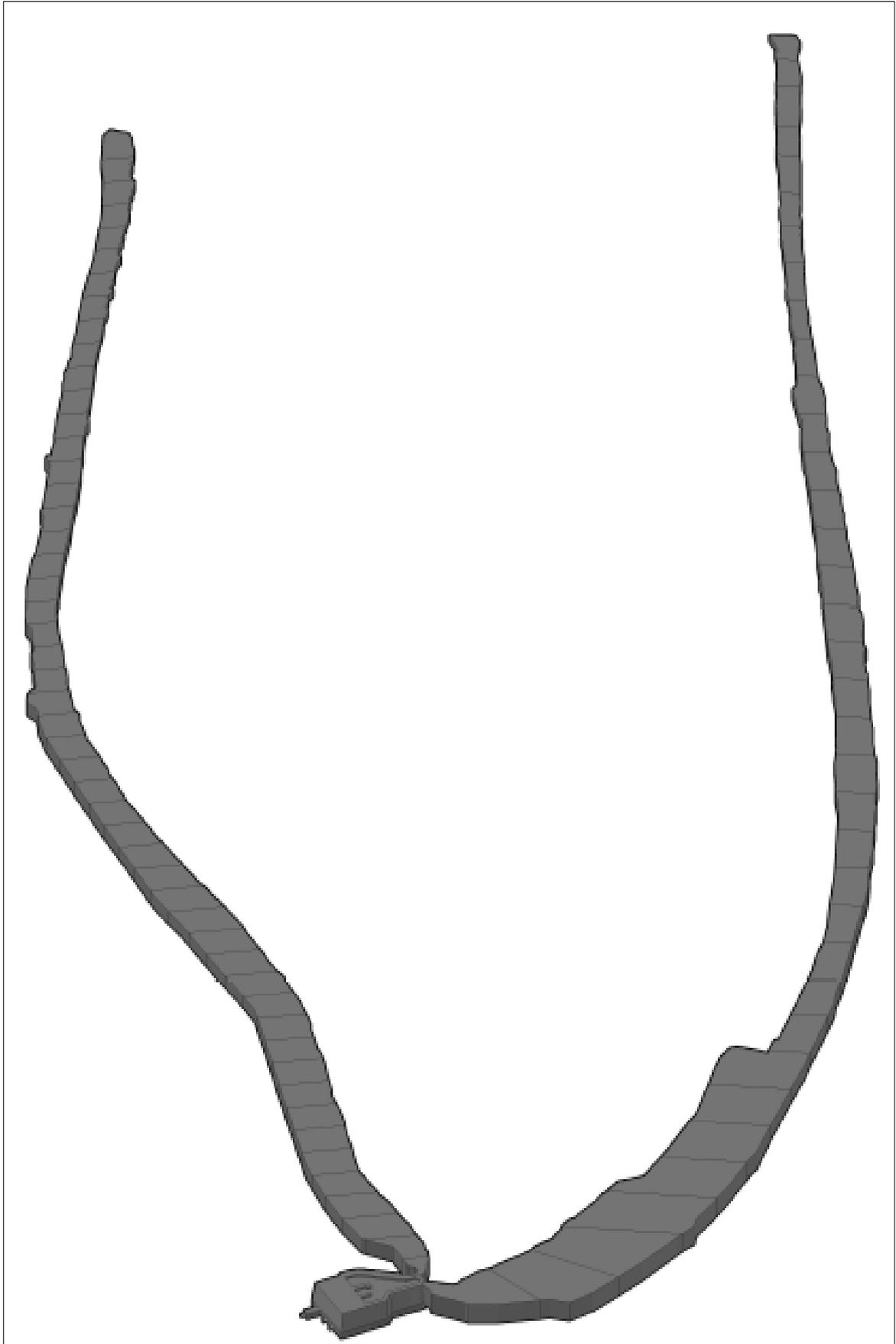


Figura A3: Modelado 3D de los tanques de la central hidroeléctrica Illuchi 1

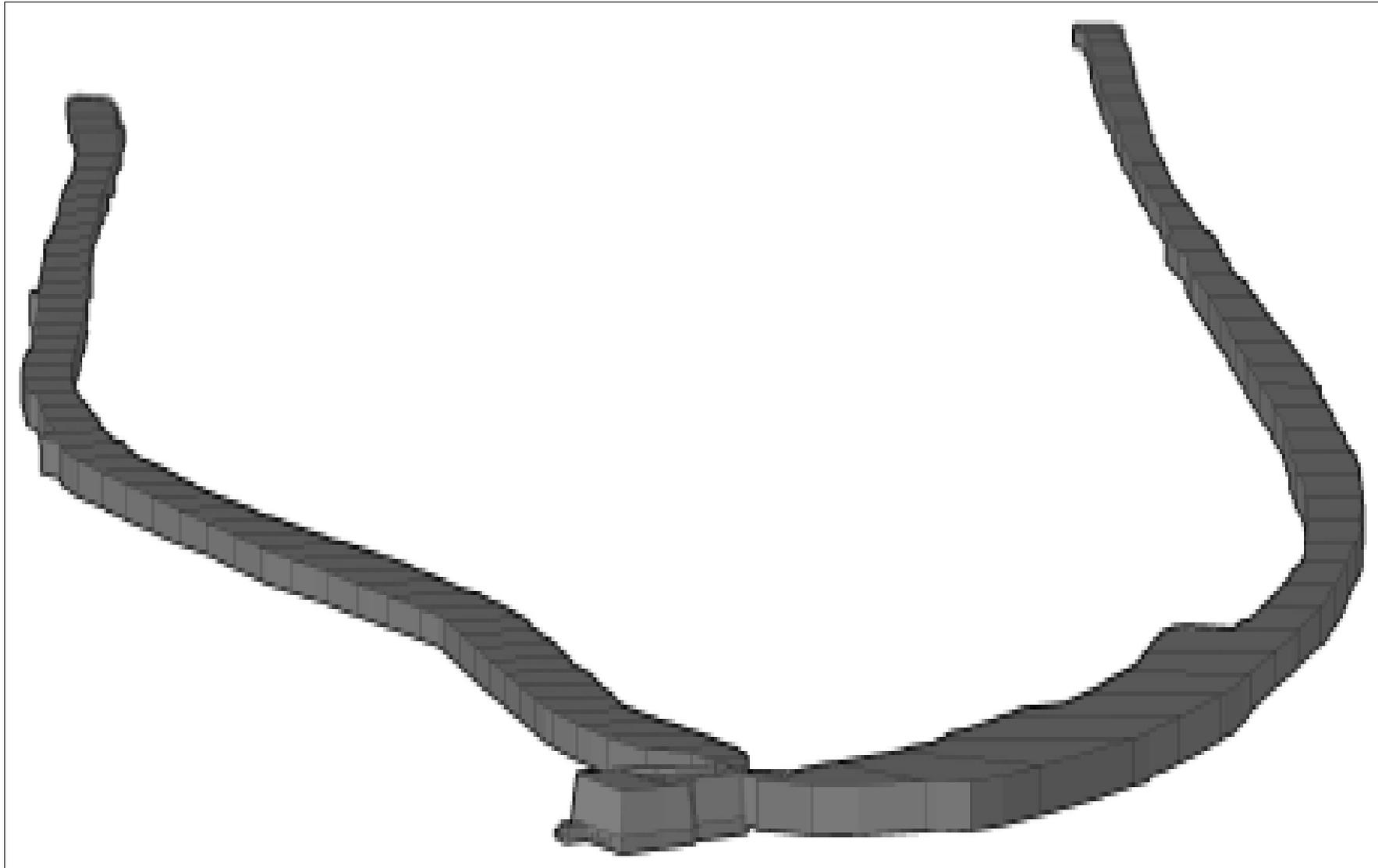


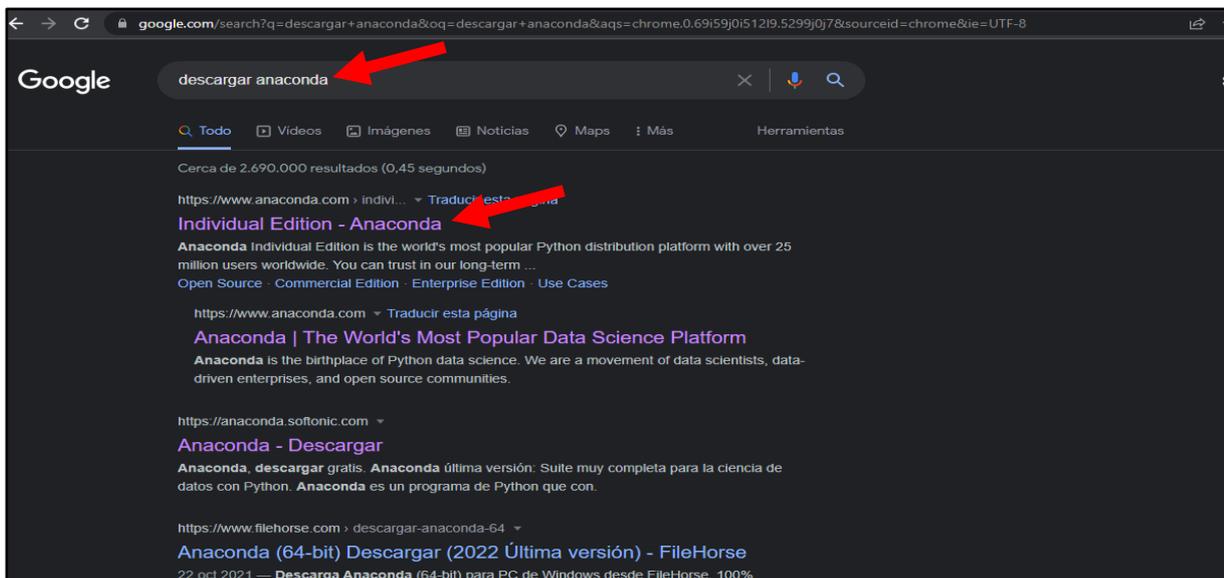
Figura A4: Modelado 3D de los tanques de la central hidroeléctrica Illuchi 1

EMPRESA ELECTRICA PROVINCIAL COTOPAXI S.A.																						
REPORTE DIARIO DE GENERACION																						
CENTRAL HIDROELECTRICA ILLUCHI No.1																						
HORAS	GENERADOR No.1					GENERADOR No.2					GENERADOR No.3					GENERADOR No.4					V <sub>φ</sub> (V)	Cos φ
	I (A)	P (KW)	E <sub>L</sub> (KW-H)	E <sub>G</sub> (KW-H)	Q (KWAR)	I (A)	P (KW)	E <sub>L</sub> (KW-H)	E <sub>G</sub> (KW-H)	Q (KWAR)	I (A)	P (KW)	E <sub>L</sub> (KW-H)	E <sub>G</sub> (KW-H)	Q (KWAR)	I (A)	P (KW)	E <sub>L</sub> (KW-H)	E <sub>G</sub> (KW-H)	Q (KWAR)		
0:00			5.546.100	0				6.405.800	0				1.158.700	0				9.369.100	0			
0:30				0	0	110	500			71	240	1.000			142	280	1.200			171	2.590	0,99
1:00			5.546.100	0	0	110	500	6.406.300	500	71	240	1.000	1.159.700	1.000	142	280	1.200	9.370.300	1.200	171	2.590	0,99
1:30				0	0	110	500			71	180	700			100	190	800			114	2.590	0,99
2:00			5.546.100	0	0	110	500	6.406.800	500	71	180	700	1.160.400	700	100	190	800	9.371.100	800	114	2.590	0,99
2:30				0	0	110	500			71	180	700			100	190	800			114	2.590	0,99
3:00			5.546.100	0	0	110	500	6.407.300	500	71	180	700	1.161.100	700	100	190	800	9.371.900	800	114	2.590	0,99
3:30				0	0	110	500			71	180	700			100	190	800			114	2.590	0,99
4:00			5.546.100	0	0	110	500	6.407.800	500	71	180	700	1.161.800	700	100	190	800	9.372.700	800	114	2.590	0,99
4:30				0	0	110	500			71	180	700			100	190	800			114	2.590	0,99
5:00			5.546.100	0	0	110	500	6.408.300	500	71	180	700	1.162.500	700	100	190	800	9.373.500	800	114	2.590	0,99
5:30				0	0	110	500			71	180	700			100	190	800			114	2.590	0,99
6:00			5.546.100	0	0	110	500	6.408.800	500	71	180	700	1.163.200	700	100	190	800	9.374.300	800	114	2.590	0,99
6:30				0	0	110	500			71	180	700			100	190	800			114	2.590	0,99
7:00			5.546.100	0	0	110	500	6.409.300	500	71	180	700	1.163.900	700	100	190	800	9.375.100	800	114	2.590	0,99
7:30				0	0	110	500			71	180	700			100	190	800			114	2.590	0,99
8:00			5.546.100	0	0	110	500	6.409.800	500	71	180	700	1.164.600	700	100	190	800	9.375.900	800	114	2.590	0,99
8:30				0	0	110	500			71	280	1.000			142	190	800			114	2.590	0,99
9:00			5.546.100	0	0	110	500	6.410.300	500	71	280	1.000	1.165.600	1.000	142	190	800	9.376.700	800	114	2.590	0,99
9:30				0	0	110	500			71	280	1.000			142	190	800			114	2.590	0,99
10:00			5.546.100	0	0	110	500	6.410.800	500	71	280	1.000	1.166.600	1.000	142	190	800	9.377.500	800	114	2.590	0,99
10:30				0	0	110	500			71	280	1.000			142	190	800			114	2.590	0,99
11:00			5.546.100	0	0	110	500	6.411.300	500	71	280	1.000	1.167.600	1.000	142	190	800	9.378.300	800	114	2.590	0,99
11:30				0	0	110	500			71	280	1.000			142	190	800			114	2.590	0,99

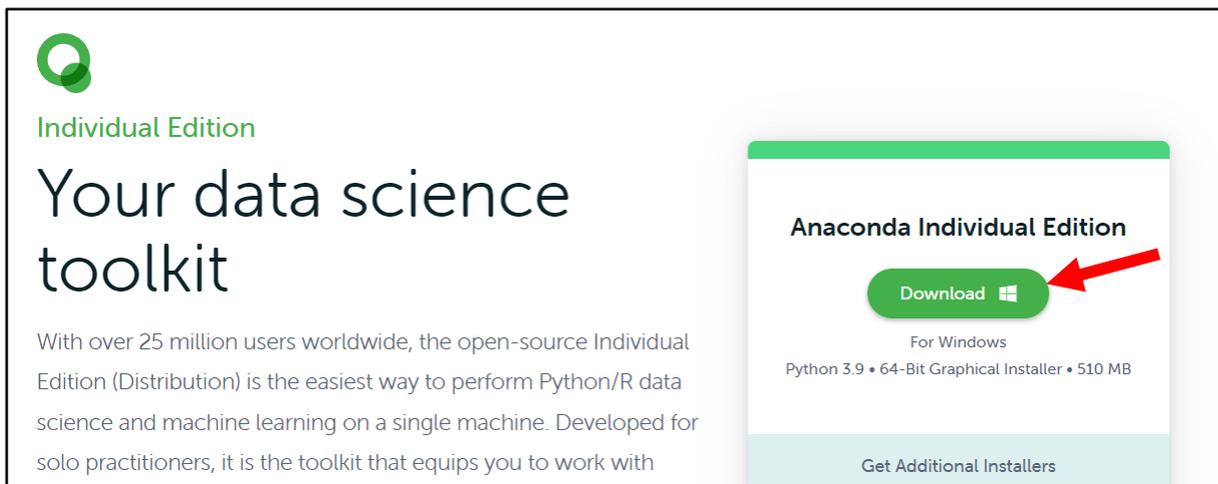
Figura A5: Datos históricos de potencia de la central hidroeléctrica Illuchi 1

## A. Manual de como descargar-instalar Anaconda (Python)

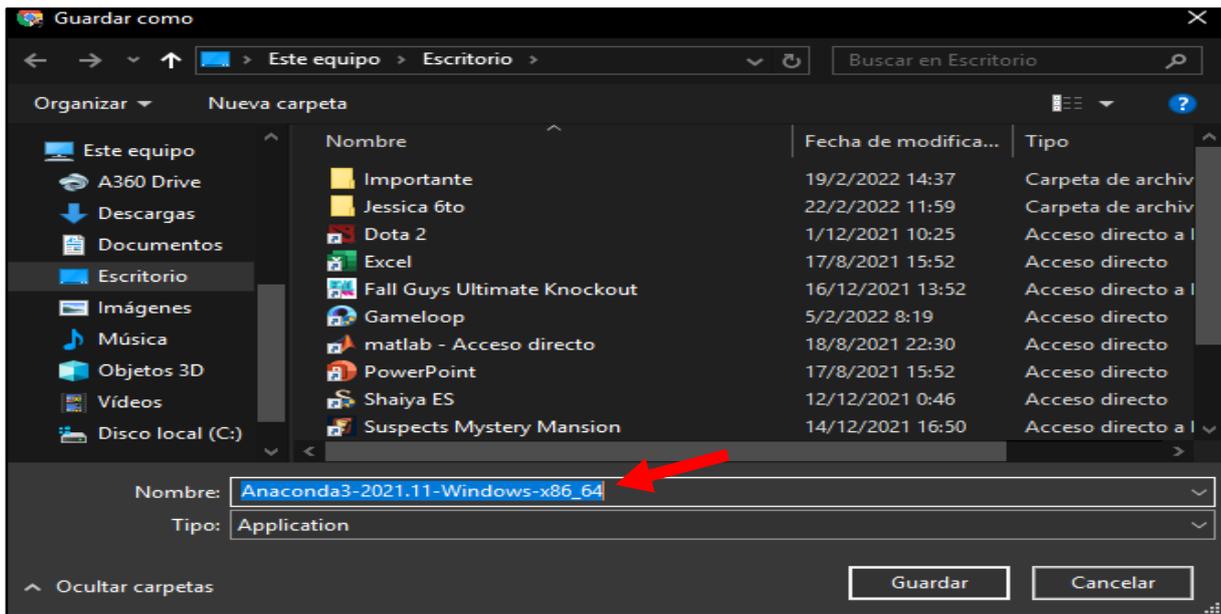
Como primer punto se tiene que entrar a Google y poner en el buscador “descargar Anaconda”, a continuación, ingresar en el primer link que aparece.



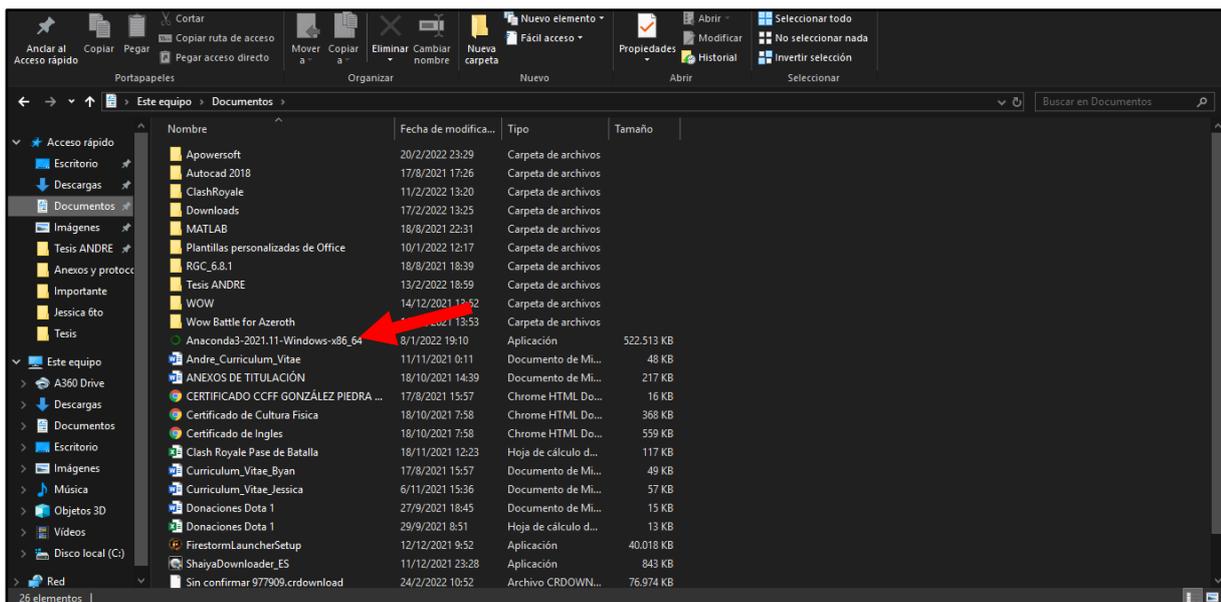
Una vez ingresado al link, presionar en “Download”.



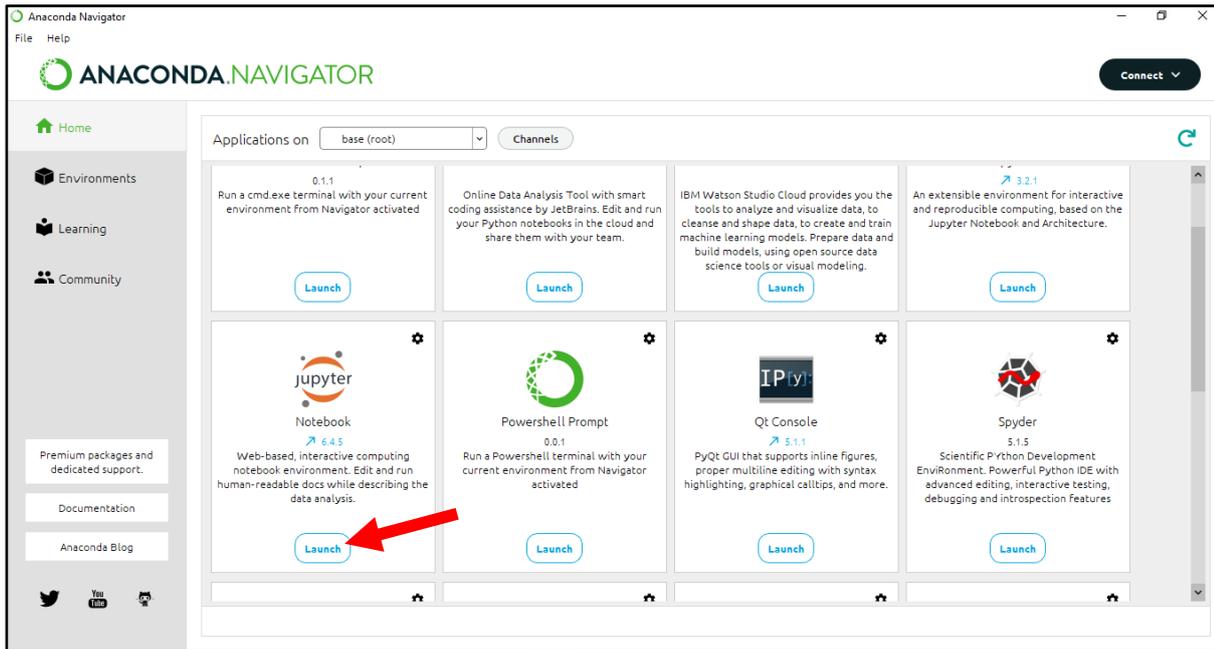
Elegir el lugar en donde guardar la descargar.



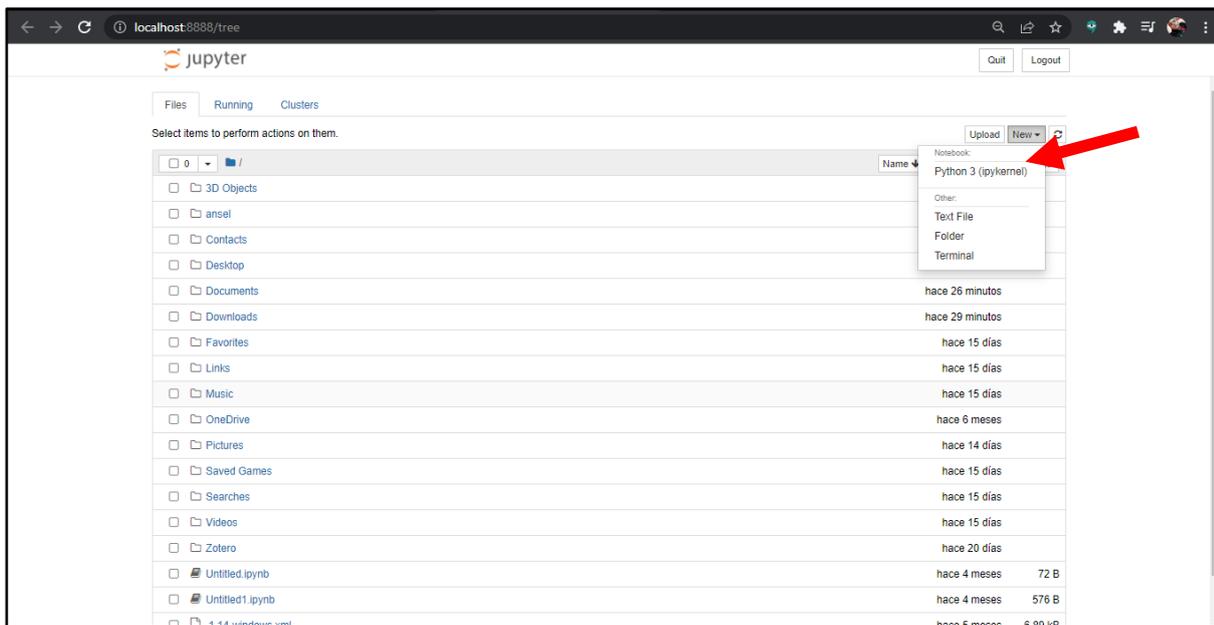
Una vez descargado el programa, hacer click derecho y ejecutar como administrador, para empezar con la instalación.



Una vez instalado el programa, abrir jupyter.



Crear un nuevo archivo en Python 3, para proceder con la programación de la red neuronal.



## B. Implementacion de la red neuronal en Python

```
In [1]: import pandas as pd
import numpy as np
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
```

### Procesamiento de Datos

```
In [4]: datos = pd.read_excel(io='data/datos.xlsx', sheet_name = "Hoja1")
datos
```

```
Out[4]:
```

	AÑO	MES	DIA	FECHA	HORA	Caudal
0	2015	1	1	jueves	00:00:00	1100.7567
1	2015	1	1	jueves	00:30:00	1100.7567
2	2015	1	1	jueves	01:00:00	1100.7567
3	2015	1	1	jueves	01:30:00	742.7767
4	2015	1	1	jueves	02:00:00	742.7767
...	...	...	...	...	...	...
81784	2019	12	31	martes	21:30:00	909.7567
81785	2019	12	31	martes	22:00:00	909.7567
81786	2019	12	31	martes	22:30:00	909.7567
81787	2019	12	31	martes	23:00:00	909.7567
81788	2019	12	31	martes	23:30:00	909.7567

81789 rows × 6 columns

### Verificación de valores nulos

```
In [5]: datos.isnull().sum()
```

```
Out[5]: Año      0
MES      0
DIA      0
FECHA    0
HORA     0
Caudal   0
dtype: int64
```

### Fecha como índice

```
In [6]: #Llevamos la hora y la fecha para los índices, por lo que nos interesa es el caudal
from datetime import datetime
datos_indexados = pd.DataFrame()
fechas_dt = []
caudal_dt = []
for registro in datos.values:
    fecha = "{}-{}-{}".format(registro[0], registro[1], registro[2], registro[4])
    fecha_dt = datetime.strptime(fecha, '%Y-%m-%d %H:%M:%S')
    fechas_dt.append(fecha_dt)
    caudal_dt.append(registro[5])
#Creamos una nueva DataFrame con el valor correcto de fecha y caudal
datos_indexados['FECHA'] = fechas_dt
datos_indexados['CAUDAL'] = caudal_dt

datos_indexados.set_index('FECHA', drop=True, inplace = True)

datos_indexados
```

## División de los Datos

```
In [7]: #Dividir Los datos de entrenamiento y de test
division = int(0.2*datos_indexados.shape[0])
fecha_fin_train = '2019-04-01 00:00:00'
datos_train = datos_indexados[:datos_indexados.index.get_loc(fecha_fin_train)[0]]
datos_test = datos_indexados[datos_indexados.index.get_loc(fecha_fin_train)[0]:]
datos_train
```

Out[7]:

	CAUDAL
FECHA	
2015-01-01 00:00:00	1100.7567
2015-01-01 00:30:00	1100.7567
2015-01-01 01:00:00	1100.7567
2015-01-01 01:30:00	742.7767
2015-01-01 02:00:00	742.7767
...	...
2019-03-31 21:30:00	711.1067
2019-03-31 22:00:00	711.1067
2019-03-31 22:30:00	711.1067
2019-03-31 23:00:00	711.1067
2019-03-31 23:30:00	711.1067

68589 rows × 1 columns

## Transformación de datos a numérica (CONFIRM)

```
In [9]: # (data_train[0:1].values)
```

## Normalización de los datos

```
In [10]: from sklearn.preprocessing import MinMaxScaler
sc = MinMaxScaler(feature_range = (0,1))
datos_train_scaled = sc.fit_transform(datos_train)
print(datos_train_scaled.shape)
datos_train_scaled
```

(68589, 1)

```
Out[10]: array([[0.70148996],
 [0.70148996],
 [0.70148996],
 ...,
 [0.49749754],
 [0.49749754],
 [0.49749754]])
```

## Agrupamiento de los datos

```
In [11]: #Se debe entrenar La red por bloques de datos
time_step = 70 #Se utiliza en time_step para poder entrenar por bloques de datos consecutivos
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.shape, Y_train.shape
```

Out[11]: ((68519, 70), (68519,))

## LSTM y Train

```
In [13]: from keras.layers import LSTM
         from sklearn.metrics import mean_squared_error
```

```
In [2]: def MODEL_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(Dropout(0.2))
         model.add(Dense(units = dim_salida))
         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 = 64)
         return model, history
```

```
In [15]: def predecirDataTest(modelo):
         x_test = datos_test.values
         X_test = sc.transform(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
```

## Ejecución del modelo

```
In [ ]: model, history = MODEL_LSTM(optimizador = 'rmsprop', neuronas = 10, epocas = 10)
         #Guardamos el modelo para no tener que entrenar otra vez para probar
         model.save('modelos/model.h5')
```

```
Epoch 1/10
1071/1071 [=====] - 72s 63ms/step - loss: 0.0139 - mean_absolute_error: 0.0842
Epoch 2/10
1071/1071 [=====] - 64s 60ms/step - loss: 0.0040 - mean_absolute_error: 0.0443
Epoch 3/10
1071/1071 [=====] - 67s 62ms/step - loss: 0.0029 - mean_absolute_error: 0.0363
Epoch 4/10
1071/1071 [=====] - 62s 58ms/step - loss: 0.0027 - mean_absolute_error: 0.0343
Epoch 5/10
1071/1071 [=====] - 63s 59ms/step - loss: 0.0026 - mean_absolute_error: 0.0336
Epoch 6/10
1071/1071 [=====] - 70s 65ms/step - loss: 0.0025 - mean_absolute_error: 0.0331
Epoch 7/10
1071/1071 [=====] - 70s 65ms/step - loss: 0.0025 - mean_absolute_error: 0.0328
Epoch 8/10
1071/1071 [=====] - 70s 65ms/step - loss: 0.0025 - mean_absolute_error: 0.0326
Epoch 9/10
1071/1071 [=====] - 61s 57ms/step - loss: 0.0024 - mean_absolute_error: 0.0324
Epoch 10/10
1071/1071 [=====] - 61s 57ms/step - loss: 0.0024 - mean_absolute_error: 0.0322
```

```
In [ ]: plt.plot(datos_test.values[time_step:(time_step + 7*48)], color='red', label='Caudal real')
plt.plot(prediccion[:7*48], color='blue', label='Predicción de Caudal')
plt.xlabel('Tiempo')
plt.ylabel('Caudal')
plt.legend()
plt.show()
```

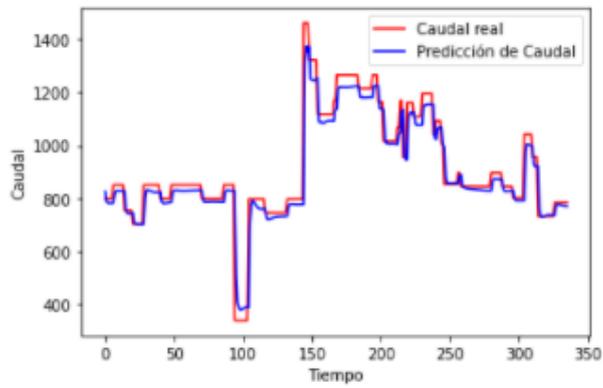


Figura A6: Código de la programación de la red neuronal artificial

### **C. Procedimiento para ejecutar la programación con una base de datos nueva**

Para poder cargar una nueva base de datos a la programación previamente es necesario realizar una depuración de la misma.

Como siguiente paso se tiene que seleccionar las variables de entrada en formato xlsx (cabe mencionar que si se elige unas variables diferentes es necesario cambiarlas también en la programación).

Una vez seleccionadas las variables de entrada se puede hacer el llamado a la base de datos en formato xlsx o csv para continuar con la programación.

Finalmente, seguir los pasos del literal B (Implementación de la red neuronal en Python) para cumplir con el desarrollo de la red neuronal artificial.

### D. Base de datos históricos con su respectivo caudal y generador

Hora	Día	Mes	Año	Potencia G1 (KW-H)	Caudal G1 (l/s)	Potencia G2 (KW-H)	Caudal G2 (l/s)	Potencia G3 (KW-H)	Caudal G3 (l/s)	Potencia G4 (KW-H)	Caudal G4 (l/s)	Potencia total (KW-H)	Caudal total (l/s)
0:30	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	1.200	496,81	2.700	1100,7567
1:00	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	1.200	496,81	2.700	1100,7567
1:30	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
2:00	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
2:30	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
3:00	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
3:30	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
4:00	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
4:30	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
5:00	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
5:30	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
6:00	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
6:30	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
7:00	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
7:30	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
8:00	1	1	2015	0	-7,315	500	216,7317	700	241,11	800	292,25	2.000	742,7767
8:30	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
9:00	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
9:30	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
10:00	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
10:30	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
11:00	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
11:30	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
12:00	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
12:30	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
13:00	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
13:30	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
14:00	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
14:30	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
15:00	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
15:30	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
16:00	1	1	2015	0	-7,315	500	216,7317	1.000	394,53	800	292,25	2.300	896,1967
16:30	1	1	2015	0	-7,315	400	173,7617	800	292,25	800	292,25	2.000	750,9467
17:00	1	1	2015	0	-7,315	400	173,7617	800	292,25	800	292,25	2.000	750,9467
17:30	1	1	2015	0	-7,315	400	173,7617	800	292,25	800	292,25	2.000	750,9467
18:00	1	1	2015	0	-7,315	400	173,7617	800	292,25	800	292,25	2.000	750,9467
18:30	1	1	2015	0	-7,315	400	173,7617	800	292,25	800	292,25	2.000	750,9467
19:00	1	1	2015	0	-7,315	400	173,7617	800	292,25	800	292,25	2.000	750,9467





15:30	3	1	2015	0	-7,315	400	173,7617	800	292,25	800	599,09	2.000	1057,7867
16:00	3	1	2015	0	-7,315	400	173,7617	800	292,25	800	599,09	2.000	1057,7867
16:30	3	1	2015	0	-7,315	400	173,7617	1.100	445,67	800	599,09	2.300	1211,2067
17:00	3	1	2015	0	-7,315	400	173,7617	1.100	445,67	800	599,09	2.300	1211,2067
17:30	3	1	2015	0	-7,315	400	173,7617	1.100	445,67	800	599,09	2.300	1211,2067
18:00	3	1	2015	0	-7,315	400	173,7617	1.100	445,67	800	599,09	2.300	1211,2067
18:30	3	1	2015	0	-7,315	400	173,7617	1.100	445,67	800	599,09	2.300	1211,2067
19:00	3	1	2015	0	-7,315	400	173,7617	1.100	445,67	800	599,09	2.300	1211,2067
19:30	3	1	2015	0	-7,315	400	173,7617	700	241,11	800	599,09	1.900	1006,6467
20:00	3	1	2015	0	-7,315	400	173,7617	700	241,11	800	599,09	1.900	1006,6467
20:30	3	1	2015	0	-7,315	400	173,7617	700	241,11	800	394,53	1.900	802,0867
21:00	3	1	2015	0	-7,315	400	173,7617	700	241,11	800	394,53	1.900	802,0867
21:30	3	1	2015	0	-7,315	400	173,7617	700	241,11	800	394,53	1.900	802,0867
22:00	3	1	2015	0	-7,315	400	173,7617	700	241,11	800	394,53	1.900	802,0867
22:30	3	1	2015	0	-7,315	400	173,7617	700	241,11	800	394,53	1.900	802,0867
23:00	3	1	2015	0	-7,315	400	173,7617	700	241,11	800	394,53	1.900	802,0867
23:30	3	1	2015	0	-7,315	400	173,7617	900	343,39	800	394,53	2.100	904,3667
24:00	3	1	2015	0	-7,315	400	173,7617	900	343,39	800	394,53	2.100	904,3667
0:30	4	1	2015	0	-7,315	400	173,7617	900	343,39	800	394,53	2.100	904,3667
1:00	4	1	2015	0	-7,315	400	173,7617	900	343,39	800	394,53	2.100	904,3667
1:30	4	1	2015	0	-7,315	400	173,7617	900	343,39	800	547,95	2.100	1057,7867
2:00	4	1	2015	0	-7,315	400	173,7617	900	343,39	800	547,95	2.100	1057,7867
2:30	4	1	2015	0	-7,315	400	173,7617	900	343,39	800	547,95	2.100	1057,7867
3:00	4	1	2015	0	-7,315	400	173,7617	900	343,39	800	547,95	2.100	1057,7867
3:30	4	1	2015	0	-7,315	400	173,7617	900	343,39	800	547,95	2.100	1057,7867
4:00	4	1	2015	0	-7,315	400	173,7617	900	343,39	800	547,95	2.100	1057,7867
4:30	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	547,95	2.000	1006,6467
5:00	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	547,95	2.000	1006,6467
5:30	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	599,09	2.000	1057,7867
6:00	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	599,09	2.000	1057,7867
6:30	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	599,09	2.000	1057,7867
7:00	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	599,09	2.000	1057,7867
7:30	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
8:00	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
8:30	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
9:00	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
9:30	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
10:00	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
10:30	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
11:00	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
11:30	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	292,25	2.000	750,9467
12:00	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	292,25	2.000	750,9467
12:30	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	292,25	2.000	750,9467
13:00	4	1	2015	0	-7,315	400	173,7617	800	292,25	800	292,25	2.000	750,9467



11:30	5	1	2015	0	-7,315	600	259,7017	1.100	445,67	1.200	292,25	2.900	990,3067
12:00	5	1	2015	0	-7,315	600	259,7017	1.100	445,67	1.200	292,25	2.900	990,3067
12:30	5	1	2015	0	-7,315	600	259,7017	1.100	445,67	1.300	292,25	3.000	990,3067
13:00	5	1	2015	0	-7,315	600	259,7017	1.100	445,67	1.300	292,25	3.000	990,3067
13:30	5	1	2015	0	-7,315	600	259,7017	1.100	445,67	1.300	445,67	3.000	1143,7267
14:00	5	1	2015	0	-7,315	600	259,7017	1.100	445,67	1.300	445,67	3.000	1143,7267
14:30	5	1	2015	0	-7,315	500	216,7317	1.000	394,53	1.200	599,09	2.700	1203,0367
15:00	5	1	2015	0	-7,315	500	216,7317	1.000	394,53	1.200	599,09	2.700	1203,0367
15:30	5	1	2015	0	-7,315	500	216,7317	900	343,39	1.000	599,09	2.400	1151,8967
16:00	5	1	2015	0	-7,315	500	216,7317	900	343,39	1.000	599,09	2.400	1151,8967
16:30	5	1	2015	0	-7,315	500	216,7317	1.000	394,53	1.000	599,09	2.500	1203,0367
17:00	5	1	2015	0	-7,315	500	216,7317	1.000	394,53	1.000	599,09	2.500	1203,0367
17:30	5	1	2015	0	-7,315	500	216,7317	1.000	394,53	1.000	599,09	2.500	1203,0367
18:00	5	1	2015	0	-7,315	500	216,7317	1.000	394,53	1.000	599,09	2.500	1203,0367
18:30	5	1	2015	0	-7,315	600	259,7017	1.000	394,53	1.400	599,09	3.000	1246,0067
19:00	5	1	2015	0	-7,315	600	259,7017	1.000	394,53	1.400	599,09	3.000	1246,0067
19:30	5	1	2015	0	-7,315	600	259,7017	1.000	394,53	1.400	599,09	3.000	1246,0067
20:00	5	1	2015	0	-7,315	600	259,7017	1.000	394,53	1.400	599,09	3.000	1246,0067
20:30	5	1	2015	0	-7,315	600	259,7017	1.000	394,53	1.400	394,53	3.000	1041,4467
21:00	5	1	2015	0	-7,315	600	259,7017	1.000	394,53	1.400	394,53	3.000	1041,4467
21:30	5	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
22:00	5	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
22:30	5	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
23:00	5	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
23:30	5	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267
24:00	5	1	2015	0	-7,315	400	173,7617	800	292,25	800	394,53	2.000	853,2267

## E. Análisis del método ARIMA

Para aplicar el método ARIMA en la presente investigación sería necesario realizar todo el análisis y programación nuevamente por lo que se decidió hacer la comparación de la confiabilidad con otro ejemplo.

### Ejemplo:

Los datos con que se elaboró este estudio corresponden a la demanda total de electricidad del sistema interconectado colombiano, en miles de GWh mensuales, entre 1 995:8 y 2 008:6, los cuales están disponibles en el sistema Neón1 y recogen las características históricas desde la creación del mercado mayorista de electricidad.

### Resultados del modelo

En esta sección se presentan los resultados obtenidos al pronosticar el logaritmo natural de la demanda mensual de electricidad en miles de GWh para el periodo comprendido entre 1 995:8 y 2 008:6 usando un modelo ARIMA. La serie consta de 155 datos; los primeros 131 (entre 1 995:8 y 2 006:6) fueron usados para estimar los parámetros de los modelos; los 24 restantes se usaron para evaluar su capacidad de predicción. Con el fin de analizar la influencia del horizonte de predicción sobre los resultados obtenidos, se consideraron horizontes de 12 (entre 2 006:07 y 2 007:06) y 24 (entre 2 006:7 y 2 008:6) meses, respectivamente.

Debido a que la serie estudiada presenta una clara componente de tendencia y un ciclo de periodicidad anual, se aplicó una diferenciación simple y una diferenciación estacional de periodo 12 al logaritmo de la demanda de electricidad. De esta forma, los parámetros de los modelos y los estadísticos de ajuste fueron obtenidos para la serie  $W_t$  que fue obtenida en la ecuación (E.1):

$$W_t = (1 - B)(1 - B^{12})\log(d_t) \quad (\text{E.1})$$

Donde  $d_t$  es la serie de demanda mensual en miles de GWh,  $B$  es el operador de diferenciación, por definición  $By_t = y_{t-1}$ ; en consecuencia,  $B^{12}y_t = y_{t-12}$ .

La bondad del ajuste de los modelos a las muestras de calibración y predicción fue medida mediante la sumatoria del error cuadrático medio (SSE, por sus siglas en inglés) de los residuales,  $et$ , y su desviación media absoluta (MAD, obtenida en (E.2)):

$$MAD = \frac{1}{R} \sum_{t=1}^R |et| \quad (\text{E.2})$$

